

N° d'ordre : 830

50376

1991

234



THESE

65493

50376

1991

234

présentée à

L'UNIVERSITE DES SCIENCES ET TECHNIQUES DE LILLE FLANDRES ARTOIS

en vue de l'obtention du grade de

DOCTEUR DE L'UNIVERSITE

Spécialité : Productique,
Automatique et Informatique Industrielle

par

Denis POMORSKI

Maître E.E.A.

APPRENTISSAGE AUTOMATIQUE
SYMBOLIQUE/NUMERIQUE

-

CONSTRUCTION ET EVALUATION
D'UN ENSEMBLE DE REGLES
A PARTIR DES DONNEES

Soutenue le 6 décembre 1991 devant la Commission d'Examen

Membres du Jury :

MM.	R. HANUS	Président
	B. DUBUISSON	Rapporteur
	P. MILLOT	Rapporteur
	M. STAROSWIECKI	Directeur de Recherche
Mme	M.C. CHAMBRIN	Examinateur

AVANT-PROPOS

Ce travail a été réalisé au Laboratoire d'Automatique et d'Informatique Industrielle de Lille (LAIL - URA 1440), grâce au soutien financier apporté par la Région "Nord - Pas-de-Calais".

Je remercie Monsieur Raymond HANUS, Professeur à l'Université Libre de Bruxelles, pour l'honneur qu'il me fait de présider le jury de cette thèse.

Je tiens à exprimer ma plus profonde gratitude à Monsieur Marcel STAROSWIECKI, Professeur à l'Ecole Universitaire D'Ingénieurs de Lille, qui a dirigé ce travail, et dont les conseils et la rigueur scientifiques ont été pour l'essentiel dans les résultats de ce travail. Conscient de ce que je lui dois, je le remercie très sincèrement.

Mes remerciements vont aussi à Monsieur Bernard DUBUISSON, Professeur à l'Université Technologique de Compiègne, et à Monsieur Patrick MILLOT, Professeur à l'Université de Valenciennes, qui ont bien voulu juger ce travail, et faire partie du jury.

Je remercie également Madame Marie-Christine CHAMBRIN, attachée de recherche à l'INSERM, pour sa participation au jury.

Enfin, je tiens à remercier tous ceux qui, de par leur compétence en la matière, ou de par leur amitié, dans des moments parfois difficiles, m'ont soutenu dans la réalisation de ce travail.

SOMMAIRE

INTRODUCTION GENERALE	1
Chapitre I : ANALYSE STRUCTURALE DES SYSTEMES	9
I. Hiérarchie épistémologique des systèmes	10
I.1. Niveau 0 : Système-Source	
I.2. Niveau 1 : Système-Données	
I.3. Niveau 2 : Système-Générateur	
I.4. Du niveau 1 vers un niveau supérieur	
II. Représentation globale : structure des données	16
II.1. Les variables primaires	
II.2. Finesse d'une variable primaire	
II.3. Variables primaires et partitions de Ω	
II.4. Variables vectorielles	
II.5. Observabilité des variables	
III. Représentation locale : le langage VL	24
III.1. L'évènement élémentaire	
III.2. L'objet assertion	
III.2.1. Définition	
III.2.2. Structure de l'ensemble des assertions	
III.3. L'objet horde	
III.4. Intérêt du langage de représentation choisi	
IV. Apprentissage à partir des données	28
IV.1. Variables à expliquer - Variables explicatives	
IV.2. Relations entre variables	
IV.2.1. Comparaison des partitions de Ω induites par deux variables	
IV.2.2. Génération de règles	
IV.3. Conclusion	

Chapitre II :	LES OUTILS DE L'APPRENTISSAGE AUTOMATIQUE APPLIQUE AU PROBLEME DE L'EXPLICATION	38
I.	Introduction	39
II.	Faisabilité du problème d'explication	40
II.1.	Introduction à la théorie de l'information	
II.2.	Définition de l'entropie	
II.3.	L'entropie de Shannon	
II.3.1.	Implications directes de la définition	
II.3.2.	Entropie d'un système composé	
II.3.3.	Entropie conditionnelle	
II.3.4.	Entropie et Information	
II.3.4.1.	Position du problème	
II.3.4.2.	Transinformation interne	
II.3.4.3.	Transinformation externe	
II.3.4.4.	Propriétés sur l'entropie et les transinformations	
II.4.	Application au problème de la visualisation	
II.5.	Application au problème de la structuration : Recherche de partitions optimales	
II.5.1.	Minimisation de la transinformation externe [CONA 72]	
II.5.2.	Décompositions optimales [TORO 82]	
II.6.	Application au problème de l'explication	
II.6.1.	Détermination de quelques indices [TORO 82]	
II.6.2.	$H(Y/X)$: intensité de liaison entre X et Y	
III.	Les méthodes d'induction par arbre de décision	51
III.1.	Position du problème et critères	
III.2.	Le système ID3	
III.3.	Remarques	
IV.	Les méthodes d'Apprentissage Symbolique	57
IV.1.	Position du problème et définitions	
IV.2.	Le système INDUCE	
IV.3.	Le système GENREG	
IV.4.	Les autres systèmes	
V.	Conclusion	64

Chapitre III : EVALUATION DU MODELE	
OBTENU PAR APPRENTISSAGE AUTOMATIQUE	66
I. Position du problème	67
I.1. Règles toujours vraies et dépendance perçue	
I.2. La procédure d'apprentissage	
II. Propriétés sur un ensemble de règles	70
II.1. Complétude sur un ensemble de variables	
II.2. Cohérence	
II.2.1. Cohérence logique	
II.2.2. Cohérence perçue	
II.3. Classification des modèles obtenus par la procédure d'apprentissage	
III. Evaluation des règles	77
III.1. Relations d'ordre	
III.2. Indices caractéristiques (critères locaux)	
III.2.1. Taux de couverture d'une règle	
III.2.2. Taux de pollution d'une règle	
IV. Généralisation (induction) sur un ensemble de règles	83
IV.1. Règles de généralisation	
IV.1.1. Approche globale	
IV.1.2. Approche locale	
IV.2. Agrégation de deux règles	
IV.3. Prolongement d'une règle	
V. Evaluation de la généralisation (de l'induction)	88
V.1. Evaluation de l'agrégation	
V.2. Evaluation du prolongement	
VI. Conclusion	91
 Chapitre IV : CONSTRUCTION D'UN MODELE	
(EXTRACTION DE REGLES)	92
I. Position du problème	93
II. Construction d'un modèle atomique	93
II.1. Définitions	
II.2. Opérateur de contraste	
II.3. Utilité de l'information perdue - Coefficient de vérité	
II.4. Efficacité d'une règle	
II.5. Modèle atomique	

III. Méthodes d'induction suivant un critère entropique	99
III.1. Position du problème	
III.2. Cas de variables explicatives non ordonnées	
III.2.1. Recherche exhaustive	
III.2.2. Recherche hiérarchique ascendante	
III.2.3. Recherche hiérarchique descendante	
III.2.4. Algorithme de transfert	
III.3. Cas de variables explicatives ordonnées	
III.3.1. Cas d'une variable explicative ordonnée ($p=1$)	
III.3.1.1. Algorithmes hiérarchiques et de transfert	
III.3.1.2. Utilisation d'une méthode d'optimisation	
III.3.1.2.1. Optimisation par programmation dynamique	
III.3.1.2.2. Méthode du gradient	
III.3.2. Cas de plusieurs variables explicatives ($p>1$)	
III.3.2.1. Position du problème	
III.3.2.2. Algorithme proposé	
IV. Méthodes d'induction symbolique	120
IV.1. Les données d'apprentissage	
IV.2. La procédure de généralisation	
CONCLUSIONS	123
ANNEXES	126
REFERENCES BIBLIOGRAPHIQUES	133

INTRODUCTION GENERALE

LE CADRE DE CE TRAVAIL ...

Ce travail s'inscrit dans le cadre du thème : *Analyse et Surveillance des Processus Industriels Complexes*, développé au Laboratoire d'Automatique et d'Informatique Industrielle de Lille (LAIL - URA 1440).

Il est justifié par la prise en compte de systèmes pour lesquels aucun modèle de comportement n'est disponible. Leur surveillance passe alors par l'établissement d'un modèle statistique, élaboré à partir de l'analyse des données relevées sur le processus.

Un modèle numérique précis n'a en général pas de sens dans ce contexte, c'est pourquoi une recherche orientée vers la détermination d'un modèle qualitatif a été entreprise, à partir de résultats antérieurs concernant l'utilisation de la *Théorie de l'Information en Analyse des Données*.

LA MODELISATION - SES PROBLEMES

La construction d'un modèle à partir des observations fait apparaître trois types de problèmes bien distincts [BOUC 90] :

- un problème de *visualisation*
- un problème de *structuration*
- un problème d'*explication*

(i) Le problème de *visualisation*

Les tableaux de données recueillis sur le système peuvent contenir plusieurs milliers d'observations effectuées sur plusieurs dizaines de variables. Le praticien se trouve alors en présence d'une masse de données difficilement interprétable. En effet, le nuage des observations décrites par deux variables est tout à fait visualisable sur un plan. Il est déjà plus difficile, pour l'homme, de se représenter un nuage d'observations décrites par trois variables (représentation dans l'espace). Pour quatre variables, et plus, une étude visuelle des données s'avère impossible.

On cherche alors une représentation de ce nuage dans un espace de dimensions plus petites.

Le problème de visualisation du nuage des observations, lorsque le nombre de variables prises en compte est important, consiste donc à le diminuer. On cherche dès lors les variables les plus importantes, les plus significatives pour notre étude. En d'autres termes, ce problème revient à trouver les variables contenant le plus d'information possible.

Deux façons d'opérer sont alors possibles :

→ On utilise les méthodes *factorielles* [FOUC 85] de l'Analyse des Données (problème de recherche de facteurs principaux posé par Thurstone dans les années 1930), qui consistent à trouver des combinaisons linéaires des variables initiales. Celles-ci sont cherchées de façon à ce que les nouvelles variables (encore appelées *facteurs principaux*) contiennent un maximum d'information.

En général, cela n'est pas possible sans une distorsion des mesures (distorsion que l'on cherche à minimiser).

Les deux principales méthodes d'analyse factorielle sont l'Analyse en Composantes Principales [HOTE 33] et l'Analyse Factorielle des Correspondances [BENZ 80] [CIBO 87] qui se distinguent l'une de l'autre par le type de variables qu'elles traitent, et donc, par le choix de la métrique utilisée.

→ On recherche directement les variables apportant la plus grande information, parmi les variables initiales prises en compte. Comme précédemment, la distorsion des mesures qui en résulte est due à la diminution des dimensions de l'espace de représentation.

De nombreuses méthodes statistiques existent (maximisation de l'inertie pour des variables quantitatives, du Khi^2 pour des variables qualitatives, ...) dans une littérature très abondante. Pour notre part, nous nous intéresserons à une méthode entropique basée sur la Théorie de l'Information.

Un compromis sera alors à faire entre, d'une part, la perte d'information occasionnée par la distorsion des mesures, et d'autre part, la simplicité de la représentation du nuage.

(ii) Le problème de structuration

Un système complexe est un ensemble de sous-systèmes (plus simples) pouvant interagir entre eux [DELA 71]. Il serait dès lors intéressant de décomposer ce système en systèmes faiblement couplés. C'est donc un problème de structuration (de décomposition, ou encore de classification automatique non supervisée*), qui consiste à munir la population d'apprentissage (l'ensemble des observations), ou l'ensemble des variables, d'une structure particulière. Diverses techniques de décomposition sont mises en place, suivant la structure désirée.

On distingue en général deux catégories de techniques de décomposition :

- les méthodes de coalescence floue
- les méthodes de classification automatique.

→ Les méthodes de coalescence floue sont développées dans [DUBU 90]. La théorie des sous-ensembles flous a été élaborée afin de traduire des règles du type "plus ... plus ...". En effet, dans la théorie des ensembles (dits vulgaires), un élément appartient, ou n'appartient pas à un ensemble, tandis que dans la théorie des sous-ensembles flous, l'appartenance d'un élément x à un ensemble E est caractérisée par sa fonction d'appartenance μ comprise entre 0 et 1.

Initiée par L.A. Zadeh [ZADE 65], cette théorie fût développée par [KAUF 73], [DUBO 79], Les premiers algorithmes de coalescence floue ont été développés par E.H. Ruspini [RUSP 69,70].

→ Le but des méthodes de classification [LERM 70] [JAMB 78] [CELE 89] (encore appelée *taxonomie*, ou *taxinomie*) est de regrouper les individus en un nombre restreint de classes homogènes. Ces classes sont obtenues au moyen d'algorithmes formalisés, et non pas par des méthodes subjectives ou visuelles faisant appel à l'initiative du praticien, d'où le terme "automatique".

Les méthodes les plus usitées sont des méthodes heuristiques : ce sont "des algorithmes dont on considère qu'ils sont suffisamment raisonnables pour donner des résultats satisfaisants" [ROUX 85].

* Cette classification est dite "non supervisée", dans le sens où on ne dispose d'aucune connaissance a priori (aucune observation) sur ces classes.

On distingue en général deux types de méthodes heuristiques :

- les méthodes de classification hiérarchique :
 - ascendante (agrégations successives d'individus, puis de groupes)
 - descendante (dichotomies successives)
- les méthodes métriques (ou de coalescence). Ce sont des méthodes d'élaboration directe d'une partition :
 - algorithme de *regroupement autour des centres mobiles* [FORG 65]
 - algorithme de J.B. Mac Queen [QUEE 67]
 - algorithme des *nuées dynamiques* [DID 72]

Des méthodes statistiques de classification existent également, par exemple la *méthode des Noyaux* de Rosenblatt-Parzen (méthode de détection des zones à forte densité dans l'espace des observations) [PARZ 60] [POST 87].

(iii) Le problème d'explication

Le but des méthodes d'explication est de rechercher des relations de dépendance entre variables. Nous distinguons alors, d'une part, la (ou les) variable(s) à *expliquer*, et d'autre part, la (ou les) variable(s) *explicative(s)*, ou considérée(s) comme telle(s).

Le problème d'explication consiste à rechercher des *procédures d'affectation* d'une observation quelconque : étant connues les modalités de certaines variables (variables explicatives), il s'agit d'expliquer (de prédire) les modalités des autres variables (variables à expliquer). On cherche donc à caractériser des *relations de dépendance* entre les premières et les secondes. Certains auteurs parlent de *classement* (en *Reconnaissance des Formes*), de *classification supervisée* (dans le sens où on connaît des représentants des diverses classes en présence), d'*apprentissage à partir d'exemples*, d'*induction* (en *Intelligence Artificielle*), ou de *modélisation*.

LES OUTILS - DU NUMERIQUE AU SYMBOLIQUE

Différentes approches ont été développées pour la modélisation à partir des données. Nous en donnons une présentation sommaire, suivant un ordre (plus ou moins) chronologique retraçant l'histoire des méthodes et des problèmes rencontrés.

Ces différentes méthodes suivent un axe (numérique—>symbolique). Ainsi, nous distinguons :

- les méthodes de la Statistique Classique
- les méthodes de l'Analyse des Données
- les méthodes de la Reconnaissance des Formes
- les méthodes de l'Apprentissage Symbolique
- et enfin, les méthodes de l'Apprentissage Symbolique/Numérique.

(*) Les méthodes de l'Analyse des Données (A.D.) :

C'est à la suite des travaux de K. Pearson [PEAR 01] et de C. Spearman [SPEA 04] que H. Hotelling posait les premiers fondements de l'A.D. : [HOTE 33] pour l'Analyse en Composantes Principales (A.C.P.), et [HOTE 36] pour l'Analyse Canonique. Ces méthodes traitent de données mises sous forme d'un tableau, et nécessitent, en général, une masse considérable de calculs. C'est donc avec l'avènement de l'ordinateur (dans les années 1960) qu'elles connurent leur véritable essor. Actuellement, il existe de nombreuses méthodes d'A.D. (et beaucoup de variantes) disponibles sous forme de logiciels, et décrites dans une littérature abondante.

E. Diday [DID 82] p.3, donne une définition de l'A.D. :

"L'analyse de données est un ensemble de méthodes dont le but essentiel est la mise en relief des relations existant entre les objets, entre les paramètres qui les caractérisent et entre les objets et les paramètres. Fréquemment, une analyse de données consistera surtout à situer globalement ces objets les uns par rapport aux autres et à mesurer l'importance relative de chacun des paramètres dans cette répartition".

(**) Les méthodes de la Reconnaissance des Formes (RdF) :

B. Dubuisson [DUBU 90], p.25, donne une définition de la RdF :

"La reconnaissance des formes est une science de définition d'algorithmes permettant de classer des objets dont l'aspect a varié par rapport à un objet type".

Le problème traité par la RdF est alors le suivant : étant donnée une forme, il s'agira de déterminer à quelle forme-type (parmi m formes-type, ou prototypes, ou classes d'appartenance)

celle-ci appartient, ou encore, à quelle classe doit-on l'affecter ?

La RdF est donc un problème de *classement* (ou de *classification supervisée*) qui se distingue de la *classification non supervisée* dans le sens où, dans le deuxième cas, nous ne disposons pas de cette connaissance *a priori* qu'est l'existence des classes. Dans le premier cas, nous disposons d'au moins un exemple pour chaque classe d'appartenance. C'est à ce titre que nous distinguons les méthodes de RdF (problème de classement) des méthodes statistiques de classification non supervisée, telles que la méthode des *Noyaux* de Rosenblatt-Parzen.

Un problème de RdF demande donc, d'une part, la définition des différentes formes-type, et d'autre part, de choisir un espace de représentation. En général, suivant le choix de cet espace, on distingue l'approche *structurelle* (ou *syntactique* : les formes y sont représentées à l'aide de *grammaires* [FU 74] [MICH 84]), et l'approche *statistique*.

(***) *Les méthodes de l'Apprentissage Symbolique :*

Nous appellerons *Apprentissage Symbolique* la partie *Apprentissage à partir d'Exemples* (S.B.L. : Similarity Based Learning) [MICH 83b] de l'*Intelligence Artificielle* (I.A.). Cette branche de l'I.A. s'est développée depuis peu (fin des années 70 pour les premiers articles). R.S. Michalski, J.R. Quinlan, M. Lebowitz sont parmi les initiateurs de cette discipline. Le principe de la S.B.L. est l'apprentissage de similarités entre exemples (problème d'induction) d'une part, et l'apprentissage de dissimilarités entre exemples et contre-exemples (problème de discrimination) d'autre part. Cette partie est largement traitée dans [KOD 86].

Nous décrirons le système d'apprentissage *INDUCE* (descendant immédiat de AQ) développé par R.S. Michalski, qui est à la base de nombreux algorithmes d'apprentissage symbolique, comme *CALM*, *GENREG*, et bien d'autres.

(****) *Les méthodes de l'Apprentissage Symbolique/Numérique :*

L'*Apprentissage Symbolique/Numérique* constitue une discipline dont l'essence même est le rapprochement de l'Analyse des Données d'une part, et de l'Apprentissage Symbolique d'autre part. Ce rapprochement est dû au fait que les deux disciplines traitent des données et des problèmes communs. C'est au cours des *journées symbolique/numérique pour l'apprentissage à partir des données* (en 1987) qui se sont déroulées à l'Université de

Paris-Dauphine, que ce terme apparût dans la communauté scientifique française.

Notre travail se situe dans le cadre statistique de la reconnaissance des formes ; cependant, le caractère général des variables considérées (quantitatives, qualitatives, structurées ou non, ...) élargit notre problématique à celle de l'apprentissage automatique symbolique/numérique. Plus précisément, notre objectif final est l'établissement d'un modèle du système étudié s'exprimant sous la forme d'un ensemble de règles, induites à partir de l'analyse du tableau de données qui constitue notre seule connaissance de départ.

PRESENTATION DU TRAVAIL

Le premier chapitre sera une mise en place des notions utilisées. Les différentes structures associées aux variables seront analysées : nature des variables, finesse, partition induite sur la population d'apprentissage (les observations), observabilité.

Nous y présenterons les différents types de variables que nous serons amenés à considérer, ainsi que les deux modes de représentation des données que nous utiliserons tout au long de ce travail :

- une représentation globale de chaque variable, associée à la notion de partition, et induisant sur les ensembles considérés (population d'apprentissage, variables) une structure liée à la relation de finesse.
- une représentation locale, introduite par la définition d'objets symboliques (assertions, hordes, ...), induisant sur les ensembles considérés une structure liée à la relation d'inclusion.

Enfin, nous aborderons le problème de l'apprentissage à partir des données (problème de modélisation). On essaiera d'expliquer certaines variables par d'autres. Les différents types de relations entre variables seront développés.

Le deuxième chapitre présentera deux systèmes qui sont à la base de nombreux algorithmes d'explication.

Le système ID3 est une méthode d'induction par arbre de décision : utilisant une représentation globale des variables, il opère par construction de partitions de plus en plus fines de la population d'apprentissage. L'évaluation de chaque partition obtenue fait appel à un critère informationnel, c'est pourquoi nous débutons le chapitre par une présentation des indices issus de la théorie de l'information et de leur utilisation dans le cadre de l'analyse structurale des systèmes. En particulier, nous présenterons des

indices permettant de juger de la faisabilité du problème d'explication et d'évaluer la qualité du système explicatif obtenu.

Le deuxième système présenté, *INDUCE*, est une méthode d'apprentissage symbolique utilisant des règles de généralisation. L'algorithme *GENREG*, utilisant la même démarche, sera aussi présenté. L'ensemble des critères permettant d'évaluer la qualité du système explicatif obtenu s'exprime sous des formes spécifiques à chacun des algorithmes proposés. Le chapitre trois tentera d'en faire une synthèse.

Dans ce sens, on mettra en évidence un certain nombre de propriétés (cohérence, complétude, ...) relatives à un ensemble de règles générées par une procédure d'apprentissage et on proposera des indices permettant leur caractérisation.

De même, on analysera les procédures de généralisation sur un ensemble de règles, qui sont à la base des méthodes d'apprentissage symbolique automatique.

On distinguera la généralisation par agrégation et la généralisation par prolongement, et on proposera les indices permettant l'évaluation du résultat de ces procédures.

Enfin, dans le dernier chapitre, nous proposerons la construction d'un modèle explicatif par généralisation des règles constituant le modèle atomique. Celui-ci, construit à l'aide d'une procédure de contraste, sépare le comportement déterministe du processus (que l'on cherche à modéliser), du comportement aléatoire dû aux perturbations et aux bruits ayant pollué les données.

Nous développons des méthodes d'induction utilisant une représentation globale (construction de partitions) et une représentation locale (prise en compte de parties).

CHAPITRE I :

ANALYSE STRUCTURALE DES SYSTEMES

A travers une *hiérarchie épistémologique* des systèmes, ce chapitre introduira les notions de *variable primaire*, d'*observation*, d'*apprentissage automatique* à partir des données (ou *modélisation*), ...

Puis, après avoir examiné les différentes *structures* possibles des variables, nous introduirons la notion d'*observabilité*, obtenue par une connaissance *a priori* du système. Cette notion pourra par exemple traduire des relations taxonomiques sur certaines variables. Elle sera très utile dans une procédure d'apprentissage (cf chapitre III).

Par opposition à la représentation globale des variables associée à la notion de partition, nous introduirons ensuite une représentation locale, associée à la notion de partie, au travers de la définition d'objets symboliques manipulables par un langage de représentation.

Enfin, nous aborderons plus en détail le problème de l'apprentissage à partir des données : problème d'explication de certaines variables (variables à *expliquer*) par d'autres (variables *explicatives*).

I. HIERARCHIE EPISTEMOLOGIQUE DES SYSTEMES

Dans le but de clarifier les différents niveaux de connaissance qu'un opérateur pourrait avoir sur un système donné, G.J. Klir (Université de New-York) a formulé une *Hiérarchie Epistémologique des Systèmes* [KLIR 75,76,77], dont l'idée de base est la suivante :

"Plus on possède d'informations sur le système considéré, et plus son niveau (hiérarchique) est élevé". Par exemple, un système sur lequel on connaît certaines relations (d'Entrées/Sorties par exemple) aura un niveau supérieur au système dont on n'a effectué qu'une simple observation.

Ainsi, un système de niveau épistémologique 0 sera défini par un ensemble de variables, chacune d'entre elles étant caractérisée par son propre ensemble de modalités observables.

Tout système de niveau 1 correspondra à un système sur lequel on a effectué une série d'observations. C'est à ce niveau que l'on distinguera les systèmes *statiques* des systèmes *dynamiques*.

Le niveau épistémologique 2 caractérisera des systèmes sur lesquels un ensemble de relations entre variables a été mis en évidence. C'est à ce niveau qu'on pourra estimer, si besoin est, les distributions de probabilité des différentes variables.

D'autres niveaux ont également été caractérisés par G.J. Klir (niveau 3 : système-structure, niveau 4 : méta-système, ...), mais ceux-ci sortent du cadre de notre travail.

Le problème de la *modélisation* d'un système complexe revient alors à passer du niveau épistémologique 1 (observation du système) à un niveau supérieur.

I.1. Niveau 0 : Système-Source

Un système de niveau 0 (niveau épistémologique le plus bas) est appelé *Système-Source*. Il est caractérisé par un ensemble de variables* primaires ainsi que par un ensemble fini d'états potentiels (ou modalités) associé à chacune d'entre elles. A ce niveau, nous ne disposons d'aucune autre connaissance.

Les variables considérées peuvent être des variables d'entrée (ou de commande), des variables de sortie, mais également des variables internes au système (encore appelées variables d'état). Ce sont les variables que l'analyste considère comme importantes, pertinentes pour l'étude du comportement de son système, leur choix ne pouvant être guidé que par une connaissance *a priori* de celui-ci.

Ainsi, C.H. Coombs [COOM 64] écrit : "si on demande à un individu s'il pense voter pour le candidat A, l'observateur, en général, enregistre sa réponse, OUI ou NON ; mais on peut se demander s'il n'y aurait pas intérêt à enregistrer le temps de réponse, ou les changements du rythme respiratoire, ou la réaction électro-dermale de l'individu, ou ce que celui-ci fait de ses mains...". Parmi toutes ces variables, l'analyste doit choisir, choix souvent guidé par son "savoir-faire". Nous verrons qu'à partir du niveau épistémologique 2, le choix de variables pertinentes (prises parmi les variables initiales) pour l'étude du système pourra être guidé. Ces variables peuvent interagir entre elles de façon non simple. De plus, aucune restriction n'est faite concernant leur structure (cf §II).

Tout au long de ce travail, nous noterons $V := \{X_1, X_2, \dots, X_N\}$ l'ensemble des N variables primaires.

Afin d'éviter toute ambiguïté lorsqu'on parlera de variables vectorielles, on introduit un ordre total o sur l'ensemble V et on note $E = (X_1, X_2, \dots, X_N)$ le vecteur des N variables primaires (après une éventuelle renumérotation, ou permutation de celles-ci), c'est-à-dire l'ensemble V ordonné par l'ordre o .

* encore appelées descripteurs, caractères, attributs, ou paramètres.

Il en sera de même pour tout sous-ensemble de V : on notera $P(V)$ l'ensemble des parties de V et $P^0(V)$ l'ensemble des parties de V , chacune étant ordonnée par l'ordre o .

A chaque variable primaire X_j ($j=1, \dots, N$) est associé un ensemble de modalités possibles (observables). Nous le noterons $O_j = \{\alpha_1^j, \alpha_2^j, \dots, \alpha_{m_j}^j\}$ ($j=1, \dots, N$; $m_j = \text{card } O_j$). A titre d'exemple, si X_j représente une variable booléenne, son ensemble de modalités observables sera $O_j = \{0, 1\}$.

Enfin, il est important de rappeler que nous considérons des ensembles de modalités observables *finis*, ce qui va à l'encontre du fait que, dans la nature, beaucoup de variables (température, taille, ...) sont des grandeurs continues. Une discrétisation préalable de ces grandeurs sera alors effectuée.

I.2. Niveau 1 : Système-Données

Tout système de niveau épistémologique 1 sera appelé *Système-Données* : il correspond à un système sur lequel on a effectué une série d'observations. A chaque observation correspondra un N -uplet représentant les N modalités observées des N variables. L'ensemble de ces observations est souvent représenté sous la forme d'un tableau (appelé *Tableau Initial des Données**). Par convention, on place les observations en lignes, et les variables en colonnes :

$\Omega \backslash \Sigma$	X_1	X_2	$\dots$	X_j	$\dots$	X_N
w_1				.		
w_2				.		
$\dots$				.		
w_i	.	.	.	$X_j(w_i)$	.	.
$\dots$				.		
w_L				.		

fig.1.1 : Tableau Initial des Données

où :

- $\Sigma := (X_1, X_2, \dots, X_N)$ est le vecteur des variables "pertinentes" pour l'étude du système.
- $\Omega := \{w_1, w_2, \dots, w_L\}$ est la population d'apprentissage (de taille L). w_i ($i=1, \dots, L$) est une observation, encore appelée *individu* (en Analyse des Données Classique), ou *objet*

* encore appelé tableau (attributs, valeurs).

(en Analyse des Données Symbolique) ou *forme* (en Reconnaissance des Formes).

→ $X_j(w_i)$ est la modalité (ou état, ou valeur) de la variable X_j correspondant à l'observation w_i , et $X_j(w_i) \in O_j$ ($i=1, \dots, L$; $j=1, \dots, N$).

Les dimensions du tableau de données sont telles que le nombre d'observations effectuées peut atteindre plusieurs centaines, voire plusieurs milliers ; le nombre de variables ne dépasse pas en général quelques dizaines.

C'est à ce niveau que nous pouvons distinguer les Systèmes Statiques des Systèmes Dynamiques :

→ Un *système statique* est un système dont les observations ne dépendent pas du temps. L'évolution temporelle de ce système est inexistante (ou quasi-inexistante). A titre d'exemple, les tableaux donnés en annexe représentent des systèmes statiques.

De plus, si aucune condition (par exemple une relation d'ordre partiel) ne caractérise la population d'apprentissage Ω , alors l'ordre des lignes du tableau de données n'a pas d'importance. Celles-ci peuvent donc être permutées à loisir.

→ Un *système dynamique* est un système possédant une évolution temporelle non négligeable. Les lignes (les observations) du tableau sont alors ordonnées par une relation d'ordre total.

Dans la plupart des cas, l'échantillonnage est effectué avec un pas constant. La population d'apprentissage se présente alors sous la forme suivante : $\Omega := \{t_0, t_0 + \delta t, t_0 + 2\delta t, \dots, t_0 + (L-1)\delta t\}$

où t_0 représente l'instant de la première observation
et δt est le pas de l'échantillonnage.

A ce niveau épistémologique, le tableau initial des données représente la seule connaissance que l'analyste possède sur le système étudié.

Nous noterons M_{X_j} l'ensemble des modalités effectivement observées de la variable X_j . Nous avons la relation $M_{X_j} \subset O_j$ ($j=1, \dots, N$).

En général, l'ensemble des modalités observables O_j de la variable X_j n'est pas connu, et on considère souvent $M_{X_j} = O_j$ ($j=1, \dots, N$).

Le même raisonnement peut être tenu pour un couple de variables, un triplet de variables ..., et plus généralement, pour le vecteur Σ de toutes les variables considérées.

Le problème que l'on se pose donc à ce niveau épistémologique, est de savoir si la population d'apprentissage Ω est un échantillon

représentatif (sous entendu des différentes combinaisons possibles des modalités observables de Σ), i.e. si Ω représente bien toutes les modalités possibles (observables) de Σ .

Ce point sera détaillé au paragraphe II du présent chapitre.

De plus, le choix de L (nombre de lignes du tableau, ou nombre d'observations effectuées) est primordial, et sera discuté au paragraphe suivant.

I.3. Niveau 2 : Système-Générateur

Un système de niveau épistémologique 2 est appelé *Système-Générateur*. C'est un système-données (niveau 1) caractérisé par un ensemble de relations génératrices, permettant de prédire, avec une précision plus ou moins grande, le comportement de certaines variables suivant l'observation effectuée sur d'autres variables.

Nous partitionnons alors l'ensemble des variables en deux classes, pour définir, d'une part, un ensemble de variables dites "à expliquer" (ou *endogènes*, ou *difficilement observables*), et d'autre part, un ensemble de variables dites "explicatives" (ou *exogènes*, ou *facilement observables*) permettant d'expliquer les premières.

A titre d'exemple, pour le tableau suivant :

	X ₁	X ₂	X ₃
w ₁	0	0	0
w ₂	0	1	1
w ₃	1	0	1
w ₄	1	1	0

fig.1.2 : un exemple de tableau de données

une des trois variables représente le bit de parité des deux autres, et nous pouvons établir la relation logique suivante :

$$X_k = \neg X_i \wedge X_j \vee \neg X_j \wedge X_i \quad \text{avec} \quad i, j, k \in \{1, 2, 3\} ; i \neq j ; j \neq k ; i \neq k$$

où $\neg$, $\wedge$, $\vee$ représentent respectivement les opérateurs de négation logique, de conjonction, et de disjonction.

Introduisons également une *mesure de probabilité* sur un ensemble fini Θ telle qu'elle l'a été dans [PARZ 60] :

$$\begin{array}{lcl} \text{Pr : } & P(\Theta) & \longrightarrow \mathbb{R} \\ & A & \longmapsto \text{Pr}(A) \end{array}$$

où $P(\Theta)$ est l'ensemble des parties de l'ensemble fini Θ .

Cette mesure vérifie les trois axiomes suivants :

- $\forall A \in P(\Theta) \quad \text{Pr}(A) \geq 0$
- si A et B sont deux ensembles disjoints, alors
 $\text{Pr}(A \cup B) = \text{Pr}(A) + \text{Pr}(B)$
- $\text{Pr}(\Theta) = 1$

En pratique, pour toute variable X_i , la probabilité d'occurrence de la modalité α ($\alpha \in M_{X_i}$) est estimée par sa fréquence relative :

$$\text{Pr}(X_i = \alpha) = \frac{\text{card} \{w \in \Omega / X_i(w) = \alpha\}}{\text{card } \Omega}$$

$$\text{De même, } \forall S \in P^0(V), \quad \text{Pr}(S = \Gamma) = \frac{\text{card} \{w \in \Omega / S(w) = \Gamma\}}{\text{card } \Omega}$$

Afin que l'estimation des probabilités par les fréquences relatives soit correcte, certaines conditions sur L (nombre d'observations) doivent être satisfaites.

→ Lorsque l'on ne considère pas de modèle mathématique caractérisant les distributions de probabilités (cas le plus général), R.C. Conant [CONA 72], faisant référence à plusieurs auteurs, propose une limite inférieure de L :

$$\text{card}(\Omega) = L \geq \prod_{i=1}^N (|O_i| - 1)$$

→ Supposons que certaines variables dépendent d'autres variables suivant une distribution de probabilité connue (cas restrictif). Cette loi peut être *normale* (loi de Laplace-Gauss), *binômiale* (ou de J. Bernoulli), de Poisson, ... Après avoir estimé les paramètres nécessaires, il y a lieu de vérifier si les données expérimentales confirment bien l'hypothèse selon laquelle certaines variables suivent la loi de répartition en question. Des critères de conformité sont alors utilisés. A titre d'exemples, nous pouvons citer le critère du Khi^2 (de Pearson), le critère de Kolmogorov, ... Nous renvoyons le lecteur aux nombreux ouvrages et articles traitant de ce sujet [PARZ 60], [VENT 73], ...

I.4. Du niveau 1 vers un niveau supérieur

Le problème de la modélisation consiste à passer du niveau épistémologique 1 (observation du système), à un niveau supérieur caractérisant le système par un ensemble de relations entre variables.

Lorsque les variables décrivant le système sont toutes quantitatives, cette démarche comprend deux étapes :

- choix d'une classe de modèles
- optimisation paramétrique, permettant de définir, parmi les modèles de la classe, celui dont la conformité avec les données sera maximale (au sens d'un critère donné).

Ainsi, par exemple, l'analyse canonique (amorcée par H. Hotelling [HOTE 36]) détermine le meilleur modèle linéaire (au sens de la dispersion des erreurs de prédiction) qu'il est possible d'établir pour relier les observations effectuées sur deux (ou plus de deux) ensembles disjoints de variables [CARR 68].

Lorsque certaines, ou toutes les variables décrivant le système sont qualitatives, le modèle est directement construit à partir des données observées.

A titre d'exemple, sur le tableau de la fig.1.2, une relation du type :

$$X_3 = \neg X_1 \wedge X_2 \vee \neg X_2 \wedge X_1$$

a pu être mise en évidence. Cette équation logique représente un modèle du système considéré.

Le problème de modélisation d'un système revient donc à *apprendre* [TIBE 87] des relations sur celui-ci, et ceci, par un procédé automatique, i.e. sans professeur.

On parlera dès lors aussi bien de *modélisation* d'un système que d'*apprentissage automatique* à partir des données (observées sur le système).

II. REPRESENTATION GLOBALE : Structure des données

Après avoir examiné les différentes structures associées aux variables primaires, nous rappellerons les notions de *finesse* d'une variable et de *partitions induites* sur la population d'apprentissage Ω . Puis, nous introduirons les notions de variable vectorielle et de domaine d'observabilité, celui-ci pouvant être utilisé dans des algorithmes d'apprentissage (cf chapitre III).

II.1. Les variables primaires

Chaque variable primaire $X_i \in V$ peut être considérée comme une application de l'ensemble d'apprentissage vers son ensemble de modalités observées :

$$\begin{aligned} X_i : \Omega &\longrightarrow M_{X_i} \\ w &\longmapsto X_i(w) \end{aligned}$$

M_{X_i} peut être caractérisé par plusieurs structures, suivant la nature de la variable en question. Celle-ci peut être :

→ *quantitative* (ou *numérique* ; $M_{X_i} \subset \mathbb{R}$) : à titre d'exemples, la température, la taille, ... sont des variables quantitatives.

→ *qualitative ordinale* (ou *ordonnée*) : $(M_{X_i}, \leq)$ est un ordre total. A titre d'exemples, la variable "taille" composée des modalités (petit, moyen, grand), le "poids" composé des modalités (léger, lourd), ... sont des variables qualitatives ordinales.

→ (*qualitative*) *nominale* : aucune relation n'est définie sur M_{X_i} . Par exemple, la couleur des yeux, la couleur des cheveux, les variables booléennes, ... sont des variables nominales (sous-entendues "qualitatives").

