De l'induction confirmatoire à la classification : contribution à l'apprentissage automatique

Nicolas Lachiche

Département de formation doctorale en informatique
UFR STMIA

École doctorale IAE + M

De l'induction confirmatoire à la classification : contribution à l'apprentissage automatique

THÈSE

présentée et soutenue publiquement le jeudi 30 octobre 1997

pour l'obtention du

Doctorat de l'université Henri Poincaré – Nancy 1

(spécialité informatique)

par

Nicolas Lachiche

Composition du jury

Président :	Monique Grandbastien, Professeur, université Henri Poincaré, Nancy I
Rapporteurs :	Christel Vrain, Maître de conférences, université d'Orléans Djamel Abdelkader Zighed, Professeur, université de Lyon II
Examinateurs :	Michèle Sebag, Chargée de Recherche CNRS, école polytechnique, Palaiseau Jean-Paul Haton, Professeur, université Henri Poincaré, Nancy I Pierre Marquis, Maître de conférences, université d'Artois, Lens

Centre de Recherche en Informatique de Nancy
INRIA–Lorraine

Remerciements

Je remercie Monique Grandbastien, Christel Vrain et Djamel Abdelkader Zighed de m'avoir fait l'honneur d'accepter d'être les rapporteurs de ce travail.

Michèle Sebag m'a encouragé à éclaircir les relations entre la classification par portée et l'espace des versions disjonctif. Je l'en remercie beaucoup. La classification par portée a déjà pu bénéficier de développements proposés dans le cadre de l'espace des versions disjonctif, j'espère humblement que la réciproque sera vraie.

Je remercie sincèrement Jean-Paul Haton de la confiance qu'il m'a accordée, en m'accueillant dans l'équipe RFIA. Jean-Paul m'a aidé à confronter les modèles et les réalités de l'apprentissage automatique. J'espère avoir réussi ce pari d'appliquer des modèles logiques aux données réelles, et m'être montré digne de sa confiance.

Pierre Marquis m'a accompagné tout au long de ce travail. À Nancy, ou à Lens, Pierre m'a toujours apporté son aide, ses précieux conseils et ses encouragements. Avec un souci constant de rigueur, Pierre m'a aidé à éclaircir les fonctionnements de l'induction confirmatoire et de la classification à base de règles. Je l'en remercie chaleureusement.

Que toutes les personnes qui m'ont apporté leur aide ou leur soutien sachent que je ne les oublient pas, et je tiens à leur dédier, ainsi qu'à vous, cher lecteur, les pages qui suivent.

Table des matières

Table des figures

Avant-propos

Dans toute cette thèse, le terme « classification » est utilisé au sens de l'apprentissage supervisé.

Introduction

Ces premiers paragraphes ont pour but de situer le contexte de l'apprentissage automatique et d'en présenter l'intérêt. L'intelligence artificielle est l'étude des comportements dits intelligents en vue de les simuler. L'apprentissage automatique est un volet important de cette étude. L'apprentissage automatique consiste à modéliser les facultés d'adaptation et d'acquisition de connaissances afin de les automatiser. L'objectif poursuivi est, d'une part, de mieux comprendre les raisonnements humains, et d'autre part, de venir en aide à la réalisation de systèmes informatiques à base de connaissances. Ces systèmes, tels que les systèmes experts, reposent sur une mise en œuvre automatique de connaissances relatives à un domaine particulier, par exemple un système de diagnostic de pannes dans une centrale nucléaire. En amont des problèmes de modélisation du raisonnement que pose la « manipulation » de connaissances est le problème de l'acquisition de ces connaissances. Les obstacles majeurs qui surgissent sont multiples. Tout d'abord, la masse des connaissances à enregistrer est souvent importante, si bien qu'une entrée manuelle est fastidieuse et coûteuse. Par ailleurs, les connaissances ne sont pas toujours explicites : on dispose souvent d'une multitude d'exemples concrets, mais pas de règles modélisant de façon concise le domaine. L'apprentissage automatique aide donc à la conception des systèmes à base de connaissances en prenant en charge la « manipulation » de grands volumes de données et l'extraction de connaissances explicites à partir des données. L'apprentissage automatique ne sert pas uniquement dans la phase de conception des systèmes à base de connaissances. Il intervient également en cours d'utilisation du système. En effet, les systèmes à base de connaissances doivent être capables d'évoluer au cours de leur « vie ». L'ensemble des connaissances est presque toujours incomplet au moment de la conception du système. Cela peut provenir de ce qu'il est impossible de prévoir toutes les données dont va avoir besoin le système. Par exemple, un robot se déplaçant dans une usine pour réparer une fuite de produits toxiques ne connaît pas à l'avance l'emplacement des machines ou des caisses de matériel au moment de l'évacuation de l'usine. Un système de surveillance de l'état de santé d'une personne doit s'adapter au patient car les données médicales ne sont pas les mêmes pour tous. Un tel système doit donc être à même d'intégrer de nouvelles données sur ce qui est normal et ce qui ne l'est pas pour une personne donnée afin de détecter les cas anormaux. Un apprentissage en cours d'utilisation présente les mêmes difficultés que l'apprentissage initial, la taille des données et leur forme implicite. Il pose un problème supplémentaire : il n'est pas toujours possible de faire modifier le système en cours d'utilisation, parce qu'il est inaccessible et doit s'adapter rapidement à son environnement (c'est le cas par exemple d'un robot d'intervention), ou parce qu'il est diffusé à grande échelle et qu'il est impossible de l'adapter manuellement au cas par cas.

Il faut donc automatiser l'apprentissage, et pour cela le modéliser. Or, il n'existe pas qu'une façon d'apprendre. Ainsi on n'apprend pas des poèmes comme on apprend à nager. Apprendre à reconnaître un chêne ou un sapin ne met pas en œuvre le même apprentissage que lorsque l'on apprend que la couleur orange s'obtient en mélangeant du rouge et du jaune. Il existe donc

plusieurs types d'apprentissage. Apprendre des numéros de téléphones est un apprentissage par cœur. Il ne requiert que des capacités de mémorisation de la part du système. Apprendre à nager est plus complexe. L'apprentissage du mouvement fait l'objet de travaux, par exemple [Sebban et Zighed, 1996]. Dans cette thèse, nous nous sommes intéressés uniquement à la modélisation du raisonnement lié à l'apprentissage. Ainsi il est possible d'apprendre à prédire la couleur obtenue en mélangeant deux couleurs à l'aide de quelques règles d'inférence : *du bleu et du jaune donne du vert, une couleur foncée donne une couleur plus claire quand elle est mélangée avec du blanc.* Il s'agit alors d'un apprentissage déductif. Reconnaître un sapin parmi des arbres requiert une extrapolation : il faut passer de la connaissance de quelques exemples à une règle générale qui permet de reconnaître un sapin même s'il ne fait pas partie de ceux déjà vus.

Nos travaux concernent la généralisation inductive. Nous proposons des modèles d'apprentissage à partir d'observations et d'apprentissage à partir d'exemples. L'apprentissage des régularités que vérifient des observations permet de déterminer des lois générales du domaine considéré. Supposons que le domaine considéré soit celui des animaux domestiques. Un certain nombre de connaissances sont fournies au système : *les animaux qui volent doivent être mis dans des cages complètement fermées, les chats ne doivent pas être mis avec les canaris.* Les connaissances sont entrées sous leur forme la plus simple. Elles ne mentionnent que les caractéristiques les plus significatives : toutes les propriétés ne sont pas mentionnées pour chaque animal. Les relations entre les propriétés doivent donc être données explicitement. Or, il est difficile de ne pas en oublier. Par exemple, il ne faut pas omettre, dans le cas présent, de préciser que *les canaris volent.* Une solution pour pallier les oublis consiste à faire apprendre ces relations à partir d'observations. Et donc à partir de l'observation d'un canari, *Titi*, et de sa capacité à voler, le système peut apprendre que *les canaris volent* et veiller à les mettre dans des cages complètement fermées. L'apprentissage de lois du domaine peut résoudre deux problèmes classiques en planification : le problème du décor et le problème de la ramification. Prenons l'action de *déplacer la boîte de stylos.* Le problème du décor est de décrire tout ce qui ne change pas quand une action est effectuée, par exemple le bureau ne bouge pas quand la boîte de stylos est déplacée. Le problème de la ramification est le problème complémentaire : décrire tout ce qui change quand une action est effectuée, par exemple tous les stylos contenus dans la boîte sont déplacés. Ces problèmes sont habituellement résolus en supposant que seuls les faits modifiés par l'action sont explicités, ceux qui ne sont pas mentionnés ne changent pas. Comme il est fastidieux d'indiquer toutes les modifications entraînées par une action, seules les effets caractéristiques de l'action sont mentionnés, *la boîte de stylos est déplacée.* Les autres effets sont déduits à partir de lois du domaine. Il faut donc être sûr de ne pas en oublier. Et il peut être utile d'apprendre à partir de quelques observations avant et après l'action de *déplacer une boîte* que tout ce qu'elle contient est déplacé au même endroit. Un type particulier d'observations est celui d'exemples de concepts. Par exemple, le concept *peut porter des lentilles de contact* peut être appris à partir d'exemples d'individus pouvant porter des lentilles de contact et d'individus ne pouvant pas le faire. Le problème qui consiste à déterminer si une personne donnée peut porter des lentilles de contact à partir de la connaissance d'exemples et de contre-exemples est un problème d'apprentissage à partir d'exemples appelé problème de classification.

Les travaux que nous présentons dans cette thèse concernent l'apprentissage de lois générales à partir d'observations, ce que nous appelons induction, et le cas particulier de la classification où les observations sont des exemples d'individus appartenant à des classes et où

les lois induites permettent de prédire la classe d'un nouvel individu. Le modèle de généralisation confirmatoire que nous proposons est fondé sur une notion de confirmation représentant l'hypothèse des similarités, c'est-à-dire le fait que les individus inconnus se comportent comme les individus connus. Ce modèle permet de pallier certaines difficultés rencontrées par les approches existantes dans le cadre d'un langage de la logique des prédicats du premier ordre. C'est d'un point de vue purement logique que nous explicitons les liens existant entre ce modèle de généralisation, spécialisé à un langage attribut-valeur, et les classifieurs à base de règles minimales cohérentes et pertinentes. Nous proposons une technique de classification à base d'instances qui, d'un point de vue logique, permet de classer tout objet comme un classifieur à base de règles minimales cohérentes et pertinentes le ferait, mais sans engendrer explicitement de telles règles. Cette technique de classification fournit les mêmes classifieurs que la classification par l'espace des versions disjonctif décrit par Michèle Sebag [Sebag, 1994; Sebag, 1996]. La classification par portée offre donc un nouvel éclairage sur cette approche. Elle contribue, en particulier, à mettre en relation les classifieurs à base de règles, à base d'espace des versions et à base d'instances. Tout comme l'espace des versions disjonctif, la classification par portée s'appuie sur l'ensemble des hypothèses cohérentes et pertinentes pour un ensemble d'exemples. La classification par portée bénéficie de plus d'un algorithme plus efficace que les approches « équivalentes ». Par ailleurs, elle offre des perspectives spécifiques de part son approche à base d'instances et sa caractérisation en terme de règles. Le modèle de généralisation confirmatoire proposé et la classification par portée ne sont fondés que sur des considérations logiques. Les biais numériques présents dans de nombreuses approches de façon plus ou moins imbriquée avec des biais logiques, comme par exemple dans les arbres de décision, sont dans notre cadre clairement séparés. Néanmoins, un raisonnement inductif sur des données réelles ne peut pas être modélisé de façon satisfaisante dans une approche purement logique. Pour pallier ce problème, deux paramètres ont été utilisés dans l'espace des versions disjonctif. Le paramètre de généralité, ou inversement de spécificité, admet deux interprétations différentes suivant que l'on le considère du point de vue d'un objet à classer à l'aide d'une règle, ou du point de vue d'une hypothèse dans l'espace des versions disjonctif. La classification par portée offre en moyenne une meilleure précision de classification et une rapidité de mise en œuvre du même ordre que celles des approches usuelles telles que C4.5 [Quinlan, 1993a], CN2 [Clark et Boswell, 1991] et PEBLS [Cost et Salzberg, 1993].

Cette thèse se compose de deux parties : *Induction confirmatoire* et *Classification*. Présentons brièvement chacune de ces parties.

La première partie, *Induction confirmatoire*, est consacrée à la modélisation de la généralisation confirmatoire, c'est-à-dire au problème de l'inférence de lois générales reflétant des régularités présentes dans les observations. Elle est composée de trois chapitres :

Le chapitre *Induction et généralisation* présente l'induction, ses caractéristiques logiques et deux de ses formes : l'induction explicative et l'induction confirmatoire. Nous exposons également dans ce chapitre les relations qu'entretiennent ces deux formes d'induction avec la déduction.

Le chapitre *Généralisation confirmatoire en logique des prédicats* est consacré à la définition d'un tel modèle. Nous rappelons la représentation des connaissances dans ce cadre, avant de présenter la problématique de la généralisation confirmatoire. Nous définissons tout d'abord des postulats de rationalité pour ce type de raisonnement avant de proposer un modèle de généralisation qui les respecte. Nous discutons ensuite des différences entre les relations de confirmation définies par Carl Hempel et celle que nous avons proposée. Nous explicitons les motivations logiques des restrictions appliquées aux langages de représentation, puis nous consi-

dérons l'influence de la représentation et de la relation de confirmation sur plusieurs paradoxes de l'induction.

Dans le chapitre *Généralisation confirmatoire dans un cadre attribut-valeur*, nous montrons comment le modèle précédent se spécialise à un cadre attribut-valeur. Nous rappelons pour commencer la représentation des connaissances utilisée dans ce cadre, puis nous montrons ses rapports avec la représentation en logique des prédicats. Nous définissons alors des relations de confirmation et de généralisation confirmatoire ; ces relations spécialisent au cadre attribut-valeur les relations correspondantes exposées au chapitre précédent. Nous présentons ensuite une technique de calcul des généralisations, les arbres d'énumération d'ensembles.

Dans la seconde partie, *Classification*, nous nous intéressons à l'inférence d'un type particulier de lois générales : celles qui permettent d'associer une classe à une description à partir d'exemples d'associations. Cette partie comporte sept chapitres :

Le chapitre *Apprentissage à partir d'exemples* est consacré à la présentation de cette problématique. Nous y exposons diverses techniques de l'apprentissage automatique, puis nous explicitons les biais, ou critères de préférence, mis en œuvre dans toute technique d'apprentissage inductif. Nous rappelons enfin le problème du non-déterminisme de l'association entre une description et sa classe, fréquent lorsque des données réelles sont prises en compte.

Dans le chapitre *Classification à partir de règles minimales cohérentes et pertinentes*, nous étudions cette famille particulière de classifieurs. Nous rappelons tout d'abord ce que sont les règles minimales cohérentes et pertinentes pour un ensemble d'apprentissage. Nous exposons les propriétés principales de ces règles, avant d'étudier leur relation avec la généralisation confirmatoire. Nous montrons que la représentation des exemples influe sur la construction des règles. Nous examinons en particulier comment une théorie du domaine peut être prise en compte. Nous expliquons enfin pourquoi les règles minimales cohérentes et pertinentes pour un ensemble d'apprentissage ne sont pas toujours ses généralisations confirmatoires.

Le chapitre *Des règles aux instances : la classification par portée* présente la classification par portée. Nous définissons pour commencer ce qu'est la portée d'un exemple, avant d'étudier comment passer des règles aux instances et donc obtenir une approche à base d'instances ayant la sémantique d'un classifieur à base de toutes les règles cohérentes et pertinentes pour un ensemble d'instances. Nous présentons un algorithme de classification par portée fondé sur le principe « diviser pour régner », puis les critères de résolution que nous avons retenus. Nous soulignons la séparation effectuée entre les biais logique et statistiques et le pouvoir explicatif du classifieur obtenu. Nous considérons les relations de la classification par portée avec ses plus proches voisins : les approches fondées sur les distances, les approches à base d'instances généralisées en règles, et surtout l'espace des versions disjonctif.

Le chapitre *Extensions de la classification par portée* contient plusieurs extensions de la technique de classification présentée au chapitre précédent. Les premières extensions concernent l'adaptation de ce modèle aux données réelles et la modulation de la cohérence. Nous considérons ensuite une généralisation de la classification par portée aux règles, puis deux extensions faciles à réaliser mais d'un grand intérêt pratique. La première concerne la prise en compte d'exemples redondants et la seconde celle d'exemples négatifs.

Le chapitre *Les plus-proches-voisins de la classification par portée* présentent les travaux semblables à classification par portée. Nous rappelons les fondements statistiques des approches à base d'instances avant de montrer les relations entre les distances classiques utilisées dans ces approches et la notion de pré-ordre utilisée dans la classification par portée. Nous caractérisons ensuite les différences entre les travaux existants combinant apprentissage à base d'instances et à base de règles et les stratégies de généralisation des exemples en règles que nous avons présentées

dans le chapitre précédent. Nous rappelons enfin le lien fort existant entre la classification par portée et les techniques fondées uniquement sur les mêmes biais logiques. Nous détaillons en particulier le cas de l'espace des versions disjonctif et montrons que les classifications produites par la classification par portée sont équivalentes dans certaines conditions, mais que leurs approches différentes conduisent à des perspectives différentes.

Une *Évaluation expérimentale* fait l'objet du chapitre suivant. Nous y présentons la plate-forme d'évaluation développée à cet effet, puis nous procédons à une évaluation globale des performances en précision et en temps de la classification par portée comparées à celles d'autres approches. Nous considérons ensuite quelques applications à des domaines réels issus de problématiques de l'équipe RFIA, ou traités par Ron Rymon.

Le chapitre *Bilan et perspectives de la classification par portée* contient un bilan des qualités et défauts de la classification par portée suivant les trois facteurs clés : précision, rapidité et compréhensibilité. Il contient également des amorces de réflexions sur la réduction de l'ensemble d'apprentissage, sur la possibilité d'accorder plus d'importance à certains ensembles d'attributs qu'à d'autres, sur l'utilisation d'une théorie du domaine dans la classification par portée, et enfin sur la nécessité de l'induction constructive et d'un contrôle de la pertinence des attributs.

Une présentation de la logique des prédicats du premier ordre et de la logique propositionnelle, ainsi que la description détaillée du contenu et des résultats de l'évaluation expérimentale sur des bases de test usuelles de l'apprentissage automatique sont reportées en annexe.

Première partie

Induction confirmatoire

Chapitre 1

Induction et généralisation

Ce chapitre est centré sur la notion de généralisation inductive. Nous présentons tout d'abord la notion d'induction et ses principales caractéristiques logiques. Nous précisons ensuite le but de notre travail : déterminer les raisons logiques conduisant à accepter une hypothèse comme étant la généralisation d'un ensemble d'observations. En particulier, nous montrons qu'une hypothèse peut être acceptée parce qu'elle explique ou parce qu'elle est confirmée par les observations, et que les deux techniques d'induction résultantes sont distinctes. Nous indiquons les liens existant entre ces deux formes d'induction et la déduction logique et définissons la généralisation.

1.1 Introduction

Intuitivement, induire, c'est dériver une loi générale à partir de quelques exemples de celle-ci. Par exemple, à partir du nombre fini de chiens que nous avons pu voir, nous avons inféré que tous les chiens ont quatre pattes. C'est aussi par induction que nous avons appris que lorsque l'on pose la main sur la porte du four, on se brûle. Nous avons pu inférer une loi générale à partir d'exemples. Induire consiste donc à considérer pour certaines des lois générales inférées à partir d'un sous-ensemble des cas où elles s'appliquent.

Plus formellement, un problème d'induction est la donnée d'un ensemble d'observations, ou d'exemples, noté E. Une théorie du domaine, notée Th, composée d'un ensemble de propriétés déjà connues, peut, éventuellement, être utilisée. À partir de ces connaissances, l'induction vise à produire des hypothèses H qui sont des propriétés des observations que l'on souhaite générales, c'est-à-dire qui applicables à d'autres cas que ceux connus.

L'induction est le raisonnement utilisé pour la découverte de théories dans les sciences expérimentales : il s'agit de trouver des lois générales expliquant un nombre fini d'expériences et, bien sûr, de vérifier que ces lois sont satisfaites, à l'aide d'expériences destinées à les confirmer, ou à les invalider. De telles lois ne proviennent pas de déductions, et il est la plupart du temps impossible de garantir leur validité. Les résultats de l'induction sont des hypothèses plutôt que des conclusions.

La problématique de l'induction n'est pas apparue avec l'intelligence artificielle. Elle a été étudiée depuis longtemps. Ainsi, Aristote en proposait une définition logique dans un cadre syllogistique. Si *Titi vole* et que nous savons que *Titi est un oiseau*, nous pouvons supposer que *Tous les oiseaux volent*. L'induction a été étudiée par de nombreux philosophes, parmi lesquels nous pouvons citer Descartes, Kant, Hume, Goodman, Hempel, Popper, ... Malgré les travaux de ces philosophes, l'induction reste difficile à automatiser car elle peut prendre

plusieurs formes. Elle possède néanmoins quelques grandes caractéristiques logiques.

1.2 Quelques caractéristiques logiques de l'inférence inductive

La première caractéristique du raisonnement inductif qui le rend difficile à automatiser est qu'il n'est pas *valide*. Son résultat n'est pas une conséquence logique des connaissances disponibles au départ. Alors que tous les oiseaux que nous avons pu voir volent, l'hypothèse *les oiseaux volent* n'est pourtant pas valide et pourrait facilement être réfutée par la première autruche venue !

Cette possibilité d'erreur vient de ce que l'induction est *ampliative*. Elle produit des connaissances nouvelles. Cela lui permet d'aller au delà des connaissances établies, mais pour cela il faut accepter le risque de se tromper. L'induction est donc *non monotone* : l'apport de nouvelles informations peut contredire des généralisations effectuées et obliger à reprendre le raisonnement.

Les relations d'inférence inductive sont, en général, pluri-extensionnelles : plusieurs conclusions incompatibles entre elles sont souvent possibles. Par exemple, si l'on veut prédire le chiffre suivant dans la suite de nombres : 2 3 5, plusieurs hypothèses sont possibles. En effet, puisque les trois premiers nombres sont les trois premiers nombres premiers, la suite peut être 2 3 5 7. On peut aussi remarquer que la différence entre 3 et 2 vaut 1, que celle entre 5 et 3 vaut 2, et que le nombre suivant est donc égal à 5+3, une autre solution possible est donc 2 3 5 8. Le nombre 1000 est également une conclusion possible car 2, 3, 5 et 1000 sont les racines du polynôme $(x-2)(x-3)(x-5)(x-1000)$.

L'induction *mathématique*, en particulier la preuve par récurrence, est un type de raisonnement inductif bien connu, mais qui ne permet que d'expliciter des connaissances qui peuvent être dérivées des connaissances initiales car c'est un raisonnement valide. L'ensemble des individus pour lesquels sont inférées des lois respecte une propriété qui garantit la validité des conclusions obtenues. Les propriétés sur les entiers naturels peuvent ainsi être prouvées par récurrence car l'ensemble des entiers naturels est lui même défini par induction (mathématique). L'induction mathématique n'est pas facile à réaliser, mais elle est une forme de déduction, elle ne prend pas le risque de se tromper comme l'induction considérée en intelligence artificielle.

En conclusion, le raisonnement inductif est en général ampliatif, et donc non valide et non monotone. De plus, les hypothèses inférées peuvent être contradictoires entre elles. Ces quelques caractéristiques ne suffisent pas à définir l' induction et ne permettent pas à elles seules d'automatiser la construction des hypothèses. D'ailleurs, s'agit-il vraiment de construire des hypothèses ?

1.3 Construction ou acceptation d'hypothèses ?

Jusqu'ici, nous avons présenté l'induction comme un processus de construction d'hypothèses rendant compte de quelques observations. Ce point de vue n'est pas partagé par tout le monde. Charles Sanders Pierce, par exemple, considère que l'induction part d'une théorie, en déduit des prédictions de phénomènes, et observe ces phénomènes pour évaluer à quel point ils « valident » la théorie. On peut distinguer deux étapes dans l'induction. Une première étape consiste en la construction d'une loi générale à partir d'exemples. La seconde est d'utiliser cette loi au delà des exemples sur lesquels elle a été établie. Pour notre part, nous nous intéressons essentiellement à l'étape de construction d'hypothèse. Cette étape peut, à son tour, être appréhendée de plusieurs façons. Les hypothèses peuvent résulter d'un processus constructif dans

lequel elles sont « inférées ». Un tel mécanisme de construction n'existe malheureusement pas dans le cas général. Le seul palliatif consiste à générer des hypothèses candidates et à tester si elles constituent des hypothèses inductives. Ce type d'algorithme est appelé *generate-and-test* en anglais. Notre tâche se réduit donc essentiellement dans le cas général à déterminer un critère permettant de reconnaître une hypothèse inductive. Dans des cas plus restreints, par exemple dans le cas de langages propositionnels, il est cependant possible de « construire directement » des hypothèses inductives selon plusieurs critères.

1.4 Principes inductifs

Toute connaissance nouvelle pour un ensemble d'observations ne peut être considérée comme une généralisation de cet ensemble. Inférer que tous les sapins sont verts à partir de l'observation de corbeaux noirs n'a *a priori* aucun sens. Il faut donc déterminer des *critères de préférence* permettant de choisir une généralisation parmi plusieurs. Ces critères de préférence, également appelés *biais*, permettent également de guider la recherche. Les généralisations doivent être de nature universelle, et pour cela apporter de l'information par rapport à celle dont on disposait. Les hypothèses doivent être compatibles avec les connaissances initiales. Elles doivent également être liées à ces connaissances, en particulier rendre compte des observations.

Une généralisation est simplement une formule qui peut être induite à partir des observations disponibles mais pas déduite à partir de celles-ci.

Définition 1.1 *Une* généralisation inductive H *est une hypothèse inférée inductivement qui n'est pas une conséquence logique de l'ensemble d'observations* E*, formellement* $E \not\models H$.

Un problème d'induction est défini formellement de la façon suivante :

Définition 1.2 *Étant donnés*

- *un ensemble d'observations* E *dans un langage* L,
- *une théorie du domaine* Th *dans* L,
- *un ensemble de critères de préférence* C *caractérisant les hypothèses acceptables (ou préférées) en intension dans le métalangage associé à* L,

un problème d'induction consiste à déterminer l'ensemble des hypothèses H *dans* L *qui rendent compte de* E *étant donné* Th *et sont préférées par* C.

Dans cette définition, « rendre compte » peut prendre plusieurs sens : il peut s'agir d'expliquer les observations ou alors de refléter les régularités apparaissant dans celles-ci. Dans ce dernier cas, l'hypothèse doit être vérifiée par l'ensemble des exemples, on dit que les exemples la confirment. Ainsi, on distingue deux formes d'induction, suivant le sens que l'on donne à rendre compte : l'induction explicative et l'induction confirmatoire [Muggleton et DeRaedt, 1994; Flach, 1995].

1.4.1 Induction explicative

Étant donnée une théorie du domaine Th, l'*induction explicative* vise à déterminer les hypothèses H qui expliquent, de façon déductive, les rapports d'observations E, c'est-à-dire les hypothèses H telles que dans tous les cas où H et Th sont vrais, E est vrai. Formellement :

Définition 1.3 *Étant donnés*

- *un ensemble d'observations* E *dans un langage* L,

- *une théorie du domaine Th dans L,*

- *un ensemble de critères de préférence C,*

un problème d'induction explicative consiste à déterminer étant donnés Th et C l'ensemble des hypothèses H dans L telles que :

- $H \wedge Th \models E$,

- *H est préférée grâce à C.*

Considérons par exemple la théorie du domaine :
$Th =_d \{oiseau(Titi), vole(Superman), chat(Sylvestre), \forall X, chat(X) \Rightarrow \neg vole(X)\}$

l'hypothèse $H =_d \forall X, oiseau(X) \Rightarrow vole(X)$ est une hypothèse inductive explicative de $E =_d vole(Titi)$.

Les critères de préférence les plus souvent retenus sont la cohérence (Th et H doivent être compatibles) et la spécificité (H doit être aussi spécifique que possible). Signalons que malheureusement la notion de généralisation la plus spécifique, si elle existe toujours dans un cadre propositionnel [Marquis, 1989] ou lorsque l'on restreint la déduction logique à une de ses restrictions décidables (la subsomption) [Plotkin, 1970b; Plotkin, 1970a], n'existe pas nécessairement au premier ordre même lorsque Th est vide [Niblett, 1988].

Ce modèle d'induction est celui qui a été le plus considéré dans les travaux réalisés par les chercheurs en intelligence artificielle jusqu'à présent, en particulier dans la communauté de programmation logique inductive. Les philosophes ont, pour leur part, davantage prêté attention à une seconde forme d'induction, l'induction confirmatoire.

1.4.2 Induction confirmatoire

Le principe de l'*induction confirmatoire* est qu'une hypothèse inductive doit refléter les régularités présentées par les observations dont on dispose, c'est-à-dire être confirmée par celles-ci.

Définition 1.4 *Étant donnés*

- *un ensemble d'observations E dans un langage L,*

- *une théorie du domaine Th dans L,*

- *un ensemble de critères de préférence C,*

un problème d'induction confirmatoire consiste à déterminer étant donnés Th et C l'ensemble des hypothèses H telles que :

- $Comp_{HS}(E \wedge Th) \models H$,

- *H est préférée grâce à C.*

$Comp_{HS}(E \wedge Th)$ représente la complétion des connaissances données au départ grâce à une hypothèse des similarités HS.

Par exemple, si tout ce que l'on sait est que Titi est un oiseau, Titi vole, Superman vole, Sylvestre est un chat et que les chats ne sont pas des oiseaux alors si l'on suppose que les individus inconnus se comportent comme les individus connus, c'est-à-dire tous les individus vérifient les propriétés vérifiées par Titi, Superman et Sylvestre, on peut supposer que Toto vole lorsque l'on sait que Toto est un oiseau. Plus formellement,

$$H =_d \forall X(oiseau(X) \Rightarrow vole(X))$$

peut être considérée comme une hypothèse inductive confirmatoire de

$$E =_d \{oiseau(Titi), vole(Titi), vole(Superman), chat(Sylvestre), \forall X(chat(X) \Rightarrow \neg oiseau(X))\}.$$

Bien que H ne soit pas une conséquence logique de E, elle est satisfaite (on dit aussi confirmée) par tous les individus connus.

Le principe d'induction utilisé, appelé l'*hypothèse des similarités*, est que *les individus inconnus sont semblables aux individus connus, i.e. ils partagent les mêmes propriétés.* Ce principe représenté par l'ensemble C des critères de préférence admet plusieurs interprétations. Nous y reviendrons.

L'induction confirmatoire ne nécessite pas de séparer les connaissances générales sur le domaine (*les chats ne sont pas des oiseaux*) des observations factuelles (*Titi est un oiseau, Titi vole, Superman vole* et *Sylvestre est un chat*). Toutes les informations, quels que soient leur type, sont considérées comme faisant partie de ce que l'on sait, et sont intégrées dans le rapport d'observation E.

L'induction confirmatoire diffère bel et bien de l'induction explicative. Ainsi, une hypothèse confirmée par les observations ne les explique pas forcément pour autant. E n'est pas une conséquence logique de H, donc H ne constitue pas une explication de E (au moins au sens déductif de l'explication) : en effet, supposer que tous les oiseaux volent n'explique pas le fait que *Titi est un oiseau.*

1.5 Relations de l'induction avec la déduction

L'induction entretient des liens étroits avec la déduction. En effet, les hypothèses induites peuvent souvent être dérivées déductivement à partir des observations et du principe inductif choisi.

L'induction explicative a un lien évident avec la déduction. Les hypothèses qu'elle vise à déterminer doivent permettre d'expliquer les observations, au sens où les observations doivent être des conséquences logiques des hypothèses. L'induction explicative correspond donc à une « inversion de la déduction ». Lorsqu'observations et théorie du domaine sont représentées par des formules fermées, on a (suivant le théorème de la déduction) :

$$H \wedge Th \models E \text{ si et seulement si } \neg E \wedge Th \models \neg H.$$

À une négation près, les généralisations peuvent donc être produites par déduction. C'est, par exemple, le principe de fonctionnement du système PROGOL [Muggleton, 1995]. La partie risquée de l'induction explicative, ce qui fait que ce n'est pas de la déduction, consiste à ne pas garder toutes les généralisations.

L'induction confirmatoire est, par définition, une déduction à partir d'une théorie complétée, comme l'expriment Stephen Muggleton et Luc De Raedt [Muggleton et DeRaedt, 1994] :

> *[L'induction confirmatoire] réalise l'induction par déduction. Le principe inductif de [l'induction confirmatoire] établit que l'hypothèse H, qui est, en un sens, déduite*

de l'ensemble des [individus] observés E et de la théorie du domaine Th (en utilisant [des techniques de complétion]), est vraie pour tous les [individus] possibles.

L'induction confirmatoire peut ainsi être vue comme une déduction dans une théorie complétée par l'hypothèse de similarité. De nombreuses formes d'inférence non monotone peuvent également se réduire à des déductions à partir d'une théorie complétée. On peut donc établir des liens avec plusieurs formalismes d'inférence non monotone : l'hypothèse du monde clos [Reiter, 1978] et ses généralisations [Minker, 1982], la complétion de Clark [Clark, 1978] [Shepherdson, 1988], ou quelques formes de circonscription [McCarthy, 1980] [McCarthy, 1986]. Cependant, comme nous le verrons dans le chapitre suivant, l'inférence inductive ne se réduit pas à n'importe quelle forme d'inférence non monotone. La confusion existe dans les modèles de [Helft, 1989; Marquis, 1992; De Raedt et Bruynooghe, 1993; De Raedt et Dzeroski, 1994] et elle pose problème.

En conclusion, la déduction joue un rôle majeur dans les deux formes d'induction, explicative et confirmatoire. C'est d'ailleurs également le cas dans la grande majorité des raisonnements non déductifs. Cependant, le fait d'induire en déduisant, c'est-à-dire d'utiliser la mécanique de la déduction pour induire, ne facilite pas la compréhension du processus inductif.

1.6 Généralisation inductive et généralité

Un grand nombre d'hypothèses peuvent être induites à partir d'un ensemble d'observations. Parmi ces hypothèses, les hypothèses les plus générales, au sens de la conséquence logique, paraissent les plus intéressantes. En effet, la connaissance des lois les plus générales permet d'obtenir par déduction l'ensemble des hypothèses induites à partir des observations. Il reste à définir ce que l'on entend par « plus générale ». Cette question a suscité un débat intense dans la communauté apprentissage (cf. par exemple [Kodratoff et Ganascia, 1986]) et plusieurs réponses sont clairement admissibles. Dans notre cadre, c'est la notion de généralité entre théories logiques qui prévaut. Ainsi :

Définition 1.5 *Une hypothèse H_1 est plus générale qu'une hypothèse H_2 si et seulement si H_2 est une conséquence logique de H_1, formellement $H_1 \models H_2$.*

La généralisation inductive est cependant la proie d'un dilemme entre généralité et spécificité. D'un côté, plus une généralisation est spécifique, plus le risque de se tromper diminue. À la limite, quand on n'apprend rien, on est sûr de ne pas se tromper. D'un autre côté, plus l'hypothèse est générale, plus elle est utile car mieux elle décrit le domaine considéré : *tous les sapins sont des arbres* apporte plus d'information que *tous les sapins des Vosges sont des arbres*. Ce problème est particulièrement aigu dans le cas de l'induction explicative. Toute généralisation d'une explication est une explication (monotonie de la déduction logique). Il faut donc choisir quand s'arrêter. Ce problème ne se pose pas dans le cas de la généralisation confirmatoire. En effet, les hypothèses doivent être confirmées par les observations. Généraliser une hypothèse confirmée ne conduit pas toujours à une hypothèse confirmée. La généralisation fondée sur l'induction confirmatoire retient les hypothèses les plus générales confirmées par les observations. La généralisation confirmatoire est donc fondée sur deux notions bien distinctes : confirmation et généralité. Alors que toute généralisation confirmatoire doit être confirmée par les observations, toute hypothèse confirmée par les observations n'est pas une généralisation confirmatoire. Par exemple, *Titi est un oiseau* est confirmée par *Titi est un oiseau*, mais n'en est pas une généralisation.

1.7 Résumé et conclusion

Dans ce chapitre, nous avons présenté l'induction, ses grandes caractéristiques, ses formes principales et leurs relations avec la déduction. L'induction est un raisonnement ampliatif, donc non valide et non monotone. Un grand nombre d'hypothèses, parfois incompatibles, peuvent être le résultat d'une induction. Dans tous les cas, on demande qu'une hypothèse rende compte des observations. Deux sens peuvent être donnés à rendre compte : ils définissent respectivement l'induction explicative et l'induction confirmatoire, l'hypothèse expliquant les observations ou l'hypothèse reflétant une régularité confirmée par les observations. La première forme peut être mise en œuvre en « inversant » la déduction logique alors que la seconde forme consiste en une déduction dans une base de connaissances complétée par l'hypothèse des similarités : les individus inconnus se comportent comme les individus connus. Parmi les hypothèses inductives, les hypothèses générales (ou généralisations) sont intéressantes car elles s'appliquent à plus d'individus.

Chapitre 2

Généralisation confirmatoire en logique des prédicats

Différents langages logiques peuvent être utilisés pour représenter les connaissances. Le langage des prédicats du premier ordre est le plus expressif des langages que nous considérons dans la suite. Il permet par exemple d'exprimer des relations entre individus. Ce chapitre est consacré à la présentation d'un modèle de généralisation fondé sur l'induction confirmatoire dans le cadre de connaissances représentées dans un langage de la logique des prédicats. Nous présentons tout d'abord la représentation choisie. Nous définissons ensuite la généralisation confirmatoire. Nous mettons l'accent sur les faiblesses des approches existantes et la marche à suivre pour les éviter. Nous exposons des postulats de rationalité pour la généralisation confirmatoire fondée sur l'hypothèse des similarités selon Hempel avant de proposer un modèle de généralisation les satisfaisant.

2.1 Représentation des connaissances

Dans ce chapitre, les connaissances sont représentées dans un langage de la logique des prédicats du premier ordre sans symboles de fonctions autres que les constantes. La logique des prédicats est présentée en annexe A.1. Une constante représente un individu connu, une variable représente un individu quelconque. Les propriétés des individus sont représentées par les prédicats. Ainsi $vole(Titi)$ indique que Titi sait voler. $\forall X, vole(X)$ indique que tous les individus savent voler. L'ensemble des constantes contenues dans les observations E désignent l'ensemble des individus connus. La formule logique E désigne les propriétés connues de ces individus.

La logique des prédicats est strictement plus expressive que les logiques propositionnelles. Elle permet en particulier d'exprimer des relations entre les individus. Par exemple, *si X est le père de Y alors Y est l'enfant X* s'écrit $\forall X, \forall Y$, $père(X, Y) \Rightarrow enfant(Y, X)$.

2.2 Généralisation confirmatoire

La généralisation confirmatoire est une généralisation inductive fondée sur la confirmation. L'induction confirmatoire peut être vue comme une déduction dans une base de connaissances complétée (cf. paragraphe 1.5). Nous montrons dans ce paragraphe que le principe de complétion nécessaire n'est malheureusement pas celui mis en œuvre dans les approches existantes.

Nous insistons sur la nécessité de séparer les niveaux rationnel et inférentiel lors de la définition d'un modèle de généralisation confirmatoire et également sur la séparation à observer entre confirmation et généralisation.

2.2.1 Hypothèse des similarités

Intuitivement, l'hypothèse des similarités exprime le fait que les individus inconnus se comportent comme les individus connus. Une des difficultés est de capturer formellement cette hypothèse (qui est d'ailleurs une méta-hypothèse). Plusieurs interprétations sont possibles. Nous en étudions deux dans la suite :

Hypothèse des similarités selon Hempel Une hypothèse inductive H est une conséquence logique du rapport d'observation E quand l'ensemble des individus est réduit à l'ensemble des individus de E.

Hypothèse des similarités selon Nicod Une hypothèse inductive est une implication $G \Rightarrow D$ cohérente avec le rapport d'observation E et satisfaite par un des individus i de E, c'est-à-dire i vérifie G et D.

Les hypothèses inductives associées à ces deux interprétations de l'hypothèse des similarités diffèrent. Étant donné le rapport d'observation $E =_d \{vole(Titi), \neg oiseau(Sylvestre)\}$, la formule $H =_d \forall X, oiseau(X) \Rightarrow vole(X)$ est une hypothèse inductive selon Hempel car les deux individus connus, $Titi$ et $Sylvestre$, satisfont H. La formule H n'est pas une hypothèse inductive selon Nicod car aucun des individus connus ne satisfait $oiseau(X)$ et $vole(X)$. Réciproquement, étant donné le rapport d'observation $E =_d \{oiseau(Titi), vole(Titi), chat(Sylvestre)\}$, la formule $H =_d \forall X, oiseau(X) \Rightarrow vole(X)$ est une hypothèse inductive selon Nicod, mais pas selon Hempel car l'individu $Sylvestre$ n'est pas connu comme vérifiant $\neg oiseau(Sylvestre)$ ou $vole(Sylvestre)$.

Dans les deux cas, l'hypothèse des similarités exprime le fait que les individus inconnus partagent les mêmes propriétés que les individus connus. Lorsque l'on assimile un individu aux propriétés qu'il satisfait, l'hypothèse des similarités exprime le fait que les individus inconnus ont les mêmes descriptions que les individus connus, donc ne peuvent être séparés de ceux-ci. Par exemple, si l'on sait que *Titi est un oiseau* et *Sylvestre est un chat*, la loi *tout individu est un chat ou un oiseau* est une hypothèse inductive (selon Hempel). Tout nouvel individu, par exemple *Superman*, doit sous l'hypothèse des similarités *être un chat ou un oiseau*. Dans le premier cas, *Superman* et *Sylvestre* partagent exactement les mêmes propriétés. Dans le second cas, *Superman* et *Titi* partagent exactement les mêmes propriétés.

La majorité des travaux existant en intelligence artificielle proposent des modèles de généralisation confirmatoire qui semblent fondés sur l'hypothèse des similarités selon Hempel. Il est toutefois bien difficile d'attribuer à chaque modèle proposé une des deux interprétations de l'hypothèse des similarités. En effet, l'interprétation retenue n'est, en général, jamais présentée explicitement dans les descriptions de ces modèles.

2.2.2 Les approches existantes

La plupart des approches existantes de la généralisation confirmatoire [Helft, 1989; Marquis, 1992; De Raedt et Dzeroski, 1994] s'insèrent dans un cadre appelé *sémantique non monotone globale* [Muggleton et DeRaedt, 1994]. Dans ce cadre, les généralisations confirmatoires sont les

formules les plus générales qui sont vraies dans le modèle de Herbrand minimal des observations $\mathcal{M}^+(E)$.

Définition 2.1 *Étant donné E sous la forme d'un ensemble de clauses* définies, *une hypothèse H est une clause qui est confirmée par E si et seulement si*

Validité : $\mathcal{M}^+(E) \models H$

Complétude : *H est complète, c'est-à-dire $\forall H'$, si $\mathcal{M}^+(E) \models H'$ alors $H \models H'$*

Minimalité : *H est minimale, c'est-à-dire aucun sous-ensemble strict de H, $H' \subset H$, n'est à la fois valide et complet.*

Dans cette approche, E est interprétée comme son modèle de Herbrand minimal. Le premier modèle de généralisation confirmatoire a été introduit par Nicolas Helft [Helft, 1988; Helft, 1989]. Dans son approche, Nicolas Helft ne considère pas que des clauses définies, mais des clauses à portée restreinte, les *g-clauses*, qui ne contiennent pas de symboles de fonction autres que les constantes, et dans lesquelles toute variable apparaissant dans un littéral positif apparaît également dans un littéral négatif. L'approche de Nicolas Helft repose sur le mécanisme de sous-implication proposé par Geneviève Bossu et Pierre Siegel [Bossu et Siegel, 1985]. Ces derniers ont en particulier montré qu'un ensemble de g-clauses possède un nombre fini de modèles minimaux et que ces modèles sont finis. Nicolas Helt s'appuie donc sur ces modèles minimaux pour définir les généralisations fortes, qui sont des clauses satisfaites par tous les modèles minimaux des observations, et les généralisations faibles, qui ne sont satisfaites que par au moins un modèle minimal des observations.

Pierre Marquis a proposé une extension de l'approche de Nicolas Helft dans laquelle il ne considère que des généralisations sous forme de clauses de Horn et des observations sous la forme d'atomes terminaux, mais où il autorise des symboles de fonction autres que les constantes [Marquis, 1992]. Luc de Raedt et Saso Dzeroski [De Raedt et Dzeroski, 1994] se placent exactement dans le cadre du modèle de Herbrand minimal présenté plus haut.

Un autre mécanisme provenant des logiques non monotones a été utilisé dans un système de généralisation confirmatoire : c'est la complétion de prédicats [Clark, 1978]. Cette complétion a été utilisée par Luc de Raedt et Maurice Bruynooghe dans leur système CLAUDIEN [De Raedt et Bruynooghe, 1993] :
Une clause H est confirmée si et seulement si elle est une conséquence logique du rapport d'observation complété à l'aide de la complétion de prédicat de Clark, $Comp_{Clark}(E) \models H$.
Le rapport d'observation est un ensemble de clauses et les généralisations sont des clauses à portée restreinte, sans symboles de fonction autres que les constantes, et dans lesquelles toute variable apparaissant dans un littéral positif apparaît également dans un littéral négatif.

Notons que l'utilisations de mécanismes provenant des logiques non monotones semble naturel puisque, comme nous l'avons dit, l'induction confirmatoire peut être vue comme une forme de déduction dans une théorie complétée et que justement ces mécanismes opèrent une complétion de la théorie. Cette complétion implique, dans le cas des modèles minimaux aussi bien que dans le cas de la complétion de prédicats, que les faits qui ne peuvent pas être déduits de E sont supposés faux : c'est l'hypothèse du monde clos. Nous allons montrer qu'une telle complétion n'est pas toujours appropriée à la formalisation de la confirmation.

2.2.3 Circonscription des individus ou de leurs propriétés?

Les approches existantes de la généralisation confirmatoire [Helft, 1989; Marquis, 1992; De Raedt et Bruynooghe, 1993; De Raedt et Dzeroski, 1994] utilisent des principes de complé-

tion plus forts que celui requis par l'induction. En fait, celui-ci est supporté par les modèles proposés, mais les contraintes imposées ne sont pas minimales. L'hypothèse des similarités selon Hempel requiert seulement de circonscrire l'univers aux *individus connus*. Les principes de complétion utilisés dans la sémantique non monotone globale consistent à circonscrire les *propriétés* de ces individus. Les approches existantes utilisent donc des techniques de complétion trop fortes. Cela peut entraîner la production de généralisations indésirables ou interdire l'obtention de généralisations intuitivement plus intéressantes.

Rappelons les techniques de complétion utilisées par les approches existantes. Le modèle de Nicolas Helft [Helft, 1989] est basé sur la sous-implication [Bossu et Siegel, 1985]. Pierre Marquis, Luc De Raedt et Saso Dzeroski [Marquis, 1992; De Raedt et Dzeroski, 1994] utilisent des modèles de Herbrand minimaux (la minimalité consiste à privilégier les modèles contenant un ensemble minimal de faits positifs pour l'inclusion ensembliste). Luc De Raedt et Maurice Bruynooghe [De Raedt et Bruynooghe, 1993] ont recours à la complétion de Clark. Toutes ces techniques circonscrivent des propriétés des individus. En particulier, elles privilégient l'information positive par rapport à l'information négative : si E est un ensemble de clauses définies, tout fait terminal positif qui ne peut être déduit de la base de connaissances est supposé faux. Cette complétion trop forte conduit à des généralisations indésirables : $E =_d \{chat(Sylvestre),\ oiseau(Titi)\}$ ne confirme pas $G =_d \forall X, \neg chat(X) \vee \neg oiseau(X)$ à moins de compléter E explicitement avec $\{\neg chat(Titi), \neg oiseau(Sylvestre)\}$, à l'aide de l'hypothèse du monde clos. Il est tout à fait possible de compléter la base de connaissances à l'aide de l'hypothèse du monde clos si cela est prévu dans le choix de la représentation (par exemple, à titre d'économie de représentation), mais il est important de noter que cette complétion n'a rien à voir avec l'induction.

Les techniques de complétion de l'inférence non monotone utilisées jusqu'ici pour généraliser présentent un deuxième inconvénient. La complétion d'une base de connaissances cohérente peut être incohérente [Thayse *et al.*, 1989; Grégoire, 1990]. En effet, compléter $E =_d \{vole(Titi), \forall X, oiseau(X) \vee chat(X)\}$ à l'aide de l'hypothèse du monde clos consiste à ajouter les formules $\neg oiseau(Titi)$ et $\neg chat(Titi)$. La conjonction de ces deux formules contredit $\forall X, oiseau(X) \vee chat(X)\}$. La complétion de prédicats de Clark peut de même conduire à des incohérences. Elle est, de plus, sensible à la syntaxe. La complétion de Clark de $\forall X, \neg animal(X) \Rightarrow animal(X)$ est la formule contradictoire $\forall X, \neg animal(X) \Leftrightarrow animal(X)$. La formule $\forall X, \neg animal(X) \Rightarrow animal(X)$ est pourtant équivalente à la formule $\forall X, animal(X)$, dont la complétion est elle-même. Ces problèmes sont résolus en restreignant les langages de représentation des observations et des généralisations (ce qui introduit un biais de langage, cf. paragraphe 4.3). Malheureusement, utiliser de telles restrictions peut conduire à manquer des généralisations intéressantes. Par exemple, aucune des approches précédentes ne considère $\forall X, chat(X) \vee oiseau(X)$ comme une généralisation de $chat(Sylvestre) \vee oiseau(Titi)$.

2.2.4 Niveaux rationnel et inférentiel

Pour éviter les problèmes précédents, il nous parait nécessaire de séparer le niveau cognitif de l'induction – qu'est une généralisation ? – et le niveau calculatoire – comment calculer les généralisations ? Avant de proposer une technique de généralisation confirmatoire, il est important de préciser les propriétés qu'elle doit vérifier. Des caractéristiques des approches existantes proviennent de la notion même de généralisation (au niveau cognitif), alors que d'autres (restrictions des langages de représentation) ne sont requises que par la machinerie utilisée (au niveau calculatoire). Les postulats de rationalité doivent être indépendants du mécanisme d'inférence mis en œuvre. Il faut ensuite vérifier que le mécanisme d'inférence

satisfait ces postulats.

La confusion entre les niveaux cognitif et inférentiel dans les approches existantes rend difficile à cerner la notion de confirmation utilisée dans celles-ci. Il faut au contraire que la confirmation apparaisse clairement de façon à pouvoir la séparer de la généralisation proprement-dite. Il est donc préférable de respecter cette séparation au niveau inférentiel.

Dans la suite, nous nous intéressons plus particulièrement aux modèles de généralisation confirmatoire fondés sur l'hypothèse des similarités selon Hempel. Nous définissons un tel modèle par un quintuplet $\langle L_E, L_H, L_G, \mathrel{|\!\sim}, \mathrel{|\!\approx} \rangle$. L_E est le langage de représentation des observations, ces dernières incluant éventuellement une théorie du domaine, L_H est le langage de représentation des propriétés confirmées et L_G celui des généralisations. $\mathrel{|\!\sim}$ est la relation de confirmation entre L_E et L_H, et $\mathrel{|\!\approx}$ la relation de généralisation entre L_E et L_G.

Bien entendu, il est souhaitable qu'un modèle de généralisation confirmatoire puisse être utilisé en pratique pour calculer des généralisations. Il est donc parfois nécessaire de renoncer à un modèle par trop « idéal » et de se satisfaire d'un modèle ne vérifiant pas tous les postulats mis en avant. La recherche du meilleur équilibre entre expressivité et efficacité est l'une des principales quêtes en intelligence artificielle [Brachman et Levesque, 1985].

2.3 Postulats de rationalité

Dans ce paragraphe, nous présentons les postulats de rationalité pour la confirmation établis par Carl Hempel et ceux pour l'induction confirmatoire établis par Peter Flach. Nous donnons ensuite des axiomes pour la généralisation confirmatoire. Nous listons les postulats de rationalité pour la confirmation qui sont compatibles avec ces axiomes et sont donc aussi des postulats de rationalité de la généralisation confirmatoire.

2.3.1 Postulats de rationalité pour la confirmation et pour l'induction confirmatoire

Carl Hempel structure les postulats de rationalité pour la confirmation en quatre groupes de conditions d'adéquation logique [Hempel, 1943; Hempel, 1945]. « *Dans chaque groupe, la première condition implique les autres. Ces dernières sont mentionnées pour deux raisons ; premièrement, parce que la plupart représentent des caractéristiques importantes qui seraient en général recherchées dans un concept de confirmation défini de façon adéquate; et deuxièmement, parce que des définitions alternatives [de la confirmation] en apparence raisonnables [...], se révèlent satisfaire certaines de ces conditions plus faibles, mais non la condition la plus forte* » :

Condition d'implication (H1). Une proposition impliquée par les observations est confirmée par celles-ci : si $E \models H$ alors $E \mathrel{|\!\sim} H$.

Réflexivité (H1.1). Un rapport d'observation est confirmé par lui-même : $E \mathrel{|\!\sim} E$.

Condition de conséquence (H2). Si un rapport d'observation confirme un ensemble K de formules alors il confirme n'importe quelle formule qui est une conséquence logique de K :
si $E \mathrel{|\!\sim} K$ et $K \models H$ alors $E \mathrel{|\!\sim} H$.

Condition de conséquence spéciale (H2.1). Si un rapport d'observation confirme une hypothèse H alors il confirme également toutes les conséquences de H :
si $E \mathrel{|\!\sim} H$ et $H \models F$ alors $E \mathrel{|\!\sim} F$.

Condition d'équivalence (H2.2). Si un rapport d'observation confirme une hypothèse H alors il confirme aussi toute hypothèse logiquement équivalente à H :
si $E \mid\!\sim H$ et $H \equiv F$ alors $E \mid\!\sim F$.

Condition de conjonction (H2.3). Si un rapport d'observation confirme deux hypothèses alors il confirme aussi leur conjonction :
si $E \mid\!\sim H$ et $E \mid\!\sim F$ alors $E \mid\!\sim H \wedge F$.

Condition de cohérence (H3). Tout rapport d'observation cohérent est compatible logiquement avec l'ensemble de toutes les hypothèses qu'il confirme :
si $E \not\models \bot$ et $E \mid\!\sim H$ alors $E \wedge H \not\models \bot$.

Condition de cohérence spéciale (H3.1). A moins d'être lui-même contradictoire, un rapport d'observation ne confirme pas d'hypothèse avec laquelle il n'est pas cohérent logiquement :
si $E \not\models \bot$ et $E \wedge H \models \bot$ alors $E \not\mid\!\sim H$.

Condition de cohérence des hypothèses (H3.2). A moins d'être lui-même contradictoire, un rapport d'observation ne confirme pas d'hypothèses qui se contredisent mutuellement :
si $E \not\models \bot$ et $F \wedge H \models \bot$ alors $E \not\mid\!\sim F \wedge H$.

Condition d'équivalence pour les observations (H4). Si un rapport d'observation E confirme une hypothèse H alors tout rapport d'observation logiquement équivalent à E confirme aussi H :
si $E \mid\!\sim H$ et $E \equiv F$ alors $F \mid\!\sim H$.

Peter Flach a proposé un ensemble de postulats de rationalité que doit vérifier tout raisonnement inductif [Flach, 1995; Flach, 1996] :

Principe de vérification (I1). Si E induit une hypothèse H et si F est une conséquence logique de E et H, c'est-à-dire F est une prédiction, alors H rend compte de l'observation de E et F :
si $E \mid\!\sim H$ et $E \wedge H \models F$ alors $E \wedge F \mid\!\sim H$.

Principe de falsification (I2). Si E induit une hypothèse H et si F est une conséquence logique de E et H, c'est-à-dire F est une prédiction, alors H ne rend pas compte de l'observation de E et $\neg F$:
si $E \mid\!\sim H$ et $E \wedge H \models F$ alors $E \wedge \neg F \not\mid\!\sim H$.
De ce principe de falsification et de l'équivalence logique à gauche postulée plus bas, on peut déduire le principe suivant (**I2'**) : si H implique la négation de E alors E n'induit pas H :
si $\models H \Rightarrow \neg E$ alors $E \not\mid\!\sim H$.

Extension à droite (I3). Lorsqu'une observation E induit une hypothèse H, les connaissances dont on dispose sont alors la conjonction de E et H. Si F est une conséquence logique de $E \wedge H$ alors H et F rendent compte de E :
si $E \mid\!\sim H$ et $\models E \wedge H \Rightarrow F$ alors $E \mid\!\sim H \wedge F$.

Réflexivité affaiblie (I4) et (I5). Le principe (I2') a pour conséquence que la réflexivité ne peut être définie que pour des formules cohérentes. On sait grâce à (I2') qu'une formule intervenant dans une relation inductive est cohérente, donc si E rend compte de H alors

H rend compte de H et E rend compte de E :
si $E \mid\!\sim H$ alors $E \mid\!\sim E$ et $H \mid\!\sim H$.

Équivalence logique à droite (I6) et à gauche (I7). La validité d'une induction ne dépend que du contenu logique des prémisses et de la conclusion, et non de leur expression syntaxique. Les mêmes hypothèses rendent compte de deux observations équivalentes. C'est la condition (H2.2) de Hempel :
si $E \mid\!\sim H$ et $\models H \Leftrightarrow F$ alors $E \mid\!\sim F$.
Si une hypothèse H rend compte d'une observation E, toute hypothèse équivalente F rend compte également de cette observation. C'est la condition (H4) de Hempel :
si $E \mid\!\sim H$ et $\models E \Leftrightarrow F$ alors $F \mid\!\sim H$.

Monotonie prudente (I8). La monotonie prudente établit que si H et F rendent compte de E alors l'ajout de l'une à E admet toujours l'autre comme hypothèse inductive :
si $E \mid\!\sim H$ et $E \mid\!\sim F$ alors $E \wedge F \mid\!\sim H$.

Les postulats (I1, I2, I2′, I3, I4, I5, I6, I7, I8) doivent être vérifiés par toute relation d'inférence inductive. Ils sont donc valables pour l'induction explicative et l'induction confirmatoire. Peter Flach a également répertorié des postulats de rationalité qui ne s'appliquent qu'à l'induction confirmatoire. Ces postulats utilisent la notion d'admissibilité d'une formule.

Définition 2.2 *Étant donnée une relation de conséquence inductive* $\mid\!\sim$, *une formule* E *est* admissible *si et seulement si* $E \mid\!\sim E$. *Elle est dite* inadmissible *sinon.*

Les postulats de rationalité propres à l'induction confirmatoire sont les suivants :

Affaiblissement à droite (C1). Si E confirme H alors E confirme toute conséquence logique de H. C'est la condition (H2.1) de Hempel :
si $E \mid\!\sim H$ et $\models H \Rightarrow F$ alors $E \mid\!\sim F$.

Réflexivité confirmatoire (C2). La réflexivité confirmatoire établit que si E est cohérente et E ne confirme pas la négation de H alors H confirme H :
si $E \mid\!\sim E$ et $E \not\mid\!\sim \neg H$ alors $H \mid\!\sim H$.
Ce principe est plus facile à comprendre lorsque l'on considère sa contraposée :
si $E \mid\!\sim E$ et $H \not\mid\!\sim H$ alors $E \mid\!\sim \neg H$.
Si H est inadmissible, c'est-à-dire si c'est une formule trop forte au regard de la théorie du domaine, sa négation $\neg H$ est si faible qu'elle est confirmée par n'importe quelle formule admissible.

Introduction du ET à droite (C3). La conjonction de deux hypothèses confirmées est elle-même confirmée. C'est la condition (H2.3) de Hempel :
Si $E \mid\!\sim H$ et $E \mid\!\sim F$ alors $E \mid\!\sim H \wedge F$.

Introduction du OU à gauche (C4). Si E et F confirment H alors leur disjonction confirme H :
si $E \mid\!\sim H$ et $F \mid\!\sim H$ alors $E \vee F \mid\!\sim H$.

Introduction du ET à gauche (C5). Si E et F confirment H alors leur conjonction confirme H :
si $E \mid\!\sim H$ et $F \mid\!\sim H$ alors $E \wedge F \mid\!\sim H$.

Un certain nombre des postulats de rationalité répertoriés par Flach reprennent des conditions établies par Hempel. Ce sont principalement toutes les versions faibles de la condition de conséquence. (H2.1) est reprise en (C1), (H2.2) en (I6) et (H2.3) en (C3). La condition d'équivalence logique à gauche apparaît également en (H4) et en (I7).

Parmi les postulats de rationalité précédents, nous ne retenons pas l'introduction du OU à gauche (C4) comme propriété devant être satisfaite par les modèles de généralisation fondés sur l'hypothèse des similarités selon Hempel. En effet, la disjonction de deux formules $A \vee B$ indique que l'une ou l'autre est vraie. Cela signifie que les propriétés des individus mentionnés dans A sont vraies ou que les propriétés des individus mentionnés dans B sont vraies. Ainsi la formule $vole(Superman) \vee vole(Titi)$ signifie que Titi ou Superman vole. Il y a deux individus $Titi$ et $Superman$ (pas forcément distincts) et au moins l'un des deux sait voler. Cette connaissance ne garantit pas que les deux savent voler. Donc bien que $\forall X, vole(X)$ soit confirmée par $vole(Superman)$ et par $vole(Titi)$, elle n'est pas confirmée par $vole(Superman) \vee vole(Titi)$. L'introduction du OU à gauche ne serait acceptable que si le sens donné à $vole(Superman) \vee vole(Titi)$ était soit on dispose de la connaissance $vole(Superman)$ et on ne sait rien de $Titi$, on ignore même son existence, soit on dispose de la connaissance $vole(Titi)$ et on ne sait rien de $Superman$, on ignore même son existence, soit on dispose de la connaissance $vole(Superman)$ et $vole(Titi)$. Cette sémantique n'est pas celle de la disjonction logique. L'introduction du OU logique à gauche n'est donc pas une propriété que doit vérifier une relation de confirmation $\mid\!\sim$ dans un modèle de généralisation confirmatoire fondé sur l'hypothèse des similarités selon Hempel.

2.3.2 Postulats de rationalité pour la généralisation confirmatoire

Comme indiqué au paragraphe 1.6, seules certaines hypothèses confirmées peuvent être considérées comme des généralisations. Une relation de généralisation ne satisfait donc pas nécessairement tous les postulats cités précédemment. Après avoir défini des axiomes pour la généralisation confirmatoire, nous examinons les incompatibilités existant entre ces axiomes et les postulats de rationalité pour la confirmation.

L'induction confirmatoire consiste à caractériser les formules confirmées par les observations. La généralisation inductive vise à caractériser les lois générales qui rendent compte des observations. Les axiomes que nous proposons pour caractériser les relations $\mid\!\approx$ de généralisation confirmatoire fondées sur l'hypothèse des similarités selon Hempel sont les suivants :

Fondement confirmatoire. Si G est une généralisation d'un ensemble d'observations E alors G est confirmé par E :
si $E \mid\!\approx G$ alors $E \mid\!\sim G$.

Raisonnement ampliatif. Si G est une conséquence logique de E alors ce n'est pas une de ses généralisations :
si $E \models G$ alors $E \not\mid\!\approx G$.

Généralité. Une généralisation est une formule la plus générale satisfaisant les deux conditions précédentes :
si $E \mid\!\approx G_1$, $E \mid\!\approx G_2$ et $G_1 \models G_2$ alors $G_1 \equiv G_2$.

Le fondement confirmatoire assure que toute généralisation confirmatoire est une hypothèse confirmée. Les deux axiomes suivants permettent de discriminer les généralisations parmi toutes les hypothèses confirmées. Une généralisation doit apporter plus d'information que les connaissances initiales (raisonnement ampliatif). Parmi les hypothèses candidates, seules les

formules les plus générales doivent être retenues. Elles apportent en effet plus d'information sur le domaine considéré que les hypothèses moins générales.

On peut se demander si l'axiome d'ampliativité n'est pas redondant avec l'axiome de généralité. En effet, considérant l'observation $p(a)$, l'hypothèse $p(a)$ est bien confirmée, mais n'est pas une généralisation puisqu'elle est impliquée par la formule $\forall X, p(X)$, elle-même confirmée. Donc la condition d'ampliativité ne semble pas nécessaire pour éviter d'obtenir une conséquence logique triviale du rapport d'observation. Considérons à présent $E = p(a) \wedge \forall X, p(X)$ alors l'hypothèse la plus générale confirmée est $\forall X, p(X)$, mais c'est une conséquence logique de E. Finalement, seul l'axiome d'ampliativité permet d'éviter de retrouver une conséquence logique des observations.

Les axiomes de raisonnement ampliatif et de généralité restreignent l'ensemble des hypothèses acceptables comme généralisations parmi les hypothèses confirmées. Accepter ces axiomes de la généralisation oblige à rejeter certains postulats pour la généralisation confirmatoire. En particulier, l'axiome de raisonnement ampliatif est contradictoire avec la condition d'implication de Hempel et toutes ses conséquences: la condition d'implication (H1) et les conditions de réflexivité (H1.1,I4,I5,C2) sont donc exclues. L'axiome de généralité de son côté s'oppose à la condition de conséquence de Hempel. Il impose de ne pas retenir les postulats d'affaiblissement à droite (H2.1 = C1), d'extension à droite (I3), ainsi que l'introduction du ET à droite (H2.3 = C3).

Les postulats de rationalité pour la confirmation satisfaits par la généralisation confirmatoire sont donc les postulats restants: les principes de vérification (I1) et de falsification (I2), la condition de cohérence (H3), l'équivalence logique à droite (H2.2 = I6), l'équivalence logique à gauche (H4 = I7), et la monotonie prudente (I8).

Ces postulats ne définissent pas un modèle unique pour la généralisation confirmatoire fondée sur l'hypothèse des similarités selon Hempel. Alors que nous considérons que tout modèle de généralisation confirmatoire rationnel doit vérifier ces postulats, nous n'avons pas examiné la complétude de leur ensemble: nous ne garantissons pas que ce sont les seuls. Cependant, tout modèle violant l'un d'entre eux ne constitue pas, à notre avis, un modèle de généralisation confirmatoire (fondée sur l'hypothèse des similarités selon Hempel) acceptable.

2.4 Un modèle de généralisation confirmatoire

Dans ce paragraphe, nous proposons un modèle de généralisation confirmatoire fondé sur l'hypothèse des similarités selon Hempel.

2.4.1 Confirmation

L'hypothèse des similarités selon Hempel traduit le fait qu'une hypothèse inductive est une conséquence logique du rapport d'observation si l'ensemble des individus est restreint aux individus connus. L'hypothèse de fermeture du domaine permet donc de la capturer au mieux.

Définition 2.3 *Étant donné un ensemble fini d'individus* $C =_d \{c_1, \ldots, c_n\}$, *on appelle* hypothèse de fermeture du domaine FD_C *pour* C *la formule* $\forall X, (X = c_1) \vee \ldots \vee (X = c_n)$.

La définition de l'hypothèse de fermeture du domaine fait intervenir le prédicat d'égalité qui ne peut pas être interprété librement mais doit satisfaire les axiomes de l'égalité. Rappelons que les *axiomes de l'égalité (AE)* sont:

Réflexivité $\forall X, (X = X)$

Symétrie $\forall X, \forall Y, (X = Y) \Rightarrow (Y = X)$

Transitivité $\forall X, \forall Y, \forall Z, ((X = Y) \wedge (Y = Z)) \Rightarrow (X = Z)$

Substitutivité Pour tout prédicat P d'arité n, on a: $\forall X_1, \ldots, \forall X_n, \forall Y_1, \ldots, \forall Y_n$,
$((X_1 = Y_1) \wedge \ldots \wedge (X_n = Y_n)) \Rightarrow (P(X_1, \ldots, X_n) \Leftrightarrow P(Y_1, \ldots, Y_n))$.

Sur cette base, nous pouvons maintenant définir la relation de confirmation utilisée dans notre modèle.

Définition 2.4 *Le langage de représentation des observations L_E est l'ensemble des formules universelles de la logique des prédicats du premier ordre sans symboles de fonctions autres que les constantes et sans égalité. L'ensemble d'individus C de chaque rapport d'observation doit être fini et non vide.*

Le langage de représentation des hypothèses confirmées L_H est la restriction aux clauses du langage de représentation des observations L_E, sans la contrainte pour chaque formule d'inclure un ensemble d'individus fini et non vide.

E confirme H étant donné l'ensemble fini d'individus C (représentés par des constantes du langage) si et seulement si H est une conséquence logique de E lorsque l'Univers est réduit à l'ensemble des individus de C. Formellement, $E \mid\sim_C H$ si et seulement si $FD_C \wedge AE \wedge E \models H$.

Rappelons que l'hypothèse de fermeture du domaine est un principe de complétion implicitement présent dans les approches fondées sur la sémantique non monotone globale de l'induction (C est alors le domaine de Herbrand de E). Comme nous l'avons évoqué, ces approches incluent en outre d'autres principes de complétion (circonscription des propriétés) qui ne sont pas forcément désirables.

La relation de confirmation que nous utilisons dans notre modèle est la relation de confirmation directe de Hempel. Carl Hempel définit la confirmation à partir de cette relation de confirmation directe en l'étendant à l'aide de la conséquence logique [Hempel, 1943].

Définition 2.5 *Un rapport d'observation E confirme H suivant la relation de confirmation étendue de Hempel si H est la conséquence logique d'un ensemble de formules directement confirmées par E ou impliquées par E.*

Carl Hempel a étendu la confirmation directe à l'aide de la conséquence logique afin de satisfaire aux conditions d'implication et de conséquence. Nous avons vu que ces conditions s'opposent aux axiomes de raisonnement ampliatif et de généralité de la généralisation confirmatoire (cf. paragraphe 2.3). Nous avons donc utilisé une notion de confirmation directe puisqu'elle se révèle mieux adaptée à notre objectif.

Pour pouvoir utiliser la définition 2.4, il reste à définir l'ensemble C des individus (du moins leur représentation) connus. Pour ce faire, on peut distinguer deux approches :

- C est le domaine de Herbrand $D_H(E)$ de E. Le domaine de Herbrand de $E =_d oiseau(Titi) \wedge (oiseau(Titi) \vee chat(Sylvestre))$ est $D_H(E) =_d \{Titi, Sylvestre\}$.

- C est l'ensemble des *individus essentiels* $I(E)$ de E. Un individu i essentiel de E est une constante i de E telle que toute formule E' logiquement équivalente à E contient au moins une occurrence de i. Étant donné le rapport d'observation précédent $E =_d oiseau(Titi) \wedge (oiseau(Titi) \vee chat(Sylvestre))$, $Titi$ est un individu essentiel, alors que $Sylvestre$ ne l'est pas (Sylvestre n'apparaît pas dans la formule $E' =_d oiseau(Titi)$ logiquement équivalente à E).

La première approche conduit à une relation de confirmation $\mid\!\sim_{D_H(E)}$ qui ne satisfait pas l'équivalence logique à gauche. C'est pourquoi Hempel a introduit la notion d'individu essentiel qui fonde la seconde approche. Nous pensons néanmoins que la première approche mérite d'être retenue. Elle traduit en effet un principe d'existence selon lequel tout individu représenté existe. Ce principe est largement satisfait dans les bases de connaissances existantes. La première approche satisfait, en fait, une forme d'équivalence à gauche qui n'est pas l'équivalence logique à gauche, mais l'équivalence logique à gauche à domaines de Herbrand identiques.

Au niveau inférentiel, une seconde raison pour accepter la première approche réside dans la complexité du calcul des individus essentiels de E. Ainsi, lorsque E est une formule terminale. déterminer si une constante i est essentielle pour E est un problème NP-difficile.

Propriété 2.1 *Soit ESSENTIEL le problème de décision suivant:*

- *Étant donnés une formule terminale E d'un langage L de la logique des prédicats du premier ordre sans symboles de fonction autres que les constantes et une constante i,*
- *Déterminer si i est essentiel pour E.*

ESSENTIEL est NP-difficile.

Preuve : Soit ϕ l'application de L_{prop} dans $L_{préd}$ telle que pour tout E de L_{prop}, on a $\phi(E) =_d E \wedge p(i)$ où p est un prédicat du langage $L_{préd}$ et i une constante de $L_{préd}$. E étant propositionnel, ni p, ni i n'apparaissent dans E. Clairement $\phi(E)$ peut être calculé en temps polynômial dans $|E|$.

On a l'équivalence : E est satisfaisable si et seulement si i est essentiel dans $\phi(E)$. En effet, comme ni p, ni i n'apparaissent dans E, la seule possibilité pour supprimer $p(i)$ de $\phi(E)$ tout en préservant l'équivalence logique est que E soit contradictoire. ϕ est une réduction polynômiale de SAT (le problème de la satisfaisabilité d'une formule propositionnelle) vers ESSENTIEL. Par conséquent, ESSENTIEL est NP-difficile. #

Comme nous le verrons au paragraphe 2.4.3, décider si une hypothèse H est confirmée par E pour $\mid\!\sim$ lorsque E est terminale nécessite la résolution d'une instance d'un problème co-NP complet, dont l'entrée peut être de taille exponentielle dans l'arité du prédicat d'arité maximale occurant dans H. Cela ne signifie pas pour autant que le coût en temps de l'élimination des individus non-essentiels soit négligeable dans tous les cas.