→ *structurée* : il existe une ou plusieurs relation(s) sur M_{X_i} . Par exemple, on peut définir une structure taxonomique sur les modalités de la variable "humain" :

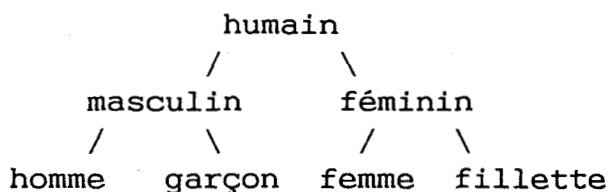


fig.1.3 : une structure taxonomique

on peut également avoir un ordre partiel combiné à une taxonomie : (bébé < enfant < adolescent < adulte)

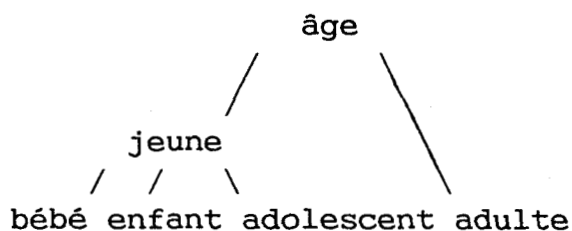


fig.1.4 : une variable structurée

Nous pouvons constater que le domaine des variables structurées est extrêmement vaste.

Remarque :

En Intelligence Artificielle, on appelle variable linéaire toute variable qui est quantitative, ou qualitative ordinale.

II.2. Finesse d'une variable primaire

Etant donnée une variable primaire X_i , il est toujours possible d'en créer d'autres, en partitionnant l'ensemble des modalités M_{X_i} .

Soit $M_{\tilde{X}_i}$ une partition de M_{X_i} . On notera $\tilde{X}_i$ la variable déduite de X_i , dont l'ensemble des modalités est $M_{\tilde{X}_i}$.

L'ensemble $\mathcal{P}(M_{X_i})$ des partitions de M_{X_i} , ordonné par la relation de finesse $\leq$, constitue un treillis [SZAS 62] [BIRK 67] [BARB 70] [KAUF 78] dont M_{X_i} est le *plus petit élément (minorant universel)*. On dira que la variable $\tilde{X}_i$ est plus grosse que X_i ($\tilde{X}_i \geq X_i$), puisque chacune des classes de M_{X_i} est entièrement incluse (par définition) dans une classe de $M_{\tilde{X}_i}$.

On peut noter que, pour tout élément de la population d'apprentissage, la connaissance de la modalité prise par X_i implique la connaissance de celle prise par $\tilde{X}_i$.

A titre d'exemple, à partir de la variable primaire âge (à 4 modalités) de la fig.1.4, on peut créer 14 variables plus grosses, dont certaines sont données ci-dessous :

âge₁ : {bébé, enfant ou adulte, adolescent}

âge₂ : {bébé ou enfant ou adolescent, adulte}

âge₃ : {jeune ou adulte}

Remarques :

→ Compte tenu de la structure de la variable primaire, le label "jeune" peut désigner la classe {bébé ou enfant ou adolescent}, de sorte que l'ensemble des modalités de la variable âge₂ peut s'écrire plus simplement : {jeune, adulte}.

→ La variable âge₃ ne possède qu'une modalité. Elle constitue le *plus grand élément (majorant universel)* du treillis. Elle est constante, puisqu'elle prend la même modalité pour tous les objets de la population d'apprentissage.

II.3. Variables primaires et partitions de Ω

A chaque variable primaire $X_i \in V$ de notre système, nous pouvons associer une partition de la population d'apprentissage Ω , de la façon suivante :

$$P_{X_i}(\Omega) = \{X_i^{-1}(u), u \in M_{X_i}\}$$

où $X_i^{-1}(u) = \{w / X_i(w) = u\}$.

L'ensemble $P(\Omega)$ des partitions de Ω , muni de la relation de finesse $\leq$, possède une structure de treillis. Ainsi, il est possible de construire une relation d'ordre et une relation d'équivalence entre les variables.

Considérons X_i et X_j ($i \neq j$) deux variables de notre système. Nous définissons :

$$X_i \equiv X_j \iff P_{X_i}(\Omega) = P_{X_j}(\Omega)$$

On dit que X_i et X_j sont des variables équivalentes.

Propriété :

$$X_i \leq X_j \iff P_{X_i}(\Omega) \leq P_{X_j}(\Omega)$$

où $\leq$ représente la relation d'ordre partiel entre partitions "être plus fine que".

Autrement dit, si X_j est une variable primaire plus grosse que X_i , la partition de Ω associée à (induite par) X_j est également plus grosse que celle induite par X_i .

II.4. Variables vectorielles

$V := \{X_1, X_2, \dots, X_N\}$ étant l'ensemble des N variables primaires, $P^0(V)$ représente l'ensemble des parties de V , chacune étant ordonnée par l'ordre $\circ$ défini sur V .

Chaque élément S de $P^0(V)$ représente une application de Ω dans son ensemble de modalités observées (noté M_S) :

$$\begin{array}{ccc} S : & \Omega & \xrightarrow{\quad} M_S \\ & w & \longmapsto S(w) \end{array} \quad \text{où } S \in P^0(V)$$

Cas particulier : Lorsque S représente une variable primaire :

$$\begin{array}{ccc} X_i : & \Omega & \xrightarrow{\quad} M_{X_i} \\ & w & \longmapsto X_i(w) \end{array}$$

La variable vectorielle S induit une partition de Ω , qui est l'intersection des partitions de Ω induites par chaque variable primaire composante de S :

$$P_S(\Omega) = \bigcap_{X_i \in S} P_{X_i}(\Omega)$$

A titre d'exemple, sur le tableau de la fig.1.2 :

$$P_{X1}(\Omega) = \{w_1 \text{ ou } w_2, w_3 \text{ ou } w_4\}$$

$$P_{X2}(\Omega) = \{w_1 \text{ ou } w_3, w_2 \text{ ou } w_4\}$$

$$P(X1, X2)(\Omega) = \{w_1, w_2, w_3, w_4\} = P_{X1}(\Omega) \cap P_{X2}(\Omega)$$

Nous pouvons alors en déduire la propriété de finesse suivante :

Propriété :

Soient S_1 et S_2 appartenant à $P^0(V)$, telles que $S_1 \subset S_2$. On a alors les relations suivantes :

$$M_{S1} \geq M_{S2} \quad \text{et} \quad P_{S1}(\Omega) \geq P_{S2}(\Omega).$$

En d'autres termes, la relation d'inclusion entre sous-ensembles de variables induit une relation de finesse entre les variables vectorielles et les partitions associées de la population d'apprentissage.

A titre d'exemple, sur le tableau de la fig.1.2 :

$$P_{X1}(\Omega) = \{w_1 \text{ ou } w_2, w_3 \text{ ou } w_4\}$$

$$P(X1, X2)(\Omega) = \{w_1, w_2, w_3, w_4\}$$

La partition de Ω induite par (X_1, X_2) est plus fine que celle induite par X_1 .

II.5. Observabilité des variables

OBSERVATION ...

Soit $S \in P^0(V)$. Nous avons vu que les ensembles M_S (ensemble de modalités observées de S) étaient construits à partir de l'observation du système.

En particulier, nous définissons :

$M_{X_i} = \{ \alpha / \exists w \in \Omega, X_i(w) = \alpha \}$. l'ensemble des modalités observées de la variable primaire X_i ($X_i \in V$).

Par contre, $\prod_i M_{X_i}$ (l'ensemble produit des modalités observées

des X_i ; $X_i \in V^0$) ne représente pas, en général, un ensemble de modalités observées :

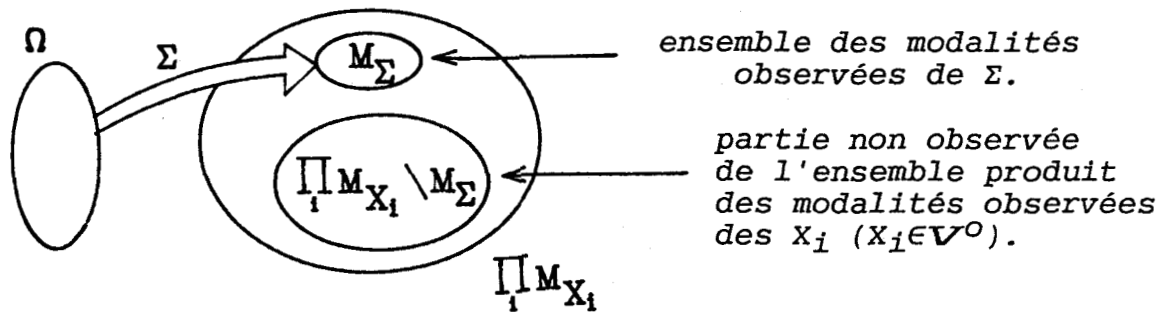


fig.1.5 : Parties observée et non observée de Σ .

En effet, deux cas de figures peuvent se présenter :

- certaines modalités de $\prod_i M_{X_i}$ n'ont pas encore été observées, mais sont observables (échantillonnage incomplet sur Σ)
- l'existence de relations (connues ou inconnues) entre les X_i ($X_i \in V$) entraîne la non observabilité de certaines modalités de $\prod_i M_{X_i}$.

A titre d'exemple, pour le tableau donné en annexe 5, aucun chien ayant une grande taille (taille +) n'a un poids faible (poids -). De même, aucun chien ayant un poids + n'a une taille -. On peut alors se demander si ces cas peuvent être observés, ou si ils ne seront (dans la nature des choses) jamais observés (donc non observables).

... et OBSERVABILITE [POM 91] :

Soit $S \in P^0(V)$. L'ensemble M_S est inclus dans O_S , qui représente l'ensemble des modalités possibles (observables) de la variable S .

$$S : \Omega \longrightarrow M_S \subset O_S$$

O_S n'est pas obtenu à partir du tableau initial des données ; seule une connaissance *a priori* de la variable S permet de le déterminer.

Notons que l'existence de relations possibles (connues ou inconnues a priori) entre les variables X_i restreint le domaine $\prod_i O_{X_i}$

(appelé l'espace de représentation des observations) à l'ensemble O_Σ des modalités observables de Σ .

Nous pouvons écrire les relations d'inclusion suivantes :

$$\rightarrow M_\Sigma \subset M_\Sigma^* \subset \prod_i M_{X_i} \subset \prod_i O_{X_i}$$

$$\rightarrow M_\Sigma \subset M_\Sigma^* \subset O_\Sigma \subset \prod_i O_{X_i}$$

où M_Σ^* est défini comme la partie observable du produit $\prod_i M_{X_i}$:

$$M_\Sigma^* = \prod_i M_{X_i} \cap O_\Sigma$$

Afin de clarifier les définitions ci-dessus, nous proposons une vue synthétique des ensembles observés et observables, en ne considérant que deux variables explicatives X_1 et X_2 :

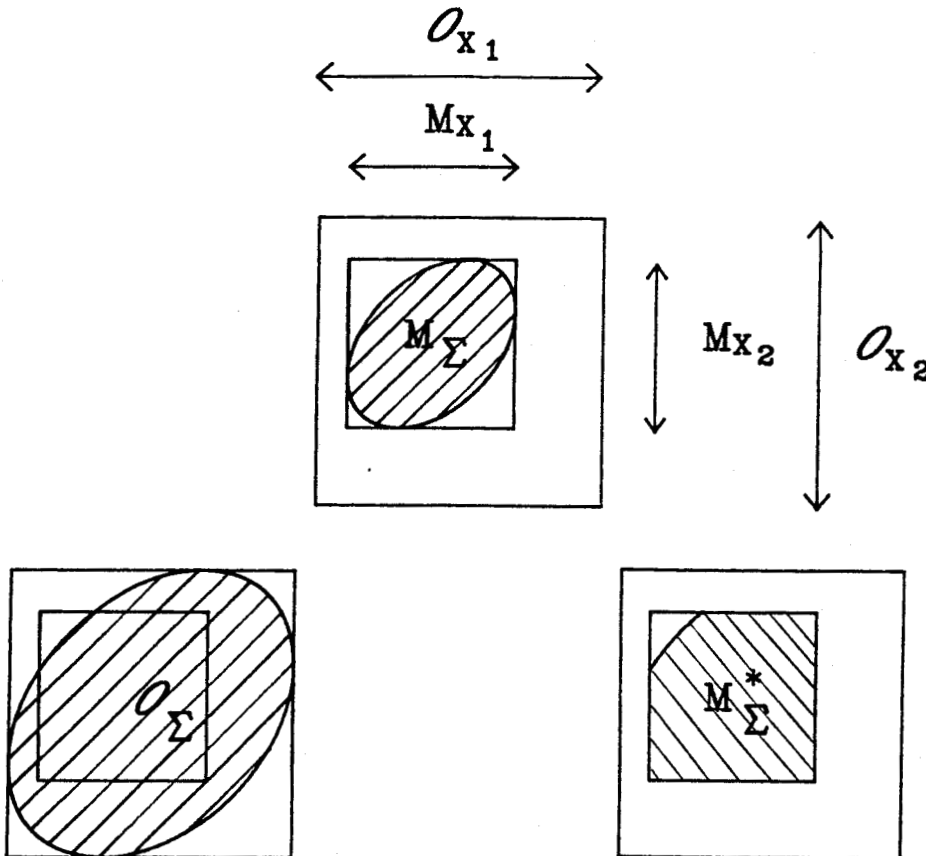


fig.1.6 : Parties observée et observable de Σ , pour $p=2$

Remarques :

→ Nous avons distingué, pour Σ :

- l'ensemble M_Σ des modalités observées.
- l'ensemble O_Σ des modalités observables.
- l'ensemble $\prod_i O_{Xi} \setminus O_\Sigma$ des modalités non observables.

→ En toute rigueur, nous devrions définir un quatrième ensemble, qui représenterait les modalités de Σ dont nous ne savons pas si elles sont observables ou non observables :

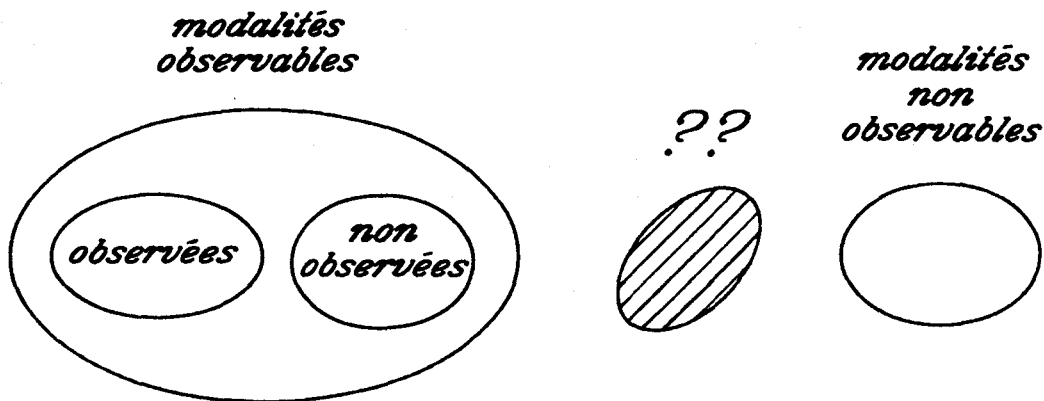


fig.1.7.

Deux attitudes sont alors possibles. La première est de les considérer comme non observables ; la deuxième, de les considérer comme observables, mais non encore observées.

→ Dans la plupart des cas, l'analyste considère que l'échantillonnage effectué est complet sur Σ : toutes les combinaisons observables de modalités ont été observées. On prendra alors : $O_{Xi} = M_{Xi}$ ($i=1, \dots, N$) (et donc $\prod_{(i)} O_{Xi} = \prod_{(i)} M_{Xi}$) et $O_\Sigma = M_\Sigma$.

A titre d'exemple, si nous ne considérons que les deux variables taille (X_1) et poids (X_2) du tableau des données de l'annexe 5, nous avons :

$$\begin{aligned}
 M_{X1} &= \{-, o, +\} = O_{X1} \quad ; \quad M_{X2} = \{-, o, +\} = O_{X2} \\
 M_{X1} \times M_{X2} &= \{(-, -), (-, o), (-, +), (o, -), (o, o), (o, +), \\
 &\quad (+, -), (+, o), (+, +)\} = O_{X1} \times O_{X2} \\
 M_\Sigma &= \{(-, -), (o, -), (o, o), (+, o), (+, +)\} \text{ (modalités observées)} \\
 O_{X1} \times O_{X2} \setminus M_\Sigma &= \{(-, o), (-, +), (o, +), (+, -)\} \text{ (modalités non observées)}
 \end{aligned}$$

De même, si on reprend le tableau de la fig.1.2 :

$$\begin{aligned} M_{X1} &= O_{X1} = \{0,1\} ; M_{X2} = O_{X2} = \{0,1\} ; M_{X3} = O_{X3} = \{0,1\} \\ M_{X1} \times M_{X2} \times M_{X3} &= O_{X1} \times O_{X2} \times O_{X3} = \{(0,0,0), (0,0,1), \dots, (1,1,1)\} \\ M_{\Sigma} &= \{(0,0,0), (0,1,1), (1,0,1), (1,1,0)\} = O_{\Sigma} \end{aligned}$$

Ici, M_{Σ} (et O_{Σ}) constitue l'ensemble des modalités observées et observables.

III. REPRESENTATION LOCALE : LE LANGAGE VL

Jusqu'à présent, nous avons représenté la structure des données en termes de treillis de partitions, de partitions de Ω associées (induites) à une variable, de finesse de variable, etc... Les variables étaient manipulées globalement.

Il est maintenant nécessaire de convenir d'un langage afin de représenter les observations effectuées localement sur notre système, afin de les manipuler, et afin de trouver des relations sur celles-ci. Le langage de représentation choisi sera *symbolique*, et les opérateurs, logiques.

Nous nous limiterons volontairement à des cas particuliers du langage VL (variable-valued logic calculus) proposé par R.S. Michalski [MICH 80,83a] et présenté dans l'annexe 1.

Nous présentons ici des objets symboliques [DID 89,90] : l'évènement élémentaire, qui est un cas particulier de l'objet assertion, qui est lui-même un cas particulier de l'objet horde.

III.1. L'évènement élémentaire

Un évènement élémentaire peut être défini comme une application :

$$\begin{aligned} \text{ex}_{iV_i} : \Omega &\longrightarrow \{\text{vrai}, \text{faux}\} \\ w &\longmapsto \text{ex}_{iV_i}(w) = \text{vrai} \iff X_i(w) \in V_i \\ &\quad (\text{faux sinon}). \end{aligned}$$

avec :

- $X_i \in V$ (V est l'ensemble des variables primaires)
- $V_i \in P(O_i)$

La définition en compréhension d'un évènement élémentaire s'écrit :

$$e = [X_i = V_i].$$

On peut également définir son extension dans Ω :

$$\begin{aligned} |e|_{\Omega} &:= \{w \in \Omega / X_i(w) \in V_i\} \\ &= e_{\text{ex}_{iV_i}}^{-1}(\text{vrai}). \end{aligned}$$

$e=[X_1=\text{noir v blanc}] ; |e|_\Omega=\{w_1, w_3, w_5\}$

$\bar{e}=[X_1=\text{brun}] ; |\bar{e}|_\Omega=\{w_2, w_4\}$

$e'=[X_2=[0..2]] ; |e'|_\Omega=\Omega \setminus \{w_3\}$

On peut également écrire : $e=[\tilde{X}_1=\text{nb}]$ où $\text{nb}=\{\text{noir v blanc}\}$
et $M_{\tilde{X}_1}=\{\text{nb, brun}\}.$

III.2. L'objet assertion

III.2.1. Définition

Un objet assertion peut être défini comme une conjonction d'évènements élémentaires :

$\forall S \in P_R(V) ; VV=(V_1, V_2, \dots, V_k)^T$ avec $V_i \in P(O_i) \quad i=1, \dots, k.$
où $P_R(V)$ est l'ensemble des parties de V à k composantes (ou variables primaires) distinctes, chacune étant ordonnée par l'ordre o .

$$\begin{aligned} a_{S,V} : \Omega &\longrightarrow \{\text{vrai, faux}\} \\ w &\longmapsto a_{S,V}(w)=\text{vrai} \iff X_i(w) \in V_i \quad \forall i=1, \dots, k. \end{aligned}$$

Remarque : On aurait pu simplifier l'écriture :
 $a_{S,V}(w)=\text{vrai} \iff S(w) \in V.$

Sa définition en compréhension est :

$$a = [X_1(w)=V_1] \sim [X_2(w)=V_2] \sim \dots \sim [X_k(w)=V_k]$$

Son extension dans Ω est :

$$\begin{aligned} |a|_\Omega &:= a_{S,V}^{-1}(\text{vrai}) \\ &= \{w \in \Omega / X_i(w) \in V_i \quad \forall i=1, \dots, k\} \\ &= \{w \in \Omega / S(w) \in V\} \end{aligned}$$

Remarques :

- Une variable X_i pour laquelle $V_i=O_i$ n'apporte aucune information. Nous dirons donc que si pour une assertion donnée, une variable couvre toutes ses modalités observables, alors dans l'écriture en compréhension de celle-ci, cette variable pourra être omise. L'extension de l'objet assertion *simplifié* est la même.
- Inversement, toute assertion $a_{S,V}$ pourra se mettre sous la forme $a_{\Sigma, V'}$, où $V'=(V_1, V_2, \dots, V_N)^T$, $\Sigma=(X_1, X_2, \dots, X_N)^T$ et $\forall X_i \in \Sigma \setminus S \quad V_i=O_i$. $a_{\Sigma, V'}$ sera noté $a_{V'}$.
- Si $k=1$, on retrouve l'évènement élémentaire.
- Afin de reprendre le vocabulaire introduit par R.S. Michalski, une assertion est encore appelée un *complexe* (conjonction de sélecteurs).

→ En reprenant la comparaison global/local, on peut dire qu'une assertion porte sur une modalité de la variable S (où $S \in P^0(V)$) ou sur une modalité de la variable $\tilde{S}$, où $\tilde{S}$ est une variable plus grosse que S .

Exemple : Si on considère le tableau de la fig.1.8 :

l'objet $a = [X_1 = \text{blanc} \vee \text{noir}] \wedge [X_3 = 0 \vee 1]$; $|a|_\Omega = \{w_1, w_3, w_5\}$
peut s'écrire : $a = [X_1 = \text{blanc} \vee \text{noir}]$.

III.2.2. Structure de l'ensemble des assertions

Définissons la notion de *finesse* sur l'ensemble des assertions noté A_Σ .

Définition : On dira que a_V est plus fine que $a_{V'}$ ($a_V \leq a_{V'}$) si et seulement si :

$$V_i \subset V'_i \quad V_i = 1, \dots, N.$$

De cette définition découle une propriété :

Propriété : $a_V \leq a_{V'} \implies |a_V|_\Omega \subset |a_{V'}|_\Omega$

Si deux assertions sont comparables par la relation de finesse, alors leurs extensions sont comparables par la relation d'inclusion.

La réciproque est fausse : dans l'exemple suivant, $|a_{V'}|_\Omega \subset |a_V|_\Omega$ et ces deux assertions ne sont pas comparables par la relation de finesse.

La réciproque aurait été vraie si nous avions défini les assertions sur les ensembles de modalités observées, et non sur les ensembles de modalités observables.

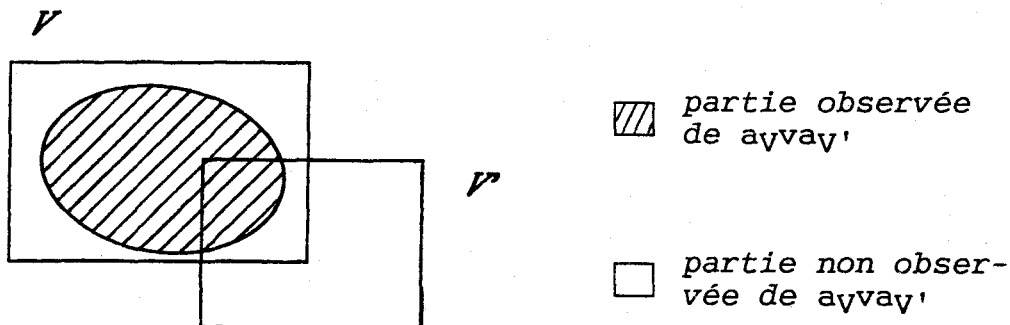


fig.1.9

III.3. L'objet horde

Une disjonction d'assertions est appelée un *objet horde*. Son extension est la réunion des extensions des assertions le constituant.

Soit H_Σ l'ensemble des objets hordes définis sur Σ .

Propriété : $(H_\Sigma, \leq)$ possède une structure de treillis :

- $h'' = (h \vee h')$ est le supremum de h et h' (où $h, h' \in H_\Sigma$).
(on a, de plus, $|h''|_\Omega = |h|_\Omega \cup |h'|_\Omega$). Le plus grand objet horde est l'objet assertion recouvrant toutes les modalités observables de Σ .
- $h'' = (h \wedge h')$ est l'infimum de h et h' (où $h, h' \in H_\Sigma$).
 $|h''|_\Omega = |h|_\Omega \cap |h'|_\Omega$. Le plus petit élément est ϕ .

Par définition, un objet horde peut être décomposé en une suite d'assertions. C'est pour cette raison que, dans la suite de ce travail, nous manipulerons essentiellement des objets assertions.

III.4. Intérêt du langage de représentation choisi

L'intérêt principal du langage de représentation choisi est la possibilité d'observer les fluctuations locales des données.

En effet, ce langage nous permet de désigner un sous-ensemble de classes de la partition produit. Cette représentation permet donc une *vue locale* des données observées.

Par contre, le langage des partitions (introduit au §II) prend en compte toutes les classes (puissance de représentation moins importante) des partitions induites. La représentation est *globale* et ne pourra pas, par conséquent, manipuler des sous-ensembles de modalités des données observées.

IV. APPRENTISSAGE A PARTIR DES DONNEES

Nous introduirons les notions de variables à *expliquer* et de variables *explicatives*, ainsi que les différentes relations envisageables entre celles-ci, suivant que les données d'apprentissage sont cohérentes ou incohérentes.

IV.1. Variables à expliquer - Variables explicatives

Le but premier de la modélisation d'un système est de montrer comment certaines variables dépendent d'autres variables (par exemple, dans le but d'obtenir des relations de surveillance ou d'effectuer un diagnostic). L'analyste partitionne alors l'ensemble des variables primaires en deux classes distinctes :

→ Appellons Y la (ou les) variable(s) à expliquer. Certains auteurs l'appellent variable endogène, ou variable difficilement observable (par exemple, d'acquisition coûteuse, ou souvent bruitée, ou non accessible à la mesure, ...).

→ $X=(x_1, x_2, \dots, x_p)$ représente le vecteur des p variables permettant éventuellement d'expliquer Y . Elles seront appelées variables explicatives, exogènes, ou facilement observables (elles peuvent être d'acquisition non coûteuse, fiables, ...).

Nous pouvons alors formuler une autre représentation du tableau des données (voir fig.1.1) :

Σ Ω	X						Y
	$[x_1$	x_2	$\dots$	x_j	$\dots$	$x_p]$	
w_1				.			$Y(w_i)$
w_2				.			
$\dots$				.			
w_i	.	.	.	$x_j(w_i)$	.	.	
$\dots$				.			
w_L				.			

fig.1.10 : Tableau des données

où :

- $\Omega=\{w_1, w_2, \dots, w_L\}$ est la population d'apprentissage. w_i ($i=1, \dots, L$) est la $i^{\text{ème}}$ observation.
- $\Sigma = (X, Y) = (x_1, x_2, \dots, x_p, Y)$ est le vecteur des variables considérées par l'analyste comme pertinentes pour l'étude de son système.
- $x_j(w_i)$ (resp. $Y(w_i)$) représente la modalité de la variable x_j (resp. Y) pour l'observation w_i ($i=1, \dots, L$; $j=1, \dots, p$).

Nous utiliserons tantôt la première représentation de Σ ($\Sigma=(X_1, X_2, \dots, X_N)$), tantôt la seconde ($\Sigma=(X, Y)$), celle-ci étant plus adaptée au problème d'explication.

Notations utilisées :

→ L'ensemble des modalités observées de la variable x_i sera noté $Mx_i := \{a_1^i, a_2^i, \dots, a_{n_i}^i\}$ ($i=1, \dots, p$; $n_i = \text{card } Mx_i$). Afin de simplifier l'écriture, son ensemble de modalités observables sera noté O_i .

→ $M_X := \{a_1, a_2, \dots, a_n\}$ représentera l'ensemble des modalités observées de la variable vectorielle X ($n = \text{card } M_X$). Ces modalités sont des conjonctions des modalités des différentes variables initiales x_i ($i=1, \dots, p$). Son ensemble de modalités observables sera noté O .

→ $M_Y := \{\beta_1, \beta_2, \dots, \beta_m\}$ est l'ensemble des modalités de la variable à expliquer Y . Ces modalités peuvent être des *modes de fonctionnement* (en diagnostic), ou des *classes d'appartenance* (en Reconnaissance des Formes).

Le cardinal de M_Y ($\text{card } M_Y = m$) est en général petit. En effet, pour diverses raisons (simplification du problème, obtention de simples résultats qualitatifs, ...), l'analyste peut être amené à créer une partition de l'ensemble des modalités de Y . A titre d'exemple, si on essaie d'expliquer la quantité de calories apportées par certaines boissons en fonction d'autres caractéristiques, on peut être amené à discrétiser l'échelle des calories : - (peu de calories), o (pas d'excédent de calories), + (beaucoup de calories).

Chaque modalité β_j ($j=1, \dots, m$) représentera alors une classe de la partition ainsi créée.

Dans la suite de notre travail, nous prendrons $O_Y = M_Y$. En effet, nous chercherons à expliquer Y par X , i.e. expliquer les modalités de Y déjà observées.

→ En regard d'une modalité β_j de Y , l'ensemble $\{w / Y(w) = \beta_j\}$ représente une *classe* (ou *concept*) de l'ensemble d'apprentissage, encore appelé l'ensemble des *exemples* (en apprentissage), sous entendu de β_j . On dit également *prototypes* (en théorie de la décision). Les autres observations sont appelées les *contre-exemples*. Certains auteurs utilisent les termes *exemples positifs* (notés P ou $+$) pour les exemples, et *exemples négatifs* (notés N ou $-$) pour les contre-exemples.

→ L'ensemble des classes $\{w / Y(w) = \beta_j\}$ forme une partition de Ω . On dit que l'ensemble des classes induites par les modalités β_j ($j=1, \dots, m$) de Y constitue une *hypothèse* [REND 86].

Cette procédure dichotomique pourra être renouvelée sur toute partition de V en deux classes distinctes. En effet, nous pouvons procéder de deux façons différentes :

- Y est fixée, et on cherchera les variables les plus explicatives de Y, suivant un critère donné.
- Y n'est pas fixée : pour chaque variable, on se posera la question de savoir si on peut l'expliquer à partir d'autres variables, et si oui, lesquelles ?

Afin de définir les variables les plus explicatives, nous utiliserons ultérieurement des indices issus de la Théorie de l'Information.

Tout au long de ce travail, nous chercherons à caractériser des relations liant Y à $S \subset \bar{X}$ (souvent ces relations sont appelées des *relations d'Entrées/Sorties*, même si X et Y ne représentent pas respectivement des entrées et des sorties physiques du système), où $\bar{X}$ est une variable plus grosse que X.

Pour ce faire, nous réécrivons le tableau des données de la fig.1.10 sous la forme d'un *tableau de contingence* (ou *tableau de fréquences*) de X et Y, représenté fig.1.11. La construction de ce tableau nécessite l'estimation des probabilités conjointes p_{ij} d'obtention des modalités α_i et β_j par les fréquences relatives ($i=1, \dots, n$; $j=1, \dots, m$).

$M_Y \backslash M_X$	β_1	β_2	...	β_j	...	β_m	
α_1				.			$P_{i.}$
α_2				.			
...				.			
α_i	.	.	.	p_{ij}	.	.	
...				.			
α_n				.			$P_{.j}$

fig.1.11 : Tableau de contingence $[P_{IJ}]$

où :

- $I=\{1,2,\dots,n\}$; $n = \text{card } M_X$; $J=\{1,2,\dots,m\}$; $m = \text{card } M_Y$
- p_{ij} est la probabilité d'occurrence conjointe des modalités α_i et β_j .
- $P_{i.} = \sum_j p_{ij}$; $P_{.j} = \sum_i p_{ij}$

IV.2. Relations entre variables

Dans un premier temps, nous exposerons les différents modèles possibles, ainsi que les différentes structures associées aux variables [POM 90b].

Puis, nous introduirons la notion de règle, qui est un objet symbolique adapté au problème de la modélisation.

IV.2.1. Comparaison des partitions de Ω induites par deux variables

Soient X et Y deux variables équivalentes. Par définition de la relation d'équivalence, il existe une relation bijective :

$$\begin{array}{ccc} f : M_X \longrightarrow M_Y & f^{-1} : M_Y \longrightarrow M_X & (1) \\ \alpha \longmapsto f(\alpha) & \beta \longmapsto f^{-1}(\beta) & \end{array}$$

telle que l'assertion $[X=\alpha] \cdot [Y=f(\alpha)]$ (ou $[X=f^{-1}(\beta)] \cdot [Y=\beta]$) prend la valeur vrai si l'assertion $[X=\alpha] \cdot [Y=M_Y]$ (ou $[X=M_X] \cdot [Y=\beta]$) est vraie.

En d'autres termes, chaque variable X et Y est un recodage univoque de l'autre, et les relations (1) constituent un modèle du système $\{X, Y\}$, sous la forme d'équivalences :

$$\begin{array}{ccc} X=\alpha & \Longleftrightarrow & Y=f(\alpha) & \forall \alpha \in M_X \\ X=f^{-1}(\beta) & \Longleftrightarrow & Y=\beta & \forall \beta \in M_Y \end{array} \quad (2)$$

remarque :

Le tableau de contingence (fig.1.11) est alors un tableau d'affectation, où à une modalité de la variable X correspond une et une seule modalité de la variable Y , et réciproquement. Il y a une et une seule probabilité conjointe non nulle par ligne et par colonne.

Considérons maintenant deux variables X et Y telles que $X \leq Y$. Dans ce cas, nous pouvons montrer que :

(i) il existe une relation $Y=f(X)$:

$$\begin{array}{ccc} f : M_X \longrightarrow M_Y \\ \alpha \longmapsto f(\alpha) \end{array}$$

telle que les implications suivantes décrivent le système $\{X, Y\}$:

$$X=\alpha \implies Y=f(\alpha) \quad \forall \alpha \in M_X \quad (3)$$

(ii) chaque ligne du tableau de contingence ne contient qu'une seule probabilité non nulle.

(iii) il existe une variable $\tilde{X}$ plus grosse que X et équivalente à Y .

démonstration :

Par définition, $X \leq Y \iff P_X(\Omega) \leq P_Y(\Omega)$

et $P_X(\Omega) \leq P_Y(\Omega)$ est équivalent à dire :

$\forall \alpha \in M_X \quad \forall w \in \Omega \quad \exists ! \beta \in M_Y \quad / \quad X(w) = \alpha \text{ et } Y(w) = \beta$

ce qui est équivalent à dire :

il existe une fonction f telle que $f : M_X \longrightarrow M_Y$
 $\alpha \longmapsto f(\alpha) = \beta$
cqfd pour (i)

ce qui est équivalent à dire qu'à chaque modalité observée de X correspond une modalité observée de Y (une probabilité non nulle par ligne du tableau de contingence).

cqfd pour (ii)

On peut également chercher une variable $\bar{X}$ plus grosse que X telle que ses modalités observées soient la réunion des modalités observées α de X pour lesquelles $f(\alpha)$ est identique. Les variables $\bar{X}$ et Y sont alors équivalentes.

cqfd pour (iii)

Remarques :

→ Pratiquement, à la seule vue du tableau de contingence, si il existe une seule probabilité non nulle par ligne, alors :

$[\bar{X} \geq X \text{ et } \bar{X} \equiv Y] \iff X \leq Y \iff Y = f(X).$

Nous dirons alors que les données d'apprentissage sont cohérentes. X et Y sont totalement liées : il existe une relation déterministe entre ces deux variables.

→ Dans le cas contraire (plus d'une case non nulle par ligne), les données d'apprentissage sont incohérentes. Cela signifie que plusieurs observations, caractérisées par la même modalité de X , présentent des modalités différentes de Y : on ne peut alors pas expliquer Y par X . Deux conclusions sont alors envisageables :

- l'absence d'explication à tort : le modèle déterministe $Y = f(X)$ est représentatif du système, mais les données relevées sont biaisées (bruit, mauvais fonctionnement d'un sous-système).

- l'absence d'explication à raison : le processus étudié ainsi est non déterministe (manque de variables explicatives, ...).

Il serait alors intéressant de donner une mesure de la faisabilité du problème d'explication lorsque les données sont incohérentes (ce point sera abordé au chapitre suivant).

Dans les cas précédemment décrits, nous pouvons remarquer que la réciproque est vraie : si un modèle tel que (3) (resp. tel que (2)) décrit le fonctionnement du système $\{X, Y\}$, alors le tableau des données issu de l'observation devra faire apparaître une relation d'ordre (resp. une équivalence) entre les variables X et Y . La recherche de relations fonctionnelles liant les variables observées (extraction de connaissance à partir des données) est donc basée sur la recherche de relations d'ordre ou d'équivalence entre celles-ci, ou ce qui revient au même, entre les partitions de la population d'apprentissage Ω induites par ces variables. La *Théorie de l'Information* appliquée à l'*Analyse Structurale* nous donne des indices efficaces pour la recherche de telles relations [STAR 84].

Nous avons vu qu'à chaque variable de V , il est possible d'associer une partition de Ω . Si nous ne faisons aucune distinction entre variables équivalentes, alors $P(\Omega)$ (l'ensemble des partitions de Ω) peut être considéré comme l'ensemble de toutes les variables définies sur Ω . Les attributs x_i et Y peuvent alors être considérés comme un sous-ensemble particulier de $P(\Omega)$, à partir duquel les autres variables peuvent être générées. En effet, considérons z , une variable parmi les variables x_i ($i=1, \dots, p$) et Y , nous pouvons toujours construire une variable $\bar{z}$ plus grosse que z , i.e. une variable telle que son sous-ensemble de modalités $M_{\bar{z}}$ vérifie :

$$\forall u \in M_z \exists \bar{u} \in M_{\bar{z}} / u \in \bar{u}.$$

$\bar{z}$ est une variable plus générale que z ($M_{\bar{z}}$ réalise une partition de l'ensemble des modalités de M_z).

Supposons que la procédure d'apprentissage signale l'existence d'un modèle sous la forme de règles (3), caractérisé par $f : M_X \longrightarrow M_Y$, et considérons $\bar{X}$, la variable la plus grosse parmi toutes les variables plus grosses que X et plus fines que Y . Il est clair que le modèle $\bar{f} : M_{\bar{X}} \longrightarrow M_Y$ décrit aussi bien que f la connaissance extraite de la population d'apprentissage, avec un plus petit nombre d'implications.

Un cas particulier est défini par $\bar{X} = \text{SEP}^0(V \setminus Y)$. Ainsi, le modèle $\bar{f} : M_S \longrightarrow M_X$ utilise un plus petit nombre de règles.

Le problème d'explication peut donc se poser en ces termes : Connaissant une partition $P_Y(\Omega)$ de la population d'apprentissage induite par la variable à expliquer Y , il s'agit de trouver une partition $P_{\tilde{X}}(\Omega)$ de Ω induite par $\tilde{X}$ (où $\tilde{X}$ est une variable plus grosse que X) telle qu'elle soit la plus proche de $P_Y(\Omega)$, au sens d'un critère donné.

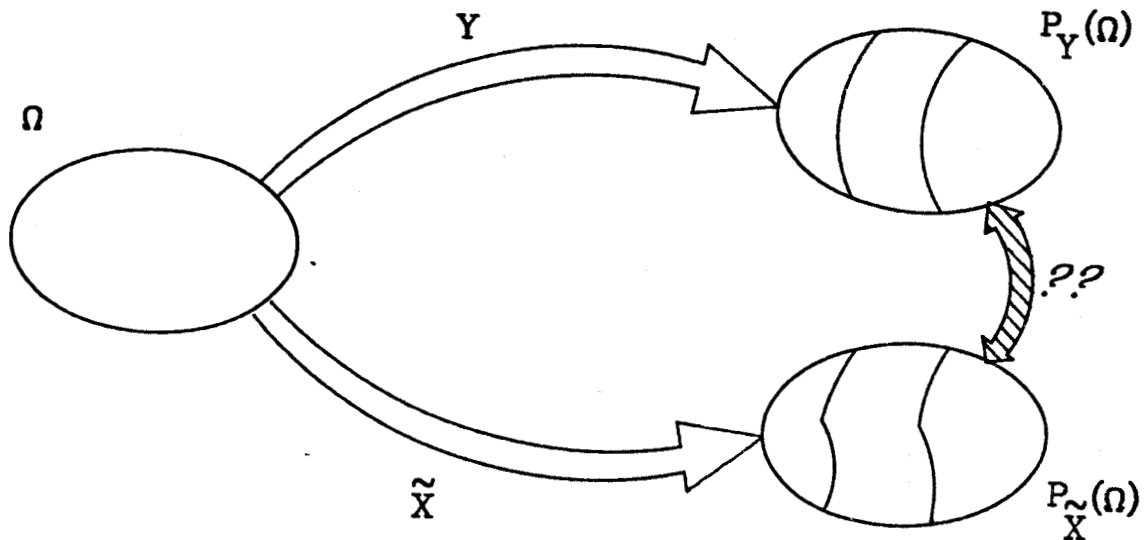


fig.1.12 : Partitions induites par Y et $\tilde{X}$

Par ailleurs, parmi toutes les variables primaires x_i ($i=1, \dots, p$) de X , certaines n'apporteront aucune information sur Y . On ne peut par conséquent les considérer comme des variables explicatives de Y . V.M. Toro Cordoba [TORO 82] a développé des indices, et des algorithmes permettant de sélectionner les k variables les plus explicatives de Y (avec $1 \leq k \leq p$), parmi toutes les variables restantes.

IV.2.2. Génération de règles

La démarche qui vient d'être exposée consiste à construire une variable $\tilde{X}$ telle que $Y=f(\tilde{X})$ dans le cas cohérent, $Y \neq f(\tilde{X})$ (ou $Y=f(S)$ où $S \subset \tilde{X}$) dans le cas incohérent. Il s'agit d'une approche globale du problème de l'apprentissage orienté. Présentons maintenant un point de vue local.

[DID 89,90] définit des objets symboliques particuliers appelés *objets règles*. Dans ce travail, nous ne nous intéresserons qu'à un cas particulier d'objets règles, appelés *clauses de Horn*.

Une *clause de Horn* est un objet symbolique de la forme :

$$r_{A,B} : A \implies B$$

où :

→ A est un objet assertion défini sur A_X , l'ensemble des assertions définies sur X. A est appelée la *prémisse* (ou partie condition) de la règle.

→ B est un événement élémentaire défini sur Y (ce sera ici une disjonction de modalités de Y). B est appelée la *conclusion* de la règle ; et par définition, on dit que la conclusion est *plus générale* que la prémisse (et inversement, que la prémisse est une *spécialisation* de la conclusion).

L'extension de cette règle est définie comme suit :

$$|r_{A,B}|_{\Omega} = \Omega \setminus \{w / A(w) \text{ vraie et } B(w) \text{ faux}\}$$

Remarques :

→ Lorsque le système est caractérisé par un modèle du type $Y=f(\tilde{X})$, celui-ci peut être exprimé sous forme de règles :

$$\tilde{X}=\alpha \implies Y=\beta \quad \text{si } f \text{ est univoque}$$

$$\tilde{X}=\alpha \iff Y=\beta \quad \text{si } f \text{ est biunivoque}$$

Un tel ensemble de règles représente le modèle global liant Y à $\tilde{X}$. En effet, on a dans ce cas : $\forall \alpha \in M_{\tilde{X}} \exists \beta \in M_Y$ tel qu'une des règles ci-dessus est vraie. Cependant, l'existence d'un tel modèle n'est pas nécessaire pour la procédure de génération de règles à partir des données. En effet, il est possible que des règles du type $\tilde{X}=\alpha \implies Y=\beta$ n'existent que pour les modalités α d'un sous-ensemble de $M_{\tilde{X}}$, et qu'aucune relation entre Y et $\tilde{X}$ ne puisse être mise en évidence à l'extérieur de ce sous-ensemble. C'est dans ce sens que le point de vue utilisant le langage des événements est un point de vue local.

→ La représentation de l'objet règle peut facilement être mise en oeuvre par les solutions informatiques de l'Intelligence Artificielle, reposant sur des langages orientés objet :

- utilisation de fonctions *LISP* :

(COND ((A₁)(B₁))...((A_i)(B_i))...)

Si A₁ alors B₁ Sinon ... Si A_i alors B_i Sinon ...

- utilisation de la représentation *PROLOG* :

B:-A (Si A alors B).

→ L'objet règle $r=[A \Rightarrow B]$ est équivalent à l'assertion $[\neg A \vee B]$ (ou encore à l'assertion fausse $\neg(A \wedge \neg B)$).

Ceci revient à dire que pour tout $w \in \Omega$, si $X(w) \notin A$ ou/et si $Y(w) \in B$, alors $r_{A,B}(w)$ est considérée vraie, et fausse sinon.

→ Un ensemble de règles est encore appelé une *description* (ou règle de classification).

IV.3. Conclusion

Dans ce chapitre, nous avons présenté la formulation des problèmes d'apprentissage automatique que nous nous proposons d'étudier. Nous avons introduit les différents types de variables que nous serons amenés à considérer, ainsi que les deux modes de représentation des données que nous utiliserons :

→ La représentation globale considère chaque variable par le biais de la partition de l'ensemble d'apprentissage qui lui est associée. Utilisant la relation de finesse et une notion de distance entre partition, cette représentation conduit à développer des algorithmes d'apprentissage construisant une variable $\tilde{X}$ telle que $Y \# f(\tilde{X})$.

→ La représentation locale considère les variables à travers leurs événements associés. Elle conduira à la construction d'algorithmes de génération de règles utilisant la notion de généralisation.

Au chapitre suivant, après avoir présenté les outils de la *théorie de l'information* qui permettent de juger de la faisabilité du problème d'explication, nous présenterons deux systèmes d'apprentissage relevant respectivement de l'approche globale et de l'approche locale.

CHAPITRE II :

LES OUTILS
DE
L'APPRENTISSAGE AUTOMATIQUE
APPLIQUE AU PROBLEME
DE L'EXPLICATION

I. INTRODUCTION

De nombreuses méthodes ont été développées dans un but explicatif. Elles se distinguent par la nature des espaces de représentation :

→ Les méthodes statistiques de la Reconnaissance des Formes (pour le problème d'explication) traitent de variables explicatives quantitatives $X (M_X \subset \mathbb{R}^p)$, la variable à expliquer Y étant qualitative. Les principales méthodes statistiques développées sont les suivantes [FENE 79] :

- les méthodes dites *paramétriques* (ou *bayésiennes*) : ce sont des méthodes qui nécessitent un modèle de distribution des probabilités de chaque classe (ce modèle est le plus souvent gaussien).
- la *méthode des plus proches voisins*, méthode non paramétrique.
- la *méthode de construction directe de frontières* séparant les différentes classes à reconnaître.

→ Les méthodes d'explication de l'Analyse des Données [BOUR 89]. Ces méthodes sont des cas particuliers de l'Analyse Canonique, amorcée par H. Hotelling [HOTE 36]. Son but essentiel est d'étudier les relations linéaires existant entre deux groupes de variables quantitatives, observées sur un même ensemble d'individus. J.D. Carroll [CARR 68] en a proposé une généralisation à plus de deux groupes de variables.

→ Les méthodes d'Apprentissage Symbolique traitent, par définition, des objets symboliques. Les structures de ces objets ne peuvent être que qualitatives (nominales, ordinales, structurées). Nous décrirons le système d'apprentissage *INDUCE* (descendant immédiat de AQ) développé par R.S. Michalski, et qui, pour nous, est à la base de nombreux algorithmes d'apprentissage symbolique, comme *CALM* [QUIN 84,87], *GENREG* [RALA 87,88] (que nous décrirons également), et bien d'autres.

→ Enfin, les méthodes d'induction par arbre de décision traitent également de variables ne pouvant posséder que des structures qualitatives. Nous décrirons l'algorithme *ID3* proposé par J.R. Quinlan, basé sur des indices issus de la Théorie de l'Information.

Nous nous limiterons volontairement aux méthodes d'apprentissage symbolique (§IV) et d'induction par arbre de décision (§III), dont l'objet est le traitement de variables qualitatives.

Au chapitre I (§IV.2.1), nous avons abordé le problème de la cohérence ou de l'incohérence des données d'apprentissage.

Lorsque les données sont cohérentes, une relation biunivoque peut être mise en évidence entre X et Y.

Par contre, lorsque les données sont incohérentes, on peut se poser le problème de connaître le *degré de faisabilité* du problème d'explication.

Nous utiliserons pour ce faire les résultats de la *théorie de l'information*, qui est un outil adapté à ce problème.

II. FAISABILITE DU PROBLEME D'EXPLICATION

Après une introduction à la *théorie de l'information*, nous définirons l'*entropie de Shannon*. Cette mesure du degré d'incertitude d'un système physique, et ses propriétés, sont tout à fait adaptées aux trois types de problèmes cités précédemment, à savoir les problèmes de visualisation, de structuration, et d'explication.

Nous définirons notamment une mesure de l'intensité de liaison entre deux variables, qui sera dès lors utilisable en tant que critère (voir paragraphe suivant et chapitre IV) dans des algorithmes d'explication.

II.1. Introduction à la Théorie de l'Information

Le caractère aléatoire inhérent aux processus de transmission de l'information implique l'utilisation des méthodes statistiques pour l'étude de ces processus. Pourtant on ne peut se limiter aux concepts classiques de la Théorie des Probabilités et d'autres notions probabilistes doivent être introduites.

C'est pour pallier à ce problème que la Théorie de l'Information (T.I.) fût amorcée dans les travaux de R.V.L. Hartley [HART 28] et de H. Nyquist [NYQ 24]. Mais cette branche de la Théorie des Probabilités connût son véritable essor vers la fin des années 1940 avec les travaux de C.E. Shannon et W. Weaver [SHAN 48,49]. Cette théorie répondait à des problèmes pratiques de Télécommunications ("Théorie mathématique de la communication"), comme celui de la transmission des messages par une ligne téléphonique (ou un canal hertzien) brouillée. Le domaine militaire, avec les travaux du cybernéticien N. Wiener*, a également contribué au développement de cette théorie.

* Le mot "cybernétique" dérive du mot grec kubernetes, pilote. Il a été créé par Wiener lui-même. La cybernétique est par définition la science de la communication et de la régulation des messages.

De nos jours, elle constitue l'un des outils mathématiques nécessaires à l'étude des processus de commande. De nombreuses communications ont été publiées sur ce thème. W.R. Ashby, R.C. Conant, Mc. Gill sont parmi les initiateurs de la T.I. appliquée à l'Analyse Structurale des Systèmes.

Supposons un système physique pouvant se trouver dans un état quelconque. Ce système est alors caractérisé par un certain degré d'incertitude. Les renseignements obtenus sur ce système sont d'autant plus importants que son incertitude *a priori* est grande. Il paraît donc important de disposer d'une mesure de son degré d'incertitude.

Afin de mieux comprendre cette notion, considérons deux systèmes : une pièce de monnaie bien équilibrée et un dé à six faces non pipé.

La pièce de monnaie, jetée en l'air, peut présenter deux côtés différents : un côté *pile* et un côté *face* équiprobables.

Le dé possède six états potentiels différents, équiprobables entre eux.

On peut alors se poser la question suivante : quel est le système possédant l'incertitude la plus grande ?

Il apparaît logique de dire que le système ayant l'incertitude la plus grande est le dé car son nombre d'états possibles (potentiels) est plus grand.

Nous pourrions donc penser que le degré d'incertitude d'un système physique est caractérisé par le nombre d'états potentiels de ce système. Montrons que la seule connaissance du nombre des états potentiels n'est pas suffisant :

Prenons par exemple un système quelconque ayant les caractéristiques suivantes :

- il peut se trouver dans un état (1) (par exemple, un état de bon fonctionnement), qui a une probabilité *a priori* de 0.99
- il peut également se trouver dans un second état, noté (2) (état de panne ou d'arrêt), qui est réalisé avec une probabilité *a priori* de 0.01.

Le degré d'incertitude sur l'état du système est très faible car il a toutes les chances de se trouver dans l'état (1) beaucoup plus souvent que dans l'état (2).

Nous voyons donc que le degré d'incertitude d'un système physique est déterminé par :

- le nombre d'états potentiels
- la probabilité d'obtention (d'occurrence) de ces états.

Appelons *Entropie* la mesure du degré d'incertitude d'un système physique.

II.2. Définition de l'entropie

Soient Ω un univers d'évènements et A un évènement donné. La réalisation de A apporte une quantité d'information notée $I_\Phi(A)$ [AGGA 74,76].

$$I_\Phi(A) := \Phi_0(p(A)) - \Phi_0(p(\Omega)) \quad \text{où :}$$

- Ω est l'ensemble des évènements possibles
- $p(A)$ est la probabilité d'occurrence de l'évènement A
- Φ est une application telle que $\Phi : [0,1] \longrightarrow \mathbb{R}^+$
et $\Phi(1) \geq 0$.

D'autre part, la réalisation d'un évènement peu probable apporte beaucoup d'information, tandis qu'un évènement fort probable, en apporte peu. $I_\Phi(A)$ sera donc une fonction décroissante des probabilités.

Considérons une partition $\{A_1, A_2, \dots, A_i, \dots, A_n\}$ de l'ensemble des états potentiels. On définit $H_\Phi(X)$ l'information fournie par (ou l'entropie de) la partition :

$$\begin{aligned} H_\Phi(X) &= \sum_{i=1}^n p(A_i) \cdot I_\Phi(p(A_i)) \\ &= \sum_i p(A_i) \cdot \Phi(p(A_i)) - \Phi(1) = \sum_i p_i \Phi_i - \Phi(1) \end{aligned}$$

en notant p_i la probabilité de trouver le système X dans l'état (i) ($i=1, \dots, n$).

De nombreuses entropies ont été définies (cf [BARB 87a], pp.I.27-I.28). Nous nous intéresserons, dans le cadre de ce travail, à deux entropies couramment utilisées :

→ l'entropie du MAX : $\Phi(t) := \left(\frac{1}{t} - 1 \right) \cdot \delta(1-p_M)$

où : $- p_M = \max_i p_i$

- $\delta(t-p_M)$ est l'impulsion de Dirac au point p_M .

L'entropie du MAX est donc définie par : $H_M(X) = 1 - \max_i p_i$

→ l'entropie de Shannon : $\Phi(t) = -\text{Log}(t)$

$$H_S(X) = - \sum_{i=1}^n p_i \cdot \text{Log } p_i = E[-\text{Log } p(X)]$$

où $E[f(x)]$ est la fonction *espérance mathématique* (encore appelée *moyenne pondérée*) de $f(x)$.

Compte tenu de ces définitions, nous pouvons remarquer que l'entropie du MAX correspond à une approche *locale*, contrairement à l'entropie de Shannon, mieux adaptée à un problème du type *global* (dû au signe *somme* Σ qui permet de faire une moyenne sur un domaine donné).

D'autre part, l'entropie de Shannon vérifie certaines propriétés intéressantes que nous examinerons par la suite, justifiant son utilisation comme mesure du degré d'incertitude d'un système.

Remarque :

La base du logarithme peut être quelconque (strictement supérieur à 1). Le changement de base est équivalent à une simple multiplication de l'entropie par un nombre constant. En pratique, on utilise des logarithmes de base 2. On mesure donc l'entropie en unités binaires (bits). Ceci s'accorde bien avec le système binaire de représentation des informations dans les calculateurs. Ultérieurement, le symbole *Log* représentera des logarithmes binaires.

II.3. L'entropie de Shannon

II.3.1. Implications directes de la définition

Reprenons la définition de l'entropie de Shannon :

$$H(X) = - \sum_{i=1}^n p_i \cdot \text{Log } p_i \quad \text{où : } \left[\begin{array}{l} - \text{Log} \equiv \text{Log}_2 \\ - \sum_{i=1}^n p_i = 1 \\ - n \text{ est le nombre d'états possibles du système.} \end{array} \right.$$

1ère implication :

$$\text{Log } n = \underset{p_i}{\text{Max}} \{ H(X) \} \quad \Longleftrightarrow \quad \text{système à } n \text{ états équiprobables}$$

L'entropie d'un système équiprobable est égale au logarithme du nombre d'états possibles :

$$\forall p_i \quad p_i = 1/n \quad (i=1, \dots, n)$$

$$H(X) = - \sum_{i=1}^n p_i \cdot \text{Log } p_i = -n(1/n \cdot \text{Log}(1/n)) = -\text{Log}(1/n) = \text{Log } n$$

On peut facilement montrer que l'entropie est maximale dans ce cas.

2ème implication :

$$H(X) = 0 \quad \Longleftrightarrow \quad \text{Système à état unique}$$

L'entropie d'un évènement certain est nulle :

- pour un seul α donné, on a $p_\alpha=1$: $p_\alpha \cdot \text{Log } p_\alpha = 0$
- $p_i=0 \quad \forall i \neq \alpha$, et par continuité : $\lim_{x \rightarrow 0} x \cdot \text{Log}(x) = 0$.

II.3.2. Entropie d'un système composé

Considérons deux variables, notées X et Y, du système étudié. Chacune de ces variables possède son propre ensemble de modalités observées :

$$M_X = \{\alpha_1, \alpha_2, \dots, \alpha_n\} \quad \text{et} \quad M_Y = \{\beta_1, \beta_2, \dots, \beta_m\}.$$

Supposons également que les probabilités conjointes d'obtention des modalités α_i et β_j soient correctement estimées par les fréquences relatives.

Nous pouvons construire le tableau de contingence suivant :

X \ Y	Y					
	β_1	β_2	...	β_j	...	β_m
α_1				.		
α_2				.		
...				.		
α_i	.	.	.	P_{ij}	.	.
...				.		
α_n				.		
	$P_{.j}$					
	$P_{i.}$					

fig.2.1 : Tableau de contingence $[P_{ij}]$ de X et de Y.

où :

→ X et Y sont deux variables du système. Elles peuvent être, ou non, vectorielles.

→ $\{\alpha_1, \alpha_2, \dots, \alpha_n\}$ et $\{\beta_1, \beta_2, \dots, \beta_m\}$ sont respectivement les ensembles de modalités potentielles de X et de Y.

→ Les p_{ij} ($i=1, \dots, n$; $j=1, \dots, m$) sont les probabilités conjointes de X et de Y.

$$\rightarrow p_{i.} = \sum_{j=1}^m p_{ij} \quad \text{pour } i=1, \dots, n.$$

$$\rightarrow p_{.j} = \sum_{i=1}^n p_{ij} \quad \text{pour } j=1, \dots, m.$$

Par définition de l'entropie de Shannon, nous avons :

$$H(X, Y) = - \sum_{i=1}^n \sum_{j=1}^m p_{ij} \log p_{ij} = E[-\log p(X, Y)]$$

Remarques :

→ Nous pouvons montrer l'inégalité suivante :

$$H(X, Y) \leq H(X) + H(Y)$$

→ En vertu du théorème de multiplication des probabilités des événements indépendants : $P(X, Y) = P(X) \cdot P(Y)$

on a $\log P(X, Y) = \log P(X) + \log P(Y)$

et donc $H(X, Y) = H(X) + H(Y)$

II.3.3. Entropie conditionnelle

Considérons maintenant deux systèmes X et Y dépendants l'un de l'autre. $p(\beta_j/\alpha_i)$ représente la probabilité conditionnelle pour que le système Y se trouve dans l'état β_j lorsque le système X se trouve dans l'état α_i .

Or, l'entropie du système Y lorsque le système X se trouve dans l'état α_i est :

$$H(Y/\alpha_i) = - \sum_{j=1}^m p(\beta_j/\alpha_i) \cdot \log p(\beta_j/\alpha_i).$$

L'entropie totale du système Y est donc :

$$H(Y/X) = \sum_{i=1}^n p_{i.} \cdot H(Y/\alpha_i) = - \sum_{i=1}^n p_{i.} \cdot \sum_{j=1}^m p(\beta_j/\alpha_i) \cdot \log p(\beta_j/\alpha_i).$$

$$H(Y/X) = - \sum_i \sum_j p_{i.} \cdot p(\beta_j/\alpha_i) \cdot \log p(\beta_j/\alpha_i).$$

Remarque :

En vertu de la définition de la probabilité conditionnelle
 ($p(X,Y)=p(X).p(Y/X)$), on démontre que : $H(Y/X) = H(X,Y) - H(X)$

II.3.4. Entropie et Information

II.3.4.1. Position du problème

Nous avons vu que l'entropie d'une variable est la mesure du degré d'incertitude de celle-ci. Il semble *naturel* de dire que l'obtention d'informations sur cette variable diminue son incertitude. La quantité d'informations obtenue sur cette variable peut donc être mesurée par la diminution de son entropie.

Prenons le cas particulier d'une variable X_i ayant une entropie *a priori* $H(X_i)$. Si après obtention d'informations, cette variable est totalement connue, alors son entropie s'annule. Notons $I(X_i:X_i)$ l'information obtenue avec la détermination de X_i :

$$I(X_i:X_i) = H(X_i) - 0 = H(X_i).$$

Ce que nous venons d'énoncer pour la variable X_i peut être généralisé pour tout $S \in P^0(V)$.

Ainsi, $H(S)$ représente une mesure de l'information nécessaire pour connaître S .

Si l'on considère maintenant deux variables X et Y , l'inégalité $H(X,Y) \leq H(X) + H(Y)$ peut être interprétée par le fait que, dans l'expression $H(X) + H(Y)$, on compte deux fois une partie de l'information qui se trouve aussi bien dans X que dans Y .

Afin de mesurer cette quantité d'information, nous définirons la *transinformation interne* [ASHB 65] et de la *transinformation externe*.

II.3.4.2. Transinformation interne

Par définition :

$$\begin{array}{ccc} \dot{I} : P^0(V) & \xrightarrow{\quad} & \mathbb{R} \\ S & \longmapsto & \dot{I} := \sum_{X \in S} H(X) - H(S) \end{array}$$

$\dot{I}$ mesure la quantité d'information qui est transmise dans un groupe de variables.

II.3.4.3. Transinformation externe

Par définition :

$$I : \begin{array}{ccc} P(P^O(V)) & \xrightarrow{\quad} & \mathbb{R} \\ R=\{R_1, R_2, \dots, R_r\} & \xrightarrow{\quad} & I(R) := \sum_{i=1}^r H(R_i) - H(\bigcup_{i=1}^r R_i) \end{array}$$

I mesure la quantité d'information qui est transmise entre plusieurs groupes de variables.

II.3.4.4. Propriétés sur l'entropie et les transinformations

De nombreuses propriétés intéressantes ont été démontrées [SHAN 49], [MILL 63], [ASH 65], [TORO 82]. M. Sbaï [SBAI 83] fait une synthèse de ces propriétés :

$\forall S_1, S_2, S_3 \in P^O(V) ; \quad \forall R, Q \in P(V)$
(où $P(V)$ représente l'ensemble des partitions de V)

Propriété 1 : $H(S_1) \geq 0 ; \quad \dot{I}(S_1) \geq 0 ; \quad H(S_1/S_2) \geq 0 ; \quad I(R) \geq 0$

Propriété 2 : $H(S_1 \cap S_2) + H(S_1 \cup S_2) \leq H(S_1) + H(S_2)$
avec égalité si et seulement si $S_1 \cap S_2 = \phi$

Propriété 3 : $S_1 \subset S_2 \implies \begin{cases} H(S_1) \leq H(S_2) \\ \dot{I}(S_1) \leq \dot{I}(S_2) \\ H(S_3/S_1) \geq H(S_3/S_2) \end{cases}$

Propriété 4 : $R \leq Q \implies I(R) \geq I(Q)$
où : - le 1^{er} signe " $\leq$ " représente la relation "... être plus fine que ..."
- le 2^{ème} signe " $\leq$ ", la relation "... être plus petit que ..."

Propriété 5 : $\forall R=\{R_1, \dots, R_r\} \in P(V)$
 $I(R) = 0 \iff R_1, R_2, \dots, R_r$ sont des sous-ensembles de variables indépendants entre eux.

Propriété 6 : $I(R) + \sum_{i=1}^r \dot{I}(R_i) = \text{Constante} = I(\{X_i\}; i=1, \dots, N)$

Propriété 7 : $\forall R=\{R_1, R_2, \dots, R_i, \dots, R_j, \dots, R_r\} \in P(V)$
et $R'=\{R_1, R_2, \dots, R_i \cup R_j, \dots, R_r\}$,
alors : $I(R') = I(R) - I(R_i, R_j)$.

Propriété 8 : Soit $R'=\{R_1, R_2, \dots, R_i \setminus S, S, \dots, R_r\}$
avec : $S \subset R_i ; S \neq \phi ; S \neq R_i ; |R_i| \geq 2$
alors : $I(R') = I(R) + I(R_i \setminus S, S)$.

Propriété 9 : Principe de conditionnement uniforme [GARN 56] :
 "Si l'on conditionne, avec le même ensemble de variables, tous les termes d'une identité valable pour H =entropie qui contient $H(\cdot)$, $H(\cdot/\cdot)$, I , I_α , $\dot{I}$, $\dot{I}_\beta$, on obtient à nouveau une identité valable pour l'entropie (le double conditionnement étant équivalent au conditionnement avec l'union)".
 où : - $I_\alpha(A:B) := H(A/\alpha) + H(B/\alpha) - H(A \cup B/\alpha)$

$$- \dot{I}_\beta(R) := \sum_{x_i \in V} H(X_i/\beta) - H(R/\beta)$$

$$\forall A, B, \alpha, \beta \in P^0(V) \text{ et } \forall R \in P(V)$$

exemple : $\forall A, B \in P^0(V) : H(A \cup B) = H(A) + H(B) - I(A:B)$.
 On a d'après ce principe : $\forall C \in P^0(V) :$
 $H(A \cup B/C) = H(A/C) + H(B/C) - I_C(A:B)$.

II.4. Application au problème de la visualisation

Le problème de la visualisation de variables primaires consistera à chercher un ensemble S (avec $S \in P^0(V)$) tel qu'il contienne un maximum d'information, c'est-à-dire tel que son entropie soit maximale.

A titre d'exemple, sur le tableau donné en annexe 5 :

$$H(\text{Tail.}) = - \left(\frac{15}{27} \log \frac{15}{27} + \frac{5}{27} \log \frac{5}{27} + \frac{7}{27} \log \frac{7}{27} \right) = 1.43 \text{ bits}$$

(15 modalités "+", 5 modalités "o", et 7 modalités "-" pour une population d'apprentissage composée de 27 individus).

De même :

$$H(\text{Poids}) = 1.46 \text{ bits} \quad (5 "+", 14 "o", 8 "-")$$

$$H(\text{Veloc.}) = 1.58 \text{ bits} \quad (9 "+", 8 "o", 10 "-")$$

$$H(\text{Intell.}) = 1.51 \text{ bits} \quad (6 "+", 13 "o", 8 "-")$$

$$H(\text{Aff.}) = H(\text{Agr.}) \# 1 \text{ bit}$$

$$H(\text{Fonction}) = 1.58 \text{ bits} \quad (8 "Ut", 9 "Ch", 10 "Co")$$

Les variables apportant le plus d'information sont les variables *Fonction* et *Vélocité*. Par contre, les variables *Affection* et *Agressivité* n'apportent que très peu d'information.

II.5. Application au problème de la structuration : recherche de partitions optimales

II.5.1. Minimisation de la transinformation externe [CONA 72]

$\forall R = \{R_1, R_2, \dots, R_r\} \in P(V)$, $I(R)$ mesure la quantité d'information circulant entre les sous-systèmes $R_1, R_2, \dots, R_r$.

$$\text{Or } I(R_1:R_2:\dots:R_r) + \sum_{i=1}^r \dot{I}(R_i) = \text{Constante.}$$

Donc minimiser I (ce qui revient à maximiser le terme $\sum_i i$)

équivaut à choisir des sous-systèmes R_i composés de variables fortement liées.

Ce problème peut donc être vu comme un problème de structuration (ou de décomposition) sur l'ensemble des variables.

II.5.2. Décompositions optimales [TORO 82]

La minimisation de I va nous conduire au cas trivial où toutes les variables du système seront contenues dans une seule et unique classe. En effet, $I(\{X_1, X_2, \dots, X_N\}) = 0$.

Une *décomposition optimale par niveaux* est donc nécessaire. On définit : $\mathbb{P}_r(V) := \{\text{partitions de } V \text{ en } r \text{ classes}\}$, $r=1, 2, \dots, N$.

On cherche alors, pour chaque r ($r=1, \dots, N$), la partition optimale qui minimisera la transinformation externe : Soit M_r^* la décomposition optimale en r classes :

$$I(M_r^*) \leq I(M_r) \quad \forall M_r \in \mathbb{P}_r(V).$$

V.M. Toro Cordoba se penche très largement sur ce problème. Il définit notamment des algorithmes permettant de trouver ces partitions optimales (en se servant de la propriété suivante :

$$I(M_{r-1}^*) \leq I(M_r^*) .$$

II.6. Application au problème de l'explication

L'ensemble des variables V du système étant partitionné en deux classes, pour définir la variable à expliquer Y d'une part et la variable explicative vectorielle X d'autre part, nous déterminerons, dans ce paragraphe, quelques indices intéressants du point de vue de l'explicabilité de Y par X .

II.6.1. Détermination de quelques indices [TORO 82]

L'incohérence des données recueillies implique un manque d'explication de Y par X . En effet, nous ne pourrions pas trouver une application f telle que $Y=f(X)$. En pratique, nous chercherons à l'approcher ($Y \approx f(X)$).

Nous déterminerons alors quelques indices mesurant l'explicabilité (ou la modélisabilité) de Y par X :

Posons tout d'abord quelques définitions :

- V est l'ensemble de toutes les variables du système.
- Y est la variable à expliquer.
- $\emptyset \subset V \setminus \{Y\}$ est l'ensemble des variables servant éventuellement à expliquer Y .

Soit $SEP^0(\theta)$:

- $I(Y:S) := H(Y) + H(S) - H(Y,S) = H(Y) - H(Y/S)$
est l'entropie de Y expliquée par S.
- $H(Y/S) := H(Y,S) - H(S)$
est l'entropie de Y non expliquée par S.
- $\%H(Y/S) := H(Y/S) / H(Y)$ est le pourcentage de
l'entropie de Y non expliquée par S.

Si l'on remplace S par X dans les trois expressions ci-dessus, nous obtenons :

- $H(Y/X) := H(Y,X) - H(X)$
est la partie non explicable de Y.
- $I(Y:X) := H(Y) - H(Y/X)$
est la partie explicable de l'entropie de Y.
- $\%H(Y/X) := H(Y/X) / H(Y)$
est le pourcentage non explicable de l'entropie de Y.
- $1 - \%H(Y/X) = I(Y:X) / H(Y)$
est le pourcentage explicable de l'entropie de Y.

D'autres indices peuvent également être définis :

$$- \%ENEPE(Y/S) := \begin{cases} * & \frac{H(Y/S) - H(Y/X)}{I(Y:X)} & \text{si } I(Y:X) > 0 \\ * & 0 & \text{si } I(Y:X) = 0 \end{cases}$$

représente le pourcentage encore non expliqué par S de la partie explicable de Y.

II.6.2. $H(Y/X)$: intensité de liaison entre X et Y

Par définition, $H(Y/X) = H(X,Y) - H(X)$.

Nous avons vu que cette quantité représente l'entropie de Y non expliquée par X. Trois cas distincts sont alors susceptibles d'apparaître [BARB 87a] :

1er cas : $H(Y/X) = H(Y)$: X et Y sont indépendants statistiquement. Aucune relation entre ces deux grandeurs ne pourra être mise en évidence.

2ème cas : $H(Y/X) = 0$: X et Y sont totalement liées et il existe une variable $\tilde{X} \geq X$ (au sens large) telle que $Y=f(\tilde{X})$. On peut alors expliquer toutes les modalités de Y, sur tout le domaine M_X .

3ème cas : $0 < H(Y/X) < H(Y)$: X n'explique que partiellement Y. Nous distinguons alors deux causes possibles :

(1) l'absence d'explication à tort : le modèle déterministe $Y=f(X)$ est représentatif du système, mais les données relevées sont biaisées. Ceci peut être dû à la présence d'un bruit important sur les mesures effectuées ou/et au mauvais fonctionnement d'un sous-système.

(2) l'absence d'explication à raison :

- certaines variables explicatives, pertinentes pour l'étude de notre système, ont été omises.
- ou le modèle est suffisamment compliqué pour ne pas être déterministe.

remarque :

Si $H_M(Y/X)=0$ (avec H_M =entropie du Max), alors il existe une partie du domaine M_X sur laquelle $Y=f(\tilde{X})$: l'entropie du Max est plus adaptée à une approche locale du problème.

Nous avons donc défini une mesure de l'intensité de liaison entre X et Y. Elle pourra être utilisée dans des algorithmes d'explication : le système ID3 présenté au paragraphe suivant, les algorithmes développés au chapitre IV, ...

III. LES METHODES D'INDUCTION PAR ARBRE DE DECISION

III.1. Position du problème et critères

Les systèmes d'induction par arbre de décision visent à l'optimisation d'un critère global afin de *spécialiser les hypothèses* [REND 86]. La démarche est la suivante :

On cherche la variable apportant le plus d'information sur les classes à expliquer, puis une deuxième prise parmi les variables restantes, puis une troisième, ...

L'introduction de variables supplémentaires permet de disposer de partitions de plus en plus fines de la population d'apprentissage et donc d'approcher de mieux en mieux la partition associée à la variable à expliquer. Cette introduction trouve cependant ses limites dans les deux points suivants [AYGA 86] :

- l'augmentation de la complexité du modèle d'explication
- la diminution de l'effectif des classes de la partition de la population d'apprentissage.

Plusieurs critères peuvent être utilisés pour définir, à chaque étape, la variable supplémentaire à prendre en compte [MING 89a]. Soit x une variable-test. Nous pouvons construire un *tableau de comptage* croisant les modalités de x et les différentes classes à expliquer (i.e. les modalités observées de Y) :

$M_x \backslash M_y$	β_1	β_2	...	β_j	...	β_m	
α^1	k_{11}	k_{12}	...	k_{1j}	...	k_{1m}	$k_{1.}$
α^2	k_{21}	k_{22}	...	k_{2j}	...	k_{2m}	$k_{2.}$
$\vdots$							
α^i	k_{i1}	k_{i2}	...	k_{ij}	...	k_{im}	$k_{i.}$
$\vdots$							
α^r	k_{r1}	k_{r2}	...	k_{rj}	...	k_{rm}	$k_{r.}$
	$k_{.1}$	$k_{.2}$	...	$k_{.j}$	...	$k_{.m}$	N

fig.2.2 : Tableau de comptage croisant la variable primaire x et Y .

où k_{ij} est le nombre d'occurrences de l'assertion $[x=\alpha^i] \wedge [Y=\beta_j]$,

$$k_{i.} = \sum_j k_{ij} \text{ et } k_{.j} = \sum_i k_{ij}.$$

Ce tableau permet le calcul de plusieurs indices :

i) La mesure du Khi^2 :

La mesure du Khi^2 est connue depuis longtemps, mais elle a été utilisée dans des algorithmes d'induction pour la première fois par [HART 84].

$$\text{Khi}^2 = \sum_i \sum_j \frac{(k_{ij} - E_{ij})^2}{E_{ij}} \quad \text{où} \quad E_{ij} = k_{i.} \cdot k_{.j} / N$$

La variable x apportant le plus d'information sera alors celle qui maximisera la mesure du Khi^2 .

A titre d'exemple, pour le tableau 1 de l'annexe 4 :

$$K_{hi}^2(\text{radiation}) = (0-1.5)^2/1.5 + \dots = 4.29$$

$$K_{hi}^2(\text{menopause}) = (3-2.5)^2/2.5 + \dots = 0.53$$

La variable *radiation* apporte le plus d'information.

ii) L'indice GINI (ou mesure de l'impureté) :

Cet indice a été proposé par [BREI 84].

$$i = \frac{1}{N} \left(\sum_i \sum_j \frac{k_{ij}^2}{k_{i.}} - \sum_j \frac{k_{.j}^2}{N} \right)$$

La variable *x* apportant le plus d'information sera alors celle qui maximisera cet indice.

$$\text{Par exemple, } i(\text{radiation}) = \frac{1}{10} \left(\frac{0^2}{3} + \frac{3^2}{3} + \frac{5^2}{7} + \frac{2^2}{7} \right) - \left(\frac{5^2}{10} + \frac{5^2}{10} \right) = 0.21$$

$$i(\text{menopause}) = 0.03.$$

iii) La mesure d'information de J.R. Quinlan :

Cette mesure est issue de la Théorie de l'Information. Elle est utilisée dans le système ID3, et dans de nombreuses variantes de celui-ci. Il est souvent considéré (et à juste titre) comme l'algorithme type de l'apprentissage numérique. Nous le présentons donc.