2.4.2 Généralisation

Une fois la relation de confirmation définie, nous pouvons maintenant présenter la relation de généralisation retenue dans notre modèle.

Définition 2.6 *Le langage de représentation des généralisations L_G est le même langage que celui des hypothèses confirmées L_H, c'est-à-dire un langage de clauses de la logique des prédicats du premier ordre sans symboles de fonctions autres que les constantes et sans égalité.*

Les généralisations sont définies simplement en suivant les axiomes de la généralisation confirmatoire.

Définition 2.7 *Soit un rapport d'observation E appartenant au langage de représentation des exemples L_E et une formule G appartenant au langage de représentation des généralisations L_G. G est une généralisation confirmatoire de E, ce que l'on note $E \mid\!\approx G$, si et seulement si G est confirmée par E, mais n'en est pas une conséquence logique, et s'il n'y pas de généralisation plus générale que G, ce qui s'écrit formellement:*

- $E \mid\!\sim G$, *et*

- $E \not\models G$, *et*

- *Si* G' *est une clause telle que* $E \mid\!\sim G'$, $E \not\models G'$ *et* $G' \models G$ *alors* $G \models G'$.

Par exemple, la formule $G =_d \forall X, oiseau(X) \vee chat(X)$ est une généralisation confirmatoire du rapport d'observation $E =_d \{oiseau(Titi), chat(Sylvestre)\}$. G est en effet confirmée par E, n'en est pas une conséquence logique, et $\forall X, oiseau(X)$ ou $\forall X, chat(X)$ ne sont pas confirmées par E.

Il reste à vérifier que la relation de généralisation confirmatoire $\not\approx$ que nous proposons satisfait les postulats de rationalité donnés au paragraphe 2.3. Il est clair que les axiomes de la généralisation confirmatoire, c'est-à-dire le fondement confirmatoire, le raisonnement ampliatif et la généralité, sont satisfaits.

Le principe de vérification n'est malheureusement compatible avec la relation de confirmation utilisant le domaine Herbrand de l'ensemble d'observation. En effet, si nous considérons $E =_d p(a)$, $H =_d \forall X, p(X)$ et $F =_d p(b) \wedge (p(b) \vee p(c)$ alors $E \mid\!\sim H$ et $E \wedge H \models F$ mais $E \wedge F \not\mid\!\sim H$.

Propriété 2.2 *La relation de généralisation confirmatoire* $\not\approx$ *satisfait le principe de falsification.*

Preuve : Supposons $E \not\approx H$ et $E \wedge H \models F$. Montrons $E \wedge \neg F \not\approx H$.
$E \wedge H \models F$ donc $E \wedge H \wedge \neg F$ est contradictoire.
Donc $FD_C \wedge AE \wedge E \wedge H \wedge \neg F$ est contradictoire, donc $FD_C \wedge AE \wedge E \wedge \neg F \models \neg H$.
Par cohérence de la déduction logique, si $FD_C \wedge AE \wedge E \wedge \neg F$ est cohérent (ce que l'on suppose sinon il est inintéressant de considérer les généralisations de $E \wedge \neg F$), $FD_C \wedge AE \wedge E \wedge \neg F \models \neg H$.
Donc $E \wedge \neg F \not\mid\!\sim H$, donc $E \wedge \neg F \not\approx H$ #

La relation de généralisation confirmatoire $\not\approx$ ne satisfait pas la monotonie prudente. En effet, si nous considérons $E =_d p(a)$, $H =_d \forall X, p(X)$ et $F =_d \forall X, p(X)$ alors $E \not\approx F$ et $E \not\approx H$ mais $E \wedge F \not\approx H$ car $E \wedge F \models H$.

La condition de cohérence est satisfaite [Hempel, 1943]. L'équivalence logique à droite est une conséquence des propriétés du développement $D_{C_E}(H)$ prouvés également par Carl Hempel [Hempel, 1943]. L'équivalence « logique » à gauche est satisfaite si l'on considère que deux rapports d'observation sont équivalents s'ils sont logiquement équivalents et contiennent les mêmes individus (première approche) ou si on limite C à l'ensemble des individus essentiels.

2.4.3 Comment calculer les généralisations?

Dans ce paragraphe, nous expliquons comment le modèle présenté aux paragraphes précédents peut être utilisé pour calculer des généralisations. L'approche proposée est, comme toutes les approches existantes, du type *generate-and-test.* Nous indiquons comment la relation de confirmation peut être testée sans nécessairement utiliser de techniques de démonstration automatique en logique des prédicats du premier ordre avec égalité.

La fonction `généralisation` ci-après construit « à la limite » l'ensemble des généralisations confirmatoires de E selon notre modèle :

```
fonction generalisation(E)
    Initialiser EG a l'ensemble vide
    Considerer toutes les clauses G de L_G par taille croissante
        Si E |~ G et E |/= G
```

```
        Alors s'il n'existe pas G' ∈ EG tel que G' ⊨ G
            Alors
                Pour tout G' ∈ EG tel que G ⊨ G' faire
                    Supprimer G' de EG
                Ajouter G a EG
    Rendre(EG)
```

Cette fonction est très inefficace en pratique : elle ne se termine pas car L_G est infini. Elle ne se termine pas également si $\vdash\!\sim$ ou $\models$ ne sont pas décidables. Il peut être intéressant de limiter l'espace des généralisations candidates parcouru, c'est-à-dire d'introduire un biais de langage plus fort que celui requis initialement. Toutefois, dans ce cas, la fonction **généralisation** ci-avant ne produira plus en général l'ensemble des généralisations confirmatoires de E telles que définies à la définition 2.6. On peut limiter l'espace des généralisations candidates de différentes façons, par exemple en n'utilisant que des clauses à champ restreint, où toutes les variables occurant dans un littéral positif apparaissent dans un littéral négatif [Helft, 1989; De Raedt et Bruynooghe, 1993]. Une autre restriction possible est d'utiliser un ordre sur les atomes et d'imposer que les littéraux négatifs soient plus petits pour cet ordre que les littéraux positifs dans les généralisations [Helft, 1988]. Notons bien qu'on ne peut pas garantir que les généralisations confirmatoires de E sont des clauses de longueur inférieure à k pour k fixé. Par exemple, étant donné le rapport d'observation $E =_d même\_famille(Titi, Titi)$, les clauses :

$G_1 =_d \forall X_0, X_1, même\_famille(X_0, X_1) \vee même\_famille(X_1, X_0)$

$G_2 =_d \forall X_0, X_1, X_2, même\_famille(X_0, X_1) \vee même\_famille(X_1, X_2) \vee même\_famille(X_2, X_0)$

$\vdots$

sont des généralisations confirmatoires de E incomparables pour la déduction logique [Helft, 1989]. Cet exemple montre aussi que l'ensemble des généralisations confirmatoires de E n'est en général pas fini. Une troisième restriction consiste à limiter les prédicats admissibles ou le nombre de littéraux intervenant dans les généralisations (de façon à favoriser leur compréhensibilité).

La relation de confirmation peut être implantée de plusieurs façons. La première implémentation consiste simplement à utiliser la définition 2.4 et à vérifier à l'aide d'un démonstrateur automatique que l'hypothèse H est une conséquence logique de $E \wedge FD_C \wedge AE$. Une seconde implémentation consiste à transformer l'hypothèse pour refléter la fermeture du domaine avant d'évaluer si elle est une conséquence logique des observations E. Cette approche repose sur la notion de *développement d'une formule pour un ensemble fini d'individus.*

Définition 2.8 *Le développement d'une formule H pour un ensemble fini d'individus C, noté $D_C(H)$, est défini inductivement :*

$$
\begin{array}{lll|lll}
D_C(\neg H) & =_d & \neg D_C(H) & D_C(H \Rightarrow G) & =_d & D_C(H) \Rightarrow D_C(G) \\
D_C(H \wedge G) & =_d & D_C(H) \wedge D_C(G) & D_C(H \vee G) & =_d & D_C(H) \vee D_C(G) \\
D_C(\forall x H) & =_d & \bigwedge_{c \in C} D_C(H[x \leftarrow c]) & D_C(\exists x H) & =_d & \bigvee_{c \in C} D_C(H[x \leftarrow c]) \\
D_C(H) & =_d & H, \textit{ si } H \textit{ est un atome terminal.} & & &
\end{array}
$$

où $H[x \leftarrow c]$ est la formule H dans laquelle chaque occurrence libre de x est remplacée par c.

L'ensemble des individus que nous considérons est l'ensemble des individus mentionnés dans E, c'est-à-dire le domaine de Herbrand $D_H(E)$ de E, mais les techniques et résultats présentés dans ce paragraphe s'appliquent également au cas où C est l'ensemble des individus essentiels $I(E)$ de E. Chaque formule candidate est donc développée pour l'ensemble C_E de tous les

symboles de constantes apparaissant dans E. Par exemple, le développement de l'hypothèse

$$H =_d \forall X(oiseau(X) \Rightarrow vole(X))$$

pour le domaine de Herbrand de

$$E =_d \{oiseau(Titi),\ vole(Titi),\ vole(Superman),\ chat(Sylvestre),\ \forall X(chat(X) \Rightarrow \neg oiseau(X))\}$$

est la formule

$$(oiseau(Titi) \Rightarrow vole(Titi)) \wedge (oiseau(Superman) \Rightarrow vole(Superman) \wedge (oiseau(Sylvestre) \Rightarrow vole(Sylvestre)).$$

Cette formule terminale représente la même information que H si l'on considère que l'Univers est réduit à $\{Titi, Superman, Sylvestre\}$.

En présence de l'hypothèse de fermeture du domaine pour un ensemble d'individu C et des axiomes de l'égalité, H est logiquement équivalente à son développement pour C.

Propriété 2.3 *Pour toute formule $H \in L_H$, $FD_C \wedge EA \models (H \Leftrightarrow D_C(H))$*

Preuve : Il est clair que $FD_C \wedge EA \models H \Rightarrow D_C(H)$ puisque $H \models D_C(H)$.

Montrons que $FD_C \wedge EA \models D_C(H) \Rightarrow H$. Ceci revient à montrer que $FD_C \wedge EA \wedge D_C(H) \models H$, d'après le théorème de la déduction, c'est-à-dire que $\neg H \wedge FD_C \wedge EA \wedge D_C(H)$ est contradictoire. Or $\neg H \wedge FD_C \wedge EA \wedge D_C(H)$ est contradictoire si et seulement si sa skolémisation est contradictoire. Soit $\alpha_1, \ldots, \alpha_m$ les constantes de Skolem introduites à la place des variables désignant les individus. Écrivons $\neg H$ sous la forme $l_1(\alpha_1, \ldots, \alpha_m) \wedge \ldots \wedge l_n(\alpha_1, \ldots, \alpha_m)$. Supposons que $\neg H \wedge FD_C \wedge EA \wedge D_C(H)$ admet un modèle $\mathcal{M}$, alors d'après l'hypothèse de fermeture du domaine, chaque α_j est l'un des individus connus. Soit $i(\alpha_j)$ l'individu associé à chaque α_j, $\mathcal{M}$ est un modèle de $D_C(H)$ donc un modèle de $\neg l_1(i(\alpha_1), \ldots, i(\alpha_m)) \vee \ldots \vee \neg l_n(i(\alpha_1), \ldots, i(\alpha_m))$. Or $\alpha_j = i(\alpha_j)$, donc $\mathcal{M}$ est un modèle de $l_1(i(\alpha_1), \ldots, i(\alpha_m)) \wedge \ldots \wedge l_n(i(\alpha_1), \ldots, i(\alpha_m))$. Ceci est contradictoire. Donc $\neg H \wedge FD_C \wedge EA \wedge D_C(H)$ n'admet pas de modèle. #

Ainsi une hypothèse H est confirmée si et seulement si $D_C(H)$ est une conséquence logique de E. Le recours au développement permet de se dispenser d'un démonstrateur au premier ordre avec égalité pour tester la confirmation. Cela n'entraîne toutefois pas forcément une plus grande efficacité car le développement d'une hypothèse H peut être de taille exponentielle dans l'arité du prédicat p de taille maximale occurant dans H. Par exemple, étant donné $C =_d \{1, 2, 3, 4\}$ et $H =_d \forall X, \forall Y, \forall Z, p(X, Y, Z)$, le développement de H est $D_C(H) =_d p(1, 1, 1) \wedge p(1, 1, 2) \wedge \ldots \wedge p(4, 4, 4)$. La formule développée $D_C(H)$ est de taille $|D_C(H)| = 64 = |C|^{arité(p)}$.

Comme nous l'avons indiqué au chapitre précédent, les généralisations de E ne sont pas « construites » à partir de E, mais elles résultent d'un processus de génération de candidats qui sont ensuite évalués. Les techniques existantes utilisent également une approche « generate-and-test ». Une généralisation confirmatoire est une hypothèse confirmée par les observations, sans en être une conséquence logique, et qui n'est pas impliquée strictement par une autre généralisation. Il est donc préférable d'examiner au préalable les hypothèses vraisemblablement parmi les plus générales. Cela peut se faire en générant les candidats dans un ordre produisant les clauses les plus courtes d'abord (même si rien ne garantit que les clauses les plus courtes soient les plus générales).

2.5 Discussion

Dans ce paragraphe, nous motivons les restrictions que nous avons appliquées aux langages de représentation utilisés dans notre modèle de généralisation. Nous comparons également l'expressivité et les résultats de notre modèle avec ceux fournis par quelques approches existantes.

2.5.1 Motivations des restrictions sur les langages de représentation

Les langages de représentation des observations, des hypothèses confirmées et des généralisations ne sont pas des langages quelconques de la logique des prédicats.

Pour commencer, nous n'avons pas considéré les symboles de fonctions autres que les constantes. Bien qu'ils apportent une expressivité plus grande, ils rendent également infini le domaine de Herbrand des observations. Dans ce cas, l'hypothèse de fermeture du domaine des observations n'est plus une formule.

Les observations doivent avoir un domaine de Herbrand non vide ou posséder un individu essentiel (selon les cas). En effet, dans le cas contraire, E peut confirmer des hypothèses incohérentes avec E. Par exemple, $\forall X, \neg vole(X)$ est confirmée par $\forall X, vole(X)$ puisque le développement de $\forall X, \neg vole(X)$ pour l'univers de Herbrand vide est $Vrai$. Il est clair que cela contredit la condition de cohérence.

L'égalité a les mêmes effets. Le développement de $H =_d \forall X, (X = Superman)$ pour le domaine de Herbrand de $E =_d \forall X, (X \neq Superman)$ est $Superman = Superman$, donc H est confirmé par E, mais est incohérent avec E. La condition de cohérence est encore contredite. De plus, si l'on autorise l'égalité, l'hypothèse de fermeture du domaine, qui établit que tout individu est un individu connu, et qui s'exprime à l'aide de l'égalité, est toujours confirmée. Elle ne peut cependant pas être considérée comme une généralisation puisqu'elle rend impossible la prise en compte de nouveaux individus (elle contredit leur existence).

Les formules existentielles sont également proscrites. $E =_d \exists X, \neg vole(X) \wedge vole(Superman)$ confirme $H =_d \forall X, vole(X)$ bien que les deux formules soient incohérentes entre elles.

Les langages de représentation retenus sont donc des langages de la logique clausale des prédicats du premier ordre sans symboles de fonctions et sans égalité (toute formule universelle peut s'écrire de façon équivalente sous forme clausale). Ce choix d'une théorie clausale est également celui retenu dans les approches existantes d'induction confirmatoire, souvent avec des restrictions supplémentaires sur la forme des clauses autorisées, par exemple pas de clauses ne contenant que des littéraux positifs ou que des littéraux négatifs, ou encore toutes les variables apparaissant dans un littéral positif doivent apparaître dans au moins un littéral négatif. L'utilisation de clauses est parfois justifiée en arguant de ce que les théories clausales sont faciles à comprendre. Cet argument est discutable car s'il est vrai qu'une clause de longueur raisonnable est facile à interpréter, les clauses de grande longueur sont moins intuitives. Toutes les restrictions listées ci-dessus sont dictées par le niveau cognitif. En effet, les exemples que nous avons donnés contredisent les postulats de rationalité. La seule restriction imposée par le niveau inférentiel est l'interdiction des fonctions autres que les constantes. Cette restriction n'est imposée par aucune raison logique, si ce n'est l'indécidabilité de la relation de déduction logique en présence de fonctions. Il serait souhaitable que les symboles de fonctions soient autorisés. Cela est une perspective de développement de nos travaux.

2.5.2 Comparaison à quelques approches existantes

La relation de confirmation que nous avons définie ne nécessite pas de complétion non nécessaire pour la généralisation fondée sur l'hypothèse des similarités selon Hempel. Les approches existantes utilisent des principes de complétion plus forts qui conduisent à manquer des généralisations intuitivement intéressantes et à proposer des hypothèses qui ne semblent pas acceptables.

Considérons par exemple, la base de connaissances :

$\{oiseau(Titi),\ imaginaire(Titi),\ vole(Superman), imaginaire(Superman)\}$.

Les seules généralisations acceptées par le modèle proposé par Nicolas Helft [Helft, 1988; Helft, 1989] sont :

$$\{\forall X, \neg oiseau(X) \vee imaginaire(X);\ \forall X, \neg vole(X) \vee imaginaire(X);$$
$$\forall X, \neg imaginaire(X) \vee oiseau(X) \vee vole(X)\}.$$

Le système Claudien [De Raedt et Bruynooghe, 1993] propose les généralisations :

$$\{\forall X, \neg oiseau(X) \vee \neg vole(X);\ \forall X, \neg oiseau(X) \vee imaginaire(X);$$
$$\forall X, \neg vole(X) \vee imaginaire(X);\ \forall X, \neg imaginaire(X) \vee oiseau(X) \vee vole(X)\}.$$

Le modèle de généralisation que nous avons développé propose les généralisations :

$$\{\forall X, imaginaire(X);\ \forall X, oiseau(X) \vee vole(X)\}$$

Les clauses que nous obtenons sont plus générales que celles proposées par le modèle de Nicolas Helft ou par Claudien ; de plus, la généralisation $\forall X, \neg oiseau(X) \vee \neg vole(X)$ n'est pas dérivée dans notre modèle sans complétion explicite additionnelle.

2.6 Paradoxes de la confirmation

L'induction semble difficile à capturer par des méthodes formelles (syntaxiques). Il n'existe pas un modèle universel et consensuel pour la généralisation inductive mais plusieurs modèles couvrant différentes facettes. Un certain nombre de paradoxes sont associés à la confirmation. Les modèles pour la généralisation fondée sur l'hypothèse des similarités selon Hempel semblent plus résistants que ceux pour la généralisation fondée sur l'hypothèse des similarités selon Nicod [Nicod, 1961]. Dans ce cadre, une observation confirme une implication $G \Rightarrow D$ si et seulement si elle satisfait G *et* D. Cela contredit l'équivalence à droite. En effet, l'implication $G \Rightarrow D$ est équivalente à sa contraposée $\neg D \Rightarrow \neg G$, mais une observation ne peut pas satisfaire en même temps $G \wedge D$ et $\neg D \wedge \neg G$, donc une observation ne peut pas confirmer une implication et sa contraposée, qui lui est pourtant équivalente.

L'écriture des hypothèses sous la forme d'implication $G \Rightarrow D$ associée à l'équivalence à droite conduit au paradoxe de Hempel. Un stylo rouge confirme que tous les corbeaux sont noirs, puisqu'un stylo rouge est un objet non noir qui n'est pas un corbeau. Une observation qui satisfait $\neg D \wedge \neg G$, dans le cas présent *n'est pas un corbeau* et *n'est pas noir*, confirme l'implication $\neg D \Rightarrow \neg G$, *tout ce qui n'est pas noir n'est pas un corbeau.* Cette implication est équivalente à sa contraposée $G \Rightarrow D$, *tout corbeau est noir*, qui, d'après l'équivalence à droite, est elle aussi confirmée. Cela heurte le sens commun car nous appliquons intuitivement le critère de Nicod et nous pensons qu'une implication est confirmée si ses prémisses et sa conclusion sont vérifiées. Ce problème vient de ce que nous oublions qu'une implication logique n'est qu'une notation particulière d'une disjonction. Ainsi les implications $G \Rightarrow D$ *et* $\neg D \Rightarrow \neg G$ sont logiquement équivalentes à la clause $\neg G \vee D$. Une observation confirme la clause $\neg G \vee D$ si elle satisfait $\neg G$ *ou* D. Le soi-disant paradoxe de Hempel n'est plus un paradoxe : un stylo rouge n'est pas un corbeau, donc confirme $\forall X, \neg corbeau(X) \vee noir(X)$. Le paradoxe de Watanabe [Watanabe, 1965] est une variante de celui de Hempel. Les deux affirment que « n'importe quoi confirme n'importe quoi », un stylo rouge confirme $\forall X, \neg corbeau(X) \vee noir(X)$, mais aussi $\forall X, \neg corbeau(X)$. Laquelle des deux clauses est une généralisation? Le prédicat *corbeau* n'est utilisé pour la construction de clauses que s'il apparaît dans la base de connaissances. On distingue deux cas : soit il apparaît sous forme positive, donc il y a un corbeau dans le rapport d'observation E, et E confirme $\forall X, \neg corbeau(X) \vee noir(X)$ et rejette $\forall X, \neg corbeau(X)$, soit il apparaît seulement sous forme négative, donc $\forall X, \neg corbeau(X)$ est confirmée, et puisque c'est la clause la plus générale, elle élimine l'autre. La généralisation contribue à éliminer les paradoxes du type « n'importe quoi confirme n'importe quoi ».

Voir les implications comme des clauses et ne retenir que les plus générales n'élimine pas tous les paradoxes, par exemple le paradoxe de Goodman subsiste. Ce paradoxe est connu comme le paradoxe des émeraudes : on a observé n émeraudes et toutes sont vertes. Ces observations confirment donc l'implication *toute émeraude est verte* et l'implication *toute émeraude n'est pas verte ou a été observée parmi les n premières observations.* Lorsque l'on considère une $n+1$-ième émeraude, la première implication suggère que l'émeraude est verte et la seconde implication qu'elle *n'est pas verte ou a été observée parmi les n premières observations.* Comme elle n'a pas été observée parmi les n premières observations, on déduit de la seconde implication que la $n+1$-ième émeraude n'est pas verte. On aboutit donc à une contradiction. Lorsque l'on se place dans le cadre de la généralisation confirmatoire, la seconde implication est impliquée par la généralisation *toute émeraude a été observée parmi les n premières observations.* Il est clair que cette implication ne sera vérifiée par aucune observation suivante. Le problème vient de ce que l'induction est non valide : une hypothèse confirmée par des observations peut être invalidée par les observations suivantes. Dans le paradoxes des émeraudes, un problème typique qui va conduire à l'induction d'implications non valides est que l'énoncé fait référence à la singularité des observations. Il revient à conclure que toute émeraude a été observée. On tombe sur un problème analogue si l'on autorise le prédicat d'égalité dans les observations : dans ce cas, l'hypothèse de fermeture du domaine $\forall x(x = a_1) \vee ... \vee (x = a_n)$ (où les a_i sont tous les individus connus) est confirmée. Bien entendu, on ne sait pas capturer syntaxiquement cette idée de référence aux observations. En outre, même si la référence aux observations est une source de paradoxes, elle n'en est pas la seule cause. Si l'on observe qu'il a fait beau les 5 premiers jours de juillet et que le soleil s'est levé ces 5 jours, on peut induire *chaque jour le soleil se lève* et *chaque jour il fait beau ou le soleil ne se lève pas*, supposons que le sixième jour, il ne fasse pas beau, la première implication prédit que le soleil se lève et la seconde qu'il ne se lève pas. Cet énoncé du paradoxe de Goodman ne repose pas sur la singularité des observations.

2.7 Résumé et conclusion

Dans ce chapitre, après avoir défini la généralisation confirmatoire, nous avons présenté des postulats de rationalité que doit satisfaire tout modèle de généralisation confirmatoire fondé sur l'hypothèse des similarités selon Hempel. Nous avons mis l'accent sur la nécessité de séparer le niveau rationnel et le niveau inférentiel lors de la définition d'un modèle de généralisation confirmatoire. Nous avons ensuite proposé un modèle de généralisation confirmatoire s'appuyant sur les relations de confirmation proposées par Carl Hempel [Hempel, 1943; Hempel, 1945]. Nous avons également montré que le choix du critère de confirmation et de la forme des hypothèses influent sur certains paradoxes de la confirmation.

Le modèle que nous proposons utilise des langages de représentation plus expressifs que ceux autorisés dans les approches existantes. Celles-ci sont, en effet, fondées sur des sémantiques non monotones qui privilégient les faits positifs et supposent que les faits inconnus sont négatifs. Nous n'utilisons pas la même sémantique et le modèle que nous proposons peut prendre en compte des descriptions comportant des faits positifs, négatifs ou absents. Il est néanmoins possible de compléter la base de connaissances en appliquant une complétion telle que l'hypothèse du monde clos, si cela fait partie du choix de représentation. Mais une telle complétion n'est pas nécessaire au mécanisme d'induction. Comme nous l'avons montré, l'induction ne requiert qu'une circonscription des individus alors que la plupart des approches existantes de la généralisation utilisent une circonscription des propriétés. L'emploi de mécanismes de complétion trop forts oblige à contraindre davantage le langage de représentation des généralisations et

peut conduire à la production de généralisations indésirables et à l'omission de généralisations attendues.

L'algorithme de généralisation que nous avons proposé est très inefficace (il ne se termine pas toujours). La stratégie de type *generate-and-test* sur lequel il s'appuie est également utilisée dans la plupart des approches existantes. Définir des modèles de généralisation plausibles d'un point de vue cognitif, fondés sur un langage de représentation très expressif, et possédant une algorithmique efficace reste encore à l'heure actuelle un pari à relever.

Chapitre 3

Généralisation confirmatoire dans un cadre attribut-valeur

Les langages de la logique des prédicats utilisés dans les chapitres précédents offrent une grande expressivité. Le prix à payer est une complexité de calcul accrue pour la confirmation et la génération des hypothèses. Des formalismes plus simples sont donc souvent préférés. Parmi ceux-ci, les descriptions à base d'attributs-valeurs sont les plus utilisées. Nous rappelons pour commencer comment sont représentées les connaissances dans ce cadre. Nous définissons ensuite un modèle de généralisation confirmatoire dans un cadre attribut-valeur. Nous vérifions que les généralisations obtenues sont identiques à celles qui seraient obtenues en logique des prédicats. L'intérêt de spécialiser notre modèle de généralisation au formalisme attribut-valeur est d'obtenir un mécanisme de calcul des généralisations plus efficace : celui des premiers impliqués. Lorsque le problème de généralisation peut se formaliser dans le cadre attribut-valeur, nous pouvons donc tirer parti des techniques de calcul de premiers impliqués existantes, en particulier les arbres d'énumération d'ensembles.

3.1 Représentation des connaissances

Dans ce chapitre, les connaissances sont représentées dans un langage d'attributs-valeurs. Ces langages sont des langages essentiellement propositionnels (la logique propositionnelle est présentée en annexe A.2). Plus précisément, un langage *attribut-valeur* est un langage propositionnel où les propositions sont des couples $(attribut = valeur)$. Contrairement à la logique propositionnelle standard, les propositions ne sont pas indépendantes. Par exemple, la conjonction de deux propositions distinctes n'est pas toujours satisfaisable ($(longueur = 2) \wedge (longueur = 3)$ est contradictoire). Ces langages sont les plus utilisés en apprentissage actuellement. Ils offrent un formalisme suffisamment expressif pour nombre d'applications.

Dans le formalisme attribut-valeur, les individus sont décrits par leurs valeurs pour un ensemble d'attributs fixés. Un couple $(attribut = valeur)$ représente une propriété d'un individu. On distingue les attributs nominaux des attributs linéaires. Un attribut *nominal* prend un nombre fini de valeurs qui ne sont pas ordonnées. Ce peut être un booléen (Vrai ou Faux). On parle d'attribut *binaire* quand un attribut nominal ne prend que deux valeurs, d'attribut *multi-valué* sinon. Le type de vin (rouge, blanc ou rosé) est un attribut multi-valué. Un attribut *linéaire* est un attribut dont l'ensemble des valeurs est muni d'un ordre total. Cet ensemble peut prendre un nombre infini ou fini de valeurs. La taille d'une bouteille (petite, moyenne, ou grande) est un attribut linéaire. On distingue parfois les attributs discrets des attributs

continus. Un attribut *continu* peut prendre une infinité de valeurs. C'est toujours un attribut linéaire, par exemple un nombre réel. Un attribut *discret* prend un nombre dénombrable de valeurs. Ces valeurs peuvent être incomparables, auquel cas l'attribut est nominal, par exemple, le type de vin, sinon c'est un attribut linéaire comme par exemple la taille de la bouteille. Ce peut être également un nombre entier. Les attributs linéaires discrets résultent fréquemment de la discrétisation d'un attribut continu. La discrétisation d'un attribut continu ne produit jamais un attribut nominal. En effet, les valeurs obtenues après discrétisation restent toujours ordonnées.

Si un couple (*attribut* = *valeur*) représente une propriété d'un individu, comment sont représentés les individus ? Alors qu'en logique des prédicats les individus connus sont représentés par des constantes, les individus n'apparaissent pas explicitement dans les langages d'attributs-valeurs. Chaque individu connu est représenté par l'ensemble (logiquement la conjonction) des valeurs associées à chaque attribut. Le rapport d'observation contient donc un ensemble de conjonctions d'attributs-valeurs où chaque conjonction décrit un individu différent. Deux bouteilles de vin, une petite bouteille de rosé et une bouteille moyenne de vin rouge sont par exemple représentées par les conjonctions $\{(taille = petite) \wedge (type = rosé), (taille = moyenne) \wedge (type = rouge)\}$. Les hypothèses sont des formules *implicitement universellement quantifiées.* La règle indiquant que *les vins de Champagne sont pétillants* s'écrit $\neg(appelation = Champagne) \vee (pétillant = Vrai)$. De telles hypothèses s'appliquent à chaque individu décrit par une conjonction de propriétés. La théorie du domaine qui contient habituellement un ensemble de formules implicitement universellement quantifiées ne peut pas être ajoutée aux observations : la formule $(taille = grande)$ décrit une grande bouteille particulière si elle est associée à une observation, alors qu'elle signifie que tous les individus sont des grandes bouteilles si elle est associée à une hypothèse ou à un élément de la théorie du domaine.

3.2 Spécialisation du modèle proposé au cadre attribut-valeur

Dans ce paragraphe, nous montrons comment le modèle de généralisation que nous avons proposé au chapitre précédent peut se spécialiser à un cadre attribut-valeur. Puisque les langages d'attributs-valeurs, en tant que langages propositionnels, sont des cas particuliers de langages du premier ordre, on pourrait envisager de réécrire les observations dans un formalisme à base de prédicats et profiter du modèle développé précédemment. Nous serions malheureusement alors soumis à la complexité de mise en œuvre de la recherche des généralisations au premier ordre. Cette complexité est d'ordinaire si élevée -et nous n'y échappons pas- que plusieurs travaux ont cherché au contraire à utiliser la logique propositionnelle comme espace de recherche lors d'un apprentissage en logique des prédicats, par exemple FOIL [Quinlan, 1990; Quinlan et Cameron-Jones, 1993], [Fensel *et al.*, 1995] ou encore [Sebag et Rouveirol, 1997]. Notre objectif est donc de déterminer comment notre modèle de généralisation confirmatoire se spécialise à une description attributs-valeurs.

3.2.1 Représentation des observations et des hypothèses

Les individus et les quantifications sont implicites dans une description attribut-valeur. Chaque proposition, couple (*attribut* = *valeur*), décrit une propriété d'un individu connu lorsqu'elle est associée à une observation. Plus précisément, tous les individus sont décrits par leurs valeurs pour un ensemble d'attributs fixé.

Définition 3.1 *Soit un ensemble d'attributs A. Chaque individu i est représenté par la conjonc-*

tion des couples $(a = v_a^i)$ *où* v_a^i *est la valeur décrivant l'individu i suivant l'attribut a. Un individu désigne toujours une description dont tous les attributs sont valués. Une telle description complète n'est toutefois pas nécessairement fournie (certaines valeurs peuvent être manquantes).*

Par exemple, les bouteilles de vin sont représentées par les attributs *couleur*, *taille*, *appelation* et *pétillant*. Les hypothèses et la théorie du domaine utilisée sont implicitement quantifiées universellement. En particulier, une théorie du domaine propositionnelle n'apporte aucune description sur un individu donné. Elle contient des lois qui s'appliquent à tous les individus.

Ainsi, suivant le contexte, une proposition (*attribut* = *valeur*) désigne soit une propriété d'un individu, soit une propriété de l'ensemble des individus. Pour dénoter l'interprétation donnée à une proposition, on peut lui associer une formule de la logique des prédicats du premier-ordre qui en donne une forme explicite. A chaque proposition p est associé le prédicat unaire $p(x)$. La propriété p de l'individu i est donc associée à $p(i)$. L'hypothèse $p \vee q$ se réécrit en $\forall x, p(x) \vee q(x)$. À chaque proposition p de la théorie du domaine est associée $p(x)$ et la théorie est quantifiée universellement.

Soient I l'ensemble des individus connus et L_{prop} le langage propositionnel de description utilisé, nous définissons un langage du premier ordre $L_{préd}$ par :

- un ensemble de constantes c_i tel que chaque individu i est associé de façon biunivoque à la constante c_i,
- un ensemble de prédicats unaires $p(\_)$ tel que chaque proposition p utilisée pour décrire un individu est associée de façon biunivoque à $p(\_)$.

Soient τ une application de $I \times L_{prop}$ vers $L_{préd}$ qui à un individu i et sa description $p_1 \wedge ... \wedge p_n$ associe $\tau(i, p_1 \wedge ... \wedge p_n) = p_1(i) \wedge ... \wedge p_n(i)$. Plus précisement, soient i un individu et f une formule propositionnelle décrivant i. L'application τ est définie par induction sur f :

- $\tau(i, Vrai) = Vrai$ et $\tau(i, Faux) = Faux$;
- si p est une proposition, $\tau(i, p) = p(i)$;
- $\tau(i, \neg f) = \neg\tau(i, f)$;
- $\tau(i, f \wedge g) = \tau(i, f) \wedge \tau(i, g)$;
- $\tau(i, f \vee g) = \tau(i, f) \vee \tau(i, g)$;
- $\tau(i, f \Rightarrow g) = \tau(i, f) \Rightarrow \tau(i, g)$;
- $\tau(i, f \Leftrightarrow g) = \tau(i, f) \Leftrightarrow \tau(i, g)$;

Une petite bouteille de rosé b est représentée par la conjonction $(couleur = rosé) \wedge (taille = petite)$. On a $\tau(b, (couleur = rosé) \wedge (taille = petite)) =_d (couleur = rosé)(b) \wedge (taille = petite)(b)$.

τ est une injection : la représentation ainsi définie est fidèle. τ n'est évidemment pas surjective.

Un rapport d'observation est l'ensemble des observations relatives aux individus connus. En logique des prédicats, l'ensemble de ces observations est représenté par une conjonction. Par exemple, une bouteille moyenne de vin rouge et une petite bouteille de rosé sont représentées par la formule $rouge(a) \wedge moyenne(a) \wedge rosé(b) \wedge petite(b)$. Cela n'est pas possible dans un cadre propositionnel car la conjonction des propriétés de tous les individus

connus n'a, en général, aucun sens et devient vite contradictoire, par exemple, $(couleur = rouge) \wedge (taille = moyenne) \wedge (couleur = rosé) \wedge (taille = petite)$. Pour pallier ce problème, un ensemble de descriptions d'individus est représenté par une disjonction en logique propositionnelle $((couleur = rouge) \wedge (taille = moyenne)) \vee ((couleur = rosé) \wedge (taille = petite))$. L'ensemble des individus connus (donnés par leurs propriétés) est exactement l'ensemble des individus satisfaisant cette disjonction lorsque les individus sont complètement décrits.

Pour expliciter l'interprétation donnée aux hypothèses et aux formules de la théorie du domaine, on peut considérer un individu X quelconque et utiliser la transformation τ. Si f est une hypothèse ou appartient à la théorie du domaine, on lui associe $\tau(X, f)$. Cette formule de la logique des prédicats est une forme explicite de f. Toute formule associée à tous les individus est implicitement universellement quantifiée $\forall X, \tau(X, f)$. Par exemple, l'hypothèse $\neg(appelation = Champagne) \vee (pétillant = Vrai)$ s'applique à tous les individus. Une forme explicite de cette hypothèse est $\forall X, \neg Champagne(X) \vee pétillant(X)$.

3.2.2 Confirmation

Considérons la description E d'un ensemble d'individus. Une hypothèse H est confirmée par E en présence d'un théorie du domaine Th si et seulement si H est une conséquence logique de E et Th si l'on suppose que tous les individus sont ceux connus.

Définition 3.2 *Soient un rapport d'observation E contenant la description d'un ensemble I d'individus i décrits chacun par une conjonction de littéraux P_i et une théorie du domaine Th. Une clause H est confirmée par E si et seulement si $\forall i, P_i \wedge Th \models H$, i.e. $\bigvee_i P_i \wedge Th \models H$. Le rapport d'observation peut donc être assimilé à $E = \bigvee_i P_i \wedge Th$.*

Nous avons exprimé l'hypothèse de similarité selon Hempel par la fermeture du domaine en logique des prédicats. Dans un cadre propositionnel, les individus étant implicites, l'induction des individus connus aux individus inconnus est implicite. L'inférence réalisée est purement déductive. La seule phase d'induction est d'accepter le résultat comme une propriété s'appliquant à tous les individus.

Par exemple, l'observation *une bouteille de vin rouge a* et *une bouteille de vin blanc b* confirme que *toutes les vins sont rouges ou blancs*. L'hypothèse, dont la représentation propositionnelle est $(couleur = rouge) \vee (couleur = blanc)$, est une conséquence logique du rapport d'observation, représenté par la même formule propositionnelle $(couleur = rouge) \vee (couleur = blanc)$. Une représentation explicite permet de mieux faire apparaître l'induction : l'hypothèse $\forall X, rouge(X) \vee blanc(X)$ est confirmée par $rouge(a) \wedge blanc(b)$.

Bien que l'inférence réalisée pour produire des hypothèses inductives en logique propositionnelle soit purement déductive, les définitions de confirmations propositionnelle et au premier ordre coïncident quand on exprime au premier ordre des observations décrites dans un cadre propositionnel.

Théorème 3.1 *Les observations $\bigvee_{i \in I} P_i$ confirme l'hypothèse H en présence de la théorie Th,*

$$\bigvee_{i \in I} P_i \wedge Th \models H,$$

si et seulement si $\forall X, \tau(X, H)$ est confirmée par $\bigwedge_{i \in I}, \tau(i, P_i) \wedge \forall X, \tau(X, Th)$,

$$\bigwedge_{i \in I}, \tau(i, P_i) \wedge \forall X, \tau(X, Th) \vdash \forall X, \tau(X, H)$$

Preuve :

$\Rightarrow$ Supposons $\bigvee_{i \in I} P_i \wedge Th \models H$. Soit I_1 un modèle de $\bigwedge_{i \in I}, \tau(i, P_i) \wedge \forall X, \tau(X, Th)$, montrons que I_1 satisfait $D_{C_E}(\forall X, \tau(X, H))$.

Soit j un individu de E, définissons l'interprétation propositionnelle I_0 qui associe à chaque proposition p la valeur de vérité associée par I_1 à $\tau(j,p)$: $[p](I_0) = [\tau(i,p)](I_1)$.
I_1 satisfait $\tau(j,P_j) \wedge \forall X, \tau(X,Th)$, donc, par construction, I_0 satisfait $P_j \wedge Th$. Donc I_0 satisfait $\bigvee_{i\in I} P_i \wedge Th$.
$\bigvee_{i\in I} P_i \wedge Th \models H$, donc I_0 satisfait H. Par construction de I_0, I_1 satisfait donc $\forall X, \tau(X,H)(j)$. Ce résultat est vrai pour tous les individus j de E, donc I_1 satisfait $D_{C_E}(\forall X, \tau(X,H))$.

$\Leftarrow$ Supposons $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th) \mid\sim \forall X, \tau(X,H)$. Soit I_0 un modèle de $\bigvee_{i\in I} P_i \wedge Th$, montrons que I_0 satisfait H.
I_0 satisfait $\bigvee_{i\in I} P_i \wedge Th$, donc il existe un individu j tel que I_0 satisfait $P_j \wedge Th$.
Soit I_1 un modèle de $\bigwedge_{i\neq j}, \tau(i,P_i) \wedge \forall X, \tau(X,Th)$. Définissons l'interprétation I_1' par $[\tau(i,p)](I_1') = [\tau(i,p)](I_1)$ si $i \neq j$ et $[\tau(j,p)](I_1') = [p](I_0)$.
I_0 satisfait $P_j \wedge Th$ et I_1 satisfait $\bigwedge_{i\neq j}, \tau(i,P_i) \wedge \forall X, \tau(X,Th)$, donc, par construction, I_1' satisfait $\bigwedge_i, \tau(i,P_i) \wedge \forall X, \tau(X,Th)$.
$\bigwedge_{i\in I}, \tau(i,P_i) \wedge \forall X, \tau(X,Th) \mid\sim \forall X, \tau(X,H)$, i.e. $\bigwedge_{i\in I}, \tau(i,P_i) \wedge Th \models \bigwedge_{i\in I} \forall X, \tau(X,H)(i)$, donc I_1' satisfait $\bigwedge_{i\in I} \forall X, \tau(X,H)(i)$, donc I_1' satisfait $\forall X, \tau(X,H)(j)$. Donc, par construction, I_0 satisfait H.

#

Dans l'exemple précédent, $(couleur = rouge) \vee (couleur = blanc) \models (couleur = rouge) \vee (couleur = blanc)$ et $rouge(a) \wedge blanc(b) \mid\sim \forall X, rouge(X) \vee blanc(X)$.

3.2.3 Généralisation

Dans un cadre propositionnel, les généralisations confirmatoires d'un rapport d'observation E sont les premiers impliqués de E, c'est-à-dire les clauses les plus générales qui sont des conséquences logiques de E.

Définition 3.3 *Un* premier impliqué *d'une formule propositionelle F est une clause C telle que :*

- *(implication) $F \models C$;*
- *(généralité) si C' est une clause telle que $F \models C'$ et $C' \models C$ alors $C \models C'$.*

On note $IP(F)$ les premiers impliqués de F. La notion de premier impliqué est duale de la notion de premier implicant: C est un premier impliqué de F si et seulement si $\neg C$ est un premier implicant de $\neg F$. La notion de premier implicant est définie comme suit :

Un premier implicant *d'une formule propositionelle F est une conjonction de littéraux C telle que :*

- *$C \models F$;*
- *si C' est une conjonction de littéraux telle que $C' \models F$ et $C \models C'$ alors $C' \models C$.*

On note $PI(F)$ les premiers implicants de F.

Définition 3.4 *L'ensemble des généralisations confirmatoires de E est l'ensemble de ses premiers impliqués $IP(E)$.*

On peut vérifier que $(couleur = rouge) \vee (couleur = blanc)$ est le seul premier impliqué de $E =_d (couleur = rouge) \vee (couleur = blanc)$.

Il peut sembler curieux que la définition ci-dessus ne mentionne pas d'exigence de nouveauté (raisonnement ampliatif). La définition d'impliqué premier ne comporte en effet qu'une

exigence de confirmation (implication) et de généralité. L'absence de l'axiome d'ampliativité est liée à la sémantique particulière de l'induction dans la représentation propositionnelle. En effet, une hypothèse p confirmée par un ensemble d'observations p est interprétée comme $\forall X, p(X)$. L'exigence de nouveauté dans le formalisme des prédicats a pour but d'éviter que $p(i)$ soit une généralisation de $p(i)$; dans le cadre propositionnel, une hypothèse p est toujours universellement quantifiée $\forall X, p(X)$. Elle apporte donc nécessairement plus d'information que celle contenue dans les observations. Même dans le cas incongru où l'on sait que l'ensemble des individus possibles se limite à ceux connus, cette connaissance ne peut pas être intégrée dans les modèles de généralisation que nous avons présentés. Il est en effet impossible de la modéliser dans le cadre propositionnel et même dans le cadre de la logique des prédicats du premier ordre, nous avons restreint le langage de représentation des observations afin qu'il soit impossible d'exprimer la connaissance "tous les individus possibles sont connus". Donc la condition d'ampliativité reste satisfaite. Par exemple, si les seuls individus connus sont $Vrai$ et $Faux$ et qu'ils vérifient la propriété *booléen*, la condition d'ampliativité est vérifiée : $booléen(Vrai) \wedge booléen(Faux) \not\models \forall X, booleen(X)$.

Comme le montre le théorème 3.2 ci-après, cette définition de la généralisation confirmatoire dans un cadre propositionnel est la spécialisation à ce cadre de celle donnée en logique des prédicats (paragraphe 2.4). Pour démontrer ce résultat, nous montrons d'abord que la notion de conséquence logique entre hypothèses est en correspondance dans les deux formalismes.

Propriété 3.1 *Soit deux clauses propositionnelles c_1 et c_2. On a $c_1 \models c_2$ si et seulement si on a $\forall X, \tau(X, c_1) \models \forall X, \tau(X, c_2)$*

Preuve :

$\Rightarrow$ Supposons $c_1 \models c_2$. Soit I_1 un modèle de $\forall X, \tau(X, c_1)$, montrons que I_1 satisfait $\forall X, \tau(X, c_2)$.
Soit j un individu du domaine, définissons l'interprétation propositionnelle I_0 par : $[p](I_0) = [\tau(j, p)](I_1)$.
I_1 satisfait $\forall X, \tau(X, c_1)$, donc I_1 satisfait $\tau(j, c_1)$, donc, par construction, I_0 satisfait c_1.
$c_1 \models c_2$, donc I_0 satisfait c_2.
Par construction, pour tout prédicat p, $[p](I_0) = [\tau(j, p)](I_1)$, donc $[c_2](I_0) = [\tau(j, c_2)](I_1)$.
Puisque I_0 satisfait c_2, I_1 satisfait $\tau(j, c_2)$.
Ce résultat est vrai pour tous les individus du domaine, donc I_1 satisfait $\forall X, \tau(X, c_2)$.

$\Leftarrow$ Supposons $\forall X, \tau(X, c_1) \models \forall X, \tau(X, c_2)$. Soit I_0 un modèle de c_1, montrons que I_0 satisfait c_2.
Définissons l'interprétation I_1 par : $\forall X, [\tau(X, p)](I_1) = [p](I_0)$, c'est-à-dire pour tout individu j du domaine d'interprétation, $[\tau(j, p)](I_1) = [p](I_0)$.
I_0 satisfait c_1, donc, par construction, I_1 satisfait $\tau(j, c_1)$, quelque soit l'individu j du domaine d'interprétation.
Donc I_1 satisfait $\forall X, \tau(X, c_1)$.
$\forall X, \tau(X, c_1) \models \forall X, \tau(X, c_2)$, donc I_1 satisfait $\forall X, \tau(X, c_2)$.
Donc, par construction, I_0 satisfait c_2.

#

Théorème 3.2 *Une clause G est une généralisation propositionnelle du rapport d'observation $E =_d \bigvee_{i \in I} P_i \wedge Th$,*

$$G \in IP(\bigvee_{i \in I} P_i \wedge Th),$$

si et seulement si $\forall X, \tau(X, G)$ est une généralisation selon $\not\approx$ du rapport d'observation $\bigwedge_{i \in I} \tau(i, P_i) \wedge \forall X, \tau(X, Th)$,

$$\bigwedge_{i \in I} \tau(i, P_i) \wedge \forall X, \tau(X, Th) \not\approx \forall X, \tau(X, G).$$

Preuve :

$\Rightarrow$ Soit $G \in IP(\bigvee_{i\in I} P_i \wedge Th)$, montrons que $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th) \mid\!\approx \forall X, \tau(X,G)$.

- (confirmation) $G \in IP(\bigvee_{i\in I} P_i \wedge Th)$, donc $\bigvee_{i\in I} P_i \wedge Th \models G$. D'après le théorème 3.1, $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th) \mid\sim \forall X, \tau(X,G)$.
- (ampliatif) Soit un individu n n'appartenant pas aux observations $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th)$ ne satisfaisant pas G.
  Soit I_1 un modèle de $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th)$ et I_2 un modèle de $t(P_n)$, définissons I_3 par : $[\tau(j,p)](I_3) = [\tau(j,p)](I_1)$ si $j \neq n$, $[\tau(n,p)](I_3) = [\tau(n,p)](I_2)$.
  I_3 est par construction un modèle de $\tau(n,P_n) \wedge \bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th)$, donc un modèle de $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th)$. I_3 est un modèle de n donc il ne satisfait pas $\forall X, \tau(X,G)$. Ainsi $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th) \not\models \forall X, \tau(X,G)$.
- (généralité) Soit H une hypothèse telle que $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th) \mid\!\approx \forall X, \tau(X,H)$, $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th) \not\models \forall X, \tau(X,H)$ et $\forall X, \tau(X,H) \models \forall X, \tau(X,G)$, montrons que $\forall X, \tau(X,G) \models \forall X, \tau(X,H)$
  D'après le théorème 3.1, $\bigvee_{i\in I} P_i \wedge Th \models H$ et d'après la propriété 3.1, $H \models G$, mais $G \in IP(\bigvee_{i\in I} P_i \wedge Th)$, donc, par définition, $G \models H$. D'après la propriété 3.1, $\forall X, \tau(X,G) \models \forall X, \tau(X,H)$.

$\Leftarrow$ Soit G une hypothèse telle que $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th) \mid\!\approx \forall X, \tau(X,G)$, montrons que $G \in IP(\bigvee_{i\in I} P_i \wedge Th)$.

- (confirmation) $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th) \mid\!\approx \forall X, \tau(X,G)$, donc $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th) \mid\sim \forall X, \tau(X,G)$. D'après le théorème 3.1, $\bigvee_{i\in I} P_i \wedge Th \models G$.
- (généralité) Soit H une hypothèse telle que $\bigvee_{i\in I} P_i \wedge Th \models H$ et $H \models G$, montrons que $G \models H$.
  D'après le théorème 3.1, $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th) \mid\!\approx \forall X, \tau(X,H)$, on peut montrer comme ci-dessus que $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th) \not\models \forall X, \tau(X,H)$ et d'après la propriété 3.1, $\forall X, \tau(X,H) \models \forall X, \tau(X,G)$, mais, $\bigwedge_{i\in I} \tau(i,P_i) \wedge \forall X, \tau(X,Th) \mid\!\approx \forall X, \tau(X,G)$ donc, par définition, $\forall X, \tau(X,G) \models \forall X, \tau(X,H)$. D'après la propriété 3.1, $G \models H$.

#

Ainsi la formule $(couleur = rouge) \vee (couleur = blanc)$ est une généralisation propositionnelle du rapport d'observation $E =_d (couleur = rouge) \vee (couleur = blanc)$, de même que $\forall X, rouge(X) \vee blanc(X)$ est une généralisation suivant $\mid\!\approx$ de $rouge(a) \wedge blanc(b)$.

Contrairement au cas du premier ordre (cf. paragraphe 2.4), il n'est pas indispensable d'utiliser une stratégie de type *generate-and-test* pour calculer les généralisations confirmatoires (c'est-à-dire les premiers impliqués de E) suivant notre modèle de généralisation dans le cadre attribut-valeur. Cela n'entraîne pas pour autant l'existence d'un algorithme de calcul des généralisations qui soit efficace.

La source de difficulté est double dans le cas général. D'une part, vérifier qu'une clause G est un premier impliqué d'un ensemble d'observations E étant donné une théorie du domaine Th est NP-difficile dans le cas général. En effet, $Th \wedge E$ est satisfaisable si et seulement si la clause vide n'est pas un premier impliqué de $Th \wedge E$ (dans le cas restant, la clause vide est nécessairement le seul premier impliqué de $Th \wedge E$). Toutefois, en l'absence de théorie du domaine Th, ce problème de décision peut être résolu en temps polynomial dans la taille de E plus la taille de G : il suffit, en effet, de vérifier que G est un impliqué de E (c'est-à-dire que tout P_i de E contient au moins un littéral de G) et qu'aucune sous-clause stricte de G ne vérifie cette propriété.

La seconde source de difficulté réside dans le nombre de premiers impliqués possibles (donc de généralisations confirmatoires) : dans le pire cas, ce nombre est exponentiel dans $|E| + |Th|$

[Chandra et Markowski, 1978]. Alors que la majorité des algorithmes de calcul des premiers impliqués produisent ceux-ci « en bloc » (c'est-à-dire tous à la fois), par exemple [Madre et Coudert, 1994; Jackson et Pais, 1990; Slagle *et al.*, 1970], certains d'entre eux, à savoir [Rymon, 1994; Castell, 1996], permettent la production d'au plus k premiers impliqués où k est une constante donnée en entrée à l'algorithme de calcul. Une telle génération incrémentale et bornée des premiers impliqués est une voie pour « pallier » le problème de leur nombre. Dans la suite, nous présentons l'algorithme de calcul des premiers impliqués proposé dans [Rymon, 1994]. Cet algorithme s'appuie sur la notion d'arbres d'énumération d'ensembles.

3.3 Arbres d'énumération d'ensembles

Les arbres d'énumération d'ensembles sont, comme leur nom l'indique, une technique permettant d'examiner tous les sous-ensembles possibles d'un ensemble. Dans notre cas, les éléments des ensembles sont des littéraux et les sous-ensembles représentent des clauses. Cette technique a été appliquée à diverses problématiques, dont le calcul de premiers impliqués, par Ron Rymon [Rymon, 1992; Rymon, 1993; Rymon, 1994; Rymon, 1996b]. Nous montrons dans ce paragraphe comment les arbres d'énumération d'ensembles peuvent être utilisés afin de générer des candidats dans une approche « generate-and-test » du calcul des généralisations.

3.3.1 Définitions

Un *arbre d'énumération d'ensembles (AEE)*, en anglais *Set-Enumeration tree (SE-tree)*, sert à représenter ou à énumérer des ensembles, c'est-à-dire des combinaisons d'objets dans lesquelles l'ordre n'intervient pas, suivant la méthode d'exploration des « meilleurs d'abord ». Un arbre d'énumération d'ensembles *complet* énumère systématiquement les éléments d'un ensemble d'ensembles en utilisant un ordre imposé au préalable sur les ensembles. Pour des problèmes dont l'espace de recherche est clos par rapport à l'inclusion d'ensembles, cette technique de recherche est complète et sans redondance [Rymon, 1992].

Soit E un ensemble dont on cherche un sous-ensemble vérifiant une propriété donnée. La méthode de Rymon consiste à indexer d'abord de façon unique chaque élément de E par un entier à l'aide d'une fonction $index : E \rightarrow \mathbb{N}$. Pour éviter de visiter plusieurs fois le même sous-ensemble, l'espace de recherche à partir d'un nœud de l'arbre est limité par sa vue.

Définition 3.5 *Étant donné un sous-ensemble de $S \subseteq E$, la* vue *de S est l'ensemble des éléments de E dont l'index est strictement supérieur aux index des éléments de S : $vue(index, S) =_d \{e \in E / index(e) > max_{e' \in S} index(e')\}$*

On peut alors définir formellement les arbres d'énumération d'ensembles.

Définition 3.6 *Soit F un ensemble d'ensembles clos par rapport à l'inclusion d'ensembles (i.e. $\forall S \in F$, si $S' \subseteq S$ alors $S' \in F$). T est un arbre d'énumération d'ensembles pour F si et seulement si :*

1. *La racine de T est étiquetée par l'ensemble vide.*
2. *Les fils d'un nœud de T étiqueté par S sont étiquetés par $\{S \cup \{e\} \in F / e \in vue(index, S)\}$.*

La figure 3.1 représente un arbre d'énumération d'ensembles pour l'ensemble complet des sous-ensembles de $\{1, 2, 3, 4\}$. L'index utilisé est ici l'identité.

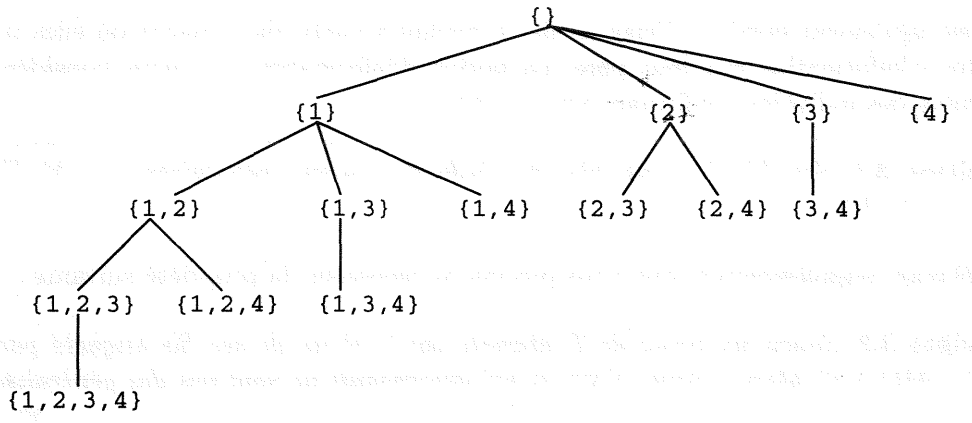


FIG. 3.1 – *Arbre d'énumération d'ensembles pour* $\mathcal{P}(\{1,2,3,4\})$.

Ron Rymon a proposé une définition spécialement adaptée à une recherche dans un espace d'attributs-valeurs. Elle requiert néanmoins que le nombre de valeurs de chaque attribut soit fini. Il est donc nécessaire de discrétiser les attributs linéaires en un ensemble fini d'intervalles.

Définition 3.7 *Soient un ensemble d'attributs* $F =_d \{A_i\}$ *de domaines respectifs* $Dom(A_i)$ *et une fonction index associée.* T *est un arbre d'énumération d'ensembles pour* F *si et seulement si :*

1. *La racine de* T *est étiquetée par l'ensemble vide.*
2. *Les fils d'un nœud de* T *étiqueté par* S *sont étiquetés par* $\{S \cup \{A = v\} \in F/A \in vue(index, S), v \in Dom(A)\}$.

3.3.2 Application au calcul des généralisations

Une clause étant un ensemble de littéraux, les arbres d'énumération d'ensembles peuvent être utilisés pour énumérer des clauses. Ils permettent de générer sans redondance toutes les clauses candidates afin de déterminer les généralisations d'un rapport d'observation E.

La structure des arbres d'énumération d'ensemble permet de ne pas développer, c'est-à-dire d'élaguer, une partie des branches de l'arbre au cours de la recherche. En particulier, les descendants d'une généralisation peuvent être élagués.

Propriété 3.2 *Soit un nœud de* T *étiqueté par* S. *Si* S *est impliquée par une généralisation de* E *alors aucun descendant de* S *n'est une généralisation de* E.

Preuve : S implique ses descendants, donc ils ne peuvent pas être des généralisations. #

Ainsi, en stockant les généralisations au fur et à mesure, on peut élaguer les sous-arbres dont la racine est impliquée par une des généralisations connues.

Une exploration en profondeur d'abord de l'arbre permet de produire les clauses confirmées, mais ne facilite pas la recherche des généralisations. Étant donné une clause confirmée C_1, il n'est pas possible de déterminer si aucune clause confirmée C_2 n'est pas plus générale que C_1 avant d'avoir développé tous les nœuds correspondant aux clauses C_2 possibles. Une exploration en largeur d'abord permet de considérer les clauses les plus générales en premier et de réduire au maximum l'espace de recherche grâce à l'élagage.

Il est également possible d'élaguer des branches à partir du moment où elles n'apportent pas plus d'information que leur père. La notion d'information que nous considérons ici est l'ensemble des individus confirmant une hypothèse.

Définition 3.8 *Soit $M(F)$ l'ensemble des individus connus satisfaisant F : $M(F) =_d \{i \in I / P_i \wedge Th \models F\}$.*

L'élagage supplémentaire que nous proposons repose sur la propriété suivante :

Propriété 3.3 *Soient un nœud de T étiqueté par S et un de ses fils étiqueté par $S \vee r$. Si $M(S) = M(S \vee r)$ alors le nœud $S \vee r$ et ses descendants ne sont pas des généralisations.*

Preuve :

- Si $M(S)$ contient tous les individus alors S est une généralisation et donc la propriété est vraie.

- SI $M(S)$ ne contient pas tous les individus, soit j un individu tel que $P_j \wedge Th \not\models S \vee r$. Supposons que $S \vee r \vee Q$ soit confirmée, montrons que $S \vee Q$ est confirmée.
  $P_j \wedge Th \not\models S \vee r$, donc il existe un modèle I de $P_j \wedge Th$ qui satisfait $\neg S \wedge \neg r$.
  $S \vee r \vee Q$ est une généralisation, donc I satisfait Q.
  Soit J un modèle de $P_j \wedge Th$, J satisfait $S \vee r \vee Q$.

  - Si J satisfait Q alors J satisfait $S \vee Q$.

  - Si J ne satisfait pas Q alors J satisfait $S \vee r$.
    Supposons J ne satisfait pas S. Alors, J est un modèle de $P_j \wedge Th$ qui satisfait $\neg S \wedge r \wedge \neg Q$.
    I est aussi un modèle de $P_j \wedge Th$ et I satisfait $\neg S \wedge \neg r \wedge Q$.
    $P_j \wedge Th$ admet un modèle qui satisfait r et un modèle qui satisfait $\neg r$, donc l'attribut r n'apparaît pas dans $P_j \wedge Th$.
    De même, les attributs de Q n'apparaissent pas dans $P_j \wedge Th$.
    Donc $P_j \wedge Th$ admet un modèle qui satisfait $\neg r \wedge \neg Q$, appelons le K. $P_j \wedge Th$ confirme $S \vee r \vee Q$, donc K satisfait S.
    K satisfait S et J satisfait $\neg S$, donc les attributs de S n'apparaissent pas dans $P_j \wedge Th$.
    Donc $P_j \wedge Th$ admet un modèle qui satisfait $\neg S \wedge \neg r \wedge \neg Q$. Ceci contredit $P_j \wedge Th$ confirme $S \vee r \vee Q$.
    Donc J satisfait S, donc J satisfait $S \vee Q$.

Donc $S \vee Q$ est confirmée, et $S \vee r \vee Q$ n'est pas une généralisation. #

Considérons le rapport d'observation $E =_d ((appellation = Sauternes) \wedge (région = bordeaux) \wedge (couleur = blanc)) \vee ((appellation = CôteRoannaise) \wedge (région = beaujolais) \wedge (couleur = rouge))$, la clause $C_1 =_d \neg(appellation = Sauternes)$ et sa spécialisation $C_2 =_d \neg(appellation = Sauternes) \wedge \neg(région = bordeaux)$ sont vérifiées par exactement les mêmes individus : $M(C_1) = M(C_2) = CôteRoannaise$. La clause C_2 et ses descendants ne peuvent pas être des généralisations. Ainsi, la clause $\neg(appelation = Sauterne) \wedge \neg(région = bordeaux) \vee (couleur = blanc)$ est confirmée par E, mais est impliquée strictement par la généralisation $\neg(appelation = Sauterne) \vee (couleur = blanc)$.

Ces propriétés permettent d'élaguer l'espace de recherche au fur et à mesure de la recherche des généralisations. L'arbre d'énumération d'ensembles est donc un outil bien adapté à la recherche des généralisations confirmatoires et, à la différence de la plupart des algorithmes de calcul des premiers impliqués, il autorise une recherche *anytime* : il est possible d'interrompre

l'algorithme de recherche avant la fin de son exécution et d'obtenir quand même des résultats exploitables, ici un sous-ensemble des généralisations confirmatoires de E.

Dans une certaine mesure, un tel algorithme peut être considéré comme une spécialisation au cadre propositionnel de l'algorithme donné au paragraphe 2.4.3. Cependant, l'exploration dans le cadre propositionnel s'effectue sans redondance.

3.4 Résumé et conclusion

Nous avons montré dans ce chapitre comment le modèle de généralisation que nous avons proposé dans le chapitre précédent se spécialise dans un cadre attribut-valeur. Nous avons dans un premier temps rappelé ce qu'est un langage attribut-valeur, avant de montrer comment les observations, la théorie du domaine et les hypothèses sont représentées dans ce cadre. Les individus n'apparaissant pas explicitement, une formule est associée à un individu quand elle apparaît dans une observation et elle est implicitement universellement quantifiée quand elle appartient à la théorie du domaine ou à une hypothèse. En conséquence, un rapport d'observation ne se représente pas de la même façon dans un langage d'attribut-valeur que dans un langage de la logique des prédicats. Par exemple, une bouteille de vin rouge et une bouteille de vin blanc sont représentées par la formule du premier ordre $rouge(a) \wedge blanc(b)$ ou par la formule propositionnelle $(couleur = rouge) \vee (couleur = blanc)$. La conjonction des descriptions d'individus au premier ordre est remplacée par la disjonction de ces descriptions. Les formules de la théorie du domaine et les hypothèses sont implicitement universellement quantifiées. $\forall X, \neg Champagne(X) \vee pétillant(X)$ est représenté par $\neg(appelation = Champagne) \vee (pétillant = Vrai)$. La relation de confirmation se réduit à la déduction logique. Par exemple, $(couleur = rouge) \vee (couleur = blanc) \models (couleur = rouge) \vee (couleur = blanc)$, c'est-à-dire $(couleur = rouge) \vee (couleur = blanc)$ confirme $(couleur = rouge) \vee (couleur = blanc)$. Cette relation, qui peut paraître triviale, signifie que l'observation d'une bouteille de vin rouge et d'une bouteille de vin blanc confirme que tous les vins sont blancs ou rouges, $rouge(a) \wedge blanc(b) \mid\sim \forall X, rouge(X) \vee blanc(X)$, ce qui est moins évident. Puisque les hypothèses sont considérées implicitement comme universellement quantifiées, une induction est réalisée : elle garantit le caractère ampliatif de l'inférence. Seul le fait d'accepter comme applicable aux individus inconnus les lois vérifiées par les individus connus (explicitées déductivement) est inductif.

Ici le principe de complétion visant à capturer l'hypothèse des similarités s'exprime dans le méta-langage de la logique (alors qu'il s'exprime dans la logique par l'hypothèse de fermeture du domaine dans le modèle présenté au chapitre précédent). Dans les deux cas, l'inférence réalisée pour généraliser est d'abord déductive. Ainsi, les généralisations confirmatoires sont les premiers impliqués des observations. Nous avons montré que les généralisations, au sens des premiers impliqués, d'observations représentées dans un langage d'attributs-valeurs sont bien les mêmes, au changement de représentation près, que celles produites par la relation de généralisation $\succcurlyeq$ donnée au chapitre précédent. L'utilisation d'une représentation attribut-valeur, quand elle est adaptée à la description du problème traité, permet donc de bénéficier des nombreuses techniques existantes de calcul des premiers impliqués. Nous avons décrit, en particulier, une de ces techniques : les arbres d'énumération d'ensembles. Nous avons montré que ces arbres sont bien adaptés à la recherche des généralisations par « generate-and-test ». En effet, une recherche en largeur d'abord permet de nombreux élagages dans l'espace de recherche. Dès qu'un nœud est impliqué par une généralisation, ou dès qu'il n'est pas satisfait par plus d'individus que son père, nous avons montré que ni lui, ni ses descendants ne peuvent être des

généralisations.

Deuxième partie

Classification

Chapitre 4

Apprentissage à partir d'exemples

La première partie de nos travaux concernait l'apprentissage inductif en général visant à mettre en évidence des lois à partir d'observations. On ne s'intéresse toutefois pas toujours à toutes les lois vérifiées par les observations mais seulement à celles qui sont caractéristiques d'une propriété des individus. Dans la classification, cette propriété est appelée l'attribut de classe. Les observations sont des exemples d'associations entre la description d'individus et la valeur prise par l'attribut de classe. La valeur de cet attribut de classe désigne la classe de l'exemple. Le rapport d'observations est donc constitué d'exemples d'individus appartenant à telle ou telle classe. On parle d'apprentissage à partir d'exemples, ou de classification.

Le cas le plus simple d'apprentissage à partir d'exemples est l'apprentissage par cœur : il nécessite de connaître la classe associée à tous les individus possibles. Ce cas est exceptionnel. La plupart du temps, il faut inférer la classe des nouveaux individus à partir de celle des individus connus.

La généralisation confirmatoire est une technique pour inférer des règles de classification. Un des objectifs de notre travail est d'estimer dans quelle mesure le modèle présenté dans les chapitres précédents peut permettre d'inférer de telles règles, et, plus globalement, de faire le lien entre une approche logique de la généralisation et la classification symbolique. Nous précisons dans ce chapitre la problématique de la classification. Puis nous brossons un bref panorama des techniques usuelles de classification, en particulier les techniques regroupées sous l'appellation « apprentissage automatique ». Tout apprentissage inductif a recours à des biais. Nous rappelons les principaux biais utilisés. Nous montrons que les biais logiques utilisés suffisent rarement à eux seuls et que les systèmes d'apprentissage doivent recourir à des biais statistiques additionnels.

4.1 La problématique de la classification

La problématique de l'apprentissage à partir d'exemples est l'apprentissage d'une fonction de classement à partir d'individus dont la classe est connue. Le terme classification est un anglicisme, le problème considéré consiste au sens commun à un problème de classement. Cette problématique a donné lieu à de nombreux travaux. En effet, les domaines d'application sont nombreux. Il peut s'agir de suggérer un diagnostic lorsque l'on considère des patients décrits par un certain nombre de symptômes. La reconnaissance de caractères à partir d'images de caractères manuscrits décrits par des valeurs de pixel est un autre exemple de domaine d'application de la classification. Le but est de prédire la classe des individus ou objets rencontrés, qu'il s'agisse de maladie ou de caractère, à partir de leur description. Les problèmes de diag-

nostics médicaux sont des problèmes fréquemment soumis à des techniques de classification automatique. L'exemple canonique que nous considérons dans ce chapitre et dans les chapitres suivants est d'ailleurs un problème de ce type. Nous considérons en effet à titre d'exemple le problème de quatre patients souffrant ou non d'une maladie x. La classe désigne le diagnostic : le patient souffre de la maladie ($x = Vrai$) ou ne souffre pas de la maladie ($x = Faux$). Supposons que les classes de trois patients soient connues. Le problème de classification que nous considérons consiste à déterminer la classe associée à un quatrième patient.

Dans ce chapitre, nous ne considérons que des objets décrits par un ensemble d'attributs-valeurs (les mêmes attributs pour tous les objets). Chaque patient est décrit par les résultats des tests à valeur booléenne a, b, c et d. Le problème des patients est donc décrit par :

Patient	a	b	c	d	x
e_1	Vrai	Vrai	Vrai	Vrai	Vrai
e_2	Faux	Faux	Faux	Faux	Faux
e_3	Vrai	Faux	Faux	Faux	Vrai
o	Vrai	Vrai	Vrai	Faux	?

Ainsi la description du patient e_1 est $(a = Vrai) \wedge (b = Vrai) \wedge (c = Vrai) \wedge (d = Vrai)$. e_1 souffre de la maladie x.

Définition 4.1 *L'UNIVERS des individus possibles est l'espace produit* $Dom_1 \times Dom_2 \times \ldots \times Dom_d$, *où* Dom_A *est l'ensemble des valeurs possibles pour l'attribut A.*

Dans le cas des patients, chaque domaine est simplement Vrai et Faux.

L'objectif de l'apprentissage est de construire une fonction de classement s'appliquant à tout l'*UNIVERS* à l'aide uniquement de la connaissance d'exemples, c'est-à-dire d'objets dont la classe est connue.

Définition 4.2 *Un* exemple *ou* instance *est un couple* (*description*, *classe*). *On note* $classe(e)$ *la valeur de la classe d'un exemple e.*

Si la valeur associée à chaque attribut est connue, la description *d'un exemple est* complète. *S'il y a des valeurs manquantes, sa description est* partielle *ou* incomplète.

Deux exemples complètement décrits sont incohérents entre eux *s'ils partagent la même description mais appartiennent à des classes différentes*

Ainsi, $e_1 =_d ((a = Vrai) \wedge (b = Vrai) \wedge (c = Vrai) \wedge (d = Vrai), (x = Vrai))$ est un exemple. x est l'attribut de classe. Ainsi $classe(e_1) = Vrai$. Un exemple e_5 peut avoir pour description $(a = Vrai) \wedge (b = Faux) \wedge (c = ?) \wedge (d = Faux)$. La valeur de l'attribut c est manquante. La description de e_5 est, dans ce cas, également notée : $(a = Vrai) \wedge (b = Faux) \wedge (d = Faux)$.