III.2. Le Système ID3

Le système ID3 a été développé par J.R. Quinlan [QUIN 83,86,87]. C'est un système d'inférence inductive à partir d'exemples. Et plus particulièrement, c'est une méthode d'apprentissage par arbre de décision (cf [HUNT 66]). Il fait partie de la famille des systèmes d'apprentissage TDIDT (Top-Down Induction of Decision Trees) :

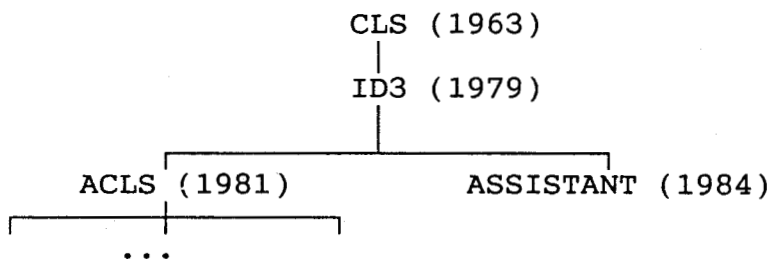


fig.2.3 : famille des TDIDT.

Mais définissons tout d'abord ce qu'est un arbre de décision :

Formellement, un arbre de décision est un arbre [PICA 72] tel que :

- 1- Une feuille (ou noeud réponse) contient un nom de classe.
- 2- Un noeud (qui n'est pas une feuille, ou noeud décision) contient un test sur un attribut avec une branche (donnant naissance à un autre arbre de décision) pour chaque valeur possible de l'attribut en question.

Description de la méthode :

Pour simplifier le problème, on ne considère que deux classes notées P et N. P et N représentent respectivement l'ensemble des instances positives ($p = \text{card } P$) et l'ensemble des instances négatives ($n = \text{card } N$).

L'information moyenne apportée par P et N est :

$$I(p,n) = - \frac{p}{p+n} \text{Log} \frac{p}{p+n} - \frac{n}{p+n} \text{Log} \frac{n}{p+n}$$

Cette quantité représente en fait l'entropie d'une variable à expliquer Y dont l'ensemble des modalités est séparé en deux classes notées N et P.

L'information moyenne apportée par l'arbre ayant x (un attribut ayant les modalités $\{\alpha^1, \alpha^2, \dots, \alpha^r\}$) comme racine est :

$$E(x) = \sum_{i=1}^r \frac{p_i + n_i}{p+n} \cdot I(p_i, n_i)$$

(cette quantité représente l'entropie conditionnelle $H(Y/x)$).

J.R. Quinlan définit alors le gain d'information qu'apporte l'attribut x par :

$$\text{gain}(x) = I(p,n) - E(x).$$

(qui représente la transinformation externe $I(x:Y) = H(Y) - H(Y/x)$).

Le système ID3 examine alors tous les candidats et choisit x qui maximise le gain(x), forme l'arbre à partir de celui-ci, et utilise le même processus récursivement (pour former l'arbre de décision).

Remarque :

Comme le dit J.R. Quinlan [QUIN 86], p.90 (dans une note), maximiser le gain(x) revient à minimiser $E(x)$, car $I(p,n)$ est constant, quel que soit l'attribut x.

Au chapitre IV, nous développerons des algorithmes basés sur la minimisation de l'entropie conditionnelle $H(Y/X)$ où Y est une variable à expliquer, et X est un ensemble de variables explicatives.

Description de l'algorithme ID3 :; SYSTEME ID3

1. Sélectionner un noeud impur n
2. Pour chaque attribut x, calculer la valeur de gain(x)
3. Pour l'attribut x maximisant le gain : étendre l'arbre à partir du noeud n (à chaque modalité de x correspond une branche)
4. Retourner en 1 si il existe encore des noeuds impurs et que toutes les variables non encore utilisées ont un pouvoir discriminant.

; Fin de ID3.

où un noeud *impur* est un noeud ne reconnaissant pas une seule et unique classe.

Exemple :

Prenons le tableau 1 de l'annexe 4 :

Nous pouvons construire les tableaux de comptage suivants :

Class Radiat.	Class		
	Recur	Not Recur	
Yes	0	3	3
No	5	2	7
	5	5	10

Class Menopause	Class		
	Recur	Not Recur	
<60	3	2	5
≥60	1	2	3
Not	1	1	2
	5	5	10

fig.2.4 : Tableaux de comptage

$$I(p,n) = -\left(\frac{5}{10} \log \frac{5}{10} + \frac{5}{10} \log \frac{5}{10}\right) = 1 \text{ bit}$$

Pour la variable *Radiation*, on obtient :

$$E(\text{Radiation}) = -\frac{3}{10} \left(0 + \frac{3}{3} \log \frac{3}{3}\right) - \frac{7}{10} \left(\frac{5}{7} \log \frac{5}{7} + \frac{2}{7} \log \frac{2}{7}\right) = 0.604$$

$$\text{gain}(\text{Radiation}) = 1 - 0.604 = 0.396 \text{ bit}$$

$$\text{de même, } E(\text{Menopause}) = 0.960 \text{ bit}$$

$$\text{gain}(\text{Menopause}) = 1 - 0.960 = 0.040 \text{ bit}$$

La variable *Radiation* apporte plus d'information sur la variable *Class* :

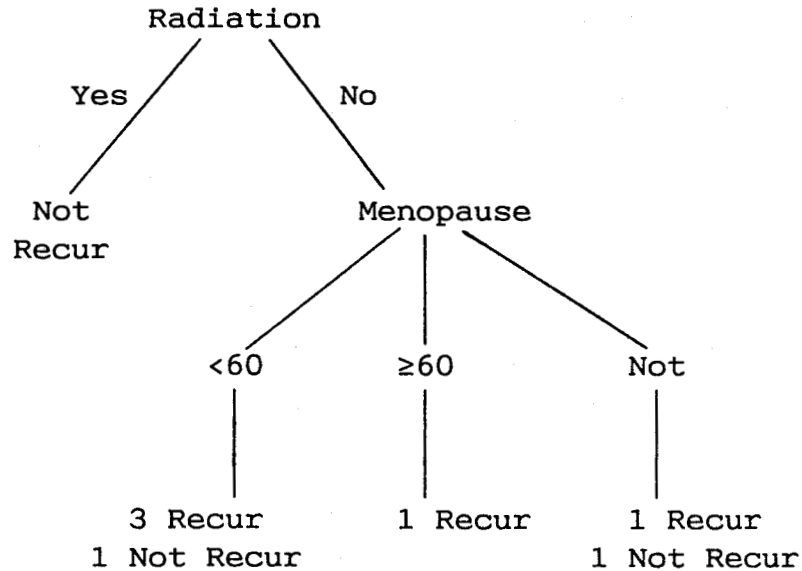


fig.2.5 : Résultat obtenu par le système ID3
sur le tableau donné en annexe 4

III.3. Remarques

→ Ce système s'applique uniquement dans le cas où la population d'apprentissage est cohérente (système déterministe). En effet, dans le cas contraire, à un noeud terminal (une feuille), correspondent plusieurs classes.

→ Il existe des versions *incrémentales* du système ID3. A titre d'exemples, l'algorithme ID4 (Schlimmer & Fisher - 1986), ID5R [UTGO 89], sont des versions incrémentales de ID3. Leur but est d'accepter les instances *incrémentalement*, sans reconstruire un nouvel arbre de décision à chaque fois ; mais en restructurant l'arbre si nécessaire.

→ Il existe d'autres systèmes proches de ID3. A titre d'exemples, citons ACLS (Paterson & Niblett - 1983), ASSISTANT (Kononenko, Bratko, Roskar - 1984).

→ ID3 introduit les variables une à une. Le gain total est donc la somme des gains apportés par chaque variable. Cependant, V.M. Toro [TORO 82] montre que :

$$\text{gain}(x_1, x_2, \dots, x_p) \neq \sum_{i=1}^p \text{gain}(x_i)$$

Dans ce sens, il peut arriver que le pouvoir explicatif de chaque variable d'un vecteur S soit faible (elles ne seront alors pas retenues comme variables discriminantes) alors que le pouvoir explicatif du vecteur S considéré globalement est important. La démarche définie par ID3 sera alors incapable de mettre en évidence la relation entre la variable à expliquer et le vecteur de variables explicatives S .

A titre d'exemple, considérons le tableau suivant :

x_1	x_2	x_3	Y
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

fig.2.6 : un exemple

Nous cherchons à expliquer Y .

$$H(Y/x_1) = H(Y/x_2) = 1$$

$$H(Y/x_3) = 0.69$$

Notre choix va se porter sur x_3 , alors que manifestement, Y est le bit de parité de x_1 et x_2 .

La solution serait de chercher le sous-ensemble de variables expliquant au mieux Y [TORO 82] :

Nous avons $H(Y/(x_1, x_2, x_3)) = 0$, ce qui signifie qu'il existe une relation entre Y et un sous-ensemble (au sens large) de $\{x_1, x_2, x_3\}$.

$$\text{Ainsi, } H(Y/(x_1, x_2)) = 0$$

$$\text{alors que } H(Y/(x_1, x_3)) = H(Y/(x_2, x_3)) = 0.5$$

Notre choix se portera donc sur (x_1, x_2) .

IV. LES METHODES D'APPRENTISSAGE SYMBOLIQUE

Les premiers systèmes d'apprentissage symbolique par détection des similarités (S.B.L. : Similarity Based Learning) ont été développés par l'équipe de R.S. Michalski.

Depuis, de nombreux systèmes traitant symboliquement les données (qui peuvent être de tous les types) sont apparus.

Au lieu de travailler sur les partitions, on va ici travailler directement sur des classes (des pavés, des hordes) de M_X . Nous utilisons donc une approche locale du problème.

Le principe est alors de chercher les classes maximales caractérisées par un indice de qualité mesurant le degré de reconnaissance des modalités de la variable à expliquer.

Après avoir posé plusieurs définitions, nous développerons le système *INDUCE* [MICH 80,83a], descendant directe de *AQ*.

Puis, nous exposerons le système *GENREG* [RALA 87,88].

IV.1. Position du problème et définitions

Les méthodes d'apprentissage symbolique (par détection des similarités) sont des méthodes d'acquisition de concepts sous forme d'une, ou de plusieurs descriptions caractéristiques d'une classe d'objets. Ces descriptions sont appelées *fonctions de reconnaissance* (FR) par R.S. Michalski (encore appelées *assertions*). Les concepts sont appris un à la fois.

Soit $\{\beta_1, \beta_2, \dots, \beta_m\}$ l'ensemble des m concepts.

On définit alors, vis-à-vis d'un concept, deux ensembles distincts :

- $\text{POS}(\beta_j)$ l'ensemble des exemples positifs du concept β_j (ou l'ensemble des instances de β_j).
- $\text{NEG}(\beta_j)$ l'ensemble des contre-exemples (ou exemples négatifs) de β_j .

Remarque :

L'incohérence des données d'apprentissage se traduit par le fait qu'il existe w appartenant à Ω tel que :

$$w \in \text{POS}(\beta_j) \cap \text{POS}(\beta_k) \text{ avec } j \neq k.$$

On cherche alors des couvertures $\text{COV}(\text{POS}(\beta_j) \setminus \text{NEG}(\beta_j))$, i.e. des fonctions de reconnaissance qui soient :

- discriminantes : une description est discriminante si elle ne reconnaît aucun contre-exemple.
- et complètes : une description est complète (ou consistante) si elle reconnaît tous les exemples.

Par convention, lorsqu'une variable prend toutes les modalités possibles (observables) dans une fonction de reconnaissance, on l'omet.

Définitions :

- Soient A et B deux FR. B est dite *plus générale* que A si $A \Rightarrow B$ (on a dès lors $|A|_{\Omega} \subset |B|_{\Omega}$).
- $\text{Gen}(A, B) = A \& B$ est appelée la *généralisée* de A et B . C'est la FR formée des termes communs à A et à B .

IV.2. Le système INDUCE

Décrivons tout d'abord succinctement l'algorithme AQ :

AQ utilise le langage de représentation VL_1 dans lequel la plus petite entité est le sélecteur $[x_i \# V_i]$ où :

- x_i est un nom d'attribut.
- $\# \in \{=, \neq\}$
- V_i est un ensemble de disjonctions de modalités. On parle de disjonctions internes (vis-à-vis d'une variable).

En reprenant le vocabulaire introduit au chapitre I, la plus petite entité est un événement élémentaire.

Une FR (ou formule VL bien formée, ou assertion) est une conjonction de sélecteurs : $[x_1 \# V_1][x_2 \# V_2] \dots [x_k \# V_k]$.

Le procédé inductif de AQ est l'utilisation des deux règles de généralisation suivantes (cf annexe 1) :

→ *the extension against rule* :
 $\Phi \sim [x=V_1] :: >\beta$, $\Phi \sim [x=V_2] :: >\neg\beta \quad | \leq \quad \Phi \sim [x \neq V_2] :: >\beta$
 où $V_1 \cap V_2 = \emptyset$.

→ *the extending reference rule* :
 $\Phi \sim [x=V_1] :: >\beta \quad | \leq \quad \Phi \sim [x=V_2] :: >\beta$
 où $V_1 \subset V_2 \subset M_x$ (ou O_x).

où les métainplications $::>$ et $| \leq$ représentent respectivement la règle de décision (si la partie prémisse est vraie, alors décider la partie conclusion) et la règle de généralisation (la partie conclusion est plus générale que la partie prémisse).

L'algorithme INDUCE :

Cet algorithme est une extension de AQ.

La plus petite entité est le sélecteur $[L \# R]$ où :

- L est une fonction n^{aire} (n =nombre de variables)
- $\# \in \{=, \neq, \leq, \geq, <, >\}$
- R est un ensemble de modalités de L (sous la forme de disjonctions internes).

INDUCE n'utilise qu'une partie du langage VL (formules bien formées, ou well-formed formulas : wff) formée de :

- sélecteurs
- et de formules construites au moyen de conjonctions, disjonctions, disjonctions internes, et du quantificateur existentiel.

Comme procédé inductif, INDUCE utilise :

- les deux règles de généralisation de AQ
 - et la règle suivante (*the dropping condition rule*) :
- $$\Phi \sim [L(x)=v]::>\beta \quad | \leq \quad \Phi::>\beta$$

Le problème consiste alors à trouver la meilleure FR qui reconnaisse les exemples et rejette les contre-exemples, en utilisant les procédés inductifs ci-dessus.

Illustrons cet algorithme par un exemple repris dans [BOUZ 90] (cf annexe 4 pour le tableau des données) :

La FR la plus générale qui reconnaisse e_1 et rejette e_2 est : $G(e_1 \setminus e_2) = (X_2 \neq \text{hexagonale}) \vee (X_3 \neq \text{cannelé})$

La FR la plus générale qui reconnaisse e_1 et rejette e_5 est : $G(e_1 \setminus e_5) = (X_1 \neq \text{moyen})$

Donc, la généralisée de $G(e_1 \setminus e_2)$ et de $G(e_1 \setminus e_5)$ est :

$$\begin{aligned} G(e_1 \setminus \{e_2, e_5\}) &= G(e_1 \setminus e_2) \& G(e_1 \setminus e_5) \\ &= [(X_2 \neq \text{hexagonale}) \vee (X_3 \neq \text{cannelé})] \& (X_1 \neq \text{moyen}) \\ &= [(X_2 \neq \text{hexagonale}) \& (X_1 \neq \text{moyen})] \\ &\quad \vee [(X_3 \neq \text{cannelé}) \& (X_1 \neq \text{moyen})] \\ &= R_1 \vee R_2 \end{aligned}$$

avec $R_1 = [(X_2 = \text{cyl.vcon.vpri.}) \& (X_1 = \text{petitvgrand})]$

et $R_2 = [(X_3 = \text{lissevclaveté}) \& (X_1 = \text{petitvgrand})]$

On se propose alors de trouver la meilleure FR qui reconnaisse e_1 et rejette $\{e_2, e_5\}$: $G_{\text{opt}}(e_1 \setminus \{e_2, e_5\})$.

Pour celà, INDUCE utilise la notion d'éparpillement :

L'éparpillement d'une FR est la différence entre le nombre total d'objets différents qu'elle peut reconnaître et le nombre d'exemples qu'elle reconnaît effectivement.

Le nombre d'objets différents pouvant donner la valeur VRAI à la FR est en fait le produit du nombre de modalités de chaque variable.

$$\text{éparpillement}(R) = \prod_{i=1}^n \text{card}(v_i) - \text{card}(|R|_{\Omega})$$

où :

- n est le nombre total de variables
- v_i est le nombre de modalités de la variable x_i ($i=1, \dots, n$) dans la FR R' , celle-ci étant la FR la moins générale couvrant tous les exemples de R . Si la variable n'apparaît pas dans celle-ci, alors $\text{card}(v_i)=1$
- $|R|_{\Omega}$ est l'extension (dans Ω) de R .

Les FR ne sont pas définies de manière unique. La notion d'éparpillement permet d'effectuer un choix parmi l'ensemble des FR équivalentes en termes de discrimination et de complétude.

La meilleure FR sera alors celle qui possèdera l'éparpillement le plus petit. La seule raison *intuitive* donnée à ce critère est que l'information est plus compacte pour des FR ayant un éparpillement petit [KOD 86], p.90.

Dans notre exemple,

R_1 reconnaît le sous-ensemble $\{e_1, e_3, e_7, e_8, e_9, e_{10}\}$.

La FR R_1' engendrée par ce sous-ensemble s'écrit :

$$\begin{aligned} R_1' &= (X_1 = \text{petitvgrand}) \ \& \ (X_2 = \text{cyl.vcon.vpri.}) \ \& \\ &\hspace{15em} (X_3 = \text{cannelévlissevclaveté}) \\ &= (X_1 = \text{petitvgrand}) \ \& \ (X_2 = \text{cyl.vcon.vpri.}) \\ \text{éparpillement}(R_1') &= 2*3 - 6 = 0. \end{aligned}$$

R_2 reconnaît le sous-ensemble $\{e_1, e_3, e_7, e_8, e_{10}\}$.

$$\begin{aligned} R_2' &= (X_1 = \text{petitvgrand}) \ \& \ (X_2 = \text{cyl.vcon.vpri.}) \ \& \\ &\hspace{15em} (X_3 = \text{lissevclaveté}) \\ &= R_2 \ \& \ (X_2 = \text{cyl.vcon.vpri.}) \\ \text{éparpillement}(R_2') &= 2*3*2 - 5 = 7. \end{aligned}$$

Donc, on obtient :

$$\begin{aligned} G_{\text{opt}}(e_1 \setminus \{e_2, e_5\}) &= R_1' = (X_1 = \text{petitvgrand}) \ \& \\ &\hspace{15em} (X_2 = \text{cyl.vcon.vpri.}) \\ &= (X_1 = \text{petitvgrand}) \ \& \\ &\hspace{10em} (X_2 = \text{révolutionvprismatique}) \end{aligned}$$

L'inconvénient majeur de ce critère est qu'il ne prend pas en compte les positions réelles des objets : en effet, dans une partie de l'espace d'observabilité, il faudrait prendre en compte le fait qu'il peut y avoir des regroupements importants d'objets dans une zone déterminée, et dans d'autres zones, beaucoup moins d'objets. Cette notion d'*importance relative* ne peut être mise en évidence par ce critère.

En fait, une règle d'éparpillement non nul est une règle dont l'obtention a fait intervenir le processus de prolongement que nous introduisons au chapitre suivant (§IV,V) et pour lequel nous proposons une démarche d'évaluation.

IV.3. Le système GENREG

GENREG est un GÉNÉrateur de REGles (ou concepts) sous forme d'assertions, qui décrivent et discriminent les classes d'une partition relative à des données qualitatives, mises sous forme d'un tableau (tableau initial des données).

C'est à la fois une méthode de discrimination, et une technique d'apprentissage à partir d'exemples et de contre-exemples (S.B.L.).

Le tableau des données est constitué d'un ensemble d'exemples (à reconnaître) et d'un ensemble de contre-exemples.

Le problème consiste alors à trouver un ensemble de "bonnes" assertions permettant de décrire et de reconnaître l'ensemble des exemples.

C'est donc bien un problème de discrimination dans lequel Y (la variable à expliquer) est une variable qualitative nominale à deux modalités (exemple et contre-exemple), et X est un ensemble de variables qualitatives.

Comme pour l'algorithme CALM, cet algorithme se décompose en trois grandes étapes :

- (1) Une étape de généralisation, qui consiste à calculer les généralisées en considérant deux à deux les assertions associées aux objets.
- (2) Une étape de sélection, qui consiste à éliminer les assertions insuffisamment discriminantes, i.e. recouvrant un nombre de contre-exemples supérieur à un seuil donné.
- (3) Une étape de compression, qui consiste à résumer les assertions obtenues en un ensemble plus réduit et significatif. Les assertions non valides, i.e. ne reconnaissant pas un nombre d'exemples supérieur à un seuil donné, sont éliminées.

Chaque assertion A retenue doit vérifier les trois conditions suivantes :

- (1) $1 \leq l_{\min} \leq \text{long}(A) \leq l_{\max} \leq p$ où :
 - p est le nombre de variables explicatives.
 - $\text{long}(A)$ est la longueur de l'assertion A : c'est le nombre de variables intervenant dans celle-ci.
 - $l_{\min}$ et $l_{\max}$ sont des constantes imposées par l'utilisateur.
- (2) $\text{card}(|A| \cap \text{NCE}) < \alpha$:

L'assertion A doit être discriminante au seuil α , i.e. le nombre de contre-exemples (CE) reconnus (par l'assertion A) doit être inférieur à α , avec α imposé par l'utilisateur.

(si $\alpha=1$, les assertions retenues ne reconnaissent pas de contre-exemples : ce cas particulier ne peut être utilisé pour des données incohérentes).

(3) $\beta \leq \text{card}(|A|_{\Omega \cap E}) :$

l'assertion A doit reconnaître au minimum β exemples. Cette condition permet d'éliminer des assertions ne reflétant que des faits isolés et garantissant une certaine robustesse des résultats.

Ces conditions étant posées, l'algorithme se décompose en deux grandes étapes :

(i) Calcul des généralisées et élimination des assertions ne vérifiant pas les conditions (1) et (2).

(ii) Élimination des règles particulières et non valides au seuil β (i.e. reconnaissant moins de β exemples).

Le résultat fourni par l'algorithme s'exprime alors sous la forme d'une disjonction d'assertions.

H. Ralambondrainy définit alors deux critères de qualité de l'algorithme :

Soit E_S l'ensemble des exemples reconnus par S (S étant l'ensemble des assertions obtenues).

$$E_S = \bigcup \{ |A|_{\Omega \cap E} / A \in S \}$$

Or, une assertion de S doit reconnaître un minimum d'exemples (au moins β exemples). Il peut alors arriver que des exemples ne soient pas reconnus. On estime alors la représentativité de S par le pourcentage de bien classés (ou pourcentage des exemples reconnus par S) :

$$p_{bc} = \frac{\text{card}(E_S)}{\text{card}(E)} * 100$$

où $\text{card}(E_S)$ représente le nombre d'exemples reconnus par S, et $\text{card}(E)$, le nombre total des exemples.

On remarque que p_{bc} augmente si on augmente le seuil de mal classés α ou/et si on diminue le nombre β d'exemples reconnus par une assertion.

Soit CE_S l'ensemble des contre-exemples reconnus par S.

$$CE_S = \bigcup \{ |A|_{\Omega \cap CE} / A \in S \}$$

On introduit alors la qualité de discrimination de S, par le pourcentage de mal classés (ou encore le pourcentage des contre-exemples reconnus par S) :

$$p_{mc} = \frac{\text{card}(CE_S)}{\text{card}(CE)} * 100$$

où $\text{card}(CE_S)$ est le nombre de contre-exemples reconnus par S, et $\text{card}(CE)$, le nombre total de contre-exemples.

(si $\alpha=1$, les assertions ne reconnaissent aucun contre-exemple, et $p_{mc}=0$).

S est dit optimal lorsque p_{bc} est maximal et p_{mc} est minimal.

IV.4. Les autres systèmes

Il existe de nombreux autres systèmes d'apprentissage symbolique. Nous ne citerons que quelques uns d'entre eux :

- CALM [QUIN 84,87], un système d'apprentissage par points de vue.
- CHARADE [GANA 87a,b], un système d'apprentissage basé sur la notion de *cube de Hilbert*.
- PLAGE [GASC 86,87].
- LEGAL et INNE [LIQ 89,90], basés sur la construction du treillis de Galois [NORR 78] [BORD 86] [GUEN 89] d'un ensemble d'objets.
- UNIMEM [LEBO 87]
- CN2 [CLAR 89], qui induit des règles sous la forme d'une conjonction de sélecteurs, associées à une description des classes utilisant des techniques de réduction logique (système inspiré de AQ et ID3).

De nombreuses méthodes incrémentales ont également été développées [DEC 89a,b] [GENN 89] [SEBA 89].

Plus récemment, une autre famille de systèmes est apparue : l'apprentissage automatique génétique. Nous renvoyons le lecteur, désireux de connaître ces systèmes, à [WILS 87] [QUIN 88] [PARO 91].

V. CONCLUSION

Nous avons vu que les méthodes d'apprentissage symbolique cherchaient directement une description de l'ensemble des modalités associées à chaque β_j ($j=1, \dots, m$). On construit ainsi des "pavés" en utilisant des règles de généralisation. C'est bien là une approche locale du problème d'explication. L'intérêt principal de ces méthodes est qu'elles fournissent des résultats directement compréhensibles par des experts humains.

Leur inconvénient majeur est la non robustesse des résultats. En effet, ces méthodes sont très sensibles au bruit, et ne sont pas applicables si la population d'apprentissage est fortement incohérente.

Une autre approche utilise des méthodes numériques, telles que les méthodes d'induction par arbre de décision, dont les principes ont été énoncés dans ce chapitre.

Ces méthodes construisent une variable vectorielle S ($S \in P^0(\bar{X})$) en introduisant les variables primaires une par une. Ce sont donc

des méthodes de recherche de partitions des modalités observées de X , basées sur un critère *global*. L'avantage majeur de ces méthodes est qu'elles sont moins sensibles au bruit.

Les règles générées à partir des méthodes d'induction par arbre de décision construisent une partition de la population d'apprentissage. Les méthodes de l'apprentissage automatique symbolique, construisant des couvertures de parties de O , aboutissent à un recouvrement de cet ensemble.

Les critères permettant d'évaluer la qualité du système explicatif obtenu s'expriment sous des formes très diverses (information, pourcentage de bien/mal classés, éparpillement, ...). Dans le chapitre suivant, nous proposons une analyse du modèle obtenu par apprentissage automatique, de façon à guider son évaluation.

CHAPITRE III :

EVALUATION DU MODELE
OBTENU PAR
APPRENTISSAGE AUTOMATIQUE

Quelle que soit la démarche retenue, la procédure d'apprentissage construit un ensemble de règles liant les modalités des variables à expliquer et des variables explicatives. Dans ce chapitre, nous définirons des critères permettant d'évaluer le modèle obtenu ; ceux-ci seront utilisables dans des algorithmes d'apprentissage.

Nous caractériserons tout d'abord une règle par rapport aux observations effectuées : règles *toujours vraies*, de *dépendance perçue*, ...

Puis, nous caractériserons un ensemble R de règles du point de vue de leur cohérence (§II.2).

Nous donnerons une mesure de l'adéquation entre le modèle obtenu et les observations (le *degré d'impureté* d'une règle non toujours vraie, ou *taux de pollution*), et nous évaluerons la maniabilité de ce modèle (efficacité, généralité, couverture) (§III).

Une procédure d'apprentissage produit des règles plus générales que les règles initiales. Deux opérations de généralisation seront envisagées : l'agrégation et le prolongement. Des indices caractériseront alors les règles obtenues.

I. POSITION DU PROBLEME

I.1. Règles toujours vraies et dépendance perçue

La définition d'une règle introduite au chapitre II est une définition *locale*, dans le sens où pour chaque w ($w \in \Omega$), il lui correspond une valeur *vrai* ou *faux*.

Nous introduisons alors une définition supplémentaire permettant de prendre en compte le fait qu'une règle peut être toujours vraie, ou pas :

Définition :

Une règle $r_{A,B}$ est dite *toujours vraie* si et seulement si :

$$|r_{A,B}|_{\Omega} = \Omega.$$

Elle est dite *non toujours vraie* dans le cas contraire.

Autrement dit, une règle est toujours vraie si et seulement si il n'existe pas d'observation w ($w \in \Omega$) telle que $r_{A,B}(w)$ soit fausse.

Posons-nous le problème suivant :

Si on considère une règle $r=[A \Rightarrow B]$ définie comme ci-dessus, et si l'assertion A n'a jamais été vraie (il n'existe aucun point observé dans le domaine A), alors cette règle sera toujours vraie, et ceci, quel que soit w appartenant à Ω .

En d'autres termes, ceci revient à se poser la question suivante : Si il n'existe aucun point observé dans le domaine A , peut-on avoir une relation de dépendance entre X et Y du type $A \Rightarrow B$?

C'est le problème de la *dépendance perçue* décrit par Ch. de Sainte Marie [SAIN 89,90] : "j'ai déjà observé le fait A (donc je sais qu'il est pertinent), et toutes les fois que je l'ai observé, j'ai aussi observé le fait B ; tout se passe donc comme si A dépendait de B".

Il est alors nécessaire d'apporter une contrainte sur l'assertion A afin de distinguer les cas où une règle est réellement observée, ou pas :

On dira qu'une règle $r=[A \Rightarrow B]$ est de *dépendance perçue* si et seulement si :

- $r_{A,B}$ (ou $\neg A \vee B$) est toujours vraie
- et si $|A|_{\Omega} \neq \emptyset$ (i.e. l'assertion A a déjà été vraie au moins une fois).

Prenant en compte les notions de règles toujours vraies et de *dépendance perçue*, nous pouvons distinguer quatre types de règles :

$r_{A,B}$	$ r_{A,B} _{\Omega} = \Omega$	$r_{A,B}$	$ r_{A,B} _{\Omega} \neq \Omega$
$ A _{\Omega} \neq \emptyset$	<i>dépendance perçue</i>	$ r_{A,B} _{\Omega} \neq \emptyset$	<i>non toujours vraie</i>
$ A _{\Omega} = \emptyset$	<i>non observée</i>	$ r_{A,B} _{\Omega} = \emptyset$	<i>toujours fausse</i>

fig.3.1 : Les différents types de règles

→ Une règle $r_{A,B}$ est de *dépendance perçue* si et seulement si elle est toujours vraie et il existe au moins un point observé dans le domaine A.

→ Une règle est *non observée* si et seulement si le domaine A ne couvre aucun point observé. La conclusion de la règle peut alors être quelconque. Nous introduirons la notion de *règle estimée* qui permettra de sélectionner telle ou telle conclusion (§IV.3).

On impose une condition plus forte concernant la notion de règle *non toujours vraie* :

→ Une règle $r_{A,B}$ est *non toujours vraie* si et seulement si on a observé au moins une fois la règle $r_{A,B}$ et il existe au moins un point observé du domaine A qui possède une conclusion n'appartenant pas à B.

→ Une règle est *toujours fausse* si et seulement si tous les points observés du domaine A possèdent une conclusion n'appartenant pas à B.

On peut représenter les différents types de règles par les schémas suivants :

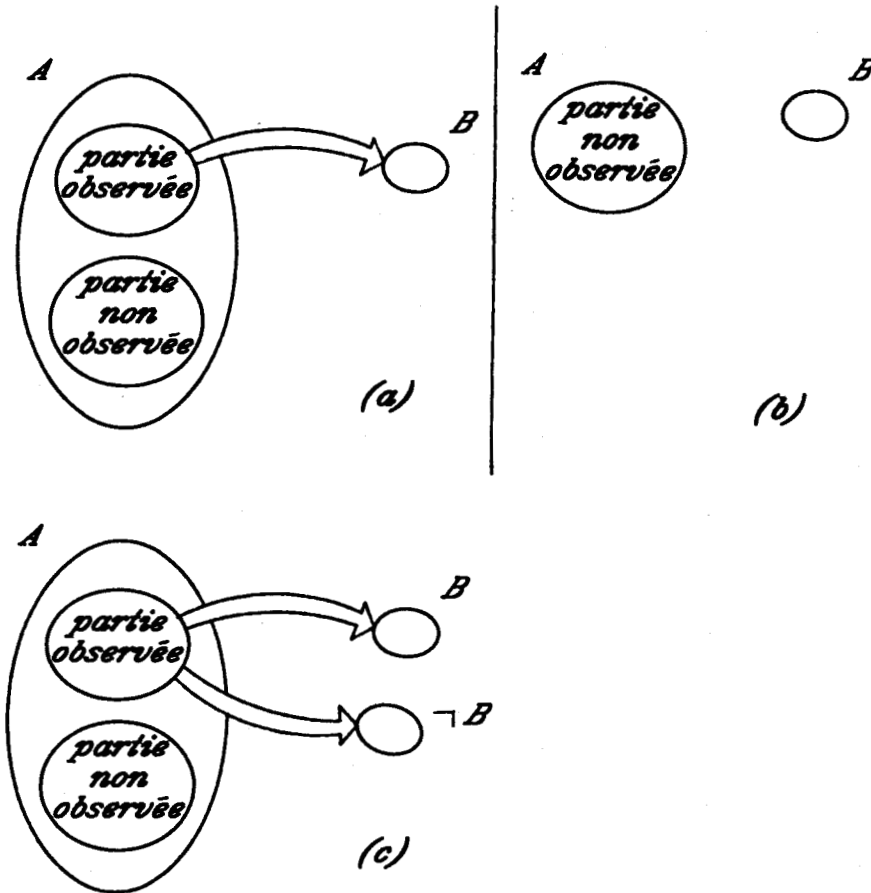


fig.3.2 : → Règles toujours vraies :

(a) de dépendance perçue

(b) non observée

→ Règle non toujours vraie : (c)

Dans les cas (a) et (c), les parties non observées peuvent être vides.

I.2. La procédure d'apprentissage

Notons R_Σ l'ensemble des règles $r_{A,B}$ telles que :

- A est un objet assertion de A_X , l'ensemble des assertions définies sur X.
- B est un événement élémentaire défini sur M_Y .

Définissons $R_\Sigma^*(\alpha) = \{ r_{A,B} \in R_\Sigma / \text{card } |r_{A,B}|_\Omega \geq \alpha \}$

$R_\Sigma^*(\alpha)$ représente l'ensemble des règles qui reconnaissent au moins α observations.

Dans le cas particulier où $\alpha=1$, $R_\Sigma^*(1)$ représente l'ensemble des règles dont l'extension n'est pas vide.

Supposons que la procédure d'apprentissage produise un ensemble R de règles décrivant le comportement du système observé.

En tenant compte de la notion de "dépendance perçue" définie par Ch. de Sainte Marie (cf §I.1), nous pouvons imposer la contrainte suivante : chaque règle de R doit reconnaître au moins α observations (ou α exemples). L'ensemble R représente alors un sous-ensemble de $R_{\Sigma}^*(\alpha)$.

II. PROPRIETES SUR UN ENSEMBLE DE REGLES

II.1. Complétude sur un ensemble de variables

Soit R un ensemble de règles constituant un modèle possible pour le système décrit par la population d'apprentissage.

R est dit complet sur S ($S \in P^0(V)$) si et seulement si toutes les modalités observées de S sont couvertes par l'ensemble des règles de R .

En particulier, R est complet sur X si et seulement si :

$$M_X \subset \bigcup_{r_{A,B} \in R} A$$

(caractérisation de la partie prémisse de la règle).

De même, R est complet sur Y si et seulement si :

$$M_Y \subset \bigcup_{r_{A,B} \in R} B$$

(caractérisation de la partie conclusion de la règle).

Nous avons vu précédemment, que certaines variables apportaient plus d'information que d'autres. On a donc tout intérêt à ce que R soit complet sur les variables apportant le plus d'information. C'est typiquement le cas de l'algorithme *ID3* de J.R. Quinlan (cf chapitre II) : le premier noeud (ou racine) de l'arbre de décision est un test sur les modalités observées de la variable apportant le plus d'information. Chaque modalité observée engendre une branche et un noeud. Si ce dernier est impur, on teste les variables apportant le plus d'information parmi toutes celles qui restent, etc... Le système de règles qui en résulte est alors complet sur la variable apportant le plus d'information, etc... ; et à la limite, les variables n'apportant que très peu d'information ne sont pas prises en compte.

On peut alors définir un degré de complétude pour S . Par exemple, on pourrait prendre le rapport du nombre de modalités observées de S couvertes par R sur le nombre de modalités réellement observées de S .

II.2. Cohérence

Une condition nécessaire pour qu'un système de règles R issues d'une procédure d'apprentissage soit utilisable, est la cohérence des règles, i.e. les règles obtenues ne doivent pas se contredire. Nous définirons dès lors plusieurs niveaux de cohérence des règles [POM 91], suivant que l'incohérence provient d'un point observé, observable, ou non observable.

II.2.1. Cohérence logique

L'ensemble $R = \{r_{A_i, B_i}, i \in I_R\}$ est dit "logiquement cohérent" si et seulement si l'une des deux conditions suivantes est vérifiée :

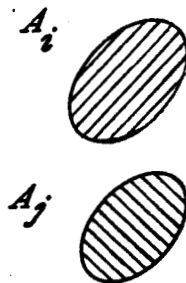
- $A_i \cap A_j = \emptyset \quad \forall i, j \in I_R$ (cohérence de type 1)
- $\forall i \in I_R$ tel que $\bigcap_{i \in I} A_i \neq \emptyset$

Quelle que soit la règle $r_{a,b}$ telle que $\exists k \in I_R \quad a \in A_k, b \in B_k$

$$a \subset \bigcap_{i \in I} A_i \implies b \subset \bigcap_{i \in I} B_i \quad (\text{cohérence de type 2})$$

Graphiquement, ce type de cohérence peut se représenter de la façon suivante :

cohérence de type 1 :



avec $B_i \cap B_j = \emptyset$
ou $B_i \cap B_j \neq \emptyset$

cohérence de type 2 :

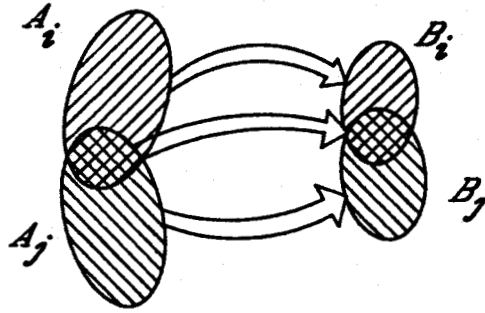


fig.3.3 : Cohérence logique

A titre d'exemple, en considérant le tableau de l'annexe 5, les deux règles suivantes :

[Poids = +] $\implies$ [Fonction=Ut]

et [Taille = +] & [Poids = 0 v +] $\implies$ [Fonction=UtvCovCh]
sont en cohérence logique de type 2.

On peut remarquer que tout système cohérent de type 2 peut être transformé en un système cohérent de type 1, via un partitionnement adéquat de l'ensemble $\bigcup_{i \in I_R} A_i$, comme il est montré sur l'exemple

suivant :

$R = \{r_{A1, B1}, r_{A2, B2}\}$ est un ensemble de règles cohérent de type 2.

Soient $a_1 = A_1 \setminus A_2$; $a_2 = A_2 \setminus A_1$; $a_3 = A_1 \cap A_2$

Il existe b_1, b_2, b_3 tels que $b_3 = B_1 \cap B_2$ et :

$R' = \{r_{a1, b1 \cup b3}, r_{a2, b2 \cup b3}, r_{a3, b3}\}$ est un ensemble de règles cohérent de type 1.

On peut définir l'incohérence logique d'un système de règles R :

On dit qu'un système est logiquement incohérent si :

- il existe une observation w pour laquelle $X(w) \in A_i \cap A_j$ et $Y(w) \in M_Y \setminus (B_i \cap B_j)$ (avec $B_i \cap B_j = \emptyset$ ou $\neq \emptyset$)
- ou il existe une intersection $A_i \cap A_j \neq \emptyset$ avec $A_i \cap A_j \cap M_X = \emptyset$ (ensemble de points non observés) et $B_i \cap B_j = \emptyset$.

Dans le premier cas, il existe au moins une règle non toujours vraie. On pourra alors lui associer un coefficient de vérité, ou taux de pollution (§III.2.2), à chaque règle.

Dans le deuxième cas, les prémisses des deux règles se recouvrent partiellement, mais leur intersection n'a pas été observée. Ces règles sont incohérentes entre elles dans le sens où, lors d'une

observation ultérieure du système, il peut se poser un problème de décision si la partie explicative de l'observation appartient à cette intersection.

Nous définissons dès lors plusieurs types d'incohérences, développées au paragraphe suivant.

II.2.2. Cohérence perçue

Considérons un ensemble de règles qui n'est pas logiquement cohérent. En tenant compte de la notion d'observabilité, nous pouvons distinguer trois types d'incohérences différentes :

(1) COHERENCE ABSOLUE :

On dit qu'un système de règles est en cohérence absolue si et seulement si la condition suivante est réalisée :

quelque soit α (α appartenant à $\prod_{i=1}^P O_i$) provoquant une incohérence logique, alors $\alpha \in \prod_{i=1}^P O_i \setminus O$; i.e. α est non observable.

Un système de règles R est donc en cohérence absolue s'il vérifie la propriété suivante :

$\forall A_i \cap A_j \neq \emptyset$ provoquant une incohérence logique, alors :

$$A_i \cap A_j \subset \prod_{i=1}^P O_i \setminus O$$

remarque :

Si la procédure d'apprentissage produit deux règles r_{A_i, B_i} et r_{A_j, B_j} en cohérence absolue avec $B_i \cap B_j \neq \emptyset$, alors on peut toujours transformer celles-ci en deux autres règles r_{A_i, b_i} et r_{A_j, b_j} , avec $b_i = B_i \setminus (B_i \cap B_j)$ et $b_j = B_j \setminus (B_i \cap B_j)$.

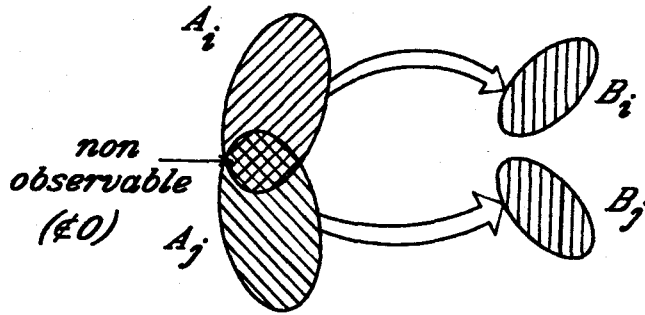


fig.3.4 : Cohérence absolue

L'incohérence de l'ensemble des règles ne peut être prouvée. En effet, toute intersection non vide des prémisses des règles de R provoquant une incohérence logique ne contient que des modalités non observables, donc qui ne seront jamais observées.

L'idée de "cohérence absolue" est liée à la notion d'observabilité sur les variables explicatives. Seule une connaissance "a priori" du système permet de déterminer cette observabilité.

A titre d'exemple, en considérant le tableau de l'annexe 5, les deux règles suivantes :

[Taille= + v o v -]&[Poids = -] $\Rightarrow$ [Fonction=CovCh]

et [Taille = +]&[Poids = - v o v +] $\Rightarrow$ [Fonction=UtvCovCh]

peuvent être considérées comme étant en cohérence absolue si on impose la contrainte : "Les chiens ayant une taille + ne peuvent pas avoir un poids -". Autrement dit, une taille + et un poids - n'est pas observable.

(2) COHERENCE PERÇUE :

On dit qu'un système de règles est en cohérence perçue si et seulement si la condition suivante est réalisée :

quelque soit α (α appartenant à $\prod_{i=1}^P O_i$) provoquant une incohérence logique, alors $\alpha \in \prod_{i=1}^P O_i \setminus M_X$; i.e. α est non observé.

Un système de règles R est donc en cohérence perçue s'il vérifie la propriété suivante :

$\forall A_i \cap A_j \neq \emptyset$ provoquant une incohérence logique, alors :

$$A_i \cap A_j \subset \prod_{i=1}^P O_i \setminus M_X$$

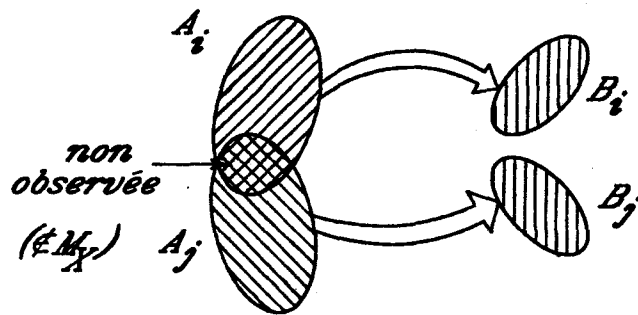


fig.3.5 : Cohérence perçue

Toute intersection non vide des prémisses des règles de R provoquant une incohérence logique ne contient aucune modalité observée de X . L'incohérence de l'ensemble des règles n'a donc pas été révélée par les données observées ; mais il n'est pas impossible qu'ultérieurement (lors d'une future observation du système), elle le soit.

On pourra également dire "cohérence relativement aux données", "cohérence observée", ou "cohérence empirique".

La notion de "cohérence perçue" est liée à l'observation du système (cohérence par rapport aux données). Elle s'applique, que nous disposions, ou non, de la connaissance du domaine d'observabilité O sur X . On tiendra donc plus facilement compte de ce type d'incohérence que de l'incohérence absolue.

A titre d'exemple, en considérant le tableau de l'annexe 5, les deux règles suivantes :

$[Taille = + \vee o \vee -] \& [Poids = -] \implies [Fonction = CovCh]$

et $[Taille = +] \& [Poids = - \vee o \vee +] \implies [Fonction = UtvCovCh]$

sont en cohérence perçue. En effet, une taille + et un poids - n'ont jamais été observés.

(3) INCOHERENCE PERÇUE (ou totale) :

Lorsque R est un ensemble de règles logiquement incohérent, il est dit en "incohérence perçue" si et seulement si :

il existe $A_i \cap A_j \neq \emptyset$ provoquant une incohérence logique, tels que :

$$A_i \cap A_j \subset M_X.$$

En d'autres termes, R est dit en incohérence perçue (par rapport aux données d'origine) si et seulement si :

il existe A_i, A_j appartenant à R , et il existe w appartenant à Ω tels que $X(w) \in A_i \cap A_j$ et $Y(w) \in M_Y \setminus (B_i \cap B_j)$.
(avec $B_i \cap B_j = \emptyset$ ou $\neq \emptyset$).

Dans le cas où $B_i \cap B_j = \emptyset$, l'ensemble des observations effectuées sur le système n'est pas entièrement recouvert par R .

A titre d'exemple, en considérant le tableau de l'annexe 5, les règles suivantes :

[Taille=+] $\implies$ [Fonction=Ut]

[Taille=+]&[Poids=0] $\implies$ [Fonction=Ch]

sont en incohérence perçue, provoquée par (10).

En fait, un ensemble de règles R est en incohérence perçue si et seulement si il existe au moins une règle non toujours vraie : en effet, une règle non toujours vraie sera toujours en incohérence avec elle-même.

On définira alors une mesure du degré de l'incohérence perçue, ou *taux de pollution* (cf §III.2.2).

II.3. Classification des modèles obtenus par la procédure d'apprentissage

Les modèles obtenus sont caractérisés par deux variables binaires qui qualifient leur complétude et leur cohérence perçue.

Les implications suivantes sont vérifiées :

$$\left\{ \begin{array}{l} \text{cohérence} \\ \text{logique} \end{array} \right\} \implies \left\{ \begin{array}{l} \text{cohérence} \\ \text{absolue} \end{array} \right\} \implies \left\{ \begin{array}{l} \text{cohérence} \\ \text{perçue} \end{array} \right\}$$

Nous donnons, dans l'annexe 3, une classification des modèles obtenus par une procédure d'apprentissage, par rapport au type de cohérence voulue.

A titre d'exemple, la création d'une partition sur l'ensemble O (ensemble des modalités observables de X) avec une complétude sur Σ (toutes les modalités observées de Σ sont prises en compte), nous assure une cohérence absolue des règles obtenues.

Si l'on impose la création d'une partition sur l'ensemble M_X (ensemble des modalités observées de X) avec une complétude sur Σ , alors on est au moins sûr d'obtenir un ensemble de règles en cohérence perçue.

La procédure aura donc pour but de produire des ensembles de règles qui seront à la fois complètes (sur Σ ou sur X) et en cohérence perçue. les considérations précédentes montrent qu'elles vérifient :

$\rightarrow \{A_i, i \in I_R\}$ constitue une partition de M_X .
 $\rightarrow \{A_i, i \in I_R\}$ peut constituer une couverture de M_X^* , de O , de $\prod_{i=1}^p O_i$ ou de $\prod_{i=1}^p M_{X_i}$.

Des simplifications peuvent alors apparaître dans l'expression du modèle si on exploite les deux constatations précédentes.

III. EVALUATION DES REGLES

L'évaluation des règles se fera par le biais de différentes mesures (notées Φ) sur l'ensemble R_Σ :

$$\Phi : \begin{array}{ccc} R_\Sigma & \longrightarrow & \mathbb{R} \\ r_{A,B} & \longmapsto & \Phi(r_{A,B}) \end{array}$$

où A et B sont définis respectivement sur $P(\prod_{i=1}^p O_i)$ et sur $P(M_Y)$.

Par exemple $\Phi(r_{A,B}) = \text{card } |r_{A,M_Y}|_\Omega \cap |r_{M_X,B}|_\Omega$ représente la fréquence de l'évènement " $X \in A$ et $Y \in B$ ".

III.1. Relations d'ordre

Nous avons introduit (chapitre I, §III.2) une relation d'ordre sur l'ensemble A_Σ des assertions définies sur $P(\prod_{i=1}^p O_i).P(M_Y)$:

Une assertion a_Y est plus fine qu'une assertion $a_{Y'}$ ($a_Y \leq a_{Y'}$) si et seulement si $V \subset V'$.

En considérant la restriction de cette relation à $P(\prod_{i=1}^p O_i)$, on

définira la notion de généralité d'une règle (relation sur les prémisses) ; la restriction à $P(M_Y)$ permettra, quant à elle, de définir la notion d'efficacité (relation sur les conclusions).

GENERALITE d'une règle :

La règle $r_{A,B}$ est plus générale que la règle $r_{A',B'}$ si et seulement si $A' \subset A$.

$r_{A,B}$ couvre donc au moins autant d'observations que $r_{A',B'}$. En effet, le domaine $A \setminus A'$ peut contenir des observations, ou pas.

Nous pouvons définir une mesure Φ associée à la prémisse de la règle considérée (degré de généralité), avec :

$$A' \subset A \implies \Phi(r_{A,B}) \geq \Phi(r_{A',B'})$$

Par exemple, $\Phi(A,B)$ peut représenter le nombre de modalités observées de X couvertes par le domaine A , ou $\text{card } |A|_{\Omega}$.

La relation de généralité définit un ordre partiel sur l'ensemble des règles R_{Σ} .

EFFICACITE d'une règle :

La règle $r_{A,B}$ est plus efficace que la règle $r_{A',B'}$ si et seulement si $B \subset B'$.

L'efficacité d'une règle peut être évaluée par une mesure Φ de sa conclusion, telle que :

$$\begin{aligned} &\forall r_{A,B}, r_{A',B'} \text{ deux règles de } R_{\Sigma} : \\ &\quad \text{si } B \subset B' \text{ alors } \Phi(r_{A,B}) \geq \Phi(r_{A',B'}) \end{aligned}$$

Par exemple, M. Barboucha propose une mesure d'efficacité [BARB 87a,b], basée sur la théorie de l'information.

$$\Phi(r_{A',B}) = 1 - \frac{\text{Log}(\text{card } B)}{\text{Log } m}$$

où $\text{Log } m$ est l'entropie (de Shannon) maximale de Y .

- Lorsque $\text{card } B = 1$ (une conclusion), l'efficacité est maximale.
- Lorsque $\text{card } B = m$ ($m = \text{card } M_Y$), l'efficacité est nulle.

La relation d'efficacité définit également un ordre partiel sur l'ensemble R_{Σ} .

A partir de ces deux définitions, nous introduisons la relation de "préférence" :

Une règle $r_{A,B}$ est préférée à une règle $r_{A',B'}$ si et seulement si $r_{A,B}$ est :

- plus générale ($A' \subset A$)
- et plus efficace ($B \subset B'$)

Une mesure de préférence peut être définie par une application Φ telle que : $r_{A,B}$ est préférée à $r_{A',B'} \implies \Phi(r_{A,B}) \geq \Phi(r_{A',B'})$

Remarques :

→ Plus une règle est générale, et plus son efficacité diminue. En effet, plus on agrandit le domaine associé à la prémisse de la règle (règle plus générale), et plus on risque d'englober des contre-exemples (règle moins efficace).

Pour pallier à ce problème, nous introduirons une procédure de contraste (chapitre IV) qui nous permettra de filtrer les données.

→ Les notions de *généralité* et d'*efficacité* sont des notions bien connues en apprentissage symbolique. A titre d'exemples, nous les trouvons dans [WILL 82] [DUQ 86] [LIQ 90] (Analyse Formelle des Concepts / treillis de Galois), [MITC 82] (espace des versions), ..., où :

$r_{A,B}$ est appelée une généralisation de $r_{A',B'}$
et $r_{A',B'}$, une spécialisation de $r_{A,B}$.

III.2. Indices caractéristiques (*critères locaux*)

Soit une règle $r_{A,B}$ ($r_{A,B} \in R_\Sigma$).

A et B étant donnés, on peut considérer X et Y comme deux variables dichotomiques, qui induisent chacune une partition en deux classes de la population d'apprentissage Ω :

→ pour X : ($|A|_\Omega, |\neg A|_\Omega$)

→ pour Y : ($|B|_\Omega, |\neg B|_\Omega$)

(où $\neg R$ est le complément d'une partie R d'un ensemble).

On peut alors définir une mesure de similarité entre les deux variables X et Y, à partir de l'intersection entre ces deux partitions [ABID 79] [TORO 82] :

$$f : P(\Omega/X) \times P(\Omega/Y) \longrightarrow \mathbb{R}^+$$

où :

- $P(\Omega/S)$ est l'ensemble des parties de la population d'apprentissage Ω induites par la variable dichotomique S ($S=X$ ou Y).

- par définition de la similarité, f doit être :

. symétrique : $f(p_1, p_2) = f(p_2, p_1) \quad \forall p_1, p_2 \in P(\Omega)$

. $f(p_1, p_1) = f(p_2, p_2) > f(p_1, p_2) \quad \forall p_1, p_2 \in P(\Omega) \quad p_1 \neq p_2$

Pour $A \in \prod_{i=1}^n O_i$ et $B \in M_Y$ donnés, on considère les quatre règles

suivantes : $r_{A,B}$, $r_{\neg A,B}$, $r_{A,\neg B}$ et $r_{\neg A,\neg B}$.

Soit f une mesure de similarité, et Φ la mesure de R_Σ dans $\mathbb{R}^+$ associée :

$$\Phi : R_\Sigma \longrightarrow P(\Omega/X) \times P(\Omega/Y) \longrightarrow \mathbb{R}^+$$

A toute règle $r_{A,B}$, on peut faire correspondre les quatre nombres :

- . $s = \Phi(r_{A,B})$ = indice caractérisant la règle $r_{A,B}$
- . $u = \Phi(r_{A,\neg B})$ = ----- $r_{A,\neg B}$
- . $v = \Phi(r_{\neg A,B})$ = ----- $r_{\neg A,B}$
- . $t = \Phi(r_{\neg A,\neg B})$ = ----- $r_{\neg A,\neg B}$

où $BU \neg B = M_Y$ et $AU \neg A = \prod_{i=1}^p O_i$

Ces indices peuvent être représentés par un tableau binaire, ou un diagramme de Venn :

$\prod O_i \backslash M_Y$	B	$\neg B$
A	s	u
$\neg A$	v	t

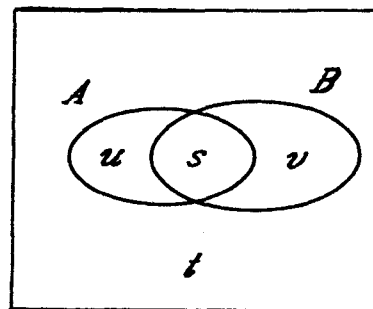


fig.3.6 : Indices caractéristiques associés à une règle $r_{A,B}$

A titre d'exemple, si $\Phi(r_{A,B}) = \text{card } |r_{A,B}|_\Omega$, l'interprétation est la suivante :

- s représente le nombre d'exemples couverts par la règle $r_{A,B}$.
- u est le nombre de contre-exemples reconnus par la règle $r_{A,B}$.
- v est le nombre d'exemples non couverts par la règle $r_{A,B}$.
- t est le nombre de contre-exemples non reconnus par la règle $r_{A,B}$.
- $s+v$ (resp. $u+t$) est le nombre total d'exemples (resp. de contre-exemples).

Toute fonction τ telle que :

$$\begin{array}{ccc} \tau : & R_{\Sigma} & \longrightarrow \mathbb{R} \\ & r_{A,B} & \longmapsto \tau[s,u,v,t] \end{array}$$

peut servir à caractériser une règle. Nous construisons deux indices caractéristiques particuliers :

- un indice de "couverture"
- un indice de "pollution".

III.2.1. Taux de couverture d'une règle

Considérons une règle $r_{A,B}$ de R_{Σ} .

Nous définissons le taux de couverture (noté τ_C) de cette règle comme une mesure de la généralité de la prémisse A, sachant que la conclusion appartient à B.

τ_C peut donc être représenté par n'importe quelle fonction croissante de s et décroissante de v.

→ τ_C est maximal si v est minimal et si s n'est pas minimal.

Dans ce cas, la couverture est totale. La règle considérée est complète. ("une règle est complète si tous les exemples sont reconnus" [RALA 87,88]). Certains auteurs parlent de règles valides (i.e. si suffisamment d'objets sont reconnus [LIQ 90]).

→ Dans le cas contraire ($\tau_C \neq \tau_C$ maximal), la couverture est dite "partielle". La règle est incomplète.

Cette mesure nous permettra, par exemple, de déterminer un seuil τ_{C0} en dessous duquel une règle sera rejetée.

Cette condition nous permettra d'éliminer des règles ne reflétant que des faits isolés (par exemple un bruit). Elle nous garantit donc une certaine robustesse des résultats obtenus.

Par exemple, prenons le cas de la mesure symbolique, avec

$$\Phi(r_{X,Y}) = \text{card } |r_{X,Y}|_{\Omega} :$$

La couverture est totale (règle complète) si $v=0$ et $s \neq 0$.

Nous pouvons caractériser τ_C par :

$$\tau_C = \frac{\text{card } |r_{A,B}|_{\Omega}}{\text{card } |r_{M_X,B}|_{\Omega}} \quad \text{qui représente le pourcentage d'exemples couverts par la règle } r_{A,B}.$$

$$\tau_C = s/(s+v).$$

(ou le pourcentage de bien classés [RALA 87,88]).

τ_C peut s'interpréter en termes de probabilité d'observer X appartenant à A, sachant que Y appartient à B : $P(X \in A / Y \in B)$.

En posant la définition ci-dessus, nous avons généralisé une notion utilisée de façon restrictive. En effet, bien souvent, on ne considère que des règles reconnaissant au minimum β exemples ($r_{A,B} \in R_{\Sigma}^*(\beta)$), β étant un seuil fixé ; alors que n'importe quelle fonction vérifiant les deux conditions ci-dessus est valable.

III.2.2. Taux de pollution d'une règle (inverse du degré de croyance [REND 86])

Considérons une règle $r_{A,\theta}$ de R_{Σ} , et de cohérence perçue :

θ représente l'ensemble des modalités observées de M_Y reconnues par A : $\theta = \bigcup_j \beta_j \in P(M_Y)$ avec $\exists \alpha \in A / r\{\alpha\}, \{\beta_j\}$ vraie.

L'incohérence des données d'apprentissage se caractérise par le fait que la règle $r_{A,\theta}$ peut être très inefficace. Il serait alors intéressant de trouver une règle $r_{A,B}$ (avec $\theta = B \cup B'$) plus efficace, bien que non toujours vraie, la réalité observée n'étant pas exactement représentée [POM 91].

Nous nous proposons d'évaluer la vérité de la règle $r_{A,B}$ (B étant une partie de θ) en introduisant un taux de pollution (noté τ_p).

Ce taux de pollution représente un coût (risque établi) lié au fait que nous ne prenons en compte qu'une partie de toutes les modalités de θ reconnues par A (avec en contre partie, une meilleure efficacité).

τ_p peut être caractérisé par n'importe quelle fonction décroissante de s et croissante de u.

→ τ_p est minimal si s n'est pas minimal, et si u est minimal.

La couverture est alors dite propre. La règle considérée est consistante ("une règle est consistante si aucun contre-exemple n'est reconnu" [RALA 87,88]).

Une extension de cette notion peut être utilisée : "une règle est quasi-cohérente si elle est valide et si elle reconnaît peu de contre-exemples" [LIQ 90].

→ Dans le cas contraire ($\tau_p \neq \tau_p$ minimal), la règle considérée est inconsistante, ou incohérente.

Cette notion peut être utilisée pour trouver des règles suffisamment discriminante (peu de contre-exemples sont reconnus) ; par

l'intermédiaire d'un seuil τ_{p0} , au dessus duquel les règles sont rejetées.

Cas particuliers :

→ mesure symbolique avec $\Phi(x,y) = \text{card } |r_{x,y}|_{\Omega}$:

La couverture est propre si $s \neq 0$ et $u=0$.

Nous pouvons caractériser τ_p par :

$$\tau_p = \frac{\text{card } |r_{A,-B}|_{\Omega}}{\text{card } |r_{A,M_Y}|_{\Omega}} \quad \text{qui représente le pourcentage de contre-exemples reconnus (couverts) par } r_{A,B}.$$

$$\tau_p = u/(s+u) \quad (\text{ou le pourcentage de mal classés}).$$

→ un exemple de taux de pollution utilisant l'entropie est proposé au chapitre IV (§II.3).

La mesure $s/(s+v)$ représente en fait le taux de croyance de la règle considérée [REND 86].

Nous pouvons formuler la même remarque que pour le taux de couverture. τ_p est une mesure de l'impureté d'une couverture, celle-ci pouvant être caractérisée par n'importe quelle fonction vérifiant les deux propriétés ci-dessus.

Un modèle idéal serait constitué d'un ensemble de règles R à la fois très générales (τ_c grand) et très efficaces.

Mais plus une règle est générale, et plus son inefficacité est probable (§III.1). La rendre plus efficace se traduit en pratique par un accroissement du taux de pollution.

Il nous faut alors trouver un compromis entre l'efficacité et le taux de pollution de la règle considérée.

IV. GENERALISATION SUR UN ENSEMBLE DE REGLES

Dans [REND 86], L. Rendell parle du problème de "généralisation" comme la formation de sous-ensembles (ou de classes) à partir d'un univers de formes, d'évènements, ou d'objets. La description des concepts doit donc caractériser les objets observés, ainsi que des objets *similaires* non encore rencontrés. C'est dans ce sens que l'induction doit être prédictive.

On étudiera deux modes possibles de généralisation :

- l'agrégation
- le prolongement

IV.1. Règles de généralisation

IV.1.1. Approche globale

Les modèles obtenus par une approche globale s'expriment sous la forme $Y=f(X)$, c'est-à-dire sous la forme de règles telles que :

$$X=\alpha \implies Y=\beta \quad \begin{array}{l} \alpha \in P(M_X) \\ \beta \in P(M_Y) \end{array}$$

La généralisation d'un tel ensemble de règles passe par la construction d'une variable $\tilde{X}$ plus générale que X , permettant d'exprimer l'ensemble de ces règles sans perte de qualité.

$\tilde{X}$ est une variable plus générale que X si :

$$\forall \alpha \in M_X \quad \exists \tilde{\alpha} \in P(M_{\tilde{X}}) \quad / \quad \alpha \in \tilde{\alpha}$$

remarque :

La partition $P_{\tilde{X}}(\Omega)$ induite par $\tilde{X}$ est alors plus grosse que la partition $P_X(\Omega)$ induite par X .

L'ensemble des modalités de $\tilde{X}$ sera noté $M_{\tilde{X}}$.

En pratique, nous essayons de minimiser la dégradation de la qualité du modèle, due à l'utilisation de $\tilde{X}$, au lieu de X .

Basés sur le calcul de $H(Y/\tilde{X})$ et $H(Y/X)$, deux indices de "modélisabilité" sont utilisés [TORO 82] :

$$\rightarrow m(Y/X) = 1 - \frac{H(Y/X)}{H(Y)}$$

mesure la modélisabilité de Y par X . C'est une mesure de l'incohérence des données d'apprentissage.

$$\rightarrow q(Y/\tilde{X}) = \frac{H(Y) - H(Y/\tilde{X})}{H(Y) - H(Y/X)}$$

mesure la qualité du modèle (plus simple) $f(\tilde{X})$ par rapport à la qualité du modèle $Y=f(X)$.

$\tilde{X}$ conduit à la règle généralisée $r_{\tilde{\alpha}, f(\tilde{\alpha})}$ où :

$\rightarrow \tilde{\alpha} \in P(M_{\tilde{X}})$ est une modalité de $\tilde{X}$, obtenue par agrégation d'une ou de plusieurs modalités de X .

$$\rightarrow f : \begin{array}{l} M_{\tilde{X}} \longrightarrow M_Y \\ \tilde{\alpha} \longmapsto f(\tilde{\alpha}) \end{array} \quad / \quad r_{\tilde{\alpha}, f(\tilde{\alpha})} \text{ vraie} \quad \forall \tilde{\alpha} \in M_{\tilde{X}}$$

remarque :

Une variable $\bar{X}$ plus grosse que X peut aussi être construite en considérant l'ensemble $P(\prod_i O_i)$ au lieu de

$P(M_X)$. On parlera alors de généralisation par prolongement, puisqu'on considère, dans les prémisses des règles, des modalités n'ayant pas été réellement observées (cf §IV.3).

IV.1.2. Approche locale

Dans l'approche locale, on généralise les règles considérées une par une. Dans ce cadre, [MICH 80,83a] et [KOD 86] ont formulé différentes règles de généralisation (cf annexe 2) :

(1) Règle de l'oubli des conjonctions (*the dropping condition rule*) : $A \& B$ est plus particulier que A .

(2) Règle de l'ajout des disjonctions (*the generalization by internal disjonction* ou *the extending reference rule*) :

C'est la règle symétrique à celle de l'oubli : $a \vee B$ est toujours plus général que A .

On peut alors définir deux cas particuliers :

(2.1) Règle d'agrandissement des domaines :

$$A(x) \& x \in \{x_1, \dots, x_n\} \equiv A(x_1) \vee A(x_2) \vee \dots \vee A(x_n) \longrightarrow A(x_1) \vee A(x_2) \vee \dots \vee A(x_n) \vee A(x_{n+1}).$$

avec A =prédicat.

(2.2) Règle de l'intervalle (*the closing interval rule*) :

$$\left. \begin{array}{l} A(x) \& x=a \\ A(x) \& x=b \end{array} \right] \longrightarrow A(x) \& x \in [a..b]$$

(si x est une variable ordonnée).

(3) the extension against rule :

$$\left. \begin{array}{l} T \wedge A \implies B \\ T \wedge C \implies \neg B \end{array} \right] \quad T \wedge \neg C \implies B$$

où $A \cap C = \emptyset$

Cette règle est à utiliser avec prudence. En effet, on agrandit ici A à $\neg C$, mais on pourrait agrandir C à $\neg A$ ($T \wedge \neg A \implies \neg B$). Il y aura alors conflit des deux règles hors du domaine AUC.

D'autres règles de généralisation existent, mais elles sortent du cadre de ce travail :

* règle de transformation des constantes en variables (*the turning constants to variables rule*).

- * règle de suppression des prémisses des implications.
- * règle d'ajout de noms de variables.
- * règle de remontée dans l'arbre de généralisation (*the climbing generalization tree rule*) pour des variables structurées.

Pour notre part, nous nous intéresserons surtout à la règle (2) : règle de l'ajout des disjonctions.

On peut remarquer, comme précédemment, que les disjonctions ajoutées peuvent correspondre à des modalités réellement observées des variables explicatives (agrégation) ou à des modalités non observées (prolongement).

Finalement, le problème de la généralisation consistera alors d'une part, à construire de nouvelles règles (par agrégation ou par prolongement) et d'autre part, à évaluer leur qualité.

IV.2. Agrégation de deux règles

Soient $A_i \Rightarrow B_i$ et $A_j \Rightarrow B_j$ deux règles de R_Σ .

L'agrégation de ces deux règles est définie par la règle :

$$A_i \cup A_j \Rightarrow B_i \cup B_j$$

Celle-ci est plus générale que les deux règles initiales ($A_i, A_j \subset A_i \cup A_j$), mais elle est en général moins efficace ($B_i, B_j \subset B_i \cup B_j$), sauf si les deux règles initiales sont ordonnées par la relation d'efficacité.

En effet, si $B_i \subset B_j$, l'agrégation des règles $A_i \Rightarrow B_i$ et $A_j \Rightarrow B_j$ donne la règle $A_i \cup A_j \Rightarrow B_j$. Celle-ci est plus générale que les règles initiales, et son efficacité n'est pas moindre que celle de la moins efficace d'entre elles.

IV.3. Prolongement d'une règle

Considérons deux règles $A \Rightarrow B$ et $\tilde{A} \Rightarrow B$ de R_Σ :

Par définition, $\tilde{A} \Rightarrow B$ est un prolongement de $A \Rightarrow B$ si et seulement si :

- $\tilde{A} \Rightarrow B$ est plus générale que $A \Rightarrow B$: $A \subset \tilde{A}$,
- $\forall \mu \in \tilde{A} \setminus A, \mu \notin M_X$ (i.e. μ n'a pas été observé. Ces deux règles sont en cohérence perçue).

A partir de cette définition, nous distinguons le cas où μ n'est pas observable ($\mu \notin O$, i.e. les règles $A \Rightarrow B$ et $A \cup \{\mu\} \Rightarrow B$ sont en cohérence absolue), et le cas où μ n'est pas observé, mais est observable ($\mu \in O \setminus M_X$, i.e. $A \Rightarrow B$ et $A \cup \{\mu\} \Rightarrow B$ en cohérence perçue) :

Dans le cas où μ n'est pas observable, la règle $AU\{\mu\} \Rightarrow B$ est plus générale que $A \Rightarrow B$, elle possède la même efficacité et les mêmes taux de couverture et de pollution, calculés à partir de l'ensemble des données d'apprentissage.

De plus, pour tous les autres ensembles de données d'apprentissage, $\{\mu\}$ n'a pas d'influence sur la qualité de $AU\{\mu\} \Rightarrow B$.

Aucun risque n'est pris à prolonger (à étendre) la règle $A \Rightarrow B$ par la règle $AU\{\mu\} \Rightarrow B$. On dira que μ est en affinité avec $A \Rightarrow B$.

A titre d'exemple, pour le tableau de l'annexe 5, on ne prend aucun risque à prolonger la règle

$[Taille = o \vee -] \ \& \ [Poids = -] \Rightarrow [Fonction = Co \vee Ch]$

à $[Taille = + \vee o \vee -] \ \& \ [Poids = -] \Rightarrow [Fonction = Co \vee Ch]$

si on considère qu'il ne peut pas exister de chiens dont la taille soit "+" et le poids "-". On peut alors simplifier la règle de la façon suivante : $[Poids = -] \Rightarrow [Fonction = Co \vee Ch]$.

En effet, si une règle couvre toutes les modalités observables d'une variable donnée, alors on peut omettre cette variable.

Dans le cas où μ est observable, mais non observé, trois approches peuvent être considérées :

Les deux premières sont issues des logiques non monotones :

→ *Hypothèse du "monde clos"* [REIT 78] [CATA 90] :

"Tout ce qui n'est pas déductible est considéré comme faux".

"si $\neg(\{\mu\} \Rightarrow B)$ alors inférer $\neg(\{\mu\} \Rightarrow B)$ ".

où $\neg$ signifie "si nous ne pouvons déduire ..."

La modalité μ n'est pas observée, donc nous ne pouvons déduire la conclusion " $\mu \in B$ " à partir de la modalité μ .

La règle $\{\mu\} \Rightarrow B$ n'est pas utilisée car $\forall \beta \in B$, la règle $\{\mu\} \Rightarrow \{\beta\}$ n'est pas démontrée vraie.

Cette approche ne sera pas utilisée : aucune règle ne pourrait être prolongée.

→ *Hypothèse de "non contradiction"* :

"x est vrai, sauf si le concepteur du système d'information s'est donné la peine de dire (ou de permettre au système de déduire) que x est faux." [KAYS 90].

La règle $\{\mu\} \Rightarrow B$ est vraie tant que le contraire n'a pas été démontré, i.e. tant que : $\nexists \beta \in M_y$ et $\beta \notin B$ / $\{\mu\} \Rightarrow \{\beta\}$ vraie.

La troisième approche prend en compte la notion de similarité :

→ "Hypothèse de continuité" :

Cette approche s'applique aux cas où une mesure de proximité (similarité) existe entre les parties de O :

$\forall (A, A') \in P^2(O)$, $\exists s$ telle que :

- $s : P(O) \times P(O) \longrightarrow \mathbb{R}^+$
- s est symétrique : $s(A, A') = s(A', A)$
- $s(A, A) = s(A', A') > s(A, A')$, avec $A \neq A'$.

Considérons deux règles $A \Rightarrow B$ et $A' \Rightarrow B$ de R_E :

Supposons que la règle $A \Rightarrow B$ soit vraie, et que nous ne sachions pas si la règle $A' \Rightarrow B$ est vraie ou fausse.

Par continuité, nous pouvons dire que plus $s(A, A')$ est grand, et plus la règle $A' \Rightarrow B$ a des chances d'être vraie.

Le prolongement de $A \Rightarrow B$ conduit alors à appliquer ce principe aux couples (A, A') avec $A' = \{\mu\}$.

Exemples de mesures de proximité :

→ Pour des variables qualitatives, nous pouvons définir une taxonomie sur les modalités observables [DID 89].

$s(A, A')$ peut être l'indice de plus bas niveau pour lequel A et A' sont regroupés.

→ Pour des variables quantitatives, nous pouvons définir une distance $\delta(\alpha, \alpha')$ entre deux modalités α et α' . Nous pouvons prendre, par exemple, la distance du lien minimal :

$$d(A, \{\mu\}) = \min_{\alpha \in A} \delta(\alpha, \mu)$$

où d représente ici la distance entre un ensemble de modalités A et une modalité μ (dans notre cas, μ est une modalité observable non observée). s est alors une fonction décroissante de d .

V. EVALUATION DE LA GENERALISATION

Nous proposons ici de déterminer des indices mesurant la dégradation du modèle simplifié, par rapport au modèle initial. Nous introduisons également la notion de risque.

V.1. Evaluation de l'agrégation

L'agrégation de deux règles $A_i \Rightarrow B_i$ et $A_j \Rightarrow B_j$ nous donne la règle $A_i \cup A_j \Rightarrow B_i \cup B_j$.

Cette dernière est plus générale, et sa prémisse, sachant que la conclusion est $B_i \cup B_j$ ($Y \in B_i \cup B_j$), peut être caractérisée par un taux de couverture et un taux de pollution.

V.2. Evaluation du prolongement

Considérons une règle $A \Rightarrow B$, et $\tilde{A} \Rightarrow B$ son prolongement.
 Nous introduisons la notion de risque, correspondant au fait que plus la règle $\tilde{A} \Rightarrow B$ est générale, et moins la conclusion B est évidente. En d'autres termes, plus $\tilde{A}$ contient de modalités observables non observées, et plus le risque d'avoir " $Y \notin B$ si $X \in \tilde{A}$ " augmente.

Intuitivement, il n'y a aucun risque si nous sommes sûrs que :

- X ne prend jamais une valeur dans $\tilde{A} \setminus A$ ($\subset \prod_{i=1}^n O_i \setminus O$)

(modalités non observables ; $A \Rightarrow B$ et $\tilde{A} \Rightarrow B$ sont en cohérence absolue).

- ou toutes les valeurs prises par X dans $\tilde{A} \setminus A$ correspondront à une modalité de Y appartenant à B .

Par contre, il y a un risque si X peut prendre une valeur dans $\tilde{A} \setminus A$, et si nous n'avons aucune raison de penser que Y prendra nécessairement une modalité de B .