Définition 4.3 *Un* problème de classification *consiste en la donnée d'un ensemble d'exemples, appelé* ensemble d'apprentissage, *noté EA. Son cardinal, noté* $N =_d |EA|$, *est un paramètre important de la complexité de l'apprentissage.*

Un ensemble d'apprentissage est cohérent *si et seulement si il ne contient pas d'exemples incohérents entre eux.*

Dans le cas des patients, l'ensemble d'apprentissage est $EA =_d \{e_1, e_2, e_3\}$, de cardinal 3. Cet ensemble d'apprentissage est cohérent.

Définition 4.4 *La fonction de classement obtenue par apprentissage s'appelle un* classifieur. *Un classifieur est* cohérent *sur l'ensemble d'apprentissage si et seulement si il associe à chaque exemple e sa classe : classe(e). La catégorisation induite par le classifieur (c'est-à-dire l'association description/classe pour tous les objets réalisée par le classifieur) doit être aussi proche que possible de la catégorisation réelle, ou* fonction cible *(qui est inconnue). Plus le nombre d'objets de l'Univers correctement classés est grand, meilleur est le classifieur. On évalue souvent la performance d'un classifieur à l'aide d'un* ensemble de test *constitué d'exemples dont il faut essayer de retrouver la classe (celle-ci étant connue).*

En plus de permettre de classer de nouveaux individus, un classifieur doit autant que possible permettre de mieux comprendre le domaine considéré en offrant une explication des classifications effectuées, par exemple sous la forme de règles générales régissant le domaine.

Un problème de classification ne comprend qu'un nombre fini de classes. Il ne s'agit pas d'interpoler la valeur d'une fonction continue. L'apprentissage de concept est un cas particulier de classification où il n'y a que deux classes et où les exemples sont des exemples positifs et des exemples négatifs du concept à apprendre.

4.2 Panorama des techniques d'apprentissage automatique appliquées à la classification

De nombreuses techniques peuvent être utilisées pour construire un classifieur. Les techniques de classification proviennent essentiellement des statistiques et de l'analyse de données (analyse discriminate [Saporta, 1990]) ou de travaux effectués en intelligence artificielle. Tous les paradigmes de l'apprentissage automatique ont ainsi été utilisés :

- L'apprentissage à base de règles est une des approches les plus répandues. Les algorithmes d'induction de règles tels que AQ [Michalski, 1983], ou les arbres de décision [Hunt *et al.*, 1966; Breiman *et al.*, 1984], dont le célèbre ID3 [Quinlan, 1986], sont des approches à base de règles.

- L'apprentissage à base d'instances [Aha *et al.*, 1991] est également très utilisé à travers un grand nombre de variantes des k-plus-proches voisins [Cover et Hart, 1967].

- Les approches neuromimétiques sont bien connues au travers du perceptron multi-couches et de son mécanisme d'apprentissage,la rétro-propagation du gradient [Rumelhart *et al.*, 1986].

- Les algorithmes génétiques ont servi à construire des classifieurs [Booker *et al.*, 1989].

- Des approches statistiques telles que les classifieurs bayésiens [Buntine, 1989] appartiennent également aux techniques regroupées sous l'appellation apprentissage automatique.

Toutes ces techniques sont décrites, par exemple, dans [Mitchell, 1997]. Dans cette thèse, nous ne considérons que les approches à base d'instances et les approches à base de règles.

4.2.1 Approches à base d'instances

Le plus proche voisin [Cover et Hart, 1967] est la technique la plus connue de l'apprentissage à base d'instances. Dans les cas les plus simples, l'apprentissage en lui-même ne consiste qu'à stocker les exemples observés. Un nouvel objet est classé en fonction de sa similarité avec les exemples enregistrés. Les exemples stockés sont appelés des instances. Dans le cas du plus proche voisin, c'est la classe de l'exemple le plus proche qui est attribuée. La proximité est mesurée à l'aide d'une similarité ou inversement d'une distance. Des distances simples, comme la distance *euclidienne* ou la distance *de Manhattan (city-bloc)* dans le cas d'attributs linéaires, ou la distance de *Hamming* dans le cas d'attributs nominaux, peuvent être utilisées. Rappelons quelques définitions de distances dans un espace d'attributs nominaux ou d'attributs linéaires.

Définition 4.5 *La distance de Hamming D_H est définie dans un espace d'attributs nominaux. La distance $D_H(o_1, o_2)$ entre deux individus o_1 et o_2 est le nombre d'attributs ne prenant pas la même valeur dans o_1 et o_2, c'est-à-dire $|\{a/v_a^1 \neq v_a^2\}|$. v_a^1 et v_a^2 sont les valeurs de l'attribut a dans les descriptions de o_1 et o_2.*

Les distances les plus courantes dans un espace d'attributs linéaires sont modélisées par la distance de Minkowski *:*

$$L_q(o_1, o_2) = \sqrt[q]{\Sigma_a (v_a^1 - v_a^2)^q}$$

L_2 est la distance euclidienne. On distingue habituellement deux autres cas particuliers, L_1 et L_∞.

La distance de Manhattan, *ou* city-bloc, *est la somme des valeurs absolues :*

$$L_1(o_1, o_2) = \Sigma_a |v_a^1 - v_a^2|$$

Le maximum des écarts absolus définit également une distance :

$$L_\infty(o_1, o_2) = max_a |v_a^1 - v_a^2|$$

La performance des techniques à base d'instances dépend crucialement de la distance utilisée. On peut remarquer dans le cas de la distance euclidienne, ou plus généralement des distances de Minkowski, que les attributs dont les valeurs sont les plus dispersées sont défavorisés car leur poids dans le résultat est plus grand que celui d'attributs dont les valeurs sont regroupées. Il est donc courant de normaliser la différence sur chaque attribut par l'écart maximum possible :

$$\frac{|v_a^1 - v_a^2|}{v_a^{max} - v_a^{min}}.$$

La distance de Hamming n'est pas non plus très précise. La mesure de différence des valeurs, en anglais *value difference metric* (VDM) [Stanfill et Waltz, 1986], consiste à considérer que deux valeurs sont similaires si elles sont associées à la même classe. Cette mesure n'étant pas symétrique, une mesure simplifiée peut être utilisée (SVDM) [Domingos, 1996] :

$$\Sigma_{i=1}^{c} |P(c_i|v_a^1) - P(c_i|v_a^2)|$$

où c est le nombre de classes, c_i la i-ième classe et $P(c_i|v_a^1)$ la probabilité conditionnelle d'observer la classe c_i sachant que l'attribut a prend la valeur v_a^1.

Ces approches sont très sensibles à la présence d'attributs non pertinents. La contribution de ces derniers à la distance mesurée peut aisément masquer l'information apportée par les attributs pertinents. Une solution consiste, par exemple, à pondérer les attributs comme dans [Cost et Salzberg, 1993]. La mesure de différence des valeurs contribue également à résoudre ce problème dans le cas d'attributs nominaux. Elle tend à associer un écart nul entre les valeurs d' attributs non pertinents puisque par nature dans le cas d'un attribut a non pertinent la

probabilité $P(c_i|v_a^1)$ ne dépend pas de la valeur v_a^1 considérée et est donc la même pour toutes les valeurs de a si le nombre d'exemples est suffisamment grand pour estimer correctement les probabilités conditionnelles.

Un autre problème des approches à base d'instances est leur sensibilité au bruit. Les instances incorrectes peuvent conduire à mal classer les objets qui appartiennent à leur voisinage. De nombreuses méthodes ont été introduites pour prendre en compte ce problème. Ainsi l'algorithme IB3 [Aha *et al.*, 1991] ne conserve que les instances estimées fiables. La fiabilité est évaluée lors d'un apprentissage incrémental en utilisant l'ensemble courant d'instances pour classer le nouvel exemple considéré et en mesurant le rapport du nombre de fois où les instances ont été utilisées pour classer un objet de même classe sur le nombre total de fois où elles ont été utilisées. Quand un certain nombre d'exemples a été considéré, les instances de faible fiabilité sont éliminées. Le système PEBLS [Cost et Salzberg, 1993] utilise une autre méthode qui consiste à pondérer les exemples en fonction de leur fiabilité afin que les distances associée aux exemples incorrects soient accrues.

Une extension classique du cadre du plus proche voisin que nous avons considéré jusqu'à présent consiste à utiliser les k plus proches voisins [Duda et Hart, 1973] et à retenir la classe majoritaire, le vote de chaque voisin décroissant avec sa distance à l'objet à classer.

4.2.2 Approches à base de règles

Nous regroupons dans cette catégorie les techniques d'induction de règles et celles d'induction d'arbres de décision. Considérons tout d'abord les techniques d'induction de règles et commençons par quelques définitions.

Une règle est composée d'une partie gauche, les prémisses (conjonction finie d'atomes), et d'une partie droite, la conclusion (valeur de l'attribut de classe). Un objet est couvert par une règle s'il satisfait sa partie gauche. Un exemple couvert par une règle mais qui ne satisfait pas la partie droite est un contre-exemple.

Définition 4.6 *Une* règle *est un couple noté* $G \rightarrow (x = v_x)$, *où* G *est une conjonction finie de couples attribut-valeur.*

Une règle $G \rightarrow (x = v_x)$ classe *(ou* couvre*) un objet* o *si et seulement si* $o \models G$. *Elle est* satisfaite *par un exemple* e *si et seulement si elle classe* e *et* $(classe(e) = v_x)$.

Soient R *une règle et* e *un exemple de l'ensemble d'apprentissage.* e *est un* contre-exemple *de* R *si et seulement si* R *couvre* e *mais* R *n'est pas satisfaite par* e.

Typiquement, les algorithmes d'induction de règles construisent une règle couvrant un maximum d'exemples d'une classe en couvrant peu de contre-exemples. Les exemples couverts sont mis de côté et le processus de génération d'une règle reprend avec les exemples restants jusqu'à ce que tous les exemples soient couverts.

La recherche de la meilleure règle en fonction de l'ensemble d'exemples courant peut se faire grâce à une recherche en faisceau, en anglais *beam-search*, comme dans CN2 [Clark et Niblett, 1989]. Dans ce cas, les b meilleures parties gauches sont conservées au lieu d'une seule lors de la construction de la meilleure règle. La qualité des règles candidates est évaluée à l'aide d'une heuristique H. Clairement, il faut que H augmente avec le nombre d'exemples de la classe considérée couverts et il faut que H diminue avec le nombre de contre-exemples. Les algorithmes de la famille AQ [Michalski, 1969] utilisent la précision apparente, c'est-à-dire celle mesurée sur l'ensemble des exemples :

$$\frac{n_e}{n_e + n_{ce}}$$

où n_e est le nombre d'exemples de la classe considérée et n_{ce} est le nombre de contre-exemples. La version initiale de CN2 utilise l'entropie introduite dans le cadre des arbres de décision [Quinlan, 1986]:

$$-\frac{n_e}{n_e+n_{ce}} ln(\frac{n_e}{n_e+n_{ce}}) - \frac{n_{ce}}{n_e+n_{ce}} ln(\frac{n_{ce}}{n_e+n_{ce}}).$$

Le problème de ces mesures est qu'elles tendent à favoriser des règles trop spécifiques. Ainsi une version plus récente de CN2 [Clark et Boswell, 1991] utilise la précision de Laplace:

$$\frac{n_e+1}{n_e+n_{ce+c}}$$

où c est le nombre de classes. Cette mesure tend vers la précision apparente quand la règle couvre beaucoup d'exemples, et vers $1/c$ quand elle en couvre peu.

La classification de nouveaux objets se fait en cherchant les règles qui les couvrent. S'il n'y a qu'une règle, sa partie droite indique la classe de l'objet. S'il n'y en a aucune, on utilise généralement une règle par défaut qui associe, par exemple, la classe la plus représentée parmi les exemples. Si plusieurs règles de conclusions distinctes couvrent l'objet à classer, une solution consiste à choisir la classe majoritaire parmi ces règles.

Les arbres de décision [Hunt *et al.*, 1966] sont construits récursivement. Étant donné l'ensemble d'exemples courant, ils utilisent une heuristique, par exemple l'entropie dans ID3 [Quinlan, 1986], pour chercher l'attribut le plus discriminant. L'entropie tendant à privilégier les attributs prenant un grand nombre de valeurs par rapport à ceux ayant peu de valeurs, d'autres critères ont été proposés, par exemple le ratio du gain d'information dans C4.5 [Quinlan, 1993a] ou GINI dans CART [Breiman *et al.*, 1984]. Un nouveau noeud est créé et étiqueté par l'attribut le plus discriminant. Ses fils sont les arbres de décision construits pour chacun des ensembles d'exemples correspondant aux différentes valeurs de l'attribut.

La génération de l'arbre de décision s'arrête idéalement quand les exemples considérés appartiennent à la même classe. Une feuille est alors créée et elle est étiquetée par cette classe. Chaque branche de l'arbre de décision depuis la racine jusqu'à une feuille correspond donc à une règle, c'est pourquoi nous avons regroupé les arbres de décision et les techniques d'induction de règles sous la dénomination générique d'approches à base de règles.

En présence de bruit et donc d'exemples incorrects, la génération de l'arbre peut s'arrêter quand le nombre de contre-exemples de la classe majoritairement représentée est suffisamment faible plutôt que nul. Cet élagage peut se faire pendant la génération ou après génération d'un arbre complet. Par exemple, dans C4.5 [Quinlan, 1993a], l'élagage se fait *a posteriori* en partant des feuilles, en remontant d'un niveau après chaque élagage et en comparant le taux d'erreur estimé du sous-arbre courant par rapport à celui consitué d'une seule feuille étiquetée par la classe majoritaire des exemples couverts à ce niveau de l'arbre. Les techniques d'induction de règles prennent en compte également la présence de bruit en acceptant les règles couvrant un petit nombre de contre-exemples.

4.3 Biais

Toutes les techniques d'apprentissage automatique reposent sur un ou plusieurs biais inductifs. Tom Mitchell [Mitchell, 1982] définit les *biais* (ou *critères de préférence*) comme « *toute base pour choisir une généralisation plutôt qu'une autre, hormis la stricte cohérence sur l'ensemble d'apprentissage.* »[1] Paul Utgoff [Utgoff, 1986] définit quant à lui les biais de la façon

1. Notons que la cohérence sur l'ensemble d'apprentissage est déjà un biais.

suivante : « *Tous les facteurs, à part les exemples et contre-exemples du concept à apprendre, qui influencent la sélection de l'hypothèse constituent des biais.* » Claire Nédellec, Céline Rouveirol, Hilde Adé, Francesco Bergadano et Birgit Tausend [Nédellec *et al.*, 1996] ont proposé la typologie des biais suivante à partir de la définition de Paul Utgoff :

Le biais de langage. Il définit le langage de description de la fonction cible, en déterminant l'espace des hypothèses possibles décrivant le classement à apprendre. Il restreint le langage de description des concepts valable pour tout apprentissage supervisé. Le biais de langage inclut toute restriction de langage qui est spécifique au concept à apprendre. Prenons l'exemple de la fonction cible $a > b$, où a et b sont deux attributs linéaires continus de domaine $[0; 1]$ et supposons que nous utilisions un classifieur à base de règles. Il est impossible d'écrire un ensemble fini de règles $(a = v_a) \wedge (b = v_b) \rightarrow (x = Vrai)$ décrivant la fonction cible $a > b$.

Le biais de recherche. Il détermine quelle partie de l'espace des hypothèses est explorée. Ce peut être un biais de *préférence* ou de *restriction*. Un biais de restriction détermine quelles hypothèses doivent être ignorées, alors qu'un biais de préférence indique celles qui doivent être considérées en premier. C'est donc une généralisation du critère de préférence évoqué plus haut.

Le biais de validation. Il détermine un critère d'arrêt du système d'apprentissage. Cela peut être lorsque l'hypothèse couvre un nombre suffisant d'exemples (complétude) ou lorsqu'elle ne couvre plus trop de contre-exemples (cohérence).

Ainsi, on impose souvent que le classifieur construit soit cohérent sur l'ensemble d'apprentissage, c'est-à-dire qu'il classe correctement les exemples. Cette contrainte de cohérence peut néanmoins être relâchée comme nous le verrons par la suite. Mais conservons cette hypothèse pour l'instant. Un grand nombre de classifieurs peuvent être cohérents sur un ensemble d'apprentissage EA donné. Considérons le cas d'objets décrits par des attributs booléens, dont les descriptions sont complètes et pour lesquels on cherche à prédire l'appartenance à une classe booléenne. Si l'ensemble d'apprentissage EA ne contient pas d'exemples redondants, le nombre de classifieurs possibles cohérents sur EA est $2^{(2^d - |EA|)}$ où $|EA|$ est le nombre d'éléments dans l'ensemble d'apprentissage EA et 2^d est le cardinal de l'$UNIVERS$. Choisir le « bon » classifieur implique d'utiliser des biais supplémentaires.

Ceux-ci peuvent être numériques ou symboliques. Un biais numérique simple consiste à déterminer la classe majoritairement représentée dans l'ensemble d'apprentissage, et à construire le classifieur qui retourne la classe majoritaire dans tous les cas. Remarquons que ce classifieur par défaut n'est pas cohérent avec l'ensemble d'apprentissage si l'ensemble d'apprentissage contient plus d'une classe, ce qui est « usuel » dans un problème de classification. Un premier biais logique consiste justement en la contrainte de cohérence du classifieur avec l'ensemble d'apprentissage. Un autre exemple de biais logique utilisé dans plusieurs approches à base de règles consiste à ne retenir que les règles les plus générales. Ce dernier biais est connu sous le nom de *rasoir d'Occam*. C'est l'adaptation d'un principe énoncé par Guillaume d'Ockham, philosophe anglais du $14^{ème}$ siècle : « *Il ne faut pas multiplier les essences sans nécessité* ». L'hypothèse des similarités, *les individus inconnus se comportent comme les individus connus*, est évidemment un biais. C'est le biais *inductif*. Il est, selon les approches, essentiellement logique ou essentiellement numérique. Ce sont deux interprétations de l'hypothèse des similarités qui fondent les approches à base d'instances et les approches à base de règles. L'interprétation de l'hypothèse des similarités utilisée dans les approches à base d'instances est que les individus inconnus appartiennent à la même classe que leurs plus proches voisins. La proximité

entre individus est le plus souvent évaluée à l'aide de distances. Le biais inductif s'exprime numériquement en général.

Les approches à base de règles visent à produire des règles dont la conclusion exprime l'appartenance à une classe. Leur interprétation de l'hypothèse de similarité est que les individus inconnus vérifient les règles satisfaites par les individus connus. Le chapitre suivant sera consacré à ces approches. Les biais utilisés pour la construction des règles sont, en général, purement logiques : la cohérence, la pertinence et la généralité sont des biais logiques. Certaines techniques comme les arbres de décision [Quinlan, 1986; Quinlan, 1993a] ont recours à des biais statistiques, par exemple le gain d'information, pour ne construire que les règles les plus discriminantes. Le gain d'information est un biais numérique qui guide la recherche de la solution. Celle-ci peut être lue comme un ensemble de règles, donc interprétée logiquement.

Même lorsque l'on n'utilise qu'un biais logique, les règles s'appliquant à un nouvel individu peuvent avoir des conclusions contradictoires. Un biais logique ne permet alors plus de trancher entre des hypothèses qui sont de même valeur de son point de vue, puisqu'il ne « connaît » que les valeurs Vrai ou Faux. L'obtention d'un classifieur qui associe au plus une classe à chaque individu nécessite d'avoir recours finalement à un biais statistique pour choisir une seule classe parmi celles proposées. Ce biais statistique est appelé *critère de résolution*. Un critère de résolution est également nécessaire dans une approche à base d'instances lorsque les k plus proches voisins qu'elle retient ne sont pas tous de la même classe.

4.4 Non-déterminisme de l'association description/classe

Les biais sont, en général, fixés lors du choix de la technique de construction du classifieur. Cependant, le choix du langage de description des exemples est parfois imposé par le problème de classification. Malheureusement, l'association description/classe n'est pas nécessairement déterministe. En effet, alors que chaque individu n'appartient qu'à une seule classe, une même description peut correspondre à plusieurs exemples appartenant à des classes différentes. Le lien entre description et classe n'est donc pas nécessairement fonctionnel ou déterministe. En pratique, il l'est même rarement, car généralement la description utilisée est incomplète ou partiellement erronée.

Pour pallier la présence de bruit ou l'incomplétude des descriptions et obtenir un classifieur performant, on peut lever la contrainte de cohérence sur EA car de bons résultats sur EA n'entraînent pas nécessairement de bons résultats sur les individus inconnus : ce phénomène est appelé *sur-adaptation*. Il faut pourtant bien apprendre à partir de quelque chose. Dans la suite, nous considérons le cas d'associations description/classe déterministes, avant d'indiquer comment prendre en compte le cas d'associations non déterministes.

Nous supposons en conséquence, dans les chapitres suivants, que les éléments de l'ensemble d'apprentissage sont cohérents entre eux, c'est-à-dire qu'une classe unique est associée à une description complète donnée.

4.5 Résumé et conclusion

Dans ce chapitre, nous avons défini l'apprentissage à partir d'exemples, ou classification. Un problème de classification consiste en un ensemble d'exemples déjà classés à partir desquels il faut construire une fonction de classement, appelée classifieur, qui associe à chaque description une classe au plus. Nous avons rappelé les différentes techniques d'apprentissage automatique qui s'appliquent à ce problème. Nous avons rappelé que toute technique de construction d'un

classifieur utilise des biais. Les biais de langage comprennent le choix de la représentation des exemples et du langage de représentation des hypothèses. Le choix de la représentation des exemples est souvent imposé, et peut être mal approprié et entraîner de ce fait un non-déterminisme de l'association entre une description et une classe. Le second type de biais regroupe les biais de recherche et de validation. Ces biais comprennent le biais inductif, l'hypothèse des similarités, et le critère de résolution permettant de choisir entre plusieurs classes proposées en se plaçant sous l'hypothèse des similarités. L'hypothèse de similarité peut être interprétée par un biais numérique ou par un biais logique. Il arrive parfois qu'un biais statistique soit associé à un biais logique, comme dans les arbres de décision où le gain d'information vise à obtenir des règles les plus générales possibles.

Chapitre 5

Classification à base de règles cohérentes et pertinentes

La classification à base de règles cohérentes et pertinentes est une technique d'apprentissage fortement liée à la généralisation confirmatoire. Dans ce chapitre, nous définissons précisément les notions de règle cohérente et pertinente, avant d'en donner quelques propriétés. Nous explicitons ensuite les relations existant entre l'inférence de règles cohérentes et pertinentes et le modèle d'induction confirmatoire que nous avons présenté auparavant. Nous précisons, en particulier, les liens existant entre règles cohérentes et pertinentes et clauses. Nous proposons enfin deux sémantiques pour les objets incomplètement décrits.

5.1 Règles cohérentes et pertinentes

Intuitivement, une règle est cohérente avec un ensemble d'exemples si chaque fois qu'un exemple vérifie sa partie gauche, il satisfait également sa partie droite. Pour éviter de considérer des règles inutiles, on requiert qu'une règle soit satisfaite par au moins un exemple, c'est-à-dire qu'au moins un exemple satisfasse à la fois ses prémisses et sa conclusion. Formellement :

Définition 5.1 *Une* règle cohérente et pertinente *pour un ensemble d'apprentissage EA est un couple noté $G \rightarrow (x = v_x)$ satisfaisant les deux conditions suivantes :*

Cohérence : *pour tout exemple e de EA, si $e \models G$ alors $classe(e) = v_x$.*

Pertinence : *il existe un exemple e de EA tel que $e \models G$.*

Une règle pertinente est non-triviale *si et seulement si elle couvre plus d'un objet.*

Deux règles $R_1 =_d G_1 \rightarrow (x = v_1)$ et $R_2 =_d G_2 \rightarrow (x = v_2)$ sont contradictoires si et seulement si il existe un objet o classé par R_1 et R_2 et $v_1 \neq v_2$.

Considérons à nouveau l'exemple des patients du chapitre précédent. La règle $R =_d (a = Vrai) \rightarrow (x = Vrai)$ est pertinente pour l'ensemble d'apprentissage EA puisqu'elle est satisfaite par e_1. Elle est cohérente pour EA puisque le seul contre-exemple possible e_2 n'est pas couvert par R. Elle est non-triviale puisqu'elle couvre plus d'un objet, par exemple e_1 et l'objet représenté par $(a = Vrai), (b = Vrai), (c = Faux), (d = Faux)$.

Ainsi, une règle R est cohérente pour EA si et seulement si aucun exemple de EA n'est un contre-exemple de R. Une règle pertinente non-triviale est satisfaite par plus d'un exemple. Les

règles pertinentes et triviales ne servent à rien (elles ne permettent pas de classer de nouveaux objets). Il n'est toutefois pas forcément possible de les éviter : la modélisation sous forme de règles d'un problème de classification comme la fonction parité (les individus sont décrits par un ensemble d'attributs booléens et la classe indique si la somme du nombre d'attributs valués par Vrai est paire) nécessite une règle par description. La classe de chacun des « voisins » (au sens de la distance de Hamming) d'un objet o est de classe opposée à celle de o et donc une règle ne peut couvrir plus d'un objet.

Toute règle R cohérente et pertinente pour EA peut être considérée comme confirmée au sens du critère de Nicod. En effet, si R est cohérente pour EA, elle est compatible avec le rapport d'observation (dans un certain sens). La propriété de pertinence assure qu'il existe un individu satisfaisant les parties gauche et droite de R.

Une règle minimale cohérente et pertinente est une règle cohérente et pertinente qui n'est pas impliquée par d'autres règles cohérentes et pertinentes.

Définition 5.2 *Une règle $G \rightarrow (x = v_x)$ cohérente et pertinente pour EA est* minimale *si et seulement si quel que soit G' tels que $G \models G'$ et $G' \not\models G$, la règle $G' \rightarrow (x = v_x)$ n'est pas une règle cohérente et pertinente pour EA.*

Dans la définition de Michalski [Michalski, 1983], une règle cohérente est dite discrimante et les règles minimales cohérentes sont donc ce que Michalski appelle des règles maximalement discriminantes.

Une règle définit une partition de l'$UNIVERS$ en deux classes : l'ensemble des individus (ou objets) satisfaisant sa partie gauche et les autres. La partie gauche d'une règle peut être vue comme un ensemble d'attributs valués, comme une région de l'$UNIVERS$ ou encore comme une conjonction de conditions.

Intuitivement, un classifieur complet et cohérent associe une classe unique à chaque individu possible. Comme nous l'avons évoqué au chapitre précédent, le biais logique ne suggère pas toujours une classe unique pour un individu donné. Il arrive qu'aucune hypothèse, dans le cas présent, aucune règle, ne s'applique, ou, au contraire que des hypothèses contradictoires s'appliquent.

Formellement, pour pallier ce problème, il suffit de garantir la complétude et la cohérence de l'ensemble de règles considéré par rapport à tous les individus possibles.

Définition 5.3 *Un ensemble de règles C est* complet *par rapport à un sous-ensemble T de l'$UNIVERS$ si et seulement si $\forall t \in T$, il existe une règle $R =_d G \rightarrow (x = v_x)$ appartenant à C telle que R couvre t, c'est-à-dire $t \models G$.*

Un ensemble de règles C est cohérent *par rapport à $T \subseteq UNIVERS$ si et seulement si il ne contient pas de règles contradictoires : $\forall t \in T, R \in C, R' \in C$, $R =_d G \rightarrow (x = v_x)$, $R' =_d G' \rightarrow (x = v'_x)$, si $t \models G$ et $t \models G'$ alors $v_x = v'_x$.*

Pour assurer la cohérence d'un classifieur à base de règles, une voie consiste à le munir d'un critère de résolution. C'est la voie que nous avons retenue pour les classifieurs à base de toutes les règles minimales cohérentes et pertinentes. Pour assurer la complétude d'un classifieur à base de règles, il est possible de choisir la classe d'un objet o au hasard lorsqu'aucune règle ne couvre o. Une autre solution fréquemment utilisé pour garantir la complétude est d'associer la classe majoritairement représentée dans l'ensemble d'apprentissage aux objets couverts par aucune règle.

Un classifieur à base de règles est un ensemble de règles répondant à un problème de classification, c'est-à-dire associant à chaque objet une classe. Comme un même objet peut,

dans le cas général, être couvert par deux règles contradictoires, il est nécessaire de recourir à un critère de résolution pour faire un choix (de classe) lorsque plusieurs classes sont proposées. Formellement :

Définition 5.4 *Un* classifieur à base de règles *est la donnée d'un ensemble fini de règles et d'un critère de résolution.*

Un classifieur à base de règles cohérentes et pertinentes *est la donnée d'un ensemble de règles cohérentes et pertinentes et d'un critère de résolution.*

Un classifieur à base de règles minimales cohérentes et pertinentes est un cas particulier de classifieur à base de règles cohérentes et pertinentes n'utilisant que des règles minimales. Ne considérer que les règles minimales permet d'obtenir une meilleure compréhensibilité du domaine que celle apportée par des règles plus spécifiques. Nous considérons dans la suite les classifieurs à base de toutes les règles minimales cohérentes et pertinentes.

Définition 5.5 *Étant donné un ensemble d'apprentissage EA, $RMIN(EA)$ est l'ensemble de toutes les règles minimales cohérentes et pertinentes pour EA. Un classifieur à base de toutes les règles minimales cohérentes et pertinentes est $RMIN(EA)$ muni d'un critère de résolution.*

Les classifieurs à base de toutes les règles minimales cohérentes et pertinentes représentent un facteur commun à tous les classifieurs à base de toutes les règles cohérentes et pertinentes puisque chaque règle cohérente et pertinente est impliquée par une règle minimale cohérente et pertinente. L'emploi de règles minimales garantit une économie de représentation : s'il existe une règle cohérente et pertinente pour EA qui classe o avec v_x, il existe une règle minimale cohérente et pertinente pour EA qui classe o avec v_x.

Disposer de toutes les règles minimales présente plusieurs intérêts. D'une part, si l'on considère le biais inductif logique traduisant l'hypothèse des similarités, il n'y a aucune raison purement logique pour préférer une règle minimale à une autre. D'autre part, les informations redondantes apportées par l'ensemble des règles permettent d'offrir une meilleure résistance au bruit et aux informations manquantes qu'un ensemble réduit de règles (cf. paragraphe 10.1.1) [Gams, 1989; Nock et Gascuel, 1995].

5.2 Quelques propriétés des règles minimales cohérentes et pertinentes

Illustrons les propriétés de l'ensemble des règles minimales cohérentes et pertinentes pour EA à l'aide du problème de classification suivant : les individus sont décrits à l'aide de deux attributs nominaux a et b pouvant prendre les valeurs 1, 2 ou 3. L'attribut de classe x peut prendre les valeurs α, β, γ ou δ. L'ensemble d'apprentissage est constitué des quatre exemples ci-dessous. L'$UNIVERS$ est représenté sur le tableau de Karnaugh joint.

a	b	x
1	2	α
2	1	β
2	3	γ
3	2	δ

		b		
		1	2	3
a	1		α	
	2	β		γ
	3		δ	

On a $RMIN(EA) =_d \{(a = 1) \rightarrow (x = \alpha), (b = 1) \rightarrow (x = \beta), (b = 3) \rightarrow (x = \gamma), (a = 3) \rightarrow (x = \delta)\}$.

Propriété 5.1 *$RMIN(EA)$ est complet par rapport à tout ensemble d'apprentissage EA cohérent, mais pas nécessairement par rapport à l'$UNIVERS$.*

Preuve : on peut construire à partir de tout exemple $e =_d (description(e), classe(e))$ la règle $description(e) \rightarrow classe(e)$ cohérente et pertinente pour EA (lorsque EA est cohérent). Si cette règle n'est pas minimale alors toute règle minimale qui l'implique couvre e (une telle règle minimale existe toujours car la taille des règles est limitée par le nombre d'attributs.) Donc tout exemple est couvert par une règle de $RMIN(EA)$.
En revanche, $RMIN(EA)$ ne couvre pas nécessairement tout l'$UNIVERS$. Ainsi, dans l'exemple courant, aucune règle minimale ne couvre $\{(a = 2), (b = 2)\}$. #

Propriété 5.2 *$RMIN(EA)$ est cohérent par rapport à tout ensemble d'apprentissage EA mais pas nécessairement par rapport à l'$UNIVERS$.*

Preuve : Dans l'exemple courant, $\{(a = 1), (b = 1)\}$ est classé α et β. #

Dans le cas particulier d'un $UNIVERS$ d'attributs binaires, si les objets et les exemples sont complètement décrits alors tout objet est couvert par une règle minimale. L'attribut de classe peut être multi-valué.

Propriété 5.3 *Dans le cas d'un problème de classification où les objets et les exemples sont complètement décrits par des attributs nominaux ne pouvant prendre que deux valeurs, par exemple des booléens, $RMIN(EA)$ est complet par rapport à l'$UNIVERS$.*

Preuve : nous prouvons la propriété pour les booléens, le cas d'attributs binaires quelconques s'y réduit facilement par un changement de représentation. Nous adoptons une notation propositionnelle pour réduire les expressions : $(p = Vrai)$ et (p = Faux) sont notés p et $\bar{p}$
Soit un objet m. Nous définissons la distance entre deux objets $d(m_1, m_2)$ comme étant le cardinal du nombre d'attributs sur lesquels m_1 et m_2 différent (distance de Hamming).

Soit e de classe x plus proche de m que tout autre exemple o, $d(e, m) \leq d(o, m)$. Il existe toujours un tel objet si EA est non-vide.
Soit $CE(e, EA)$ l'ensemble des exemples de classe différente de celle de e.
Ordonnons les attributs de façon à ce que e et m différent sur les d premiers :
$e = l_1 \wedge \ldots \wedge l_d \wedge l_{d+1} \wedge \ldots \wedge l_n$
$m = \neg l_1 \wedge \ldots \wedge \neg l_d \wedge l_{d+1} \wedge \ldots \wedge l_n$

Montrons que $(l_{d+1} \wedge \ldots \wedge l_n, x)$ est une règle.

- $e \models l_{d+1} \wedge \ldots \wedge l_n$
- Supposons qu'il existe c appartenant à $CE(e, EA)$ tel que $c \models l_{d+1} \wedge \ldots \wedge l_n$,
  Alors $d(c, m) \leq d$. On peut distinguer deux cas :

  1. $d(c, m) < d$: ceci contredit l'hypothèse sur e.
  2. $d(c, m) = d$: $c = e$. Ceci est impossible si les exemples sont cohérents, booléens et complètement décrits.

  Donc pour tout élément c de $CE(e, EA)$, $c \not\models l_{d+1} \wedge \ldots \wedge l_n$.

Donc $(l_{d+1} \wedge \ldots \wedge l_n, x)$ est une règle.

Il faut encore distinguer le cas où ce n'est pas une règle minimale.

1. Soit une règle minimale (g, x) telle que $g \subset l_{d+1} \wedge \ldots \wedge l_n$,
   Alors $m \models g$. Donc (g, x) classe m.
2. $(l_{d+1} \wedge \ldots \wedge l_n, x)$ est une règle minimale qui classe m.

#

Ainsi dans le cas d'une description binaire et d'un ensemble d'apprentissage EA ne contenant que des exemples complètement décrits, tout objet est couvert par au moins une règle minimale cohérente et pertinente pour EA. Toutefois, $RMIN(EA)$ n'est pas pour autant cohérent par rapport à l'$UNIVERS$. Considérons l'exemple décrit ci-dessous :

a	b	x
Vrai	Vrai	α
Faux	Faux	β

	Vrai	Faux
Vrai	α	?
Faux		β

$RMIN(EA)$ contient les règles $(a = Vrai) \rightarrow (x = \alpha)$, $(b = Vrai) \rightarrow (x = \alpha)$, $(a = Faux) \rightarrow (x = \beta)$ et $(b = Faux) \rightarrow (x = \beta)$. L'individu décrit par $\{(a = Vrai), (b = Faux)\}$ est couvert par les deux règles contradictoires $(a = Vrai) \rightarrow (x = \alpha)$ et $(b = Faux) \rightarrow (x = \beta)$.

L'absence de cohérence par rapport à l'$UNIVERS$ justifie l'adjonction d'un critère de résolution aux ensembles de règles minimales cohérentes et pertinentes pour EA pour constituer des classifieurs, c'est-à-dire garantir qu'une seule classe est associée à chaque objet de l'$UNIVERS$.

5.3 L'approche de Ron Rymon

Ron Rymon a montré qu'il existe des relations entre les règles minimales cohérentes et pertinentes et les premiers implicants [Rymon, 1994].

Considérons l'exemple du paragraphe précédent. Nous avons choisi de représenter tout exemple comme une association $(description, classe)$ qui est, comme nous l'avons vu au paragraphe précédent, équivalent à la conjonction de la description et de la classe. L'ensemble d'apprentissage est constitué des exemples $EA = \{((a = 1) \wedge (b = 2) \wedge (x = \alpha)), ((a = 2) \wedge (b = 1) \wedge (x = \beta)), ((a = 2) \wedge (b = 3) \wedge (x = \gamma)), ((a = 3) \wedge (b = 2) \wedge (x = \delta))\}$. Nous avons montré au chapitre 3 que l'ensemble des exemples est une disjonction dans ce cas. Ron Rymon représente chaque exemple $(description, classe)$ sous la forme d'une implication $description \Rightarrow classe$, c'est-à-dire $\neg description \vee classe$. L'ensemble d'apprentissage qu'il considère est donc $EA'' = \{(\neg(a = 1) \vee \neg(b = 2) \vee (x = \alpha)), (\neg(a = 2) \vee \neg(b = 1) \vee (x = \beta)), (\neg(a = 2) \vee \neg(b = 3) \vee (x = \gamma)), (\neg(a = 3) \vee \neg(b = 2) \vee (x = \delta))\}$. L'ensemble des exemples est dans ce cas une conjonction. Nous pouvons remarquer que EA'' n'est pas la négation de EA qui correspondrait à la conjonction des disjonctions $\neg description \vee \neg classe$.

Définissons l'ensemble des formules appartenant à un ensemble et non impliquées par une formule d'un second ensemble :

Définition 5.6 *$q \in E_1$ modulo implication avec E_2 si et seulement si $q \in E_1$ et il n'existe pas $p \in E_2$ tel que $p \models q$.*

Dans le cas d'une classe booléenne et de descriptions complètes, Ron Rymon [Rymon, 1994] a utilisé ce formalisme pour construire $RMIN(EA)$ en établissant une relation entre les règles minimales cohérentes et pertinentes et les premiers implicants.

Propriété 5.4 *Soient EA''^{+} (resp. EA''^{-}) l'ensemble des exemples booléens complètement décrits de classe $x = V$ représentés sous la forme d'implications, et $RMIN^{+}(EA)$ (resp. $RMIN^{-}(EA)$) l'ensemble des règles minimales cohérentes et pertinentes de conséquence $(x = V)$ (resp. $(x = F)$).*
$G \rightarrow (x = V) \in RMIN^{+}$ si et seulement si $G \in PI(EA''^{-})\backslash\{(x = F)\}$ modulo implication avec $PI(EA''^{+})$
$G \rightarrow (x = F) \in RMIN^{-}$ si et seulement si $G \in PI(EA''^{+})\backslash\{(x = V)\}$ modulo implication avec $PI(EA''^{-})$

Considérons l'exemple des patients. $EA''^{+} =_d \{(a = Vrai) \wedge (b = Vrai) \wedge (c = Vrai) \wedge (d = Vrai) \Rightarrow (x = Vrai), (a = Vrai) \wedge (b = Faux) \wedge (c = Faux) \wedge (d = Faux) \Rightarrow (x = Vrai)\}$ et $EA''^{-} =_d \{(a = Faux) \wedge (b = Faux) \wedge (c = Faux) \wedge (d = Faux) \Rightarrow (x = Faux)\}$. La condition $G =_d (a = Vrai)$ est un premier implicant de EA''^{-} différent de $(x = Faux)$ et elle n'est pas impliquée par un premier implicant de E''^{+}, donc l'implication $(a = Vrai) \Rightarrow (x = Vrai)$ est une règle minimale cohérente et pertinente pour EA.

Nous pouvons simplifier la propriété montrée par Ron Rymon et établir par la même occasion une relation entre les ensembles de règles minimales cohérentes et pertinentes et le modèle de généralisation confirmatoire développé au chapitre 3.

5.4 Un cas d'équivalence avec la généralisation confirmatoire

Nous pouvons remarquer que dans le mécanisme proposé par Ron Rymon l'attribut de classe n'intervient pas dans le calcul des premiers implicants : on peut par exemple ne pas le prendre en compte. Dans ce cas, EA'' peut être vue comme la négation de EA et par dualité entre les premiers implicants et les premiers impliqués, on obtient la propriété suivante :

Propriété 5.5 *Soient EA'^{+} l'ensemble des descriptions des exemples de classe $x = v_x$, c'est-à-dire sans la description de l'attribut de classe, et EA'^{-} l'ensemble des descriptions des exemples d'autres classes. Soit $RMIN^{+}(EA)$ l'ensemble des règles cohérentes et pertinentes minimales ayant pour conclusion $x = v_x$.*
$G \rightarrow (x = v_x)$ appartient à $RMIN^{+}(EA)$ si et seulement si $\neg G$ appartient à $IP(EA'^{-})$ modulo implication avec $IP(EA'^{+})$

Madre et Coudert [Madre et Coudert, 1994] ont proposé, dans un cadre propositionnel, la technique de construction des premiers impliqués suivante, que nous transcrivons seulement dans un cadre attribut-valeur.

Propriété 5.6 *$P \in IP(EA)$ si et seulement si*

- *$P = q \vee (x = v_x)$ et $q \in IP(EA'^{-})$ modulo implication avec $IP(EA'^{+})$*
- *P ne contient pas l'attribut x*
  et $P \in min_{\models}(q \vee r | q \in IP(EA'^{+}), r \in IP(EA'^{-}))$

Nous sommes maintenant en mesure de montrer qu'en présence d'objets complètement décrits, une règle minimale cohérente et pertinente pour EA est une généralisation de EA suivant le modèle de généralisation confirmatoire développé au chapitre 3.

Théorème 5.1 *Lorsque les objets de EA sont complètement décrits,*
$G \rightarrow (x = v_x)$ *est une règle minimale cohérente et pertinente pour EA si et seulement si* $G \Rightarrow (x = v_x)$ *est une généralisation de EA, c'est-à-dire* $\neg G \vee (x = v_x) \in IP(EA)$.

Preuve : Ce théorème est une conséquence de propriétés montrées par Ron Rymon [Rymon, 1994], à travers la technique de construction des premiers impliqués proposée par Madre et Coudert [Madre et Coudert, 1994]. #

En conséquence, l'ensemble $RMIN(EA)$ des règles minimales cohérentes et pertinentes pour EA est équivalent à la conséquence logique de EA la plus informative possible sur l'attribut x au sens de [Lakemeyer, 1995]. En effet, Lakemayer a montré qu'une telle formule est équivalente à la conjonction des premiers impliqués de EA qui contiennent x. $RMIN(EA)$ peut donc être calculée à l'aide des arbres d'énumération d'ensembles (cf. paragraphe 3.3).

La différence entre le formalisme de Ron Rymon et celui que nous avons choisi se réduit en fait à une dualité. Nous allons voir que lorsqu'on ajoute une théorie du domaine le formalisme de représentation des exemples utilisé ne conduit pas aux mêmes résultats.

5.5 Utilisation d'une théorie du domaine

L'utilisation d'une théorie du domaine présente au moins deux intérêts. Du point de vue de la représentation des connaissances, elle permet de simplifier la description des exemples. En effet, lorsque l'on veut décrire un individu, il suffit de décrire ses propriétés « principales », les valeurs prises par les autres attributs peuvent être déduites en utilisant la théorie du domaine. Un deuxième avantage est de restreindre l'ensemble des individus possibles. Toutes les descriptions qui ne sont pas cohérentes avec la théorie du domaine ne sont pas considérées.

Nous supposons que la théorie du domaine est une théorie clausale, c'est-à-dire un ensemble fini de clauses propositionnelles. C'est le cas usuel ; de plus, le cadre clausal est celui dans lequel notre modèle de généralisation confirmatoire est défini.

Dans la suite, nous exposons deux stratégies pour prendre en compte une théorie du domaine. La première est celle employée par Ron Rymon, la seconde est celle que nous proposons. Nous expliquons pourquoi notre stratégie est préférable à celle de Ron Rymon.

5.5.1 Ajout de ses contre-exemples

Dans le cadre d'attributs et d'une classe booléens, en présence de descriptions complètes, Ron Rymon a choisi de représenter les exemples comme des implications (paragraphe 5.3). Dans son approche, Ron Rymon prend en compte une théorie du domaine Th en considérant comme évidence la conjonction $Th \wedge EA''$ où EA'' est la conjonction des exemples sous forme implicative.

Considérons à nouveau l'exemple des patients et ajoutons la théorie du domaine $b \Rightarrow a$ et $c \Rightarrow d$. La conjonction $Th \wedge EA''$ est donc :
$((a = Vrai) \wedge (b = Vrai) \wedge (c = Vrai) \wedge (d = Vrai) \Rightarrow (x = Vrai))$
$\bigwedge((a = Faux) \wedge (b = Faux) \wedge (c = Faux) \wedge (d = Faux) \Rightarrow (x = Faux))$
$\bigwedge((a = Vrai) \wedge (b = Faux) \wedge (c = Faux) \wedge (d = Faux) \Rightarrow (x = Vrai))$
$\bigwedge((b = Vrai) \Rightarrow (a = Vrai))$
$\bigwedge((c = Vrai) \Rightarrow (d = Vrai))$.

Ron Rymon calcule alors les règles minimales cohérentes et pertinentes pour l'ensemble d'apprentissage à l'aide des premiers implicants, comme décrit à la propriété 5.4. Si l'on passe à la forme duale, et que l'on se « ramène » au formalisme de premiers impliqués que nous

utilisons, ce calcul est dual du calcul des règles données par $IP(EA \vee \neg Th)$. Puisque Th est une théorie clausale, $\neg Th$ est équivalent à l'ensemble des contre-exemples de Th, c'est-à-dire des objets incohérents avec Th.

Dans l'exemple courant :
$((a = Vrai) \wedge (b = Vrai) \wedge (c = Vrai) \wedge (d = Vrai) \wedge (x = Vrai))$
$\bigvee((a = Faux) \wedge (b = Faux) \wedge (c = Faux) \wedge (d = Faux) \wedge (x = Faux))$
$\bigvee((a = Vrai) \wedge (b = Faux) \wedge (c = Faux) \wedge (d = Faux) \wedge (x = Vrai))$
$\bigvee((b = Vrai) \wedge (a = Faux))$
$\bigvee((c = Vrai) \wedge (d = Faux))$.

La formule $EA \vee \neg Th$ représente un ensemble d'apprentissage composé des individus connus et des contre-exemples de la théorie du domaine. L'ajout des contre-exemples modifie l'ensemble d'apprentissage en lui ajoutant des contraintes excessives. Nous allons montrer, en effet, qu'un tel ajout conduit à un ensemble de règles minimales trop spécifiques.

5.5.2 Simplification par la théorie du domaine

De façon cohérente avec la description des exemples et conformément au formalisme utilisé dans le modèle de généralisation confirmatoire que nous avons proposé, l'ensemble d'apprentissage est la conjonction de l'ensemble d'exemples et de la théorie du domaine.

Définition 5.7 *Soient EA un ensemble d'apprentissage et Th une théorie du domaine tel que chaque exemple de EA soit cohérent au sens logique avec Th. Une règle cohérente et pertinente pour EA étant donné Th est un couple noté $G \rightarrow (x = v_x)$ tel que :*

- *Pour tout exemple e de EA, si $e \wedge Th \models G$ alors $classe(e) = v_x$.*
- *Il existe un exemple e tel que $e \wedge Th \models G$.*

Définition 5.8 *Une règle $G \rightarrow (x = v_x)$ classe un objet o en présence d'une théorie du domaine Th si et seulement si $o \wedge Th \models G$ et o est cohérent logiquement avec Th.*

En présence d'une théorie du domaine, les objets dont la description n'est pas cohérente logiquement avec la théorie ne sont pas considérés. Nous pouvons noter que lorsque les exemples sont des descriptions complètes cohérentes avec la théorie du domaine, celle-ci est une conséquence logique des exemples. Donc l'ensemble des règles minimales cohérentes et pertinentes pour EA est équivalent à l'ensemble des règles minimales cohérentes et pertinentes pour $EA \wedge Th$, par rapport à la relation d'équivalence induite par la déduction logique modulo Th, c'est-à-dire qu'aucune information sur la classe n'est perdue. $RMIN(EA \wedge Th)$ et $RMIN(EA)$ associent la (ou les) même(s) classe(s) à tout objet cohérent avec Th. L'ajout d'une théorie du domaine peut réduire le nombre de règles minimales qui peuvent se retrouver impliquées en présence de Th, mais leur nombre ne peut pas augmenter quand on renforce Th. Ces deux aspects sont illustrés au paragraphe 5.5.3.

5.5.3 Comparaison de ces deux techniques

Nous pouvons vérifier sur un exemple que la technique d'ajout des contre-exemples conduit à des résultats moins intéressants que la technique de simplification que nous proposons. Consi-

dérons l'ensemble d'apprentissage EA de l'exemple des patients :

Patient	a	b	c	d	x
e_1	vrai	vrai	vrai	vrai	vrai
e_2	faux	faux	faux	faux	faux
e_3	vrai	faux	faux	faux	vrai

La figure 5.1 présente l'ensemble des règles minimales cohérentes et pertinentes pour EA et la (ou les) classe(s) qu'elles associent à chaque individu de l'$UNIVERS$. Les méthodes de simplification par la théorie du domaine et d'ajout de ses contre-exemple sont comparées lorsqu'une théorie du domaine logiquement de plus en plus forte ($\emptyset$, $\{b \Rightarrow a\}$ et $\{b \Rightarrow a$ et $c \Rightarrow d\}$) est considérée.

Nous vérifions que l'ensemble des règles minimales est resté équivalent en présence de la théorie du domaine dans le cas de la simplification par la théorie et que le nombre de règles décroît de façon monotone. Dans le cas de l'ajout des contre-exemples, le nombre de règles peut croître soudainement. De plus, les règles inférées sont de plus en plus spécifiques. La simplification par la théorie du domaine conduit à un ensemble de règles qui classe toujours les individus de l'$UNIVERS$ dans la ou les mêmes classes. Elle interdit simplement de classer les individus incohérents avec la théorie. L'ajout des contre-exemples peut conduire à des ensembles de règles n'associant pas la (ou les) même(s) classe(s) aux objets de l'$UNIVERS$. Cela peut être bénéfique en effectuant des résolutions de conflits, par exemple dans le cas de la seconde théorie du domaine $\{a \vee \bar{b}\}$, mais c'est parfois au prix d'une plus grande incomplétude ; ainsi, l'individu $\bar{a}\bar{b}cd$ n'est plus couvert dans le cas de la troisième théorie du domaine $\{(a \vee \bar{b}) \bigwedge (\bar{c} \vee d)\}$.

5.6 Les règles minimales cohérentes et pertinentes ne sont pas toujours des généralisations.

Lorsque l'on considère des individus incomplètement décrits, les règles minimales cohérentes et pertinentes et les généralisations confirmatoires pour l'ensemble d'apprentissage diffèrent. Considérons le problème de classification suivant :

$$EA =_d \{((a = V), (x = V)), ((b = V), (x = F))\}.$$

Dans ce problème, a et b sont les deux seuls attributs booléens utilisés pour décrire les objets. Les règles minimales cohérentes et pertinentes pour EA sont :

$$(a = V) \rightarrow (x = V) \text{ et } (b = V) \rightarrow (x = F).$$

Les généralisations confirmatoires pour EA sont ses premiers impliqués :

$$(a = V) \vee (x = F) \text{ et } (b = V) \vee (x = V),$$

c'est-à-dire les règles :

$$(a = F) \rightarrow (x = F) \text{ et } (b = F) \rightarrow (x = V).$$

Ces règles sont cohérentes pour EA mais pas pertinentes pour EA.

Lorsque l'on considère des exemples complètement décrits, $G \rightarrow (x = v_x)$ *est une règle minimale* équivaut à $\neg G \vee (x = v_x)$ *est une généralisation confirmatoire de* EA (Théorème 5.1). Ce résultat n'est pas valable en présence d'exemples incomplètement décrits. La différence entre la définition des règles minimales cohérentes et pertinentes et les généralisations confirmatoires apparaît clairement lorsque l'on fait apparaître les modèles des formules considérées. Commençons par expliciter les relations de conséquence logique mises en œuvre dans la définition des règles cohérentes et pertinentes pour EA :

soient EA^+ l'ensemble des exemples de classe $x = v_x$ et EA^- l'ensemble des exemples d'autres classes. Le couple $G \rightarrow (x = v_x)$ est une règle cohérente et pertinente pour EA si et

Ajout des contre-exemples. **Simplification par la théorie.**

$Th = \emptyset$

$a \to x, \bar{a} \to \bar{x},$
$b \to x, c \to x, d \to x$

	ab	$a\bar{b}$	$\bar{a}b$	$\bar{a}\bar{b}$
cd	x	x	$x\bar{x}$	$x\bar{x}$
$c\bar{d}$	x	x	$x\bar{x}$	$x\bar{x}$
$\bar{c}\bar{d}$	x	x	$\bar{x}$	$x\bar{x}$
$\bar{c}d$	x	x	$x\bar{x}$	$x\bar{x}$

$Th = \{a \vee \bar{b}\}$

$a \to x, \bar{a} \wedge \bar{b} \to \bar{x}$

	ab	$a\bar{b}$	$\bar{a}b$	$\bar{a}\bar{b}$
cd	x	x	$\bar{x}$	
$c\bar{d}$	x	x	$\bar{x}$	
$\bar{c}\bar{d}$	x	x	$\bar{x}$	
$\bar{c}d$	x	x	$\bar{x}$	

$a \to x, \bar{a} \to \bar{x},$
$c \to x, d \to x$

	ab	$a\bar{b}$	$\bar{a}b$	$\bar{a}\bar{b}$
cd	x	x	$x\bar{x}$	
$c\bar{d}$	x	x	$x\bar{x}$	
$\bar{c}\bar{d}$	x	x	$\bar{x}$	
$\bar{c}d$	x	x	$x\bar{x}$	

$Th = \{(a \vee \bar{b}) \bigwedge (\bar{c} \vee d)\}$

$a \wedge \bar{c} \to x, a \wedge d \to x$
$\bar{a} \wedge \bar{b} \wedge \bar{c} \to \bar{x}$

	ab	$a\bar{b}$	$\bar{a}b$	$\bar{a}\bar{b}$
cd	x	x		
$c\bar{d}$				
$\bar{c}\bar{d}$	x	x	$\bar{x}$	
$\bar{c}d$	x	x	$\bar{x}$	

$a \to x, \bar{a} \to \bar{x},$
$c \to x$

	ab	$a\bar{b}$	$\bar{a}b$	$\bar{a}\bar{b}$
cd	x	x	$x\bar{x}$	
$c\bar{d}$				
$\bar{c}\bar{d}$	x	x	$\bar{x}$	
$\bar{c}d$	x	x	$x\bar{x}$	

FIG. 5.1 – *Règles minimales cohérentes et pertinentes et la(les) classe(s) associée(s) aux individus de l'UNIVERS pour trois théories du domaine.*

seulement si

- pour tout exemple e de EA, si $e \models G$ alors $e \models (x = v_x)$, c'est-à-dire pour tout exemple e de EA, $e \not\models G$ ou $e \models (x = v_x)$
- il existe un exemple e tel que $e \models G$

ce qui équivaut à :

- pour tout exemple e de EA^-, $e \not\models G$,
  c'est-à-dire pour tout exemple e de EA^-, il existe un modèle de e qui satisfait $\neg G$.
- il existe un exemple e de EA^+ tel que $e \models G$,
  c'est-à-dire il existe un exemple e appartenant à EA^+ et dont tous les modèles satisfont G.

Dans une représentation attributs-valeurs, les généralisations confirmatoires de EA sont ses premiers impliqués. Caractérisons en terme de modèles les généralisations confirmatoires :

Propriété 5.7 *$\neg G \vee (x = v_x)$ est une généralisation confirmatoire de EA, c'est-à-dire appartient à $IP(EA)$ si et seulement si $(G, (x = v_x))$ satisfait les conditions (C):*

- *pour tout exemple e de EA^-, $e \models \neg G$,*
  c'est-à-dire pour tout exemple e de EA^-, tous les modèles de e satisfont $\neg G$.
- *il existe un exemple e appartenant à EA^+ tel que e et G sont cohérents,*
  c'est-à-dire il existe un exemple e appartenant à EA^+ dont un modèle satisfait G.

et $\forall G'$ tel que $G \models G'$ et $G' \not\models G$, $(G', (x = v_x))$ ne satisfait pas les conditions (C).

Preuve :

$\Rightarrow$ Soit $\neg G \vee (x = v_x) \in IP(EA)$,

- pour tout e de EA^-, $e \models \neg(x = v_x)$
  et $\neg G \vee (x = v_x) \in IP(EA)$, donc $e \models \neg G \vee (x = v_x)$,
  donc $e \models \neg G$.
- Supposons que pour tout e appartenant à EA^+, on ait $e \models \neg G$.
  Alors, $\neg G \in IP(EA)$. Ceci contredit $\neg G \vee (x = v_x) \in IP(EA)$.
  Donc il existe e appartenant à EA^+ tel que $e \not\models \neg G$, c'est-à-dire il existe un modèle de e qui satisfait G.
- Pour montrer la minimalité, on suppose montré que $(G, (x = v_x))$ satisfait (C) implique $EA \models \neg G \vee (x = v_x)$.
  Supposons qu'il existe $G' \subset c$ tel que $(G', (x = v_x))$ satisfait (C). Alors, $E \models \neg G' \vee (x = v_x)$.
  Et $\neg G' \vee (x = v_x) \models \neg G \vee (x = v_x)$. Ceci contredit $\neg G \vee (x = v_x) \in IP(EA)$. Donc $(G, (x = v_x))$ est minimale.

$\Leftarrow$ Soit $(G, (x = v_x))$ vérifiant (C) et minimale,
pour tout o appartenant à EA^+, $o \models (x = v_x)$ donc $o \models \neg G \vee (x = v_x)$
et pour tout o appartenant à EA^-, $o \models \neg G$ donc $o \models \neg G \vee (x = v_x)$.
Ainsi, $(G, (x = v_x))$ vérifie (C) implique $EA \models \neg G \vee (x = v_x)$.
Donc il existe $p \in IP(EA)$ tel que $p \models \neg G \vee (x = v_x)$.
Supposons que $p \models \neg G \vee (x = v_x)$ soit vérifié. On peut distinguer deux cas :

1. $p \subseteq \neg G$.
   $p \in IP(EA)$, donc pour tout e de EA^+, $e \models p$, donc pour tout e de EA^+, $e \models \neg G$. Ceci contredit l'existence d'un exemple e de EA^+ tel que $e \not\models \neg G$.

2. $p = \neg q \vee (x = v_x)$ où $q \subset G$.
   Alors, d'après la démonstration du sens direct, $(q, (x = v_x))$ satisfait (C). Ceci contredit $(G, (x = v_x))$ minimale.

Donc $p = \neg G \vee (x = v_x)$. Ainsi $\neg G \vee (x = v_x) \in IP(EA)$.

#

Ainsi, quand on utilise des descriptions incomplètes des objets, on ne peut pas caractériser les règles minimales cohérentes et pertinentes pour EA comme ses premiers impliqués.

Remarquons que généralisations et règles minimales cohérentes et pertinentes diffèrent également au premier ordre. Considérons le rapport d'observation :

$$E = \{a(i_1), x(i_1), b(i_2), \neg x(i_2)\}.$$

Les règles minimales cohérentes et pertinentes pour E sont :

$$\forall y, a(y) \rightarrow x(y) \text{ et } \forall y, b(y) \rightarrow \neg x(y).$$

Et les généralisations de E sont :

$$\forall y, a(y) \vee \neg x(y) \text{ et } \forall y, b(y) \vee x(y),$$

c'est-à-dire les règles :

$$\forall y, \neg a(y) \rightarrow \neg x(y) \text{ et } \forall y, \neg b(y) \rightarrow x(y).$$

Ces règles cohérentes et pertinentes ne sont-elles pas confirmées? Les règles minimales ne sont-elles plus des généralisations? Tout dépend de l'interprétation de l'hypothèse des similarités que l'on prend en compte. La confirmation selon Hempel correspond essentiellement à la partie « cohérence » de la définition des règles alors que le critère de Nicod correspond à la partie « pertinence ». Notons au passage une conséquence de la pertinence : les règles pertinentes ne satisfont pas l'équivalence à droite. Toute règle logiquement « équivalente » à une règle n'est pas nécessairement confirmée quand la règle l'est. Ainsi, l'exemple $(a = Vrai), (x = Vrai)$ satisfait la règle minimale cohérente et pertinente $(a = Vrai) \rightarrow (x = Vrai)$, mais pas la règle logiquement équivalente $(x = Faux) \rightarrow (a = Faux)$.

Une règle n'est pas une implication logique : la distinction est sémantique, la partie gauche et la partie droite jouent des rôles différents alors que ce n'est pas le cas des littéraux d'une clause. En fait une règle ne se représente pas simplement par une formule logique ayant exactement le même sens. Ainsi, une règle $G \rightarrow (x = v_x)$ pourrait s'exprimer au premier ordre par la formule $\forall I, \neg\tau(I, G) \vee \tau(I, x = v_x) \wedge \exists J / \tau(J, G)$. Malheureusement, cette formule n'appartient pas au langage des généralisations L_G du modèle de généralisation que nous avons proposé.

5.7 Sémantique des valeurs manquantes vis-à-vis des règles cohérentes et pertinentes

Pour comprendre pourquoi les règles minimales cohérentes et pertinentes ne sont plus des généralisations confirmatoires en présence d'exemples incomplètement décrits, il est nécessaire de s'intéresser aux sémantiques possibles de tels exemples.

Un exemple est incomplètement décrit quand la valeur associée à chaque attribut n'est pas nécessairement connue. On peut concevoir au moins deux « interprétations » d'un objet incomplètement décrit :

« **Interprétation conjonctive** » : les attributs non-valués ne sont pas pertinents pour la classification. Par exemple, la nationalité du patient n'influe pas sur la maladie x. Dans ce cas, l'objet désigne tous ses modèles, c'est-à-dire tous les objets satisfaisant sa description sont des exemples de sa classe. Un tel objet peut donc être assimilé à une représentation compacte d'un ensemble d'exemples. Un exemple incomplètement décrit désigne alors

autant d'exemples complètement décrits qu'il a de modèles (un tel objet incomplètement décrit peut être assimilé à une règle, cf. paragraphe 7.3).

« **Interprétation disjonctive** » : les attributs non-valués sont *a priori* pertinents pour la classification mais leurs valeurs sont inconnues. Dans ce cas, l'objet désigne un de ses modèles, mais on ne sait pas lequel.

Différentes « interprétations » de l'objet incomplètement décrit $(b, \bar{x})$ de EA sont représentées à la figure 5.2, avec les règles minimales cohérentes et pertinentes pour EA associées.

$$EA =_d \{(ab, x), (\mathbf{b}, \bar{\mathbf{x}})\}$$
$$a \rightarrow x$$

$\swarrow ?$ $\downarrow ?$ $\searrow ?$

$EA =_d \{(ab, x), (\mathbf{ab}, \bar{\mathbf{x}}), (\bar{\mathbf{ab}}, \bar{\mathbf{x}})\}$ $EA =_d \{(ab, x), (\mathbf{ab}, \bar{\mathbf{x}})\}$ $EA =_d \{(ab, x), (\bar{\mathbf{ab}}, \bar{\mathbf{x}})\}$

$\bar{a} \rightarrow \bar{x}$ $\emptyset$ $a \rightarrow x$, $\bar{a} \rightarrow \bar{x}$

FIG. 5.2 – *Différentes « interprétations » d'un objet incomplètement décrit et les règles minimales cohérentes et pertinentes associées.*

Si l'on considère $EA =_d \{(ab, x), (b, \bar{x})\}$ dans un espace décrit par deux attributs et une classe booléens, la seule règle minimale cohérente et pertinente pour EA est $a \rightarrow x$. Si l'on remplace l'exemple $(b, \bar{x})$ par tous ses modèles $\{(ab, \bar{x}), (\bar{a}b, \bar{x})\}$, la seule règle minimale cohérente et pertinente pour EA est $\bar{a} \rightarrow \bar{x}$. On constate clairement qu'un objet incomplètement décrit ne se comporte pas comme l'ensemble de ses modèles vis-à-vis de l'ensemble des règles minimales cohérentes et pertinentes pour l'ensemble d'apprentissage.

Puisqu'un objet incomplètement décrit ne se comporte pas comme l'ensemble de ses modèles vis-à-vis de l'ensemble des règles minimales cohérentes et pertinentes pour l'ensemble d'apprentissage, voyons ce qu'il se passe quand on ne le remplace que par un seul de ses modèles.

- Si l'on remplace $(b, \bar{x})$ par $(\bar{a}b, \bar{x})$, les règles minimales cohérentes et pertinentes pour EA sont $a \rightarrow x$ et $\bar{a} \rightarrow \bar{x}$.

- Si l'on remplace $(b, \bar{x})$ par $(ab, \bar{x})$, EA n'admet aucune règle cohérente et pertinente !

La seule règle minimale cohérente et pertinente pour $EA =_d \{(ab, x), (b, \bar{x})\}$ est $a \rightarrow x$. Il peut paraître étonnant qu'elle n'apparaisse que dans le cas où $(b, \bar{x})$ désigne $(\bar{a}b, \bar{x})$ et que la règle $\bar{a} \rightarrow \bar{x}$, qui est produite dans le même cas, ne soit pas également une règle minimale cohérente et pertinente pour $EA =_d \{(ab, x), (b, \bar{x})\}$. Cette discrimination vient de ce que les deux exemples (ab, x) et $(b, \bar{x})$ ne jouent pas le même rôle vis-à-vis de ces règles. En effet, les deux règles doivent être cohérentes avec tous les exemples, mais chacune doit être rendue pertinente par au moins un exemple.

Pour qu'une règle $G \rightarrow (x = v_x)$ soit cohérente avec un objet o, il suffit que $classe(o) = v_x$ ou que G ne soit pas satisfaite par au moins un modèle de o. La règle $a \rightarrow x$ est cohérente avec (ab, x) car $classe((ab, x)) = x$ et elle est cohérente avec $(b, \bar{x})$ car un de ses modèles, $\bar{a}b$, ne satisfait pas a. De même, la règle $\bar{a} \rightarrow \bar{x}$ est cohérente avec les deux objets complètement décrits.

Pour qu'une règle $G \rightarrow (x = v_x)$ soit pertinente pour l'ensemble d'apprentissage, il faut que sa partie gauche G soit satisfaite par au moins un objet de classe v_x, c'est-à-dire par *tous*

ses modèles. La règle $a \rightarrow x$ est rendue pertinente par (ab, x), puisque $ab \models a$. Mais la règle $\bar{a} \rightarrow \bar{x}$ ne peut être rendue pertinente par $(b, \bar{x})$ puisque son modèle ab ne satisfait pas $\bar{a}$.

Ainsi, vis-à-vis des règles minimales cohérentes et pertinentes pour un ensemble d'apprentissage, un objet incomplètement décrit ne peut être assimilé, dans le cas général, ni à l'ensemble de ses modèles, ni à l'un d'entre eux en particulier. En effet, pour la propriété de pertinence, c'est la sémantique conjonctive qui s'applique, alors que c'est la sémantique disjonctive qui s'applique pour la propriété de cohérence. Cela explique pourquoi les généralisations confirmatoires ne sont pas les règles minimales cohérentes et pertinentes en présence d'exemples incomplètement décrits. En effet, pour la généralisation confirmatoire fondée sur l'hypothèse des similarités selon Hempel, c'est la sémantique conjonctive qui s'applique toujours.

Parmi les programmes de classification que nous utilisons dans l'évaluation expérimentale du chapitre 9, SE-Learn [Rymon, 1996a] calcule l'ensemble des règles $RMIN(EA)$ à l'aide d'arbres d'énumération d'ensembles. Nous avons vu au paragraphe 3.3 que les arbres d'énumération d'ensembles permettent de calculer des généralisations confirmatoires. Sachant que les généralisations confirmatoires et les règles minimales cohérentes et pertinentes ne sont pas équivalentes dans le cas général, cela peut paraître surprenant. SE-Learn ne calculerait-il pas ce pour quoi il a été construit? L'astuce est simplement que SE-Learn ne prend pas en compte la sémantique des valeurs manquantes. Il assimile l'absence de valeur à une valeur à part entière. Plus précisément, la description des exemples comporte une valeur particulière désignant la valeur manquante (nous avons choisi -1), et il faut la déclarer à SE-Learn comme une valeur ordinaire. Par ce biais, toutes les descriptions sont complètes et les règles minimales cohérentes et pertinentes sont les généralisations confirmatoires. Notons que d'autres programmes n'acceptent les valeurs manquantes que grâce à cet artifice de représentation. C'est le cas par exemple de PEBLS [Salzberg, 1991], ou de RISE [Domingos, 1996].

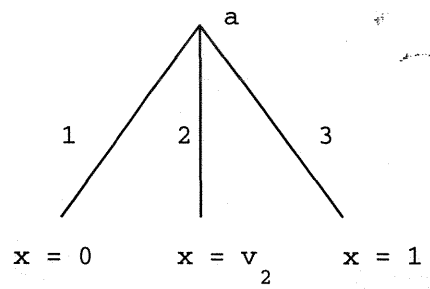
5.8 Classifieurs à base de règles quelconques

Nous n'avons jusqu'ici considéré que des règles cohérentes et pertinentes. La contrainte de cohérence avec l'ensemble d'apprentissage est un premier biais logique. De nombreuses techniques de classification n'imposent pas au classifieur d'être cohérent avec l'ensemble d'apprentissage. C'est le cas, par exemple, des arbres de décision avec élagage. L'élagage est utilisé pour prendre en compte la présence d'éventuels exemples incorrects, et dans ce cas, il est normal de lever la contrainte de cohérence. L'élagage est également utilisé pour obtenir des règles plus générales, couvrant plus d'exemples.

La contrainte de pertinence constitue également un biais logique. Ce biais suppose qu'une règle intéressante couvre au moins un exemple. Si l'ensemble d'apprentissage n'est pas représentatif de toutes les règles nécessaires à la description de la fonction cible, la contrainte de pertinence ne permet pas d'obtenir le « bon classifieur ».

Comme nous l'avons vu ci-dessus, les arbres de décision ne sont pas toujours cohérents pour EA, ils ne sont pas non plus nécessairement pertinents pour EA. Considérons un $UNIVERS$ décrit par un seul attribut nominal a pouvant prendre les valeurs 1, 2 et 3. Étant donné l'ensemble d'apprentissage $EA =_d \{((a = 1), (x = 0)), ((a = 3), (x = 1))\}$, le seul arbre de

décision possible pour EA est le suivant :



La règle $(a = 2) \rightarrow (x = v_2)$ n'est pas pertinente pour EA, quelle que soit la valeur v_2.

Ainsi, tous les classifieurs à base de règles n'utilisent pas les biais de cohérence et de pertinence : ils ne sont pas tous des classifieurs à base de règles cohérentes et pertinentes. C'est par exemple le cas de C4.5 qui utilisent un élagage.

5.9 Résumé et conclusion

Dans ce chapitre, nous avons considéré une famille particulière de classifieurs à base de règles : les classifieurs à base de toutes les règles minimales cohérentes et pertinentes pour l'ensemble d'apprentissage. Ces classifieurs contiennent toutes les règles cohérentes avec l'ensemble d'apprentissage, c'est-à-dire n'admettant pas de contre-exemple, et pertinentes, c'est-à-dire satisfaites par au moins un exemple. Ces deux conditions raisonnables constituent évidemment des biais inductifs. Tous les classifieurs à base de règles n'utilisent pas que des règles cohérentes ou pertinentes (par exemple les arbres de décision) et, en général, ne les contiennent pas toutes. Nous avons montré que les classifieurs à base de toutes les règles minimales cohérentes et pertinentes pour EA vérifient certaines propriétés intéressantes. En particulier, lorsque les individus/objets sont complètement décrits, les règles minimales cohérentes et pertinentes pour EA sont les généralisations confirmatoires de EA contenant l'attribut de classe. Elles peuvent donc être calculées à l'aide des arbres d'énumération d'ensembles. Nous avons montré que les règles minimales cohérentes et pertinentes diffèrent des généralisations confirmatoires dans le cas de descriptions incomplètes et expliqué cette différence en nous focalisant sur la sémantique des objets incomplètement décrits. L'ensemble des règles minimales cohérentes et pertinentes pour EA ne peut donc être calculé à l'aide d'un arbre d'énumération d'ensembles en présence d'objets incomplètement décrits. Nous proposons dans le chapitre suivant une technique de classification qui résout ce problème.