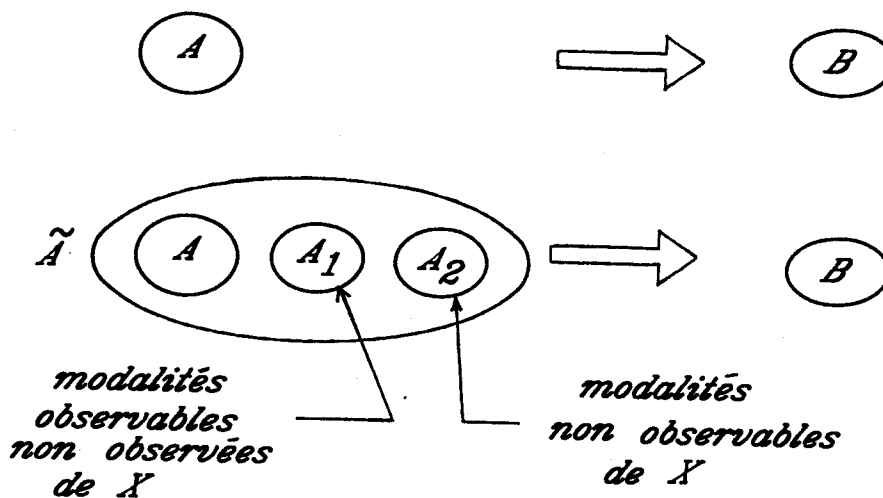


fig.3.9 : Prolongement d'une règle

De plus, deux approches (issues respectivement des logiques non monotones, et du principe de continuité) caractérisant le prolongement d'une règle sont possibles (§IV.3) :

→ La première approche consiste à dire que $\forall \mu \in \tilde{A} \setminus A$, la règle $\{\mu\} \Rightarrow B$ est vraie tant que le contraire n'a pas été démontré. Deux hypothèses extrêmes sont alors possibles :

■ Une hypothèse dite "optimiste" : $\forall \mu \in A_1, Y \in B$.

Il n'y a aucun risque à dire "si $X \in \tilde{A}$ alors $Y \in B$ ".

Le taux de couverture, le taux de pollution, et l'efficacité des deux règles $\tilde{A} \Rightarrow B$ et $A \Rightarrow B$ sont identiques.

D'autre part, la généralité augmente ($A \subset \tilde{A}$).

■ Une hypothèse dite "pessimiste" : $\forall \mu \in A_1, Y \notin B$.

Il y a un risque à dire "si $X \in \tilde{A}$ alors $Y \in B$ ".

Le taux de couverture et l'efficacité de ces deux règles sont identiques.

La règle $\tilde{A} \Rightarrow B$ est plus générale que la règle $A \Rightarrow B$, mais par contre, son taux de pollution augmente.

Ce taux de pollution est lié au nombre de modalités observables non observées de $\tilde{A}$.

En considérant ces hypothèses extrêmes, nous pouvons remarquer que l'existence d'un risque est liée à une augmentation du taux de pollution.

Ainsi, le risque de dire "si $X \in \tilde{A}$ alors $Y \in B$ " peut être caractérisé par le coût τ_p d'une hypothèse (définie par l'analyste).

Cette hypothèse peut être :

→ "optimiste" (aucun risque)

→ "pessimiste" (risque maximal)

→ "réaliste" (cette hypothèse peut être établie via une connaissance "a priori" du système).

D'un point de vue pratique, l'analyste pourra se fixer un nombre (un seuil) de modalités observables non observées de X . Il pourra alors prolonger une règle tant que ce seuil n'est pas atteint.

→ La seconde approche utilise l'hypothèse de continuité :

Lorsque l'ensemble A_1 des modalités observables non observées est proche de A (au sens de la relation de similarité), le risque de dire "si $X \in \tilde{A}$ alors $Y \in B$ " n'est pas très important.

Plus les ensembles A et A_1 sont dissemblables, et plus le risque augmente.

Nous considérons donc le risque comme une fonction décroissante de la similarité entre A et A_1 .

En pratique, l'analyste pourra effectuer un prolongement d'une règle (par des modalités observables non observées de X) tant que la distance est inférieure à un seuil donné.

Exemple : Le risque de prolonger A à $\tilde{A}$ avec la conclusion B peut être caractérisé par :

$$\sum_{\substack{\mu \in \tilde{A} \setminus A \\ \& \mu \in O \setminus M_X}} \frac{s(\mu, C(\neg B))}{s(\mu, C(B))}$$

où $s(x,y)$ est une mesure de similarité entre x et y ;
 et $C(B) = \bigcup_{\beta \in B} C(\beta)$

avec $C(\beta) := \{\alpha \in M_X / \exists w \in \Omega \ (X(w), Y(w)) = (\alpha, \beta)\}$
 $= \{\alpha \in M_X / \{\alpha\} \Rightarrow \{\beta\} \text{ vraie}\}$
 $= \{\alpha \in M_X / |\{\alpha\} \Rightarrow \{\beta\}|_{\Omega} \neq \emptyset\}$

Le risque peut alors être considéré comme une pénalisation caractérisant le manque de connaissance *a priori* du système.

VI. CONCLUSION

Dans ce chapitre, nous avons présenté une démarche d'analyse des propriétés d'un ensemble de règles obtenues par apprentissage à partir des données. On a caractérisé les propriétés de cohérence et de complétude, et on a proposé des critères permettant l'évaluation des règles obtenues au moyen des taux de couverture et de pollution ; la distinction entre modalités observées et modalités observables, introduite au début de ce travail, nous a permis de distinguer deux procédures de construction de règles plus générales : l'agrégation et le prolongement. Nous avons donné, pour ces deux cas, des moyens d'évaluation des résultats obtenus reposant sur différentes hypothèses (monde clos, non contradiction, continuité).

CHAPITRE IV :

CONSTRUCTION D'UN MODELE

(extraction de règles)

I. POSITION DU PROBLEME

Dans ce travail, nous nous attacherons plus particulièrement au cas où les données d'apprentissage sont incohérentes, tout en sachant que les méthodes décrites dans ce cas sont aussi applicables pour des données cohérentes.

La solution idéale serait de construire des règles complètes sur Σ , toutes les observations étant prises en compte. Ces règles reflèteraient alors aussi bien le comportement que l'on cherche à modéliser, que les éventuels bruits (comportement aléatoire). Si le système étudié est très bruité, ce type de raisonnement ne peut conduire à des résultats satisfaisants.

On s'oriente alors vers l'application d'une procédure de contraste sur les données observées. Cette opération consiste à garder le comportement déterministe (ou considéré comme tel) du système étudié, et à éliminer son comportement aléatoire.

A partir des données contrastées, nous proposons deux approches inductives :

- une approche basée sur un critère entropique global : plusieurs algorithmes seront développés [POM 90a].
- une approche plus symbolique, basée sur les diverses notions introduites jusqu'à présent [POM 90b].

II. CONSTRUCTION D'UN MODELE ATOMIQUE [BARB 87a] [BARB 87b] [STAR 91]

M. Barboucha propose de construire des relations d'Entrée/Sortie entre les deux variables X (qui peut être une variable vectorielle) et Y , dont les ensembles de modalités observées sont respectivement $\{\alpha_1, \alpha_2, \dots, \alpha_n\}$ et $\{\beta_1, \beta_2, \dots, \beta_m\}$.

Ces relations sont exprimées sous forme de règles.

L'approche proposée s'appuie sur l'hypothèse suivante : les données relevées sur le système sont le résultat :

- d'un comportement déterministe (que l'on cherche à extraire et à modéliser).
- d'un comportement aléatoire superposé au premier : aléas de mesure, bruits, erreurs d'interprétation, ...

On cherchera alors à éliminer la composante aléatoire.

II.1. Définitions

Chaque modalité observée α_i de X engendre une règle du type :

$$X=\alpha_i \implies Y=B(\alpha_i) \quad (1)$$

où $B(\alpha_i)$ représente une disjonction des modalités observées de Y sachant que $X=\alpha_i$.

Appellons $m_i = \text{card } B(\alpha_i)$.

Remarque :

Si p_{ij} est la probabilité d'obtention conjointe de α_i et de β_j , alors $B(\alpha_i)$ représente l'ensemble des modalités de Y telles que $p_{ij} \neq 0$.

Une règle du type (1) est en général peu efficace (incohérence des données d'apprentissage). Le problème que l'on se pose tout naturellement est le suivant : Pour chaque règle du type (1), pouvons-nous éliminer certaines modalités de Y que l'on considère comme la conséquence d'un phénomène purement aléatoire ?

Cette opération conduit à chercher des règles non toujours vraies. On détermine, pour chaque règle, un niveau de contraste σ_i ($i=1, \dots, n$). Il nous faut alors caractériser l'information perdue, due au fait qu'on ne prend en compte qu'une partie des conclusions réellement observées. Deux démarches peuvent être appliquées :

→ Introduction d'un coefficient de pollution, qui ne prend pas en compte les distributions de l'information utile (ou considérée comme telle), et de l'information perdue. La règle obtenue serait alors du type : $X=\alpha_i \implies Y=B_{\sigma_i}(\alpha_i)$ (où $B_{\sigma_i}(\alpha_i)$ est l'ensemble des σ_i modalités de Y ayant les plus grandes probabilités $p_{j/i}$) avec un coefficient (global) de pollution (ou un coefficient de croyance) de la règle. On crée sur Y une partition $(B_{\sigma_i}(\alpha_i), M_Y \setminus B_{\sigma_i}(\alpha_i))$, et les indices caractéristiques s, t, u, v définis au chapitre précédent représentent alors des poids, et non pas des répartitions.

→ Introduction de coefficient de pondération $r_{j/i}$ pour chaque modalités β_j de Y ($j=1, \dots, \sigma_i$). Dans ce cas, nous prenons en compte les distributions de probabilités de l'information utile, et de l'information perdue.

C'est sur la seconde formulation que les travaux de M. Barboucha ont porté, à savoir la construction de règles de type 2 :

$$X=\alpha_i \implies Y=(\beta_1, r_{1/i}) \vee (\beta_2, r_{2/i}) \vee \dots \vee (\beta_{\sigma_i}, r_{\sigma_i/i}) \quad (2)$$

avec $1 \leq \sigma_i \leq m_i$.

$r_{j/i}$ est un coefficient de pondération de la modalité β_j .
 σ_i est appelée la troncature de la ligne i ($i=1, \dots, n$).

Le problème consiste à formuler les différents $r_{j/i}$.

II.2. Opérateur de contraste

Pour chaque règle $X=\alpha_i \Rightarrow Y=B(\alpha_i)$, nous pouvons ordonner les modalités β_j ($\beta_j \in B(\alpha_i)$) suivant les $p_{j/i}$ (avec $p_{j/i} = p_{ij}/p_i$) de façon décroissante :

$$p(Y/X=\alpha_i) = \{p_{j/i} ; j=1, \dots, B(\alpha_i) / p_{1/i} \geq p_{2/i} \geq \dots \geq p_{m_i/i}\}$$

Cherchons alors à élaborer à partir de la distribution $P = p(Y/X=\alpha_i)$, une troncature σ_i fixée ($1 \leq \sigma_i \leq m_i$) et normalisons cette nouvelle distribution appelée R .

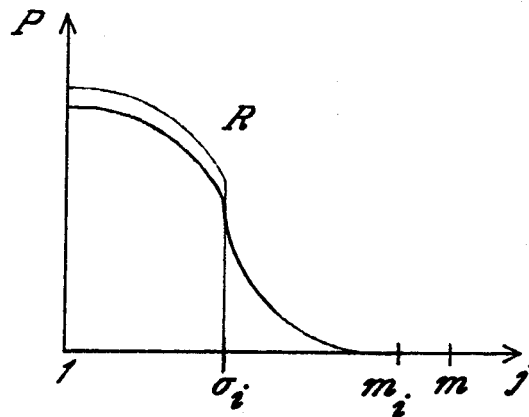


fig.4.1 : Tracé de R à partir de P pour σ_i fixée

De nombreux indices (distance du Khi^2 , le Khi'^2 de Neyman's, ...) existant dans la littérature permettent la minimisation de $D(P, R)$ représentant la distance entre les deux distributions P et R .

La solution optimale de R (appelée R^*) retenue est la suivante :

$$R^*(Y/X=\alpha_i) = \begin{cases} \cdot r_{j/i} = \frac{p_{j/i}}{\sum_{k=1, \dots, \sigma_i} p_{k/i}} & \text{pour } j=1, \dots, \sigma_i \\ \cdot 0 & \text{pour } j=\sigma_i+1, \dots, m_i. \end{cases}$$

L'opérateur de contraste $C\sigma_i$ de niveau σ_i relatif à la règle $X=\alpha_i \Rightarrow Y=B(\alpha_i)$ est défini par :

$$C\sigma_i = \begin{cases} \cdot & \frac{1}{\sum_{k=1, \sigma_i} p_{k/i}} & \text{pour } j=1, \dots, \sigma_i \\ \cdot & 0 & \text{pour } j=\sigma_i+1, \dots, m_i. \end{cases}$$

et donc,

$$R^*(Y/X=a_i) = \begin{cases} \cdot & C\sigma_i \cdot p(Y/X=a_i) & \text{pour } j=1, \dots, \sigma_i \\ \cdot & 0 & \text{pour } j=\sigma_i+1, \dots, m_i. \end{cases}$$

On appelle *résidu* ce qui est éliminé après l'opération de troncature sur chaque règle de la forme (1).

Normalisons la distribution des résidus (notée Q) :

$$Q(Y/X=a_i) = \begin{cases} \cdot & q_{j/i} = \frac{p_{j/i}}{\sum_{k=\sigma_i+1, m_i} p_{k/i}} & \text{pour } j=\sigma_i+1, \dots, m_i \\ \cdot & 0 & \text{pour } j=1, \dots, \sigma_i. \end{cases}$$

Schématiquement, nous obtenons pour chaque règle "i" ($1 \leq i \leq n$) :

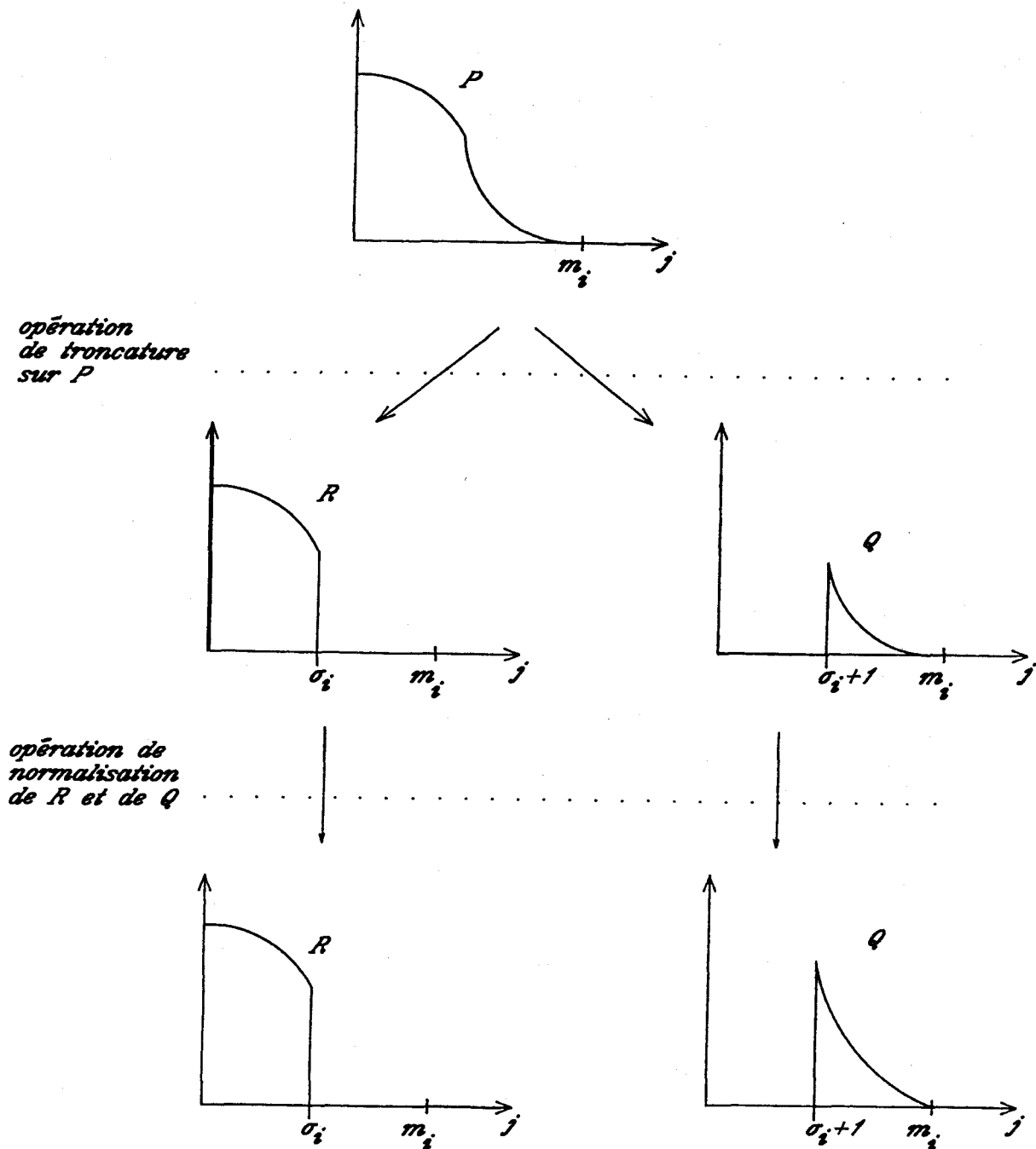


fig.4.2 : Distributions de P et de Q pour σ_i fixée

II.3. Utilité de l'information perdue - Coefficient de vérité

Le fait de négliger, pour chaque règle de la forme (1), une partie des modalités observées de Y , entraîne une perte d'information. Celle-ci sera mesurée par le coefficient $U(\sigma_i)$ appelé *utilité de l'information perdue*. Cet indice doit satisfaire les deux conditions suivantes :

C.1 : $U(\sigma_i)$ est une fonction strictement décroissante de $H(Q)$, c'est-à-dire : $H(Q_k) \leq H(Q_t) \implies U_k(\sigma_i) \geq U_t(\sigma_i)$.

C.2 : $U(\sigma_i)$ est une fonction décroissante de σ_i , c'est-à-dire : $U(\sigma_i) \geq U(\sigma_{i+1})$.

L'indice $U(\sigma_i)$ choisi est le suivant :

$$U(\sigma_i) := \left[\sum_{\sigma_i+1}^m p_{j/i} \right] \cdot \frac{\text{Log}(m-\sigma_i) - H(Q)}{\text{Log}(m)}$$

et qui est en fait le *coefficient de Non-Vérité* de la règle extraite (normalisé à 1).

Le *coefficient de Vérité* d'une règle est donc :

$$V(\sigma_i) := 1 - U(\sigma_i).$$

C'est une mesure du degré de croyance introduit au chapitre précédent (§III.2.2).

II.4. Efficacité d'une règle

M. Barboucha définit également une mesure de l'efficacité d'une règle, notion introduite également au chapitre précédent.

Une règle est dite d'*efficacité maximale* lorsqu'elle peut se traduire par : $X=\alpha_i \Rightarrow Y=\beta_j$.

Par contre, une règle du type : $X=\alpha_i \Rightarrow Y=\beta_1 \vee \beta_2 \vee \dots \vee \beta_m$ n'est pas efficace car l'analyste est confronté à un problème de décision quant au choix de la modalité de Y.

Le coefficient d'efficacité d'une règle doit donc être tel que :

- $e(1)=1$: efficacité maximale et normalisation du coefficient.
- $e(\sigma_i) \geq 0$: coefficient positif.
- $e(\sigma_i)$ est une fonction décroissante de σ_i .

Le coefficient retenu par M. Barboucha est le suivant :

$$e(\sigma_i) := \frac{\text{Log}(m) - H(R)}{\text{Log}(m)}$$

En particulier, nous avons :

- pour $\sigma_i=1$: $e(1)=1$ (efficacité maximale de la règle).
- pour $\sigma_i=m$: $e(m) = \frac{\text{Log}(m) - H(P)}{\text{Log}(m)}$ (efficacité minimale).

II.5. Modèle atomique

Le modèle atomique du système est formé des n règles obtenues par l'opération de contraste. Chaque règle est affectée d'un coefficient de vérité et d'un coefficient d'efficacité.

Compte tenu du nombre de règles important d'un tel modèle, il convient à présent de mettre en oeuvre une procédure de générali-

sation qui, opérant par agrégation et/ou par prolongement, permettra d'obtenir un modèle de dimensions plus réduites. Dans une approche globale, le problème d'agrégation peut s'exprimer comme celui de la recherche d'une partition de l'ensemble des modalités observées de X (création d'une variable plus grosse $\bar{X}$).

Le problème d'agrégation/prolongement conduit à la recherche d'une partition de l'ensemble des modalités observables de X.

Ces problèmes font l'objet du paragraphe suivant.

III. METHODES D'INDUCTION SUIVANT UN CRITERE ENTROPIQUE



III.1. Position du problème

Nous avons vu (chapitre II) que la quantité $H(Y/X)$ représente l'entropie de Y non expliquée par X. Cet indice peut dès lors être utilisé dans des algorithmes d'induction.

Rappelons que l'ensemble des modalités de Y est composé de m modalités distinctes (appelées $\beta_1, \beta_2, \dots, \beta_m$) et que n est le nombre de modalités observées de X.

Appelons $P(M_X)$ l'ensemble des partitions possibles des modalités observées de X. $P(M_X) = \{P_k(M_X) ; k=1, \dots, n\}$, où $P_k(M_X)$ est l'ensemble des partitions en k classes de l'ensemble des modalités observées de X.

Il existe deux cas triviaux :

- $k=1$: l'ensemble des modalités observées de X se trouve dans une classe unique : $\{a_1, a_2, \dots, a_n\}$.
- $k=n$: la seule partition possible est la suivante : $(\{a_1\}/\{a_2\}/\dots/\{a_n\})$.

Nous chercherons à simplifier le modèle atomique. Pour cela, nous recherchons une variable $\bar{X}$ plus grosse que X. Le problème que l'on se pose est alors le suivant :

pour $k=2, \dots, (n-1)$, on cherchera la partition d'ordre k (partition en k sous-ensembles) qui minimise $H(Y/P_k)$ (où P_k est une partition quelconque d'ordre k des modalités observées de X), ce qui se traduira par une recherche de k règles en cohérence logique de type 1.

La partition optimale P_k^* est la partition P_k telle que :

$$H(Y/P_k^*) = \min \{ H(Y/P_k) \}.$$

Nous traiterons le cas où l'ensemble des modalités de X n'est pas ordonné, puis le cas où il l'est [POM 90a]. Pour cette dernière

hypothèse, nous considérerons tout d'abord le problème où X est une variable primaire du tableau des données ; puis nous généraliserons à p variables ($p \geq 1$).

Le problème où il y a mélange de variables explicatives ordonnées et non ordonnées se résout simplement en ne considérant la contrainte d'ordre que pour les premières.

III.2. Cas de variables explicatives non ordonnées

Pour de petits ensembles de modalités observées (n très petit), une *recherche exhaustive* permet de parcourir l'ensemble des partitions en k classes.

Celle-ci n'est, par contre, pas envisageable dès lors qu'on dépasse quelques dizaines d'observations et de variables. En effet, il y a une *explosion combinatoire* du nombre de possibilités. Dans la pratique, beaucoup de tableaux sont de grandes dimensions.

Pour cette raison, nous nous orientons vers des méthodes dites heuristiques.

Nous développerons une méthode hiérarchique ascendante, descendante, et enfin un algorithme de transfert (basé sur les propriétés contractantes de l'entropie conditionnelle).

III.2.1. Recherche exhaustive

Pour de petits ensembles de modalités observées, une recherche exhaustive peut être appliquée. Elle consiste, pour un nombre de classes k donné, à trouver toutes les partitions possibles de M_X , et à retenir la partition qui minimisera l'entropie conditionnelle $H(Y/\tilde{X}_k)$.

Soit $S(n,k)$ le nombre de partitions possibles en k classes de M_X ($\text{card } M_X = n$). Ce nombre peut être calculé par récurrence :

$$S(n,k) = S(n-1,k-1) + k.S(n-1,k)$$

où :

- $S(n-1,k-1)$ résulte du fait que le $n^{\text{ième}}$ élément peut former à lui seul une classe.
- $S(n-1,k)$ est dû au fait que le $n^{\text{ième}}$ élément soit mis dans une classe (il y a k façons de le faire).

$S(n,k)$ est encore appelé le *nombre de Stirling de 2^e espèce*.

Pour 14 modalités observées, il y a plus de 10 millions de partitions possibles en 4 classes ! Nous voyons donc que pour de grands ensembles, il y a une *explosion combinatoire* du nombre de partitions possibles. Cette méthode n'est, en général, pas applicable, le temps de calcul étant trop long.

III.2.2. Recherche hiérarchique ascendante

Le principe de la recherche hiérarchique ascendante est l'agrégation successive des modalités de X : pour chaque ordre k , la partition qui minimisera l'entropie conditionnelle dépendra de la partition trouvée à l'ordre $(k+1)$.

Ces partitions seront alors appelées *partitions optimales localement*.

Essayons de regrouper deux modalités observées de X , et calculons leurs entropies conditionnelles :

(i) à l'ordre n : $H(Y/P_n) = H^*(Y/P_n)$

$$H(Y/P_n) = - \sum_{i=1}^n \sum_{j=1}^m p_{ij} \log p_{ij} + \sum_{i=1}^n p_i \log p_i$$

$$\text{avec } \sum_{i,j} p_{ij} = 1 \quad \text{et} \quad p_i = \sum_j p_{ij}$$

Supposons qu'à l'ordre $(n-1)$, on regroupe a_k et a_l :

$$\begin{aligned} H(Y/P_n) = & - \sum_{\substack{i=1 \\ i \neq k,l}}^n \sum_{j=1}^m p_{ij} \log p_{ij} + \sum_{\substack{i=1 \\ i \neq k,l}}^n p_i \log p_i \\ & - \sum_{j=1}^m p_{kj} \log p_{kj} - \sum_{j=1}^m p_{lj} \log p_{lj} \\ & + p_k \log p_k + p_l \log p_l . \end{aligned}$$

(ii) à l'ordre $(n-1)$, nous obtenons :

$$\begin{aligned} H(Y/P_{n-1}) = & - \sum_{\substack{i=1 \\ i \neq k,l}}^n \sum_{j=1}^m p_{ij} \log p_{ij} + \sum_{\substack{i=1 \\ i \neq k,l}}^n p_i \log p_i \\ & - \sum_{j=1}^m (p_{kj} + p_{lj}) \log (p_{kj} + p_{lj}) \\ & + (p_k + p_l) \log (p_k + p_l) . \end{aligned}$$

Nous obtenons donc :

$$\begin{aligned}
 H(Y/P_{n-1}) = H(Y/P_n) &+ \sum_{j=1}^n p_{kj} \text{ Log } p_{kj} + \sum_{j=1}^n p_{lj} \text{ Log } p_{lj} \\
 &- p_k \text{ Log } p_k + p_l \text{ Log } p_l \\
 &- \sum_{j=1}^n (p_{kj}+p_{lj}) \text{ Log } (p_{kj}+p_{lj}) \\
 &+ (p_k+p_l) \text{ Log } (p_k+p_l) .
 \end{aligned}$$

ou encore : $H(Y/P_{n-1}) = H(Y/P_n) + \delta H_{n,n-1}$.

Au chapitre II, nous avons vu que $H(Y/P_k)$ est une fonction décroissante de k , c'est-à-dire que pour une partition plus fine, l'entropie conditionnelle diminuait.

Donc, on a $H(Y/P_{n-1}) \geq H(Y/P_n)$.

Par conséquent, $\delta H_{n,n-1} \geq 0$.

Or, on cherche à minimiser $H(Y/P_{n-1})$, donc à minimiser $\delta H_{n,n-1}$.

On peut recommencer le même raisonnement à l'ordre r :

$H(Y/P_{r-1}) = H(Y/P_r) + \delta H_{r,r-1}$ pour $r=n, n-1, \dots, 2$.

La partition retenue est celle qui minimisera $\delta H_{r,r-1}$.

Nous proposons alors un algorithme pour une recherche hiérarchique ascendante :

```

; RECHERCHE HIERARCHIQUE ASCENDANTE
; initialisation
1.  p <— partition en n classes
2.  écrire la partition
; boucle
3.  min <— ∞
4.  pour k:=1 à (n-1) faire
5.      pour l:=(k+1) à n faire
6.          calcul du δH dû au regroupement des 2 modalités
              d'indices k et l
7.          si δH < min alors faire
8.              min <— δH
9.              mémorisation du regroupement
              fin si
          fin boucle1
      fin boucle2
10. n <— n-1
11. p <— nouvelle partition en n ensembles
12. écrire p
13. retourner à 3 si n > 1.
; Fin de l'algorithme

```

Remarques :

→ Comme nous l'avons déjà dit, rien ne prouve l'optimalité de cet algorithme. En effet, en général :

$$H(Y/P_{r-1}^*) = \min [H(Y/P_r^*) + \delta H_{r,r-1}] \\ \neq H(Y/P_r^*) + \min (\delta H_{r,r-1}).$$

En pratique, on préférera ce type d'algorithme à la recherche exhaustive (trop combinatoire).

→ Des algorithmes de recherche hiérarchique ascendante (et descendante) plus rapides ont été développés. Notre but n'est pas ici de trouver des algorithmes performants, mais de montrer qu'il est possible de les utiliser avec le critère choisi.

Nous renvoyons le lecteur désireux de connaître des algorithmes hiérarchiques plus rapides que celui-ci, à [BRUY 78,89] [RALA 84].

III.2.3. Recherche hiérarchique descendante

Cette recherche utilise la démarche inverse de la recherche ascendante :

On part de la partition des modalités observées de X en une classe sur laquelle on opère des dichotomies successives. On crée ainsi des partitions de plus en plus fines.

Soit $H(Y/P_k^*)$ l'entropie conditionnelle de Y connaissant une partition localement optimale d'ordre k de l'ensemble des modalités observées de X. Supposons que pour passer à l'ordre k+1 de façon localement optimale, il faille scinder la partie S de P_k^* en deux parties distinctes Q et $S \setminus Q$. Notons $P_{k+1}(Q, S \setminus Q)$ cette partition :

$$H(Y/P_{k+1}(Q, S \setminus Q)) \leq H(Y/P_k^*)$$

car $P_{k+1}(Q, S \setminus Q)$ est plus fine que P_k^* .

Donc, on a :

$$H(Y/P_{k+1}(Q, S \setminus Q)) - H(Y/P_k^*) = \delta H_{k,k+1}(Q, S \setminus Q) \leq 0.$$

Minimiser $H(Y/P_{k+1}(Q, S \setminus Q))$ revient à minimiser $\delta H_{k,k+1}(Q, S \setminus Q)$.

La nouvelle partition localement optimale d'ordre (k+1) sera donc celle qui minimisera $\delta H_{k,k+1}(Q, S \setminus Q)$, avec $S \in P_k^*$, $Q \subset S$, $Q \neq \emptyset$, $Q \neq S$.

En résumé :

→ à l'ordre k, on a P_k^* qui est la partition localement optimale en k classes de M_X , $H(Y/P_k^*)$ l'entropie conditionnelle associée.

→ à l'ordre $(k+1)$, on scinde $S \in P_k^*$ en deux parties Q et $S \setminus Q$ telles que :

$$\min_{\substack{S \in P_k^* \\ Q \subset S \\ Q \neq \emptyset \\ Q \neq S}} \{ \delta H_{k,k+1}(Q, S \setminus Q) \} \implies P_{k+1}^* \quad (\text{et } H(Y/P_{k+1}^*)).$$

algorithme proposé :

```
; RECHERCHE DESCENDANTE
; initialisation
1.  p ← partition la plus grosse
2.  écrire p
; boucle
3.  pour i:=2 à n faire
4.  | trouver toutes les partitions qui diviseraient une
5.  |   classe de p en 2 parties non vides
6.  |   prendre le minimum des  $\delta H$ 
7.  |   p ← nouvelle partition en i ensembles
   |   écrire p
   fin boucle
; FIN de l'algorithme
```

Remarque : $\min \{ \delta H_{k,k+1}(Q, S \setminus Q) \} = - \max \{ \delta H_{k+1,k} \}.$

De même que pour la recherche ascendante, rien ne garantit l'optimalité de cet algorithme. Nous essaierons donc de mettre en oeuvre un nouvel algorithme qui permettrait d'améliorer une partition quelconque d'ordre k . C'est le but de l'algorithme de transfert.

III.2.4. Algorithme de transfert

Aux deux paragraphes précédents, nous avons défini les deux opérateurs suivants :

opérateur A :

$$\begin{array}{lcl} A : & P_r(M_X) & \longrightarrow P_{r-1}(M_X) \\ & R = \{R_1, \dots, R_r\} & \longmapsto A(R) \quad (2 \leq r \leq n). \end{array}$$

$$\begin{aligned} A(R) &:= \{R_1, \dots, R_\alpha \cup R_\beta, \dots, R_r\} \\ &= \{R \setminus \{R_\alpha, R_\beta\}\} \cup \{R_\alpha \cup R_\beta\}. \end{aligned}$$

avec : R_α et $R_\beta / \min (\delta H_{r-1,r}).$

opérateur D :

$$\begin{array}{ccc} D : P_r(M_X) & \xrightarrow{\quad} & P_{r+1}(M_X) \\ R=\{R_1, \dots, R_r\} & \xrightarrow{\quad} & D(R) \quad (1 \leq r \leq n-1). \end{array}$$

$$D(R) := \{R_1, R_2, \dots, R_\alpha \setminus S^*, S^*, \dots, R_r\}$$

$$\begin{array}{l} \text{avec : } R_\alpha \in R ; S^* \subset R_\alpha ; S^* \neq R_\alpha ; S^* \neq \emptyset \\ \text{et } \min (\delta H_{r, r+1}(R_\alpha \setminus S^*, S^*)). \end{array}$$

Nous pouvons alors définir les deux opérateurs contractants AoD et DoA [POM 90a].

Propriété 1 :

Quelle que soit la partition P_r d'ordre r sur l'ensemble des modalités de X :

$$\rightarrow H(Y/\text{AoD}(P_r)) \leq H(Y/P_r) \quad (1)$$

$$\rightarrow H(Y/\text{DoA}(P_r)) \leq H(Y/P_r) \quad (2)$$

Démonstration du (2) :

Posons $P_r = \{\pi_1, \pi_2, \dots, \pi_r\}$. $H(Y/P_r)$ est connue.

On peut toujours trouver P_{r-1} telle que :

$$\begin{aligned} H(Y/P_{r-1}) &= H(Y/A(P_r)) \\ &= H(Y/P_r) + \min(\delta H_{r, r-1}) \end{aligned}$$

$$\begin{aligned} \text{De même, } H(Y/\text{DoA}(P_r)) &= H(Y/P_{r-1}) - \max(\delta H_{r, r-1}) \\ &= H(Y/P_r) + \min(\delta H_{r, r-1}) - \max(\delta H_{r, r-1}) \end{aligned}$$

Par définition du min et du max, on a :

$$\begin{aligned} \min(\delta H_{r, r-1}) &\leq \max(\delta H_{r, r-1}) \\ \implies H(Y/\text{DoA}(P_r)) &\leq H(Y/P_r). \end{aligned}$$

cqfd.

La démonstration de (1) est en tout point analogue à celle-ci.

Propriété 2 :

Si P_r^* est la partition optimale du niveau r , alors :

$$H(Y/\text{DoA}(P_r^*)) = H(Y/\text{AoD}(P_r^*)) = H(Y/P_r^*).$$

Démonstration :

Si $H(Y/P_r^*)$ est l'entropie conditionnelle pour la partition optimale de niveau r , alors :

$H(Y/P_r^*) \leq H(Y/P_r)$, et ceci quelle que soit une autre partition P_r d'ordre r .

Or, la propriété (2) implique que :

$$H(Y/\text{DoA}(P_r^*)) \leq H(Y/P_r^*).$$

De même pour la propriété (1).

Les inégalités dans les deux sens sont démontrées, il y a donc égalité des deux membres.

cqfd.

Ainsi, l'application récurrente des opérateurs AoD et DoA permet d'améliorer, au sens du critère de l'entropie conditionnelle, toute partition initiale en k classes, quel que soit le procédé par lequel elle a été obtenue.

Nous proposons alors l'algorithme suivant :

```

; ALGORITHME DE TRANSFERT
; initialisation
1.  générer une partition  $P_r$ 
2.  stable  $\leftarrow$  faux
; boucle principale
;   boucle 1
3.  |    $P_r \leftarrow$  AoD( $P_r$ )
4.  |   Si AoD( $P_r$ )  $\neq$   $P_r$  Alors retourner à 3
;   fin de boucle 1
;   boucle 2
5.  |    $P_r \leftarrow$  DoA( $P_r$ )
6.  |   Si DoA( $P_r$ )  $\neq$   $P_r$  Alors retourner à 5
;   fin de boucle 2
7.  Si AoD( $P_r$ )=DoA( $P_r$ )= $P_r$  Alors stable  $\leftarrow$  vrai
8.  Si (non stable) Alors retourner à 3
; fin de boucle principale
9.  écrire  $P_r$ 
; Fin de l'algorithme de transfert

```

La démarche de l'algorithme est la suivante :

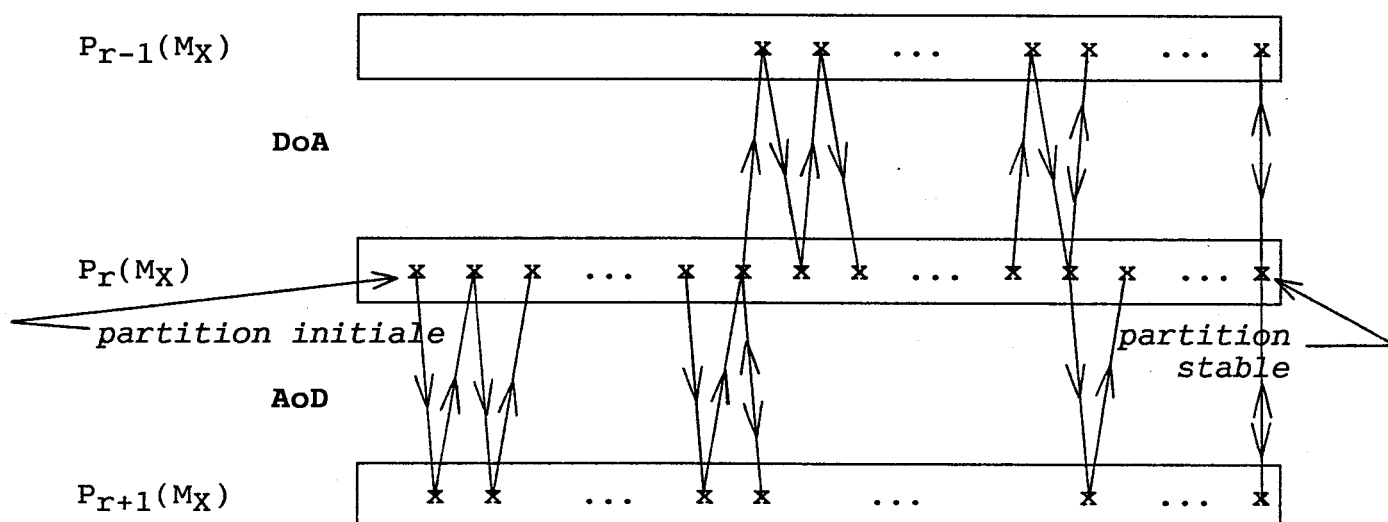


fig.4.3 : Démarche de l'algorithme de transfert

Remarques :

- La propriété (2) est la condition d'arrêt de notre algorithme de transfert.
- Le temps de calcul des algorithmes hiérarchiques ascendant, descendant et de l'algorithme de transfert peut être réduit. En effet, dans les boucles "pour", nous calculons des δH déjà trouvés aux itérations précédentes.