Chapitre 6

Des règles aux instances : la classification par portée

Nous proposons dans ce chapitre une technique de classification que nous avons appelée classification par portée. Nous définissons dans un premier temps ce qu'est la portée d'un exemple. Nous proposons ensuite un point de vue original sur la classification à base de règles qui consiste à chercher les exemples à partir desquels au moins une règle pertinente et cohérente pour l'ensemble d'apprentissage et couvrant l'objet à classer peut être construite. Cette approche garantit l'existence d'une correspondance entre l'ensemble des exemples retenus et l'ensemble de toutes les règles cohérentes et pertinentes pour l'ensemble d'apprentissage. Nous proposons un algorithme de classification par portée tirant parti d'une recherche « diviser pour régner ». Un intérêt majeur des approches auxquelles notre technique se rattache est de séparer clairement les biais logique et statistique intervenant dans la classification. La construction de l'ensemble des exemples, de l'ensemble des voisins ou de l'ensemble des règles ne repose que sur des critères logiques. Un biais numérique n'intervient que pour compléter le biais logique et choisir entre des hypothèses non-séparables par le biais logique seul. Bien que notre approche ne nécessite pas de générer explicitement de règles, il est aisé d'engendrer des règles à la demande et de les généraliser afin de justifier les classifications proposées. Enfin, nous étudions les liens existants avec les approches fondées sur les distances, les approches à base d'instances, et surtout l'espace des versions disjonctif, qui peut être considéré comme une autre vision de la classification par portée.

6.1 La portée

Considérons à nouveau l'ensemble d'apprentissage de l'exemple des patients :

Patient	a	b	c	d	x
e_1	V	V	V	V	V
e_2	F	F	F	F	F
e_3	V	F	F	F	V

L'ensemble des règles minimales cohérentes et pertinentes pour cet ensemble d'apprentissage est :

$$\begin{aligned} RMIN(EA) = \{&(a = Vrai) \rightarrow (x = Vrai),\\ &(b = Vrai) \rightarrow (x = Vrai),\\ &(c = Vrai) \rightarrow (x = Vrai),\\ &(d = Vrai) \rightarrow (x = Vrai),\\ &(a = Faux) \rightarrow (x = Faux)\}. \end{aligned}$$

Une règle $G \rightarrow (x = v_x)$ classe conformément à sa partie droite chaque individu couvert par sa partie gauche. La connaissance de l'ensemble des règles minimales cohérentes et pertinentes pour EA permet de proposer au moins une classe pour chaque individu couvert par au moins une règle.

Une règle $G \rightarrow (x = v_x)$ devant être cohérente pour EA, chaque exemple qu'elle couvre appartient à la classe v_x. Réciproquement, toutes les règles cohérentes pour EA qui couvrent un exemple e ont pour conséquence $(x = classe(e))$. L'ensemble des prémisses des règles couvrant un exemple e définit la portée de e.

Définition 6.1 *La* portée *d'un exemple e pour un ensemble d'apprentissage EA est l'ensemble des individus couverts par toutes les règles cohérentes et pertinentes avec EA satisfaites par e.*

La portée de l'exemple e_3 est représentée par la zone grisée sur le tableau de Karnaugh ci-dessous. a et $\bar{a}$ représentent respectivement $(a = V)$ et $(a = F)$.

	ab	$a\bar{b}$	$\bar{a}\bar{b}$	$\bar{a}b$
cd	x			
$c\bar{d}$	?			
$\bar{c}\bar{d}$		x	$\bar{x}$	
$\bar{c}d$				

La portée d'un exemple est la même que l'on se limite aux règles minimales ou non.

Propriété 6.1 *La portée d'un exemple e pour un ensemble d'apprentissage EA est aussi l'ensemble des individus couverts par toutes les règles minimales cohérentes et pertinentes pour EA satisfaites par e.*

Preuve :

⇒ Chaque règle cohérente et pertinente est impliquée par une règle minimale, donc l'ensemble des individus couverts par une règle cohérente et pertinente est couvert par une règle minimale.

⇐ L'ensemble de toutes les règles de classification contient l'ensemble de toutes les règles minimales, donc les individus couverts par l'ensemble des règles minimales sont couverts par l'ensemble de toutes les règles de classification.

#

La portée d'un exemple e est également l'union de tous les sous-ensembles de la description de e qui ne couvrent aucun exemple d'une classe différente de celle de e. La portée est donc également une forme d'étoile, en anglais *star* [Michalski, 1983]. Les algorithmes AQ [Michalski, 1969] utilisent l'étoile pour calculer des hypothèses cohérentes et pertinentes pour l'ensemble

d'apprentissage EA. Nous effectuons le raisonnement inverse car nous supposons disposer de toutes les règles et nous en déduisons la portée de chaque exemple. La connaissance de la portée de tous les exemples peut remplacer l'ensemble de toutes les règles cohérentes et pertinentes pour EA et permettre de connaître la ou les classes associées par $RMIN(EA)$ à chaque individu. Pour classer un individu o, il faut déterminer l'ensemble des portées auxquelles il appartient, c'est-à-dire l'ensemble des exemples dont la portée contient o. Bien sûr, s'il fallait calculer toutes les règles pour déterminer la portée et classer de nouveaux individus, cela présenterait peu d'intérêt. Nous allons montrer qu'il est possible de se dispenser du calcul des règles et qu'il n'est pas utile de déterminer explicitement la portée des exemples pour pouvoir classer un nouvel individu.

6.2 Des règles aux instances

Essayons de classer un nouvel individu $o =_d (a = V) \wedge (b = V) \wedge (c = V) \wedge (d = F)$ à partir de l'ensemble d'apprentissage EA constitué des trois exemples/patients $\{e_1, e_2, e_3\}$ sans calculer explicitement toutes les règles cohérentes et pertinentes pour EA, ni la portée des exemples. Déterminons d'abord, pour chaque exemple e, la règle $R(o, e)$ la plus spéficique couvrant o et e (cette notion sera formellement définie à la définition 6.4). $R(o, e) =_d C(o, e) \rightarrow (x = classe(e))$, où $C(o, e)$ est dans le cas d'attributs nominaux l'ensemble des couples attributs-valeurs communs à o et e : $C(o, e_1) = \{(a = V), (b = V), (c = V)\}$, $C(o, e_2) = \{(d = F)\}$ et $C(o, e_3) = \{(a = V), (d = F)\}$.

Déterminons les règles $R =_d G \rightarrow (x = v_x)$ couvrant o. Une règle $R =_d G \rightarrow (x = v_x)$ couvrant e_1 et o existe si et seulement si $e_1 \models G$ et $o \models G$. Ainsi, une condition nécessaire pour qu'une telle règle existe est $C(o, e_1) \models G$. Toute règle couvrant e_1 et o implique donc nécessairement la règle $R(o, e_1) =_d (a = V) \wedge (b = V) \wedge (c = V) \rightarrow (x = V)$. La règle $R(o, e_1)$ est pertinente puisqu'elle est satisfaite par e_1. Pour être acceptable (c'est-à-dire appartenir à l'ensemble des règles cherchées), elle doit aussi être cohérente avec tous les exemples. Supposons que cela ne soit pas le cas. Alors, il existe un exemple ce de la classe $(x = F)$ couvert par $R(o, e_1)$. En conséquence, ce contient au moins les couples $(a = V)$, $(b = V)$ et $(c = V)$. Donc $ce \in C(o, e_1)$. Cette relation peut être interprétée par « ce est plus proche de o que e_1 », que nous notons $ce \leq_o e_1$. Ainsi, vérifier que la règle $R(o, e_1)$ est cohérente avec l'ensemble d'apprentissage EA peut se faire simplement en vérifiant qu'aucun exemple ce appartenant à une classe différente de $(x = V)$ ne satisfait $ce \leq_o e_1$. Puisque e_2 est le seul contre-exemple et puisque $e_2 \notin C(o, e_1)$ est vérifié, $R(o, e_1)$ est une règle cohérente et pertinente pour EA qui classe o dans la classe $(x = V)$. Ainsi, o appartient à la portée de e_1. Il n'est pas utile de calculer explicitement cette portée, ni l'ensemble des règles cohérentes et pertinentes pour EA couvrant e_1.

En appliquant la même technique, nous pouvons déterminer si o appartient à la portée de e_2 ou de e_3. Nous pouvons facilement vérifier que $R(o, e_2) =_d C(o, e_2) \rightarrow (x = F)$ n'est pas une règle cohérente et pertinente pour EA puisque $e_3 \in C(o, e_2)$: $R(o, e_2)$ n'est pas cohérente avec e_3. En revanche, $R(o, e_3) =_d C(o, e_3) \rightarrow (x = V)$ est une règle cohérente et pertinente pour EA qui couvre o.

Nous avons montré que o appartient à la portée de e_1 et à la portée de e_3 sans avoir à calculer explicitement les portées, et sans générer toutes les règles cohérentes et pertinentes pour EA possibles. Nous avons uniquement eu recours à des comparaisons entre les descriptions des exemples. Il est donc possible de classer comme avec des règles en n'utilisant que des instances. C'est l'idée de base de la classification par portée.

6.3 Formalisation de la classification par portée

Intuitivement, déterminer tous les exemples e de l'ensemble d'apprentissage EA tels qu'une règle cohérente et pertinente pour EA couvre e et o revient à déterminer tous les exemples e plus proches de o que tous les exemples d'une classe différente de la leur. Dans notre approche, la notion de proximité repose sur des pré-ordres partiels sur l'$UNIVERS$, indexés par les objets. Définissons tout d'abord l'*hyper-rectangle* $C(o_1, o_2)$ défini par deux objets o_1 et o_2.

Définition 6.2 *Soient o_1 et o_2 deux objets, éventuellement partiellement décrits. L'hyper-rectangle $C(o_1, o_2)$ défini par o_1 et o_2 est la partie gauche de la règle la plus spécifique couvrant tous les modèles de o_1 et tous les modèles de o_2. $C(o_1, o_2)$ est une conjonction de conditions (ou description partielle) étendues aux intervalles dans le cas des attributs linéaires. La condition ou sélecteur, $C_a(o_1, o_2)$ de $C(o_1, o_2)$ sur chaque attribut a (ou valeur associée à l'attribut a) est :*

- *Vrai (ou manquante) si a n'est pas valué dans o_1 ou o_2;*
- *$(a = v_a^1)$ où v_a^1 est la valeur commune à o_1 et o_2 si a est nominal et valué de même valeur dans o_1 et o_2, Vrai (ou manquante) sinon;*
- *$(a \in [min(v_a^1, v_a^2), max(v_a^1, v_a^2)])$ si a est linéaire et prend les valeurs v_a^1 et v_a^2 dans les descriptions de o_1 et o_2.*

$C(o_1, o_2)$ est la généralisation maximalement spécifique $sup(o_1, o_2)$ des espaces des versions [Mitchell, 1982; Michalski, 1983].

Un pré-ordre partiel entre les individus est défini à l'aide de $C(o_1, o_2)$.

Définition 6.3 *Pour chaque objet o de l'$UNIVERS$, nous définissons le pré-ordre partiel $\leq_o$ sur l'$UNIVERS$ par :*
un objet o_1 est plus proche de o qu'un objet o_2, noté $o_1 \leq_o o_2$, si et seulement si tous les modèles de o_1 sont des modèles de $C(o, o_2)$, c'est-à-dire $o_1 \models C(o, o_2)$. Nous notons également cette relation $o_1 \in C(o, o_2)$.

Remarquons que la relation $\leq_o$ peut être définie également de la façon suivante :

Propriété 6.2 *Soient o, o_1 et o_2 des objets, éventuellement incomplètement décrits.*
$o_1 \models C(o, o_2)$ est équivalent à $C(o, o_1) \models C(o, o_2)$.

Preuve : considérons le cas de chaque attribut a et vérifions $(a = v_a^1) \models C_a(o, o_2)$ équivaut à $C_a(o, o_1) \models C_a(o, o_2)$:

- si a est non-valué dans o, toutes les conditions sont vraies, sinon,
- si a est non-valué dans o_1, $C_a(o, o_1) =_d Vrai$, toutes les valeurs de a sont considérées dans les deux cas, sinon,
- si a est nominal, $C_a(o, o_1) =_d (a = v_a^1)$ donc les deux conditions sont identiques,
- si a est linéaire, $C_a(o, o_1) =_d (a \in [min(v_a^o, v_a^1), max(v_a^o, v_a^1)])$. On a $(a = v_a^1) \models C_a(o, o_2)$ implique $(a \in [min(v_a^o, v_a^1), max(v_a^o, v_a^1)]) \models C_a(o, o_2)$ et réciproquement.

#

Vérifions que $\leq_o$ est un pré-ordre.

Propriété 6.3 *La relation $\leq_o$ est un pré-ordre, c'est-à-dire $\leq_o$ est réflexive et transitive.*

Preuve : Soient e, e_1, e_2 et e_3 des objets.
Il est clair que $\leq_o$ est réflexive : $e \models C(o,e)$.

Montrons que si $e_1 \leq_o e_2$ et $e_2 \leq_o e_3$ alors $e_1 \leq_o e_3$.
D'après la propriété 6.2, $e_1 \leq_o e_2$ si et seulement si $C(o,e_1) \models C(o,e_2)$.
De même $e_2 \leq_o e_3$ si et seulement si $C(o,e_2) \models C(o,e_3)$.
Donc $C(o,e_1) \models C(o,e3)$. Et, d'après la propriété 6.2, $e_1 \leq_o e_3$. #

Reprenons l'exemple courant. On a $e_3 \leq_o e_2$ puisque $C(o,e_2) = \{(d = F)\}$ contient e_3. Notons bien que $\leq_o$ n'est en général qu'un pré-ordre partiel : e_1 et e_2, par exemple, ne sont pas comparables par rapport à $\leq_o$ puisque $C(o,e_1) = \{(a = V), (b = V), (c = V)\}$ ne contient pas e_2 et $C(o,e_2) = \{(d = F)\}$ ne contient pas e_1. Notons également que le cas des valeurs manquantes est pris en compte dans la définition ci-dessus : dès qu'un attribut n'est pas valué dans l'un des deux objets o_1 ou o_2, il est exclu de $C(o_1,o_2)$.

Considérons un autre problème de classification décrit par deux attributs linéaires continus y et z. Soient les exemples $p_1 =_d ((y = 1),(x = 0))$ et $p_2 =_d (((y = 2),(z = 1)),(x = 1))$ et l'objet $o =_d (y = 0) \wedge (z = 0)$, $C(o,p_1) = (y \in [0;1])$ et $C(o,p_2) = (y \in [0;2]) \wedge (z \in [0;1])$. L'exemple p_1 n'est pas plus proche de o que p_2, $p_1 \notin C(o,p_2)$, et p_2 n'est pas plus proche de o que p_1, $p_2 \notin C(o,p_1)$.

Ayant défini $C(o_1,o_2)$, nous pouvons maintenant définir formellement la notion de règle la plus spécifique couvrant un objet et un exemple.

Définition 6.4 *Étant donné un objet o à classer et un élément e de l'ensemble d'apprentissage EA, la règle la plus spécifique couvrant o et e est, par définition, $R(o,e) =_d C(o,e) \rightarrow (x = classe(e))$. $R(o,e)$ est pertinente pour EA par construction.*

Très clairement, toute règle couvrant o et satisfaite par e implique la règle $R(o,e)$. $R(o,e)$ est bien la règle la plus spécifique couvrant o et satisfaite par e.

Définissons à présent l'ensemble $cons(EA,\leq_o)$ des exemples e contenant o dans leur portée. e contient o dans sa portée si et seulement si il existe une règle cohérente et pertinente pour EA couvrant o et e. $R(o,e)$ est la règle la plus spécifique couvrant o et e. Elle est donc forcément pertinente pour EA. Il suffit donc de vérifier que $R(o,e)$ n'admet aucun contre-exemple dans EA. Cela revient à montrer qu'aucun exemple d'une autre classe n'est plus proche de o que e.

Définition 6.5 *Soit e un exemple de l'ensemble d'apprentissage EA et o un objet. $CE(e,EA)$ est l'ensemble des contre-exemples de e, c'est-à-dire l'ensemble des éléments de EA appartenant à des classes différentes de celle de e.*
$e \in cons(EA,\leq_o)$ si et seulement si pour tout $e' \in CE(e,EA), e' \not\leq_o e$.

Lorsque $e \in cons(EA,\leq_o)$, on dit que e est un *voisin* de o pour la classification par portée. Il est important de noter que l'on ne demande pas à e d'être plus proche de o que tous ses contre-exemples pour appartenir à $cons(EA,\leq_o)$. Il suffit qu'aucun contre-exemple ce ne soit plus proche de o que lui. Cela signifie que e est plus proche de o que ce, ou alors que e et ce sont incomparables par rapport à $\leq_o$, ce qui est le cas le plus fréquent.

Dans l'exemple des patients, e_1 et e_2 sont incomparables suivant le pré-ordre partiel $\leq_o$ où l'objet o est représenté par $(a = Vrai) \wedge (b = Vrai) \wedge (c = Vrai) \wedge (d = Faux)$. En effet, e_2 ne satisfait pas $C(o,e_1) =_d \{(a = V),(b = V),(c = V)\}$ et e_1 ne satisfait pas $C(o,e_2) = \{(d = F)\}$ ne contient pas e_1.

6.4 Correspondance avec les règles cohérentes et pertinentes

Il découle de la définition de $cons(EA, \leq_o)$ que chaque fois qu'un exemple de la classe v_x appartient à $cons(EA, \leq_o)$, une règle cohérente et pertinente pour EA classant o dans la classe v_x existe. De plus, la réciproque est vraie. Ainsi, la classification par portée est une approche à base d'instances pour la classification à base de règles ; plus formellement :

Théorème 6.1 *Soit EA un ensemble d'exemples et o un objet de l'$UNIVERS$. Il existe une règle cohérente et pertinente $R = G \rightarrow (x = v_x)$ pour EA qui classe o si et seulement si il existe un élément e de $cons(EA, \leq_o)$ tel que $classe(e) = v_x$. De plus, R implique $R(o, e)$.*

Preuve :

$\Rightarrow$ Soit une règle cohérente et pertinente $R = G \rightarrow (x = v_x)$ couvrant o, il existe un exemple e qui rend R pertinente. Montrons que e appartient à $cons(EA, \leq_o)$.
Supposons qu'il existe $e' \in CE(e, EA)$ tel que $e' \leq_o e$., c'est-à-dire $e' \in C(o, e)$.
$R(o, e) = C(o, e) \rightarrow (x = classe(e))$ est la règle la plus spécifique couvrant o et e. Donc G implique $C(o, e)$, donc G couvre e'. Puisque R est cohérente pour EA, on aboutit à une contradiction.

$\Leftarrow$ Soit $e \in cons(EA, \leq_o)$. La règle $R(o, e)$ est une règle cohérente et pertinente pour EA qui classe o.

#

Dans l'exemple canonique, $cons(EA, \leq_o) =_d \{e_1, e_3\}$ et l'ensemble des règles minimales cohérentes et pertinentes pour EA qui classent o sont : $(a = Vrai) \rightarrow (x = Vrai)$, $(b = Vrai) \rightarrow (x = Vrai)$ et $(c = Vrai) \rightarrow (x = Vrai)$. Il y a bien correspondance entre les classes proposées par l'ensemble des voisins pour la classification par portée et par l'ensemble des règles minimales cohérentes et pertinentes pour EA. Notons cependant que le nombre de voisins n'est pas nécessairement égal au nombre de règles (ce qui a, en général, une influence sur la classe finalement retenue lorsque le critère de résolution doit être utilisé).

La classification par portée s'appuie sur ce théorème pour classer un objet comme si toutes les règles minimales cohérentes et pertinentes couvrant cet objet étaient disponibles, mais en construisant $cons(EA, \leq_o)$ au lieu de générer explicitement les règles. Bien entendu, cette correspondance est seulement logique : le nombre d'exemples de $cons(EA, \leq_o)$ de classe v_x ne correspond pas en général au nombre de règles de $RMIN(EA)$ qui couvrent o et concluent sur v_x. Munis du même critère de résolution, $cons(EA, \leq_o)$ et $\{R \in RMIN(EA)$ tel que R couvre $o\}$ peuvent parfaitement suggérer des classes différentes pour o.

Clairement, la classification par portée ne requiert aucune phase d'apprentissage et elle n'a pas besoin de construire d'abstractions telles que des arbres de décision ou des règles. Cependant, des règles cohérentes et pertinentes peuvent facilement être engendrées, à la demande, mais un grand nombre de règles sont possibles. Seule une partie de la puissance explicative des approches à base de règles est préservée comme nous le verrons plus loin.

6.5 Approche « diviser pour régner »

Un algorithme naïf de calcul de $cons(EA, \leq_o)$ consiste à considérer tous les exemples incrémentalement et, pour chaque exemple e, vérifier qu'aucun exemple d'une autre classe n'est plus proche de o que lui suivant $\leq_o$. Cet algorithme requiert $O(d \times |EA|^2)$ comparaisons entre valeurs d'attributs, où d est le nombre d'attributs. Une approche « diviser pour régner » qui consiste à calculer $cons(E_1 \cup E_2, \leq_o)$ comme $cons(cons(E_1, \leq_o) \cup cons(E_2, \leq_o), \leq_o)$ est plus efficace. Malheureusement, $cons(E_1 \cup E_2, \leq_o)$ n'est qu'un sous-ensemble de $cons(cons(E_1, \leq_o)$

$\cup cons(E_2, \leq_o), \leq_o)$ en général. Donc chaque élément résultant de l'approche « diviser pour régner » doit être comparé avec tous les éléments de EA une seconde fois. L'algorithme obtenu en utilisant dans un premier temps l'approche « diviser pour régner » est malgré tout beaucoup plus efficace en pratique que l'approche naïve.

6.5.1 Algorithme de classification par portée

La fonction `chercher_cons` réalise l'algorithme utilisant l'approche « diviser pour régner » que nous venons de décrire. Nous notons $sur\_cons(o, EA)$ le résultat fourni par la recherche « diviser-pour-régner » réalisée par la fonction `chercher_sur_cons` ci-après. Bien entendu, ce sur-ensemble de $cons(EA, \leq_o)$ n'est pas défini mathématiquement, son contenu dépend des partitions réalisées dans `chercher_sur_cons`.

fonction chercher_cons(o, E)
```
Soit E1 le résultat de chercher_sur_cons(o, E)
Initialiser E2 à l'ensemble vide
Pour tout e1 appartenant à E1 faire
  sortir := faux
  Pour tout exemple e2 de E jusqu'à sortir faire
    Si classe(e1) ≠ classe(e2) et e2 ≤o e1
    Alors sortir := vrai
  Si (sortir n'est pas vrai) Alors E2 := E2 ∪ {e1}
Rendre(E2)
```

fonction chercher_sur_cons(o, E)
```
Si E ne contient qu'un élément
Alors Rendre(E)
Sinon
  diviser E en deux ensembles E1 et E2 de même taille (plus ou moins un)
  E1' := chercher_sur_cons(o, E1)
  E2' := chercher_sur_cons(o, E2)
Rendre (fusionner (o, E1', E2'))
```

fonction fusionner(o, E_1, E_2)
```
Initialiser E' à l'ensemble vide
Pour tout e1 appartenant E1 faire
  sortir := faux
  Pour tout exemple e2 de E2 jusqu'à sortir faire
    Si classe(e1) ≠ classe(e2) et e2 ≤o e1
    Alors sortir := vrai
  Si (sortir n'est pas vrai) Alors E' := E' ∪ {e1}
Pour tout e2 appartenant E2 faire
  sortir := faux
  Pour tout exemple e1 de E1 jusqu'à sortir faire
    Si classe(e2) ≠ classe(e1) et e1 ≤o e2
    Alors sortir := vrai
  Si (sortir n'est pas vrai) Alors E' := E' ∪ {e2}
Rendre(E')
```

6.5.2 Correction de l'algorithme

Vérifions que l'algorithme est correct. Montrons pour cela que le résultat de l'application de la fonction `chercher_sur_cons` à EA contient bien $cons(EA, \leq_o)$.

Théorème 6.2 $cons(E_1 \cup E_2, \leq_o) \subseteq cons(cons(E_1, \leq_o) \cup cons(E_2, \leq_o), \leq_o)$

Preuve : Soit $e \in cons(E_1 \cup E_2, \leq_o)$. Montrons que $e \in cons(cons(E_1, \leq_o) \cup cons(E_2, \leq_o), \leq_o)$.
$e \in E_1 \cup E_2$, donc $e \in E_1$ ou $e \in E_2$.
Supposons $e \in E_1$. Montrons que $e \in cons(E_1, \leq_o)$.
Supposons qu'il existe $e' \in CE(e, E_1)$ tel que $e' \leq_o e$. $E_1 \subseteq E_1 \cup E_2$, donc $e' \in CE(e, E_1 \cup E_2)$.
$e' \leq_o e$ contredit $e \in cons(E_1 \cup E_2, \leq_o)$.
$e \in cons(E_1, \leq_o)$, montrons que $e \in cons(cons(E_1, \leq_o) \cup cons(E_2, \leq_o), \leq_o)$,
Supposons qu'il existe $e' \in CE(e, cons(E_1, \leq_o) \cup cons(E_2, \leq_o))$ tel que $e' \not\leq_o e$.
$cons(E_1, \leq_o) \cup cons(E_2, \leq_o) \subseteq E_1 \cup E_2$, donc $e' \in CE(e, E_1 \cup E_2)$.
$e' \leq_o e$ contredit $e \in cons(E_1 \cup E_2, \leq_o)$. #

La réciproque est fausse. Considérons les objets suivants (tous les attributs sont booléens)
$o = abcd, e_1 = abc\bar{d}\bar{x}, e_2 = ab\bar{c}\bar{d}\bar{x}, e_3 = a\bar{b}\bar{c}\bar{d}x$.
$cons(\{e_1, e_2\} \cup \{e_3\}, \leq_o) = \{e_1\}$
$cons(cons(\{e_1, e_2\}, \leq_o) \cup cons(\{e_3\}, \leq_o), \leq_o) = cons(\{e_1\} \cup \{e_3\}, leq_o) = \{e_1, e_3\}$

6.5.3 Complexité de l'algorithme

Analytiquement, la complexité en espace de *chercher_cons* est en $\Theta(d \times N)$ dans tous les cas. Sa complexité en temps est en $O(d \times C(N))$, où $C(N)$ est le nombre de comparaisons entre objets par rapport à $\leq_o$. $C(N)$ est la somme du nombre $C_1(N)$ de comparaisons requises par *chercher_sur_cons*(o, EA) et du nombre $C_2(N)$ de comparaisons requises pour supprimer les éléments de *sur_cons*(o, EA) qui n'appartiennent pas à $cons(EA, \leq_o)$. Le pire cas est obtenu quand les exemples ne sont pas comparables par rapport à $\leq_o$, car tout élément de EA appartient à $cons(EA, \leq_o)$. Dans cette situation, $C_1(N) = 2 \times N/2 \times N/2 + 2 \times C_1(N/2)$. Cette équation se simplifie asymptotiquement en $C_1(N) = \Theta(N^2)$. Dans le pire cas, $C_2(N) = N \times (N-1)$, donc $C(N) = \Theta(2 * N^2)$. Dans le meilleur cas, $cons(EA, \leq_o)$ ne contient qu'un élément à chaque étape, donc $C_1(N) = 2 + 2 \times C_1(N/2)$, ce qui se simplifie en $C_1(N) = 2(N-1)$. Dans ce cas, $C_2(N) = N$; ainsi, la complexité en temps de *chercher_cons* est linéaire en la taille de EA dans le meilleur cas. Puisque l'on utilise l'idée de « diviser pour régner », on peut espérer obtenir une complexité moyenne en $O(d \times N \times log_2(N))$ pour certaines distributions des exemples. Cela se vérifie expérimentalement (figure 6.1).

Considérons un problème de classification composé de 10 attributs booléens et une classe booléenne. La fonction cible consiste en une forme normale disjonctive générée aléatoirement comportant 20 conjonctions d'au plus 5 littéraux. L'ensemble d'apprentissage est constitué de N individus complètement décrits tirés aléatoirement, mais sans remise, parmi les 1024 individus possibles. Chaque problème de classification, fonction cible et ensemble d'apprentissage de taille N, est évalué en moyennant les résultats sur 10 problèmes générés aléatoirement. La figure 6.1 montre clairement que le nombre moyen de comparaisons $C(N)$ est proche de $N \times log_2(N)$ et son maximum est nettement plus petit que le pire cas (i.e. $2 \times N^2$).

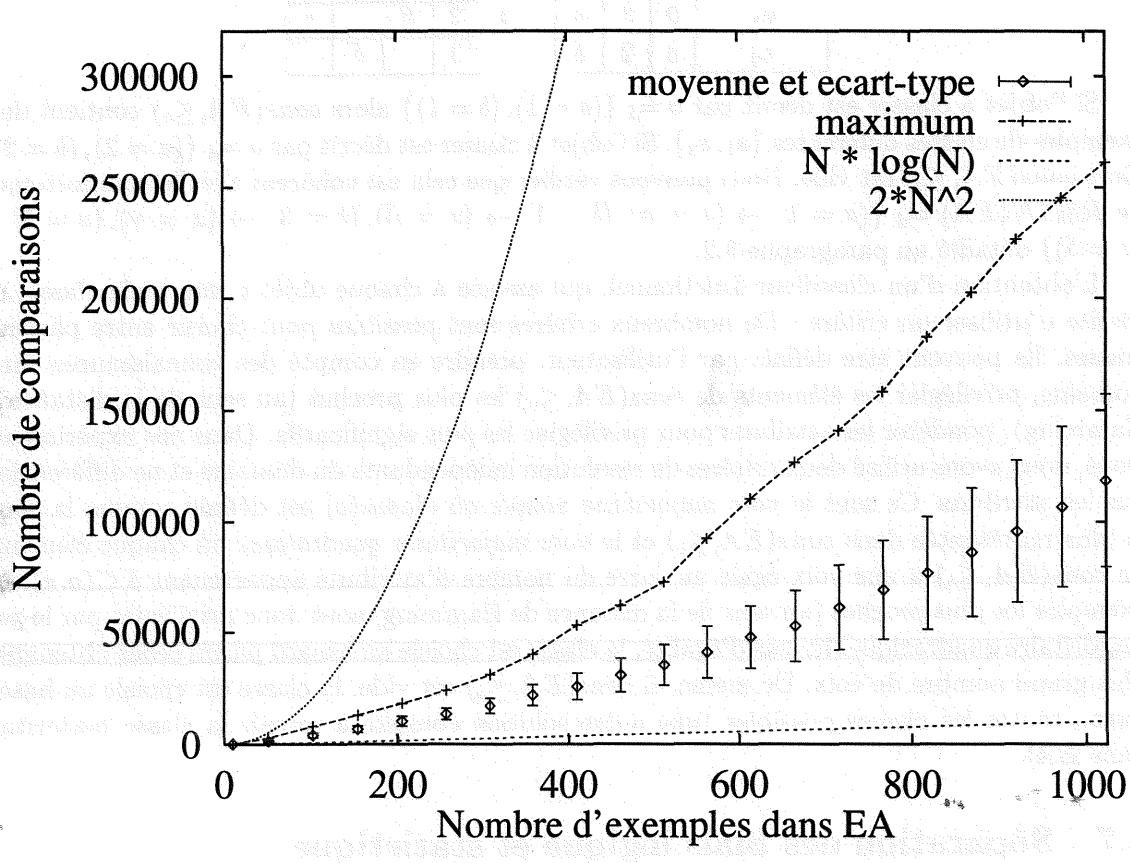


FIG. 6.1 – *Complexité en temps de la classification par portée (expérimentale). Moyenne, écart-type et maximum du nombre de comparaisons entre objets effectuées.*

6.6 Critères de résolution

$cons(EA, \leq_o)$ peut contenir des exemples étiquetés par des classes différentes ou être vide. Reprenons l'exemple du paragraphe 5.2 : les individus sont décrits à l'aide de deux attributs nominaux a et b pouvant prendre les valeurs 1, 2 ou 3. L'attribut de classe x peut prendre les valeurs α, β, γ ou δ. L'ensemble d'apprentissage est constitué des quatre exemples ci-dessous. L'$UNIVERS$ est représenté sur le tableau de Karnaugh joint.

exemple	a	b	x
e_1	1	2	α
e_2	2	1	β
e_3	2	3	γ
e_4	3	2	δ

		b		
		1	2	3
a	1		α	
	2	β		γ
	3		δ	

Si l'objet à classer est décrit par $o =_d \{(a = 1), (b = 1)\}$ alors $cons(EA, \leq_o)$ contient deux exemples de classes différentes $\{e_1, e_2\}$. Si l'objet à classer est décrit par $o =_d \{(a = 2), (b = 2)\}$, alors $cons(EA, \leq_o)$ est vide. Nous pouvons vérifier que cela est cohérent avec le comportement de $RMIN(EA) =_d \{(a = 1) \rightarrow (x = \alpha), (b = 1) \rightarrow (x = \beta), (b = 3) \rightarrow (x = \gamma), (a = 3) \rightarrow (x = \delta)\}$ détaillé au paragraphe 5.2.

L'obtention d'un classifieur fonctionnel, qui associe à chaque objet o une seule classe, nécessite d'utiliser un critère . De nombreux critères sont possibles pour choisir entre plusieurs classes. Ils peuvent être définis par l'utilisateur, prendre en compte des connaissances sur le domaine, privilégier les éléments de $cons(EA, \leq_o)$ les plus proches (au sens de la distance de Hamming), pondérer les attributs pour privilégier les plus significatifs. Dans nos expérimentations, nous avons utilisé deux critères de résolution indépendants du domaine et ne différenciant pas les attributs. Ce sont le *vote majoritaire simple* où $classe(o)$ est définie comme la classe la plus représentée dans $cons(EA, \leq_o)$ et le *vote majoritaire quadratique* où chaque élément e de $cons(EA, \leq_o)$ a une voix égale au carré du nombre d'attributs appartenant à $C(o, e)$. Les exemples les plus proches (au sens de la distance de Hamming) sont donc privilégiés par le vote majoritaire quadratique. En cas d'égalité, la classe est choisie au hasard parmi celles obtenant le plus grand nombre de voix. De même, si $cons(EA, \leq_o)$ est vide, la classe est choisie au hasard parmi toutes les classes possibles (une autre solution consiste à retenir la classe majoritaire dans EA).

6.7 Séparation des biais logique et statistique

La classification par portée sépare bien les biais logique et statistique de la classification. Un biais logique est utilisé pour le calcul de $cons(EA, \leq_o)$. Comme nous l'avons vu, il met en œuvre un principe de similarité qui est un point commun entre la similarité à base d'instances et la similarité à base de règles. Il permet de classer comme en utilisant $RMIN(EA)$ sans générer explicitement aucune règle, uniquement en comparant les exemples entre eux. Le calcul de $cons(EA, \leq_o)$ ne fait intervenir aucun biais numérique. Un biais numérique n'intervient qu'en complément pour résoudre les conflits possibles ou suggérer une classe lorsque $cons(EA, \leq_o)$ est vide.

Les biais logique et numérique utilisés sont donc clairement identifiés et séparés dans notre approche. Ce n'est pas le cas, par exemple, des arbres de décision qui font intervenir un critère statistique, le gain d'information, en cours de construction des règles. Une bonne compréhension des biais utilisés permet d'éviter d'utiliser une technique de classification sur un domaine pour lequel il apparaît clairement que les biais ne sont pas adaptés.

6.8 Pouvoir explicatif: génération de règles à la demande

Bien que la classification par portée n'ait pas besoin de construire d'abstractions telles que des arbres de décision ou des règles, des règles cohérentes et pertinentes peuvent facilement être engendrées, à la demande, et donc une partie de la puissance explicative des approches à base de règles est préservée. Ainsi, pour chaque e de $cons(EA, \leq_o)$, la règle $R(o,e) = (C(o,e) \rightarrow (x = classe(e)))$ est une règle cohérente et pertinente pour EA. $R(o,e)$ est non triviale, c'est-à-dire elle couvre plus d'un objet si o n'appartient pas à EA. De plus, la règle $R(o,e)$ peut être généralisée en $O(|EA| \times |C(o,e)|^2)$ comparaisons entre valeurs d'attributs :

```
fonction généraliser_règle(R,E)
    Pour tout attribut a faire
        Enlever l'éventuelle condition sur l'attribut a
        de la partie gauche de la règle R.
        Si la règle R' obtenue n'admet pas de contre-exemple dans E
            Alors R = R'
            Sinon conserver la condition sur a dans R
    Rendre(R)
```

La règle finalement obtenue en appliquant la fonction **généraliser_règle** à la règle $R(o,e)$ pour un élément e de $cons(EA, \leq_o)$ est une règle minimale cohérente et pertinente pour EA qui implique $R(o,e)$.

Propriété 6.4 *Soit $e \in cons(EA, \leq_o)$. Le résultat de* **généraliser_règle**$(R(o,e), EA)$ *est une règle minimale cohérente et pertinente pour EA.*

Preuve : la règle $R(o,e)$ est pertinente pour EA puisqu'elle est satisfaite par l'exemple e. L'exemple e satisfait toute règle R' obtenue en éliminant une condition de la partie gauche de $R(o,e)$. Donc la règle finale est pertinente pour EA.

Par construction, la règle R' est cohérente pour EA. Donc la règle finale est cohérente pour EA.

Supposons que la règle finale RES ne soit pas minimale, c'est-à-dire que la condition sur au moins un attribut a' puisse être supprimée tout en conservant une règle cohérente et pertinente. Soit $\mathcal{R}$ une règle minimale impliquant R. Supposons que a' soit le premier attribut que l'on puisse supprimer de RES et plaçons nous « au moment » où cet attribut est considéré : la règle R' obtenue après suppression de la condition sur a' est impliquée par $\mathcal{R}$, tout objet couvert par R' est couvert par $\mathcal{R}$. Or, la condition sur a' n'a pas été supprimée dans RES, c'est-à-dire R' admet au moins un contre-exemple. Donc $\mathcal{R}$ admet au moins un contre-exemple. Ceci contredit la cohérence de $\mathcal{R}$.
Donc la règle finale est minimale. #

D'autres procédures de généralisation peuvent être également utilisées, de façon, par exemple, à privilégier des attributs, ou à utiliser d'autres critères liés au domaine d'application et définis par l'utilisateur.

Reprenons l'exemple des patients pour illustrer ce mécanisme. Le patient décrit par $(a = V) \wedge (b = V) \wedge (c = V) \wedge (d = F)$ appartient à la portée de e_3, donc il peut être classé dans la classe $(x = V)$. Si besoin est, la règle cohérente et pertinente $(a = V) \wedge (d = F) \rightarrow (x = V)$ peut être engendrée afin d'expliquer la classe proposée. Cette règle peut même être généralisée en la règle minimale $(a = V) \rightarrow (x = V)$ (cf. figure 6.2).

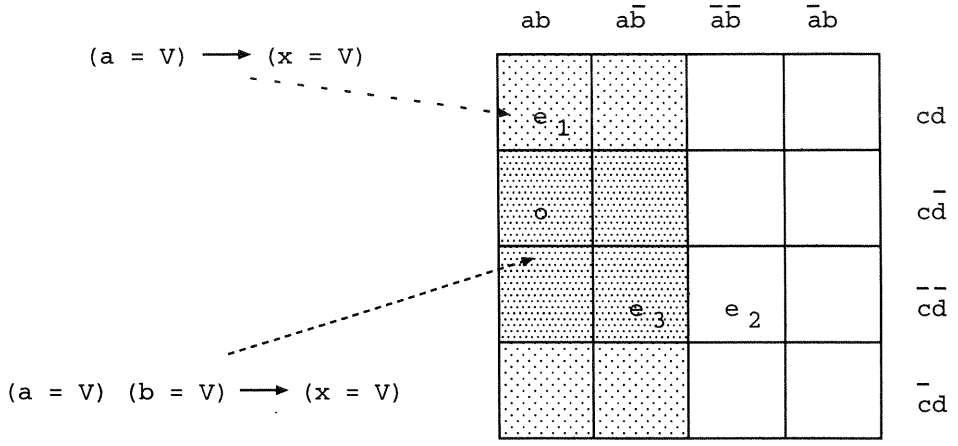


FIG. 6.2 – *Génération de règles explicatives.*

6.9 Résumé et conclusion

Dans ce chapitre, nous avons proposé une nouvelle technique de classification appelée classification par portée. La portée d'un exemple e est l'ensemble des points couverts par toutes les règles cohérentes et pertinentes couvrant e. C'est une forme d'étoile des algorithmes AQ [Michalski, 1983], et plus précisement un espace des versions $H(e)$ de l'espace des versions disjonctif. Si la portée d'un exemple e de EA contient l'objet o à classer, cela indique qu'il existe une règle cohérente et pertinente pour EA couvrant o. L'intérêt de la classification par portée est de ne pas calculer l'ensemble de toutes les règles cohérentes et pertinentes pour EA explicitement. Dans cette approche, la portée n'est pas calculée explicitement, ni même implicitement comme dans l'approche de l'espace des versions disjonctif. La classification par portée construit simplement la règle la plus spécifique couvrant o et e et vérifie qu'elle est cohérente (elle est forcément pertinente). Si tel est le cas, o appartient à la portée de e. La classification par portée ne construit aucune abstraction à partir des exemples. C'est donc une approche à base d'instances. Elle procède à l'aide de comparaisons entre les individus. Le biais inductif sur lequel elle s'appuie est exactement celui des ensembles de toutes les règles minimales cohérentes et pertinentes pour EA. La classification par portée constitue un point commun entre les approches à base d'instances, fondées ou non sur des distances, les approches à base de règles, et celles fondées sur des espaces des versions.

Nous avons proposé un algorithme de classification suivant le principe de « diviser pour régner ». Cet algorithme permet d'obtenir, dans de nombreux cas, une complexité en temps inférieure à $d \times |EA|^2$. Dans la classification par portée, comme dans les approches « équivalentes » (classifieur à base de toutes les règles cohérentes et pertinentes ou espace des versions disjonctif), les biais logique et statistique utilisés sont clairement séparés. La construction de l'ensemble des voisins, $cons(EA, \leq_o)$ pour la classification par portée, $\{e/o \in H(e)\}$ pour l'espace des versions disjonctif, $\{R =_d G \rightarrow (x = v_x)/R$ est cohérente et pertinente pour EA et $o \models G\}$ dans le cas des classifieurs à base de toutes les règles cohérentes et pertinentes pour EA, s'effectue uniquement sur la base de biais logiques : cohérence et pertinence avec l'ensemble d'apprentissage. Le biais statistique ne sert qu'à résoudre des conflits entre plusieurs hypothèses contradictoires de même valeur logique. Nous allons montrer dans le chapitre suivant comment ces biais logiques peuvent être modulés par des biais numériques pour s'adapter davantage aux

données réelles.

Chapitre 7

Extensions de la classification par portée

Nous présentons dans ce chapitre plusieurs extensions de la classification par portée. Nous considérons dans une première partie les difficultés que présentent les problèmes de classification à partir de données réelles. Nous proposons une modification simple des pré-ordres utilisés dans la classification par portée afin de prendre en compte les exemples incohérents, résultant par exemple d'une mauvaise représentation des exemples. Nous rappelons les paramètres de modulation de la cohérence proposés par Michèle Sebag [Sebag, 1996] et nous proposons plusieurs interprétations du paramètre de généralité dans l'espace des versions, mais aussi dans la classification par portée. Nous proposons également un premier protocole de réglage automatique de ces paramètres. Nous présentons dans une seconde partie une extension de la classification par portée à la classification à partir de règles à la place d'exemples ; nous proposons dans cette optique plusieurs stratégies de généralisation des exemples en règles. Nous citons dans un troisième volet deux extensions supplémentaires de la classification par portée. La première est la prise en compte explicite des redondances ; elle permet un gain en temps de classification sans modifier les biais logique et statistique utilisés. La seconde concerne la prise en compte des exemples négatifs qui peuvent survenir dans le cas d'une utilisation incrémentale de la classification par portée en présence d'un oracle.

7.1 Adaptation du modèle de la classification par portée aux données réelles

Dans les problèmes de classification « réels », les associations description/classe sont souvent non déterministes. Il y a au moins deux raisons à cela : une mauvaise description des données ou la présence de bruit. Nous montrons comment les exemples incohérents entre eux peuvent être pris en compte dans la classification par portée.

7.1.1 Bruit et non-déterminisme

Pour que la classification soit possible, chaque individu doit appartenir à une seule classe. Toutefois, une même description peut correspondre à plusieurs exemples appartenant à des classes différentes. Le lien entre la description et la classe n'est donc pas nécessairement fonctionnel ou déterministe. En pratique, il l'est même rarement car les descriptions des exemples sont souvent erronées ou il manque des attributs pertinents pour la classification.

Étant donné un problème de classification, un exemple est *incorrect* si la classe associée à sa description n'est pas la bonne. Puisque les seules données dont on dispose sont des couples (*description*, *classe*), l'induction se fonde sur l'association d'une classe à une description donnée et donc elle est souvent induite en erreur (sic) lorsque la classe associée à une description n'est pas correcte. Le bruit présent dans les problèmes réels n'affecte pas la classe de l'exemple, mais sa description : un exemple *bruité* est un exemple dont la classe est correcte, mais dont la description est incorrecte.

Considérons le problème de classification : déterminer si une configuration d'un affichage de sept diodes électro-luminescentes représente un chiffre (cf. figure 7.1). Chacun des sept attributs

FIG. 7.1 – *Un afficheur à sept diodes électro-luminescentes (LED).*

correspond à une des diodes et indique si elle est allumée ou éteinte. Supposons que la diode verticale inférieure gauche soit défectueuse. La description  du chiffre 2 est un exemple bruité et incorrect de chiffre. La description  du chiffre 8 est un exemple bruité, mais correct de chiffre.

On pourrait croire que si la classe associée par la fonction cible à la description bruitée d'un exemple est la même que celle associée à l'exemple, l'exemple est correct, sinon il est incorrect. Le problème est qu'en présence de bruit, la notion de fonction cible est utopique. En effet, lorsque l'on veut classer un nouvel individu, sa description peut elle-même être bruitée. Est-il possible de s'appuyer sur la description bruitée de l'objet à classer pour la comparer aux descriptions bruitées des exemples? Ce serait possible si les descriptions étaient bruitées de la même façon, si le bruit était déterministe. Malheureusement, le bruit n'est, en général, pas déterministe, mais aléatoire. Il n'est donc pas possible de raisonner sur les descriptions bruitées et de déterminer une fonction associant une description bruitée à sa classe. Comment procède donc l'esprit humain? Il utilise bien une fonction cible qui associe un exemple à une classe. Il ne raisonne pas sur la description, mais sur l'individu réel au complet, c'est-à-dire que soit il est capable de modéliser le bruit et de débruiter la description pour reconnaître l'individu réel, soit il utilise des informations supplémentaires. Dans ce dernier cas, les descriptions utilisées sont incomplètes : il manque des attributs pertinents pour la classification. Une chose est certaine, mis en face de descriptions abstraites d'exemples et d'un individu à classer, c'est-à-dire où les noms d'attributs et leurs valeurs ont été remplacés par des symboles sans sémantique particulière, donc sans possibilité d'accéder à des informations supplémentaires, une intelligence naturelle peut difficilement faire mieux que recourir à des principes d'induction. Deux types de biais sont possibles : les biais logiques et les biais statistiques. Nous avons jusqu'à présent utilisé un biais inductif purement logique et un biais statistique pour lever les conflits. Il est clair qu'en présence de données bruitées, les contraintes logiques doivent être relâchées afin d'autoriser une certaine dose d'incohérence. Les paragraphes suivants présentent quelques relaxations possibles.

7.1.2 Prise en compte d'exemples incohérents

Nous avons supposé dans les chapitres précédents que l'association description/classe est déterministe et, en particulier, que l'ensemble d'apprentissage est cohérent, c'est-à-dire qu'une

classe unique est associée à chaque exemple complètement décrit.

Notons pour commencer que la description d'un objet est incomplète quand des attributs ou des valeurs nécessaires pour le séparer d'autres objets manquent. Considérons un exemple de réglementation routière où la classe est *être en infraction.* Prenons le cas de deux exemples de voitures bleues grillant un feu rouge dont l'une est en infraction et pas l'autre :
$(couleur = bleue), (infraction = Vrai)$ et $(couleur = bleue), (infraction = Faux)$
Si le seul attribut considéré est la couleur, ce sont deux exemples incohérents entre eux. Cela peut provenir de l'incomplétude de la description si un attribut est manquant, indiquant par exemple, si le véhicule est une voiture de police. Les exemples complétés ne sont plus incohérents entre eux :
$(couleur = bleue), (police = Faux), (infraction = Vrai)$ et $(couleur = bleue), (police = Vrai), (infraction = Faux)$.
Il faut distinguer le cas des attributs manquants et celui des valeurs manquantes. En effet, si l'on sait que les individus sont décrits par les attributs *couleur* et *police* alors les valeurs manquantes des exemples :
$(couleur = bleue), (police =?), (infraction = Vrai)$ et $(couleur = bleue), (police =?), (infraction = Faux)$
ne rendent pas ces exemples incohérents entre eux. Deux descriptions incomplètes identiques ne désignent pas nécessairement le même individu complètement décrit.

Un ensemble d'apprentissage peut contenir des exemples incohérents entre eux. Cela est problématique pour toutes les approches de classification fondées sur le biais logique de cohérence. En effet, aucune hypothèse cohérente pour l'ensemble d'apprentissage ne peut couvrir deux exemples incohérents entre eux. la présence d'exemples incohérents entre eux nuit à la généralité des hypothèses cohérente possibles, donc limite le nombre d'objets couverts par de telles hypothèses. La présence d'exemples incohérents entre eux perturbe également le biais logique de pertinence. En effet, une hypothèse cohérente et pertinente pour EA ne peut être rendue pertinente par un exemple incohérent avec un autre. La présence d'exemples incohérents entre eux limite l'ensemble des hypothèses cohérentes et pertinentes pour EA.

Pour pallier le problème de sur-spécificité des règles cohérentes lié à la présence d'exemples incohérents entre eux, une première solution consiste à supprimer tous les exemples incohérents entre eux. Garder une seule classe par description nécessite idéalement de deviner la « bonne ». Pour ce faire, dans le cas d'un ensemble d'apprentissage comportant un grand nombre de redondances, il est possible de retenir la classe la plus souvent associée à une description donnée et de supprimer seulement les exemples appartenant aux autres classes. Cette solution met en œuvre un biais statistique. Nous préconisons d'éviter autant que possible d'éliminer de l'information de l'ensemble d'apprentissage lorsque l'on ne sait pas s'il existe vraiment une liaison déterministe entre description et classe.

Supposons que nous disposions de plusieurs exemples ayant la même description et appartenant à des classes différentes. Dans la classification par portée, il est possible de conserver cette information lors de la construction de $cons(EA, \leq_o)$ de façon à la prendre en compte lors du vote final : un individu retenu apporterait sa voix à plusieurs classes. Le problème qui se pose alors est que la construction de $cons(EA, \leq_o)$ nécessite de connaître la classe de chaque description, puisque l'on compare un exemple e avec tous ses contre-exemples $CE(e, EA)$. Une solution est d'utiliser $min(EA, \leq_o)$ à la place de $cons(EA, \leq_o)$. En effet, la recherche des descriptions les plus proches de o au sens de $\leq_o$ ne fait pas intervenir la classe des exemples. Malheureusement, $min(EA, \leq_o)$ est un sous-ensemble de $cons(EA, \leq_o)$ et son utilisation fait perdre l'information apportée par les « voisins » de o qui ne sont pas forcément les plus proches de o pour $\leq_o$.

La solution que nous avons finalement retenue pour mieux prendre en compte les exemples incohérents dans la classification par portée est la suivante. Soient deux exemples e_1 et e_2 incohérents entre eux. La situation est représentée figure 7.2 dans un $UNIVERS$ décrit par deux attributs linéaires continus.

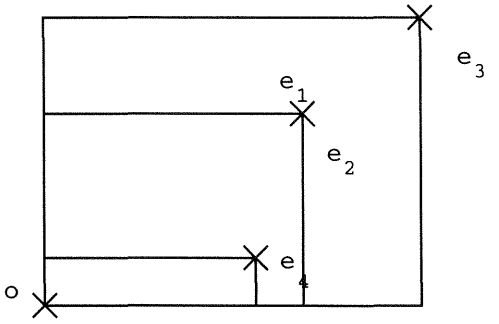


FIG. 7.2 – *Influence des exemples incohérents lors de la classification par portée.*

On a $e_1 \leq_o e_2$ et $e_1 \in CE(e_2, EA)$ donc $e_2 \notin cons(EA, \leq_o)$, et réciproquement $e_1 \notin cons(EA, \leq_o)$. Les deux exemples s'éliminent mutuellement. Alors que l'on conçoit qu'un contre-exemple strictement plus proche de o que e_1 l'élimine, il est dommage qu'il soit éliminé par un contre-exemple ayant même description que lui. Supposons, en effet, qu'il n'existe aucun e_4 tel que $e_4 \leq_o e_1$. Puisque e_1 et e_2 sont de classes différentes, ils éliminent tous les exemples e_3 tels que $e_1 \leq_o e_3$. Si e_1 et e_2 s'éliminent mutuellement, on peut obtenir un ensemble de voisins, $cons(EA, \leq_o)$, vide. Lorsque l'on applique le critère de résolution, la classe de o est tirée au sort parmi toutes les classes possibles. C'est catastrophique quand, par exemple, e_1 appartient à la classe 1 et e_2 à la classe 2 et que le tirage au sort donne la classe 7 parmi 10 classes possibles. C'est d'autant plus dommage que, dans de tels cas, o est souvent de classe 1 ou 2 si l'ensemble d'apprentissage est « représentatif » de la fonction cible sur l'$UNIVERS$. C'est encore plus dommage si o est de classe 1, par exemple, et que e_1 apparaît 5 fois dans EA et e_2 une seule. Nous proposons donc de modifier $cons(EA, \leq_o)$ en $cons(EA, <_o)$ pour mieux prendre en compte les exemples incohérents entre eux dans la classification par portée.

Définition 7.1 *$o_1 <_o o_2$ si et seulement si $o_1 \in C(o, o_2)$ et $o_2 \notin C(o, o_1)$.*
$e \in cons(EA, <_o)$ si et seulement si pour tout contre-exemple $ce \in CE(e, EA)$, $ce \not<_o e$

Considérons un ensemble d'apprentissage constitué uniquement de deux exemples incohérents entre eux e_1 et e_2. L'ensemble des voisins de tout objet o est vide ou non, suivant que le pré-ordre partiel utilisé est $\leq_o$ ou $<_o$: on a $cons(e_1, e_2\}, \leq_o) =_d \emptyset$ alors que $cons(\{e_1, e_2\}, <_o) =_d \{e_1, e_2\}$.

L'utilisation du pré-ordre partiel strict $<_o$ permet de prendre en compte un ensemble d'apprentissage incohérent sans éliminer d'information lors de la « phase logique », c'est-à-dire la construction des voisins de o, et donc de pouvoir l'utiliser lors de la « phase statistique », lorsque le critère de résolution entre les voisins de o est utilisé. Cette approche permet donc de prendre en compte les cas où le langage de représentation des exemples est incomplet et ne permet pas de discriminer deux exemples de classes distinctes partageant la même description. Elle ne résout pas les difficultés liées à la présence de descriptions d'exemples bruitées.

7.2 Modulation des biais logiques

Pour mieux prendre en compte la présence de descriptions d'exemples bruitées, Michèle Sebag a proposé deux paramètres de modulation des biais logiques [Sebag, 1996]. Le premier concerne spécifiquement le réglage de la cohérence, le second règle la généralité. Ces deux facteurs sont liés. Dans [Sebag, 1996], ces paramètres sont réglés manuellement. Nous proposons de les régler de façon automatique et présentons des résultats préliminaires sur un tel réglage automatique.

7.2.1 Paramètre de cohérence (ϵ)

En présence d'exemples bruités, un biais statistique raisonnable est d'accepter que chaque règle cohérente pour EA admette un certain nombre de contre-exemples.

Définition 7.2 *Soit $NI(e, EA)$ le nombre de contre-exemples plus proches de o que e, $NI(e, EA) = |\{ce \in CE(e, EA)/ce \leq_o e\}|$.*
Soit ϵ le taux de contre-exemples acceptés lors de la recherche des voisins d'un objet à classer o. On dit que e est un ϵ-voisin de o, noté $e \in \epsilon - cons(EA, \leq_o)$, si et seulement si $NI(e, EA) \leq \epsilon \times |EA|$.

Considérons le problème de classification représenté sur la figure 7.3. e_1 est un 0.2-voisin de o alors que e_1 n'appartient pas à $cons(\{e_1, e_2, e_3, ce_1, ce_2\}, \leq_o)$.

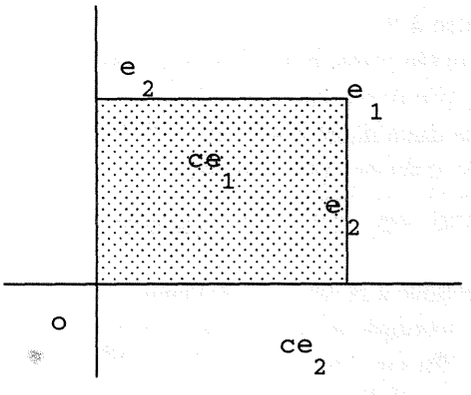


FIG. 7.3 – *Exemple de ϵ-voisin.*

ϵ modélise clairement le degré de relaxation de la propriété de cohérence : toute règle cohérente pour EA accepte au plus $\epsilon \times |EA|$ contre-exemples. La classification par portée pure correspond au cas où $\epsilon = 0$.

L'approche « diviser pour régner » peut toujours s'appliquer avec les ϵ-voisins à condition que le nombre d'incohérences autorisé ne soit relatif qu'à la taille de l'ensemble d'apprentissage initial et ne varie donc pas au cours du déroulement de l'algorithme :

Théorème 7.1 *Pour tout sous-ensemble E de l'ensemble d'apprentissage EA, un exemple e appartient à $\epsilon - cons(E, \leq_o)$ si et seulement si $NI(e, E) \leq \epsilon \times |EA|$.*
$\epsilon - cons(E_1 \cup E_2, \leq_o) \subseteq \epsilon - cons(\epsilon - cons(E_1, \leq_o) \cup \epsilon - cons(E_2, \leq_o), \leq_o)$

Preuve : soit $e \in \epsilon - cons(E_1 \cup E_2, \leq_o)$, donc $NI(e, E_1 \cup E_2) \leq \epsilon \times |EA|$.
Supposons $e \in E_1$, $E_1 \subseteq E_1 \cup E_2$, donc $NI(e, E_1) \leq NI(e, E_1 \cup E_2)$.
Donc $NI(e, E_1) \leq \epsilon \times |EA|$, donc $e \in \epsilon - cons(E_1, \leq_o)$.
$\epsilon - cons(E_1, \leq_o) \cup \epsilon - cons(E_2, \leq_o) \subseteq E_1 \cup E_2$, donc $NI(e, \epsilon - cons(E_1, \leq_o) \cup \epsilon - cons(E_2, \leq_o)) \leq NI(e, E_1 \cup E_2)$
et $NI(e, E_1 \cup E_2) \leq \epsilon \times |EA|$ donc $e \in \epsilon - cons(\epsilon - cons(E_1, \leq_o) \cup \epsilon - cons(E_2, \leq_o), \leq_o)$ #

7.2.2 Paramètre de généralité (M)

Soient o un objet à classer et e un exemple de EA. Un élément ce de EA contredit la règle $C(o,e) \rightarrow (x = classe(e))$ s'il satisfait la partie gauche $C(o,e)$ et si $classe(ce) \neq classe(e)$. $ce \in C(o,e)$ si et seulement si, pour tout attribut a, on a $v_a^{ce} \in C_a(o,e)$. Le bruit affectant les valeurs réelles de certains attributs, il peut être intéressant de ne pas systématiquement considérer tous les attributs lorsqu'on compare deux exemples par rapport à $\leq_o$. Pour cela, il suffit d'accepter que M attributs ne satisfassent pas $v_a^{ce} \in C_a(o,e)$.

Définition 7.3 *e_1 est plus proche de o que e_2 à M attributs près, noté $e_1 \leq_o^M e_2$, si et seulement si e_1 appartient à $C(o,e_2)$ à M attributs près, c'est-à-dire $|\{a/v_a^{e_1} \notin C_a(o,e_2)\}| < M$.*
e est un M-voisin de o, noté $e \in M - cons(EA, \leq_o)$, si et seulement si $e \in cons(EA, \leq_o^M)$.

Considérons deux exemples décrits par quatre nombres réels : $e_1 =_d (a_1 = 1) \wedge (a_2 = 1) \wedge (a_3 = 1) \wedge (a_4 = 3)$ et $e_2 =_d (a_1 = 2) \wedge (a_2 = 2) \wedge (a_3 = 2) \wedge (a_4 = 2)$. e_1 est plus proche de $o =_d (a_1 = 0) \wedge (a_2 = 0) \wedge (a_3 = 0) \wedge (a_4 = 0)$ que e_2 à 2 attributs près, c'est-à-dire le nombre d'attributs de e_1 dont la valeur n'est pas plus proche de celle de o que celle de e_2 (l'attribut a_4) est strictement inférieure à 2.

La classification par portée pure, $cons(EA, \leq_o)$, est obtenue pour $M = 1$. Notons que la relation $\leq_o^M$ n'est pas un pré-ordre car elle n'est pas transitive. Néanmoins la propriété de transitivité n'intervient pas dans notre approche. L'utilisation de la relation $\leq_o^M$ est également compatible avec l'approche « diviser pour régner ».

Théorème 7.2 $M - cons(E_1 \cup E_2, \leq_o) \subseteq M - cons(M - cons(E_1, \leq_o) \cup M - cons(E_2, \leq_o), \leq_o^{\acute{e}})$

Preuve : la preuve est semblable à la preuve précédente. #

Puisque M autorise un exemple e_1 à être plus proche de o qu'un exemple e_2 à M attributs près, il est plus facile pour un contre-exemple d'être couvert par une règle. Donc, plus M est grand, plus $cons(EA, \leq_o)$ se réduit.

Donnons une autre interprétation à M :

Définition 7.4 *Soient o un objet à classer, e un exemple, ce un contre-exemple de e, $ce \in CE(e, EA)$.*
La distance $d(o, e, ce)$ entre ce et l'hyper-rectangle $C(o,e)$ est le nombre d'attributs pour lesquels ce n'appartient pas à $C(o,e)$: $d(o, e, ce) = |\{a/v_a^{ce} \notin C_k(o,e)\}|$.

La mesure de spécificité $s(o,e)$ d'un hyper-rectangle $C(o,e)$ est la valeur minimale de $d(o, e, ce)$ pour l'ensemble des contre-exemples $CE(e, EA)$: $s(e, o) = min_{ce \in CE(e,EA)} d(o, e, ce)$.

$s(o,e)$permet de mesurer la spécificité de la règle $R(o,e)$. En particulier, si $s(o,e) = 0$, la règle $R(o,e) =_d C(o,e) \rightarrow (x = classe(e))$ est trop générale car elle couvre au moins un contre-exemple.

Propriété 7.1 *$s(o,e) \neq 0$ si et seulement si la règle $R(o,e) =_d C(o,e) \rightarrow (x = classe(e))$ est cohérente pour EA.*

Preuve : la preuve découle des définitions. #

L'ensemble $M - cons(EA, \leq_o)$ des M-voisins de o s'exprime aisément à l'aide de la mesure de spécificité.

Propriété 7.2 *Soit o un objet à classer. L'exemple e est un M-voisin de o, c'est-à-dire $R(o,e)$ n'admet pas de contre-exemples à M attributs près, si et seulement si $s(o,e) \geq M$.*

Preuve : la preuve découle des définitions. #

L'intérêt de la définition de la mesure de spécificité ne réside pas dans les deux propriétés précédentes mais dans l'éclairage qu'elle apporte sur le rôle de M. Étant donnés un objet à classer o et un exemple e, $s(o,e)$ mesure la distance à laquelle se trouve le plus proche contre-exemple de e. Elle mesure le nombre minimum d'attributs de la partie gauche de la règle $R(o,e) = C(o,e) \rightarrow (x = classe(e))$ qu'il faudrait généraliser pour couvrir un des contre-exemples les plus proches. Elle indique donc le nombre d'attributs de la partie gauche de $R(o,e)$ qui peuvent être généralisés sans risque de couvrir de contre-exemples. Dans le cas qui nous intéresse, la règle est fixée ainsi que les contre-exemples, donc $s(o,e)$ mesure simplement la distance à laquelle se trouve le plus proche contre-exemple. Elle mesure l'isolement de la règle $R(o,e)$. Ainsi, plus M est grand, plus les règles que l'on retient sont isolées de leurs contre-exemples. Puisque les contre-exemples sont fixés, plus M est grand, plus les hypothèses retenues sont spécifiques. M peut donc être considéré comme une seuil de spécificité (ou inversement de généralité).

7.2.3 Vers un réglage automatique

Les deux paramètres ϵ et M peuvent être considérés en même temps.

Définition 7.5 *e est un (ϵ, M)-voisin de o, noté $e \in (\epsilon, M) - cons(EA, \leq_o)$, si et seulement si $e \in \epsilon - cons(EA, \leq_o^M)$.*

L'approche « diviser pour régner » peut encore être utilisée :

Théorème 7.3 $(\epsilon, M) - cons(E_1 \cup E_2, \leq_o) \subseteq (\epsilon, M) - cons(\epsilon - cons(E_1, \leq_o) \cup (\epsilon, M) - cons(E_2, \leq_o), \leq_o)$

Preuve : la preuve du théorème 7.1 ne dépend pas de la relation de proximité utilisée. Elle est donc valable pour $\leq_o^M$. #

On a clairement $cons(EA, \leq_o) = (0,1) - cons(EA, \leq_o)$. Les paramètres ϵ et M jouent des rôles opposés : plus M est grand, plus la règle $R(o,e)$ associée à un exemple e admet de contre-exemples et plus ϵ est grand, plus $R(o,e)$ peut tolérer de contre-exemples sans se que e soit éliminé de l'ensemble des voisins de l'objet o à classer. Ces deux paramètres sont complémentaires. À M fixé, on peut faire croître ϵ de 0 à 1, jusqu'à accepter tous les exemples comme voisins, $cons(EA, \leq_o) = EA$. À ϵ fixé, on peut faire varier M de 1 à d, le nombre d'attributs, et pour un ϵ suffisamment élevé, faire varier $cons(EA, \leq_o)$ de l'ensemble d'apprentissage au complet EA à l'ensemble vide. Donc la classe suggérée varie de la classe majoritaire quand $cons(EA, \leq_o) = EA$ à une classe au hasard quand $cons(EA, \leq_o) = \emptyset$.

Nous considérons une procédure de réglage automatique des paramètres ϵ et M semblable à [Kohavi et John, 1995]. Pour déterminer le meilleur réglage à partir d'un ensemble d'apprentissage EA, il est possible de le diviser en un ensemble de test EA_1 et un ensemble d'apprentissage EA_2, faire varier ϵ et M et à chaque fois évaluer les performances du classifieur s'appuyant sur EA_2 pour retrouver la classe des éléments de EA_1. Nous avons adopté une amélioration

usuelle de cette technique : la validation croisée. Nous avons divisé EA en dix sous-ensembles de même taille (plus ou moins un élément). Nous classons, pour chaque valeur des paramètres, chaque dixième en fonction des neuf autres sous-ensembles. La moyenne $v(\epsilon, M)$ du nombre de classifications correctes estime la qualité des valeurs données aux paramètres. Pour éviter des variations aléatoires dans le calcul de $v(\epsilon, M)$, nous n'utilisons pas de tirage au sort dans le critère de résolution. En cas d'égalité entre plusieurs classes, ou si l'ensemble des voisins est vide, nous considérons que l'objet n'est pas classé correctement. Le temps de calcul nécessaire à l'évaluation de $v(\epsilon, M)$ n'étant pas négligeable (cf. chapitre 9), il faut limiter autant que possible le nombre de couples (ϵ, M) testés. L'idéal serait que la courbe décrite par $v(\epsilon, M)$ soit convexe, ne serait-ce que suivant M à ϵ fixé, ou suivant ϵ à M fixé. Malheureusement la figure 7.4 obtenue en faisant varier M entre 1 et d, ϵ entre 0 et 0.5 et en évaluant $v(\epsilon, M)$ sur un dixième de l'ensemble de données réelles lung-cancer (présenté dans le chapitre suivant) montre que ce n'est pas le cas en général.

L'axe horizontal de gauche à droite représente M, l'axe « venant vers nous » représente ϵ. Nous pouvons observer les trois caractéristiques suivantes sur la courbe à ϵ fixé :

- si ϵ est suffisamment élevé, la courbe commence à gauche (M faible) par un palier qui est la valeur du classifieur donnant par défaut la classe majoritaire. Nous sommes dans le cas où $cons(EA, \leq_o) = EA$;
- la courbe finit à droite, quand M augmente, par un palier à la valeur nulle, c'est le cas où $cons(EA, \leq_o)$ est vide ;
- entre les deux, la courbe passe par un ou plusieurs maxima locaux.

L'algorithme `chercher_max_M_`ϵ que nous proposons discrétise les valeurs de ϵ et cherche le maximum de $v(\epsilon, M)$ pour ϵ variant de 0 à 0.5 du résultat de la recherche du maximum de $v(\epsilon, M)$ à ϵ fixé. Cette recherche s'effectue en faisant varier M avec un pas de 1 entre M_{min} et M_{max} où M_{min} est la valeur à partir de laquelle le cardinal de $cons(EA, \leq_o)$ est plus petit que celui de EA, et M_{max} celle à partir de laquelle $cons(EA, \leq_o)$ est vide. Les valeurs M_{min} et M_{max} sont obtenues par recherche dichotomique.

fonction chercher_max_M_ϵ
```
Initialiser v_max à 0
Pour ε variant de 0 à 0.5 faire
  chercher_max_M(ε)
  Si v_res > v_max
  Alors v_max := v_res, M_max := M_res et ε_max := ε
Rendre(ε_max et M_max)
```

fonction chercher_max_M(ϵ**)**
```
Initialiser v_res à 0
Calculer M_min et M_max par dichotomie
Pour M variant de M_min à M_max faire
  évaluer v := v(ε, M)
  Si v > v_res
  Alors v_res := v et M_res := M
Rendre(v_res et M_res)
```

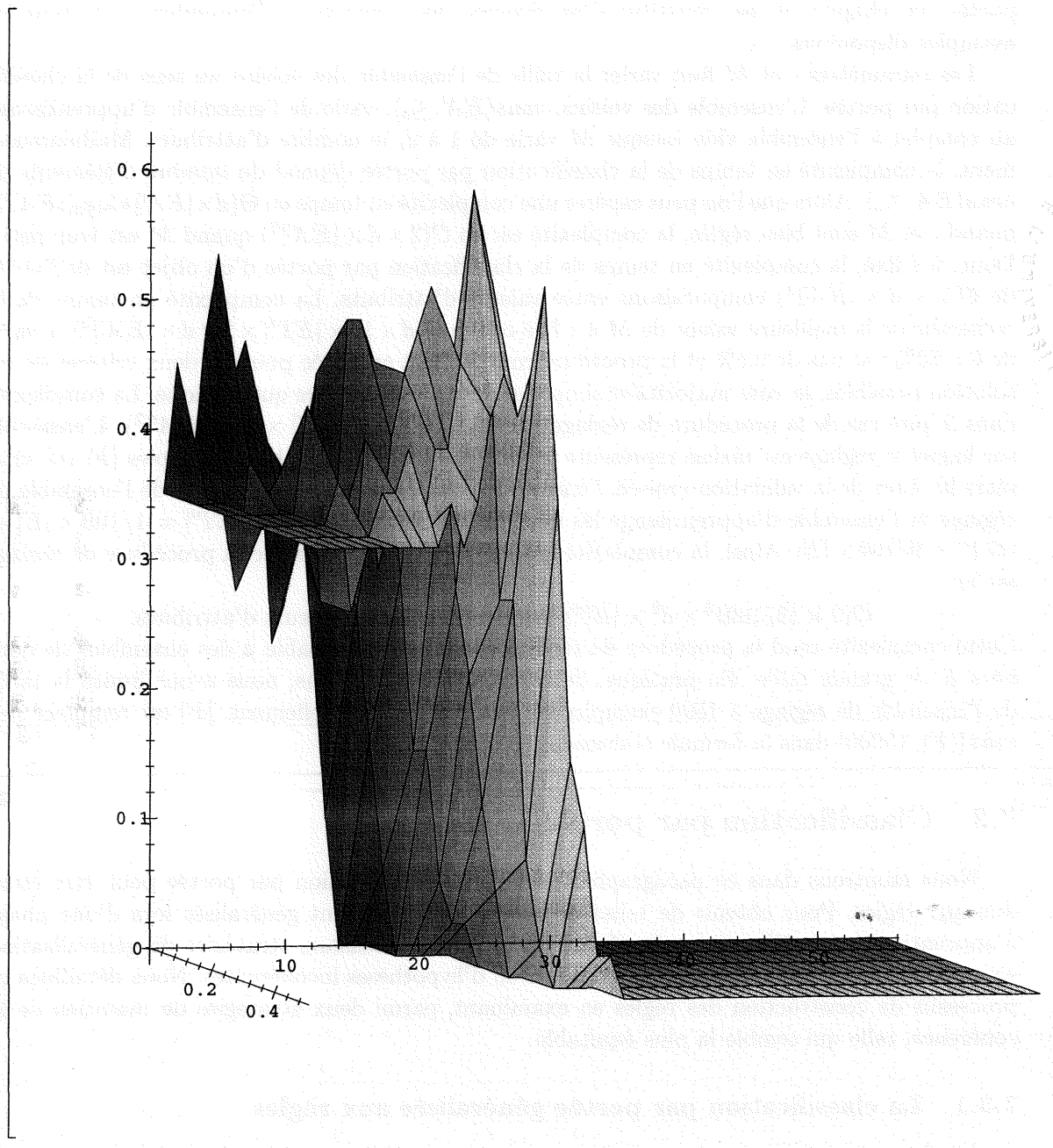


FIG. 7.4 – *Évaluation de ϵ et M sur un problème de classification réel.*

Soient EA' et ET' les ensembles d'apprentissage et de test utilisés pour l'évaluation de $v(\epsilon, M)$ à l'aide d'une validation croisée. EA' et ET' contiennent respectivement neuf dixièmes et un dixième de l'ensemble E' d'exemples utilisés pour le réglage de ϵ et M. L'ensemble de réglage E' que nous avons retenu lors de l'évaluation expérimentale de la classification par portée (cf. chapitre 9) est constitué d'un dixième des éléments de l'ensemble E de tous les exemples disponibles.