III.3. Cas de variables explicatives ordonnées

Dans le cas où l'ensemble des modalités de X est ordonné, il convient de tenir compte de la contrainte introduite par cette relation d'ordre (afin de pouvoir simplifier le modèle).

Lorsque l'ensemble des variables explicatives n'est constitué que d'une variable possédant des modalités ordonnées, la relation d'ordre "<" (plus petit que) est évidente. Mais si l'ensemble des variables explicatives est constitué de plusieurs variables, toute relation d'ordre ne peut qu'être partielle.

Un cas particulier est constitué par $O_i = \mathbb{R}$ ($i=1, \dots, p$), ce qui nous donne $M_X \subset O \subset \mathbb{R}^p$.

D'autre part, la contrainte d'ordre sur les modalités des variables explicatives diminue fortement l'aspect combinatoire des partitions possibles :

En effet, si on considère un ensemble non ordonné de n éléments, le nombre de partitions de cet ensemble en k classes peut être calculé de façon récurrente par la formule suivante :

$$S(n,k) = S(n-1,k-1) + k.S(n-1,k)$$

Lorsque ce même ensemble est ordonné, le nombre de partitions devient :

$$C_{n-1}^{k-1} = \frac{(n-1)!}{(k-1)!(n-k)!}$$

A titre d'exemple, le nombre de partitions possibles lorsque $n=9$ et $k=4$ est :

- dans le cas non ordonné : 7770
- dans le cas ordonné : 56.

Pour ces raisons, nous traiterons dans un premier temps le cas où $p=1$ (i.e. pour une variable explicative x).

Un algorithme de recherche exhaustive de la partition optimale reste, en général, inutilisable dans la pratique, les dimensions du tableau des données étant importantes. Nous retrouvons alors les trois heuristiques décrits précédemment, en y ajoutant la contrainte d'ordre.

L'aspect combinatoire étant moindre, nous développerons une méthode d'optimisation qui permettra de trouver une partition P_k d'ordre k de l'ensemble ordonné des modalités de X en minimisant $H(Y/P_k)$. Plusieurs approches peuvent s'adapter à ce problème. Dans ce travail, nous n'envisagerons que l'approche "programmation dynamique" et la méthode du gradient.

Dans un deuxième temps, nous généraliserons les résultats précédemment trouvés au cas de p variables explicatives.

III.3.1. Cas d'une variable explicative ordonnée ($p=1$)

Moyennant une renumérotation, on peut toujours écrire

$M_X = \{a_1, a_2, \dots, a_n\}$, avec $a_1 < a_2 < \dots < a_n$.

Par la suite, nous travaillerons sur cet ensemble ordonné.

III.3.1.1. Algorithmes hiérarchiques et de transfert

Nous pouvons reprendre les trois algorithmes détaillés au paragraphe III.2, en y introduisant une contrainte d'ordre, qui conduit à prendre en compte le fait que deux modalités, ou deux ensembles de modalités, sont consécutifs ou pas.

→ Pour les algorithmes hiérarchiques :

. Dans la recherche hiérarchique ascendante, la contrainte d'ordre peut être mise en évidence en posant $l=k+1$ (cf §III.2.2).

. Pour la recherche hiérarchique descendante, la contrainte est qu'on ne peut travailler que sur des ensembles consécutifs.

Les algorithmes sont alors les suivants :

```

; RECHERCHE HIERARCHIQUE ASCENDANTE avec x ordonnée
; initialisation
1.  p ← partition en n classes
2.  écrire la partition
; boucle
3.  min ← ∞
4.  pour k:=1 à (n-1) faire
5.    pour l:=(k+1) à n faire
6.      calcul du  $\delta H$  dû au regroupement des 2 modalités
          d'indices k et k+1
7.      si  $\delta H < \text{min}$  alors faire
8.        min ←  $\delta H$ 
9.        mémorisation du regroupement
          fin si
      fin boucle1
    fin boucle2
10. n ← n-1
11. p ← nouvelle partition en n ensembles
12. écrire p
13. retourner à 3 si  $n > 1$ .
; Fin de l'algorithme

; RECHERCHE DESCENDANTE avec x ordonnée
; initialisation
1.  p ← partition la plus grosse
2.  écrire p
; boucle
3.  pour i:=2 à n faire
4.    trouver toutes les partitions qui diviseraient une
          classe de p en 2 parties consécutives non vides
5.    prendre le minimum des  $\delta H$ 
6.    p ← nouvelle partition en i ensembles
7.    écrire p
    fin boucle
; FIN de l'algorithme

```

Comme pour le cas où les variables explicatives sont non ordonnées, rien ne prouve l'optimalité de ces deux algorithmes ; et comme précédemment, leur temps de calcul peut être amélioré.

→ Compte tenu du fait que l'algorithme de transfert découle directement des deux précédents, il suffit pour y introduire la contrainte d'ordre, de rectifier ceux-ci comme ci-dessus.

III.3.1.2. Utilisation d'une méthode d'optimisation

Tout problème d'optimisation est défini par un triplet (action, contraintes, critère). Formulons tout d'abord celui qui correspond à la recherche d'une partition optimale.

Action :

Une partition (en m classes) sur les modalités de Y étant établie, le problème est ici de trouver une partition P_k d'ordre k sur l'ensemble ordonné des modalités observées de x qui expliquerait au mieux celles de Y .

Shématiquement, cela se traduirait comme suit :

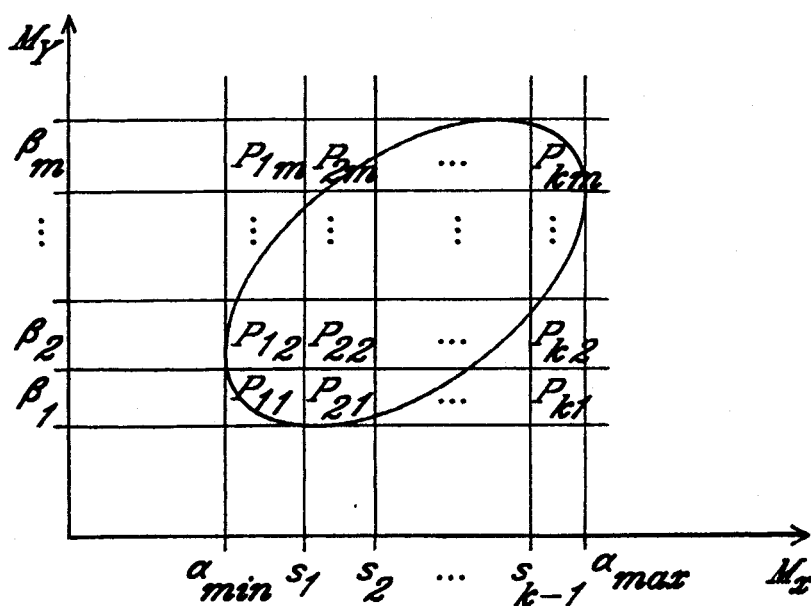


fig.4.4 : Partition P_k d'ordre k de M_x , avec x ordonnée

Notons $S=(s_1, s_2, \dots, s_{k-1}) \in P_{k-1}(M_x)$

où $P_{k-1}(M_x)$ est l'ensemble des parties de M_x constituées de $(k-1)$ éléments.

M_x étant un ensemble ordonné, trouver k classes de cet ensemble est équivalent à trouver $(k-1)$ séparateurs s_μ ($\mu=1, \dots, k-1$).

Dans le cas particulier où $M_x \subset \mathbb{R}$, $S \in \mathbb{R}^{k-1}$, et l'espace compris entre deux séparateurs est un segment.

Contraintes :

Nous devons toutefois introduire des contraintes d'ordre :

$$\alpha_{\min} < s_1 < s_2 < \dots < s_{k-1} < \alpha_{\max}$$

Par la suite, nous noterons $\alpha_{\min}=s_0$ et $\alpha_{\max}=s_k$, s_0 et s_k étant deux constantes.

Critère utilisé :

Le critère choisi dans ce travail est la minimisation de $H(Y/P_k)$, qui représente l'entropie conditionnelle de Y sachant qu'on a fait une partition P_k d'ordre k de l'ensemble des modalités observées de x .

Plusieurs méthodes d'optimisation peuvent permettre de trouver une partition sur l'ensemble M_x ordonné. Nous nous limiterons, dans ce travail, à deux d'entre elles : la *programmation dynamique* et la *méthode du gradient* (cette dernière étant limitée au travail sur $\mathbb{R}$).

III.3.1.2.1. Optimisation par programmation dynamique

Notons :

- $S := (s_1, s_2, \dots, s_{k-1})$
- $A(S) :=$ discrétisation des modalités de x en k ensembles.
- $A_h := \{a_i / s_{h-1} \leq a_i \leq s_h\} ; h=1, \dots, k$
- $s_0 := \alpha_{\min}$ et $s_k := \alpha_{\max}$ sont deux constantes.
- avec la contrainte d'ordre : $s_0 < s_1 < s_2 < \dots < s_{k-1} < s_k$.

Graphiquement, nous obtenons :

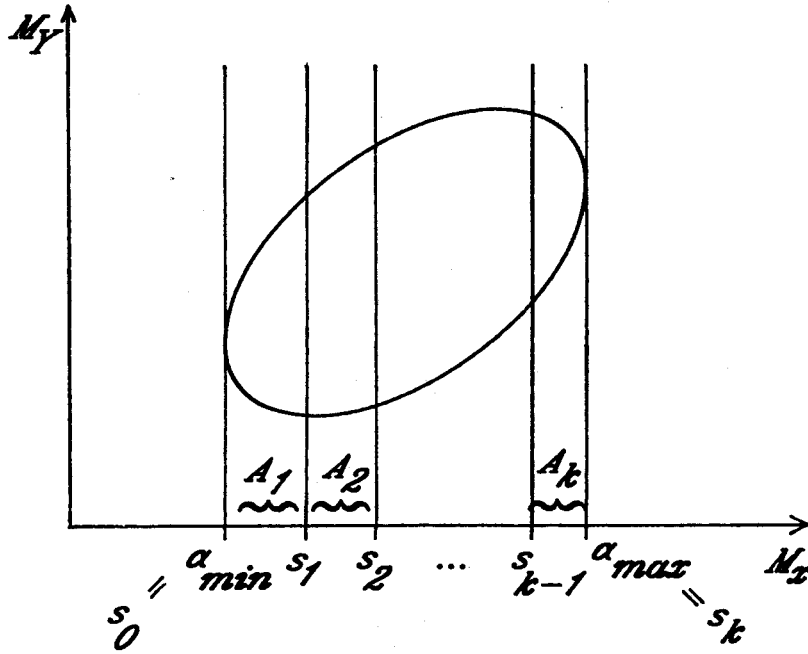


fig.4.5 : Discrétisation des modalités de x en k ensembles

Par définition même de l'entropie conditionnelle de Y sachant la discrétisation des modalités de x en k classes :

$$H(Y/A(S)) = \sum_{i=1}^k p(A_i) \cdot H(Y/x \in A_i)$$

Notons $H_k^*(\alpha_{\min}, \alpha_{\max}) = H_k^*(s_0, s_k)$ l'entropie conditionnelle optimale de Y par rapport à x dont on a effectué une partition en k classes de l'ensemble des modalités observées (comprises entre $\alpha_{\min}$ et $\alpha_{\max}$).

D'après le principe d'optimalité :

$$H_k(s_0, s_k) = p(A_1) \cdot H(Y/x \in A_1) + H_{k-1}^*(s_1, s_k).$$

Or, $p(A_1) \cdot H(Y/x \in A_1) = f(s_1)$ est une fonction de s_1 .

De même pour $H_{k-1}^*(s_1, s_k)$.

Ce qui entraîne que :

$$H_k^*(s_0, s_k) = \min_{s_1 \in]s_0, s_k[} [f(s_1) + H_{k-1}^*(s_1, s_k)]$$

Nous pouvons recommencer le même raisonnement pour $H_{k-1}^*(s_1, s_k)$:

$$H_{k-1}^*(s_1, s_k) = \min_{s_2 \in]s_1, s_k[} [f(s_2) + H_{k-2}^*(s_2, s_k)]$$

et ainsi de suite ...

Initialisation :

$H_1^*(s_{k-1}, s_k)$ est connu $\forall s_{k-1} \in]s_0, s_k[$.

$$\begin{aligned} H_1^*(s_{k-1}, s_k) &= p(A_k) \cdot H(Y/x \in A_k) \\ &= - p(A_k) \cdot \sum_{j=1}^m p(\beta_j/x \in A_k) \cdot \text{Log } p(\beta_j/x \in A_k) \end{aligned}$$

L'algorithme de recherche, par programmation dynamique, de la partition optimale d'ordre k des modalités observées de x est alors le suivant :

; PROGRAMMATION DYNAMIQUE

; initialisation

1. $\forall s_{k-1} \in]s_0, s_k[:$

$$H_1^*(s_{k-1}, s_k) = - p(A_k) \cdot \sum_{j=1}^m p(\beta_j/x \in A_k) \cdot \text{Log } p(\beta_j/x \in A_k)$$

; récurrence

2. pour $i:=0$ à $(k-2)$ faire

3. $H_{i+2}^*(s_{k-i-2}, s_k) = \min_{s_{k-i-1} \in]s_{k-i-2}, s_k[} \left[p(A_{k-i-1}) \cdot H(Y/x \in A_{k-i-1}) + H_{i+1}^*(s_{k-i-1}, s_k) \right]$
avec s_0 et s_k constantes.

; fin de boucle

; solution optimale

4. pour $i:=1$ à $(k-1)$ faire

5. $s_i^* = \min_{s_i} \{ H_{k-i+1}^*(s_{i-1}^*, s_k) \}$

; fin de boucle

; Fin de l'algorithme

III.3.1.2.2. Méthode du gradient

La méthode du gradient s'applique lorsque x est une variable numérique ($M_x \subset \mathbb{R}$).

La population d'apprentissage ne contenant qu'un nombre fini de points $(x(w), Y(w))$ $w \in \Omega$, il nous faut supposer qu'une estimation continue, $p(x, Y)$, de la densité de probabilité conjointe de x et de Y est disponible, pour $x \in [\alpha_{\min}, \alpha_{\max}]$.

Sous cette hypothèse, le raisonnement est le suivant : la partition sur l'ensemble des modalités observées de Y étant fixée, pour chaque partition P_k en k classes du segment $[\alpha_{\min}, \alpha_{\max}]$, nous pouvons calculer $H(Y/P_k)$.

$H(Y/P_k)$ est une fonction de $s_1, s_2, \dots, s_{k-1}$.

Minimiser $H(Y/P_k)$ pour avoir une partition optimale du segment $[\alpha_{\min}, \alpha_{\max}]$ revient donc à minimiser $H(Y/P_k)$ par rapport à $s_1, s_2, \dots, s_{k-1}$.

Une condition nécessaire de stationnarité est :

$$\delta H(Y/P_k) / \delta s_\mu = 0 \quad \text{pour } 1 \leq \mu \leq k-1.$$

Le signe δ représente ici une dérivée partielle.

Reprenons la définition même de l'entropie conditionnelle :

$$H(Y/P_k) = H(x, Y) - H(x) = - \sum_{i,j} p_{ij} \cdot \text{Log } p_{ij} + \sum_i p_i \cdot \text{Log } p_i$$

avec $p_i = \sum_j p_{ij}$; $\sum_{i,j} p_{ij} = 1$.

La relation entre p_{ij} et $p(x, Y)$ est donnée par :

$$p_{ij} = \int_j \int_{s_{i-1}}^{s_i} p(x, Y) \cdot dx \cdot dY$$

et ceci pour $i=1, \dots, k$; $j=1, \dots, m$ en posant $\alpha_{\min}=s_0$ et $\alpha_{\max}=s_k$.
(voir fig.4.4)

$$\frac{\delta H(Y/P_k)}{\delta s_\mu} = - \sum_{i,j} \left(\frac{\delta p_{ij}}{\delta s_\mu} \text{Log } p_{ij} + p_{ij} \frac{\delta \text{Log } p_{ij}}{\delta s_\mu} \right) + \sum_i \left(\frac{\delta p_i}{\delta s_\mu} \text{Log } p_i + p_i \frac{\delta \text{Log } p_i}{\delta s_\mu} \right)$$

et ceci, pour $\mu=1, \dots, k-1$.

$$\begin{aligned} \text{or : } \sum_i \left(\frac{\delta p_i}{\delta s_\mu} \text{Log } p_i + p_i \frac{\delta \text{Log } p_i}{\delta s_\mu} \right) &= \sum_i \left(\frac{\delta \sum_j p_{ij}}{\delta s_\mu} \text{Log } p_i + p_i \frac{\delta \sum_j p_{ij}}{\delta s_\mu} \right) \\ &= \sum_{i,j} \left(\frac{\delta p_{ij}}{\delta s_\mu} \text{Log } p_i + p_{ij} \frac{\delta \text{Log } p_i}{\delta s_\mu} \right) \end{aligned}$$

$\implies$

$$\frac{\delta H(Y/P_k)}{\delta s_\mu} = - \sum_{i,j} \left(\frac{\delta p_{ij}}{\delta s_\mu} \text{Log } p_{ij} + p_{ij} \frac{\delta \text{Log } p_{ij}}{\delta s_\mu} - \frac{\delta p_{ij}}{\delta s_\mu} \text{Log } p_i - p_{ij} \frac{\delta \text{Log } p_i}{\delta s_\mu} \right)$$

La condition nécessaire de stationnarité implique :

$$\frac{\delta H(Y/P_k)}{\delta s_\mu} = - \sum_{i,j} \frac{\delta p_{ij}}{\delta s_\mu} \text{Log } \frac{p_{ij}}{p_i} = 0 \quad \forall \mu=1, \dots, k-1.$$

ou encore :

$$\frac{\delta H(Y/P_k)}{\delta s_\mu} = - \sum_{i,j} \frac{\delta p_{ij}}{\delta s_\mu} \text{Log } p_{j/i} = 0 \quad \forall \mu=1, \dots, k-1.$$

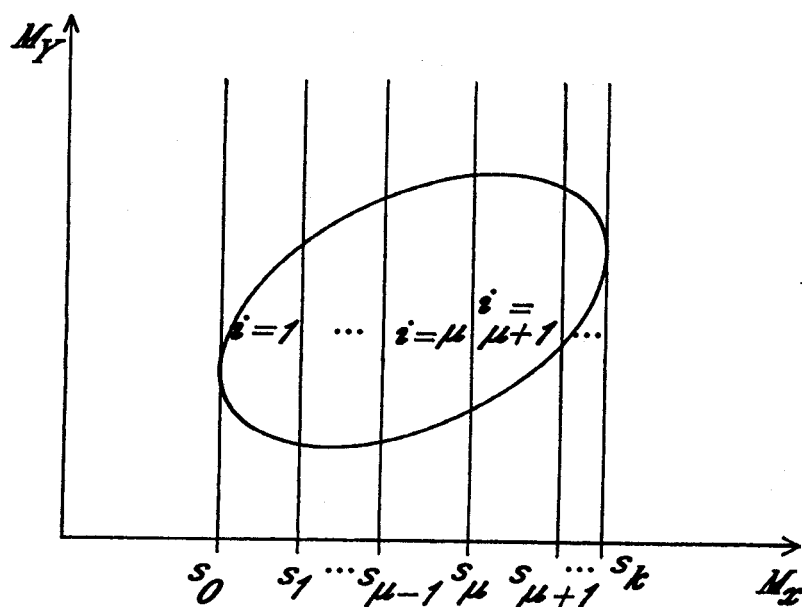


fig.4.6.

Nous pouvons remarquer que $\frac{\delta p_{ij}}{\delta s_{\mu}} = 0$ si $i \neq \mu, \mu+1$.

De plus, $p_{j/\mu} + p_{j/\mu+1} = \text{cte}$

Donc : $\frac{\delta p_{j/\mu}}{\delta s_{\mu}} + \frac{\delta p_{j/\mu+1}}{\delta s_{\mu}} = 0$

$$\begin{aligned} \frac{\delta H(Y/P_k)}{\delta s_{\mu}} &= - \sum_j \left(\frac{\delta p_{\mu j}}{\delta s_{\mu}} \text{Log } p_{j/\mu} + \frac{\delta p_{\mu+1, j}}{\delta s_{\mu}} \text{Log } p_{j/\mu+1} \right) \\ &= - \sum_j \left(\frac{\delta p_{\mu j}}{\delta s_{\mu}} \text{Log } p_{j/\mu} - \text{Log } p_{j/\mu+1} \right) \end{aligned}$$

$$\begin{aligned} \frac{\delta H(Y/P_k)}{\delta s_{\mu}} &= - \sum_{j=1} \frac{\delta p_{\mu j}}{\delta s_{\mu}} \text{Log } \frac{p_{j/\mu}}{p_{j/\mu+1}} = 0 \\ &= f(s_{\mu-1}, s_{\mu}, s_{\mu+1}). \end{aligned}$$

$$\begin{array}{c}
 \frac{\delta H(Y/P_k)}{\delta s_1} \\
 \frac{\delta H(Y/P_k)}{\delta s_2} \\
 \vdots \\
 \frac{\delta H(Y/P_k)}{\delta s_{\mu-1}} \\
 \frac{\delta H(Y/P_k)}{\delta s_{\mu}} \\
 \frac{\delta H(Y/P_k)}{\delta s_{\mu+1}} \\
 \vdots \\
 \frac{\delta H(Y/P_k)}{\delta s_{k-1}}
 \end{array}
 \begin{pmatrix}
 s_0 & s_1 & s_2 & \dots & s_{\mu-1} & s_{\mu} & s_{\mu+1} & \dots & s_k \\
 \times & \times & \times & & & \cdot & \cdot & \cdot & \\
 \cdot & \times & \times & \times & & \cdot & \cdot & \cdot & \\
 & & & & \times & \times & \times & \cdot & \cdot \\
 & & & & \cdot & \times & \times & \times & \cdot \\
 & & & & \cdot & \cdot & \times & \times & \times \\
 & & & & & & & & \times & \times & \times
 \end{pmatrix}
 = 0$$

fig.4.7 : Système à résoudre - Méthode du gradient

Nous obtenons donc un système de $(k-1)$ équations à $(k-1)$ inconnues qui sont : $s_1, s_2, \dots, s_{k-1}$.

Il existe donc une solution, la matrice bande diagonale ci-dessus étant en général inversible.

La formule du gradient est la suivante :

$$S_{i+1} = S_i - \tau \cdot \frac{\delta H(Y/P_k)}{\delta S}$$

où : $S = (s_1, s_2, \dots, s_{k-1})$

et $\frac{\delta H(Y/P_k)}{\delta S}$ représente les dérivées partielles de $H(Y/P_k)$ par rapport aux s_{μ} ($\mu=1, \dots, k-1$).

S_i représente le vecteur S à l'itération "i".

Le choix de τ est tout à fait arbitraire. Ce paramètre doit être suffisamment petit pour avoir une bonne précision des différents s_{μ} ($\mu=1, \dots, k-1$). Il doit aussi assurer une convergence suffisamment rapide.

L'initialisation de S (S_0) est aussi complètement arbitraire. Une solution possible est de choisir les s_μ ($\mu=1, \dots, k-1$) équidistants.

III.3.2. Cas de plusieurs variables explicatives ($p>1$)

III.3.2.1. Position du problème

Dans ce paragraphe, X représente le vecteur $(x_1, x_2, \dots, x_p)$ où x_i ($i=1, \dots, p$) est une variable explicative de Y . Les ensembles des modalités observées M_{x_i} de chaque variable x_i ($i=1, \dots, p$) sont ordonnés.

M_X , qui est l'ensemble des modalités observées de la variable vectorielle X , est inclus dans l'ensemble $(M_{x_1} \times M_{x_2} \times \dots \times M_{x_p})$.

Une relation d'ordre total ne peut être envisagée car *a priori* il n'existe aucune relation entre les différentes variables x_i ($i=1, \dots, p$). Une relation d'ordre ne peut qu'être partielle.

La question que l'on se pose naturellement est alors la suivante : Comment définir une partition P_k d'ordre k des modalités observées de $X=(x_1, x_2, \dots, x_p)$ sachant que les modalités observées des différentes variables x_i sont définies respectivement sur des ensembles quelconques ordonnés M_{x_i} ?

Une possibilité est de fixer, pour chaque variable x_i ($i=1, \dots, p$), un nombre donné k_i de classes formant une partition sur l'ensemble des modalités observées de cette variable. La partition sur X est alors l'intersection des partitions sur les x_i . Un algorithme de recherche des partitions stables sera développé.

III.3.2.2. Algorithme proposé

Pour chaque variable explicative x_i ($i=1, \dots, p$), nous nous fixons un nombre donné k_i de classes de la partition de M_{x_i} .

A titre d'exemple, pour $p=2$ (c'est-à-dire $X=(x_1, x_2)$), nous pourrions chercher une partition en deux classes sur x_1 , et une partition en trois classes sur x_2 .

Graphiquement, cela se traduit comme suit :

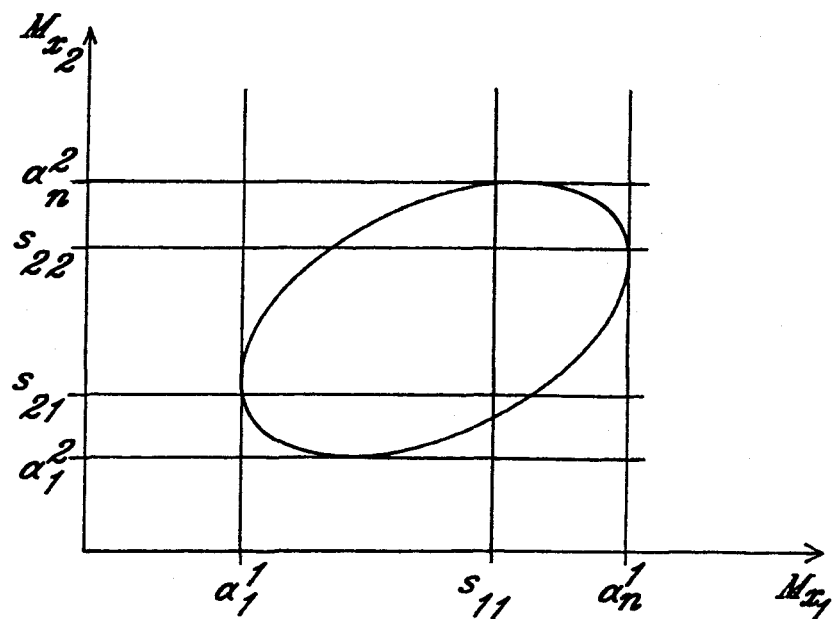


fig.4.8 : Espace des variables explicatives -
Méthode "axe par axe"

où $\{\alpha_i = (\alpha_i^1, \alpha_i^2, \dots, \alpha_i^p) ; i=1, \dots, n\}$ est l'ensemble des modalités observées de $X=(x_1, x_2, \dots, x_p)$; et où s_{ij} ($j=1, \dots, k_i$) sont les séparateurs des classes de la partition de l'ensemble des modalités observées de la variable x_i .

Par la suite, nous appellerons $P(x_i)$ la partition des modalités observées de la variable x_i en k_i classes ($i=1, \dots, p$).

Présentation de l'algorithme :

On développera ici une méthode axe par axe : l'algorithme ajuste les séparateurs s_{ij} ($j=1, \dots, k_i$) pour la variable x_i , tandis que les autres $s_{\mu j}$ ($\mu \neq i$) restent fixes. On retient les valeurs des s_{ij} ($j=1, \dots, k_i$) pour lesquelles $H(Y/P(x_1)P(x_2)\dots P(x_p))$ est minimale.

Nous pouvons ainsi utiliser un algorithme de transfert, la programmation dynamique ou la méthode du gradient pour trouver une partition optimale sur une variable.

On recommence le même raisonnement avec une autre variable, etc..., jusqu'à stabilisation des séparateurs s_{ij} ($i=1, \dots, p ; j=1, \dots, k_i$).

Nous appellerons $P^T(x_i)$ la partition $P(x_i)$ à l'itération T .

L'initialisation des différentes partitions, notée $P^0(x_1)P^0(x_2)\dots P^0(x_p)$, est quelconque. Elle dépend du choix de l'analyste. Celui-ci pourrait par exemple choisir les différents s_{ij} ($j=1, \dots, k_i$) équidistants sur $[\alpha_{\min}^i, \alpha_{\max}^i[$.

L'algorithme est le suivant :

```

; Recherche d'une Partition Stable
; avec les contraintes suivantes :
;   →  $M_X$  est partiellement ordonné
;   →  $X=(x_1, x_2, \dots, x_p)$  ;  $p > 1$ 
; -----
; Initialisation
1.  $P^0(X) \leftarrow$  partition initiale
2. Stable  $\leftarrow$  faux
3.  $T \leftarrow 0$ 
; Boucle
4. Tant que (non Stable) Faire
5.    $T \leftarrow T+1$ 
6.   Pour  $i:=1$  à  $p$  Faire
7.     Trouver la partition  $P^T(x_i)$  sur  $M_{x_i}$  qui minimise
       l'entropie conditionnelle
;     Fin du Pour
8.   Pour  $i:=1$  à  $p$  Faire
9.     Si  $P^T(X) = P^{T-1}(X)$  Alors Stable  $\leftarrow$  vrai
       Fin du Pour
;   Fin du Tant que
; Fin de l'algorithme.

```

L'opération (7) pourra être effectuée, au choix de l'analyste, par :

- la méthode du gradient (si x_i est numérique)
- ou l'algorithme de transfert
- ou la programmation dynamique.

Convergence de l'algorithme :

On cherche les partitions optimales sur les ensembles de modalités observées des différentes variables x_i ($i=1, \dots, p$).

A chaque itération, nous prenons la partition qui minimise la quantité $H(Y/.).$

$$\begin{aligned}
 H(Y/P^0(x_1)P^0(x_2)\dots P^0(x_p)) & \leftarrow \text{initialisation} \\
 H(Y/P^1(x_1)P^0(x_2)\dots P^0(x_p)) & \leq H(Y/P^0(x_1)P^0(x_2)\dots P^0(x_p)) \\
 H(Y/P^1(x_1)P^1(x_2)\dots P^0(x_p)) & \leq H(Y/P^1(x_1)P^0(x_2)\dots P^0(x_p)) \\
 & \vdots \\
 & \vdots \\
 H(Y/P^1(x_1)P^1(x_2)\dots P^1(x_p)) & \leq H(Y/P^1(x_1)\dots P^1(x_{p-1})P^0(x_p)) \\
 H(Y/P^2(x_1)P^1(x_2)\dots P^1(x_p)) & \leq H(Y/P^1(x_1)P^1(x_2)\dots P^1(x_p)) \\
 & \vdots \\
 & \vdots
 \end{aligned}$$

Nous arriverons donc obligatoirement à une stabilisation des séparateurs s_{ij} ($i=1, \dots, p$; $j=1, \dots, k_i$) des différentes variables x_i .

Cet algorithme est intéressant car il reprend les résultats antérieurs pour $p=1$: programmation dynamique, méthode du gradient, algorithme de transfert (cf §III.3.1).

IV. METHODES D'INDUCTION SYMBOLIQUE

IV.1. Les données d'apprentissage

Lorsque les données d'apprentissage sont cohérentes, le problème de l'explication consiste à chercher des fonctions de reconnaissance pour chaque modalité de la variable à expliquer Y . Les méthodes symboliques (*INDUCE*, ...), ainsi que les méthodes de bloc-sérialisation [MARC 90] sont alors tout à fait indiquées pour résoudre ce type de problème. Il existe donc une partition de M_X qui induit une partition de M_Y en m classes ($m = \text{card } M_Y$). Les règles produites sont ainsi complètes sur Σ (toutes les observations sont prises en compte), en cohérence logique (de type 1) et de dépendance perçue.

$M_Y \backslash M_X$	β_1	β_2	β_3	β_4
$\bar{\alpha}_1$				
$\bar{\alpha}_2$				
$\bar{\alpha}_3$				
$\bar{\alpha}_4$				

fig.4.9 : Apprentissage dans le cas de données cohérentes

Lorsque les données d'apprentissage sont trop incohérentes, on applique sur celles-ci une procédure de contraste, avec σ_i faible (fort contraste). Chaque règle ainsi obtenue est alors caractérisée par son degré de croyance, qui peut être le coefficient de vé-

rité introduit au paragraphe II.3, ainsi que par son coefficient d'efficacité (§II.4).

On travaillera donc sur ces *données contrastées* (qui sont un peu plus cohérentes entre elles) : plus les données sont incohérentes, plus le contraste sera fort (efficacité accrue), et plus le degré de croyance de la règle en question sera faible (τ_p fort).

Dans les deux cas, le modèle atomique constitué, on peut envisager d'étendre ces règles afin d'apporter des simplifications possibles des parties prémisses.

IV.2. La procédure de généralisation

La procédure d'apprentissage consistera à rechercher des règles plus générales sous la forme de clause de Horn. La partie prémisse est une assertion de A_x et la partie conclusion, un événement élémentaire défini sur M_y .

Le problème de généralisation consiste alors d'une part, à agréger des règles, et d'autre part, à les prolonger de telle façon que la partie prémisse de la nouvelle règle soit une assertion.

A titre d'exemple, pour le deuxième tableau de l'annexe 4, la procédure de généralisation appliquée sur les deux règles suivantes :

$[X_1=\text{petit}] \& [X_2=\text{hexa.}] \Rightarrow [X_3=\text{cannelé}]$

$[X_1=\text{grand}] \& [X_2=\text{hexa.vcyl.}] \Rightarrow [X_3=\text{cannelé}]$

donnera la règle suivante :

$[X_1 = \text{petit} \vee \text{grand}] \& [X_2 = \text{hexa.vcyl.}] \Rightarrow [X_3 = \text{cannelé}]$.

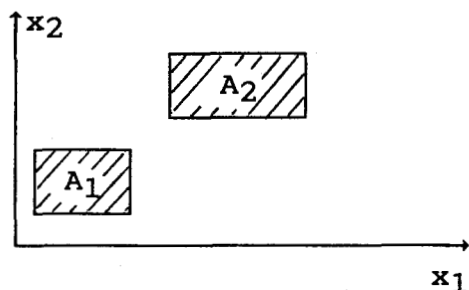
Nous avons donc bien agrégé les deux règles initiales (reconnaisant respectivement les exemples e_2 et $\{e_6, e_9\}$), et prolongé par la modalité non observée $[X_1=\text{petit}] \& [X_2=\text{cyl.}]$.

remarque :

Une contrainte d'ordre peut éventuellement être formulée sur les modalités des variables explicatives ordinales. On ne pourra alors travailler que sur des modalités consécutives, ou que sur des ensembles de modalités consécutifs.

Le schémas suivant représente ce cas de figure : x_1 et x_2 sont des variables explicatives ordinales.

avant généralisation des règles
 $A_1 \Rightarrow B_1$ et $A_2 \Rightarrow B_2$



généralisation des règles
 $A_1 \Rightarrow B_1$ et $A_2 \Rightarrow B_2$

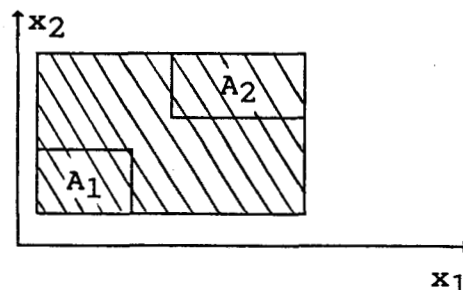


fig.4.10 : Généralisation de règles sous contrainte d'ordre

La procédure idéale serait de rechercher un ensemble de règles (au moins) en cohérence perçue sur les données contrastées. Par exemple, on recherchera une partition de M_X avec un recouvrement sur O . L'incohérence (vis-à-vis des données contrastées) ne sera alors pas vérifiée, les modalités provoquant celle-ci n'étant pas observées. On peut dès lors évaluer le risque à prolonger les règles en considérant l'une des deux hypothèses proposées au chapitre III (cf §V.2).

En pratique, pour des données d'apprentissage fortement incohérentes, on essaiera de s'en approcher. On cherchera alors des règles se recouvrant le moins possible. On généralisera celles-ci tant qu'elles obéissent à un critère donné.

Exemple d'algorithme :

- 1- On part des données d'apprentissage initiales si celles-ci sont (plus ou moins) cohérentes, et des données contrastées sinon.
- 2- On généralisera les règles. La règle généralisée retenue sera celle qui minimisera son taux de pollution.
- 3- On recommence l'opération 2 tant qu'il existe un taux de pollution inférieur à un taux de pollution "seuil", fixé par l'analyste.

De multiples algorithmes de généralisation peuvent être construits, à partir des définitions, des propriétés, et des hypothèses de travail énoncées au chapitre III.

CONCLUSIONS

La *procédure d'apprentissage* consiste à chercher un ensemble de règles liant les modalités (observées ou observables, suivant le cas) des *variables explicatives* (notées X) et les modalités observées de la *variable à expliquer* (notée Y). Cette procédure produit des règles plus générales que les règles initiales composant le *modèle atomique*. Celui-ci permet de séparer, dans le cas de données *incohérentes*, le *comportement déterministe* du système, du *comportement aléatoire* inhérent à celui-ci.

Deux approches sont dès lors utilisables dans une procédure d'apprentissage :

→ une *approche globale* construisant une partition de la *population d'apprentissage*. Le problème de l'explication consiste alors à chercher la meilleure variable $\tilde{X}$ telle que $Y \# f(\tilde{X})$.

Nous proposons différents algorithmes basés sur un *critère entropique* mesurant l'*intensité de liaison entre variables*. Ces procédures permettent de considérer (si l'analyste le désire) la contrainte d'ordre sur les modalités des variables explicatives ordinales. Le choix de l'algorithme utilisé incombe à l'analyste (voir les perspectives).

Nous remarquons que depuis quelques années, des travaux de recherche ont été entrepris concernant l'apport de la théorie de l'information en apprentissage automatique : [GOOD 88], [MANT 91] (évaluation de la distance entre partitions), [KONO 91] (classifieurs), ... L'intérêt majeur de la théorie de l'information est qu'elle apporte des critères permettant une évaluation globale du modèle obtenu, ainsi qu'une certaine robustesse des résultats (dû au caractère numérique des critères proposés). [TORO 82], [SBAI 83], [BARB 87], ... ont développé des outils issus de la théorie de l'information appliquée à l'*analyse structurale des systèmes*.

→ une *approche locale* construisant des couvertures de parties de l'ensemble des *modalités observables* des variables explicatives (ou recouvrement de la population d'apprentissage).

La procédure d'apprentissage génère des règles caractérisées par certaines propriétés (différents types de *cohérence*, ...). Des critères permettent également d'évaluer

les règles ainsi obtenues (taux de pollution, taux de couverture, ...).

Deux opérations de *généralisation*, basées sur les notions de modalités observées et observables des variables explicatives, ont été envisagées : l'*agrégation* et le *prolongement*.

Suivant l'hypothèse de travail considérée (*monde clos*, *non contradiction*, *continuité*), nous donnons les moyens d'évaluer les résultats obtenus par la procédure d'apprentissage.

Différentes procédures d'*induction symbolique* peuvent dès lors être construites. Nous terminons ce travail en proposant un algorithme. Celui-ci constitue en fait une ouverture possible vers d'autres procédures d'apprentissage.

L'inconvénient majeur des méthodes symboliques est la non robustesse des résultats obtenus lorsque les données sont trop incohérentes.

... ET PERSPECTIVES

Le formalisme des différents algorithmes d'apprentissage présentés au chapitre IV est tout à fait général, parce que théorique.

Nous avons l'intention d'appliquer ces procédures d'apprentissage sur des exemples concrets de taille importante (de petits exemples ont déjà été traités), et de comparer les résultats obtenus. On pourrait dès lors définir des critères de choix d'utilisation des différents algorithmes proposés.

Il serait également important de comparer leurs performances à celles d'autres algorithmes déjà existants (*ID3*, *INDUCE*, *GENREG*, ...).

Une autre voie de recherche conduit à comparer les règles obtenues à chaque itération de la procédure d'apprentissage. Il faudrait alors définir des indices de dissimilarités entre les règles. Certains auteurs commencent à aborder ce problème : [CARV 90] pour des objets symboliques (approche locale), et [MANT 91] pour la construction de partitions (approche globale).

Une procédure d'apprentissage consiste à trouver des règles plus générales que les règles initiales, suivant un critère donné. On pourrait dès lors chercher le "degré" de *simplicité* d'une règle. Intuitivement, nous pouvons dire que la simplicité d'une règle se caractérisera par sa partie prémisses.

Quelques critères de simplicité ont déjà été établis [KOD 86], p.90 : "On peut dire qu'une fonction est simple quand elle contient moins qu'un nombre fixé à l'avance de descripteurs, ou de disjonctions, etc ...", mais aucun résultat général et aucune application du concept de simplicité dans la procédure d'apprentissage n'existent actuellement à notre connaissance.

La résolution de ces problèmes pourra faire l'objet de recherches ultérieures.

ANNEXES

ANNEXE 1 : Le langage de représentation VL

Le langage de représentation VL (*Variable-Valued Logic calculus*) a été proposé par R.S. Michalski [MICH 80,83a] pour l'apprentissage inductif. Il peut être vu comme une extension du calcul des prédicats.

Introduisons tout d'abord deux définitions.

→ *Fonction atomique* :

Une fonction atomique est :

- une variable
- ou une fonction (avec entre ses parenthèses une séquence de fonctions atomiques ou/et de constantes).

→ *Sélecteur* :

Un sélecteur est une forme $[L\#R]$, où :

- L est une fonction atomique (ou une séquence de fonctions atomiques séparées par des "." qui représentent des conjonctions).
- # est un opérateur relationnel. Il appartient à l'ensemble $\{=, \neq, \geq, \leq, >, <\}$.
- R est :
 - . une constante
 - . une fonction atomique
 - . une séquence de constantes ou de fonctions atomiques séparées par :
 - "," (disjonction interne (de modalités))
 - ".." (intervalle, range operator)

→ *Le langage de représentation VL* :

Une expression VL est définie par les règles suivantes :

(i) Les deux constantes TRUE et FALSE, ainsi que "?" sont des VL-expressions.

(ii) Un sélecteur est une VL expression.

(iii) Si v , v_1 et v_2 sont des VL expressions, alors le sont également :

- (v) (une VL expression entre parenthèses)
 - $\neg v$ (le complément d'une VL expression)
 - $v_1 \cdot v_2$ (ou $v_1 v_2$: conjonction)
 - $v_1 \vee v_2$ (disjonction)
 - $v_1 \setminus v_2$ (v_1 sans v_2)
 - $v_1 \models v_2$ (métainplication)
- où $\models \in \{\rightarrow, \leftarrow, ::, \Rightarrow, \leq, \models\}$
- $\rightarrow$ (implication)
 - $\leftarrow$ (équivalence)
 - $::$ (règle de décision)
 - $\Rightarrow$ (règle d'inférence)
 - $\leq$ (généralisation : v_2 est plus général que v_1)
 - $\models$ (équivalence sémantique : lorsque les deux ensembles sont égaux).

ANNEXE 2 : REGLES DE GENERALISATION

Ces règles de généralisation sont décrites dans [MICH 80,83a] et [KOD 86].

→ Règle de l'oubli des conjonctions (*the dropping condition rule*) :

$A \& B$ est plus particulier que A .

→ Règle de l'ajout des disjonctions (*the generalization by internal disjonction* ou *the extending reference rule*) :

C'est la règle symétrique à celle de l'oubli des conjonctions : $A \vee B$ est toujours plus général que A .
On définit deux cas particuliers :

- Règle d'agrandissement des domaines :

$$A(x) \& x \in \{x_1, \dots, x_n\} \equiv A(x_1) \vee \dots \vee A(x_n) \quad | \leq \quad A(x_1) \vee \dots \vee A(x_n) \vee A(x_{n+1})$$

où A est un prédicat.

- Règle de l'intervalle (*the closing interval rule*) :

$$A(x) \& x = a \quad \& \quad A(x) \& x = b \quad | \leq \quad A(x) \& x \in [a..b]$$

(si x est une variable ordonnée ; $a < b$).

→ the Extension against rule :

$$T \& A \implies B \quad \& \quad T \& C \implies \neg B \quad | \leq \quad T \& \neg C \implies B$$

où $A \cap C = \emptyset$

→ Règle de transformation des constantes en variables (*the turning constants to variables rule*) :

Si A est un prédicat d'arité (nombre de fois que le prédicat est vrai) plus grande que zéro, et si CTE est une constante, alors $A(x)$ est plus général que $A(CTE)$.

→ Règle de suppression des prémisses des implications :

B est plus général que $A \implies B$

→ Règle d'ajout de noms de variables :

Si une variable possède plusieurs occurrences, $A(x,y)$ est plus général que $A(\bar{x},x)$.

→ Règle de remontée dans l'arbre de généralisation - encore appelé "espace des versions" (*the climbing generalization tree rule*) :

Cette règle s'adresse particulièrement aux variables structurées. A titre d'exemple, dans l'annexe 4, X_2 =révolution est plus général que X_2 =conique.

Le système LEX [MITC 83] est basé sur cette règle.

ANNEXE 3 : Classification des modèles obtenus par
la procédure d'apprentissage

On considérera que la procédure d'apprentissage produit des règles complètes sur Σ : toutes les modalités observées de Σ sont prises en compte.

πO_i i	O	M_X	type de cohérence
partition	partition	partition	logique de type 1
recouvrement* et $\forall a \in \bigcap_{i \in I} A_i \neq \emptyset$ $\exists b \in \bigcap_{i \in I} B_i$ avec $a \in \pi O_i \setminus O$	partition	partition	logique de type 2
recouvrement et $\exists a \in \bigcap_{i \in I} A_i \neq \emptyset$ / $\bigcap_{i \in I} B_i = \emptyset$ avec $a \in \pi O_i \setminus O$	partition	partition	absolue
recouvrement et $\forall a \in \bigcap_{i \in I} A_i \neq \emptyset$ $\exists b \in \bigcap_{i \in I} B_i$ avec $a \in \pi O_i \setminus O$	recouvrement et $\forall a \in \bigcap_{i \in I} A_i \neq \emptyset$ $\exists b \in \bigcap_{i \in I} B_i$ avec $a \in O \setminus M_X$	partition	logique de type 2
recouvrement et $\exists a \in \bigcap_{i \in I} A_i \neq \emptyset$ / $\bigcap_{i \in I} B_i = \emptyset$ avec $a \in \pi O_i \setminus O$	recouvrement et $\forall a \in \bigcap_{i \in I} A_i \neq \emptyset$ $\exists b \in \bigcap_{i \in I} B_i$ avec $a \in O \setminus M_X$	partition	absolue
recouvrement et dans les deux cas	recouvrement et $\exists a \in \bigcap_{i \in I} A_i \neq \emptyset$ / $\bigcap_{i \in I} B_i = \emptyset$ avec $a \in O \setminus M_X$	partition	perçue

* On appellera "recouvrement" une couverture avec chevauchement.

recouvrement et $\forall a \in N_{A_i} \neq \emptyset$ $\exists b \in N_{B_i}$ $\text{avec } a \in \pi O_i \setminus 0$	recouvrement et $\forall a \in N_{A_i} \neq \emptyset$ $\exists b \in N_{B_i}$ $\text{avec } a \in O \setminus M_X$	recouvrement et $\forall a \in N_{A_i} \neq \emptyset$ $\exists b \in N_{B_i} /$ $r_{a,b}$ vraie et $a \in M_X$	logique de type 2
recouvrement et $\exists a \in N_{A_i} \neq \emptyset /$ $N_{B_i} = \emptyset$ $\text{avec } a \in \pi O_i \setminus 0$	recouvrement et $\forall a \in N_{A_i} \neq \emptyset$ $\exists b \in N_{B_i}$ $\text{avec } a \in O \setminus M_X$	recouvrement et $\forall a \in N_{A_i} \neq \emptyset$ $\exists b \in N_{B_i} /$ $r_{a,b}$ vraie et $a \in M_X$	absolue
recouvrement et dans les deux cas	recouvrement et $\exists a \in N_{A_i} \neq \emptyset /$ $N_{B_i} = \emptyset$ $\text{avec } a \in O \setminus M_X$	recouvrement et $\forall a \in N_{A_i} \neq \emptyset$ $\exists b \in N_{B_i} /$ $r_{a,b}$ vraie et $a \in M_X$	perçue
recouvrement et dans les deux cas	recouvrement et dans les deux cas	recouvrement et $\exists a \in N_{A_i} \neq \emptyset /$ $\nexists b \in N_{B_i}$ $\text{avec } r_{a,b} \text{ vraie}$ et $a \in M_X$	incohérence perçue

remarque :

Si on effectue une partition sur M_X (construction d'une variable $\bar{X}$ plus grosse que X), on est au moins sûr d'avoir une cohérence perçue.

ANNEXE 4 :1^{er} exemple :

Tableau extrait de [MING 89a], p.320, et représentant les modalités de deux attributs à partir de données sur le cancer du sein :

radiation	menopause	class
no	<60	recur
no	≥60	recur
no	<60	recur
no	not	recur
yes	≥60	not recur
yes	<60	not recur
yes	≥60	not recur
no	not	not recur
no	<60	not recur
no	<60	recur

On cherchera à caractériser les classes "recur" et "not recur".

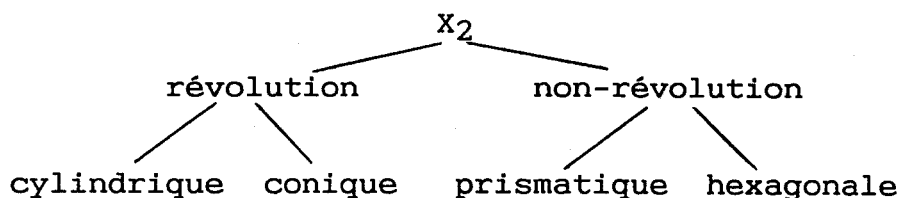
2^{ème} exemple :

Tableau extrait de [BOUZ 90] : 10 pièces mécaniques sont décrites par 3 variables.

	diamètre X ₁	forme-pièce X ₂	type-alésage X ₃
e1	petit	conique	lisse
e2	petit	hexagonale	cannelé
e3	petit	conique	claveté
e4	moyen	prismatique	claveté
e5	moyen	conique	lisse
e6	grand	hexagonale	cannelé
e7	grand	cylindrique	lisse
e8	grand	prismatique	claveté
e9	grand	cylindrique	cannelé
e10	grand	prismatique	claveté

→ La variable X₁ est une variable linéaire (petit<moyen<grand).

→ X₂ est une variable nominale structurée :



→ X₃ est une variable nominale.

ANNEXE 5 :

Tableau extrait de [MARC 87] (proposé par G. Saporta) :

27 races de chiens ont été caractérisées par 7 paramètres :
taille (Tail.), poids, vitesse (Véloc.), intelligence (Intell.),
affection (Aff.), agresseivité (Agr.), Fonction.

	Tail.	Poids	Véloc.	Intell.	Aff.	Agr.	Fonction
	-,0,+	-,0,+	-,0,+	-,0,+	-,+	-,+	Ut,Ch,Co
1 beauceron	+	0	+	0	+	+	Ut
2 basset	-	-	-	-	-	+	Ch
3 berger allemand	+	0	+	+	+	+	Ut
4 boxer	0	0	0	0	+	+	Co
5 bull-dog	-	-	-	0	+	-	Co
6 bull-mastiff	+	+	-	+	-	+	Ut
7 caniche	-	-	0	+	+	-	Co
8 chihuahua	-	-	-	-	+	-	Co
9 cocker	0	-	-	0	+	+	Co
10 colley	+	0	+	0	+	-	Co
11 dalmatien	0	0	0	0	+	-	Co
12 dobermann	+	0	+	+	-	+	Ut
13 dogue allemand	+	+	+	-	-	+	Ut
14 épagneul breton	0	0	0	+	+	-	Ch
15 épagneul franç.	+	0	0	0	-	-	Ch
16 fox-hound	+	0	+	-	-	+	Ch
17 fox-terrier	-	-	0	0	+	+	Co
18 grand bleu	+	0	0	-	-	+	Ch
19 labrador	0	0	0	0	+	-	Ch
20 lévrier	+	0	+	-	-	-	Ch
21 mastiff	+	+	-	-	-	+	Ut
22 pékinois	-	-	-	-	+	-	Co
23 pointer	+	0	+	+	-	-	Ch
24 saint-bernard	+	+	-	0	-	+	Ut
25 setter	+	0	+	0	-	-	Ch
26 teckel	-	-	-	0	+	-	Co
27 terre-neuve	+	+	-	0	-	-	Ut

où :

→ -,0,+ signifient respectivement "faible", "moyen" et "fort".

→ Ut, Ch, Co signifient respectivement "chien Utilitaire", "chien de Chasse" et "chien de Compagnie".

A titre d'exemple, le *chihuahua* est caractérisé par une taille, un poids, une vitesse, une intelligence, et une agresseivité faibles, par une affection forte, et par une fonction de chien de compagnie.

On cherchera par exemple à caractériser les concepts "chien Utilitaire", "chien de Chasse" et "chien de Compagnie".

REFERENCES BIBLIOGRAPHIQUES

- [ABID 79] Z. ABID
"Contribution à l'analyse structurale par la théorie des demi-treillis", Thèse de 3^e cycle, Univ. Claude Bernard, Lyon I, 1979.
- [AGGA 74] N.L. AGGARWAL
"Mesures d'information et questionnaires arborescents", Thèse 3^e cycle, Besançon, 1974.
- [AGGA 76] N.L. AGGARWAL
"Mesures d'information : caractéristiques et propriétés", Ecole de l'INRIA* : Théorie de l'Information, 1976.
- [ASH 65] R. ASH
"Information theory", John Wiley & Sons, 1965.
- [ASHB 65] W.R. ASHBY
"Measuring the internal informational exchange in a system", Cybernetica Namur, Belgique, vol.8, 1965.
- [AYGA 86] P. AYGALINC
"Application de la reconnaissance des formes au diagnostic médical, sélection multi-critère des variables explicatives", Thèse de Doctorat, USTLFA*, 1986.
- [BARB 87a] M. BARBOUCHA
"Modélisation structurale des systèmes complexes - Extraction et validation des règles d'un système expert", Thèse d'état, USTLFA, 26 juin 1987.
- [BARB 87b] M. BARBOUCHA, M. STAROSWIECKI, P. AYGALINC
"Building rules from contingency tables", 5^e Journées Internationales "Analyse des Données et Informatique", INRIA, 29 sept-2 oct 1987, Versailles, tome 1.
- [BARB 70] M. BARBUT, B. MONJARDET
"Ordre et classification - algèbre et combinatoire", tome 1&2, Hachette Université, 1970.
- [BENZ 80] J.P. BENZECRI
"Analyse des données", Dunod, 1980.
- [BIRK 67] G. BIRKHOFF
"Lattice theory", Amer. Math. Soc., Providence, 3^e éd., 1967.
- [BORD 86] J.P. BORDAT
"Calcul pratique du treillis de Galois d'une correspondance", Math. Sci. Hum., 24^e année, n°96, 1986, pp.31-47.

- [BOUC 90] B. BOUCHON-MEUNIER, S. DESPRES, D. DUBOIS, O. GASCUEL, A. GUENOCHÉ, H. PRADE
"Aspects de l'interface entre symbolique et numérique",
 3^e Journées Nationales PRC-GDR* Intelligence Artificielle,
 5-7 mars 1990, CNIT, Paris-la-Défense, Hermes, pp.90-112.
- [BOUR 89] J.M. BOUROCHE, G. SAPORTA
"L'analyse des données", Coll. Que sais-je ? , Presses
 Universitaires de France, 1989, 4^e éd.
- [BOUZ 90] L. BOUZID, B. MUTEL
*"Application de la sériation par blocs au problème de la
 classification symbolique de produits"*,
 Colloque International sur "les Méthodes de Blocs Sériation
 et applications", ENSAIS*, Strasbourg, 3-5 avril 1990.
- [BREI 84] L. BREIMAN, J. FRIEDMAN, R. OLSHEN, C. STONE
"Classification and regression trees",
 Belmont, CA : Wadsworth International Group, 1984.
- [BRUY 78] M. BRUYNOOGHE
*"Classification ascendante hiérarchique de grands ensembles
 de données : un algorithme rapide fondé sur la construction
 des voisinages réductibles"*,
 Les Cahiers de l'Analyse des Données, 3, 1, 1978.
- [BRUY 89] M. BRUYNOOGHE
*"Nouveaux algorithmes en classification automatique appli-
 cables aux très grands ensembles de données, rencontrés en
 traitement d'images et en reconnaissance des formes"*,
 Thèse d'état, Univ. Paris 6, 23 janvier 1989.
- [CARR 68] J.D. CARROLL
*"A generalization of canonical correlation analysis to three
 or more sets of variables"*, 76th Convention American
 Psychological Association, 1968.
- [CARV 90] F.A.T. de CARVALHO
"Dissimilarité en analyse des données symboliques",
 JFA* 90, Lannion, 25-26 avril 1990, pp.47-75.
- [CATA 90] L. CATACH
*"Logiques non classiques - Présentation et applications à
 l'intelligence artificielle"*, Modèles Logiques et Systèmes
 d'Intelligence Artificielle, Hermes, série Intelligence Ar-
 tificielle, Paris, 1990, pp.19-49.
- [CELE 89] G. CELEUX, E. DIDAY, G. GOVAERT, Y. LECHEVALLIER,
 H. RALAMBONDRAIN
"Classification automatique des données", Dunod, 1989.
- [CIBO 87] Ph. CIBOIS
"L'analyse factorielle", Coll. Que sais-je ? , Presses
 Universitaires de France, 1987, 2^e éd.

- [CLAR 89] P. CLARK, T. NIBLETT
"The CN2 induction algorithm",
 Machine Learning, 3, 1989, pp.261-283.
- [CONA 72] R.C. CONANT
"Detecting subsystems of a complex system",
 IEEE Trans. on Systems Man and Cybernetics, vol. SMC-2, n°4,
 sept. 1972, pp.550-553.
- [COOM 64] C.H. COOMBS
"A theory of data", New-York, Wiley, 1964.
- [DEC 89a] Ch. DECAESTECKER
"Formation incrémentale de concepts par un critère d'adéquation", 4^e JFA, mai 1989, pub. INRIA Rocquencourt.
- [DEC 89b] Ch. DECAESTECKER
"Incremental concept formation via a suitability criterion",
 Data Analysis, Learning Symbolic and Numeric Knowledge,
 INRIA, Antibes, 11-14 sept. 1989, pp.435-442.
- [DELA 71] P. DELATTRE
"Système, structure, fonction, évolution",
 Maloine-Doin, Paris, 1971, p.11.
- [DID 72] E. DIDAY
"Nouvelles méthodes et nouveaux concepts en classification automatique et reconnaissance des formes",
 Thèse de doctorat, Univ. Paris 6, 4 déc. 1972.
- [DID 82] E. DIDAY, J. LEMAIRE, J. POUGET, F. TESTU
"Eléments d'analyse de données", Dunod, 1982.
- [DID 89] E. DIDAY
"Introduction à l'approche symbolique en analyse des données", RAIRO*, Recherche opérationnelle/operations research, AFCET*-Gauthier-Villars, vol. 23, n°2, 1989, pp.193-236.
- [DID 90] E. DIDAY
"Introduction à l'analyse des données symboliques",
 JFA 90, Lannion, 25-26 avril 1990, pp.131-168.
- [DUBO 79] D. DUBOIS, H. PRADE
"Fuzzy real algebra : some results", Fuzzy Sets and Systems
 2, North-Holland Publishing Company, 1979, pp.327-348.
- [DUBU 90] B. DUBUISSON
"Diagnostic et reconnaissance de formes",
 Hermes, Paris, 1990.
- [DUQ 86] V. DUQUENNE
"Contextual implications between attributes and some representation properties for finite lattices", Colloque Arbeitstagung Begriffsanalyse, Darmstadt, 17-19 janvier 1986.

- [FENE 79] J.P. FENELON, L. LEBART, A. MORINEAU
"Traitement des données statistiques",
 Dunod, Méthodes+Programmes, 1979.
- [FORG 65] E.W. FORGY
"Cluster analysis of multivariate data",
 Biometrics, vol. 21, n°3, sept. 1965.
- [FOUC 85] T. FOUCART
"Analyse factorielle - programmation sur micro-ordinateurs",
 Masson, Méthode+Programmes, 1985.
- [FU 74] K.S. FU
"Syntactic methods in pattern recognition",
 Academic Press, 1974.
- [GANA 87a] J.G. GANASCIA
"AGAPE et CHARADE : deux mécanismes d'apprentissage symbolique appliqués à la construction de bases de connaissance",
 Thèse d'état, Univ. Paris-Sud, mai 1987.
- [GANA 87b] J.G. GANASCIA
"CHARADE : apprentissage de bases de connaissance",
 Journées "Symbolique-Numérique" pour l'Apprentissage de Connaissances à partir des Données, organisées par Diday & Kodratoff, Univ. Paris-Dauphine, 8-9 déc. 1987.
- [GARN 56] W.R. GARNER, N.J. Mc GILL
"The relation between information and variance analysis",
 Psychometrika, vol. 21, 1956, pp.219-228.
- [GASC 86] O. GASCUEL
"PLAGE : a way to give and use knowledge in learning",
 1st European Working Session on Learning, LRI*, Orsay, 1986.
- [GASC 87] O. GASCUEL
"Un algorithme pour explorer l'espace des versions - illustration sur une application", Journées "Symbolique-Numérique" pour l'Apprentissage de Connaissances à partir de Données, organisées par Diday & Kodratoff, 8-9 déc. 1987.
- [GENN 89] J.H. GENNARI, P. LANGLEY, D. FISHER
"Models of incremental concept formation",
 Artificial Intelligence, 40, 1989, pp.11-61.
- [GOOD 88] R.M.F. GOODMAN, P. SMYTH
"Information-theoretic rule induction",
 ECAI*, Munich, 1-5 août 1988, pp.357-362.
- [GUEN 89] A. GUENOCHÉ
"Construction du treillis de Galois d'une relation binaire",
 Preprint GRTC-CNRS, Marseille, 1989.

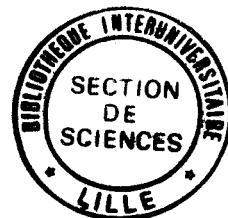
- [HART 84] A. HART
"Experience in the use of an inductive system in knowledge engineering", M. Bramer ed., Research and developments in expert systems, Cambridge : Cambridge University Press.
- [HART 28] R.V.L. HARTLEY
"Transmission of information",
 Bell System Technical Journal, 1928, n°7, p.535.
- [HOTE 33] H. HOTELLING
"Analysis of a complex of statistical variables into principal components", Journal of Educational Psychology, 1933, vol. 24, pp.417-441, pp.498-520.
- [HOTE 36] H. HOTELLING
"Relations between two sets of variables",
 Biometrika, 1936, vol. 28, pp.129-149.
- [HUNT 66] E. HUNT, J. MARIN, P. STONE
"Experiments in induction", New-York, Academic Press, 1966.
- [JAMB 78] M. JAMBU, M.O. LEBEAUX
"Classification automatique pour l'analyse des données",
 tome 1 : Méthodes et algorithmes, Dunod Décision, 1978.
- [KAUF 73] A. KAUFMANN
"Introduction à la théorie des sous-ensembles flous",
 tome 1 : "Eléments théoriques de base", Masson & Cie, 1973.
- [KAUF 78] A. KAUFMANN, G. BOULAYE
"Théorie des treillis en vue des applications",
 Masson, Paris, 1978.
- [KAUF 87] A. KAUFMANN
"Nouvelles logiques pour l'intelligence artificielle",
 Traité des Nouvelles Technologies, série Math. Appliquées,
 Hermes, Paris, 1987.
- [KAYS 90] D. KAYSER
"La représentation des connaissances", Modèles Logiques et Systèmes d'Intelligence Artificielle, Hermes, série Intelligence Artificielle, Paris, 1990, pp.59-71.
- [KLIR 75] G.J. KLIR
"On the representation of activity arrays", International Journal of General Systems, 2(3), 1975, pp.149-168.
- [KLIR 76] G.J. KLIR
"Identification of generative structures in empirical data",
 International Journal of General Systems, 3(2), 1976, pp.89-104.
- [KLIR 77] G.J. KLIR, H.J.J. UYTENHOVE
"On the problem of computer-aided structure identification : some experimental observations and resulting guidelines",
 International Journal of Man-Machine Studies, 9, 1977, pp.593-628.

- [KOD 86] Y. KODRATOFF
"Leçons d'apprentissage symbolique automatique", Techniques Avancées de l'Informatique, Cepadues Editions, 1986.
- [KONO 91] I. KONONENKO, I. BRATKO
"Information-based evaluation criterion for classifier's performance", Machine Learning, 6, 1991, pp.67-80.
- [LEBO 87] M. LEBOWITZ
"Experiments with incremental concept formation : UNIMEM", Machine Learning, 2, 1987, pp.103-138.
- [LERM 70] I.C. LERMAN
"Les bases de la classification automatique", Gauthier-Villars, Coll. Programmation, Paris, 1970.
- [LIQ 89] M. LIQUIERE
"INNE : un algorithme d'apprentissage de structures dans le cas d'exemples bruités",
 4^e JFA, St Malo, 22-24 mai 1989, pp.93-106.
- [LIQ 90] M. LIQUIERE, E. MEPHU NGUIFO
"LEGAL : un système d'apprentissage de concepts à partir d'exemples",
 5^e JFA 90, Lannion, 25-26 avril 1990, pp.93-113.
- [MANT 91] R.L. DE MANTARAS
"A distance-based attribute selection measure for decision tree induction", Machine Learning, 6, 1991, pp.81-92.
- [MARC 87] F. MARCOTORCHINO
"Une approche unifiée des problèmes de sériation par blocs",
 5^e Journées Internationales "Analyse des Données et Informatique", organisées par l'INRIA, 29sept-2oct 1987, Versailles, tome 2.
- [MARC 90] F. MARCOTORCHINO
"Les techniques de sériation : Etat de l'art",
 Colloque International sur "les Méthodes de Blocs Sériation et applications", ENSAIS, Strasbourg, 3-5 avril 1990.
- [MICH 80] R.S. MICHALSKI
"Pattern recognition as rule-guided inductive inference",
 IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-2, n°4, july 1980.
- [MICH 83a] R.S. MICHALSKI
"A theory and methodology of inductive learning",
 Machine Learning : An Artificial Intelligence Approach, R.S. Michalski, J.G. Carbonell, T.M. Mitchell eds, Tioga, Palo Alto, 1983, pp.83-134.
- [MICH 83b] R.S. MICHALSKI, R.E. STEPP
"Learning from observation : conceptual clustering",
 Machine Learning : an Artificial Intelligence Approach, Tioga, Palo Alto, 1983, pp.331-363.

- [MICL 84] L. MICLET
"Méthodes structurales pour la reconnaissance des formes", Eyrolles, Paris, 1984.
- [MILL 63] G.A. MILLER
"What is information measurement", American Psychologist, vol. 8, n°2, 1963, pp.50-51.
- [MING 89a] J. MINGERS
"An empirical comparison of selection measures for decision-tree induction", Machine Learning, 3, 1989, pp.319-342.
- [MING 89b] J. MINGERS
"An empirical comparison of pruning methods for decision tree induction", Machine Learning, 4, 1989, pp.227-243.
- [MITC 82] T.M. MITCHELL
"Generalization as search", Artificial Intelligence, 18, 1982, pp.203-226.
- [MITC 83] T.M. MITCHELL, P.E. UTGOFF, R. BANERJI
"Learning by experimentation, acquiring and refining problem-solving heuristics", Machine Learning : an Artificial Intelligence Approach, R.S. Michalski, J.G. Carbonell, T.M. Mitchell eds, Tioga Publishing Company, 1983, pp.163-190.
- [NORR 78] E.M. NORRIS
"An algorithm for computing the maximal rectangles in a binary relation", Revue Roumaine de Mathématiques Pures et Appliquées, tome XXIII, n°2, Bucarest, 1978, pp.243-250.
- [NYQ 24] H. NYQUIST
"Certain factors affecting telegraph speed", Bell System Technical Journal, vol. 3, 1924, p.324.
- [PARO 91] A. PARODI, P. BONELLI
"Un système de classifieurs efficace et sa comparaison avec deux méthodes représentatives d'apprentissage automatique dans trois domaines médicaux", 6^e JFA, Sète, 15-16 mai 1991.
- [PARZ 60] E. PARZEN
"Modern probability theory and its application", John Wiley and Sons, New-York, 1960.
- [PEAR 01] K. PEARSON
"On lines and planes of closest fit to system of points in space", Phil. Mag., 1901, vol.2, n°11, pp.559-572.
- [PICA 72] C.F. PICARD
"Graphes et questionnaires", tome 1&2, Coll. Programmation, Gauthier-Villars, 1972.
- [POM 90a] D. POMORSKI, M. STAROSWIECKI, M. BARBOUCHA
"Learning a qualitative model of a complex system via data analysis", International Conference : Artificial Intelligence - Industrial Application, Leningrad, 15-19 avril 1990, pp.55-60.

- [POM 90b] D. POMORSKI, M. STAROSWIECKI, M. BARBOUCHA
"Modelling complex systems via the analysis of qualitative data", MIM-s2'90, Bruxelles, 3-7 sept. 1990.
- [POM 91] D. POMORSKI, M. STAROSWIECKI
"Covers and partitions for observation based learning", IMACS* symposium on Modelling and Control of Technological Systems 91, 7-10 mai 1991, Lille, pp.505-510, vol. 2.
- [POST 87] J.G. POSTAIRE
"De l'image à la décision", Dunod Informatique, Paris, 1987.
- [QUEE 67] J.B. Mac QUEEN
"Some methods for classification and analysis of multivariate observations", 5th Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, n°1 Berkeley University of California Press, 1967.
- [QUIN 83] J.R. QUINLAN
"Learning efficient classification procedures and their application to chess and games", Machine Learning : an Artificial Intelligence Approach, R.S. Michalski, J.G. Carbonell, T.M. Mitchell eds, Tioga Publishing Company, 1983, pp.463-482.
- [QUIN 86] J.R. QUINLAN
"Induction of decision trees", Machine Learning Journal, 1, 1986, pp.81-106.
- [QUIN 87] J.R. QUINLAN
"Simplifying decision trees", International Journal of Man-Machine Studies, 27, 1987, pp.221-234.
- [QUIN 88] J.R. QUINLAN
"An empirical comparison of genetic and decision-tree classifiers", 5th International Conference on Machine Learning, Univ. of Michigan, 12-14 juin 1988, pp.135-141.
- [QUIN 84] J. QUINQUETON, J. SALLANTIN
"Généralisation par points de vue et apprentissage de concept", rapport de recherche INRIA, n°265, 1984.
- [QUIN 87] J. QUINQUETON
"Le système d'apprentissage CALM", Journées "Symbolique-Numérique" pour l'Apprentissage de Connaissances à partir de Données, organisées par Diday & Kodratoff, 8-9 déc. 1987.
- [RALA 84] H. RALAMBONDRAINY, E. DIDAY
"Optimisation d'une classification hiérarchique", RAIRO, recherche opérationnelle/operations research, vol. 18, n°3, août 1984, pp.279-295.

- [RALA 87] H. RALAMBONDRAINY
"GENREG un générateur de règles combinant techniques d'apprentissage et techniques d'analyse des données", Journées "Symbolique-Numérique" pour l'Apprentissage de Connaissances à partir de Données, organisées par Diday & Kodratoff, Univ. Paris-Dauphine, 8-9 déc. 1987, pp.40-44.
- [RALA 88] H. RALAMBONDRAINY
"The algorithm GENREG for generating rules from symbolic or numerical data", rapport INRIA-Rocquencourt, n°910, oct.1988.
- [REIT 78] R. REITER
"On closed world databases", Logic and Databases (Gallaire & Minker eds), Plenum Press, 1978, pp.55-76.
- [REND 86] L. RENDELL
"A general framework for induction and a study of selective induction", Machine Learning 1, 1986, pp.177-226.
- [ROUX 85] M. ROUX
"Algorithmes de classification", Masson, Méthode+Programmes, 1985.
- [RUSP 69] E.H. RUSPINI
"A new approach to clustering", Information and Control, vol. 15, n°1, juillet 1969, pp.22-32.
- [RUSP 70] E.H. RUSPINI
"Numerical methods for fuzzy clustering", Information Sciences, vol. 2, 3, juillet 1970, pp.319-350.
- [SAIN 89] Ch. de SAINTE MARIE
"Dépendance perçue - une introduction", 7^e congrès RFIA*, Paris, déc. 1989, pp.1445-1459.
- [SAIN 90] Ch. de SAINTE MARIE
"De l'observation à la connaissance empirique", JFA 90, Lannion, 25-26 avril 1990, pp.251-272.
- [SBAI 83] M. SBAI
"Analyse structurale des systèmes complexes : méthodes d'explication et de partition", Thèse de 3^e cycle, USTLFA, 29 sept. 1983.
- [SEBA 89] M. SEBAG, M. SCHOENAUER
"Iterative learning and redundant generalizations", Data Analysis, Learning Symbolic and Numeric Knowledge, Antibes, INRIA, 11-14 sept. 1989, pp.511-518.
- [SHAN 48] C.E. SHANNON
"A mathematical theory of communication", Bell System Technical Journal, vol. 27, 1948.



- [SHAN 49a] C.E. SHANNON
"Communication in the presence of noise",
 IRE, vol. 37, 1949, p.10.
- [SHAN 49b] C.E. SHANNON, W. WEAVER
"The mathematical theory of communications",
 Urbana, IL : The University of Illinois Press.
- [SPEA 04] C. SPEARMAN
"General intelligence objectively determined and measured",
 American Journal of Psychology, 1904, vol.15, pp.201-292.
- [STAR 84] M. STAROSWIECKI
"Analyse structurale des systèmes complexes",
 RAIRO Automatique, vol. 18, n°2, 1984.
- [STAR 91] M. STAROSWIECKI, M. BARBOUCHA
"Apprentissage automatique par filtrage de tableaux de contingence", Induction Symbolique et Numérique à partir des Données, Diday & Kodratoff eds, Cepadues, Paris, 1991.
- [SZAS 62] G. SZÄSZ
"Introduction to lattice theory",
 Ed. Academic Press, New-York and London, 1962.
- [TIBE 87] G. TIBERGHIE, J.L. ROULIN, A. POLLIÉ
"Apprendre : quoi, quand, comment et pourquoi ?",
 2^e JFA, Chamrousse, 12-13 mars 1987, LIFIA.
- [TORO 82] V.M. TORO CORDOBA
"Contribution à l'analyse structurale de systèmes complexes à l'aide de l'entropie et ses généralisations",
 Thèse de 3^e cycle, USTLFA, 8 mars 1982.
- [UTGO 89] P.E. UTGOFF
"Incremental induction of decision trees",
 Machine Learning, 4, 1989, pp.161-186.
- [VENT 73] H. VENTSEL
"Théorie des probabilités", Editions MIR MOSCOU, 1973.
- [WILL 82] R. WILLE
"Restructuring lattice theory : an approach based on hierarchies of concepts", Symp. Ordered Sets (I. Rival ed.),
 Reidel, Dordrecht-Boston, 1982, pp.445-470.
- [WILS 87] S.W. WILSON
"Quasi-Darwinian learning in a classifier system",
 4th International Workshop on Machine Learning, Univ. of California, Irvine, 22-25 juin 1987, pp.59-65.
- [ZADE 65] L.A. ZADEH
"Fuzzy sets",
 Information and Control, vol. 8, 1965, pp.338-353.