Les paramètres ϵ et M font varier la taille de l'ensemble des voisins au sens de la classification par portée. L'ensemble des voisins, $cons(EA', \leq_o)$, varie de l'ensemble d'apprentissage au complet à l'ensemble vide lorsque M varie de 1 à d, le nombre d'attributs. Malheureusement, la complexité en temps de la classification par portée dépend du nombre d'éléments de $cons(EA, \leq_o)$. Alors que l'on peut espérer une complexité en temps en $\Theta(d \times |EA'| * log_2(|EA'|))$ quand ϵ et M sont bien réglés, la complexité est en $\mathcal{O}(2 \times d \times |EA'|^2)$ quand M est trop petit. Donc, à ϵ fixé, la complexité en temps de la classification par portée d'un objet est de l'ordre de $\mathcal{O}(2 \times d \times |EA'|^2)$ comparaisons entre valeurs d'attributs. La complexité en temps de la recherche de la meilleure valeur de M à ϵ fixé est en $\mathcal{O}(d \times 10 \times |ET'| \times 2 \times d \times |EA'|^2)$. ϵ varie de 0 à 50% par pas de 0.5% et la procédure complète est exécutée pour les deux critères de résolution possibles, le vote majoritaire simple et le vote majoritaire quadratique. La complexité dans le pire cas de la procédure de réglage est donc $\mathcal{O}(200 \times |ET'| \times d^2 \times |EA'|^2)$. L'ensemble sur lequel le réglage est réalisé représente un dixième du nombre total d'exemples $|E|$ (cf. chapitre 9). Lors de la validation croisée, l'ensemble de test comporte un dixième de l'ensemble de réglage et l'ensemble d'apprentissage les neuf dixièmes restants, donc $|ET'| = 1/100 \times |E|$ et $|EA'| = 9/100 \times |E|$. Ainsi, la complexité en temps dans le pire cas de la procédure de réglage est en :

$$\mathcal{O}(2 \times (9/100)^2 \times d^2 \times |E|^3) \text{ comparaisons entre valeurs d'attributs.}$$

Cette complexité rend la procédure de réglage proposée inapplicable à des ensembles de données E de grande taille. En pratique, dans nos expérimentations, nous avons limité la taille de l'ensemble de réglage à 1500 exemples tirés au hasard ; formellement, $|E|$ est remplacé par $max(|E|, 15000)$ dans la formule ci-dessus.

7.3 Classification par portée et règles

Nous montrons dans ce paragraphe comment la classification par portée peut être étendue aux règles. Pour obtenir de telles règles, les exemples sont généralisés lors d'une phase d'apprentissage en utilisant la classification par portée. Plusieurs stratégies de généralisation sont possibles autorisant ou non la construction d'hypothèses incohérentes. Nous détaillons ce processus de construction des règles en examinant, parmi deux stratégies de maintien de la cohérence, celle qui semble la plus équitable.

7.3.1 La classification par portée généralisée aux règles

L'information apportée par une règle cohérente et pertinente $G \rightarrow (x = v_x)$ est que tous les objets satisfaisant G appartiennent à la classe v_x. Une règle peut donc être vue comme une notation simplifiée d'un ensemble d'exemples. D'un point de vue syntaxique, une règle peut ainsi être décrite comme un exemple dont tous les attributs ne sont pas valués $(G, (x = v_x))$. Il importe toutefois de distinguer les descriptions des règles de celles des objets incomplètement décrits : la sémantique d'une règle est *conjonctive*. Alors qu'un exemple e incomplètement décrit signifie l'existence d'un exemple complètement décrit dans l'hyper-rectangle défini par la description partielle de e, une règle signifie que tous les objets de cet hyper-rectangle sont

des exemples. Formellement, à tout ensemble EA_r formé par des descriptions de règles et d'exemples, on peut associer un ensemble EA_c contenant explicitement les exemples couverts par les règles de EA_r.

Définition 7.6 *Soit EA_r un ensemble de règles et d'exemples. EA_c est l'union de l'ensemble des exemples appartenant à EA_r et des objets de l'UNIVERS couverts par une règle R de EA_r, pour chaque règle R appartenant à EA_r.*

Considérons par exemple, l'ensemble d'apprentissage $EA_r =_d \{(a = Vrai) \rightarrow (x = Vrai), ((a = Faux) \wedge (b = Faux), (x = Faux))\}$, dans un *UNIVERS* décrit par deux attributs booléens a et b. L'ensemble EA_c est l'ensemble des exemples complètement décrits $\{((a = Vrai) \wedge (b = Vrai), (x = Vrai)), ((a = Vrai) \wedge (b = Faux), (x = Vrai)), ((a = Faux) \wedge (b = Faux), (x = Faux))\}$. Dans EA_c, chaque règle de EA_r est simplement remplacée par l'ensemble des exemples qu'elle couvre. Notons que cet ensemble est infini dans le cas des attributs continus.

Illustrons sur un exemple la différence entre les règles et les exemples incomplètement décrits. Étant donnés un objet i incomplètement décrit $i =_d ((a = V) \wedge (b = V) \wedge (c = F))$, désigne un seul des deux exemples complètement décrits $o =_d (a = V) \wedge (b = V) \wedge (c = F) \wedge (d = F)$ et $o' =_d (a = V) \wedge (b = V) \wedge (c = F) \wedge (d = V)$, et une règle $r =_d (b = V) \wedge (c = V) \rightarrow (x = V)$, représente l'ensemble des exemples de la zone grisée dans le tableau ci-dessous.

	ab	$a\bar{b}$	$\bar{a}\bar{b}$	$\bar{a}b$
cd	r_1			r_2
$c\bar{d}$	r_3			r_4
$\bar{c}\bar{d}$	o			
$\bar{c}d$	o'			

Du point de vue de la classification par portée, lorsque l'on classe un objet o, un exemple e incomplètement décrit est équivalent à un de ses modèles m_e les plus éloignés de o. Il faut en effet que la règle $R(o, e)$ la plus spécifique couvrant o et e couvre tous les modèles de e (cf. définition 6.4). Du point de vue de la classification par portée d'un objet o, une règle r est au contraire équivalente à un des objets complètement décrits qu'elle couvre parmi ceux qui sont les plus proches de o (cf. propriété 7.3). La règle la plus spécifique couvrant o et r qui ne couvre que l'ensemble des modèles de r les plus proches de o est définie de la façon suivante :

Définition 7.7 *Soient o_1 un objet, éventuellement partiellement décrit, et o_2 une règle, $R(o_1, o_2) =_d C(o_1, o_2) \rightarrow (x = classe(o_2))$ est la règle la plus spécifique couvrant (tous les modèles de) o_1 et le modèle de o_2 le plus proche de o_1 si et seulement si $C(o_1, o_2)$ est la conjonction de conditions (ou la description partielle), étendues aux intervalles dans le cas des attributs linéaires, où la condition $C_a(o_1, o_2)$ pour chaque attribut a est :*

- *Vrai (ou manquante) si a n'est pas valué dans o_1;*
- *$(a = v_a^1)$ où v_a^1 est la valeur de o_1 si a est nominal et valué dans o_1 et de même valeur* **ou manquant** *dans o_2, Vrai (ou manquante) sinon;*
- *$(a \in [min(v_a^1, v_a^2), max(v_a^1, v_a^2)])$ si a est linéaire et prend la valeur v_a^1 dans o_1 et $v_a^2 = v_{min}$, si $v_a^1 < v_{min}$, $v_a^2 = v_{max}$, si $v_a^1 > v_{max}$, et $v_a^2 = v_a^1$ sinon, où la description de o_2 pour l'attribut a est l'intervalle $[v_{min}, v_{max}]$.*

Considérons à nouveau la règle $r =_d (b = V) \wedge (c = V) \rightarrow (x = V)$ et l'objet $o =_d ((a = V) \wedge (b = V) \wedge (c = F) \wedge (d = F))$. Suivant la définition précédente, $C(i, r) =_d (a = V) \wedge (b = V) \wedge (d = F)$. Nous pouvons vérifier que r se comporte du point de vue de la classification par portée comme son modèle $r_3 =_d ((a = V) \wedge (b = V) \wedge (c = V) \wedge (d = F), (x = V))$, en effet $C(o, r_3) = C(o, r)$.

Dans le cas des règles, il suffit qu'un modèle de o_2 appartienne à $C(o, o_1)$ pour être plus proche de o que o_1. Il suffit donc de comparer le modèle de o_2 le plus proche de o.

Définition 7.8 *Soit un objet, éventuellement incomplètement décrit, o, et o_1 et o_2 des objets, éventuellement incomplètement décrits, ou des règles, nous définissons l'ordre partiel $\leq_o$ sur l'UNIVERS par:*
$o_1 \leq_o o_2$ si et seulement si $C(o, o_1) \models C(o, o_2)$.

Cette définition généralise la définition 6.3. La définition de $cons(EA, \leq_o)$ n'a pas à être modifiée (cf. définition 6.5) et l'algorithme de classification par portée à partir d'exemples seulement s'applique encore.

Nous pouvons vérifier que, du point de vue de la classification par portée ainsi définie, lorsque l'on classe un objet o, une règle r est équivalente à un de ses modèles m_r les plus proches de o au sens où $R(o, r)$ est égal à $R(o, m_r)$ et il n'existe pas de modèle m'_r de r tel que $R(o, m'_r) = R(o, r)$, $m'_r \models C(o, m_r)$ et $m_r \not\models C(o, m'_r)$.

Propriété 7.3 *Soit un objet à classer o, éventuellement partiellement décrit. Du point de vue de la classification par portée de o, une règle $r = G \rightarrow (x = v_x)$ est équivalente à l'exemple complètement décrit $(m_r, (x = v_x))$, où m_r est un des modèles de G les plus proches de o défini pour chaque attribut a par:*

- *v_a^e si a prend la valeur v_a^e dans e, sinon,*
- *une valeur quelconque si a n'est pas valué dans o, sinon,*
- *une valeur différente de celle de o, si a est nominal,*
- *si a est linéaire, v_a^o est la valeur de a dans o et G est décrit par l'intervalle $[v_{min}, v_{max}]$:*
  - *v_{min} si $v_a^o < v_{min}$,*
  - *v_a^o si $v_a^o \in [v_{min}, v_{max}]$,*
  - *v_{max} si $v_a^o > v_{max}$.*

Preuve : il est aisé de vérifier que $C(o, m_r)$ est identique à $C(o, r)$. Donc la règle $R(o, m_r)$ est identique à la règle $R(o, r)$. Et, pour tout objet ou règle o_1, $o_1 \leq_o r$ est équivalent à $C(o, o_1) \models C(o, r)$, donc à $C(o, o_1) \models C(o, m_r)$, donc à $o_1 \leq_o m_r$.

De même, pour tout objet ou règle o_1, $r \leq_o o_1$ est équivalent à $m_r \leq_o o_1$. #

Le théorème suivant montre l'équivalence existant entre la classification à partir des règles (EA_r) ou à partir de tous les exemples qu'elles couvrent (EA_c).

Théorème 7.4 *Il existe e_r appartenant à $M - cons(EA_r, \leq_o)$ tel que $classe(e_r) = v_x$ si et seulement si il existe e_c appartenant à $M - cons(EA_c, \leq_o)$ tel que $classe(e_c) = v_x$.*

Preuve :

$\Rightarrow$ Soit $e_r \in cons(EA_r, \leq_o^M)$, montrons qu'il existe $e_c \in cons(EA_c, \leq_o^M)$.

- Si e_r est une règle r dans EA_r, soit e_c un des modèles de r les plus proches de o, $e_c \in EA_c$ et $e_c \leq_o^M e_r$
  Supposons qu'il existe $ce \in CE(e_c, EA_c)$ tel que $ce \leq_o^M e_c$
  - si ce correspond à un objet dans EA_r alors $ce \leq_o^M e_c$ contredit $e_r \in cons(EA_r, \leq_o^M)$.
  - si ce est un modèle d'une règle r' dans EA_r alors $r' \leq_o^M ce$ contredit $e_r \in cons(EA_r, \leq_o^M)$.
- Si e_r est un objet de EA_r alors $e_c = e_r \in EA_c$
  Supposons qu'il existe $ce \in CE(e_c, EA_c)$ tel que $ce \leq_o^M e_c$
  De même que dans le cas précédent, cela contredit $e_r \in cons(EA_r, \leq_o^M)$.

$\Leftarrow$ Soit $e_c \in cons(EA_c, \leq_o^M)$, montrons qu'il existe $e_r \in cons(EA_r, \leq_o^M)$.

- Si e_c est un modèle d'une règle r dans EA_r alors $e_r = r \in EA_r$ et $r \leq_o^M e_c$
  Supposons qu'il existe $ce \in CE(e_r, EA_r)$ tel que $ce \leq_o^M r$
  - si ce est un objet de EA_r alors $ce \in EA_c$ et $ce \leq_o^M e_r$ contredit $e_c \in cons(EA_c, \leq_o^M)$.
  - si ce est une règle r' de EA_r, soit ce' le modèle de r' le plus proche de o, $ce' \in CE(e, EA_c)$ et $ce' \leq_o^M ce$ contredit $e_c \in cons(EA_c, \leq_o^M)$.
- Si e_c est un objet de EA_r alors $e_r = e_c \in EA_c$
  Supposons qu'il existe $ce \in CE(e_r, EA_r)$ tel que $ce \leq_o^M e_r$
  De même que dans le cas précédent, cela contredit $e_c \in cons(EA_c, \leq_o^M)$.

#

Il faut noter que l'équivalence précédente n'est pas vérifiée lorsque l'on considère le paramètre de cohérence ϵ. Considérons l'*UNIVERS* décrit par quatre attributs booléens, l'objet $o =_d (a = V) \wedge (b =?) \wedge (c =?) \wedge (d = V)$ et l'ensemble d'apprentissage EA_r :

	a	b	c	d	x
$r_1 =_d$	V	?	?	V	V
$r_2 =_d$	F	?	V	V	F

$C(o, r_1) =_d (a = V) \wedge (d = V)$ et $C(o, r_2) =_d (a = V)$, donc $r_1 \leq_o r_2$. Néanmoins r_2 appartient à $0.5 - cons(EA_r, \leq_o)$, puisque $R(o, r_2)$ admet un nombre de contre-exemples (1) inférieur ou égal à 0,5 fois le nombre d'exemples ($|EA_r| =_d 2$).

L'ensemble d'apprentissage EA_r est équivalent à EA_c :

	a	b	c	d	x
$e_1^1 =_d$	V	V	V	V	V
$e_1^2 =_d$	V	V	F	V	V
$e_1^3 =_d$	V	F	F	V	V
$e_1^4 =_d$	V	F	V	V	V
$e_2^1 =_d$	F	V	V	V	F
$e_2^2 =_d$	F	F	V	V	F

Quels que soient j et k, $C(o, e_1^j) =_d (a = V) \wedge (d = V)$ et $C(o, e_2^k) =_d (a = V)$, donc $e_1^j \leq_o e_2^k$. Donc e_2^1 et e_2^2 admettent quatre contre-exemples alors que l'ensemble d'apprentissage est de taille $|EA_c| =_d 6$. Les exemples e_2^1 et e_2^2 n'appartiennent pas à $0.5 - cons(EA_c, \leq_o)$.

La correspondance entre la classification par portée à partir de règles et la classification par portée à partir de l'ensemble des exemples couverts par les règles est seulement logique. Les nombres de règles et d'exemples diffèrent et influent donc sur le résultat du critère de résolution. L'intérêt de l'utilisation de la prise en compte explicite des règles est bien sûr de procéder aux classifications sans expansion des règles. Cela est très utile en présence d'attributs continus car, comme nous l'avons noté précédemment, le nombre de modèles est infini dans ce cas ; il n'est donc pas possible d'effectuer une telle expansion des règles.

7.3.2 Généralisation des exemples en règles

Pour pouvoir appliquer la classification par portée à partir de règles, il faut évidemment disposer de règles. Des règles peuvent être fournies par l'utilisateur. Elles peuvent également être construites à partir d'exemples. Pour cela, une approche consiste à généraliser chaque exemple de EA en un ensemble de règles couvrant ses « voisins » pour la classification par portée. Formellement, nous déterminons pour chaque exemple e l'ensemble de ses voisins $cons(EA, \leq_e)$, et pour chaque voisin $ve \in cons(EA, \leq_e)$ de même classe que e, nous construisons la règle cohérente et pertinente pour EA, $R(e, ve) = C(e, ve) \rightarrow (x = classe(e))$. L'ensemble des règles ainsi obtenu remplace les exemples couverts.

Considérons l'exemple des patients :

Patient	a	b	c	d	x
e_1	V	V	V	V	V
e_2	F	F	F	F	F
e_3	V	F	F	F	V

La seule règle possible entre deux exemples appartenant à la même classe, voisins pour la classification par portée est $R(e_1, e_3) =_d (a = Vrai) \rightarrow (x = Vrai)$. L'ensemble d'apprentissage obtenu après généralisation est : $\{R(e_1, e_3), e_3\}$.

Cette technique de généralisation d'un exemple pour couvrir ses voisins est classiquement utilisée dans les approches à base d'instances, par exemple dans EACH [Salzberg, 1991] ou dans RISE [Domingos, 1996]. Notre particularité est d'utiliser l'ensemble des voisins pour la classification par portée plutôt que pour une distance, ce qui n'est pas sans répercussion. En particulier, dans notre approche, il n'est pas possible d'obtenir des ensembles de règles couvrant plus d'exemples que les ensembles de règles engendrés comme indiqué plus haut. En effet, tout exemple e couvert par une règle cohérente et pertinente pour EA fait partie de $cons(EA, \leq_e)$. Cela ne signifie pas pour autant que des règles plus générales ne peuvent pas être obtenues par d'autres approches, par exemple en généralisant autant que possible les règles que nous construisons à partir de chaque élément de $cons(EA, \leq_e)$. Cela indique seulement que l'ensemble de ces règles ne couvre pas davantage d'exemples de EA que l'ensemble de règles que nous construisons. Par ailleurs, le fait que nous ne construisons que les règles les plus spécifiques couvrant e et ses voisins n'implique pas que notre approche soit incapable de déterminer les règles minimales cohérentes et pertinentes pour EA décrivant le concept. En effet, pour construire une règle, il suffit de deux exemples « diagonalement opposés » dans l'hyper-rectangle constituant la partie gauche de la règle (les attributs nominaux ont des valeurs différentes, sauf ceux définissant l'hyper-rectangle et les attributs linéaires prennent les

valeurs délimitant l'intervalle associé à la règle minimale). Donc même les règles minimales sont apprenables sous la forme d'un hyper-rectangle entre deux exemples.

Pour garder l'ensemble des règles engendrées aussi compact que possible, nous ne retenons que les règles minimales parmi celles construites. A chaque fois qu'une règle est créée à partir de e, nous ne la conservons que si elle n'est pas impliquée par une règle existante et nous supprimons les règles qu'elle implique.

```
fonction généraliser_exemples_en_règles(E)
    Initialiser ER à l'ensemble vide
    Pour tout exemple e de E faire
        Calculer cons(E\{e}, ≤_e)
        Pour tout voisin v de e faire
            S'il n'existe pas de règle R' de ER telle que R' ⊨ R(e,v) alors
                Pour toute R' ∈ ER telle que R(e,v) ⊨ R' faire
                    Supprimer R' de ER
                Ajouter R à ER
    Rendre(ER)
```

Nous appelons $gen(EA)$ l'ensemble des règles minimales construites par la fonction `généraliser_exemples_en_règles`(EA). Cet ensemble est représenté sur la figure 7.5 pour un $UNIVERS$ représenté par deux attributs linéaires continus contenant trois exemples positifs du concept à apprendre ($e1$, $e2$, $e3$) et trois exemples négatifs ($f1$, $f2$, $f3$).

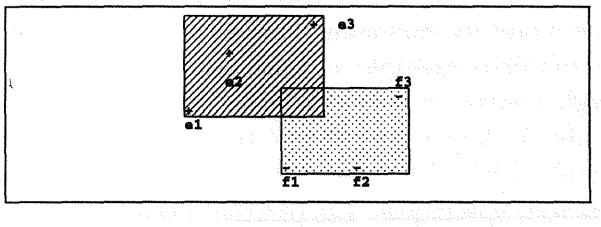


FIG. 7.5 – $gen(EA)$.

Une fois toutes les règles de $gen(EA)$ générées, nous retirons de l'ensemble d'apprentissage tous les exemples qu'elles couvrent. Nous obtenons donc un nouvel ensemble d'apprentissage EA_r composé de règles et d'exemples ponctuels non couverts par les règles. Afin de conserver la « sémantique » du vote majoritaire simple où chaque exemple dispose d'une voix, le nouvel ensemble d'apprentissage est pondéré de la façon suivante : un poids de 1 est associé à chaque exemple ponctuel non couvert et le poids de chaque exemple couvert par m règles est réparti uniformément : on ajoute $1/m$ au poids de chacune des m règles. Le critère de résolution ainsi défini est donc un vote majoritaire pondéré pour tenir compte du nombre d'exemples initial.

7.3.3 Stratégies de généralisation

Éviter les incohérences ou les prendre en compte

L'ensemble EA_r ainsi défini n'est malheureusement pas un ensemble d'apprentissage exploitable par $\leq_o$, car l'ensemble $gen(EA)$ sur lequel il est construit contient souvent trop de règles contradictoires (couvrant au moins un même objet et concluant sur des classes distinctes). En effet, la construction des règles est réalisée à partir des exemples initiaux seulement,

c'est-à-dire sans prendre en compte au fur et à mesure du processus de généralisation les hypothèses (exemples) introduites par les généralisations obtenues à partir des autres exemples. Or, plusieurs règles admettant des parties droites distinctes peuvent couvrir le même objet. La génération des règles s'accompagnant d'un ajout d'hypothèses obtenues par généralisation de l'ensemble d'apprentissage initial, il faut éviter autant que possible d'introduire des hypothèses incohérentes, c'est-à-dire éviter d'avoir des objets couverts par deux règles contradictoires si l'on veut pouvoir appliquer le modèle initial de la classification utilisant $\leq_o$.

Dans cet objectif, une solution consiste à contrôler l'ensemble des règles générées afin qu'elles ne soient pas contradictoires. Le contrôle peut s'effectuer au cours de la généralisation des exemples. Nous montrons dans le paragraphe suivant que cela conduit à privilégier les premières hypothèses. La cohérence peut également être contrôlée *a posteriori* en ne conservant que les règles qui n'ont « aucune intersection » (c'est-à-dire ne couvrent pas un même objet) avec une règle concluant sur une autre classe. Nous présentons une stratégie de ce type. Nous montrons qu'elle représente une stratégie de moindre engagement.

Une autre approche, diamétralement opposée, consiste à conserver toutes les règles de $gen(EA)$ et à utiliser $<_o$ (cf. paragraphe7.1.2) pour gérer les exemples incohérents.

Privilégier les premières hypothèses pour maintenir la cohérence

La solution la plus immédiate pour maintenir la cohérence du classifieur par rapport à l'$UNIVERS$ consiste à introduire les règles, c'est-à-dire tous les exemples qu'elles couvrent, dans l'ensemble d'apprentissage dès leur construction, pour qu'elles soient prises en compte lors de la généralisation des exemples suivants. Cette première solution présente l'intérêt de conduire à des règles couvrant un maximum d'objets de l'$UNIVERS$ dans le sens où les règles minimales, engendrées par deux exemples et cohérentes avec tous les exemples connus (initiaux ou couvert par une règle construite jusque là) sont retenues. Cette solution a cependant l'inconvénient d'être sensible à l'ordre dans lequel les exemples sont généralisés ; en particulier, plus le nombre d'exemples généralisés augmente, plus les hypothèses faites sont nombreuses, plus les règles générées sont spécifiques. Les premiers exemples sont clairement privilégiés. Ce phénomène est illustré à la figure 7.6 dans le cas où on généralise $e1$ (ou $e3$) avant de généraliser $f1$ (ou $f3$).

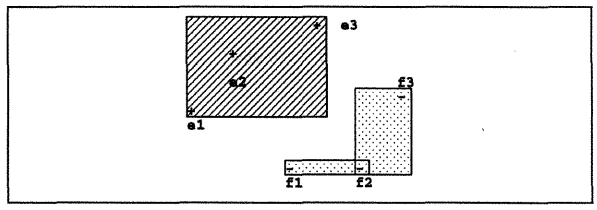


FIG. 7.6 – *Privilégier les premières hypothèses.*

Cette stratégie où on absorbe (couvre) à chaque étape un maximum d'individus est appelée stratégie gloutonne, en anglais *greedy*.

Une stratégie de moindre engagement

Une autre solution consiste à ne retenir que les règles minimales ne couvrant aucun objet couvert par une règle concluant sur une classe différente. Lorsque l'on construit une règle R couvrant e et un de ses voisins de même classe, il faut vérifier que R ne couvre pas d'objet

couvert par une des règles qui peuvent être construites à partir des exemples de classes différentes. En fait, il n'est pas nécessaire de prendre en compte toutes les règles concluant sur des classes différentes : il suffit de considérer les règles minimales parmi celles-ci, c'est-à-dire $gen(EA)$. L'ensemble $gen\_cons(EA)$ des règles minimales ne couvrant aucun objet couvert par un élément de $gen(EA)$ est plus long à calculer que celui associé à la stratégie précédente (privilégier les premières hypothèses). En outre, il offre une couverture plus faible de l'espace dans le cas général[2]. L'intérêt de cette seconde solution est qu'elle n'est pas sensible à l'ordre de généralisation des exemples et qu'elle ne privilégie donc aucune hypothèse. Tous les exemples sont traités de la même façon. Cette seconde solution, que l'on pourrait qualifier de *moindre engagement* puisqu'elle ne fait pas plus d'hypothèses qu'il n'est possible sans léser les suivants, est illustrée à la figure 7.7.

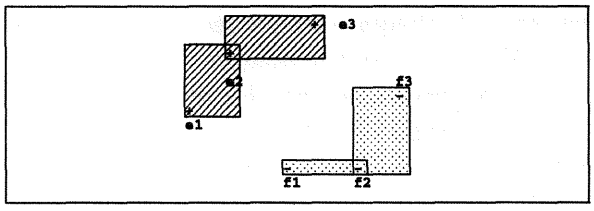


FIG. 7.7 – *Moindre engagement.*

Dans nos expérimentations, nous avons retenu la seconde stratégie (moindre engagement) de maintien de la cohérence des règles engendrées. Au prix d'un temps de calcul en phase d'apprentissage plus long, nous avons privilégié une généralisation équitable des exemples. Nous pensons, en effet, que la variance de la précision des classifieurs associés à la première solution (qui dépend de l'ordre dans lequel les exemples sont généralisés) est grande. Puisque davantage d'hypothèses sont faites dans la première approche, davantage d'hypothèses fausses peuvent être faites. La qualité du classifieur obtenu dépend de l'ordre dans lequel les exemples sont considérés et il semble bien difficile de pouvoir prévoir un ordonnancement qui minimise l'erreur.

Lorsqu'un objet à classer o est couvert par une règle R de EA_r, l'ensemble $cons(EA_r, \leq_o)$ ne contient que R. Pour des raisons évidentes d'efficacité, il est bien entendu préférable de tester si o est couvert par une règle de EA_r et de calculer $cons(EA_r, \leq_o)$ seulement si tel n'est pas le cas. Toutefois, ce traitement particulier des règles de EA_r ne change rien au résultat fourni par l'approche. Pour faire un parallèle avec la classification par portée à partir d'exemples seuls, tout se passe comme si l'on testait si o appartient à EA_c avant de calculer $cons(EA_c, \leq_o)$.

7.4 Deux extensions supplémentaires

Nous présentons ici deux extensions supplémentaires de l'algorithme de classification par portée. La première concerne l'élimination des exemples redondants dans l'ensemble d'apprentissage afin d'accélérer le processus de classification. La seconde concerne la prise en compte d'exemples négatifs, fournis, par exemple, lors d'une utilisation incrémentale de la classification par portée en présence d'un oracle.

2. Bien entendu, les points de l'*UNIVERS* non couverts par les règles peuvent toujours être classés par portée à partir des règles construites.

7.4.1 Prise en compte de la redondance

Les bases de données réelles contiennent souvent des exemples redondants, c'est-à-dire plusieurs fois le même exemple (*description*, *classe*). Il est évident que plusieurs exemplaires d'un exemple n'apporte pas d'information supplémentaire lors de la phase d'induction reposant sur le biais logique seulement. Nous pourrions ne garder qu'un exemplaire de chaque exemple redondant. Mais cela éliminerait une information pertinente pour les biais statistiques : la fréquence relative de chaque exemple. En effet, si le voisinage d'un objet o contient $\{e_1, e_2, e_3, e_4, e_5, e_6\}$ où e_1, e_2, e_3, e_4 sont quatre exemples identiques de classe 1 et e_5, e_6 deux exemples distincts de classe 2, un vote majoritaire conduit à choisir la classe 1. Si l'on ne garde qu'un exemplaire de e_1, e_2, e_3, e_4, c'est la classe 2 qui est choisie. Des problèmes semblables peuvent modifier le comportement du biais statistique introduit par ϵ. La solution que nous avons retenue est de ne conserver qu'un exemplaire de chaque exemple redondant, mais de lui accorder, lors du vote ou de l'utilisation de ϵ, autant de voix que le nombre d'exemples de l'ensemble d'apprentissage initial qu'il représente. Cela ne modifie en rien les résultats de $(\epsilon, M) - cons(EA, \leq_o)$ et des critères de résolution (vote majoritaire simple ou quadratique) que nous utilisons (donc la précision du classifieur obtenu), mais cela permet d'accélérer considérablement la classification de certaines bases de données, comme les bases de parole que nous considérons dans le chapitre suivant.

7.4.2 Prise en compte d'exemples négatifs

Considérons le cas d'exemples négatifs. Ce sont des exemples dont la description est connue, mais pour lesquels la seule information disponible sur la classe est que l'exemple n'appartient pas à la classe v_x. La prise en compte de ce type d'exemple peut résulter d'une utilisation incrémentale du classifieur par portée en présence d'un oracle. Le terme classification incrémentale désigne les problèmes de classification dont l'ensemble d'apprentissage croît au cours du temps car des exemples y sont ajoutés. Une source possible d'exemples peut provenir d'un oracle (ou professeur). Considérons le cas où le classifieur classe les objets o que l'on lui présente et, périodiquement, un professeur valide les classes proposées par le classifieur. Dans les cas où la classe proposée v_x est incorrecte, l'indication du professeur peut être de deux types : *la bonne classe est* v'_x, ou v_x *n'est pas la bonne classe*. Dans le premier cas, les objets et leur classe corrigée peuvent être ajoutés à l'ensemble d'apprentissage. Dans le second cas, le fait que l'objet o n'appartienne pas à la classe v_x est une information. Dans le cas d'une classe booléenne, cela indique même la classe correcte. Dans le cas général, cela signifie que o appartient à l'une des autres classes. La classification par portée peut facilement prendre en compte de tels exemples négatifs. Un exemple négatif ne de classe $classe(ne) = \neg v_x$ est un contre-exemple pour tout exemple de classe v_x et réciproquement, seuls les exemples de classe v_x sont des contre-exemples de ne lors de la construction de $(\epsilon, M) - cons(EA, \leq_o)$. L'ensemble des voisins d'un objet peut donc contenir des exemples négatifs. Ceux-ci peuvent être pris en compte de deux façons dans le critère de résolution : soit leur voix est répartie uniformément en faveur de toutes les autres classes, soit leur voix est soustraite des voix accordées à leur classe v_x. Nous avons retenu la seconde solution. Le nombre de voix accordées à une classe peut donc être négatif. Même dans le cas où toutes les classes sont créditées d'un score négatif, le choix d'une classe parmi celles obtenant le meilleur score (le plus grand pour l'ordre des nombres réels) est conforme à l'intuition : c'est en effet la classe la moins contre-indiquée. Considérons le cas où les scores associés aux classes 0, 1 et 2 sont respectivement -3, -1 et 0. Il y a donc trois voix contre la classe 0, c'est-à-dire pour la classe 1 ou la classe 2 et une voix contre la classe 1 donc pour la classe 0

ou la classe 2, c'est clairement la classe 2 qui est préconisée.

7.5 Résumé et conclusion

Dans ce chapitre, nous avons présenté diverses extensions de la classification par portée. Les premières extensions exposées concernent la prise en compte de données réelles. Nous avons détaillé les problèmes de non-déterminisme de l'association d'une classe à une description dans le cas de descriptions bruitées et dans le cas d'une représentation incomplète. La première solution que nous avons proposée consiste à modifier le pré-ordre $\leq_o$ utilisé par la classification par portée afin de prendre en compte les exemples incohérents. Une autre voie consiste à moduler le biais inductif logique de la classification par portée à l'aide de biais statistiques. Nous avons considéré pour cela deux paramètres proposés par Michèle Sebag dans le cadre de l'espace des versions disjonctif [Sebag, 1996]. Le paramètre de cohérence ϵ permet d'accepter des règles admettant un certain taux de contre-exemples. Le paramètre de généralité M permet d'admettre que des contre-exemples satisfassent une règle à M attributs près. Nous avons montré que ce paramètre admet plusieurs interprétations selon que l'on le considère du point de vue de la classification par portée ou du point de vue de l'espace des versions disjonctif.

Nous avons proposé un autre type d'extension de la classification par portée : la généralisation des exemples de l'ensemble d'apprentissage en règles, comme cela a été fait dans le cadre de techniques d'apprentissage à base d'instances fondées sur des distances, par exemple par Steven Salzberg [Salzberg, 1991] ou Pedro Domingos [Domingos, 1996]. L'apport escompté de cette généralisation préliminaire est d'une part une amélioration de la précision du classifieur, d'autre part, une réduction de l'ensemble d'apprentissage, et donc un temps de classification plus court, et une meilleure intelligibilité. Pour réaliser cette extension, nous avons généralisé les définitions de la classification par portée afin qu'elle s'applique à des règles, et non plus à des exemples seulement. Nous avons ensuite présenté plusieurs stratégies de généralisation des exemples en règles.

Nous avons enfin présenté deux extensions supplémentaires de notre approche. La première consiste à prendre en compte explicitement les redondances, les occurrences multiples d'un même couple $(description, classe)$ dans l'ensemble d'apprentissage, de façon à accélérer le processus de classification sans modifier l'influence des biais statistiques. La seconde extension concerne la prise en compte d'exemples négatifs, introduits par exemple par un professeur dans le cadre d'un apprentissage incrémental.

La plupart de ces extensions ont été mises en œuvre et sont évaluées dans le chapitre 9.

Chapitre 8

Les plus-proches-voisins de la classification par portée

Nous présentons dans ce chapitre un ensemble de techniques ayant des similitudes plus ou moins fortes avec la classification par portée. Nous considérons tout d'abord les approches statistiques, avant de nous intéresser aux approches fondées sur des distances. Nous considérons ensuite les approches combinant apprentissage à base d'instances et à base de règles, en particulier les approches à base d'instances généralisées en règles. Nous considérons enfin les approches fondées sur les mêmes biais logiques de cohérence et de pertinence que la classification par portée.

8.1 Approches statistiques

Des techniques non-paramétriques sont à l'origine d'approches telles que les k-plus proches voisins. Nous présentons dans ce paragraphe les fondements statistiques de ces approches. Nous détaillons en particulier les fenêtres de Parzen et les k_n-plus proches voisins, avant de montrer comment ces techniques peuvent être utilisées pour résoudre un problème de classification. L'ensemble des définitions fournies dans ce paragraphe peut être retrouvé dans l'ouvrage [Duda et Hart, 1973]. Nous comparons enfin ces approches statistiques avec la classification par portée.

8.1.1 Fondements

Le problème considéré est celui de l'estimation d'une fonction de densité de probabilité. Contrairement aux techniques paramétriques qui supposent que la forme de la fonction de densité de probabilité est connue et qui estiment uniquement les paramètres de la fonction de densité de probabilité, les approches non-paramétriques peuvent être utilisées quand la forme de la fonction de densité de probabilité est inconnue. Le fondement de ces techniques est que la probabilité P qu'un vecteur o (c'est-à-dire un objet) appartienne à une région $\mathcal{R}$ est donnée par

$$P = \int_{\mathcal{R}} p(y)dy. \tag{8.1}$$

Ainsi P représente une valeur moyenne de la fonction de densité $p(o)$, et cette valeur moyenne de p peut être estimée à l'aide de la probabilité P. Si l'on dispose de n exemples tirés indépendamment en fonction de la loi de probabilité $p(o)$, la probabilité que k d'entre eux appartiennent

à $\mathcal{R}$ est donnée par la loi binômiale

$$P_k = \binom{n}{k} P^k (1-P)^{n-k},$$

la valeur estimée de k est

$$E[k] = nP. \tag{8.2}$$

Supposons maintenant que $p(o)$ est continue et que la région $\mathcal{R}$ est suffisamment petite pour pouvoir écrire

$$\int_{\mathcal{R}} p(y)dy \simeq p(o)V, \tag{8.3}$$

où o est un point de $\mathcal{R}$ et V est le volume de $\mathcal{R}$. On peut déduire des formules 8.1, 8.2 et 8.3 l'estimation suivante de $p(o)$:

$$p(o) \simeq \frac{k/n}{V}. \tag{8.4}$$

Si nous fixons le volume V et considérons de plus en plus d'exemples, le rapport k/n va converger vers la valeur moyenne de $p(o)$ sur $\mathcal{R}$. Pour obtenir $p(o)$, nous devons faire tendre V vers 0. Cependant, si nous fixons le nombre n d'exemples et faisons tendre V vers 0, la région $\mathcal{R}$ va devenir si petite qu'elle ne contiendra plus d'exemples et l'estimation $p(o) \simeq 0$ est inutile. Or, en pratique, le nombre d'exemples est limité.

Voyons, d'un point de vue théorique, comment évaluer $p(o)$ si un nombre illimité d'exemples était disponible. Considérons la séquence de région $\mathcal{R}_1$, $\mathcal{R}_2$, ..., contenant o. La région $\mathcal{R}_1$ correspond au cas où on ne considère qu'un exemple, la région $\mathcal{R}_2$ au cas où on a tiré deux exemples et ainsi de suite. Soient V_n le volume de $\mathcal{R}_n$, k_n le nombre d'exemples appartenant à $\mathcal{R}_n$ et $p_n(o)$ la n-ième estimation de $p(o)$:

$$p_n(o) = \frac{k_n/n}{V_n}. \tag{8.5}$$

Pour que $p_n(o)$ converge vers $p(o)$, il faut que :

1. $lim_{n\to\infty} V_n = 0$
2. $lim_{n\to\infty} k_n = \infty$
3. $lim_{n\to\infty} k_n/n = 0$

La première condition assure que la valeur moyenne P/V converge vers $p(o)$ si la région rétrécit uniformément et si p est continue en o. La seconde condition, qui n'a de sens que si $p(o) \neq 0$, assure que le rapport des fréquences converge vers la probabilité P. La troisième condition est clairement nécessaire pour que l'équation 8.5 converge. Elle établit que bien qu'un grand nombre k_n d'exemples appartiennent à la région $\mathcal{R}_n$ quand n tend vers l'infini, ils ne représentent qu'une fraction négligeable du nombre total n d'exemples.

Il y a deux façons classiques pour construire une séquence de région satisfaisant ces conditions. La première consiste à rétrécir une région initiale en spécifiant le volume V_n par une fonction de n, par exemple $V_n = V_1/\sqrt{n}$. C'est le principe des fenêtres de Parzen. La seconde méthode consiste à spécifier k_n par une fonction de n, par exemple $k_n = k_1\sqrt{n}$. Dans ce cas, le volume V_n est ajusté de façon à contenir k_n voisins de o. C'est le principe des k_n-plus proches voisins.

8.1.2 Les fenêtres de Parzen

Supposons temporairement que la région $\mathcal{R}_n$ est un hyper-cube de dimension d (c'est-à-dire que les objets sont décrits par d attributs continus). Soit h_n la longueur d'une arête de l'hyper-cube, son volume vaut :

$$V_n = h_n^d.$$

Rappelons que la fonction caractéristique d'un hyper-cube unitaire centré à l'origine est définie par

$$\phi(u) = \begin{cases} 1 & |u_j| \leq 1/2 \text{ pour chaque attribut } j \\ 0 & \text{sinon.} \end{cases}$$

$\phi((o - e_i)/h_n)$ vaut donc 1 si l'exemple e_i appartient à l'hyper-cube de volume V_n centré en o, 0 sinon. Donc le nombre d'exemples appartenant à l'hyper-cube est donné par

$$k_n = \sum_{i=1}^{n} \phi(\frac{o - e_i}{h_n}).$$

Donc en remplaçant k_n dans la formule 8.5, nous obtenons

$$p_n(o) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{V_n} \phi(\frac{o - e_i}{h_n}). \tag{8.6}$$

Cette formule suggère une approche plus générale pour estimer une fonction de densité, en utilisant des classes de fenêtres plus générale que des hyper-cubes. La formule 8.6 estime $p(o)$ comme la moyenne de fonctions de o et des exemples e_i. Intuitivement, les fonctions caractéristiques des fenêtres servent à faire une interpolation, dans laquelle chaque exemple contribue à l'estimation en fonction de sa distance à o.

8.1.3 Les k_n-plus proches voisins

L'approche des fenêtres de Parzen requiert de choisir avec soin la séquence de fenêtres utilisée. Par exemple, si le volume choisi est $V_n = V_1/\sqrt{n}$, la n-ième estimation $p_n(o)$ est très sensible au choix du volume initial V_1. Si V_1 est trop petit, la plupart des volumes V_n ne contiendront aucun exemple et l'estimation $p_n(o)$ variera énormément. Inversement, si V_1 est trop grand, des variations de $p(o)$ n'apparaîtront pas à cause du calcul de la moyenne sur le volume de la fenêtre.

Une solution consiste à utiliser un volume fonction des données, à la place d'une fonction du nombre d'exemples. Par exemple, $p(o)$ peut être estimé à partir de n exemples en centrant une fenêtre sur o et en la faisant croître jusqu'à ce qu'elle contienne k_n exemples, où k_n est une fonction du nombre d'exemples n. Ces k_n exemples sont les k_n-plus proches voisins de o. Si la densité est élevée près de o, la fenêtre sera petite, sinon elle grossira plus vite mais s'arrêtera rapidement en entrant dans une zone de densité plus élevée.

8.1.4 Application à la classification

Les techniques que nous venons de présenter peuvent être utilisées pour estimer les probabilités *a posteriori* $P(x = v_i|o)$ à partir d'un ensemble d'exemples étiquetés en utilisant les exemples pour calculer les densités nécessaires au calcul. Supposons que nous placions une

fenêtre de volume V autour de o contenant k exemples, dont k_i appartiennent à la classe v_i. Une estimation évidente de la probabilité $p(o, x = v_i)$ est

$$P_n(o, x = v_i) = \frac{k_i/n}{V}.$$

Ainsi, un estimateur raisonnable de $P(x = v_i|o)$ est

$$P_n(x = v_i|o) = \frac{p_n(o, x = v_i)}{\sum_{j=1}^{c} p_n(o, x = v_j)} = \frac{k_i}{k},$$

où c est le nombre de classes. Donc l'estimation de la probabilité que o est de classe v_i est le taux d'exemples contenus dans la fenêtre étiquetés par v_i. La classe choisie est la classe la plus représentée dans la fenêtre.

8.1.5 Comparaison avec la classification par portée

Les analogies existant entre ces techniques non-paramétriques et la classification par portée consistent en l'utilisation d'hyper-rectangles et en l'utilisation d'un vote majoritaire étant donné un ensemble de voisins. Il est clair cependant que l'ensemble des voisins considérés n'est pas le même. Les techniques présentées dans ce paragraphe construisent une fenêtre de volume fixé arbitrairement dans le cas des fenêtres de Parzen ou contenant un nombre d'exemples fixé arbitrairement. La classification par portée construit des hyper-rectangles $C(o, e)$ uniquement pour vérifier que la règle $R(o, e)$ est cohérente à ϵ contre-exemples près. Chaque exemple à partir duquel une règle cohérente et pertinente à ϵ contre-exemples et à M attributs près peut être construite fait partie des voisins de l'objet o à classer au sens de la classification par portée. Dans notre cas, la taille de la fenêtre n'est pas fixée, l'ensemble des voisins retenus n'appartient même pas à une fenêtre commune, et le nombre de voisins n'est pas non plus fixé. Nous préciserons dans le paragraphe suivant les relations existant entre la classification par portée et les k-plus proches voisins du point de vue des distances utilisées.

8.2 Approches fondées sur des distances

La notion de pré-ordre sur laquelle repose la classification par portée est moins restrictive que les distances utilisées habituellement dans les techniques d'apprentissage à base d'instances. En effet, si e est plus proche de o que tout autre exemple pour la distance de Manhattan ou la distance de Hamming alors il appartient à $cons(EA, \leq_o)$ mais la réciproque est fausse. Ces distances sont définies dans un chapitre précédent (définition 4.5).

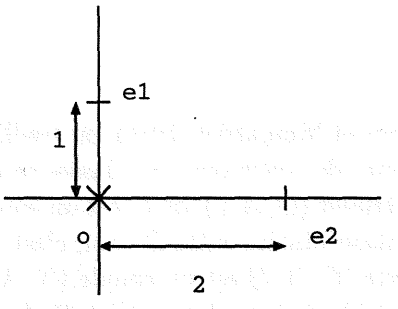
Propriété 8.1 *Dans le cas d'attributs tous nominaux (respectivement tous linéaires), tout exemple maximalement proche de o au sens de la distance de Hamming (resp. de la distance de Minkowski L_q) appartient à $cons(EA, \leq_o)$. Les réciproques sont fausses.*

Preuve : Raisonnons par l'absurde.
Supposons que e n'appartient pas $cons(EA, \leq_o)$, c'est-à-dire qu'il existe $e' \in CE(e, EA)$ tel que $e' \leq_o$. $e' \in C(o, e)$ signifie pour toutes les distances évoquées $d(o, e') \leq d(o, e)$. Donc e n'est pas l'exemple le plus proche de o.
Exhibons un contre-exemple de chacune des réciproques.
Dans le cas de la distance de Hamming, considérons l'exemple des patients,

Patient	a	b	c	d	x
e_1	V	V	V	V	V
e_2	F	F	F	F	F
e_3	V	F	F	F	V

$EA = \{e_1, e_2, e_3\}$ et $o = (a = V) \wedge (b = V) \wedge (c = V) \wedge (d = F)$, on a $e_3 \in cons(EA, \leq_o)$, pourtant sa distance de Hamming à o vaut 2 tandis que e_1 est à une distance de 1.

Dans le cas d'attributs linéaires, considérons le problème de classification suivant dans $\mathbb{R}^2$:



Quel que soit $q \in \mathbb{N}$ ou $q = \infty$, $L_q(o, e1) = 1$ et $L_q(o, e2) = 2$, donc $e2$ est plus loin de o que $e1$ et pourtant $e2 \in cons(EA, \leq_o)$. #

Les attributs sont considérés comme des points de vue différents dans la classification par portée, tandis qu'ils sont considérés simplement de même importance dans les approches fondées sur des distances. En particulier, d'un point de vue logique, dans le cas des patients, le fait que $o = (a = V) \wedge (b = V) \wedge (c = F) \wedge (d = F)$ ait en commun $\{(a = V), (b = V)\}$ avec l'exemple e_1 n'est pas équivalent au fait que o ait en commun $\{(c = F), (d = F)\}$ avec l'exemple e_2. De même dans le problème de classification dans $\mathbb{R}^2$ utilisé dans la preuve précédente, les règles $R(o, e1)$ et $R(o, e2)$ ne couvrent pas les mêmes exemples, il n'est donc pas possible d'en préférer une en s'appuyant sur des critères logiques seulement.

La classification par portée ne peut être considérée comme un algorithme du plus proche voisin que dans le cas très particulier où on considère que l'algorithme du plus proche voisin se caractérise par un vote entre l'ensemble des voisins et si on remplace les distances usuelles en apprentissage à base d'instances par une distance *ad hoc* définie pour retenir exactement un ensemble de voisins connu à l'avance. Considérons la mesure de similarité booléenne définie entre un exemple e et l'objet à classer o par $d_c(o, e) = 0$ si $e \in cons(EA, \leq_o)$, $d(o, e) = 1$ sinon. La classification par portée cherche, en effet, les plus proches voisins de o dans EA étant donné la mesure de similarité d_c, et applique un critère de résolution quand les plus proches voisins appartiennent à des classes distinctes ou quand aucun plus proche voisin n'est exhibé. On peut donc définir une distance à partir de $cons(EA, \leq_o)$. Muni de cette distance, l'algorithme du plus proche voisin permet d'associer à tout objet o la classe v_x si et seulement si il existe une règle de $RMIN(EA)$ couvrant o et concluant sur v_x.

Cette distance diffère de la distance d_{sc} fondée sur un ensemble de règles proposée par Michèle Sebag et Marc Schoenauer dans [Sebag et Schoenauer, 1994]. La distance d_{sc} est en effet définie à partir d'un ensemble de règles données *au préalable*. La distance d_{sc} entre deux points est le nombre de règles qui sont déclenchées par exactement un des deux points. En fait, si l'ensemble des règles sur la base desquelles d_{sc} est définie est l'ensemble de toutes les règles cohérentes et pertinentes pour EA, $d_c(o_1, o_2) = 1$ si $d_{sc}(o_1, o_2) = 0$, $d_c(o_1, o_2) = 0$ sinon.

8.3 Approches combinant apprentissage à base d'instances et à base de règles

La classification par portée constitue un point commun entre les approches à base d'instances et les approches à base de règles puisque c'est une approche à base d'instances fondée sur

les biais inductifs logiques de cohérence et de pertinence fondant les approches à base de règles. D'autres travaux reposent sur un lien entre les règles et les instances. Dans ce paragraphe, nous traitons tout d'abord le cas particulier du treillis de Galois [Barbut et Monjardet, 1970], puis nous considérons des approches à base d'instances où les instances sont généralisées en règles.

8.3.1 Treillis de Galois

Le treillis de Galois [Barbut et Monjardet, 1970], ou treillis de concepts [Wille, 1982], est un cadre mathématique qui permet de construire des classes se recouvrant à partir d'un ensemble d'objets. Un contexte est un triplet (O, A, I), où O est un ensemble d'objets, A est un ensemble de propriétés et I est une relation binaire entre O et A, c'est-à-dire $I \subseteq O \times A$. Chaque élément du treillis $\mathcal{T}$ dérivé du contexte (O, A, I) est un couple (X, X') où X est un ensemble d'objets, $X \in \mathcal{P}(O)$, et X' est un ensemble de propriétés, $X' \in \mathcal{P}(A)$, tels que :

- $X' = f(X)$ où $f(X) = \{x' \in A | \forall x \in X, (x, x') \in I\}$
- $X = f'(X')$ où $f'(X') = \{x \in O | \forall x' \in X', (x, x') \in I\}$

X est le plus grand ensemble d'objets satisfaisant l'ensemble des propriétés de X' et, symétriquement, X' est le plus grand ensemble de propriétés satisfaites par les objets X. La conjonction de propriétés X' est la description la plus spécifique de X. Le couple de fonctions (f, f') forme une connexion de Galois entre $\mathcal{P}(O)$ et $\mathcal{P}(A)$. Le treillis de Galois est ordonné par la relation d'ordre suivante : étant donnés deux couples, ou concepts, $C_1 = (X_1, X_1')$ et $C_2 = (X_2, X_2')$, $C_1 < C_2$ si et seulement si $X_1' \subseteq X_2'$. Il existe une relation de dualité entre l'extension X et l'intension X' d'un concept (X, X'), donc $C_1 < C_2$ si et seulement si $X_2 \subseteq X_1$. Cet ordre partiel est utilisé pour construire un graphe dans lequel une arête relie C_1 et C_2 si et seulement si $C_1 < C_2$ et il n'existe pas de concept C_3 tel que $C_1 < C_3 < C_2$. On dit que C_1 est couvert par C_2. Le graphe construit est appelé un diagramme de Hasse.

Le treillis de Galois peut être utilisé pour construire des règles de classification [Sahami, 1995]. Il est clair que l'intension d'un concept dont l'extension ne contient que des objets de même classe v_i constitue la partie gauche d'une règle cohérente et pertinente. Le théorème 6.1 établit que cette règle classe l'objet o si et seulement si il existe un élément de $cons(EA, \leq_o)$ de classe v_i. Alors que le temps et l'espace mémoire requis pour la construction du treillis de Galois sont exponentiels en le nombre d'attributs, la classification par portée d'un exemple est linéaire en le nombre d'attributs et quadratique dans le pire cas en le nombre d'exemples.

8.3.2 Les approches à base d'instances généralisées en règles

Nous présentons dans ce paragraphe plusieurs techniques de classification à base d'instances généralisées en règles. Ces techniques sont des approches à base d'instances. Elles classent les exemples qui ne sont pas couverts par les instances en cherchant ses plus proches voisins parmi un ensemble d'instances à l'aide d'une distance. Leur particularité est d'utiliser des instances qui ne sont pas simplement des éléments de l'ensemble d'apprentissage, mais des règles. Ces règles sont obtenues en itérant un processus de généralisation des exemples.

Exemples généralisés emboités NGE

Steven Salzberg a proposé un système appelé EACH basé sur ce qu'il appelle des exemples généralisés emboités, en anglais *Nested-Generalized Exemplars (NGE)*. EACH construit des

hyper-rectangles en considérant les exemples de l'ensemble d'apprentissage un-à-un. Il commence par prendre au hasard un nombre d'exemples défini par l'utilisateur et construit pour chaque exemple un hyper-rectangle ne contenant que ce point, donc une règle triviale. Pour chaque nouvel exemple e, la distance de e à chaque hyper-rectangle de l'ensemble courant d'hyper-rectangles est calculée. Si la classe de l'hyper-rectangle le plus proche est celle de e, alors l'hyper-rectangle le plus proche est étendu de façon à couvrir e, sinon le second hyper-rectangle le plus proche de e est essayé. Si aucun des deux hyper-rectangles les plus proches n'est étiqueté avec la même classe que e, alors e est enregistré comme un nouvel hyper-rectangle (correspondant à une règle triviale). Un nouvel objet est classé en fonction de la classe de l'hyper-rectangle le plus proche. La distance $D(o, H)$ d'un objet o à un hyper-rectangle H est mesurée de la façon suivante :

$$D(o, H) = w_H \times \sqrt{\sum_{i=1}^{d} (w_{a_i} \times (D_{a_i}(o, H))^2)}$$

où w_H est le poids accordé à l'hyper-rectangle H, w_{a_i} est le poids accordé à l'attribut a_i et $D_{a_i}(o, H)$ est la distance de o à H suivant l'attribut a_i :

$$D_{a_i}(o, H) = \begin{cases} v_i^e - v_{max,i}^H & \text{si } v_i^e > v_{max,i}^H \\ v_{min,i}^H - v_i^e & \text{si } v_i^e < v_{min,i}^H \\ 0 & \text{sinon} \end{cases}$$

où v_i^e est la valeur de l'attribut a_i dans e, et $v_{max,i}^H$ et $v_{min,i}^H$ les valeurs maximale et minimale de l'attribut a_i dans H. Si l'exemple est équidistant de plusieurs hyper-rectangles (en particulier s'il appartient à plusieurs hyper-rectangles, ce qui est possible puisque les hyper-rectangles peuvent être emboités), c'est le plus petit hyper-rectangle qui est choisi. L'algorithme original ne considère que des attributs linéaires sans aucune valeur manquante. Dietrich Wettschereck et Thomas Dietterich [Wettschereck et Dietterich, 1995] ont étendus les définitions de Steven Salzberg afin de prendre en compte les attributs nominaux et les valeurs manquantes.

L'approche NGE diffère clairement de la classification par portée. Si un objet à classer est couvert par un hyper-rectangle dans NGE ou une règle dans la classification par portée, l'objet est classé en accord avec la règle dans les deux techniques, sinon il est classé par rapport à son plus proche voisin au sens d'une distance dans NGE alors qu'il est classé en fonction de l'ensemble des exemples à partir desquels une règle cohérente et pertinente peut être construite dans la classification par portée. De plus la stratégie de généralisation des exemples n'est pas la même dans NGE que dans la classification par portée avec généralisation des exemples en règles. NGE ne génère qu'une règle à chaque étape, mais cette règle peut être généralisée plusieurs fois par la suite. La généralisation effectuée par NGE est incrémentale et dépend clairement de l'ordre de présentation des exemples. De plus, les règles peuvent volontairement se recouvrir. Nous ne générons que l'ensemble des règles R les plus générales telles que :

- R est cohérente et pertinente avec l'ensemble d'apprentissage EA ;
- il existe e_1 et e_2 appartenant à EA tels que $classe(e_1) = classe(e_2)$ et $R = R(e_1, e_2)$;

Clairement, les ensembles de règles engendrés sont différents, ainsi que la façon de classer des objets à partir de ces règles.

Exemples généralisés itérativement sans recouvrement BNGE

Dietrich Wettschereck et Thomas Dietterich [Wettschereck et Dietterich, 1995] ont proposé un algorithme appelé *Batch NGE (BNGE)* qui apporte plusieurs modifications fondamentales à NGE. Tout d'abord, BNGE commence avec tous les exemples en mémoire en les considérant comme des hyper-rectangles réduits à des points. Il les généralise ensuite itérativement en remplaçant deux hyper-rectangles H_1 et H_2 en l'hyper-rectangle H le plus petit couvrant H_1 et H_2 en respectant les conditions suivantes :

- H_1 et H_2 sont de même classe ;
- H_1 est l'hyper-rectangle le plus proche de H_2 ;
- H ne contient aucun objet appartenant à un hyper-rectangle d'une autre classe.

Cet algorithme est plus ressemblant à la classification par portée avec généralisation des exemples en règles que ne l'était NGE, cependant il en diffère sur de nombreux points :

- comme précédemment, la classification d'un nouvel objet à partir des règles est basée sur une distance dans le cas de NGE ;
- les règles construites dépendent de l'ordre de déroulement de l'algorithme : les premières règles sont plus facilement généralisées que les suivantes ;
- les règles construites par BNGE ne correspondent pas nécessairement à une règle généralisée dans le cadre de la classification par portée.

Le dernier point est illustré sur la figure 8.1. Si on considère dans $\mathbb{R}^3$ les exemples de même classe

$$\begin{aligned} e_1 &= (x=0),(y=0),(z=0) \\ e_2 &= (x=1),(y=1),(z=1) \\ e_3 &= (x=-1),(y=2),(z=2) \end{aligned}$$

Supposons que BNGE considère les exemples dans l'ordre e_1, e_2, puis e_3. Il construit tout d'abord l'hyper-rectangle

$$C(e_1,e_2) = (x \in [0;1]),(y \in [0;1]),(z \in [0;1]).$$

Il généralise ensuite $C(e_1,e_2)$ en l'hyper-rectangle H couvrant e_3

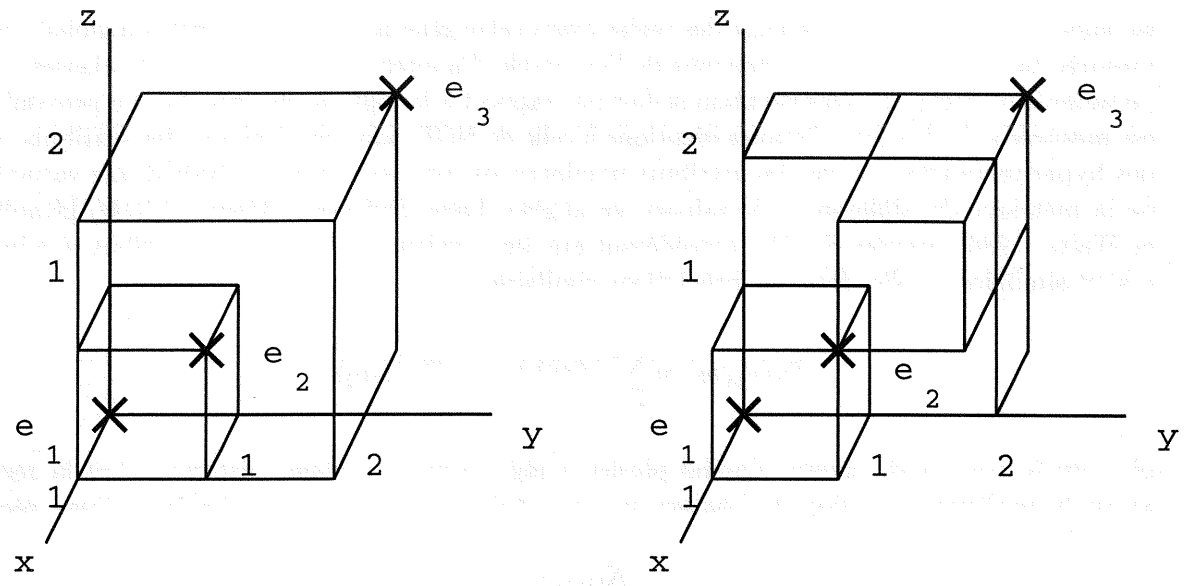
$$H = (x \in [-1;1]),(y \in [0;2]),(z \in [0;2])$$

La classification par portée généralise simplement les exemples en les règles de parties gauches :

$$\begin{aligned} C(e_1,e_2) &= (x \in [0;1]),(y \in [0;1]),(z \in [0;1]). \\ C(e_1,e_3) &= (x \in [-1;0]),(y \in [0;2]),(z \in [0;2]). \\ C(e_2,e_3) &= (x \in [-1;1]),(y \in [1;2]),(z \in [1;2]). \end{aligned}$$

Clairement, l'objet $x = (x = 0,5),(y = 1,5),(z = 0,5)$ appartient à H mais pas à $C(e_1,e_2)$, $C(e_1,e_3)$ ou $C(e_1,e_3)$.

La stratégie gloutonne de généralisation adoptée par BNGE explique qu'une partie des règles générées par BNGE soient plus générales que celles produites par la classification par portée. Mais, comme nous l'avons expliqué au paragraphe 7.3.3, cette stratégie privilégie les premières règles aux dépens des suivantes. Il est donc probable qu'une partie des règles générées par



BNGE **Classification par portée**

FIG. 8.1 – *Hyper-rectangles produits par BGNE et par la classification par portée.*

BNGE soient moins générales que celles produites par la classification par portée. Considérons à nouveau l'exemple ci-dessus et supposons que le point $x = (x = 0,5), (y = 1,5), (z = 0,5)$ soit un exemple de classe différente de celle de e_1, e_2 et e_3. L'ensemble des hyper-rectangles produits par la classification par portée est $C(e_1, e_2)$, $C(e_1, e_3)$, $C(e_2, e_3)$ et l'hyper-rectangle trivial x. L'ensemble des hyper-rectangles produits par BNGE en considérant e_1, e_2, e_3 puis x est $C(e_1, e_2)$ et les hyper-rectangles triviaux e_3 et x. Clairement les généralisations produites par BNGE sont moins générales que celles de la classification par portée dans ce cas. À vouloir produire des hypothèses plus générales, la stratégie gloutonne produit parfois des hypothèses moins générales que nécessaire. Un autre inconvénient majeur de cette stratégie est que les généralisations obtenues peuvent varier énormément en fonction de l'ordre de considération des exemples.

Dans le cas de BNGE, comme dans le cas de NGE, l'ensemble des règles produites diffèrent de celles produites par la classification par portée car les techniques de généralisation ne sont pas les mêmes, en particulier le principe de recherche des voisins repose sur une distance dans les premier cas alors que nous n'utilisons pas de distance et considérons tous les exemples couverts par les biais inductifs logiques de cohérence et de pertinence.

Exemples généralisés itérativement en fonction de la précision globale RISE

Pedro Domingos a proposé une technique de classification appelée RISE [Domingos, 1994; Domingos, 1995; Domingos, 1996] « unifiant » les approches à base d'instances et à base de règles dont l'idée est très proche de celle de BNGE. L'algorithme commence avec l'ensemble des exemples en les considérant comme des règles triviales. Il essaie ensuite itérativement de généraliser chaque règle afin de couvrir l'exemple le plus proche, de même classe et non-couvert par la règle. La règle est retenue si la précision globale de l'ensemble des règles ainsi obtenues

est supérieure à celle de l'ensemble des règles avant cette généralisation. La précision globale est mesurée par le pourcentage d'éléments de l'ensemble d'apprentissage correctement classés par l'ensemble de règles. La classification se fait par rapport à la règle la plus proche. La proximité est mesurée à l'aide d'une distance identique à celle de NGE, sans pondération des attributs, ni des hyper-rectangles. Le cas des attributs nominaux est pris en compte à l'aide d'une variante de la métrique des différences de valeurs, en anglais *Value Difference Metric (VDM)* [Stanfill et Waltz, 1986], appelée SVDM, considérant que deux valeurs v_1 et v_2 d'un attribut nominal a sont similaires si elles font des prédictions similaires :

$$D_a(v_1, v_2) = \sum_{k=1}^{c} |P(C_k|v_1) - P(C_k|v_2)|$$

où c est le nombre de classes. Quand plusieurs règles sont à la même distance, c'est la règle ayant la meilleure précision de Laplace qui est choisie. La précision de Laplace d'une règle $G \rightarrow D$ est définie par

$$\frac{N_{G \wedge D} + 1}{N_G + c}$$

où c est le nombre de classes, N_G et $N_{G \wedge D}$ sont les nombres d'éléments de l'ensemble d'apprentissage satisfaisants la partie gauche et, la partie gauche et la partie droite de la règle.

La façon de classer un nouvel objet n'est clairement pas la même dans RISE, où elle repose sur la recherche de la règle la plus proche et sur l'utilisation de la précision de Laplace, que dans la classification par portée, où elle consiste à chercher l'ensemble des instances à partir desquelles une règle cohérente et pertinente peut être construite. De plus, la façon d'engendrer des règles diffère également et les ensembles de règles obtenus sont distincts. RISE emploie une stratégie gloutonne et en a donc les inconvénients.

Exemples généralisés itérativement en règles cohérentes PmBc

L'algorithme PmBc proposé par Paul Monteanu [Monteanu, 1996] consiste à généraliser un exemple e non-couvert par l'ensemble courant de règles de la façon suivante :

```
fonction apprentissage_PmBc(e,E)
    R = e
    Pour tout exemple e de E du plus proche au plus éloigné faire
        Si la généralisation de R couvrant e est cohérente pour E
        Alors généraliser R pour couvrir e
    Rendre(R)
```

Cette généralisation ne conduit donc qu'à des règles cohérentes pour l'ensemble d'apprentissage. Cette technique diffère clairement de la classification par portée pour les raisons suivantes :

- Chaque règle construite ne correspond pas nécessairement à une règle $R(e_1, e_2)$. Le cas représenté à la figure 8.1 et les remarques faites dans le cas de BNGE s'appliquent à nouveau.

- PmBc utilise une stratégie de généralisation gloutonne.

8.4 Approches fondées sur les biais logiques de cohérence et de pertinence

Nous présentons enfin les approches fondées sur les mêmes biais logiques que la classification par portée.

8.4.1 Classification à base de toutes les règles cohérentes et pertinentes SE-Learn

Ron Rymon a proposé un algorithme d'énumération des règles utilisant les arbres d'énumération d'ensembles [Rymon, 1992; Rymon, 1993; Rymon, 1994; Rymon, 1996b]. Il construit l'ensembles des règles minimales cohérentes et pertinentes pour l'ensemble d'apprentissage, puis classe un nouvel objet o à l'aide d'un critère de résolution et de l'ensemble des règles couvrant o. Nous avons montré à travers le théorème 6.1 que chaque fois qu'une règle cohérente et pertinente couvrant o existe, un exemple de même classe appartient à l'ensemble des voisins retenus par la classification par portée, et réciproquement. Cette équivalence ne porte que sur l'existence d'au moins un exemple ou une règle de même classe. Elle ne garantit pas que le nombre de règles et d'exemples sont identiques. L'application du critère de résolution peut donc produire des classes différentes dans SE-Learn que dans la classification par portée.

8.4.2 L'espace des versions disjonctif

L'espace des versions disjonctif caractérise toutes les hypothèses cohérentes pour un ensemble d'apprentissage EA qui couvrent au moins un exemple de EA [Sebag, 1994; Sebag, 1996]. Il repose donc sur les biais inductifs logique de cohérence et de pertinence communs à la classification par portée et aux classifieurs à base de toutes les règles cohérentes et pertinentes. Le principe de l'espace des versions disjonctif est de construire un espace des versions pour chaque exemple e en considérant e comme seul exemple positif (germe) et ses contre-exemples $CE(e, EA)$ comme exemples négatifs. L'espace des versions obtenu est la disjonction de toutes les hypothèses cohérentes couvrant le germe e. C'est une forme de l'étoile, en anglais *star*, des algorithmes AQ [Michalski, 1983], et également la portée comme nous l'avons vu au paragraphe 6.1.

Rappelons la définition formelle de l'espace des versions disjonctif.

Définition 8.1 *L'*espace des versions disjonctif *pour l'ensemble d'apprentissage EA est la théorie $\mathcal{H}$ qui est la disjonction des étoiles $H(e)$ apprises à partir des germes e, pour chaque exemple e de EA.*

$H(e)$ contient toutes les hypothèses qui couvrent e et discriminent les contre-exemples de e : $H(e)$ est la conjonction des hypothèses $D(e, ce)$ qui couvrent e et discriminent ce, pour tous les contre-exemples $ce \in CE(e, EA)$ de e.

$D(e, ce)$ est la conjonction pour chaque attribut a des sélecteurs $SEL_a(e, ce)$ les plus généraux discriminant l'exemple e de son contre-exemple ce :

- *lorsque a est un nominal, $SEL_a(e, ce) = (a = v_a^e)$ si $v_a^e \neq v_a^{ce}$, sinon a n'intervient pas,*
- *lorsque a est linéaire, $SEL_a(e; ce) = (a < v_a^{ce})$ si $v_a^e < v_a^{ce}$, $SEL_a(e, ce) = (a > v_a^{ce})$ dans le cas contraire.*

Considérons l'exemple des patients

Patient	a	b	c	d	x
e_1	V	V	V	V	V
e_2	F	F	F	F	F
e_3	V	F	F	F	V

et construisons l'espace des versions $H(e_3)$ associé à l'exemple e_3. Le seul contre-exemple de e_3 est e_2. Le seul attribut discriminant e_3 et e_2, c'est-à-dire le seul attribut valué différemment dans e_3 et e_2 (donc permettant de les différencier), est l'attribut a. Le sélecteur sur l'attribut a est $SEL_a(e_3, e_2) =_d (a = Vrai)$, donc $D(e_3, e_2) =_d (a = Vrai)$ et $H(e_3) =_d D(e_3, e_2) =_d (a = Vrai)$. De même, $H(e_1) =_d D(e_1, e_2) =_d (a = Vrai) \vee (b = Vrai) \vee (c = Vrai) \vee (d = Vrai)$ et $H(e_2) =_d D(e_2, e_1) \wedge D(e_2, e_3) =_d ((a = Faux) \vee (b = Faux) \vee (c = Faux) \vee (d = Faux)) \wedge (a = Faux)$. L'espace des versions disjonctif pour l'ensemble des patients est $\mathcal{H} =_d H(e_1) \vee H(e_2) \vee H(e_3)$.

Considérons à présent un exemple dans un $UNIVERS$ décrit par deux attributs linéaires continus y et z. Étant donné un ensemble d'apprentissage constitué d'un exemple e de la classe v_x et de deux contre-exemples ce_1 et ce_2, voyons comment la classification par portée et l'espace des versions disjonctif procèdent pour classer un nouvel objet o (figure 8.2).

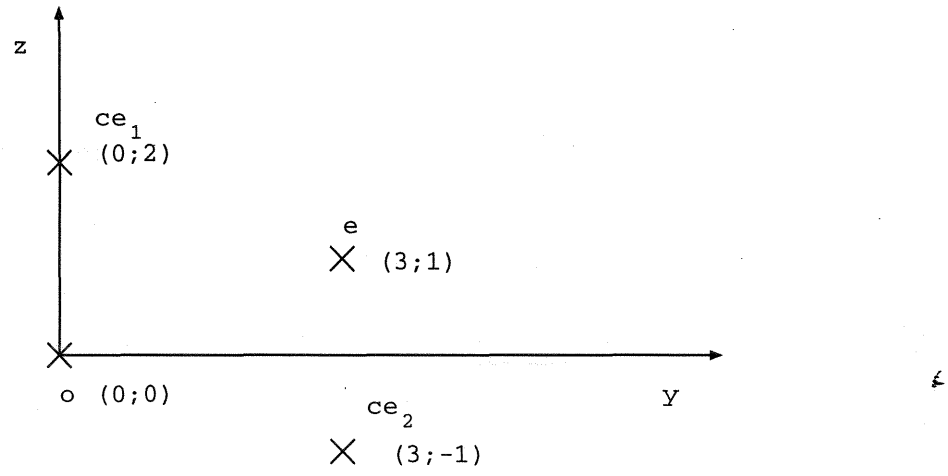


FIG. 8.2 – *Description du problème.*

L'espace des versions disjonctif est utilisé pour classer un nouvel objet o en déterminant ses voisins, c'est-à-dire les exemples dont l'espace des versions contient une hypothèse couvrant o. Les *voisins* de o sont donc les exemples e de EA tels que $H(e)$ contient o. La classe de o est déterminée par un vote majoritaire entre ses voisins. L'approche de l'espace des versions disjonctifs est centrée sur les exemples. Pour chaque exemple e de EA, la théorie $H(e)$ est construite. Dans le cas présent, $H(e) =_d D(e, ce_1) \wedge D(e, ce_2)$, où :

- $D(e, ce_1) =_d (y > 0) \vee (z < 2)$
- $D(e, ce_2) =_d (z > -1)$

$H(e)$ est représentée sur la figure 8.3. L'objet o est classé dans la classe v_x parce qu'il appartient à $H(e)$.

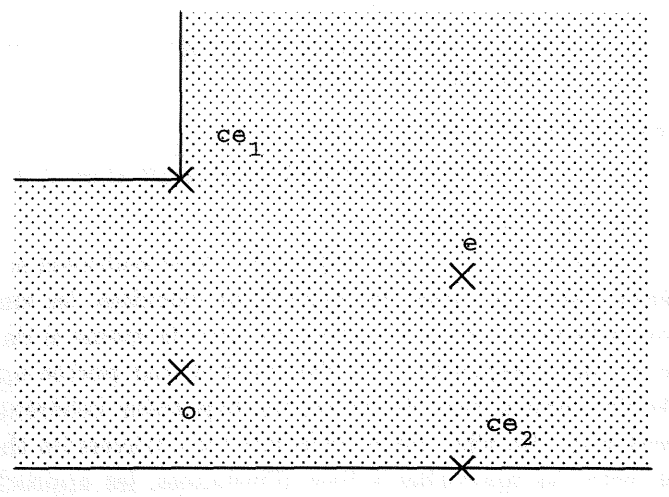


FIG. 8.3 – *L'hypothèse construite par l'espace des versions disjonctif.*

La classification par portée adopte le point de vue de l'objet o et cherche les exemples e' tels que la règle $R(o, e')$ couvrant o et e' est cohérente. Il faut donc vérifier que $R(o, e')$ n'admet aucun contre-exemple dans EA. La classification par portée construit donc la règle :

$$R(o, e) =_d (y \in [0; 3]) \wedge (y \in [0; 1]) \rightarrow (x = v_x)$$

et vérifie qu'elle ne couvre aucun des contre-exemples ce_1 et ce_2 (figure 8.4). La classification par portée se focalise sur l'objet à classer o. Elle détermine, à partir de o, les exemples e tels que $e \in cons(EA, \leq_o)$.

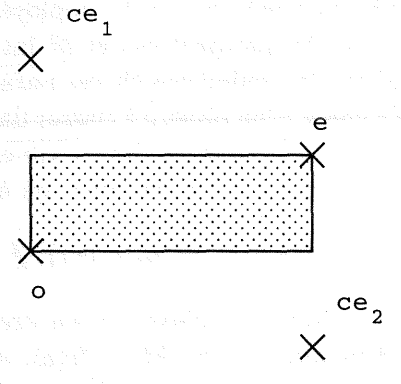


FIG. 8.4 – *L'hypothèse construite par la classification par portée.*

Clairement, les hypothèses construites par l'espace des versions disjonctif et la classification par portée ne sont pas les mêmes. Cependant l'ensemble des voisins de o dans l'espace des versions disjonctif est égal à l'ensemble $cons(EA, \leq_o)$ des voisins de o dans la classification par portée.

Théorème 8.1 *Étant donnés un objet o à classer et un exemple e, e est un voisin de o dans l'espace des versions disjonctif, c'est-à-dire $o \in H(e)$, si et seulement si e est un voisin de o dans la classification par portée, c'est-à-dire $e \in cons(EA, \leq_o)$.*

Preuve : Soit $e \in EA$, $o \in H(e)$ si et seulement si $\forall ce \in CE(e, EA), e \in D(e, ce)$, c'est-à-dire il existe a tel que o satisfait $SEL_a(e, ce)$

- si a est nominal, $v_a^o = v_a^e$ et $v_a^e \neq v_a^{ce}$,
- si a est linéaire, si $v_a^e > v_a^{ce}$ alors $v_a^o > v_a^{ce}$ sinon, si $v_a^e < v_a^{ce}$ alors $v_a^o < v_a^{ce}$.

donc $v_a^{ce} \notin C_a(o, e)$, c'est-à-dire $ce \notin C(o, e)$.
Donc $o \in H(e)$ si et seulement si $\forall ce \in CE(e, EA), ce \not\leq_o e$ si et seulement si $e \in cons(EA \leq_o)$. #

Malgré la différence entre la forme des hypothèses calculées, les ensembles de voisins sont identiques. La classe affectée à un nouvel objet est donc la même si on utilise le même critère de résolution dans les deux approches. La classification par portée apporte un point de vue différent sur la classification à base de toutes les hypothèse cohérentes et pertinentes pour l'ensemble d'apprentissage. En effet, nous avons montré à travers les théorèmes 6.1 et 8.1 qu'il existe une relation entre les approches à base d'instances, les approches à base de règles et les approches fondées sur l'espace des versions. Nous avons également montré dans ce chapitre les relations de ces trois approches avec les approches statistiques, les approches fondées sur des distances et les approches combinant l'apprentissage à base d'instances et à base de règles. La classification par portée bénéficie de plus d'une algorithme de calcul relativement efficace. L'algorithme de classification proposé par Michèle Sebag [Sebag, 1994; Sebag, 1996] a une complexité en temps en $\mathcal{O}(d \times |EA|^2)$ dans tous les cas. L'approche « diviser pour régner » utilisée par la classification par portée a un coût en temps plus proche de $d \times |EA| \times log_2(|EA|)$ (une complexité en temps quadratique est seulement atteinte dans le pire cas). Il est clair que l'espace des versions disjonctif peut se dispenser de la construction de chaque espace des versions et procéder directement à la classification d'un nouvel exemple à partir des exemples. Il peut donc faire de la classification par portée et bénéficier également d'une approche « diviser pour régner ».

Nous pouvons vérifier que les paramètres ϵ et M employés dans la classification par portée produisent les mêmes résultats que les paramètres ϵ et M introduits dans l'espace des versions disjonctif. Rappelons tout d'abord les définitions de ces paramètres dans ce cadre.

Un exemple e est un (ϵ, M)-voisin d'un objet o à classer dans l'espace des versions disjonctif, $o \in (\epsilon, M) - H(e)$, si et seulement si le nombre des contre-exemples ce tels que o ne satisfait pas au moins M des sélecteurs $SEL_a(e, ce)$ est inférieur ou égal à $\epsilon \times |EA|$, formellement

$$|\{ce \in CE(e, EA)/|\{a/o \models SEL_a(e, ce)\}| \not\geq M\}| \leq \epsilon \times |EA|.$$

Théorème 8.2 *Étant donnés un objet o à classer et un exemple e, e est un (ϵ, M)-voisin de o dans l'espace des versions disjonctif, $o \in (\epsilon, M) - H(e)$, si et seulement si e est (ϵ, M)-un voisin de o dans la classification par portée, c'est-à-dire $e \in (\epsilon, M) - cons(EA, \leq_o)$.*

Preuve : Nous avons montré dans la preuve du théorème 8.1 que $o \models SEL_a(e, ce)$ si et seulement si $v_a^{ce} \notin C_a(o, e)$. Donc $|\{a/o \models SEL_a(e, ce)\}| \not\geq M$ si et seulement si $|\{a/v_a^{ce} \notin C_a(o, e)\}| \not\geq M$, c'est-à-dire $|\{a/v_a^{ce} \notin C_a(o, e)\}| < M$.

Ainsi $o \in (\epsilon, M) - H(e)$ si et seulement si $|\{ce \in CE(e, EA)/|\{a/o \models SEL_a(e, ce)\}| \not\geq M\}| \leq \epsilon \times |EA|$ si et seulement si $|\{ce \in CE(e, EA)/|\{a/v_a^{ce} \notin C_a(o, e)\}| < M\}| \leq \epsilon \times |EA|$ si et seulement si $|\{ce \in CE(e, EA)/ce \leq_o^M e\}| \leq \epsilon \times |EA|$ si et seulement si $e \in (\epsilon, M) - cons(EA, \leq_o)$. #

Nous avons proposé dans le chapitre précédent une procédure de réglage automatique des paramètres ϵ et M qui peut clairement être utilisée par l'espace des versions disjonctif dans lequel leurs valeurs sont fixées manuellement [Sebag, 1996]. Il est évident que le paramètre ϵ a le même sens dans les deux approches. Le paramètre M prend cependant des significations différentes dans l'espace des versions disjonctif et dans la classification par portée. Dans le

premier cas, un objet o appartient à l'espace discriminant l'exemple e du contre-exemple ce si o satisfait au moins M sélecteurs discriminant ce de e. Il faut donc que o satisfasse au moins M conditions pour chacun des contre-exemples (à $\epsilon \times |EA|$ près). Dans le cas de la classification par portée, un exemple e appartient au voisinage de o si la règle $R(o, e)$ n'admet pas de contre-exemple (pas plus de $\epsilon \times |EA|$) à M attributs près. Nous avons montré au paragraphe 7.2.2 que M peut admettre plusieurs autres interprétations. L'utilisation de la mesure de spécificité $s(o, e)$ semble en particulier offrir des perspectives de développement de la classification par portée. La différence de point de vue de la classification par portée permet d'envisager des développements qui n'ont pas de sens intuitif dans l'espace des versions disjonctifs. Par exemple, la généralisation des exemples en règles mise en oeuvre dans le cadre de la classification par portée ne se conçoit qu'en travaillant avec des règles et plus difficilement avec des espaces des versions. De même, nous envisageons d'introduire une modulation de la pertinence des règles considérées. La pertinence est clairement associée à la notion de règle et semble difficilement se transposer dans le cadre de l'espace des versions disjonctif.

8.5 Résumé et conclusion

Dans ce chapitre, nous avons présenté un ensemble de techniques de classification ayant des similarités avec la classification par portée. Nous nous sommes attachés à préciser dans chaque cas les liens entre ces approches. Nous avons dans un premier temps considéré les approches statistiques. Ces approches consistent essentiellement à construire une fenêtre autour de l'objet à classer pour lui associer la classe majoritairement représentée par les exemples contenus dans cette fenêtre. La classification par portée ne repose pas sur la construction d'une telle fenêtre. Nous avons considéré ensuite les approches fondées sur des distances et montré que le plus proche voisin au sens de distances usuelles est un voisin au sens de la classification par portée, mais la réciproque est clairement fausse. De manière générale, la classification par portée ne fait aucune hypothèse sur le nombre de voisins, ni sur la taille du voisinage considéré. Nous avons ensuite considéré des approches combinant instances et règles. Nous avons ainsi montré que le treillis de Galois n'est pas une approche à base d'instances, mais plutôt une approche à base de règles. De ce point de vue, il peut permettre de calculer l'ensemble des règles cohérentes et pertinentes, mais à un coût exponentiel en le nombre d'attributs. Nous avons porté un intérêt particulier aux approches à base d'instances généralisées en règles. Quelque soit l'approche considérée, NGE [Salzberg, 1991], BNGE [Wettschereck et Dietterich, 1995] ou RISE [Domingos, 1996], ces approches reposent sur une distance et généralisent les exemples suivant une stratégie gloutonne. La classification par portée ne repose pas sur une distance, mais sur des considérations logiques : les biais de cohérence et de pertinence. De plus, aucune des stratégies de généralisation que nous avons retenues n'est gloutonne. En particulier, les généralisations obtenues sont indépendantes de l'ordre de considération des exemples. Nous avons enfin considéré les approches fondées sur les biais logiques de cohérence et de pertinence. Nous avons rappelé la correspondance montrée précédemment entre l'ensemble des règles cohérentes et pertinentes et la classification par portée. L'équivalence entre ces deux techniques est purement logique et le recours à un critère de résolution ne garantit pas d'obtenir des classifieurs identiques. Nous avons montré au contraire que la classification par portée et l'espace des versions disjonctif produisent les mêmes classifieurs à critères de résolution identiques. Pourtant ces deux techniques construisent des hypothèses de forme très différentes. Certains développements de la classification par portée peuvent être utilisés dans le cadre de l'espace des versions disjonctif, par exemple le calcul du voisinage suivant une approche « diviser

pour régner » ou la procédure de réglage automatique des paramètres. D'autres développements n'ont de sens que dans la classification par portée et montrent que le fait d'utiliser des règles à la place d'un espace des versions, outre un point de vue différent, permet d'envisager des développements difficiles à concevoir du point de vue d'un espace des versions. Nous avons ainsi étendu la classification par portée afin de généraliser les exemples en règles. Nous envisageons par exemple de moduler la pertinence des règles considérées. Ces développements sont plus intuitifs dans notre approche que dans celle de l'espace des versions disjonctif.

Chapitre 9

Évaluation expérimentale

Les performances en précision et en temps d'exécution de la classification par portée avec et sans généralisation des exemples en règles sont présentées dans ce chapitre. Ces performances sont comparées à celles d'autres techniques de classification, à base de règles et à base d'instances, représentées par les versions les plus récentes des programmes les plus connus de chaque approche : PEBLS 3.0 [Cost et Salzberg, 1993] pour les approches à base d'instances, CN2 [Clark et Niblett, 1989] et C4.5 [Quinlan, 1993a] qui ne génèrent qu'un petit nombre de règles « choisies », et SE-Learn [Rymon, 1996a], qui construit explicitement toutes les règles minimales cohérentes et pertinentes pour l'ensemble d'apprentissage, pour les approches à base de règles. Nous mentionnons pour chaque problème les résultats fournis par le classifieur par défaut (choix de la classe la plus fréquente) à titre de référentiel.

Nous présentons tout d'abord la plate-forme d'évaluation que nous avons utilisée. Nous donnons ensuite les résultats d'une comparaison sur un ensemble important de bases de données réelles, dont la plupart sont des bases de tests usuelles dans la communauté d'apprentissage automatique. Les bases restantes concernent des problématiques propres à l'équipe RFIA : l'aide au diagnostic médical et la reconnaissance de la parole. Nous commentons plus particulièrement ces derniers résultats.

9.1 La plate-forme d'évaluation

Nous avons évalué les performances en précision de classification et en temps d'exécution de chaque programme. Nous avons également évalué la taille et le nombre de règles des ensembles d'apprentissage obtenus par la généralisation des exemples en règles dans la classification par portée. Les programmes ont été évalués sur chaque ensemble de données à l'aide d'une validation croisée en dix sous-ensembles. Nous décrivons dans les paragraphes suivants les programmes comparés et leurs paramètres.

Nous n'évaluerons pas l'algorithme de classification fondé sur l'espace des versions disjonctif développé par Michèle Sebag [Sebag, 1994; Sebag, 1996]. Cet algorithme a nécessairement la même précision que la classification par portée sans généralisation des exemples en règles, à condition d'utiliser le même critère de résolution et les mêmes valeurs des paramètres ϵ et M. Il est donc inutile d'évaluer sa précision. Par ailleurs, sa complexité est quadratique dans tous les cas, alors que notre algorithme n'est quadratique que dans le pire cas.

Les approches connexionnistes n'ont pas été considérées dans cette évaluation bien que le perceptron multi-couches soit un classifieur répandu. La mise en œuvre de la rétro-propagation requiert, en effet, un réglage long et minutieux et un temps d'apprentissage trop long pour

pouvoir être appliquée sur le grand nombre de tests que nous avons effectués.

9.1.1 Validation croisée

La base de données testée E est divisée aléatoirement en dix sous-ensembles de même taille. Chaque sous-ensemble constitue un ensemble de test ET et est classé en fonction des neuf autres sous-ensembles, constituant l'ensemble d'apprentissage EA. La moyenne des résultats obtenus lors des 10 tests est une estimation des résultats qui pourraient être obtenus sur la base complète.

9.1.2 La classification par portée et ses variantes

Nous avons comparé trois variantes de la classification par portée utilisant ou non la généralisation des exemples en règles. Ces trois variantes utilisent toutes les paramètres ϵ et M. Le réglage de ces paramètres se fait automatiquement de la façon décrite au paragraphe 7.2.3 en utilisant un ensemble de données de réglage d'un dixième de la base de données initiale (limité à 1500 exemples). Lorsqu'un dixième de la base ne contient pas au moins dix exemples, nous avons utilisé un ensemble de réglage correspondant à un ensemble d'apprentissage lors de la validation croisée, c'est-à-dire neuf dixièmes de la base complète. Le critère de résolution (CR) est choisi de la même façon entre le vote majoritaire simple et le vote majoritaire quadratique. Le critère de résolution retenu est celui qui donne la meilleure précision lors d'une validation croisée sur l'ensemble de données de réglage. Les valeurs de CR, ϵ et de M sont alors utilisées pour les 10 tests de la validation croisée sur la base complète. Une fois les paramètres fixés, les redondances de la base de données sont prises en compte comme expliqué au paragraphe 7.4.1.

Les différentes options de la classification par portée sont intégrées dans un même programme. Toutes les combinaisons d'options peuvent donc être utilisées. L'ensemble du prototype réalisé en langage C comporte plus de 3000 lignes. Nous utiliserons ce programme avec trois configurations notées cons, gen et gen_cons.

cons

La première version évaluée est la classification par portée à partir d'exemples seulement. L'ensemble des voisins utilisé est $(\epsilon, M) - cons(EA, \leq_o)$. Les exemples incohérents ne sont pas pris en compte explicitement. Cela signifie qu'ils s'éliminent mutuellement lors de la classification (cf. paragraphe 7.1.2).

gen

La seconde version de la classification par portée calcule d'abord l'ensemble des règles minimales (cf. paragraphe 7.3.3), $gen(EA)$, en utilisant $(\epsilon, M) - cons(EA, <_o)$. Puis l'ensemble des voisins utilisé est $(0, 1) - cons(gen(EA), <_o)$. Les exemples incohérents sont pris en compte explicitement comme décrit au paragraphe 7.1.2.

gen_cons

La troisième version de la classification par portée calcule l'ensemble des règles minimales cohérentes entre elles (cf. paragraphe 7.3.3), $gen\_cons(EA)$ en utilisant $(\epsilon, M) - cons(EA, \leq_o)$. Puis l'ensemble des voisins utilisé est $(0, 1) - cons(gen\_cons(EA), \leq_o)$. Les exemples incohérents ne sont pas pris en compte explicitement. Cela signifie qu'ils s'éliminent mutuellement lors de la classification (cf. paragraphe 7.1.2).

9.1.3 Les approches à base d'instances : PEBLS

Les approches à base d'instances sont représentées par PEBLS 3.0 [Cost et Salzberg, 1993]. Ce programme est accessible par ftp anonyme (ftp://condor.cs.jhu.edu/pub/pebls). Nous avons utilisé les paramètres par défaut du programme à l'exception de la pondération des exemples. Les exemples sont pondérés par leur taux d'« utilisation correcte » comme cela est décrit dans [Cost et Salzberg, 1993]. Ce taux est évalué sur 10 essais.

9.1.4 Les approches à base de quelques règles « choisies » : CN2 et C4.5

Nous considérons deux groupes de classifieurs à base de règles. Le premier groupe contient des approches qui n'utilisent qu'un petit nombre de règles. Ces règles ne sont pas toujours cohérentes et pertinentes pour l'ensemble d'apprentissage. C'est le cas en particulier de C4.5, qui utilise un élagage statistique. Les approches à base de règles choisies sont CN2 [Clark et Niblett, 1989; Clark et Boswell, 1991] et C4.5 [Quinlan, 1993a].

CN2

Nous avons utilisé la version (6.1) de CN2 avec ses paramètres par défaut [Clark et Boswell, 1991]. Ce programme est accessible par ftp anonyme (ftp://ftp.cs.utexas.edu/pub/pclark/cn2.tar.Z).

C4.5

C4.5 construit des arbres de décision en utilisant le ratio du gain d'information. Ce programme est fourni avec l'ouvrage du même nom [Quinlan, 1993a]. Nous l'avons configuré pour utiliser la génération de règles et le fenêtrage, en générant 10 arbres. 10 arbres est la valeur par défaut lorsque le fenêtrage est utilisé. Les paramètres d'élagage ont été fixés à 4 exemples au minimum au lieu de 2 pour introduire un test, et le seuil de confiance à 37.5% au lieu de 25%.

9.1.5 Les approches à base de toutes les règles minimales cohérentes et pertinentes : SE-Learn

Le second groupe de classifieurs à base de règles utilisés contient des classifieurs utilisant toutes les règles minimales cohérentes et pertinentes pour l'ensemble d'apprentissage (en relâchant parfois la contrainte logique de cohérence des règles pour l'ensemble d'apprentissage). Dans ce groupe, nous ne considérons que les arbres d'énumération d'ensemble utilisés avec et sans élagage. Ces arbres sont construits par le programme SE-Learn de Ron Rymon [Rymon, 1996a]. Les paramètres par défaut ont été retenus. Le critère de résolution utilisé est le vote majoritaire simple pour tous les tests.

SE

SE-Learn a été utilisé dans une première configuration sans élagage afin de calculer toutes les règles minimales cohérentes et pertinentes pour l'ensemble d'apprentissage.

SE 0.05

SE-Learn a été utilisé dans une seconde configuration avec un élagage statistique des règles pour un seuil $p < 0.05$.

9.1.6 Un référentiel : le classifieur par défaut

Le classifieur par défaut est le classifieur qui classe tous les objets dans la classe majoritairement représentée dans l'ensemble d'apprentissage. Nous avons mentionné la précision de la classification ainsi obtenue à titre de référentiel.

9.2 Évaluation globale des performances en précision et en temps d'exécution

Les performances de la classification par portée, en précision et en temps d'exécution, ont été évaluées ainsi que celles des autres approches à l'aide de la plate-forme d'évaluation que nous venons de présenter. Chaque classifieur a été évalué sur la même base d'apprentissage et la même base de test que les autres classifieurs. Ce processus a été répété dix fois par validation croisée. L'ensemble des bases de données qui ont été utilisées est présenté dans un premier temps. Nous étudions ensuite quelles conclusions peuvent être tirées de cette évaluation du point de vue de la précision d'une part, et du point de vue du temps de calcul d'autre part. Les résultats que nous analysons concernent l'ensemble des bases de test pris dans sa totalité. Les résultats de chaque base de données sont discutés en annexe B. Celles d'entre elles qui nous intéressent plus particulièrement font l'objet du paragraphe suivant.

9.2.1 Caractéristiques des bases de test utilisées

La plupart des bases de test utilisées contiennent des données décrivant des problèmes réels. Certaines présentent donc les défauts présentés au paragraphe 7.1, à savoir des descriptions bruitées, des valeurs manquantes, des exemples incohérents. Dans ces cas, l'association description/classe n'est pas nécessairement déterministe. Les caractéristiques de chaque base, aussi appelée domaine ou problème de classification, sont résumées dans le tableau 9.1.

La plupart de ces ensembles de test proviennent du répertoire de l'Université d'Irvine en Californie où sont déposées les bases de test utilisées pour l'évaluation des algorithmes d'apprentissage automatique [Merz et Murphy, 1996]. Les données contenues dans ces bases proviennent de problèmes réels. Leurs contenus et les résultats les concernant sont détaillés en annexe B. Nous avons noté quelques différences entre les caractéristiques que nous avons mesurées et reportées, et celles mentionnées par Pedro Domingos [Domingos, 1996]. Ces différences sont notées en gras sur le tableau 9.1. Des précisions supplémentaires sur chacune de ces bases figurent en annexe B. Les bases restantes nous ont été fournies par des chercheurs de l'équipe RFIA. Ce sont les bases Diagnostic et Malades qui concernent l'aide au diagnostic dans le domaine médical, et 2 bandes, 3 bandes et 4 bandes qui concernent la reconnaissance de la parole.

L'ensemble de ces bases couvre une grande partie des domaines réels concernés par l'apprentissage automatique : médical, reconnaissance de formes, aide à la décision, identification, ... Les caractéristiques de ces ensembles de données sont variées. Ces bases sont de tailles diverses et comportent un nombre variable d'attributs. Ces attributs sont tous nominaux, tous linéaires, ou mixtes. Le nombre de classe varie également. Ce dernier paramètre est important car, à nombre d'exemples égal, plus le nombre de classe est grand, moins le nombre moyen d'exemples par classe est grand.

Nom	code	|E|	Att.	Nom.	Discr.	Cont.	Cl.	Manqu.	Incoh.
Audiology	AD	200	69	69	0	0	24	Oui	Non
Annealing	AN	798	38	29	6	3	6	Oui	Non
Credit screening	CE	690	15	9	0	6	2	Oui	Non
Pima diabetes	DI	768	8	0	0	8	2	Non	Non
Diagnostic	DIAG	198	153	43	103	7	12	Oui	Non
2 bandes	DICO2	8189	4	4	0	0	29	Non	Oui
3 bandes	DICO3	14132	4	4	0	0	29	Non	Oui
4 bandes	DICO4	30608	4	4	0	0	29	Non	Oui
Echocardiogram	EC	131	6	1	0	5	2	Oui	Oui
Glass	GL	214	9	0	0	9	**7**	Non	Non
Heart disease (Cleveland)	HDc	302	13	6	4	3	2	Oui	Non
Heart disease (Hungarian)	HDh	293	13	6	4	3	2	Oui	Non
Heart disease (Swizerland)	HDs	122	13	6	4	3	2	Oui	Non
Heart disease (va)	HDv	199	13	6	4	3	2	Oui	Non
Hepatitis	HE	155	19	13	0	6	2	Oui	Non
Horse colic	HO	368	21	11	3	7	2	Oui	Oui
Iris	IR	150	4	0	0	4	3	Non	Non
Labor negotiations	LA	57	16	8	6	2	2	Oui	Non
Lung cancer	LC	**33**	56	56	0	0	3	Oui	Non
Liver disease	LD	**344**	5	0	0	5	2	Non	Non
Contact lenses	LE	24	4	4	0	0	3	Non	Non
LED	LI	100	7	7	0	0	10	Non	Non
Malades	MAL	355	153	43	103	7	2	Oui	Non
Post-operative	PO	90	8	0	8	0	3	Oui	Oui
DNA promoters	PR	106	57	57	0	0	2	Non	Non
Solar flare (common)	SFc	323	10	6	4	0	4	Non	Oui
Solar flare (moderate)	SFm	323	10	6	4	0	4	Non	Oui
Solar flare (severe)	SFx	323	10	6	4	0	4	Non	Oui
Soybean	SO	**48**	35	35	0	0	4	Non	Non
Splice junctions	SP	3190	60	60	0	0	3	Oui	Oui
Voting records	VO	435	16	16	0	0	2	**Non**	Non
Wine	WI	178	13	0	0	13	3	Non	Non
Zoology	ZO	101	16	16	0	0	7	Non	Non

TAB. 9.1 – *Ensembles de données utilisés dans l'évaluation expérimentale. Les colonnes mentionnent de gauche à droite: le nom du domaine, le code utilisé dans les tableaux suivants, le nombre d'exemples qu'il contient |E|, le nombre d'attributs et leur répartition en attribut nominaux, linéaires discrets et linéaires continus, le nombre de classes, la présence de valeurs manquantes et d'exemples incohérents.*

9.2.2 Comparaison des précisions

La précision de la classification est mesurée par le pourcentage d'exemples de test correctement classés *PCC*. La valeur moyenne ainsi que l'écart-type du PCC mesuré sur les 10 tests lors de la validation croisée est donnée dans le tableau 9.2.2. L'exposant placé après l'écart-type pour chaque classifieur différent de cons mesure le seuil de confiance dans l'hypothèse statistique : « la différence observée entre la précision de la classification par portée sans généralisation des exemples en règles (cons) et celle du classifieur est significative ». Ce seuil de confiance a été calculé à l'aide d'un test t en séries appariées unilatéral [Doly *et al.*, 1994]. L'exposant 1 indique une confiance supérieure à 99,5%, l'exposant 2 correspond à 99%, 3 à 97,5%, 4 à 95%, 5 à 90% et 6 indique que le seuil de confiance est inférieur à 90% donc que la différence n'est pas significative. Nous examinons tout d'abord la précision de la classification par portée sans généralisation des exemples en règles (cons) par rapport à celles des autres classifieurs. Nous évaluons ensuite celles des deux configurations avec généralisation des exemples en règles (gen) et (gen_cons).

La classification par portée à partir d'exemples seulement : cons

La lecture de l'ensemble des résultats pour chaque domaine ne permet pas aisément de distinguer des tendances entre les performances des classifieurs. Nous avons donc utilisé cinq mesures de leurs performances relatives qui sont reportées dans le tableau suivant :

Mesure	cons	gen	gen_cons	PEBLS	C4.5	CN2	SE	SE 0.05	Défaut
Nb. vict.	-	-	-	20-12	22-10	25-6	13-4	9-10	29-0
Nb. vict. signif.	-	-	-	13-5	15-1	12-3	2-0	5-4	22-0
Wilcoxon	-	-	-	96,5	99,5	99,9	80	<80	100
Moyenne	81,1	71,1	75,5	78,9	79,3	77,9	82,8	79,3	53,6
Rang	0,76	0,39	0,41	0,55	0,56	0,50	0,67	0,69	0,2

Examinons les résultats de chaque mesure :

Nombre de victoires. Ce test consiste simplement à compter le nombre n_1 de fois où la précision moyenne de la classification par portée à partir d'exemples seulement est strictement supérieure à celle du classifieur considéré et le nombre n_2 de fois où l'inverse se produit. Les cas d'égalité ne sont pas comptés. Le résultat porté dans le tableau est $n_1 - n_2$. Ainsi cons a obtenu une précision strictement supérieure à celle de PEBLS sur 20 problèmes de classification, alors que l'inverse ne s'est produit que sur 12 domaines. Les deux algorithmes ont obtenus la même précision sur un domaine (Soybean (SO)). La classification par portée obtient deux fois plus de victoires que PEBLS et C4.5, trois à quatre fois plus que CN2 et SE. Elle est presque à égalité avec les arbres d'énumération d'ensembles avec élagage (SE 0.05). Notons également qu'elle obtient toujours une précision supérieure à celle du classifieur par défaut. Cela peut paraître insignifiant, mais c'est le seul classifieur à réussir cette performance. Tous les autres obtiennent, sur un domaine ou un autre, au moins une fois une précision inférieure au classifieur par défaut. Nous analysons plus loin le cas des bases provoquant ce comportement.

Nombre de victoires significatives. La mesure précédente peut paraître discutable car certaines différences sont petites et ne sont pas significatives. Cette seconde mesure ne fait donc le compte que des différences significatives à un seuil d'au moins 95% (exposant 1, 2, 3 et 4). Les résultats respectent les proportions précédentes. En fait, un grand nombre

Dom	cons	gen	gen_cons	PEBLS	C4.5	CN2	SE	SE 0.05	Défaut
AD	66.5±11.0	59.0±11.7[3]	61.5±8.8 [4]	71.5±9.6 [3]	70.3±10.3[5]	77.0±12.0[1]	-	-	18.0±9.7 [1]
AN	94.2±2.8	86.0±3.5 [1]	89.3±2.3 [1]	99.2±1.0 [1]	93.1±1.7 [6]	85.7±3.6 [1]	-	-	76.2±3.9 [1]
CE	86.2±5.1	-	-	83.0±5.1 [3]	83.6±5.0 [2]	82.6±4.6 [1]	80.2±28.6[6]	89.0±5.0 [3]	55.5±4.7 [1]
DI	74.4±5.2	-	-	73.7±4.4 [6]	73.1±2.5 [6]	72.1±3.8 [5]	73.5±1.9 [6]	76.5±7.9 [5]	65.0±3.8 [1]
DIAG	37.9±9.6	36.4±10.4[5]	36.4±10.4[5]	28.0±8.4 [1]	33.7±11.2[3]	34.4±12.5[4]	-	-	36.4±10.4[5]
DICO2	84.1±1.4	2.8±1.2 [1]	87.9±1.5 [1]	92.3±1.3 [1]	82.3±1.0 [1]	86.0±1.5 [1]	-	-	13.7±1.3 [1]
DICO3	83.8±0.6	-	-	85.9±0.6 [1]	73.5±0.9 [1]	78.9±0.5 [1]	-	-	16.5±0.6 [1]
DICO4	82.5±0.5	-	-	87.6±0.6 [1]	78.0±0.7 [1]	84.2±0.6 [1]	-	-	13.2±0.4 [1]
EC	67.8±14.5	67.0±14.6[6]	70.8±15.8[5]	63.3±15.4[6]	67.9±12.3[6]	70.1±10.0[6]	67.8±11.5[6]	68.5±13.4[6]	67.0±14.6[6]
GL	73.8±12.8	24.7±11.8[1]	35.0±15.9[1]	68.1±9.6 [4]	62.9±6.2 [3]	64.0±5.9 [4]	69.9±8.5 [6]	58.2±15.0[1]	31.1±14.8[1]
HDc	84.1±8.0	53.9±6.6 [1]	68.8±9.7 [1]	81.9±6.6 [6]	75.5±4.0 [1]	77.8±6.0 [3]	83.4±6.7 [6]	88.8±7.0 [3]	53.9±6.6 [1]
HDh	83.2±7.4	64.8±10.8[1]	80.8±9.4 [5]	75.9±7.7 [1]	77.6±4.1 [1]	78.1±7.6 [3]	80.4±10.8[6]	87.1±7.1 [3]	64.1±10.5[1]
HDs	93.4±3.4	93.4±3.4 [6]	93.4±3.4 [6]	90.8±4.8 [4]	90.9±2.0 [5]	92.6±2.5 [6]	93.4±3.4 [6]	82.1±19.0[4]	93.4±3.4 [2]
HDv	75.8±11.9	74.3±11.1[5]	74.3±12.1[4]	63.8±8.4 [1]	75.1±12.9[6]	74.8±11.1[6]	75.3±12.6[6]	79.0±13.1[6]	74.3±11.1[5]
HE	79.9±13.9	79.3±14.2[6]	78.7±15.1[6]	81.8±10.1[6]	81.6±10.1[6]	81.2±11.1[6]	-	-	79.3±14.2[6]
HO	82.8±5.5	75.8±6.2 [3]	75.8±7.4 [3]	81.6±5.0 [6]	83.8±4.8 [6]	83.4±5.5 [6]	-	-	63.0±7.6 [1]
IR	94.6±6.1	86.0±15.5[4]	75.3±15.0[1]	95.1±5.5 [6]	95.7±5.5 [6]	94.0±5.8 [6]	96.0±5.6 [6]	90.5±7.2 [3]	21.3±5.2 [1]
LA	85.6±14.4	64.3±29.4[4]	76.6±25.3[6]	85.2±15.1[6]	78.4±10.7[5]	78.6±11.4[6]	89.0±20.4[6]	83.0±33.3[6]	64.3±29.4[4]
LC	53.3±19.3	41.6±17.5[5]	38.3±15.3[4]	44.8±21.0[6]	57.7±29.9[6]	28.3±22.2[1]	-	-	41.6±17.5[5]
LD	79.0±8.1	70.0±10.5[2]	76.6±7.0 [6]	72.0±6.3 [1]	74.7±5.5 [2]	78.1±5.9 [6]	78.4±8.5 [6]	81.4±7.7 [5]	74.6±8.8 [3]
LE	70.0±28.1	81.6±25.3[5]	75.0±28.5[6]	72.0±31.0[6]	87.7±19.0[4]	70.0±28.1[6]	70.0±28.1[6]	93.3±21.0[3]	63.3±26.9[6]
LI	65.0±13.5	60.0±15.6[4]	64.0±13.4[6]	51.3±11.0[1]	55.7±8.4 [2]	63.0±14.1[6]	67.0±23.2[6]	60.3±17.3[6]	18.0±13.9[1]
MAL	66.9±8.4	55.7±9.2 [3]	55.7±9.2 [3]	69.4±2.8 [6]	68.8±3.3 [6]	65.0±9.0 [6]	-	-	55.8±9.2 [3]
PO	71.1±11.9	71.1±11.9[6]	64.4±15.5[4]	56.5±19.7[1]	67.9±11.0[4]	63.3±14.8[4]	63.7±14.1[1]	26.6±14.0[1]	71.1±11.9[1]
PR	81.9±12.6	74.3±19.4[4]	77.3±13.1[6]	85.5±5.7 [6]	83.8±7.7 [6]	66.2±20.7[3]	-	-	37.8±4.6 [1]
SFc	88.8±3.9	88.8±3.9 [6]	87.6±4.7 [5]	83.3±5.6 [1]	87.8±3.9 [3]	88.8±3.9 [6]	86.3±4.9 [4]	88.6±8.3 [6]	88.8±3.9 [6]
SFm	90.0±6.2	90.0±6.2 [6]	88.1±7.4 [1]	83.4±6.3 [1]	88.3±6.0 [3]	89.1±6.8 [6]	89.1±6.6 [6]	92.2±6.1 [5]	90.0±6.2 [6]
SFx	97.8±1.4	97.8±1.4 [6]	95.3±3.0 [3]	96.2±2.3 [4]	97.8±1.4 [1]	97.5±1.2 [6]	97.4±1.3 [6]	47.1±35.6[1]	97.8±1.4 [1]
SO	100.0±0.0	89.5±14.6[3]	89.5±14.6[3]	100.0±0.0 [6]	96.8±7.4 [6]	97.5±7.9 [6]	100.0±0.0 [6]	100.0±0.0 [6]	37.5±26.3[1]
SP	95.3±1.5	-	-	93.9±0.7 [1]	93.7±1.3 [1]	91.2±1.9 [1]	-	-	51.8±2.3 [1]
VO	94.7±3.4	94.4±1.9 [6]	88.2±5.9 [1]	94.3±3.1 [6]	95.1±2.6 [6]	94.4±3.6 [6]	-	-	61.3±6.9 [1]
WI	97.2±5.3	39.8±10.0[1]	39.8±10.0[1]	97.4±3.0 [6]	94.9±3.7 [6]	92.1±4.7 [4]	98.8±3.7 [6]	96.4±4.1 [6]	39.8±10.0[1]
ZO	94.0±9.6	94.0±5.1 [6]	90.0±12.4[3]	95.6±7.5 [6]	88.6±13.9[3]	90.0±15.6[5]	95.0±9.7 [6]	95.8±6.8 [6]	40.4±16.0[1]

TAB. 9.2 – *Précisions moyennes et écarts-types. L'exposant indique le seuil de confiance dans l'hypothèse « la différence observée entre la précision de chaque algorithme et la classification par portée sans généralisation des exemples en règles (cons) est significative ». Le cas où la complexité de SE-Learn ou de la classification par portée avec généralisation des exemples en règles les rendent trop longs (de l'ordre de dix jours) à mettre en œuvre sont signalés par des tirets (-).*

de différences sont significatives donc les scores ne s'annulent pas. En outre, les différences non significatives se produisent dans les deux sens (victoire de cons ou de l'autre classifieur) donc les rapports sont conservés.

Test de Wilcoxon. Nous avons évalué à l'aide du test T de Wilcoxon si les différences mesurées sur l'ensemble des domaines considérés entre la classification par portée à partir d'exemples seulement et chaque autre classifieur est significative. Le principe de ce test [Doly *et al.*, 1994] est d'ordonner les différences de précision (pcc(cons) - pcc(autre classifieur)) non nulles par valeur absolue croissante, puis de calculer la somme S^+ des rangs des différences positives et la somme S^- des rangs des différences négatives. L'hypothèse considérée est que la différence entre la classification par portée sans généralisation des exemples en règles (cons) et les autres algorithmes est strictement positive. Le test montre que :

- La différence entre cons et PEBLS est significative au seuil de 96,5%.
- La différence entre cons et C4.5 est significative au seuil de 99,5%.
- La différence entre cons et CN2 est significative au seuil de 99,9%.
- La différence entre cons et SE-Learn n'est pas significative. Le seuil est en effet de 80%.
- La différence entre cons et SE-Learn avec élagage n'est pas significative. Le seuil est en effet de moins de 80%.
- La différence entre cons et le classifieur par défaut est significative avec un seuil supérieur à 99,9%.

Le test de Wilcoxon montre donc que la classification par portée à partir d'exemples seulement offre une précision supérieure à celles de C4.5, de CN2 et, bien sûr, du classifieur par défaut avec un seuil de confiance de plus de 99%, et supérieure à celle de PEBLS avec un seuil de confiance de plus de 95%. En revanche, la différence avec SE et SE 0.05 n'est pas significative.

Moyenne. Cette mesure est la moyenne des précisions sur tous les domaines. La validité de cette mesure peut être discutée, mais elle est souvent utilisée [Quinlan, 1993b; Clark et Boswell, 1991; Domingos, 1996] et elle apporte un point de vue différent. Nous constatons à nouveau que la classification par portée a une meilleure précision moyenne que PEBLS, C4.5, CN2 et le classifieur par défaut. Cette précision moyenne est néanmoins inférieure à celle obtenue par SE et supérieure à celle de SE 0.05. Cette mesure semble clairement moyenner les résultats. Les écarts entre les classifieurs sont moins nets. Les meilleurs classifieurs ont tous une précision moyenne de l'ordre de 80%.

Rang Cette mesure est la moyenne des rangs de chaque classifieur par rapport aux autres. Pour chaque problème de classification, les classifieurs sont notés de 0 à n de la plus mauvaise précision à la meilleure (n + 1-ième). Comme le nombre de classifieurs utilisables dans chaque domaine n'est pas le même, la valeur associée à chaque classifieur est normalisée en divisant par n. Les précisions de chacun pour un domaine donné sont donc notées de 0 à 1, de la plus mauvaise à la meilleure. La moyenne des rangs obtenus est calculée sur l'ensemble des domaines où un classifieur a été évalué. La classification par portée à partir d'exemples seulement a un rang nettement supérieur à tous les autres classifieurs, c'est-à-dire le rang moyen de notre algorithme par rapport aux autres classifieurs est nettement supérieur aux rangs des autres.

Les cinq mesures considérées montrent nettement que la classification par portée à partir d'exemples seulement offre une meilleure précision que les autres classifieurs utilisés. La différence par rapport à C4.5 et CN2 est très nette, alors qu'elle est moins marquée avec SE et SE 0.05. Cela confirme deux points essentiels de notre approche. D'une part, la classification par portée est bien « équivalente » aux approches utilisant toutes les règles. D'autre part, les approches à base de toutes les règles offrent une meilleure précision que les approches à base de règles « choisies » (CN2, C4.5), et que les approches à base d'instances à nombre de voisins constant (PEBLS).

La classification par portée avec généralisation des exemples en règles : gen et gen_cons

La classification par portée avec généralisation des exemples en règles telle que nous l'avons mise en œuvre offre des précisions inférieures à celle des autres classifieurs, mais reste très supérieure à celle du classifieur par défaut. Les tableaux suivants présentent les résultats des mesures en nombre de victoires, précision moyenne et rang moyen des deux stratégies de généralisation, avec (gen) ou sans (gen_cons) recouvrement des règles. L'utilisation de mesures plus précises telles que le nombre de victoires significative ou le test de Wilcoxon n'est pas nécessaire dans ce cas car les différences sont suffisamment visibles.

gen									
Mesure	cons	gen	gen_cons	PEBLS	C4.5	CN2	SE	SE 0.05	Défaut
Nb. Vict.	-	-	-	12-16	7-20	10-16	5-13	4-15	13-3
Moyenne	81,1	71,1	75,5	78,9	79,3	77,9	82,8	79,3	53,6
Rang	0,76	0,39	0,41	0,55	0,56	0,50	0,67	0,69	0,2

gen_cons									
Mesure	cons	gen	gen_cons	PEBLS	C4.5	CN2	SE	SE 0.05	Défaut
Nb. Vict.	-	-	-	11-17	8-20	11-16	5-12	5-13	19-5
Moyenne	81,1	71,1	75,5	78,9	79,3	77,9	82,8	79,3	53,6
Rang	0,76	0,39	0,41	0,55	0,56	0,50	0,67	0,69	0,2

Les performances de la classification par portée avec généralisation des exemples en règles peuvent s'expliquer par une utilisation non optimale des règles générées. En effet, la généralisation des exemples en règles s'effectue à partir des voisins calculés à l'aide des paramètres de cohérence ϵ et de spécificité M évalués sur l'ensemble d'apprentissage initial. Une fois les règles générées, la classification par portée s'effectue à partir d'un nouvel ensemble d'apprentissage composé des règles engendrées et des exemples restant non couverts par ces règles. Ce nouvel ensemble ne présentant pas les mêmes caractéristiques (couverture, nombre d'exemples, dispersion) que l'ensemble d'apprentissage initial, nous n'avons pas utilisé les paramètres ϵ et M initiaux. Il faudrait évaluer ces paramètres à nouveau, sur ce nouvel ensemble d'apprentissage avant de procéder à la classification par portée généralisée aux règles. Nous ne l'avons pas fait et nous avons simplement utilisé les valeurs $\epsilon = 0$ et $M = 1$ correspondant à une situation pure d'un point de vue logique où une règle n'admet pas de contre-exemple et un exemple doit être inclus dans l'hyper-rectangle représenté par la partie gauche de la règle suivant tous les attributs pour en être un contre-exemple. Nous n'avons pas procédé au second réglage pour des contraintes de temps. L'algorithme de réglage que nous proposons est un premier pas vers un réglage automatique des paramètres et il est trop coûteux en temps. Nous l'avons appliqué à la base de données initiale car le réglage n'est fait qu'une fois pour les dix tests de la validation croisée. Lorsque la généralisation des exemples en règles est utilisée, l'ensemble d'apprentissage

généré change à chaque test et le réglage aurait dû être effectué dix fois. Par ailleurs, l'ensemble des règles générées est parfois bien plus grand que la base initiale et le temps de calcul s'en trouve augmenté. Un test de grande ampleur comme celui que nous avons effectué nécessiterait plusieurs mois de calculs sur des machines rapides telles que des SUN Ultra 1. Les résultats présentés dans ces pages représentent en effet plus de 10 jours de CPU sur plusieurs stations de ce type.

Cas des bases où la classe majoritaire l'emporte

Certaines bases comportent une répartition inégale des exemples en fonction des classes. Cela se reflète clairement dans la précision du classifieur par défaut. Celui-ci indique, en effet, la proportion d'exemples de la classe majoritairement représentée dans la base de données. Plus la précision du classifieur par défaut est élevée par rapport à 100% divisé par le nombre de classes -ce qui correspond à une répartition homogène des exemples dans les classes-, plus la répartition est déséquilibrée. Ces cas sont des problèmes de classification souvent difficiles, car s'il est assez aisé de « retrouver » la classe des exemples de la classe majoritaire, il est difficile de prédire celles des exemples restants et il est parfois difficile de faire mieux que le classifieur par défaut. Un exemple typique est celui de la base Hepatitis (HE) comportant deux classes, mais pour laquelle la précision du classifieur par défaut est de plus de 79%. Dans ce cas, la précision des autres classifieurs est à peine supérieure à cette valeur. Il arrive néanmoins que les classifieurs fassent mieux que le classifieur par défaut même en présence de classes inégalement représentées. C'est le cas du domaine Annealing (AN) qui comporte 6 classes et pour lequel les meilleures précisions atteignent 99,2% alors que celle du classifieur par défaut est de 76,2%. Malheureusement, de nombreux domaines conduisent à des contre-performances des classifieurs. Les domaines Heart-Disease-Switzerland (HDs), Liver-Disease (LD), Post-Operative (PO), Solar-Flare (SFc,SFm,SFx) ou Diagnostic (DIAG) sont des problèmes pour lesquels de nombreux classifieurs « évolués » obtiennent des précisions inférieures, parfois nettement inférieures, à celle du classifieur par défaut. Comme nous l'avons indiqué, le seul classifieur dont la précision n'est jamais inférieure à celle du classifieur par défaut est la classification par portée à partir d'exemples seulement (cons). Il est vraisemblable, en effet, que dans le cas de classes inégalement représentées, l'ensemble des voisins de l'objet à classer comporte une majorité d'exemples de la classe majoritaire et conduisent à ce résultat. Toutefois, ce comportement n'est pas garanti. En effet, l'intérêt des classifieurs « évolués » est qu'ils prennent des risques et prédisent des classes différentes de la classe majoritaire. Ces risques sont bien entendu « calculés » puisque les prédictions sont basées sur l'hypothèse des similarités et sur la représentativité des exemples. En particulier, un problème de classification où un nouvel objet ne peut pas être classé par similarité avec les exemples connus est actuellement insoluble. Il est clair que les bases sur lesquelles les classifieurs « évolués » obtiennent une précision inférieure à celle du classifieur par défaut ne satisfont pas les biais inductifs de ces classifieurs.

9.2.3 Comparaison des temps d'exécution

Les temps d'exécution mesurés sont bien sûr les temps d'utilisation effective du microprocesseur. Ils sont donc indépendants de la charge de la machine. Nous avons effectué toute la procédure de comparaison entre les classifieurs sur une même machine pour un domaine fixé. En revanche, nous avons utilisé près de quinze machines différentes pour ces tests et leurs rapidités de calcul sont distinctes, même pour deux modèles identiques. En conséquence, les temps

de calcul entre deux domaines ne doivent pas être comparés.

Le temps d'exécution moyen nécessaire à la classification d'un dixième de la base en fonction des neuf dixièmes restants pour chacun des domaines est donné dans le tableau 9.3, ainsi que l'écart-type mesuré sur les dix tests de la validation croisée. Le temps de classification du classifieur par défaut n'a pas été évalué, mais il est évidemment bien plus rapide que tous les autres classifieurs.

Étant donné un ensemble de données, le calcul des paramètres de cohérence ϵ et de généralité M a été effectué une seule fois pour les trois variantes de la classification par portée (cons, gen et gen_cons) et indépendamment du processus de validation croisée, c'est-à-dire que les valeurs de ϵ et M ont été déterminées dans un premier temps, puis les valeurs obtenues ont servi aux dix tests de la validation croisée, pour les trois configurations du programme. Le réglage des paramètres de modulation de la cohérence diffère de l'apprentissage effectué par les autres classifieurs : il est fait une seule fois alors que les autres classifieurs effectuent un apprentissage pour chacun des dix ensembles d'apprentissage lors de la validation croisée. Il faut donc comparer les classifieurs suivant le temps d'exécution total, c'est-à-dire :

- en ce qui concerne la classification par portée, le temps de réglage des paramètres, puis le temps de résolution (classification puisque la classification par portée ne nécessite pas d'apprentissage) des dix problèmes de classification fournis par la validation croisée ;
- en ce qui concerne les autres classifieurs, le temps de résolution (apprentissage et classification) des dix problèmes de classification fournis par la validation croisée.

Nous avons ramené ce temps total à un temps moyen de « résolution » d'un seul problème de classification d'un dixième de l'ensemble de données par rapport aux neuf dixièmes restants. Cela n'a pas pour but de diviser par dix le temps de réglage des paramètres ϵ et M, mais permet simplement de mentionner de façon cohérente l'écart-type du temps de résolution d'un problème lors des dix tests.

Nous comparons tout d'abord la classification par portée à partir d'exemples seulement aux autres classifieurs. Nous évaluons ensuite les deux variantes de la classification par portée avec généralisation des exemples en règles.

La classification par portée à partir d'exemples seulement : cons

Afin de distinguer plus nettement les performances de la classification par portée à partir d'exemples seulement, nous avons eu recours à diverses mesures données dans le tableau 9.3. Les seules mesures considérées sont le nombre de victoires et le rang. Leurs résultats sont les suivants :

Mesure	cons	gen	gen_cons	PEBLS	C4.5	CN2	SE	SE 0.05
Nb. Vict.	-	-	-	18-15	16-17	12-21	11-9	6-14
Rang	0,59	0,07	0,15	0,58	0,59	0,74	0,53	0,75

La différence des temps d'exécution est suffisamment nette pour que les mesures du nombre de victoires significatives et le test de Wilcoxon ne soient pas nécessaires. La mesure de la moyenne sur l'ensemble des bases n'a pas été considérée car les temps d'exécution varient énormément d'un problème à l'autre ; le résultat aurait donc peu de sens. Les informations des deux mesures considérées sont donc :

Nombre de victoires. CN2 et SE-Learn avec élagage obtiennent deux fois plus souvent un temps d'exécution inférieur à celui de la classification par portée que les cas où l'inverse

Dom	cons	gen	gen_cons	PEBLS	C4.5	CN2	SE	SE 0.05
AD	3.32±0.06	19.93±1.96	21.98±0.66	13.73±0.01	9.67±0.73	22.59±1.61	-	-
AN	1:21±0.47	14:21±44.19	7:66±9.81	53.91±0.26	29.53±2.78	16.63±0.86	-	-
CE	36.99±1.01	-	-	22.82±0.05	22.78±2.34	6.01±0.25	23.93±1.57	8.28±0.84
DI	20.63±0.32	-	-	21.22±0.04	1:20±4.96	5.91±0.68	1:16±1.61	3.66±0.34
DIAG	19.57±0.77	13:14±56.62	12:01±31.29	22.27±0.21	1:13±7.02	1:04±1.72	-	-
DICO2	49:50±54.89	8h56±10:54	6h15±7:25	16:55±8:30	6h17±18:03	3:04±5.26	-	-
DICO3	5h47±7:25	-	-	39:24±16:42	25h40±1h04	16:33±17.97	-	-
DICO4	22h30±42:38	-	-	1h24±1:19	2j. 21h13±2h34	50:43±1:12	-	-
EC	0.47±0.02	1:57±1:40	13.00±0.49	0.85±0.01	3.46±0.53	0.80±0.06	0.56±0.02	0.37±0.02
GL	2.04±0.11	9:58±46.67	61.44±14.41	2.16±0.06	14.58±2.14	2.08±0.14	3.83±0.26	0.68±0.05
HDc	10.57±0.11	5h58±10:44	32:19±3:04	4.78±0.04	15.07±1.46	2.08±0.12	59.62±2.87	1.49±0.35
HDh	13.05±0.19	2h34±10:31	25:14±2:49	3.91±0.02	11.52±1.03	2.16±0.17	15.06±0.63	1.90±0.22
HDs	0.46±0.01	1:32±6.57	1:02±6.33	0.66±0.01	0.18±0.02	0.38±0.04	0.87±0.07	2.60±0.42
HDv	1.64±0.04	21:26±51.84	1:35±7.47	1.66±0.04	5.08±0.78	2.04±0.11	8.90±0.83	5.28±0.83
HE	1.62±0.07	18:29±59.02	11:53±1:26	1.85±0.01	2.52±0.25	2.04±0.16	-	-
HO	5.50±0.14	1h03±17:42	11:01±1:04	5.81±0.03	7.76±1.05	11.46±0.97	-	-
IR	0.42±0.01	7.07±1.04	10.76±0.70	0.87±0.01	0.37±0.08	0.31±0.05	0.21±0.01	0.19±0.01
LA	7.57±0.01	9.97±0.23	8.76±0.16	0.54±0.01	0.38±0.11	0.77±0.08	2.24±0.21	3.28±0.28
LC	9.12±0.03	12.44±0.07	11.67±0.11	0.74±0.02	0.63±0.06	1.96±0.17	-	-
LD	1.66±0.05	22:04±2:37	14:45±2:03	3.53±0.05	18.84±1.46	1.27±0.07	1.26±0.05	0.48±0.01
LE	0.13±0.01	0.15±0.01	0.17±0.01	0.34±0.01	0.07±0.01	0.04±0.01	0.11±0.01	0.10±0.01
LI	4.93±0.01	5.85±0.02	5.84±0.02	0.94±0.01	0.79±0.04	0.67±0.05	0.46±0.03	0.34±0.01
MAL	1:27±0.81	1h17±5:46	1h07±6:07	1:11±1.28	1:19±5.33	49.55±3.57	-	-
PO	3.30±0.01	5.44±0.27	4.71±0.07	0.63±0.01	0.65±0.06	1.93±0.16	0.59±0.02	0.71±0.05
PR	2.29±0.02	39.72±2.15	19.73±1.07	3.27±0.01	3.08±0.67	2.69±0.16	-	-
SFc	1.74±0.11	9.58±0.66	12.89±3.57	5.37±1.70	2.70±1.15	2.91±0.92	1.80±0.64	2.61±0.60
SFm	1.71±0.06	8.50±0.38	11.04±0.40	4.47±0.06	1.74±0.86	2.34±0.19	1.42±0.05	1.84±0.11
SFx	1.41±0.05	5.26±0.13	8.11±0.33	4.52±0.02	0.22±0.02	1.63±0.11	0.65±0.04	2.41±0.34
SO	17.86±0.02	18.44±0.02	18.04±0.04	0.94±0.02	0.55±0.04	0.75±0.12	21.57±2.31	2.64±0.26
SP	4h21±11.51	-	-	4:12±9.26	3:60±10.30	2:15±3.75	-	-
VO	6.66±0.27	50.64±4.09	1:26±3.89	12.31±0.21	1.69±0.44	1.65±0.12	-	-
WI	16.82±0.41	1h17±1:08	39:36±1:27	4.26±0.04	6.42±1.13	2.14±0.24	54.06±14.52	5.57±0.54
ZO	0.38±0.01	0.91±0.04	0.99±0.05	0.79±0.01	0.34±0.03	0.40±0.02	5.02±0.32	0.28±0.02

TAB. 9.3 – *Temps d'exécution (en jours. heures h minutes:secondes.centièmes). Le cas où la complexité de SE-Learn ou de la classification par portée avec généralisation des exemples en règles les rendent trop longs (de l'ordre de dix jours à mettre en œuvre sont signalés par des tirets (-).*

se produit. Suivant la mesure du nombre de victoires, la classification par portée à partir d'exemples seulement a un temps d'exécution semblable à ceux de PEBLS, C4.5 et SE-Learn sans élagage.

Rang. Le rang moyen des classifieurs confirme les tendances précédentes. La classification par portée à partir d'exemples seulement est en moyenne moins bien classée que CN2 et SE-Learn avec élagage qui sont presque *ex-æquo*. Elle a le même rang moyen que C4.5 et est légèrement mieux classée que PEBLS et SE-Learn sans élagage.

Nous rappelons que ces résultats ne sont significatifs que du temps global comme expliqué plus haut. Le temps global tient compte du temps de réglage des paramètres de cohérence ϵ et de généralité M. Si nous parvenons à améliorer la procédure de réglage que nous avons proposé, ce qui est l'objet de travaux à venir, nous pourrons réduire le temps global.

La classification par portée avec généralisation des exemples en règles : gen et gen_cons

La classification par portée avec généralisation des exemples en règles présente des temps d'exécution souvent plus longs que ceux des autres classifieurs. Néanmoins, elle n'est pas toujours la plus lente. Sur des problèmes de petites tailles, où le nombre de règles engendrées est de l'ordre de la taille de l'ensemble d'apprentissage initial, comme Contact Lenses (LE) ou LED (LI), la classification par portée avec généralisation des exemples en règles est plus rapide que PEBLS, C4.5, CN2, SE et SE 0.05.

gen								
Mesure	cons	gen	gen_cons	PEBLS	C4.5	CN2	SE	SE 0.05
Nb. Vict.	-	-	-	1-27	0-28	1-27	1-17	0-18
Rang	0,59	0,07	0,15	0,58	0,59	0,74	0,53	0,75
gen_cons								
Mesure	cons	gen	gen_cons	PEBLS	C4.5	CN2	SE	SE 0.05
Nb. Vict.	-	-	-	1-27	1-27	1-27	1-17	0-18
Rang	0,59	0,07	0,15	0,58	0,59	0,74	0,53	0,75

9.2.4 Les paramètres de la classification par portée

Ce paragraphe est consacré aux paramètres de la classification par portée. Nous considérons dans un premier temps les paramètres de réglage. Puis nous évaluons l'influence de la prise en compte des redondances et de la généralisation des exemples en règles sur la taille de l'ensemble d'apprentissage.

Réglage de la cohérence et du critère de résolution

Le temps nécessaire au choix du critère de résolution et au réglage de ϵ et M ainsi que les valeurs de ces paramètres sont regroupés dans le tableau 9.4.

Le temps de réglage des paramètres est nettement supérieur au temps d'exécution de la classification à partir d'exemples seulement figurant dans le tableau 9.3. Cette procédure demande à être améliorer. Certaines valeurs de ϵ dépassent les 40%. Il est peu probable qu'une règle admette 40% de contre-exemples dans l'ensemble d'apprentissage. Soient deux valeurs de ϵ, notée ϵ_1 et ϵ_2, et M_1 (respectivement M_2) la valeur de M pour laquelle la valeur de la qualité

Dom	Temps	CR	ϵ	M
AD	22.30	C	0	2
AN	11:51	C	0	2
CE	2:27	C	0	2
DI	1:27	S	0	3
DIAG	1:12	C	0	9
DICO2	2h12	C	50	2
DICO3	12h04	C	10	2
DICO4	6h42	C	10	2
EC	0.71	S	10	2
GL	4.12	S	10	6
HDc	30.33	S	20	7
HDh	23.04	C	10	4
HDs	2.59	S	0	1
HDv	7.00	S	0	1
HE	5.46	C	0	3
HO	34.82	C	0	3
IR	1.03	S	0	2

Dom	Temps	CR	ϵ	M
LA	1:15	C	0	2
LC	1:29	S	40	16
LD	4.29	S	0	2
LE	1.03	C	10	1
LI	48.64	C	0	1
MAL	12:02	C	0	12
PO	31.82	S	10	1
PR	17.75	C	0	11
SFc	7.80	S	10	1
SFm	8.08	S	10	1
SFx	7.73	S	10	1
SO	2:57	S	0	2
SP	41h49	C	0	9
VO	31.86	S	0	1
WI	3.63	S	40	11
ZO	2.95	C	10	4

TAB. 9.4 – *Temps de réglage des paramètres et les valeurs « optimales » obtenues : critère de résolution (CR) où S désigne le vote majoritaire simple et C le vote majoritaire quadratique, ϵ en pourcentage et M.*

$v(\epsilon_1, M_1)$ (resp. $v(\epsilon_2, M_2)$) est maximale (cf. paragraphe 7.2.3). Lorsque ϵ_1 et ϵ_2 sont proches, M_1 et M_2 le sont également, de même que les qualités $v(\epsilon_1, M_1)$ et $v(\epsilon_2, M_2)$. En fait, nos expérimentations ont montré qu'il arrive assez souvent que plusieurs couples (ϵ, M) offrent des valeurs élevées. Choisir un de ces couples de plus grande valeur, mais dont ϵ est plus petit que 40% permettrait peut-être d'obtenir une meilleure précision. C'est l'une de nos perspectives de travail.

Prise en compte de la redondance et généralisation des exemples en règles

Le tableau 9.5 indique, dans une première partie, la taille moyenne d'un ensemble d'apprentissage EA (neuf dixièmes de la base complète E) et le nombre d'exemples (moyenne et écart-type) réellement utilisés par la classification par portée après la prise en compte des redondances. Les deux autres tiers du tableau correspondent à la taille des ensembles d'apprentissage obtenus après la généralisation des exemples en règles, ainsi que le nombre de règles qu'ils contiennent. La différence entre ces deux valeurs correspond donc au nombre d'exemples initiaux non couverts par les règles. Les résultats de la génération de toutes les règles (gen) ou des règles n'introduisant pas d'incohérence (gen_cons) sont mentionnés avec leur valeur moyenne et leur écart-type.

La prise en compte des redondances permet de réduire parfois grandement le nombre d'exemples considérés. C'est le cas par exemple des bases de données (DICO2, DICO3, DICO4) servant à la reconnaissance de la parole qui comportent un grand nombre de répétitions. La prise en compte des redondances permet de réduire le nombre d'exemples considérés et donc d'accélérer le calcul.

On pourrait attendre de la généralisation des exemples en règles, d'une part d'améliorer la

Dom	9/10 $\|E\|$	$\|EA\|$ réduit	gen	nb règles	gen_cons	nb règles
AD	180	161,2±1,6	267,8±13,4	250,3±14,1	194,3±5,4	153,6±4,4
AN	718,2	711,7±0,9	3629,7±44,9	3626,4±45	1421,2±24,3	1171,4±26,7
CE	621	621±0	-	-	-	-
DI	691,2	691,2±0,4	-	-	-	-
DIAG	178,2	178,2±0,4	2078,5±72,5	2076,1±72,7	1832,2±71,9	1828,8±72,1
DICO2	7370,1	1809±11,86	29±0	29±0	1809±11,86	0±0
DICO3	12718,8	5138.0±15.2	-	-	-	-
DICO4	27547,2	9940,2±35,6	-	-	-	-
EC	117,9	117,9±0,3	1938,9±43,9	1938,9±43,9	255,1±49,2	186,9±50
GL	192,6	191,8±0,6	2803±87,7	2802,9±87,6	805,8±33,6	784,2±35,3
HDc	271,8	271,8±0,4	12506±173,3	12506±173,3	4377,8±169	4353,8±170,8
HDh	263,7	262,9±0,7	6721,1±196,2	6721,1±196,2	3732,2±167,3	3687,9±169,1
HDs	109,8	109,8±0,4	2393,3±94,2	2393,2±94,1	1517,4±118,9	1509,6±118,5
HDv	179,1	178,3±0,5	4470,5±116,3	4470,5±116,3	390,8±81,7	304,7±90,4
HE	139,5	139,5±0,5	4544±176,3	4544±176,3	3573,2±249,6	3567,4±249,7
HO	331,2	320,6±0,9	10291,3±256,5	10291,3±256,5	3441,9±196,6	3410±197,6
IR	135	132,5±0,5	359,8±18,1	359,1±18,3	342,7±16,1	341,2±15,9
LA	51,3	51,3±0,5	374±23,3	374±23,3	223±27,1	219,9±27,8
LC	29,7	28,8±0,4	127,9±3,1	127,9±3,1	31,5±3	21,8±5,7
LD	309,6	306,3±0,8	5797,6±254,8	5794,9±254,3	4750,1±325,8	4725,6±327,9
LE	21,6	21,6±0,5	7±0,6	7±0,6	11,7±0,8	1,3±0,5
LI	90	34,9±1,6	35,5±2,4	22,6±2,1	18,1±1,4	12,9±1,4
MAL	319,5	319,5±0,5	4386,6±175,4	4386,5±175,3	3937±192,3	3935,7±192,8
PO	81	67,7±1,1	212,2±11,5	210,4±11,5	67±1,3	6,9±3,4
PR	95,4	95,4±0,5	688,2±22,8	688,2±22,8	376±19,6	374,9±19,8
SFc	290,7	147,1±1,4	50,6±3,9	50,5±4	117,3±8,4	27,5±6,7
SFm	290,7	147,1±1,4	43,9±9	42,1±9	112,3±9,8	20,3±5,4
SFx	290,7	147,1±1,4	11,5±2,4	11,4±2,5	46,5±21,2	40,7±18,6
SO	43,2	42,3±0,5	21,5±5,6	21,5±5,6	21,5±5,6	21,5±5,6
SP	2871	2715,9±2,9	-	-	-	-
VO	391,5	311,4±3,7	704,8±69,1	704,8±69,1	586,4±43,9	536±44,9
WI	160,2	160,2±0,4	4306,8±42,2	4306,8±42,2	3006±54,9	3006±54,9
ZO	90,9	54,8±1,2	25,9±3,1	25,9±3,1	26,2±2	22,4±2,2

TAB. 9.5 – *Valeurs moyenne et écart-type de la taille de l'ensemble d'apprentissage avant et après la réduction des redondances, et après généralisation des exemples en règles. Dans ce dernier cas, le nombre de règles que comporte l'ensemble d'apprentissage est indiqué.*

précision de la classification, d'autre part, de réduire la taille de l'ensemble d'apprentissage. Ces deux attentes ne sont pas satisfaites. Nous avons proposé précédemment une première explication possible des précisions obtenues. Une explication supplémentaire des performances en précision de la classification par portée avec généralisation des exemples en règles est liée au nombre de règles. Si un problème de classification peut être traité avec une des techniques existantes, l'ensemble d'apprentissage résultant de la généralisation des exemples en règles doit vraisemblablement comporter moins de règles que d'exemples dans l'ensemble d'apprentissage initial. Les stratégies de généralisation des exemples en règles que nous proposons requièrent donc d'être améliorées pour permettre de représenter de façon plus concise et plus précise la fonction cible. Notons quand même que les stratégies actuelles produisent d'ores et déjà de bons résultats sur plusieurs bases (LE, LI, SFc, SFm, SFx et ZO).

9.3 Application à des domaines réels de la classification par portée

La plupart des domaines considérés dans l'évaluation précédente sont des bases de test usuelles en apprentissage automatique. Nous présentons dans ce paragraphe les domaines réels qui ne proviennent pas du répertoire de l'Université d'Irvine en Californie, mais de problèmes réels traités au sein de l'équipe de recherche à laquelle nous appartenons. Ces problèmes concernent l'aide au diagnostic médical et la reconnaissance de la parole. Nous présentons également quelques résultats fournis par Ron Rymon concernant l'application de la classification par portée à un problème d'aide à la décision dans le domaine bancaire.

9.3.1 Aide au diagnostic médical

Le premier problème de classification traitant d'un domaine réel que nous avons considéré concerne le registre lorrain du cancer de l'enfant. Les hôpitaux de Nancy tiennent à jour un registre répertoriant les cas de cancer survenant chez des enfants lorrains. Ce registre a pour but de mettre en évidence des facteurs de risques particuliers pouvant provoquer un cancer chez l'enfant. L'ensemble des descriptions d'enfants malades a été complétée par des descriptions de cas témoins. Un cas témoin est un enfant du même âge et du même sexe que l'enfant malade auquel il est associé. Les problèmes de classification que nous avons considérés consistent à déterminer si un enfant est malade ou non, à partir de sa description (Malades (MAL)) ou à déterminer le type de cancer d'un enfant malade à partir de la description de l'enfant (Diagnostic (DIAG)).

Diagnostic (DIAG)

Les individus sont décrits par 153 attributs représentant les maladies infantiles que l'enfant a pu avoir, ses vaccinations, ses opérations, le nombre de radiographies qu'il a subies, les produits toxiques que son père utilise dans son activité professionnelle, les traitements reçus par sa mère lors de la grossesse, etc.

Caractéristiques

$\|E\|$	: 198	Classes	: 12	Valeurs manquantes	: Oui	Incohérences	: Non
Attributs	: 153	Nominaux	: 43	Linéaires discrets	: 103	Linéaires continus	: 7

Les attributs sont de tous les types (nominaux, linéaires discrets ou continus). Le nombre d'exemples (198) est très petit par rapport au nombre de classes (12), et surtout par rapport

au nombre d'attributs.

PCC					Défaut : 36.4±10.4
cons	: 37.9±9.6	gen	: 36.4±10.4	gen_cons	: 36.4±10.4
PEBLS	: 28.0±8.4	C4.5	: 33.7±11.2	CN2	: 34.4±12.5
SE	: -	SE 0.05	: -		

Cette base est typique des bases où les classes ne sont pas représentées par des nombres d'exemples proches. L'ensemble d'exemples contient en effet plus de 36% d'exemples de leucémies sur les 12 classes possibles. La précision de chaque classifieur ne dépasse guère celle du classifieur par défaut. PEBLS, C4.5 et CN2 obtiennent même des précisions inférieures.

Temps					
cons	: 19.57±0.77	gen	: 13:14±56.62	gen_cons	: 12:01±31.29
PEBLS	: 22.27±0.21	C4.5	: 1:13±7.02	CN2	: 1:04±1.72
SE	: -	SE 0.05	: -		

Les temps d'exécution traduisent l'obtention de réponses dans un délai très court. Cela n'est pas surprenant vu le faible nombre d'exemples.

Paramètres de la classification par portée				
$\|EA\|$: 178.2±0.4	Temps : 1:12	Critère : C	Cohérence (ϵ) : 0	Généralité (M) : 9
$\|gen\|$: 2078.5±72.5 (2076.1±72.7)		$\|gen\_cons\|$: 1832.2±71.9 (1828.8±72.1)		

Les paramètres résultant du réglage de la classification par portée sont assez surprenants. En effet, il est probable que sur les 153 attributs, un grand nombre ne soient pas pertinents. Il est fréquent que M prenne des valeurs élevées lorsqu'il y a beaucoup d'attributs. Ici, la valeur retenue (9) est très faible par rapport aux 153 attributs. Il est vrai que plus la valeur de ϵ est élevée, plus M croît, et donc comme ϵ vaut ici 0%, M peut être faible. Néanmoins, cette valeur est surprenante, et les résultats complets du réglage, que nous ne présentons pas ici, montrent que M varie seulement de 1 à 38 pendant tout le réglage automatique.

Cette expérimentation montre qu'il est difficile de retrouver le diagnostic en fonction de la description telle qu'elle existe. Nous allons voir qu'il est également difficile de distinguer les enfants malades des cas témoins.

Malades (MAL)

La description des individus est la même que précédemment. L'ensemble d'exemples est seulement plus grand puisque l'on considère les enfants malades et les témoins. Le nombre de classe est réduit à 2.

Caractéristiques							
$\|E\|$	: 355	Classes	: 2	Valeurs manquantes	: Oui	Incohérences	: Non
Attributs	: 153	Nominaux	: 43	Linéaires discrets	: 103	Linéaires continus	: 7

PCC					Défaut : 55.8±9.2
cons	: 66.9±8.4	gen	: 55.7±9.2	gen_cons	: 55.7±9.2
PEBLS	: 69.4±2.8	C4.5	: 68.8±3.3	CN2	: 65.0±9.0
SE	: -	SE 0.05	: -		

La précision des divers classifieurs est cette fois meilleure que celle du classifieur par défaut. Mais elle n'est pas suffisante pour être utilisée par les médecins.

Temps		
cons : 1:27±0.81	gen : 1h17±5:46	gen_cons : 1h07±6:07
PEBLS : 1:11±1.28	C4.5 : 1:19±5.33	CN2 : 49.55±3.57
SE : -	SE 0.05 : -	

Les temps d'exécution sont là encore raisonnables. Un temps de calcul plus long serait acceptable s'il pouvait conduire à une plus grande précision.

Paramètres de la classification par portée				
$\|EA\|$: 319.5±0.5	Temps : 12:02	Critère : C	Cohérence (ϵ) : 0	Généralité (M) : 12
$\|gen\|$: 4386.6±175.4 (4386.5±175.3)		$\|gen\_cons\|$: 3937.0±192.3 (3935.7±192.8)		

La valeur de M est faible comme dans le cas précédent. Le nombre de règles provenant de la généralisation des exemples en règles est élevé. Les résultats des classifieurs sont décevants. Ils offrent une précision insuffisante.

Le registre lorrain du cancer de l'enfant a surtout pour but de déceler des facteurs de risques. Il s'agit d'aider les médecins dans leur compréhension de ces facteurs plutôt que de réaliser un système de diagnostic automatique. A ce titre, le nombre des règles générées par la classification par portée avec généralisation des exemples en règles est clairement trop important pour aider quiconque à comprendre les liens existant entre les attributs et la classe.

Nous avons également testé sur ce problème un programme de génération des arbres d'énumération d'ensembles. Nous avons développé ce programme à la suite des travaux que nous avons effectués sur le modèle de généralisation confirmatoire dans un cadre attribut-valeur (cf. chapitre 3). A la différence de SE-Learn, ce programme permet une exploration en largeur de l'arbre et s'arrête lorsque l'arbre devient trop grand pour la mémoire de la machine sur laquelle il est exécuté. Cela se produit assez rapidement puisque la croissance de cet arbre est exponentielle en le nombre d'attributs. Dans le cas présent, l'utilisation de cinq cents mégaoctets de mémoire n'a permis d'explorer l'arbre que jusqu'à une profondeur de 5 attributs, classe comprise. Cela est suffisant pour savoir qu'aucune règle comportant moins de 4 attributs en partie gauche n'est cohérente et pertinente pour un nombre suffisant d'exemples.

9.3.2 Reconnaissance de la parole

Un problème de classification provenant des thèmes de recherche de l'équipe RFIA concerne la reconnaissance automatique de la parole. De nombreux travaux de l'équipe concernent les modèles de Markov cachés [Jelinek, 1976]. Ces modèles sont en effet bien adaptés à l'apprentissage de séquences. Ce type d'apprentissage sort cependant du cadre des problèmes de classification que nous considérons. Dans ce paragraphe, nous nous intéressons à un problème de classification posé par une nouvelle approche du décodage acoustico-phonétique. Un système de reconnaissance automatique de la parole se décompose en une suite de modules permettant à partir du signal sonore de reconnaître une suite de mots. Une étape cruciale est réalisée par le module de décodage acoustico-phonétique [Haton *et al.*, 1991b]. Ce module doit déterminer la suite de phonèmes, unités de sons élémentaires, prononcés à partir de la donnée du signal sonore. Cette suite de phonèmes est ensuite traitée par d'autres modules afin de construire une suite de mots. Une nouvelle approche du décodage acoustico-phonétique consiste à procéder en fait à plusieurs décodages acoustico-phonétiques et à combiner leurs réponses. Le signal de

la parole est divisé suivant plusieurs bandes de fréquences et une reconnaissance des phonèmes est appliquée pour chaque bande de fréquences. L'ensemble est schématisé sur la figure 9.1.

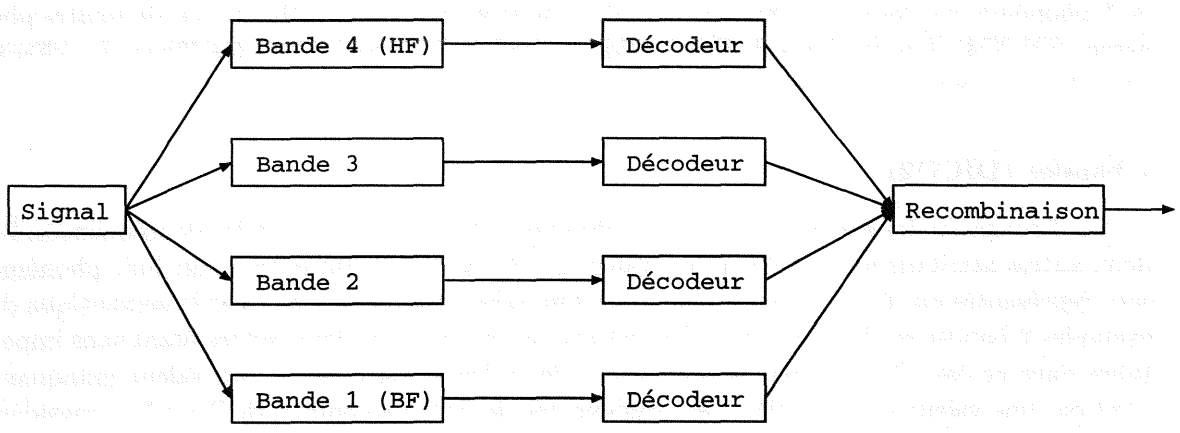


FIG. 9.1 – *Principe du décodage acoustico-phonétique sur plusieurs bandes de fréquences.*

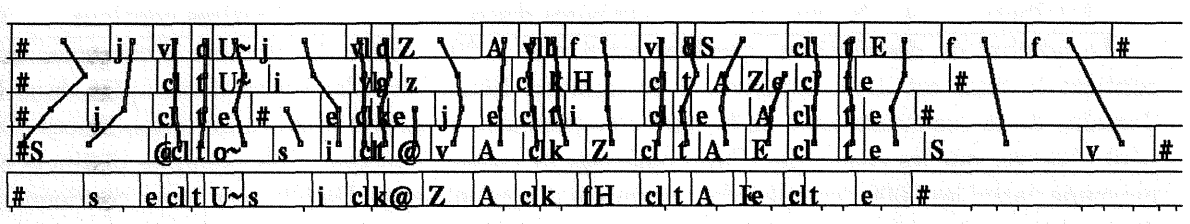


FIG. 9.2 – *Séquences de phonèmes sur les quatre bandes de fréquences et la séquence finale associée. Les doubles traits reliant plusieurs segments des quatre séquences représentent les regroupements effectués entre les phonèmes.*

Chaque décodeur fournit une séquence de phonèmes. Il reste alors à combiner les quatre séquences en une seule séquence. La figure 9.2 présente un exemple de quatre séquences résultant du décodage effectué sur chaque bande de fréquences. La recombinaison consiste à regrouper les phonèmes par groupes de 2 à 4 entre les quatre bandes de fréquence et à associer un phonème unique à l'association des deux, trois ou quatre phonèmes présents sur les quatre bandes. Les regroupements de deux, trois ou quatre phonèmes proviennent d'un traitement effectué par Christophe Cerisara qui travaille sur ce problème de décodage acoustico-phonétique [Cerisara *et al.*, 1997]. Il est évident que la manière de regrouper les phonèmes est une étape clé de l'approche. Christophe Cerisara construit les regroupements de la façon suivante : il construit tout d'abord l'ensemble des suites de regroupements couvrant toutes les séquences de phonèmes. Le nombre de combinaisons est bien sûr très important, mais des contraintes intuitives, comme le fait que les phonèmes regroupés doivent se produire sur un intervalle de temps commun, permettent de réduire grandement l'ensemble des suites de groupements possibles. Ces suites, ou chemins dans le graphe obtenu, sont ensuite évaluées par une fonction de coût estimant la probabilité que les groupements constitués représentent un unique phonème. Le meilleur chemin donne donc la suite des regroupements. Nous ne nous sommes intéressés qu'à la phase suivante de la recombinaison, qui consiste à associer un phonème unique à un regroupement de phonèmes. Nous sommes donc en présence de trois problèmes de classification : étant donnée

une description correspondant à la présence de k phonèmes sur les quatre bandes, lui associer un phonème classe unique. Les bases de données contiennent des exemples de regroupement de 2 phonèmes sur les quatre bandes (DICO2), de trois phonèmes (DICO3) ou de quatre phonèmes (DICO4). Les résultats de l'évaluation par validation croisée des classifieurs sur chaque ensemble de données sont présentés dans les paragraphes suivants.

2 bandes (DICO2)

La description des exemples ne contient de valeurs que pour deux bandes de fréquences. Les deux autres attributs sont valués à une valeur constante (29, puisque les vingt-huit phonèmes sont représentés par 0-28). Nous avons utilisé une valeur supplémentaire car la sémantique des exemples 2 bandes est bien que deux bandes sont considérées les deux autres étant sans importance dans ce cas. Ce n'est pas la sémantique des valeurs manquantes ! La valeur manquante n'est pas une valeur à part entière : elle désigne une des valeurs connues de l'attribut considéré. Ici, la valeur 29 indique que l'attribut n'intervient pas dans la classification.

Caractéristiques							
$\|E\|$	: 8189	Classes	: 29	Valeurs manquantes	: Non	Incohérences	: Oui
Attributs	: 4	Nominaux	: 4	Linéaires discrets	: 0	Linéaires continus	: 0

Le nombre de descriptions possibles est $6 \times 29 \times 29 = 5046$. Le nombre d'exemples que contient la base est donc supérieur à ce nombre. Néanmoins, les bases d'exemples dans le domaine de la reconnaissance de la parole contiennent un grand nombre de redondances. On peut observer que les ensembles d'apprentissage ne contiennent que 1809 exemples distincts en moyenne parmi les 9000 exemples des ensembles d'apprentissage. De plus, ces bases contiennent de nombreuses incohérences. Ainsi une même description peut être associée à plusieurs classes. Si l'on prend en compte les incohérences, le nombre de couples (*description*, *classe*) possibles est $29 \times 5046 = 146334$. Ainsi nous n'avons donc que 1809 exemples distincts dans l'ensemble d'apprentissage sur les 146334 possibles.

PCC					
				Défaut	: 13.7±1.3
cons	: 84.1±1.4	gen	: 2.8±1.2	gen_cons	: 87.9±1.5
PEBLS	: 92.3±1.3	C4.5	: 82.3±1.0	CN2	: 86.0±1.5
SE	: -	SE 0.05	: -		

Les précisions des classifieurs sont assez bonnes, sauf celle de la classification par portée après généralisation de toutes les règles sans vérification de la cohérence (gen). Nous pouvons vérifier que cette mauvaise performance est liée au très petit nombre de règles engendrées. Cela semble provenir de la combinaison d'un mauvais réglage automatique de la classification par portée et de la présence d'incohérences. La valeur de ϵ utilisée (c'est-à-dire résultant du réglage automatique) est, en effet, beaucoup trop grande. Nous avons conservé ces valeurs pour respecter l'intégrité du test et rester cohérents avec le traitement appliqué à toutes les bases. Il semble toutefois évident qu'il faudrait améliorer la procédure de réglage automatique afin d'éviter de retenir des valeurs de ϵ trop grandes.

Temps					
cons	: 49:50±54.89	gen	: 8h56±10:54	gen_cons	: 6h15±7:25
PEBLS	: 16:55±8:30	C4.5	: 6h17±18:03	CN2	: 3:04±5.26
SE	: -	SE 0.05	: -		

Les temps d'exécution obtenus sont un peu trop grands pour une reconnaissance de la

parole en temps réel, surtout quand on considère les six heures qu'il faut à C4.5 pour classer moins de 1000 phonèmes.

Paramètres de la classification par portée

$\|EA\|$: 1809±11.86	Temps : 2h12	Critère : C	Cohérence (ϵ) : 50 Généralité (M) : 2
$\|gen(EA)\|$: 29±0 (29±0)		$\|gen\_cons(EA)\|$: 1809±11.86 (0±0)	

Les règles générées par gen sont trop générales puisque ce sont les 29 règles de partie gauche vide et ayant pour conséquence chacune des 29 classes possibles. La classification se fait donc par tirage au sort entre les 29 classes. La précision obtenue est même inférieure au $1/29 = 3.4\%$ que l'on pourrait espérer dans ce cas. D'un autre côté, toutes les règles sont mutuellement contradictoire et donc $gen\_cons(EA)$ ne contient aucune règle, mais simplement tous les exemples de l'ensemble d'apprentissage. Ces deux comportements ne sont pas surprenants étant donnés les paramètres de modulation de la cohérence retenus. La précision obtenue par gen_cons permet d'ailleurs de vérifier que le réglage automatique n'a pas permis à la classification par portée d'atteindre sa meilleure précision. En effet, une fois l'ensemble des exemples généralisé en règles, gen_cons réalise une classification par portée à partir du nouvel ensemble d'apprentissage (dans le cas présent, il est égal à l'ensemble d'apprentissage initial), avec les paramètres $\epsilon = 0$ et $M = 1$. Nous constatons que ces paramètres par défaut conduisent à une précision de 87.9% au lieu des 84.1% obtenus après le réglage automatique. Les valeurs obtenues à la suite du réglage automatique ne permettent donc pas nécessairement d'obtenir la meilleure précision du classifieur sur d'autres problèmes de classification (ensembles d'apprentissage et de test) extraits du même ensemble de données. Régler un classifieur de façon à ce que sa précision soit optimale sur des ensembles d'apprentissage et de test fixés ne garantit pas que sa précision reste optimale sur d'autres ensembles.

3 bandes (DICO3)

La taille de l'ensemble d'apprentissage rendant trop longue la généralisation des exemples en règles, nous n'avons pas utilisé gen et gen_cons. Dans ce problème de classification, les exemples ont la valeur constante 29 pour une des bandes de fréquences. Le nombre de descriptions possibles est $4 \times 29^3 = 97556$, le nombre d'exemples possibles est donc $29 \times 97556 = 2829124$. Une fois les redondances éliminées, les ensembles d'apprentissage ne contiennent plus que 5138 exemples en moyenne. Le problème de classification est donc loin de se réduire ici à un apprentissage par cœur.

Caractéristiques

$\|E\|$: 14132	Classes : 29	Valeurs manquantes : Non	Incohérences : Oui
Attributs : 4	Nominaux : 4	Linéaires discrets : 0	Linéaires continus : 0

PCC		Défaut : 16.5±0.6
cons : 83.8±0.6	gen : -	gen_cons : -
PEBLS : 85.9±0.6	C4.5 : 73.5±0.9	CN2 : 78.9±0.5
SE : -	SE 0.05 : -	

Les précisions des classifieurs sont encore très intéressantes. Il faut souligner que, dans tous ces tests, l'écart-type associé à la validation croisée est très faible. Les données semblent donc être assez homogènes. Cela pourrait indiquer que les données offrent une bonne représentativité des couples $(description, classe)$ existant réellement. Il faut souligner, en effet, que sur les

2829124 combinaisons théoriquement possibles, une partie seulement décrivent des situations réelles. Les faibles valeurs des écarts-types indiqueraient donc que les classifieurs ont atteint leur précision maximale sur le problème posé par la recombinaison de 3 bandes.

Temps		
cons : 5h47±7:25	gen : -	gen_cons : -
PEBLS : 39:24±16:42	C4.5 : 25h40±1h04	CN2 : 16:33±17.97
SE : -	SE 0.05 : -	

Paramètres de la classification par portée				
$\|EA\|$: 5138.0±15.2	Temps : 12h04	Critère : C	Cohérence (ϵ) : 10	Généralité (M) : 2
$\|gen(EA)\|$: - -		$\|gen\_cons(EA)\|$: - -		

Les valeurs des paramètres résultant du réglage automatique semblent plus raisonnables que dans le cas de DICO2. Nous pouvons d'ailleurs constater que la précision de la classification par portée est plus proche des meilleures précisions des autres classifieurs qu'elle ne l'était dans le cas précédent.

4 bandes (DICO4)

La taille de l'ensemble d'apprentissage rendant encore trop longue la généralisation des exemples en règles, nous n'avons toujours pas utilisé gen et gen_cons. Le nombre de descriptions possibles dans ce problème est $29^4 = 797281$, le nombre d'exemples possibles est $29 \times 707281 = 20511149$. Une fois les redondances éliminées, les ensembles d'apprentissage ne contiennent plus que 9940 exemples en moyenne. Le problème semble de plus en plus difficile.

Caractéristiques				
$\|E\|$: 30608	Classes : 29	Valeurs manquantes : Non	Incohérences : Oui	
Attributs : 4	Nominaux : 4	Linéaires discrets : 0	Linéaires continus : 0	

PCC			Défaut : 13.2±0.4
cons : 82.5±0.5	gen : -	gen_cons : -	
PEBLS : 87.6±0.6	C4.5 : 78.0±0.7	CN2 : 84.2±0.6	
SE : -	SE 0.05 : -		

Pourtant la précision des classifieurs reste du même ordre que dans les deux problèmes précédents.

Temps		
cons : 22h30±42:38	gen : -	gen_cons : -
PEBLS : 1h24±1:19	C4.5 : 2j.21h13±2h34	CN2 : 50:43±1:12
SE : -	SE 0.05 : -	

Les temps d'exécution augmentent encore puisque le nombre d'exemples est plus grand. Le temps de classification de C4.5 est toujours conséquent : près de 3 jours pour classer 3000 exemples.

Paramètres de la classification par portée				
$\|EA\|$: 9940.2±35.6	Temps : 6h43	Critère : C	Cohérence (ϵ) : 10	Généralité (M) : 2

$\|gen(EA)\|$: - -	$\|gen\_cons(EA)\|$: - -

Les paramètres obtenus par le réglage automatique semblent encore bien réglés.

Conclusion

Les résultats obtenus sur les trois problèmes de classification relatifs à la reconnaissance de la parole sont très semblables. La précision moyenne des classifieurs est de l'ordre de 80% à 90%. Leur écart-type très faible semble indiquer que les bases de test sont homogènes et qu'elles représentent bien le domaine considéré. Les précisions obtenues sont donc vraisemblablement les meilleures précisions que peut atteindre chaque classifieur sur ces problèmes.

Les temps d'exécution de la classification par portée, et surtout de C4.5, ne les rendent pas utilisables dans une application en temps réel. Même si on ne prend pas en compte le temps d'apprentissage, leurs temps de reconnaissance sont trop importants. PEBLS et CN2 semblent plus appropriés.

En ce qui concerne les paramètres de la classification par portée, nous avons remarqué sur le problème DICO2 que les paramètres obtenus à la suite du réglage automatique ne conduisent pas toujours à une meilleure précision que les valeurs « par défaut » $\epsilon = 0$ et $M = 1$. Nous avons vérifié sur ce même domaine qu'un mauvais réglage de ϵ et M peut entraîner de mauvais résultats de la classification par portée avec généralisation des exemples en règles.

9.3.3 Aide à la décision dans le domaine bancaire

Nous présentons dans ce paragraphe des résultats de la classification par portée qui nous ont été fournis par Ron Rymon. Nous avions pris contact avec Ron Rymon pour lui faire part de nos travaux sur les arbres d'énumération d'ensemble et sur la classification par portée. Il a été très intéressé par la complémentarité que la classification par portée apportait à son programme SE-Learn et il l'y a intégrée. L'implantation qu'utilise Ron Rymon ne bénéficie pas de la rapidité de l'implantation « diviser pour régner » que nous avons retenue. Elle n'exploite pas, à notre connaissance, les paramètres de réglage de la cohérence. Néanmoins, Ron Rymon nous a fait des commentaires encourageants sur les performances qu'il observait sur les données sur lesquelles il travaillait. Pour illustrer ses dires, il nous a même communiqué les résultats que nous présentons dans ce paragraphe.

Le problème réel considéré a pour but d'aider une banque à décider d'accorder ou non un prêt à un client. L'ensemble des données provient de la banque ABN AMRO aux Pays-Bas. Cet ensemble de données a été traité à l'aide d'arbres de décision par Feelders, Le Loux et Van't Zand et a fait l'objet d'une publication à l'édition 1995 de la conférence internationale sur la fouille de données [Feelders *et al.*, 1995]. L'ensemble de données comporte 52569 exemples décrits par 13 attributs. Ces attributs indiquent si le client est propriétaire ou locataire, le nombre de prêts qu'il a déjà contractés, etc. La classe indique si le client est un mauvais payeur. Ron Rymon a essayé d'utiliser son logiciel SE-Learn pour développer un arbre d'énumération d'ensembles. Comme il serait très long de développer un arbre complet, sa stratégie consiste à développer tout d'abord un arbre de décision en choisissant pour chaque nœud l'attribut le plus discriminant (au sens d'un critère comme le gain d'information [Quinlan, 1986]), puis à ajouter à chaque nœud les fils correspondant au deuxième attribut le plus discriminant, et ainsi de suite. Ron Rymon a ainsi développé un arbre pendant plusieurs jours sans obtenir de résultats significativement différents de ceux de l'arbre de décision initial. Il a évalué sur ce problème la classification par portée. Les résultats qu'il nous a communiqués sont représentés

sur la figure 9.3.

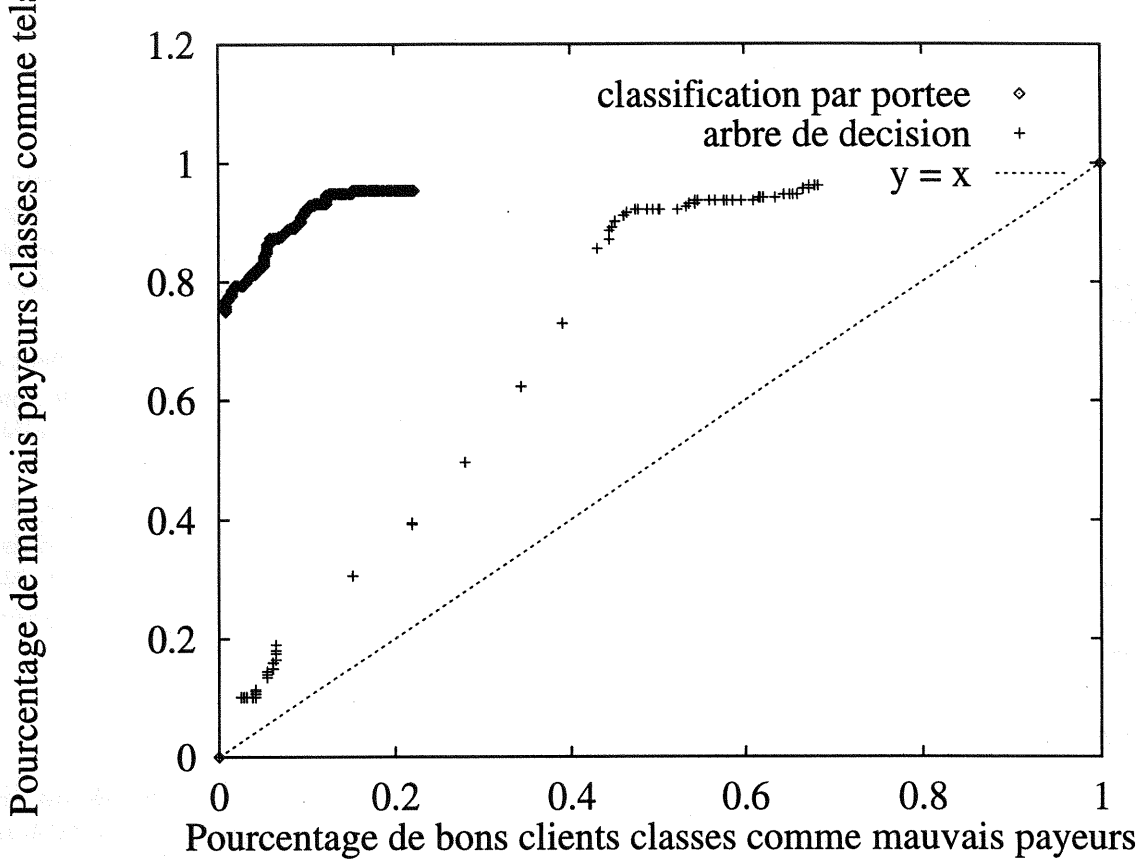


FIG. 9.3 – *Précision de la classification par portée comparée à celle d'arbres de décision dans l'aide à la décision d'attribution de prêts bancaires.*

Les courbes sont construites de la façon suivante : les individus à classer sont triés en fonction de la vraisemblance qu'ils soient de mauvais payeurs. La vraisemblance est le rapport du nombre de règles (ou de voisins) concluant à la classe mauvais payeur par rapport au nombre total de règles s'appliquant au (ou de voisins du) client. Chaque point des courbes est obtenu en fixant un seuil de vraisemblance au delà duquel les individus sont considérés comme mauvais payeur. L'abscisse du point représente le pourcentage de faux positifs, c'est-à-dire le pourcentage de bons clients qui ont été classés comme mauvais payeur. L'ordonnée est le pourcentage de vrais positifs, c'est-à-dire le pourcentage de mauvais payeur qui ont été classés comme tels. L'idéal serait d'obtenir des points les plus proches possibles de $(0, 1)$ où aucun bon client n'est rejeté et où tous les mauvais clients sont refusés. Si les clients sont classés au hasard, les points sont situés le long de la droite $y = x$. Plus les points sont au-dessus de cette droite et sont proches de l'angle supérieur gauche, meilleur est le classifieur.

Je laisse Ron Rymon conclure :

> « *You will note that the [scope alg.* (classification par portée)*] does WAY better than the decision tree. I would never have reached the full SE-tree without the scope alg. Of course, now that I know that, maybe I can run traditional SE-Learn on a larger computer for longer time until it obtains similar results. Then, I will have the benefit of quick application, and (maybe) interpretability. Or, I can continue to*

use scope for classification. At any rate, I would never have known that there is an advantage in exploring an SE-tree vs. a decision tree without the scope alg. »

9.4 Résumé et conclusion

Dans ce chapitre, nous avons présenté les résultats d'une évaluation expérimentale de la classification par portée. Nous avons, dans un premier temps, présenté la plate-forme de test que nous avons développée et qui permet de comparer la classification par portée avec des programmes de référence dans le domaine de l'apprentissage automatique. Les approches considérées sont les approches à base d'instances fondées sur des distances (PEBLS), les approches à base de quelques règles « choisies » (C4.5 et CN2), et les approches à base de toutes les règles minimales cohérentes et pertinentes pour l'ensemble d'apprentissage, avec ou sans élagage (SE-Learn).

Trois variantes de la classification par portée ont été évaluées : la classification par portée à partir d'exemples seulement, et les deux stratégies de généralisation des exemples en règles gen(EA) et gen_cons(EA) (cf. paragraphe 7.3.3). Nous avons constaté que ces stratégies de généralisation n'apportent pas les gains de précision, de rapidité, et d'intelligibilité espérés. Cela s'explique clairement par leur sensibilité aux paramètres de modulation de la cohérence ϵ et M. Ces paramètres influent crucialement sur le nombre de règles générées. Nous avons d'ailleurs constaté que la procédure de réglage automatique demande à être améliorée dans ses résultats et dans son temps de calcul.

La classification par portée à partir d'exemples seulement a un temps d'exécution semblable à ceux de C4.5 et PEBLS. Les programmes CN2 et SE-Learn avec un élagage statistique sont plus rapide à mettre en oeuvre en moyenne. Notons cependant que SE-Learn ne s'applique que sur des problèmes de petite taille. En effet, il construit l'ensemble des règles minimales cohérentes et pertinentes, or ce nombre de règles peut être exponentiel dans le nombre d'attributs. SE-Learn est inapplicable en pratique sur des problèmes de plus de 15 attributs, sauf pour certains cas particuliers, par exemple quand le nombre d'exemples est très petit. La classification par portée offre donc une alternative pour les problèmes comportant un grand nombre d'attributs.

Nous avons également pu vérifier expérimentalement que la classification par portée à partir d'exemples seulement offre une précision un peu supérieure à celle des arbres d'énumération d'ensembles. Dans l'ensemble, ces deux techniques de classification à base de toutes les règles minimales cohérentes et pertinentes pour l'ensemble d'apprentissage ont une bien meilleure précision que les approches à base de quelques règles « choisies », ou les approches à base d'instances utilisant un nombre constant de voisins.

Les avantages et les inconvénients de la classification par portée, et donc ses domaines d'application, ainsi que des perspectives de développement font l'objet du chapitre suivant.

Chapitre 10

Bilan et perspectives de la classification par portée

Nous présentons dans ce chapitre un bilan des avantages et inconvénients de la classification par portée. Plus généralement, nous précisons les facteurs à prendre en compte dans le choix d'un classifieur. Ces facteurs permettent de choisir l'approche la plus adaptée à un problème de classification donné. Après ce bilan des qualités et défauts de notre approche, nous présentons quelques perspectives de développement telles que la prise en compte incrémentale d'exemples, l'introduction d'un ordre sur les attributs et la prise en compte d'événements temporels, ou l'utilisation d'une théorie du domaine. Nous soulignons également quelques limites inhérentes au biais inductif et au biais de langage des classifieurs à base de règles.

10.1 Bilan

Les trois facteurs principaux qui caractérisent la performance d'un classifieur, une fois le langage de représentation fixé, sont sa précision, sa rapidité d'apprentissage et de classification, et sa compréhensibilité. Nous traitons ces trois facteurs dans cette section. Nous considérons tout d'abord la précision. Nous soulignons à nouveau la séparation entre les biais logique et statistique que nous avons retenue et nous montrons comment notre biais logique pur, sans modulation de la cohérence, peut offrir une bonne résistance au bruit. Cette qualité est bien sûr renforcée par la modulation de la cohérence. Nous considérons ensuite le second facteur. Nous soulignons que le facteur de rapidité regroupe deux facteurs distincts et qu'un classifieur rapide en apprentissage ne l'est pas toujours en classification, et réciproquement. Nous précisons donc les domaines privilégiés de chaque approche. Nous terminons par le facteur de compréhensibilité. En particulier, nous montrons, qu'en pratique, ce facteur s'oppose curieusement à la précision.

10.1.1 Précision

Dans nos expérimentations, la classification par portée obtient en moyenne une meilleure précision que celles obtenues par les approches à base d'instances ou à base de règles. Cette moyenne a été évaluée sur un grand nombre de problèmes réels usuellement utilisés pour tester les algorithmes d'apprentissage automatique.

Le biais inductif de la classification par portée est le biais logique de l'ensemble de toutes les règles cohérentes et pertinentes pour l'ensemble d'apprentissage. La classification par por-

tée est équivalente à l'espace des versions disjonctif du point de vue de la classification (cf. paragraphe 8.4.2). Elle est également équivalente, d'un point de vue logique (cf. paragraphe 6.4) et en pratique (cf. chapitre 9) avec les arbres d'énumération d'ensembles. Ces trois techniques bénéficient d'une même résistance au bruit. En effet, même sans modulation de la cohérence ($\epsilon = 0$, $M = 1$), dans le cadre de règles pertinentes et cohérentes pour l'ensemble d'apprentissage, l'utilisation de toutes les règles compense la présence de bruit et permet d'utiliser un principe inductif logique pur. Cette résistance au bruit provient simplement de ce que plus on dispose d'indicateurs de la classe, plus on s'immunise contre le bruit [Gams, 1989; Nock et Gascuel, 1995]. Chaque technique offre une vision différente de cette idée :

- Adoptons tout d'abord le point de vue des instances, qui est celui de la classification par portée. Étant donné un objet o à classer, la classification par portée construit un ensemble de voisins de plus grande taille que les approches à base de distances. Nous avons montré en effet (cf. propriété 8.1) que tout voisin pour une distance usuelle est un voisin pour la classification par portée. Au lieu de considérer un nombre fixe k de voisins, nous retenons tous ceux à partir desquels une règle cohérente peut être construite. Si un des voisins de o est incorrect, c'est-à-dire si sa classe est incorrectement prédite, son influence sera réduite lors du vote puisqu'un plus grand nombre de voisins interviennent.

- L'explication est semblable lorsque l'on adopte le point de vue des règles. Dans les approches à base de quelques règles « choisies », comme les arbres de décision (C4.5) ou CN2, le nombre de règles qui couvrent un objet o à classer est faible. Si une règle est incorrecte, o a plus de chances d'être mal classé que dans une approche à base de toutes les règles, telle que les arbres d'énumération d'ensembles (SE-Learn). Dans cette dernière approche, un grand nombre de règles peuvent couvrir o et l'influence des règles incorrectes est corrigée par le nombre de règles correctes qui couvrent o également.

- La résistance au bruit a également été expliquée par Michèle Sebag dans le cadre de l'espace des versions disjonctif [Sebag, 1996]. L'espace des versions construit en considérant un exemple e comme seul exemple positif et ses contre-exemples $CE(e, EA)$ comme exemples négatifs « *est indûment spécifique dans le cas où des exemples négatifs incorrects sont rencontrés, mais il n'inclut encore que des hypothèses cohérentes. Maintenant, si la racine elle-même est un exemple positif incorrect, l'étoile correspondante contient des hypothèses hasardeuses. Cette limitation est traitée en considérant symétriquement le concept cible et sa négation : plus précisément, chaque exemple à son tour est pris comme racine et généralisé par rapport à tous les exemples appartenant à d'autres concepts que la racine. Cela permet heureusement aux hypothèses hasardeuses apprises à partir d'exemples négatifs incorrects de contre-balancer les hypothèses hasardeuses apprises à partir d'exemples positifs incorrects.* »

Un classifieur ne peut pas obtenir la meilleure précision sur tous les problèmes de classification mathématiquement possibles [Schaffer, 1994]. Néanmoins, Pedro Domingos suppose qu'il doit être possible d'obtenir un classifieur qui ne soit pas le meilleur sur tous les problèmes possibles, mais simplement sur une partie de ceux-ci. Il s'agit simplement d'être le meilleur sur tous les problèmes réels, quitte à être moins précis sur les autres. Ce pari ne semble pas si utopique. Pour preuve, son algorithme RISE, tout comme la classification par portée, constitue une avancée dans cette direction [Domingos, 1996]. Il faut cependant noter que la plupart des problèmes provenant de l'Université d'Irvine sur lesquels RISE et la classification par portée sont évalués respectent les biais inductifs utilisés par les programmes d'apprentissage usuels.

Il est clair que sur des problèmes réels tels que l'apprentissage de la fonction parité (cf. paragraphe 5.1) les biais de nos techniques sont inadaptés [Rendell et Cho, 1990]. Il faut donc espérer que les problèmes réels sur lesquels nos techniques sont évaluées satisfont leurs biais afin de pouvoir clamer qu'elles sont les meilleures sur tous *ces* problèmes réels.

10.1.2 Rapidités d'apprentissage et de classification

La classification par portée bénéficie de la rapidité, et souffre de la lenteur, des approches à base d'instances. En effet, les approches à base d'instances n'incluent, en principe, pas de phase d'apprentissage. Elles sont également nommées *apprentissage paresseux*, en anglais *lazy learning,* car elles se contentent de stocker les exemples avant de procéder à la classification. Le problème est que chaque opération de classification oblige à prendre en compte tous les exemples au moins un fois pour procéder à la classification. La complexité en temps est au moins linéaire en $|EA|$ puisqu'il faut considérer au moins une fois tous les exemples avant de se prononcer. Elle est parfois quadratique en $|EA|$, comme dans le pire cas de la classification par portée. Le temps d'apprentissage est donc très avantageux (il est nul !), mais le temps de classification peut être important.

Les approches à base de règles ont, en principe, les caractéristiques opposées. Elles nécessitent une phase d'apprentissage pour construire un arbre ou un ensemble de règles. En revanche, elles permettent souvent de classer un nouvel objet de façon rapide à l'aide de la structure abstraite, arbre ou ensemble de règles, qu'elles ont construite auparavant.

Les approches à base d'instances sont donc mieux appropriées à des problèmes de classification *dynamiques*, c'est-à-dire des problèmes nécessitant de reprendre fréquemment l'apprentissage. Cela couvre les problèmes où les données sont fournies incrémentalement en cours d'utilisation du système de classification, les cas où la validité des exemples est limitée dans le temps, ou encore les cas où la description des exemples évolue au cours du temps. Les approches à base de règles sont inversement mieux appropriées à des problèmes *statiques* où un ensemble d'exemples de grande taille est fourni initialement et le système doit classer une grande quantité de nouveaux objets sans que l'ensemble d'apprentissage ne change. Le temps de classification d'une approche à base d'instances dépend des mêmes facteurs que le temps d'apprentissage d'une approche à base de règles. Ce sont bien sûr la taille de l'ensemble d'apprentissage $|EA|$ et le nombre d'attributs d utilisés. Le temps de classification t_1 d'une approche à base d'instances est en général bien plus faible que le temps d'apprentissage t_2 d'une approche à base de règles. Il existe donc un nombre d'objets à classer de l'ordre de t_2/t_1 au delà duquel une approche à base de règles est préférable et en deçà duquel une approche à base d'instances est plus rapide.

Les avantages et inconvénients respectifs des approches à base d'instances et des approches à base de règles ne sont cependant pas toujours très nets du point de vue du temps d'exécution. En effet, un grand nombre d'approches à base d'instances ont recours à une phase d'apprentissage. Cet apprentissage peut avoir pour but d'éliminer les attributs non pertinents ou les exemples incorrects [Aha *et al.*, 1991; Domingos, 1997] ou encore de régler les seuils de cohérence et de spécificité dans la classification par portée. Il peut également servir à réduire le nombre d'exemples et à ne conserver que des instances représentatives de l'ensemble d'apprentissage[Aha *et al.*, 1991] ou à généraliser les exemples en règles [Salzberg, 1991; Domingos, 1996]. Dans ce cas, une approche à base d'instances devient une approche à base de règles. Les approches à base de règles nécessitent, quant à elles, un temps d'apprentissage parfois très long et génèrent parfois un nombre de règles très important. Le nombre de règles possibles est, en effet, exponentiel en le nombre d'attributs multiplié par le nombre de valeurs

et nous avons pu vérifier expérimentalement que, dans le cas d'un grand nombre d'attributs et d'un ensemble d'apprentissage restreint (cas du registre lorrain du cancer de l'enfant), le nombre de règles apprises peut facilement être très supérieur au nombre d'exemples initiaux. Le temps de classification à partir des règles n'est donc pas toujours plus faible que celui nécessaire à la classification à partir des exemples.

Le choix d'un classifieur rapide doit donc prendre en compte la taille de l'ensemble d'apprentissage et de l'ensemble d'objets à classer, le nombre d'attributs et leur arité (nombre de valeurs). Ces paramètres permettent d'évaluer les temps d'apprentissage et de classification, et donc le temps total d'exécution de la classification.

10.1.3 Compréhensibilité

Les approches à base de règles offrent traditionnellement une meilleure intelligibilité que les approches à base d'instances. Les approches à base d'instances expliquent la classification qu'elles proposent par la donnée des exemples les plus proches de l'objet à classer. Dans la plupart des domaines réels, le domaine juridique par exemple, la référence à des exemples proches constitue une bonne explication. Les approches à base de règles fournissent, pour leur part, l'ensemble des règles qui s'appliquent à l'objet classé. Ces règles sont assez souvent générales. Un avantage des approches à base de règles est que l'ensemble des règles construites lors de la phase d'apprentissage permet d'offrir une vision synthétique du domaine considéré. Cet avantage n'est cependant pas exclusivement l'apanage des approches à base de règles. En effet, comme nous l'avons évoqué dans le paragraphe précédent, certaines approches à base d'instances construisent aussi une abstraction [Salzberg, 1991; Domingos, 1996] qui peut offrir une meilleure compréhensibilité du domaine.

La sémantique de la classification par portée permet de construire des règles expliquant la classification effectuée (cf. paragraphe 6.8). Ces règles peuvent être généralisées et constituer une représentation compacte du domaine. Néanmoins, ces règles sont souvent nombreuses et concluent parfois sur des classes différentes. Elles ne permettent donc pas forcément d'avoir une meilleure compréhension du domaine. Ce problème n'est pas propre à la classification par portée. En fait, toutes les techniques qui acceptent plus d'une hypothèse couvrant un individu donné souffrent de ce manque d'intelligibilité. À l'opposé, nous avons pu constater en pratique que l'utilisation d'un grand nombre d'hypothèses permet d'obtenir une meilleure précision de classification. Ce point est détaillé ci-dessus (paragraphe 10.1.1). Il semble donc que les contraintes d'intelligibilité et de précision s'opposent. Cela est surprenant car, intuitivement, il semble qu'une bonne intelligibilité du problème permet justement de pallier le bruit. Nous pensons qu'un bon réglage de la cohérence doit permettre d'obtenir une représentation compacte et résistante au bruit. Nous avons vu en effet (cf. paragraphe 7.2) que ce réglage permet de faire varier la taille du voisinage $(\epsilon, M) - cons(EA, \leq_o)$ d'un objet o à classer. Un bon élagage devrait donc permettre d'obtenir une meilleure compréhension de la classification de o, que ce soit simplement en donnant ses voisins, comme dans une approche à base d'instances, ou en générant des règles (cf. paragraphe 6.8). De plus, il devrait permettre de réduire le nombre de règles provenant de la généralisation des exemples, et donc d'obtenir une description concise du domaine.

10.1.4 Conclusion

La classification par portée offre une meilleure précision et une rapidité comparable à celles des approches à base d'instances et des approches à base de règles avec lesquelles nous l'avons

comparée. Elle offre la possibilité d'expliquer les classifications effectuées à l'aide des plus proches voisins ou grâce à un ensemble de règles. Cette compréhensibilité est certes limitée, mais est la même que celle offerte par la plupart des approches à base d'instances ou à base de règles de même précision.

La classification par portée peut facilement être parallélisée. Elle ne nécessite pas de discrétiser les attributs linéaires. Elle offre donc un avantage considérable par rapport à la plupart des approches logiques qui ne savent prendre en compte les attributs linéaires qu'à l'aide d'une discrétisation. Comme le note Michèle Sebag dans le cadre de l'espace des versions disjonctif [Sebag, 1996] : « *de nombreux travaux ont rapporté combien la précision de l'induction symbolique pure dépend de l'étape de discrétisation.* » Il faut noter, par exemple, que PEBLS et SE-Learn ne savent pas traiter les attributs continus. Ils les discrétisent et, à notre connaissance, ils ne tiennent pas compte de l'ordre existant entre les valeurs discrètes obtenues. En outre, PEBLS et SE-Learn ne savent pas traiter les valeurs manquantes. Il traitent l'absence de valeur comme une valeur supplémentaire. Deux objets peuvent donc avoir en commun la valeur manquante ! Si une règle contient un attribut valué par la valeur manquante, cela n'est pas équivalent à l'absence de condition sur cet attribut. Au contraire, cette condition ne peut être satisfaite que par un objet ayant également une valeur manquante pour cet attribut ! Tous les programmes que nous avons testés sont prévus pour classer un ensemble de test par rapport à un ensemble d'apprentissage. Nous ne savons pas si parmi ces programmes, certains prennent en compte explicitement les redondances. Il est évident que cette opération permet à la classification par portée de gagner énormément de temps sur des bases telles que les bases de parole (DICO2, DICO3, DICO4), sans influer sur les classifications effectuées.

10.2 Perspectives

La classification par portée présente, à notre avis, un bilan positif. Néanmoins, de nombreux points peuvent encore être améliorés. Nous présentons dans les paragraphes suivants quelques réflexions sur des améliorations possibles de notre approche. Ces améliorations peuvent d'ailleurs être bénéfiques à d'autres approches à base d'instances ou à base de règles.

10.2.1 Réduction de l'ensemble d'apprentissage

Un des principaux défauts des approches à base d'instances et de la classification par portée est le temps de classification. Le temps de classification de la classification par portée est, dans le pire cas, quadratique en $|EA|$, en moyenne expérimentalement proche de $|EA| \times log_2(|EA|)$, et dans le meilleur cas linéaire. Même dans le meilleur cas, cela rend la classification très longue sur des grands ensembles de test. Or, les ensembles de données réelles sont souvent de grande taille. Réduire l'ensemble d'apprentissage est donc une nécessité. Nous avons déjà proposé et utilisé une prise en compte des redondances qui a permis de réduire considérablement certains ensembles de données, comme les bases de paroles (DICO2, DICO3, DICO4), mais cela n'est pas suffisant.

Pour réduire la taille de l'ensemble d'apprentissage et améliorer la précision de la classification, une première approche est de procéder à la classification de façon incrémentale. Cette approche consiste à fixer un ensemble d'apprentissage initial, puis à considérer les exemples restants un par un, et à décider de les intégrer ou non à l'ensemble d'apprentissage. Plusieurs variantes ont été évaluées par David Aha, Dennis Kibler et Marc Albert [Aha *et al.*, 1991]. La première variante consiste à n'ajouter que les exemples qui sont mal classés par l'ensemble d'apprentissage courant. En effet, ces exemples apportent clairement une information nouvelle

à l'ensemble d'apprentissage. Elle conduit malheureusement à retenir une grande proportion d'exemples incorrects. En effet ceux-ci sont souvent mal classés par leurs voisins qui leur attribuent leur vraie classe. Une amélioration de cette technique consiste à intégrer les exemples mal classés par l'ensemble d'apprentissage courant, mais à les évaluer par la suite afin de les retirer s'ils paraissent incorrects. Cette procédure de filtrage utilise les classifications suivantes pour évaluer les exemples contenus dans l'ensemble d'apprentissage. Selon l'hypothèse des similarités, un exemple doit appartenir à la classe de ses voisins, et réciproquement. Il est donc possible d'évaluer la classe d'un exemple e à partir de l'observation des classes des exemples que e classe par la suite : s'il n'intervient que dans le voisinage d'exemples d'autres classes, il est supposé incorrect.

Une seconde approche consiste à généraliser les exemples en règles, comme l'ont fait Steven Salzberg [Salzberg, 1991] ou Pedro Domingos [Domingos, 1996]. Nous avons présenté dans les chapitres précédents les stratégies de généralisation que nous avons proposées, ainsi que leurs résultats expérimentaux. Ces résultats ne répondent malheureusement pas à notre attente. Nous pensons néanmoins qu'il est possible de réduire le nombre de règles en réglant précisément les paramètres de modulation de la cohérence de façon à ce que l'ensemble des voisins de chaque objet à classer soit de taille raisonnable. Par ailleurs, les règles comportant des attributs numériques ne sont pas assez générales. Cela pourrait expliquer le grand nombre de règles générées sur des bases contenant des attributs linéaires continus. Il faudrait généraliser les règles en élargissant chaque intervalle numérique jusqu'aux contre-exemples les plus proches. Dériver des règles plus générales devrait permettre de réduire le nombre de règles conservées.

10.2.2 Extension du pré-ordre et prise en compte d'événements temporels

Tous les attributs n'ont pas nécessairement la même importance. Nous proposons de modifier les pré-ordres partiels utilisés par la classification par portée afin de considérer un ordre sur les attributs.

Définition 10.1 *Soit $A = \{a_1, \ldots, a_d\}$ l'ensemble des attributs utilisés pour décrire les objets. On munit $\mathcal{P}(A)$ d'un ordre total $\succ$ indiquant l'importance relative des sous-ensembles d'attributs : $A_1 \succ A_2 \succ \ldots \succ A_k$. Les A_i sont les sous-ensembles de A considérés. Soient $\alpha \in \mathcal{P}(A)$ et o un objet. On note $o_{|\alpha}$ la restriction de o à α, c'est-à-dire la description de o limitée aux attributs apparaissant dans α. On note $\alpha_1 \succ \alpha_2$ si et seulement si α_1 est au moins aussi important que α_2. Le pré-ordre $o_1 \overset{A_1 \succ A_2 \succ \ldots \succ A_k}{\leq_o} o_2$ est défini récursivement de la façon suivante :*

- $o_1 \overset{\emptyset}{\leq_o} o_2$ *est toujours vrai,*
- $o1_{|A_1} \leq_{o|A_1} o2_{|A_1}$ *et* $o2_{|A_1} \not\leq_{o|A_1} o1_{|A_1}$,
- $o1_{|A_1} \leq_{o|A_1} o2_{|A_1}$, $o2_{|A_1} \leq_{o|A_1} o1_{|A_1}$ *et* $o_1 \overset{A_2 \succ \ldots \succ A_k}{\leq_o} o_2$.

Considérons les objets suivants :

	a_1	a_2	a_3
o	1	1	1
o_1	1	1	2
o_2	1	2	1

Ces objets vérifient les relations suivantes :

- $o_1 \overset{\{a_1\} \succ \{a_2\} \succ \{a_3\}}{\leq_o} o_2$

- $o_2 \overset{\{a_1\}\succ\{a_2\}\succ\{a_3\}}{\not\leq_o} o_1$
- $o_1 \not\leq_o o_2$ et $o_2 \not\leq_o o_1$.

La relation $\overset{A_1\succ\ldots\succ A_k}{\leq_o}$ spécialise la relation $\leq_o$.

Propriété 10.1 *Soient o, o_1 et o_2 des objets et A_1, ..., A_k des sous-ensembles d'attributs. Si $o_1 \leq_o o_2$ alors $o_1 \overset{A_1\succ\ldots\succ A_k}{\leq_o} o_2$.*

Preuve : si $o_1 \leq_o o_2$ alors $\forall A_i, o_{1_{|A_i}} \leq_o o_{2_{|A_i}}$. #

L'utilisation d'un ordre sur les attributs est particulièrement souhaitable lorsque l'on raisonne sur des événements temporels. Supposons que l'on considère des objets décrits par les valeurs prises par δ attributs lors des τ derniers intervalles de temps. Ces attributs peuvent être la probabilité de chaque phonème dans les deux dernières trames dans un problème de reconnaissance automatique de la parole, la température et le poids d'un patient mesurés au cours des dix derniers jours dans le cas de surveillance d'un malade ou les distances mesurées par les capteurs lors des dix derniers cycles dans le cas d'un robot mobile autonome. De manière générale, soient $a_1, \ldots, a_\delta$ les attributs considérés à chaque instant. Si les objets sont décrits par les valeurs prises par ces attributs au cours des τ dernières mesures, il semble intéressant d'ordonner les attributs de la façon suivante : $\{a_1, \ldots, a_\delta\}_t \succ \{a_1, \ldots, a_\delta\}_{t-1} \succ \ldots \succ \{a_1, \ldots, a_\delta\}_{t-\tau}$.

10.2.3 Utilisation d'une théorie du domaine

Nous avons montré au chapitre 5 comment une théorie du domaine peut être prise en compte dans un classifieur à base de règles (cf. paragraphe 5.5). Rappelons la définition des règles cohérentes et pertinentes pour un ensemble d'apprentissage en présence d'une théorie du domaine Th :

Soient EA un ensemble d'apprentissage et Th une théorie du domaine tel que chaque exemple de EA soit cohérent avec Th. Une règle cohérente et pertinente pour EA est un couple noté $G \rightarrow (x = v_x)$ tel que :

- *Pour tout exemple e de EA, si $e \wedge Th \models G$ alors $classe(e) = v_x$.*
- *Il existe un exemple e tel que $e \wedge Th \models G$.*

Pour pouvoir appliquer la classification par portée en présence d'une théorie du domaine Th, il faut être capable de construire toutes les règles cohérentes pour EA rendues pertinentes par un exemple en présence de Th. Considérons les exemples :
$e_1 = ((a = Vrai), (x = Vrai))$ et $e_2 = ((c = Vrai), (x = Faux))$. Supposons que l'on veuille classer l'objet o décrit par $(p = Vrai)$ en présence de la théorie du domaine Th :
$(a = Vrai) \Rightarrow (b = Vrai)$, $(c = Vrai) \Rightarrow (q = Vrai)$ et $(p = Vrai) \Rightarrow (b = Vrai)$. Une première approche consiste à compléter les descriptions des objets à l'aide de Th :
$e_1 = ((a = Vrai), (b = Vrai), (x = Vrai))$, $e_2 = ((c = Vrai), (q = Vrai), (x = Faux))$ et $o = ((p = Vrai), (b = Vrai))$. Il est alors facile de construire la règle $R(o, e) = (b = Vrai) \rightarrow (x = Vrai)$.

Le développement des descriptions à l'aide de la théorie du domaine permet de réduire le problème à un problème de classification sans théorie du domaine et donc d'utiliser la classification par portée telle que nous l'avons définie. Cette complétion des descriptions étant donnée une théorie du domaine est malheureusement coûteuse à réaliser dans le cas général. Il reste à construire une extension de la classification par portée qui prend en compte explicitement une théorie du domaine sans utiliser de complétion des descriptions.

10.2.4 Induction constructive et pertinence des attributs

La classification par portée est, comme beaucoup d'approches logiques, sensible à la pertinence des attributs. Tous les attributs sont considérés comme d'égale importance. Considérons deux patients venant consulter un médecin. Si les deux patients ont une chemise blanche et sont enrhumés et que le diagnostic du médecin est un rhume et si aucun patient rencontré souffrant d'une autre affection, c'est-à-dire d'une autre classe, n'est enrhumé et ne porte une chemise blanche, alors les deux règles *si le patient a une chemise blanche alors il a un rhume* et *si le patient est enrhumé alors il a un rhume* sont deux hypothèses inductives confirmées par l'ensemble d'apprentissage. Supposons qu'un nouveau patient ait une chemise blanche et le bras cassé, le diagnostic risque d'être un rhume. La présence d'attributs non pertinents est donc à éviter. Malheureusement, lorsque l'apprentissage automatique est utilisé comme outil de découverte de connaissances afin de déterminer des relations entre un ensemble d'attributs et une classe, la pertinence des attributs n'est en général pas connue (sinon il suffit de ne considérer que les attributs pertinents pour décrire les objets). Le paramètre de spécificité M permet de ne pas tenir compte d'un certain nombre d'attributs dans la classification, mais il n'est pas précisément adapté à la prise en compte des attributs non pertinents.

Un second problème est lié au choix des attributs décrivant les objets. Considérons l'ensemble d'exemples du concept booléen x :
$((a = Vert), (b = Vert)), (x = Vrai)$
$((a = Vert), (b = Bleu)), (x = Faux)$
$((a = Bleu), (b = Bleu)), (x = Vrai)$
$((a = Rouge), (b = Bleu)), (x = Faux)$
$((a = Rouge), (b = Rouge)), (x = Vrai)$
$((a = Rouge), (b = Jaune)), (x = Faux)$
Quelle est la classe de l'objet décrit par $((a = Jaune), (b = Jaune))$? Le concept décrit est l'égalité des valeurs de a et b : *Si (a = v) et (b = v) alors (x = Vrai).* Le concept peut être décrit en extension par l'ensemble des valeurs possibles de a et b :
Si (a = Vert) et (b = Vert) alors (x = Vrai);
Si (a = Bleu) et (b = Bleu) alors (x = Vrai);
Si (a = Rouge) et (b = Rouge) alors (x = Vrai).
Néanmoins la règle *Si (a = Jaune) et (b = Jaune) alors (x = Vrai)* n'est pas décrite. Il est préférable d'utiliser une description intensionnelle :
Si ((a égal b) = Vrai) alors (x = Vrai).
Pour pouvoir apprendre ce concept, il faut que l'attribut (*a égal b*) soit ajouté aux descriptions des exemples lors de la conception de la base, ou qu'il soit construit par le classifieur. On imagine bien qu'il est impossible d'entrer toutes les combinaisons possibles des attributs. C'est donc un programme qui doit s'en charger. Ce type d'induction est appelé induction *constructive* [De Raedt et Bruynooghe, 1989; Aha, 1991]. La complexité des combinaisons possibles est la même pour la machine que pour nous. L'induction constructive est donc plus délicate à réaliser que celle que nous avons considérée jusqu'à présent, mais elle est incontournable si l'on veut prendre en compte des descriptions telles qu'elles apparaissent dans les ensembles de données existants.

10.3 Résumé et conclusion

Dans ce chapitre, nous avons présenté un bilan de la classification par portée. Les facteurs primordiaux de qualité d'un classifieur sont sa précision, sa rapidité d'apprentissage et de

classification, et l'intelligibilité qu'il apporte. Nous avons structuré notre bilan suivant ces facteurs. La classification par portée à partir d'exemples seulement apparaît plus précise que les autres techniques, en particulier celles à base d'instances usuelles, ou les approches à base de quelques règles « choisies ». Nous avons rappelé que la classification par portée est équivalente, d'un point de vue logique, à la classification à base de toutes les règles cohérentes et pertinentes, et à la classification fondée sur l'espace des versions disjonctif. Nous avons montré, suivant ces trois points de vue, que l'utilisation d'un plus grand nombre d'hypothèses offre une meilleure résistance au bruit.

La classification par portée offre un temps d'exécution comparable à ceux associés aux autres techniques. Nous avons cependant souligné qu'il faut distinguer les temps d'apprentissage et de classification. La classification par portée étant une approche à base d'instances a en général un temps d'apprentissage inférieur mais un temps de classification supérieur à une approche à base de règles. Elle est donc plus appropriée à des problèmes dynamiques nécessitant des apprentissages fréquents.

En ce qui concerne l'intelligibilité, la classification par portée permet d'expliquer les classifications qu'elle effectue à l'aide de cas semblables, ou à l'aide de règles générales. Toutefois, elle souffre pour le facteur d'intelligibilité de ce qui fait sa force pour le facteur de précision : elle utilise un grand nombre d'hypothèses, ce qui améliore sa précision, mais nuit à l'intelligibilité. Pour y remédier et pour accroître l'intérêt de la généralisation des exemples en règles, une solution consiste à améliorer le réglage des paramètres de modulation de la cohérence ϵ et M. C'est une de nos perspectives de travail.

Nous envisageons également d'autres perspectives. Nous avons présenté une première réflexion sur une procédure de réduction de l'ensemble d'apprentissage. Nous avons montré comment les pré-ordres partiels utilisés par la classification par portée peuvent être spécialisés pour tenir compte d'un ordre d'importance des attributs, en particulier lorsque des événements temporels sont considérés. Nous avons montré comment utiliser une théorie du domaine dans l'état actuel de la classification par portée, en soulignant le coût de cette procédure. Nous avons également mis l'accent sur la sensibilité de la classification par portée, en tant qu'approche logique, aux attributs non pertinents. Nous avons enfin rappelé les limites imposées par les biais de représentation des hypothèses et mentionné la nécessité de l'induction constructive.

Conclusion

Cette thèse comporte deux volets. Elle se compose de trois éléments : chacun des volets et leur articulation. Dans le premier volet, intitulé *Induction confirmatoire*, nous nous sommes attachés à modéliser le processus logique de construction de lois générales, appelées généralisations confirmatoires, reflétant des régularités présentes dans un ensemble d'observations. Dans le second volet, intitulé *Classification*, nous nous sommes intéressés à un type particulier de lois, les règles associant une classe à une description. La problématique de la classification consiste à apprendre des règles à partir d'exemples. Nous avons proposé pour cela une technique de classification, appelée classification par portée. Nous avons montré qu'elle est « équivalente » d'un point de vue logique à l'utilisation de toutes les hypothèses cohérentes et pertinentes pour l'ensemble d'apprentissage, et nous avons évalué ses performances par rapport à d'autres approches de référence en apprentissage automatique. La troisième composante de cette thèse est l'articulation entre ces deux volets. Nous avons ainsi considéré les classifieurs à base de règles cohérentes et pertinentes pour un ensemble d'apprentissage et avons montré les relations existant entre les règles minimales cohérentes et pertinentes et les généralisations confirmatoires. Précisons les contributions de notre travail à chacune des trois composantes.

Nous avons rappelé ce qu'est l'induction et présenté l'induction explicative et l'induction confirmatoire. Nous avons ensuite proposé un modèle de généralisation fondé sur l'induction confirmatoire. Pour cela, nous avons d'abord proposé des postulats de rationalité que tout modèle de généralisation confirmatoire devrait satisfaire. Nous avons montré que notre modèle capture plus fidèlement l'hypothèse des similarités selon Hempel que les techniques existantes. Notre modèle réalise en effet une circonscription des individus alors que les modèles existants ont recours à une circonscription des propriétés des individus. Notre modèle est fondé sur une relation de confirmation proche de celles proposées par Carl Hempel [Hempel, 1943; Hempel, 1945]. Nous avons montré que le modèle proposé, défini dans le cadre de la logique des prédicats du premier ordre, peut se spécialiser au cadre des langages attribut-valeur. Nous avons montré que les généralisations confirmatoires sont, dans ce cadre, les premiers impliqués des observations. Elles peuvent donc être calculées à l'aide des techniques de calcul des premiers impliqués. Nous avons montré qu'une de ces techniques, les arbres d'énumération d'ensembles [Rymon, 1992; Rymon, 1993], est bien appropriée au calcul des généralisations confirmatoires dans un langage attribut-valeur.

La partie dédiée à la classification commence par une présentation de l'apprentissage à partir d'exemples. Nous avons explicité les biais, ou critères de préférence, nécessaires à la définition d'une technique de classification. Nous nous sommes attachés à séparer clairement les biais logique et numérique mis en œuvre dans la construction d'un classifieur. Nous avons étudié en particulier les classifieurs à base de règles cohérentes et pertinentes pour l'ensemble d'apprentissage, car ils réalisent une séparation entre les biais logiques de cohérence et de pertinence, et le biais numérique représenté par un critère de résolution. Nous avons montré que les règles minimales cohérentes et pertinentes pour un ensemble d'apprentissage ne sont

malheureusement pas ses généralisations confirmatoires. Cet ensemble ne peut donc pas être calculé par un arbre d'énumération d'ensembles, ou par une autre technique de calcul de premiers impliqués. Pour expliquer cette différence, nous avons explicité la sémantique des valeurs manquantes dans le cas des règles cohérentes et pertinentes et dans le cas des généralisations confirmatoires.

Nous avons proposé une technique de classification, la classification par portée, fondée sur l'ensemble de toutes les règles cohérentes et pertinentes pour l'ensemble d'apprentissage, mais qui permet de classer un nouvel individu à partir d'exemples, sans générer explicitement toutes ces règles. La classification par portée construit simplement la règle la plus spécifique couvrant o et e, et vérifie qu'elle est cohérente. Si tel est le cas, o appartient à la portée de e. La classification par portée ne construit aucune abstraction à partir des exemples. C'est donc une approche à base d'instances. Elle procède à l'aide de comparaisons entre individus. Elle repose néanmoins sur le biais inductif logique des classifieurs à base de règles. Elle constitue un point commun entre les approches à base d'instances, fondées ou non sur des distances, et les approches à base de règles, ou fondées sur des espaces des versions. La classification qu'elle effectue est « équivalente » d'un point de vue logique à celle des classifieurs à base de toutes les règles cohérentes et pertinentes, ou à l'espace des versions disjonctif. Elle bénéficie donc de certaines de leurs qualités dont une bonne résistance au bruit. Le biais logique de cohérence peut être modulé à l'aide des paramètres définis dans le cadre de l'espace des versions disjonctif par Michèle Sebag [Sebag, 1996]. Nous avons explicité les interprétations que prend le paramètre de généralité M suivant les points de vue de la classification par portée et des approches à base de règles. Nous avons en outre proposé une procédure de réglage automatique de ces paramètres. Nous avons proposé d'autres extensions de la classification par portée destinées à améliorer la prise en compte de données réelles, comme la gestion des incohérences ou la gestion des redondances. Nous avons également généralisé notre approche afin de pouvoir procéder à une classification à partir de règles, et non plus d'exemples seulement. Nous avons aussi proposé plusieurs stratégies de généralisation des exemples en règles. Nous avons évalué la classification par portée, ainsi que ses extensions, sur un grand nombre de bases de test usuelles de la communauté d'apprentissage automatique. Nous les avons comparées avec des programmes de référence des approches à base d'instances (PEBLS), des approches à base de quelques règles « choisies » (C4.5 et CN2) et des approches à base de toutes les règles cohérentes et pertinentes pour l'ensemble d'apprentissage (SE-Learn). Cette évaluation expérimentale a montré que la classification par portée à partir d'exemples seulement offre une meilleure précision et un temps d'exécution comparable à ceux des autres approches. Nous avons vérifié également que les approches à base de toutes les règles cohérentes et pertinentes, comme la classification par portée ou les arbres d'énumération d'ensembles (SE-Learn), obtiennent une meilleure précision que les approches à base de quelques règles ou d'un nombre constant de voisins. Nous avons appliqué la classification par portée à des problèmes de classification rencontrés dans l'équipe RFIA. La première étude sur un problème d'aide au diagnostic médical a confirmé que les techniques de classification usuelles, y compris la classification par portée, n'apportent pas une précision, et surtout une intelligibilité, suffisantes pour venir en aide aux médecins sur ce problème. La seconde étude dans le domaine de la reconnaissance de la parole a montré que tous les classifieurs utilisés font preuve d'une bonne précision, mais certains ont des temps d'exécution inadaptés à une reconnaissance en temps réel. Les difficultés rencontrées dans ces deux applications proviennent de ce que la représentation de ces problèmes et de leur solution n'est pas adaptée aux biais de représentation ou procéduraux des classifieurs étudiés. La comparaison effectuée par Ron Rymon sur un problème d'aide à la décision d'attribution de prêts bancaires a montré que la classification par portée a une meilleure précision que les arbres

de décision sur ce problème. Il reste encore de nombreux points à améliorer. La généralisation des exemples en règles n'a pas produit les gains escomptés en précision, en rapidité et en intelligibilité. Une première solution serait de généraliser davantage les règles construites dans le cas d'attributs linéraires. Cette mauvaise performance est aussi liée au réglage des paramètres de modulation de la cohérence. La procédure de réglage que nous avons ébauchée demande à être améliorée.

Nous avons présenté quelques réflexions sur des développements à apporter à la classification par portée. Une réduction de l'ensemble d'apprentissage à l'aide d'un filtrage des exemples « peu informatifs » et des exemples incorrects permettrait de réduire les temps et d'améliorer la précision de classification. Nous avons montré comment les pré-ordres utilisés par la classification par portée peuvent être spécialisés pour tenir compte d'un ordre sur les attributs, en particulier lorsque des événements représentant une séquence temporelle sont considérés. Des séquences d'observations sont en effet fréquentes dans des applications de surveillance médicale, ou dans le cas de *Gaston*, le robot mobile de l'équipe RFIA. Par ailleurs, nous avons montré que la prise en compte d'une théorie du domaine est possible dans l'état actuel de la classification par portée, mais au prix d'un développement des descriptions coûteux. Il serait préférable d'intégrer explicitement la prise en compte de la théorie du domaine dans la définition de la classification par portée. Enfin, nous avons mentionné la nécessité d'une évaluation de la pertinence des attributs et l'intérêt de l'induction constructive. La précision de notre algorithme par rapport à celles de C4.5, CN2 et PEBLS est semblable à celle obtenue par le programme RISE de Pedro Domingos [Domingos, 1996]. Il serait intéressant de comparer ces deux techniques entre elles. Michèle Sebag et Céline Rouveirol ont proposé une adaptation de l'espace des versions disjonctif à la logique des prédicats du premier ordre [Sebag et Rouveirol, 1997]. La classification par portée peut-elle s'adapter également à ce cadre? Dans le cadre de la programmation logique inductive, il faudrait mettre en œuvre le modèle de généralisation confirmatoire que nous avons proposé, et en particulier développer le processus de construction des hypothèses. Il faudrait également étendre ce modèle afin d'accepter des fonctions autre que les constantes.

En conclusion, nous avons proposé un modèle d'induction permettant d'inférer des lois générales reflétant des régularités présentes dans les observations. L'inférence de ces lois peut être utile dans de nombreux contextes : par exemple, en planification, pour apprendre les lois du domaine, et faciliter la représentation des actions en résolvant les problèmes du décor et de la ramification, ou encore, dans les bases de données, en inférant de contraintes d'intégrité. Notons qu'il est fréquent que ces règles n'aient pas à être explicitées. C'est davantage les conclusions qu'elles permettent d'obtenir que leur énoncé qui est intéressant. Dans ce contexte, la classification par portée peut être très utile puisqu'elle permet de prédire à partir d'observations les valeurs de n'importe quel attribut choisi comme attribut de classe. Sa rapidité de mise en œuvre permet d'envisager de compléter des descriptions d'individus sans avoir à apprendre un classifieur par attribut.

Annexe A

La logique des prédicats du premier ordre et la logique propositionnelle

Nous rappelons dans les paragraphes suivants les définitions principales de la logique des prédicats du premier ordre et de la logique propositionnelle. Les logiques mathématiques et leur utilisation dans la modélisation du raisonnement sont présentées en détail dans [Haton *et al.*, 1991a].

A.1 La logique des prédicats du premier ordre

Soit L un langage de la logique des prédicats du premier-ordre sans symboles de fonctions autres que les constantes. L est défini par un ensemble dénombrable L_P de symboles de prédicats (plus éventuellement le symbole d'égalité =) d'arité 0,1,2,..., par un ensemble dénombrable non-vide L_C de symboles de constantes, et par un ensemble dénombrable X de symboles de variables. Dans la représentation des connaissances que nous avons utilisée, les symboles de constantes représentent les individus du domaine considéré, et les symboles de prédicats représentent leurs propriétés. Les termes sont construits de la manière habituelle :

- Un symbole de constante est un terme ;
- Un symbole de variable est un terme ;
- Puisqu'il n'y a pas de symboles de fonctions, ce sont les seuls termes possibles.

Les formules sont construites en utilisant les connecteurs standard $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$ et les quantificateurs universel $\forall$ et existentiel $\exists$:

- $Vrai$ et $Faux$ sont des formules;
- Si P est un prédicat d'arité n et $t_1, t_2, \ldots, t_n$ sont n termes alors $P(t_1, t_2, \ldots, t_n)$ est une formule,
- Si F_1 et F_2 sont des formules alors $\neg F_1$, $F_1 \wedge F_2$, $F_1 \vee F_2$, $F_1 \Rightarrow F_2$ et $F_1 \Leftrightarrow F_2$ sont des formules;
- Si x est une variable et F une formule alors $\forall x, F$ et $\exists x, F$ sont des formules;
- Ce sont les seules formules.

Un ensemble fini de formules est identifié à la conjonction des formules qu'il contient. $P(t_1, t_2, \ldots, t_n)$ est une formule *atomique* ou *atome*. Un atome est un *littéral* positif et la négation d'un atome est un littéral négatif. La *portée* d'une quantification, par exemple $\forall x$, dans la formule $\forall x, F$ est F. Une occurrence d'une variable dans une formule est *liée* si elle est immédiatement après un quantificateur, ou si elle se trouve dans la portée d'un quantificateur de la même variable. Toute occurrence non liée est *libre*. Une substitution σ est une application dont l'ensemble de départ est l'ensemble des variables sur lesquelles les formules sont construites, l'ensemble d'arrivée est l'ensemble des termes correspondants et σ est l'identité sauf pour un ensemble fini de variables appelé domaine de la substitution. Appliquer une substitution à une clause remplace toutes les occurrences des variables appartenant au domaine de la substitution par les termes correspondants. Une formule est *terminale* quand elle ne contient pas de variable. Une formule est *close* quand elle ne contient pas de variable libre. Une *clause* est une disjonction finie de littéraux dont les variables sont implicitement universellement quantifiées. Une clause *de Horn* est une clause contenant au plus un littéral positif. Une clause *définie* est une clause contenant exactement un littéral positif. L'*Univers de Herbrand* ou *domaine de Herbrand* d'un langage des prédicats du premier ordre sans symbole de fonctions est l'ensemble des termes terminaux, i.e. l'ensemble des constantes. Sa *base de Herbrand* est l'ensemble des atomes terminaux.

$BOOL =_d \{Vrai, Faux\}$ est l'ensemble des valeurs de vérité. Une *interprétation* I (sur L) est un couple $\langle D_I, M_I \rangle$, où D_I est un ensemble non-vide et M_I est une application qui associe un élément de D_I à chaque symbole de constante de L_C et une fonction d'arité n de D_I^n dans $BOOL$ à chaque prédicat d'arité n de L_P. Une *assignation* ou *valuation* v pour une interprétation I est une application de X dans D_I. Soit D_I^X l'ensemble de toutes les valuations sur D_I. La *sémantique* d'une formule A dans une interprétation I est notée $[\,A\,](I)$; c'est une application de D_I^X dans $BOOL$. La sémantique d'une formule A dans une interprétation I étant donné une valuation v pour I appartient à $BOOL$ et est noté $[\,A\,](I)(v)$. Sa valeur est définie par les règles de composition usuelles :

- Si t est un symbole de constante, $[\,t\,](I)(v) = [\,t\,](I)$;
- Si t est un symbole de variable, $[\,t\,](I)(v) = v(t)$;
- $[\,Vrai\,](I)(v) = Vrai$; $[\,Faux\,](I)(v) = Faux$
- Si F est un atome $P(t_1, \ldots, t_n)$, $[\,F\,](I)(v) = [\,P\,](I)([\,t_1\,](I)(v), \ldots, [\,t_n\,](I)(v))$
- $[\,\neg F\,](I)(v) = \neg[\,F\,](I)(v)$;
- $[\,F_1 \wedge F_2\,](I)(v) = [\,F_1\,](I)(v) \wedge [\,F_2\,](I)(v)$;
- $[\,F_1 \vee F_2\,](I)(v) = [\,F_1\,](I)(v) \vee [\,F_2\,](I)(v)$;
- $[\,F_1 \Rightarrow F_2\,](I)(v) = \neg[\,F_1\,](I)(v) \vee [\,F_2\,](I)(v)$;
- $[\,F_1 \Leftrightarrow F_2\,](I)(v) = [\,(F_1 \Rightarrow F_2) \wedge (F_2 \Rightarrow F_1)\,](I)(v)$;
- $[\,\forall x, F\,](I)(v) = Vrai$ si $[\,F\,](I)(v_{x \to d}) = Vrai$ pour tout $d \in D_I$, et $Faux$ sinon, où $v_{x \to d}$ est coïncide avec l'application v sauf en x et $v_{x \to d}(x) =_d d$;
- $[\,\exists x, F\,](I)(v) = [\,\neg\forall x, \neg F\,](I)(v)$.

Une interprétation I étant donné une valuation v (pour I) est un *modèle* de la formule A si et seulement si $[\,A\,](I)(v) = Vrai$; A est dite *satisfaite par* I. Si une formule A a un modèle, elle est dite *cohérente*, *incohérente* sinon. Quand tout modèle de A étant donnée une valuation v quelconque est une modèle de B étant donné v, B est une conséquence logique de A, noté $A \models B$. Si A est vide, B est une *tautologie*, noté $\models B$. B est une *contradiction* si $\neg B$ est une tautologie. Quand $[\,A\,](I)(v) = [\,B\,](I)(v)$ pour chaque interprétation I étant donné n'importe quelle valuation v pour I, A et B sont dits *équivalents*; noté $A \equiv B$. Si le langage L contient l'égalité, la sémantique de ce symbole de prédicat est contrainte dans toute interprétation : elle doit coïncider avec l'identité dans le domaine correspondant. Une interprétation *de Herbrand* est une interprétation $\langle D_I, M_I \rangle$ où D_I est l'univers de Herbrand et M_I associe chaque symbole de constante à lui-même. Ainsi, les interprétations de Herbrand I diffèrent seulement dans les valeurs de vérité associées par I aux symboles de prédicats. En conséquence, une interprétation de Herbrand est complètement définie par, et est habituellement identifiée avec, l'ensemble des atomes terminaux prenant la valeur de vérité $Vrai$ dans I. L'interprétation minimale de Herbrand d'une théorie T composée de clauses définies est notée $\mathcal{M}^+(T)$.

A.2 La logique propositionnelle

Un langage propositionnel est un cas particulier de langage des prédicats, construit uniquement avec des prédicats d'arité 0, appelés symboles propositionnels ou atomes propositionnels. Les formules propositionnelles sont donc définies inductivement par :

- Un symbole de proposition est une formule,
- Si F_1 et F_2 sont des formules alors $\neg F_1$, $F_1 \wedge F_2$, $F_1 \vee F_2$, $F_1 \Rightarrow F_2$ et $F_1 \Leftrightarrow F_2$ sont des formules
- Ce sont les seules formules.

Une interprétation propositionnelle I associe à chaque proposition une valeur de vérité. La valeur d'une formule est définie selon les règles de composition usuelles :

- $[\,\neg F\,](I) = \neg[\,F\,](I)$;
- $[\,F_1 \wedge F_2\,](I) = [\,F_1\,](I) \wedge [\,F_2\,](I)$;
- $[\,F_1 \vee F_2\,](I) = [\,F_1\,](I) \vee [\,F_2\,](I)$;
- $[\,F_1 \Rightarrow F_2\,](I) = \neg[\,F_1\,](I) \vee [\,F_2\,](I)$;
- $[\,F_1 \Leftrightarrow F_2\,](I) = [\,(F_1 \Rightarrow F_2) \wedge (F_2 \Rightarrow F_1)\,](I)$;

Annexe B

Les bases de tests usuelles en apprentissage automatique

Nous détaillons dans les paragraphes suivants les résultats de l'évaluation expérimentale que nous avons conduite sur des bases de tests usuelles dans la communauté d'apprentissage automatique. Ces bases sont disponibles à l'Université d'Irvine en Californie [Merz et Murphy, 1996].

Nous rappelons pour chaque base de test :

- ses caractéristiques : le nombre total d'exemples qu'elle contient ($|E|$), le nombre de classes, si elle contient des exemples incomplètement décrits et des exemples incohérents entre eux. Nous précisons enfin le nombre d'attributs et leur répartition en attributs nominaux, linéaires discrets et linéaires continus ;

- les précisions obtenues par les différentes techniques de classification utilisées (Pourcentage d'exemples de test Correctement Classés) : les programmes considérés sont la classification par portée à partir d'exemples seulement (cons) et à partir d'exemples généralisés en règles, en conservant les règles les plus générales (gen) ou les règles les plus générales cohérentes entre elles (gen_cons), PEBLS [Cost et Salzberg, 1993], C4.5 [Quinlan, 1993a], CN2 [Clark et Niblett, 1989; Clark et Boswell, 1991] et les arbres d'énumération d'ensembles sans élagage (SE) et avec élagage statistique (SE 0.05) ;

- le temps d'exécution global, plus précisément un dixième de ce temps, et l'écart-type du temps de résolution d'un des dix problèmes résolus lors de la validation croisée en dix sous-ensembles pour chacune des techniques précédemment décrites ;

- les paramètres de la classification par portée : la taille moyenne d'un ensemble d'apprentissage ($|EA|$) après prise en compte des exemples redondants lors de la validation croisée, le temps d'exécution (Temps) de la procédure de réglage automatique du critère de résolution (CR), vote majoritaire simple (S) ou quadratique (C), du paramètre de cohérence (ϵ) en pourcentage et du paramètre de généralité (M), la taille de l'ensemble d'apprentissage après généralisation des exemples en règles pour gen ($|gen(EA)|$) et gen_cons ($|gen\_cons(EA)|$) avec entre parenthèses le nombre de règles générées.

Les tirets (-) représentent les cas où la complexité de SE-Learn ou de la classification par portée avec généralisation des exemples en règles les rendent trop longs (de l'ordre de dix jours) à mettre en œuvre.

B.1 Audiology (AD)

Cette base a été fournie par le Professeur Jergen du Baylor College of Medicine. Elle concerne des diagnostics de troubles auditifs.

Caractéristiques

$\|E\|$	: 200	Classes	: 24	Valeurs manquantes	: Oui	Incohérences	: Non
Attributs	: 69	Nominaux	: 69	Linéaires discrets	: 0	Linéaires continus	: 0

Elle est décrite par 69 attributs nominaux, principalement des attributs booléens. Est-ce que ça bourdonne? Est-ce que vous avez des vertiges? ...

PCC					Défaut	: 18.0±9.7
cons	: 66.5±11.0	gen	: 59.0±11.7	gen_cons	: 61.5±8.8	
PEBLS	: 71.5±9.6	C4.5	: 70.3±10.3	CN2	: 77.0±12.0	
SE	: -	SE 0.05	: -			

La précision de la classification par portée est en dernière position par rapport à celles obtenues avec les autres classifieurs. L'écart n'est toutefois pas plus important que celui qui sépare les autres résultats. La généralisation des règles donne des résultats encore moins bons, mais sans que l'écart soit plus grand que les précédents. Notons également que le classifieur par défaut a un score nettement plus faible que ceux offerts par les autres classifieurs étant donné le nombre important de classes.

Temps

cons	: 3.32±0.06	gen	: 19.93±1.96	gen_cons	: 21.98±0.66
PEBLS	: 13.73±0.01	C4.5	: 9.67±0.73	CN2	: 22.59±1.61
SE	: -	SE 0.05	: -		

La classification par portée sans généralisation des exemples en règles fait preuve de la plus grande rapidité d'exécution. Remarquons que les temps d'exécution des classifieurs par portée avec généralisation sont du même ordre que ceux de PEBLS ou CN2. Cela vient de la taille de $gen(EA)$ et de $gen\_cons(EA)$ qui restent du même ordre que l'ensemble d'apprentissage.

Paramètres de la classification par portée

$\|EA\|$: 161.2±1.6	Temps : 22.30	Critère : C	Cohérence (ϵ) : 0	Généralité (M) : 2
$\|gen(EA)\|$: 267.8±13.4 (250.3±14.1)		$\|gen\_cons(EA)\|$: 194.3±5.4 (153.6±4.4)		

Le temps de paramétrage est toujours considérable, surtout pour le taux de cohérence ϵ et la spécificité M retenus. Notons que, curieusement, M ne vaut que 2 sur une base décrite par 69 attributs. C'est assez inhabituel et cela tendrait à montrer que tous les attributs sont pertinents.

B.2 Annealing (AN)

C'est une base concernant le recuit dans le domaine de la sidérurgie.

Caractéristiques

$\|E\|$	: 798	Classes	: 6	Valeurs manquantes	: Oui	Incohérences	: Non
Attributs	: 38	Nominaux	: 29	Linéaires discrets	: 6	Linéaires continus	: 3

Les attributs représentent le taux de carbone, la température, le type des matériaux, leur forme, ... Remarquons qu'elle comporte tous les types d'attributs.

PCC				Défaut	: 76.2±3.9
cons	: 94.2±2.8	gen	: 86.0±3.5	gen_cons	: 89.3±2.3
PEBLS	: 99.2±1.0	C4.5	: 93.1±1.7	CN2	: 85.7±3.6
SE	: -	SE 0.05	: -		

cons se classe second, et *gen* et *gen_cons* se placent entre C4.5 et CN2. Notons le score important du classifieur par défaut, alors qu'il y a six classes possibles. Cela indique clairement que les classes ne sont pas représentées par un même nombre d'exemples dans l'ensemble d'apprentissage.

Temps					
cons	: 1:21±0.47	gen	: 14:21±44.19	gen_cons	: 8:06±9.81
PEBLS	: 53.91±0.26	C4.5	: 29.53±2.78	CN2	: 16.63±0.86
SE	: -	SE 0.05	: -		

La base comportant 798 exemples décrits par 38 attributs, l'écart entre les temps d'exécution est net. La généralisation des exemples conduit clairement à un trop grand nombre de règles, et donc à un temps d'exécution trop important.

Paramètres de la classification par portée				
$\|EA\|$: 711.7±0.9	Temps : 11:51	Critère : C	Cohérence (ϵ) : 0	Généralité : 2
$\|gen(EA)\|$: 3629.7±44.9 (3626.4±45.0)			$\|gen\_cons(EA)\|$: 1421.2±24.3 (1171.4±26.7)	

Le temps de réglage est beaucoup trop long pour obtenir des paramètres aussi standard. Le score très élevé du classifieur par défaut a ralenti l'obtention d'un maximum « global ».

B.3 Credit (CE)

Cette base concerne l'attribution de cartes de crédit. Sa source est confidentielle et les attributs et leurs valeurs ont été remplacés par des symboles sans signification pour préserver la confidentialité des données.

Caractéristiques							
$\|E\|$	: 690	Classes	: 2	Valeurs manquantes	: Oui	Incohérences	: Non
Attributs	: 15	Nominaux	: 9	Linéaires discrets	: 0	Linéaires continus	: 6

Cette base comporte un nombre assez élevé d'individus par rapport au nombre d'attributs. Les attributs sont continus et nominaux. Certains attributs nominaux prennent une dizaine de valeurs, alors que d'autres sont binaires.

PCC				Défaut	: 55.5±4.7
cons	: 86.2±5.1	gen	: -	gen_cons	: -
PEBLS	: 83.0±5.1	C4.5	: 83.6±5.0	CN2	: 82.6±4.6
SE	: 80.2±28.6	SE 0.05	: 89.0±5.0		

cons obtient le deuxième meilleur score derrière SE-Learn avec élagage. La généralisation des règles n'a pas été utilisée car lors du premier essai, *gen* a généré 46164 règles et n'avait pas fini de classer 69 exemples test au bout de 39 heures CPU sur une SPARC Ultra 1.

Temps

cons	: 36.99±1.01	gen	: -	gen_cons	: -
PEBLS	: 22.82±0.05	C4.5	: 22.78±2.34	CN2	: 6.01±0.25
SE	: 23.93±1.57	SE 0.05	: 8.28±0.84		

Les temps de classification se divisent en deux groupes. Remarquons que SE-Learn avec élagage se classe second derrière CN2.

Paramètres de la classification par portée

$\|EA\|$: 621.0±0.0	Temps : 2:27	Critère : C	Cohérence (ϵ) : 0	Généralité (M) : 2
$\|gen(EA)\|$: - -		$\|gen\_cons(EA)\|$: - -		

Le temps de réglage semble long pour obtenir des paramètres aussi standard, mais la base contient un nombre important d'exemples et l'espace de recherche en (ϵ, M) a dû être parcouru quasiment en entier, car les maxima $max_{M \in [1:d]} v(\epsilon, M)$ (cf. paragraphe 7.2.3) entre deux valeurs de ϵ sont proches et leur calcul nécessite de faire varier M sur une grande partie de $[1; d]$.

B.4 Pima Diabetes (DI)

Cette base décrit un ensemble de femmes appartenant à un certain groupe d'indiens, *Pima Indians*. La classe indique si le test de diabète est positif pour celles-ci.

Caractéristiques

$\|E\|$	: 768	Classes	: 2	Valeurs manquantes	: Non	Incohérences	: Non
Attributs	: 8	Nominaux	: 0	Linéaires discrets	: 0	Linéaires continus	: 8

Les huit attributs sont tous linéaires continus. Ils indiquent le nombre de grossesses, la concentration en glucose du plasma après un test, la pression sanguine, ...

PCC				Défaut	: 65.0±3.8
cons	: 74.4±5.2	gen	: -	gen_cons	: -
PEBLS	: 73.7±4.4	C4.5	: 73.1±2.5	CN2	: 72.1±3.8
SE	: 73.5±1.9	SE 0.05	: 76.5±7.9		

Toutes les techniques autres que le classifieur par défaut obtiennent des précisions semblables. Les écarts entre les précisions ne sont pas significatifs.

Les techniques de généralisation des règles n'ont pas été évaluées sur cette base car elles ont un temps d'exécution trop long. Elles génèrent, en effet, un très grand nombre de règles, 22613 en ce qui concerne gen, et 17573 en ce qui concerne gen_cons.

Temps

cons	: 20.63±0.32	gen	: -	gen_cons	: -
PEBLS	: 21.22±0.04	C4.5	: 1:20±4.96	CN2	: 5.91±0.68
SE	: 1:16±1.61	SE 0.05	: 3.66±0.34		

Paramètres de la classification par portée

$\|EA\|$: 691.2±0.4	Temps : 1:27	Critère : S	Cohérence (ϵ) : 0	Généralité (M) : 3
$\|gen(EA)\|$: - -		$\|gen\_cons(EA)\|$: - -		

B.5 Echocardiogram (EC)

Cette base décrit des patients ayant eu un infarctus du myocarde dans le passé.

Caractéristiques							
$\|E\|$	: 131	Classes	: 2	Valeurs manquantes	: Oui	Incohérences	: Oui
Attributs	: 6	Nominaux	: 1	Linéaires discrets	: 0	Linéaires continus	: 5

La classe indique si le patient est encore en vie un an après son attaque. Les 6 attributs désignent l'âge du patients et des caractéristiques cardiaques. La composition de la base ainsi que les résultats obtenus par le programme EACH sont donnés par Steven Salzberg [Salzberg, 1991]. C'est d'ailleurs Steven Salzberg qui a déposé cette base dans le répertoire d'Irvine.

PCC					
				Défaut	: 67.0±14.6
cons	: 67.8±14.5	gen	: 67.0±14.6	gen_cons	: 70.8±15.8
PEBLS	: 63.3±15.4	C4.5	: 67.9±12.3	CN2	: 70.1±10.0
SE	: 67.8±11.5	SE 0.05	: 68.5±13.4		

La classification par portée avec généralisation des exemples en règles obtient de bonnes précision sur ce problème. Nous pouvons remarquer d'ailleurs que le nombre de règles retenues par gen_cons est nettement inférieur à l'ensemble de toutes les règles utilisées par gen. Ce nombre est du même ordre de grandeur que le nombre d'exemples initial, donc ne permet pas d'accélérer la classification. Toutefois, gen_cons offre une meilleure précision que la classification par portée à partir d'exemples seulement.

Temps					
cons	: 0.47±0.02	gen	: 1:57±1:40	gen_cons	: 13.00±0.49
PEBLS	: 0.85±0.01	C4.5	: 3.46±0.53	CN2	: 0.80±0.06
SE	: 0.56±0.02	SE 0.05	: 0.37±0.02		

Paramètres de la classification par portée				
$\|EA\|$: 117.9±0.3	Temps : 0.71	Critère : S	Cohérence (ϵ) : 10	Généralité (M) : 2
$\|gen(EA)\|$: 1938.9±43.9 (1938.9±43.9)			$\|gen\_cons(EA)\|$: 255.1±49.2 (186.9±50.0)	

B.6 Glass (GL)

Ce problème de classification consiste à identifier un type de verre. Cette étude est motivée par des besoins d'enquêtes criminelles, les débris de verre présents sur les lieux du crime pouvant fournir de précieuses indications.

Caractéristiques							
$\|E\|$	: 214	Classes	: 7	Valeurs manquantes	: Non	Incohérences	: Non
Attributs	: 9	Nominaux	: 0	Linéaires discrets	: 0	Linéaires continus	: 9

Les sept classes sont deux types de vitres d'habitations, deux types de vitres automobiles, les récipients en verre, la vaisselle et les ampoules électriques. Les neuf attributs linéaires continus décrivent la composition chimique du verre (Sodium, Magnésium, Aluminium, Silicium, Potassium, Calcium, Baryum, Fer) et l'indice de réfraction.

PCC					
				Défaut	: 31.1±14.8
cons	: 73.8±12.8	gen	: 24.7±11.8	gen_cons	: 35.0±15.9
PEBLS	: 68.1±9.6	C4.5	: 62.9±6.2	CN2	: 64.0±5.9
SE	: 69.9±8.5	SE 0.05	: 58.2±15.0		

Temps					
cons	: 2.04±0.11	gen	: 9:58±46.67	gen_cons	: 61.44±14.41
PEBLS	: 2.16±0.06	C4.5	: 14.58±2.14	CN2	: 2.08±0.14
SE	: 3.83±0.26	SE 0.05	: 0.68±0.05		

La précision de la classification par portée à partir d'exemples seulement est bonne, contrairement à gen et gen_cons qui ont encore généré beaucoup trop de règles. Notons que SE-Learn avec élagage se positionne assez mal, mais c'est le plus rapide.

Paramètres de la classification par portée				
$\|EA\|$: 191.8±0.6	Temps : 4.12	Critère : S	Cohérence (ϵ) : 10	Généralité (M) : 6
$\|gen(EA)\|$: 2803.0±87.7 (2802.9±87.6)		$\|gen\_cons(EA)\|$: 805.8±33.6 (784.2±35.3)		

La valeur du seuil de spécificité M (6) est curieusement élevée par rapport au nombre d'attributs (9) et à la faible valeur de ϵ (10%).

B.7 Heart disease

Quatre bases traitent de diagnostic de troubles cardiaques. Ces bases proviennent des personnes et des institutions suivantes :

- Robert Detrano, Cleveland Clinic Foundation (cleveland)
- Andras Janosi, Hungarian Institute of Cardiology, Budapest (hungarian)
- William Steinbrunn, University Hospital, Zurich, Switzerland (switzerland)
- Robert Detrano, V.A. Medical Center, Long Beach, CA (va)

Toutes ces bases ont le même format. Tous les individus sont décrits par les mêmes attributs. Ces attributs indiquent l'âge de la personne, son sexe, sa pression sanguine, son taux de cholestérol, etc. La classe indique si la personne a des troubles cardiaques.

B.7.1 Cleveland (HDc)

Caractéristiques							
$\|E\|$	: 302	Classes	: 2	Valeurs manquantes	: Oui	Incohérences	: Non
Attributs	: 13	Nominaux	: 6	Linéaires discrets	: 4	Linéaires continus	: 3

PCC					
				Défaut	: 53.9±6.6
cons	: 84.1±8.0	gen	: 53.9±6.6	gen_cons	: 68.8±9.7
PEBLS	: 81.9±6.6	C4.5	: 75.5±4.0	CN2	: 77.8±6.0
SE	: 83.4±6.7	SE 0.05	: 88.8±7.0		

Les précisions obtenues sont bonnes dans l'ensemble.

Temps		
cons : 10.57±0.11	gen : 5h58±10:44	gen_cons : 32:19±3:04
PEBLS : 4.78±0.04	C4.5 : 15.07±1.46	CN2 : 2.08±0.12
SE : 59.62±2.87	SE 0.05 : 1.49±0.35	

Paramètres de la classification par portée				
$\|EA\|$: 271.8±0.4	Temps : 30.33	Critère : S	Cohérence (ϵ) : 20	Généralité (M) : 7
$\|gen(EA)\|$: 12506.0±173.3 (12506.0±173.3)			$\|gen\_cons(EA)\|$: 4377.8±169.0 (4353.8±170.8)	

Les paramètres ϵ et M prennent des valeurs élevées. Elles ne paraissent cependant pas anormales. Les temps d'exécution de la classification par portée avec généralisation des exemples en règles sont conséquents. gen génère en particulier un grand nombre de règles.

B.7.2 Hungarian (HDh)

Les résultats sont comparables à ceux de la base précédente.

Caractéristiques							
$\|E\|$	: 293	Classes	: 2	Valeurs manquantes : Oui		Incohérences	: Non
Attributs : 13		Nominaux : 6		Linéaires discrets	: 4	Linéaires continus :	3

PCC			Défaut : 64.1±10.5
cons : 83.2±7.4	gen : 64.8±10.8	gen_cons : 80.8±9.4	
PEBLS : 75.9±7.7	C4.5 : 77.6±4.1	CN2 : 78.1±7.6	
SE : 80.4±10.8	SE 0.05 : 87.1±7.1		

Temps		
cons : 13.05±0.19	gen : 2h34±10:31	gen_cons : 25:14±2:49
PEBLS : 3.91±0.02	C4.5 : 11.52±1.03	CN2 : 2.16±0.17
SE : 15.06±0.63	SE 0.05 : 1.90±0.22	

Paramètres de la classification par portée				
$\|EA\|$: 262.9±0.7	Temps : 23.04	Critère : C	Cohérence (ϵ) : 10	Généralité (M) : 4
$\|gen(EA)\|$: 6721.1±196.2 (6721.1±196.2)			$\|gen\_cons(EA)\|$: 3732.2±167.3 (3687.9±169.1)	

B.7.3 Switzerland (HDs)

Caractéristiques							
$\|E\|$	: 122	Classes	: 2	Valeurs manquantes : Oui		Incohérences	: Non
Attributs : 13		Nominaux : 6		Linéaires discrets	: 4	Linéaires continus :	3

PCC					Défaut : 93.4±3.4
cons	: 93.4±3.4	gen	: 93.4±3.4	gen_cons	: 93.4±3.4
PEBLS	: 90.8±4.8	C4.5	: 90.9±2.0	CN2	: 92.6±2.5
SE	: 93.4±3.4	SE 0.05	: 82.1±19.0		

Cette base comporte un grand déséquilibre de l'ensemble d'apprentissage entre les deux classes. Seuls les trois variantes de la classification par portée, cons, gen et gen_cons, et les arbres d'énumération d'ensembles sans élagage, SE, parviennent à une précision égale à celle du classifieur par défaut. Tous les autres classifieurs obtiennent des précisions inférieures !

Temps					
cons	: 0.46±0.01	gen	: 1:32±6.57	gen_cons	: 1:02±6.33
PEBLS	: 0.66±0.01	C4.5	: 0.18±0.02	CN2	: 0.38±0.04
SE	: 0.87±0.07	SE 0.05	: 2.60±0.42		

Paramètres de la classification par portée				
$\|EA\|$: 109.8±0.4	Temps : 2.59	Critère : S	Cohérence (ϵ) : 0	Généralité (M) : 1
$\|gen(EA)\|$: 2393.3±94.2 (2393.2±94.1)			$\|gen\_cons(EA)\|$: 1517.4±118.9 (1509.6±118.5)	

B.7.4 V.A. Medical Center, Long Beach (HDv)

Caractéristiques							
$\|E\|$	: 199	Classes	: 2	Valeurs manquantes	: Oui	Incohérences	: Non
Attributs	: 13	Nominaux	: 6	Linéaires discrets	: 4	Linéaires continus	: 3

PCC					Défaut : 74.3±11.1
cons	: 75.8±11.9	gen	: 74.3±11.1	gen_cons	: 74.3±12.1
PEBLS	: 63.8±8.4	C4.5	: 75.1±12.9	CN2	: 74.8±11.1
SE	: 75.3±12.6	SE 0.05	: 79.0±13.1		

Les précisions obtenues ne sont guère supérieures à celle du classifieur par défaut. PEBLS obtient même une précision bien inférieure.

Temps					
cons	: 1.64±0.04	gen	: 21:26±51.84	gen_cons	: 1:35±7.47
PEBLS	: 1.66±0.04	C4.5	: 5.08±0.78	CN2	: 2.04±0.11
SE	: 8.90±0.83	SE 0.05	: 5.28±0.83		

Paramètres de la classification par portée				
$\|EA\|$: 178.3±0.5	Temps : 7.00	Critère : S	Cohérence (ϵ) : 0	Généralité (M) : 1
$\|gen(EA)\|$: 4470.5±116.3 (4470.5±116.3)			$\|gen\_cons(EA)\|$: 390.8±81.7 (304.7±90.4)	

B.8 Hepatitis (HE)

Cet ensemble de données décrit des patients souffrant d'hépatite. La classe est binaire. Elle indique si la personne est en vie ou décédée.

Caractéristiques

$\|E\|$	: 155	Classes	: 2	Valeurs manquantes	: Oui	Incohérences	: Non
Attributs	: 19	Nominaux	: 13	Linéaires discrets	: 0	Linéaires continus	: 6

Les attributs nominaux correspondent à la présence ou l'absence de symptômes. Les attributs numériques mesurent des paramètres physiologiques.

PCC				Défaut	: 79.3±14.2
cons	: 79.9±13.9	gen	: 79.3±14.2	gen_cons	: 78.7±15.1
PEBLS	: 81.8±10.1	C4.5	: 81.6±10.1	CN2	: 81.2±11.1
SE	: -	SE 0.05	: -		

Les classifieurs n'obtiennent pas de précision significativement différente de celle du classifieur par défaut. Les deux classes sont inégalement réparties.

Temps

cons	: 1.62±0.07	gen	: 18:29±59.02	gen_cons	: 11:53±1:26
PEBLS	: 1.85±0.01	C4.5	: 2.52±0.25	CN2	: 2.04±0.16
SE	: -	SE 0.05	: -		

Paramètres de la classification par portée

$\|EA\|$: 139.5±0.5	Temps : 5.46	Critère : C	Cohérence (ϵ) : 0	Généralité (M) : 3
$\|gen(EA)\|$: 4544.0±176.3 (4544.0±176.3)		$\|gen\_cons(EA)\|$: 3573.2±249.6 (3567.4±249.7)		

B.9 Horse colic (HO)

Cette base concerne les coliques des chevaux. La classe indique si le cheval est mort ou non.

Caractéristiques

$\|E\|$	: 368	Classes	: 2	Valeurs manquantes	: Oui	Incohérences	: Oui
Attributs	: 21	Nominaux	: 11	Linéaires discrets	: 3	Linéaires continus	: 7

Les attributs nominaux indiquent si le cheval a été opéré, s'il est jeune, ... Les attributs linéaires mesurent sa température, son rythme de respiration, etc.

PCC				Défaut	: 63.0±7.6
cons	: 82.8±5.5	gen	: 75.8±6.2	gen_cons	: 75.8±7.4
PEBLS	: 81.6±5.0	C4.5	: 83.8±4.8	CN2	: 83.4±5.5
SE	: -	SE 0.05	: -		

Les résultats des classifieurs sont assez bons dans l'ensemble.

Temps

cons	: 5.50±0.14	gen	: 1h03±17:42	gen_cons	: 11:01±1:04
PEBLS	: 5.81±0.03	C4.5	: 7.76±1.05	CN2	: 11.46±0.97
SE	: -	SE 0.05	: -		

Paramètres de la classification par portée

$\|EA\|$: 320.6±0.9	Temps : 34.82	Critère : C	Cohérence (ϵ) : 0	Généralité (M) : 3
$\|gen(EA)\|$: 10291.3±256.5 (10291.3±256.5)			$\|gen\_cons(EA)\|$: 3441.9±196.6 (3410±197.6)	

Le nombre de règles générées et donc le temps d'exécution de la classification par portée avec généralisation des exemples sont encore trop grands.

B.10 Iris (IR)

Cette base décrit 150 exemples de fleurs de la famille des iris. Cette base est très connue dans le domaine de la reconnaissance des formes. R. A. Fisher l'a collectée et utilisée dès 1936 [Fisher, 1936]. Chaque exemple est décrit par quatre attributs linéraires continus mesurant la longueur et la largeur des sépales, et la longueur et la largeur des pétales. La classe couvre les trois variétés : *Iris Virginica, Iris Setosa*, et *Iris Versicolor*.

Caractéristiques

$\|E\|$: 150	Classes : 3	Valeurs manquantes : Non	Incohérences : Non
Attributs : 4	Nominaux : 0	Linéaires discrets : 0	Linéaires continus : 4

PCC		Défaut : 21.3±5.2
cons : 94.6±6.1	gen : 86.0±15.5	gen_cons : 75.3±15.0
PEBLS : 95.1±5.5	C4.5 : 95.7±5.5	CN2 : 94.0±5.8
SE : 96.0±5.6	SE 0.05 : 90.5±7.2	

Les précisions obtenues sont très bonnes. De très nombreux classifieurs ont été utilisés sur cette base et ont obtenus également un faible taux d'erreur de classification. La précision de la classification par portée avec généralisation des exemples en règles est très significativement inférieure à la précision des autres classifieurs. Le nombre de règles engendrées semble en effet trop grand pour un problème décrit par seulement quatre attributs. La mauvaise performance obtenue par gen et gen_cons semble provenir d'une trop faible généralisation des règles dans le cas d'attributs linéaires continus.

Temps

cons : 0.42±0.01	gen : 7.07±1.04	gen_cons : 10.76±0.70
PEBLS : 0.87±0.01	C4.5 : 0.37±0.08	CN2 : 0.31±0.05
SE : 0.21±0.01	SE 0.05 : 0.19±0.01	

Paramètres de la classification par portée

$\|EA\|$: 132.5±0.5	Temps : 1.03	Critère : S	Cohérence (ϵ) : 0	Généralité (M) : 2
$\|gen(EA)\|$: 359.8±18.1 (359.1±18.3)			$\|gen\_cons(EA)\|$: 342.7±16.1 (341.2±15.9)	

B.11 Labor negociation (LA)

Cet ensemble de données représente toutes les conventions collectives du travail dans le secteur des services pour des entreprises de plus de 500 personnes (enseignement, hôpitaux, police, etc.) conclues au Canada en 1987 et au premier trimestre 1988. La classe indique les contrats

acceptables. Les exemples de contrats inacceptables proviennent d'interviews d'experts, ou sont des exemples inventés sous la forme de contre-exemples les plus proches (*near-misses*).

Caractéristiques							
$\|E\|$	: 57	Classes	: 2	Valeurs manquantes	: Oui	Incohérences	: Non
Attributs	: 16	Nominaux	: 8	Linéaires discrets	: 6	Linéaires continus	: 2

Les attributs désignent la durée de validité du contrat, les taux d'augmentation des salaires la première, deuxième et troisième année du contrat, le nombre d'heures hebdomadaires de travail, le type d'indemnité liée au coût de la vie, la cotisation de l'employeur à un plan de retraite, etc.

PCC					
				Défaut	: 64.3±29.4
cons	: 85.6±14.4	gen	: 64.3±29.4	gen_cons	: 76.6±25.3
PEBLS	: 85.2±15.1	C4.5	: 78.4±10.7	CN2	: 78.6±11.4
SE	: 89.0±20.4	SE 0.05	: 83.0±33.3		

Temps					
cons	: 7.57±0.01	gen	: 9.97±0.23	gen_cons	: 8.76±0.16
PEBLS	: 0.54±0.01	C4.5	: 0.38±0.11	CN2	: 0.77±0.08
SE	: 2.24±0.21	SE 0.05	: 3.28±0.28		

Paramètres de la classification par portée				
$\|EA\|$: 51.3±0.5	Temps : 1:15	Critère : C	Cohérence (ϵ) : 0	Généralité (M) : 2
$\|gen(EA)\|$: 374.0±23.3 (374.0±23.3)		$\|gen\_cons(EA)\|$: 223.0±27.1 (219.9±27.8)		

B.12 Lung cancer (LC)

Ce problème comporte trois classes, 56 attributs et seulement 33 exemples. La signification des attributs et l'origine des données ne sont pas connues. Ce sont des attributs nominaux pouvant prendre trois valeurs.

Caractéristiques							
$\|E\|$	: 33	Classes	: 3	Valeurs manquantes	: Oui	Incohérences	: Non
Attributs	: 56	Nominaux	: 56	Linéaires discrets	: 0	Linéaires continus	: 0

PCC					
				Défaut	: 41.6±17.5
cons	: 53.3±19.3	gen	: 41.6±17.5	gen_cons	: 38.3±15.3
PEBLS	: 44.8±21.0	C4.5	: 57.7±29.9	CN2	: 28.3±22.2
SE	: -	SE 0.05	: -		

Nous pouvons remarquer que la précision de CN2 est nettement inférieure à celle du classifieur par défaut. Néanmoins les écart-types sont tels et le nombre d'exemples si petit qu'aucune différence entre les classifieurs n'est significative.

Temps		
cons : 9.12±0.03	gen : 12.44±0.07	gen_cons : 11.67±0.11
PEBLS : 0.74±0.02	C4.5 : 0.63±0.06	CN2 : 1.96±0.17
SE : -	SE 0.05 : -	

Paramètres de la classification par portée				
$\|EA\|$: 28.8±0.4	Temps : 1:29	Critère : S	Cohérence (ϵ) : 40	Généralité (M) : 16
$\|gen(EA)\|$: 127.9±3.1 (127.9±3.1)		$\|gen\_cons(EA)\|$: 31.5±3.0 (21.8±5.7)		

B.13 Liver Disease (LD)

Cette base comporte 344 exemples correspondant à la description d'individus mâles décrits par cinq tests sanguins qui sont censés être sensibles à des troubles du foie provenant d'une consommation d'alcool excessive. La classe indique simplement si l'individu boit plus de l'équivalent de 1,25 litres de boissons alcoolisées par jour.

Caractéristiques			
$\|E\|$: 344	Classes : 2	Valeurs manquantes : Non	Incohérences : Non
Attributs : 5	Nominaux : 0	Linéaires discrets : 0	Linéaires continus : 5

PCC			Défaut : 74.6±8.8
cons : 79.0±8.1	gen : 70.0±10.5		gen_cons : 76.6±7.0
PEBLS : 72.0±6.3	C4.5 : 74.7±5.5		CN2 : 78.1±5.9
SE : 78.4±8.5	SE 0.05 : 81.4±7.7		

Les deux classes ne sont pas également représentées. La précision du classifieur par défaut est élevée. La précision des autres classifieurs est sensiblement voisine de celle du classifieur par défaut. Nous pouvons remarquer que, cette fois, c'est PEBLS qui obtient une moins bonne précision que le classifieur par défaut. La classification par portée et les arbres d'énumération d'ensembles obtiennent les meilleurs résultats.

Temps		
cons : 1.66±0.05	gen : 22:04±2:37	gen_cons : 14:45±2:03
PEBLS : 3.53±0.05	C4.5 : 18.84±1.46	CN2 : 1.27±0.07
SE : 1.26±0.05	SE 0.05 : 0.48±0.01	

Paramètres de la classification par portée				
$\|EA\|$: 306.3±0.8	Temps : 4.29	Critère : S	Cohérence (ϵ) : 0	Généralité (M) : 2
$\|gen(EA)\|$: 5797.6±254.8 (5794.9±254.3)		$\|gen\_cons(EA)\|$: 4750.1±325.8 (4725.6±327.9)		

B.14 Lenses (LE)

Cet ensemble de données concerne le port de lentilles de contact. La classe indique si le patient doit porter des lentilles de contact rigides, des lentilles souples, ou ne doit pas porter

de lentilles de contact. Les quatre attributs nominaux indiquent l'âge du patient (jeune, pré-presbyte, presbyte), le type de lunettes (myope ou hypermétrope), s'il est astigmate, et le taux de larmes (réduit ou normal). La base décrit l'ensemble des 24 descriptions d'individus possibles.

Caractéristiques

$\|E\|$	: 24	Classes	: 3	Valeurs manquantes	: Non	Incohérences	: Non
Attributs	: 4	Nominaux	: 4	Linéaires discrets	: 0	Linéaires continus	: 0

PCC					
				Défaut	: 63.3±26.9
cons	: 70.0±28.1	gen	: 81.6±25.3	gen_cons	: 75.0±28.5
PEBLS	: 72.0±31.0	C4.5	: 87.7±19.0	CN2	: 70.0±28.1
SE	: 70.0±28.1	SE 0.05	: 93.3±21.0		

Nous pouvons noter les bons résultats de la classification par portée avec généralisation des exemples en règles. Leurs précisions sont supérieures à celles de PEBLS, CN2 et la classification par portée à partir d'exemples seulement. Ces bons résultats sont à corréler avec le petit nombre de règles générées.

Temps

cons	: 0.13±0.01	gen	: 0.15±0.01	gen_cons	: 0.17±0.01
PEBLS	: 0.34±0.01	C4.5	: 0.07±0.01	CN2	: 0.04±0.01
SE	: 0.11±0.01	SE 0.05	: 0.10±0.01		

Paramètres de la classification par portée

$\|EA\|$: 21.6±0.5	Temps : 1.03	Critère : C	Cohérence (ϵ) : 10	Généralité (M) : 1
$\|gen(EA)\|$: 7.0±0.6 (7.0±0.6)		$\|gen\_cons(EA)\|$: 11.7±0.8 (1.3±0.5)		

Le nombre de règles générées par gen et gen_cons est plus petit que la taille de l'ensemble d'apprentissage. Ces règles offrent, en outre, une bonne intelligibilité. Parmi les règles générées figure la règle :

Si le patient n'a pas assez de larmes alors ne pas lui donner de lentilles de contact.

Notons, pour terminer, que neuf règles suffisent théoriquement à couvrir l'ensemble des cas.

B.15 LED (LI)

Ce problème consiste à reconnaître un chiffre à partir d'un affichage de sept diodes électroluminescentes (figure 7.1). Chacun des sept attributs correspond à une des diodes et indique si elle est allumée ou éteinte. Il ne s'agit pas d'apprendre par cœur la description des dix chiffres. En effet, la description des exemples d'apprentissage et de test est bruitée : la valeur de 10% des attributs a été inversée.

Caractéristiques

$\|E\|$	: 100	Classes	: 10	Valeurs manquantes	: Non	Incohérences	: Non
Attributs	: 7	Nominaux	: 7	Linéaires discrets	: 0	Linéaires continus	: 0

PCC					
				Défaut	: 18.0±13.9
cons	: 65.0±13.5	gen	: 60.0±15.6	gen_cons	: 64.0±13.4
PEBLS	: 52.0±10.4	C4.5	: 55.7±8.4	CN2	: 63.0±14.1
SE	: 67.0±23.2	SE 0.05	: 60.3±17.3		

La classification par portée avec généralisation des exemples en règles obtient de bons résultats, en accord avec le petit nombre de règles générées.

Temps					
cons	: 4.93±0.01	gen	: 5.85±0.02	gen_cons	: 5.84±0.02
PEBLS	: 0.94±0.01	C4.5	: 0.79±0.04	CN2	: 0.67±0.05
SE	: 0.46±0.03	SE 0.05	: 0.34±0.01		

Paramètres de la classification par portée				
$\|EA\|$: 34.9±1.6	Temps : 48.64	Critère : C	Cohérence (ϵ) : 0	Généralité (M) : 1
$\|gen(EA)\|$: 35.5±2.4 (22.6±2.1)		$\|gen\_cons(EA)\|$: 18.1±1.4 (12.9±1.4)		

Parmi les règles obtenues, on peut trouver la règle établissant que si le segment horizontal supérieur et le segment vertical inférieur gauche sont allumés et si le segment vertical supérieur droit est éteint alors c'est le chiffre 6, comme cela est représenté sur la figure B.1 (les lignes pointillées désignent des segments qui n'apparaissent pas dans la règle, ils peuvent être allumés ou éteints).

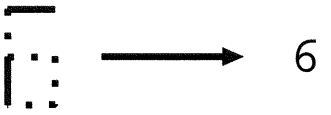


FIG. B.1 – *Une règle obtenue pour la reconnaissance du chiffre 6.*

B.16 Post-operative (PO)

La problème de classification consiste ici à déterminer l'endroit où des patients en salle de réanimation doivent être envoyés par la suite. Les trois classes considérées sont les soins intensifs, le retour dans une chambre d'hôpital, ou la sortie de l'hôpital. L'hypothermie est la principale préoccupation après une opération. Les attributs correspondent essentiellement à des mesures de températures de diverses parties du corps, ainsi que leur stabilité. Il est important de remarquer ici que, comme nous l'avons indiqué au paragraphe 10.2.4, les classifieurs que nous considérons ne savent pas prendre en compte des relations entre attributs. Il serait inutile ici de mentionner les températures précédentes, car les classifieurs ne sauraient pas calculer la différence entre deux attributs. Il faut donc ajouter explicitement des attributs codant la stabilité de la température au cours du temps.

Caractéristiques							
$\|E\|$	: 90	Classes	: 3	Valeurs manquantes	: Oui	Incohérences	: Oui
Attributs	: 8	Nominaux	: 0	Linéaires discrets	: 8	Linéaires continus	: 0

PCC					
				Défaut	: 71.1±11.9
cons	: 71.1±11.9	gen	: 71.1±11.9	gen_cons	: 64.4±15.5
PEBLS	: 56.5±19.7	C4.5	: 67.9±11.0	CN2	: 63.3±14.8
SE	: 63.7±14.1	SE 0.05	: 26.6±134.0		

Les précisions obtenues sont quasiment toutes strictement inférieures à celle du classifieur par défaut. Nous pouvons remarquer ici que la classification par portée est la seule à conserver une précision égale à celle du classifieur par défaut.

Temps					
cons	: 3.30±0.01	gen	: 5.44±0.27	gen_cons	: 4.71±0.07
PEBLS	: 0.63±0.01	C4.5	: 0.65±0.06	CN2	: 1.93±0.16
SE	: 0.59±0.02	SE 0.05	: 0.71±0.05		

Paramètres de la classification par portée				
$\|EA\|$: 67.7±1.1	Temps : 31.82	Critère : S	Cohérence (ϵ) : 10	Généralité (M) : 1
$\|gen(EA)\|$: 212.2±11.5 (210.4±11.5)			$\|gen\_cons(EA)\|$: 67.0±1.3 (6.9±3.4)	

B.17 DNA Promoters (PR)

Ce problème de classification concerne la biologie et consiste à prédire si une séquence de gènes d'ADN a une activité de « promoteur », c'est-à-dire si cette séquence initie le processus d'expression d'un gène. Les 57 attributs décrivent une séquence de nucléotides.

Caractéristiques							
$\|E\|$	: 106	Classes	: 2	Valeurs manquantes	: Non	Incohérences	: Non
Attributs	: 57	Nominaux	: 57	Linéaires discrets	: 0	Linéaires continus	: 0

PCC					
				Défaut	: 37.8±4.6
cons	: 81.9±12.6	gen	: 74.3±19.4	gen_cons	: 77.3±13.1
PEBLS	: 85.5±5.7	C4.5	: 83.8±7.7	CN2	: 66.2±20.7
SE	: -	SE 0.05	: -		

Les précisions obtenues sont plutôt bonnes. Néanmoins la précision moyenne de CN2 est très inférieure à celle des autres classifieurs, mais l'écart-type des précisions est important.

Temps					
cons	: 2.29±0.02	gen	: 39.72±2.15	gen_cons	: 19.73±1.07
PEBLS	: 3.27±0.01	C4.5	: 3.08±0.67	CN2	: 2.69±0.16
SE	: -	SE 0.05	: -		

Paramètres de la classification par portée				
$\|EA\|$: 95.4±0.5	Temps : 17.75	Critère : C	Cohérence (ϵ) : 0	Généralité (M) : 11
$\|gen(EA)\|$: 688.2±22.8 (688.2±22.8)			$\|gen\_cons(EA)\|$: 376.0±19.6 (374.9±19.8)	

B.18 Solar flare

Ces problèmes de classification concernent l'éclat du soleil. Chaque exemple représente les caractéristiques d'une région active du soleil. Trois types d'événements ont été mesurés déterminant ainsi trois attributs de classes possibles. Solar flare comporte donc trois problèmes de classification dans lesquels les exemples ont des descriptions identiques, mais des attributs de classe distincts. La valeur de la classe est le nombre de fois où un certain type d'éclat du soleil s'est produit durant une période de 24 heures. Le nombre de valeurs de la classe est 4 au maximum. Certains événements ne se sont produits que 1 à 2 fois. Le nombre d'exemples associées à l'observation d'aucun événement est beaucoup plus grand que ceux des autres valeurs de la classe. Il va donc être difficile de faire mieux que le classifieur par défaut.

B.18.1 Common flares (SFc)

Caractéristiques

$\|E\|$	: 323	Classes	: 4	Valeurs manquantes	: Non	Incohérences	: Oui
Attributs	: 10	Nominaux	: 6	Linéaires discrets	: 4	Linéaires continus	: 0

PCC					
				Défaut	: 88.8±3.9
cons	: 88.8±3.9	gen	: 88.8±3.9	gen_cons	: 87.6±4.7
PEBLS	: 83.3±5.6	C4.5	: 87.8±3.9	CN2	: 88.8±3.9
SE	: 86.3±4.9	SE 0.05	: 88.6±8.3		

L'ensemble des classifieurs obtiennent une précision inférieure ou égale à celle du classifieur par défaut. PEBLS obtient une précision sensiblement inférieure à celles des autres classifieurs. La classification par portée avec généralisation des exemples en règles se comporte bien. Le nombre de règles générées est effectivement intéressant puisqu'inférieur à la taille de l'ensemble d'apprentissage.

Temps

cons	: 1.74±0.11	gen	: 9.58±0.66	gen_cons	: 12.89±3.57
PEBLS	: 5.37±1.70	C4.5	: 2.70±1.15	CN2	: 2.91±0.92
SE	: 1.80±0.64	SE 0.05	: 2.61±0.60		

Paramètres de la classification par portée

$\|EA\|$: 147.1±1.4	Temps : 7.80	Critère : S	Cohérence (ϵ) : 10	Généralité (M) : 1
$\|gen(EA)\|$: 50.6±3.9 (50.5±4.0)		$\|gen\_cons(EA)\|$: 117.3±8.4 (27.5±6.7)		

B.18.2 Moderate flares (SFm)

Le comportement des divers classifieurs est sensiblement le même que sur la base précédente.

Caractéristiques

$\|E\|$	: 323	Classes	: 4	Valeurs manquantes	: Non	Incohérences	: Oui
Attributs	: 10	Nominaux	: 6	Linéaires discrets	: 4	Linéaires continus	: 0

PCC					
				Défaut	: 90.0±6.2
cons	: 90.0±6.2	gen	: 90.0±6.2	gen_cons	: 88.1±7.4
PEBLS	: 83.4±6.3	C4.5	: 88.3±6.0	CN2	: 89.1±6.8
SE	: 89.1±6.6	SE 0.05	: 92.2±6.1		

Temps					
cons	: 1.71±0.06	gen	: 8.50±0.38	gen_cons	: 11.04±0.40
PEBLS	: 4.47±0.06	C4.5	: 1.74±0.86	CN2	: 2.34±0.19
SE	: 1.42±0.05	SE 0.05	: 1.84±0.11		

Paramètres de la classification par portée

$\|EA\|$: 147.1±1.4	Temps : 8.08	Critère : S	Cohérence (ϵ) : 10	Généralité (M) : 1
$\|gen(EA)\|$: 43.9±9.0 (42.1±9.0)		$\|gen\_cons(EA)\|$: 112.3±9.8 (20.3±5.4)		

B.18.3 Severe flares (SFx)

Les événements considérés dans cette base sont plus rares que ceux considérés dans les deux bases précédentes. Près de 98% des exemples sont de classe 0, c'est-à-dire aucun éclat du type considéré n'a été observé durant une période de 24 heures.

Caractéristiques							
$\|E\|$	: 323	Classes	: 4	Valeurs manquantes	: Non	Incohérences	: Oui
Attributs	: 10	Nominaux	: 6	Linéaires discrets	: 4	Linéaires continus	: 0

PCC					
				Défaut	: 97.8±1.4
cons	: 97.8±1.4	gen	: 97.8±1.4	gen_cons	: 95.3±3.0
PEBLS	: 96.2±2.3	C4.5	: 97.8±1.4	CN2	: 97.5±1.2
SE	: 97.4±1.3	SE 0.05	: 47.1±35.6		

Nous pouvons constater que la précision de tous les classifieurs est inférieure ou égale à celle du classifieur par défaut, mais que plus la classe majoritaire est massivement représentée, plus les écart entre les classifieurs diminuent.

Temps					
cons	: 1.41±0.05	gen	: 5.26±0.13	gen_cons	: 8.11±0.33
PEBLS	: 4.52±0.02	C4.5	: 0.22±0.02	CN2	: 1.63±0.11
SE	: 0.65±0.04	SE 0.05	: 2.41±0.34		

Paramètres de la classification par portée

$\|EA\|$: 147.1±1.4	Temps : 7.73	Critère : S	Cohérence (ϵ) : 10	Généralité (M) : 1
$\|gen(EA)\|$: 11.5±2.4 (11.4±2.5)		$\|gen\_cons(EA)\|$: 46.5±21.2 (40.7±18.6)		

B.19 Soybean (SO)

Cet ensemble de données concerne les maladies du soja. Nous n'avons utilisé que la base intitulée soybean-small. Elle comporte 4 classes seulement. Les attributs décrivent le mois de l'année, la taille de la plante, les précipitations, la température, la grêle, etc.

Caractéristiques

$\|E\|$	: 48	Classes	: 4	Valeurs manquantes	: Non	Incohérences	: Non
Attributs	: 35	Nominaux	: 35	Linéaires discrets	: 0	Linéaires continus	: 0

PCC					
				Défaut	: 37.5±26.3
cons	: 100.0±0.0	gen	: 89.5±14.6	gen_cons	: 89.5±14.6
PEBLS	: 100.0±0.0	C4.5	: 96.8±7.4	CN2	: 97.5±7.9
SE	: 100.0±0.0	SE 0.05	: 100.0±0.0		

Les précisions sont étonnamment bonnes étant donnés le faible nombre d'exemples et le nombre d'attributs. La généralisation des exemples en règles produit un nombre de règles raisonnable. La mauvaise précision de gen et gen_cons pourrait peut-être être améliorée en adaptant les paramètres de modulation de la cohérence ϵ et M aux ensembles d'apprentissage après génération des règles.

Temps

cons	: 17.86±0.02	gen	: 18.44±0.02	gen_cons	: 18.04±0.04
PEBLS	: 0.94±0.02	C4.5	: 0.55±0.04	CN2	: 0.75±0.12
SE	: 21.57±2.31	SE 0.05	: 2.64±0.26		

Paramètres de la classification par portée

$\|EA\|$: 42.3±0.5	Temps : 2:57	Critère : S	Cohérence (ϵ) : 0	Généralité (M) : 2
$\|gen(EA)\|$: 21.5±5.6 (21.5±5.6)		$\|gen\_cons(EA)\|$: 21.5±5.6 (21.5±5.6)		

B.20 Splice junctions (SP)

Les *splice junctions* sont des points de séquences d'ADN où de l'ADN « superflu » est enlevé lors du processus de création de protéines dans les organismes évolués. Le problème posé dans cette base est de reconnaître, étant donné une séquence d'ADN, la frontière entre les exons (les parties de la séquence d'ADN conservées après la jonction) et les introns (les parties de la séquence d'ADN qui sont éliminées). Ce problème comporte deux sous-tâches : reconnaître les frontières intron/exon (les sites IE) et reconnaître les frontières exon/intron (les sites EI). Le problème consiste à, étant donné une position au milieu d'une fenêtre de 60 éléments d'une séquence d'ADN, décider si c'est une frontière :

- « intron ->exon » (appelée « donneur » dans la communauté biologie)
- « exon ->intron » (appelée « accepteur » dans la communauté biologie)
- ni l'un, ni l'autre.

Caractéristiques							
$\|E\|$	: 3190	Classes	: 3	Valeurs manquantes	: Oui	Incohérences	: Oui
Attributs	: 60	Nominaux	: 60	Linéaires discrets	: 0	Linéaires continus	: 0

Cet ensemble de données contient un grand nombre d'attributs et un grand nombre d'exemples. Nous n'avons donc pas mis en œuvre les arbres d'énumération d'ensembles, ni les techniques de généralisation des exemples en règles dans la classification par portée.

PCC					Défaut	: 51.8±2.3
cons	: 95.3±1.5	gen	: -		gen_cons	: -
PEBLS	: 93.9±0.7	C4.5	: 93.7±1.3		CN2	: 91.2±1.9
SE	: -	SE 0.05	: -			

La précision des classifieurs est bonne et assez homogène.

Temps						
cons	: 4h21±11.51	gen	: -		gen_cons	: -
PEBLS	: 4:12±9.26	C4.5	: 3:60±10.30		CN2	: 2:15±3.75
SE	: -	SE 0.05	: -			

Les temps des quatre classifieurs qui ont pu être appliqués à ce problème sont également homogènes.

Paramètres de la classification par portée					
$\|EA\|$: 2715.9±2.9	Temps : 41h49	Critère : C	Cohérence (ϵ) : 0	Généralité (M) : 9	
$\|gen(EA)\|$: -	-	$\|gen\_cons(EA)\|$: -	-		

Le temps de réglage de la classification par portée est long. Cela vient surtout du nombre d'attributs et donc du nombre de valeurs possibles de M. Les paramètres obtenus paraissent raisonnables.

B.21 Voting records (VO)

Cet ensemble de données contient les votes de chacun des membres du congrès américain sur les seize votes clés identifiés dans l'almanach trimestriel du congrès, 98-ième congrès, seconde session de 1984. Les valeurs des votes et donc des attributs indiquent, pour chacun des membres, s'il a voté plutôt pour, plutôt contre ou a décidé de ne pas prendre position. Cette dernière valeur est une valeur à part entière et non une valeur manquante. Les attributs considérés sont donc des votes clés tels que l'adoption du budget, l'aide au Salvador, l'aide au contras du Nicaragua, etc. La classe indique si le représentant au congrès est démocrate ou républicain.

Caractéristiques							
$\|E\|$	: 435	Classes	: 2	Valeurs manquantes	: Non	Incohérences	: Non
Attributs	: 16	Nominaux	: 16	Linéaires discrets	: 0	Linéaires continus	: 0

PCC					Défaut	: 61.3±6.9
cons	: 94.7±3.4	gen	: 94.4±1.9		gen_cons	: 88.2±5.9
PEBLS	: 94.3±3.1	C4.5	: 95.1±2.6		CN2	: 94.4±3.6
SE	: -	SE 0.05	: -			

La précision des classifieurs est bonne et très homogène. Seul gen_cons obtient une précision sensiblement inférieure à celle des autres classifieurs « évolués », mais très supérieure à celle du classifieur par défaut.

Temps		
cons : 6.66±0.27	gen : 50.64±4.09	gen_cons : 1:26±3.89
PEBLS : 12.31±0.21	C4.5 : 1.69±0.44	CN2 : 1.65±0.12
SE : -	SE 0.05 : -	

Paramètres de la classification par portée				
$\|EA\|$: 311.4±3.7	Temps : 31.86	Critère : S	Cohérence (ϵ) : 0	Généralité (M) : 1
$\|gen(EA)\|$: 704.8±69.1 (704.8±69.1)		$\|gen\_cons(EA)\|$: 586.4±43.9 (536.0±44.9)		

B.22 Wine (WI)

Ces données décrivent les résultats d'analyses chimiques de vins produits dans la même région d'Italie par trois viticulteurs différents.

Caractéristiques				
$\|E\|$: 178	Classes : 3	Valeurs manquantes : Non	Incohérences : Non	
Attributs : 13	Nominaux : 0	Linéaires discrets : 0	Linéaires continus : 13	

Les 13 attributs mesurent la quantité de 13 constituants présents dans les vins des trois viticulteurs.

PCC			Défaut : 39.8±10.0
cons : 97.2±5.3	gen : 39.8±10.0	gen_cons : 39.8±10.0	
PEBLS : 97.4±3.0	C4.5 : 94.9±3.7	CN2 : 92.1±4.7	
SE : 98.8±3.7	SE 0.05 : 96.4±4.1		

La précision obtenue par les classifieurs est dans l'ensemble très bonne. La classification par portée avec généralisation des exemples en règles ne fait malheureusement pas mieux que le classifieur par défaut. Il est vrai que le nombre de règles générées est clairement trop important.

Temps		
cons : 16.82±0.41	gen : 1h17±1:08	gen_cons : 39:36±1:27
PEBLS : 4.26±0.04	C4.5 : 6.42±1.13	CN2 : 2.14±0.24
SE : 54.06±14.52	SE 0.05 : 5.57±0.54	

Paramètres de la classification par portée				
$\|EA\|$: 160.2±0.4	Temps : 3.63	Critère : S	Cohérence (ϵ) : 40	Généralité (M) : 11
$\|gen(EA)\|$: 4306.8±42.2 (4306.8±42.2)		$\|gen\_cons(EA)\|$: 3006.0±54.9 (3006.0±54.9)		

Le grand nombre de règles générées est à rapprocher de la valeur anormalement élevée de ϵ. Ce problème n'est décrit que par des attributs numériques : il est probable que les règles générées soient trop spécifiques. Il serait préférable de généraliser davantage les règles dans le cas d'attributs linéaires.

B.23 Zoology (ZO)

Ce problème de classification concerne un ensembles d'animaux regroupés en sept classes : les mammifères, les oiseaux, les reptiles, les poissons, les batraciens, les insectes et divers. Le problème consiste à déterminer la classe d'un animal à partir de sa description.

Caractéristiques

$\|E\|$	: 101	Classes	: 7	Valeurs manquantes	: Non	Incohérences	: Non
Attributs	: 16	Nominaux	: 16	Linéaires discrets	: 0	Linéaires continus	: 0

Les animaux sont décrits par quinze attributs booléens indiquant si l'animal a des poils, des plumes, des œufs, du lait, des ailes, sait voler, vit dans l'eau, est un prédateur, a des dents, une colonne vertébrale, respire dans l'air, est venimeux, a des nageoires, une queue, est domestique, a la taille d'un chat. Un attribut est nominal multi-valué, c'est le nombre de pattes. Nous ne l'avons pas modélisé par un attribut linéaire car un animal à quatre pattes n'est pas plus proche d'un animal à deux pattes que ne l'est un animal à six pattes. En fait, ces valeurs ne sont pas comparables.

PCC

cons	: 94.0±9.6	gen	: 94.0±5.1	Défaut	: 40.4±16.0
PEBLS	: 95.6±7.5	C4.5	: 88.6±13.9	gen_cons	: 90.0±12.4
SE	: 95.0±9.7	SE 0.05	: 95.8±6.8	CN2	: 90.0±15.6

Les classifieurs obtiennent tous une précision du même ordre. Notons que la classification par portée avec généralisation des exemples en règles se classe bien et que le nombre de règles générées est très intéressant.

Temps

cons	: 0.38±0.01	gen	: 0.91±0.04	gen_cons	: 0.99±0.05
PEBLS	: 0.79±0.01	C4.5	: 0.34±0.03	CN2	: 0.40±0.02
SE	: 5.02±0.32	SE 0.05	: 0.28±0.02		

Paramètres de la classification par portée

$\|EA\|$: 54.8±1.2	Temps : 2.95	Critère : C	Cohérence (ϵ) : 10	Généralité (M) : 4
$\|gen(EA)\|$: 25.9±3.1 (25.9±3.1)		$\|gen\_cons(EA)\|$: 26.2±2.0 (22.4±2.2)		

Les paramètres ϵ et M prennent des valeurs élevées, mais non anormales. Le nombre de règles générées est nettement inférieur à la taille de l'ensemble d'apprentissage. Considérons une des règles générées :
Si l'animal n'a pas de poils, a des plumes, des œufs, pas de lait, pas de dents, une colonne vertébrale, respire dans l'air, n'est pas venimeux, n'a pas de nageoire, a deux pattes et une queue alors c'est un oiseau.
La règle demande à être généralisée, mais elle est compréhensible.

Bibliographie

[Aha *et al.*, 1991] David W. Aha, Dennis Kibler et Marc K. Albert. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991.

[Aha, 1991] David W. Aha. Incremental constructive induction: An instance-based approach. Dans *Proceedings of the Eighth International Workshop on Machine Learning*, pages 117–121, Evanston, ILL, 1991. Morgan Kaufmann.

[Barbut et Monjardet, 1970] Marc Barbut et Bernard Monjardet. *Ordre et Classification, Algèbre et Combinatoire*, volume 2. Hachette, 1970.

[Booker *et al.*, 1989] L. B. Booker, D. E. Goldberg et J. H. Holland. Classifier systems and genetic algorithms. *Artificial Intelligence*, 40:235–282, 1989.

[Bossu et Siegel, 1985] Geneviève Bossu et Pierre Siegel. Saturation, nonmonotonic reasoning and the closed-world assumption. *Artificial Intelligence*, 25:13–63, 1985.

[Brachman et Levesque, 1985] R. J. Brachman et H. J. Levesque. A fundamental trade off in knowledge representation and reasoning. Dans *Readings in Knowledge Representation*, rédacteurs R. J. Brachman et H. J. Levesque, pages 41–70. Morgan Kaufmann, 1985.

[Breiman *et al.*, 1984] L. Breiman, J. H. Friedman, R. A. Olshen et C. J. Stone. *Classification and Regression Trees*. Wadsworth International Group, 1984.

[Buntine, 1989] Wray Buntine. Learning classification rules using bayes. Dans *Proceedings of the Sixth International Workshop on Machine Learning*, pages 94–98, Ithaca, New York, USA, 1989. Morgan Kaufmann.

[Castell, 1996] Thierry Castell. Computation of prime implicates and prime implicants by a variant of the davis and putnam procedure. Dans *Proceedings of the Eigth IEEE International Conference on Tools with Artificial Intelligence (ICTAI'96)*, pages 428–429, Toulouse, France, 1996.

[Cerisara *et al.*, 1997] Christophe Cerisara, Jean-Paul Haton, Jean-Francois Mari et Dominique Fohr. Multi-band continuous speech recognition. Dans *Proceedings of the Fifth European Conference on Speech Communication and Technology (EUROSPEECH'97) (à paraître)*, Rhodes, Grece, 1997.

[Chandra et Markowski, 1978] A. K. Chandra et G. Markowski. On the number of prime implicants. *Discrete Mathematics*, 24:5–11, 1978.

[Clark et Boswell, 1991] Peter Clark et Robin Boswell. Rule induction with CN2: some recent improvements. Dans *Proceedings of the Sixth European Working Session on Learning*, pages 151–163, Porto, Portugal, 1991. Lecture Notes on Artificial Intelligence 482, Springer-Verlag.

[Clark et Niblett, 1989] Peter Clark et Tim Niblett. The CN2 induction algorithm. *Machine Learning*, 3(4):261–283, 1989.

[Clark, 1978] Keith L. Clark. Negation as failure. Dans *Proceedings of the Symposium on Logic and Databases*, rédacteurs Hervé Gallaire et Jack Minker, pages 293–322, New York, 1978. Plenum Press.

[Cost et Salzberg, 1993] Scott Cost et Steven Salzberg. A weighted nearest neighbor algorithm for learning with symbolic features. *Machine Learning*, 10:57–78, 1993.

[Cover et Hart, 1967] T.M. Cover et P.E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13:21–27, 1967.

[De Raedt et Bruynooghe, 1989] Luc De Raedt et Maurice Bruynooghe. Constructive induction by analogy: a method to learn how to learn. Dans *Proceedings of the Fourth European Working Session on Learning (EWSL'89)*, rédacteur Katharina Morik, pages 189–199, Pitman, 1989. Morgan Kaufmann.

[De Raedt et Bruynooghe, 1993] Luc De Raedt et Maurice Bruynooghe. A theory of clausal discovery. Dans *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI'93)*, pages 1058–1063, Chambéry, France, 1993.

[De Raedt et Dzeroski, 1994] Luc De Raedt et Saso Dzeroski. First-order jk-clausal theories are PAC-learnable. *Artificial Intelligence*, 70:375–392, 1994.

[Doly *et al.*, 1994] Michel Doly, Jean-Claude Vennat, Jean-Michel Cardot et Pascale Cardot. *Éléments de statistique et de probabilités.* InterÉditions, 1994.

[Domingos, 1994] Pedro Domingos. The RISE system: Conquering without separating. Dans *Proceedings of the Sixth IEEE International Conference on Tools with Artificial Intelligence*, pages 704–707, New Orleans, LA, 1994. IEEE Computer Society Press.

[Domingos, 1995] Pedro Domingos. Rule induction and instance-based learning: A unified approach. Dans *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI'95)*, pages 1226–1232, 1995.

[Domingos, 1996] Pedro Domingos. Unifying instance-based and rule-based induction. *Machine Learning*, 24:141–168, 1996.

[Domingos, 1997] Pedro Domingos. Context-sensitive feature selection for lazy learners. *Artificial Intelligence Review, Special issue on lazy learning*, 11:227–253, 1997.

[Duda et Hart, 1973] Richard O. Duda et Peter E. Hart. *Pattern classification and scene analysis.* John Wiley & and Sons, 1973.

[Feelders *et al.*, 1995] A.J. Feelders, A.J.F. le Loux et J.W. van't Zand. Data mining for loan evaluation at ABN AMRO: a case study. Dans *Proceedings of the First International Conference on Knowledge Discovery and Data Mining (KDD'95)*, rédacteurs U. Fayyad et R. Uthurusamy, pages 106–111, Menlo Park (CA), 1995. AAAI Press.

[Fensel *et al.*, 1995] Dieter Fensel, Monika Zickwolff et Markus Wiese. Are substitutions the better examples? Learning complete sets of clauses with Frog. Dans *Proceedings of the Fifth International Workshop on Inductive Logic Programming (ILP'95)*, Technical report, Department of Computer Science, Katholieke Universiteit Leuven, Belgium, 1995.

[Fisher, 1936] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annual Eugenics*, 7, Part II:179–188, 1936. Paru également dans *Contributions to Mathematical Statistics* (John Wiley, NY, 1950).

[Flach, 1995] Peter Flach. *Conjectures: an inquiry concerning the logic of induction.* PhD thesis, Tilburg University, 1995.

[Flach, 1996] Peter Flach. Rationality postulates for induction. Dans *Proceedings of the Sixth Conference on Theoretical Aspects of Rationality and Knowledge (TARK'96)*, rédacteur Yoav Shoham, pages 267–281, De Zeeuwse Stromen, The Netherlands, 1996.

[Gams, 1989] M. Gams. New measurements highlight the importance of redundant knowledge. Dans *Prococeedings of the Fourth European Working Session on Learning (EWSL'89)*, pages 71–79, London, 1989. Pitman.

[Grégoire, 1990] Eric Grégoire. *Logiques non monotones et intelligence artificielle.* Hermès, 1990.

[Haton *et al.*, 1991a] Jean-Paul Haton, Nadget Bouzid, François Charpillet, Marie-Christine Haton, Brigitte Lâasri, Hassan Lâasri, Pierre Marquis, Thierry Mondot et Amedeo Napoli. *Le raisonnement en intelligence artificielle. Modèles, techniques et architectures pour les systèmes à base de connaissances.* InterEditions, 1991.

[Haton *et al.*, 1991b] Jean-Paul Haton, Jean-Marie Pierrel, Guy Perennou, Jean Caelen et Jean-Luc Gauvain. *Reconnaissance automatique de la parole.* DUNOD Informatique, Bordas, 1991.

[Helft, 1988] Nicolas Helft. *L'Induction en Intelligence Artificielle: Théorie et Algorithmes.* Thèse de doctorat, Université d'Aix-Mareille II, 14 septembre 1988.

[Helft, 1989] Nicolas Helft. Induction as nonmonotonic inference. Dans *Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning (KR'89)*, pages 149–156, Toronto, Canada, 1989. Morgan Kaufmann.

[Hempel, 1943] Carl. G. Hempel. A purely syntactical definition of confirmation. *Journal of Symbolic Logic*, 8(4):122–143, 1943.

[Hempel, 1945] Carl. G. Hempel. Studies in the logic of confirmation. *Mind*, 54(213 & 214):1–26 & 97–121, 1945.

[Hunt *et al.*, 1966] Earl B. Hunt, Janet Marin et Philip J. Stone. *Experiments in induction.* Academic Press, 1966.

[Jackson et Pais, 1990] Peter Jackson et John Pais. Computing prime implicants. Dans *Proceedings of the Tenth International Conference on Automated Deduction (CADE'90), Lecture Notes in Artificial Intelligence 449*, pages 543–557, Kaiserslautern, FR Germany, 1990. Springer-Verlag.

[Jelinek, 1976] F. Jelinek. Continuous speech recognition by statistical methods. *IEEE Transactions on Acoustic Speech and Signal Processing*, 64(4):532–556, 1976.

[Kodratoff et Ganascia, 1986] Yves Kodratoff et Jean-Gabriel Ganascia. *MACHINE LEARNING. An Artificial Intelligence Approach*, volume 2, chapitre 9 - Improving the generalization step in learning, édité par Ryszard S. Michalski and Jaime G. Carbonell and Tom M. Mitchell, pages 215–244. Morgan Kaufmann, 1986.

[Kohavi et John, 1995] Ron Kohavi et George H. John. Automatic parameter selection by minimizing estimated error. Dans *Proceedings of the Twelfth International Conference on Machine Learning (ICML'95)*, rédacteurs Armand Prieditis et Stuart Russell, San Francisco, CA, 1995. Morgan Kaufmann.

[Lakemeyer, 1995] Gerhard Lakemeyer. A logical account of relevance. Dans *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI'95)*, pages 853–859, Montreal, Canada, 1995. Morgan Kaufmann.

[Madre et Coudert, 1994] J.-C. Madre et O. Coudert. Une approche intensionnelle du calcul des implicants premiers et essentiels des fonctions booléennes. *Informatique Théorique et Applications*, 28:125–149, 1994.

[Marquis, 1989] Pierre Marquis. Computing most specific generalisations in propositional calculus. Dans *Proceedings of the Fourth European Working Session on Learning (EWSL'89)*, pages 135–138. Pitman Publishing, 1989.

[Marquis, 1992] Pierre Marquis. Building up inductive generalizations from facts. Dans *Proceedings of the Tenth European Conference on Artificial Intelligence (ECAI'92)*, pages 446–450, Vienna, Austria, 1992. Wiley & Sons.

[McCarthy, 1980] John McCarthy. Circumscription: a form of non-monotonic reasoning. *Artificial Intelligence*, 13:27–39, 1980.

[McCarthy, 1986] John McCarthy. Applications of circumscription to formalizing commonsense knowledge. *Artificial Intelligence*, 28:89–116, 1986.

[Merz et Murphy, 1996] C.J. Merz et P.M. Murphy. UCI repository of machine learning databases. http://www.ics.uci.edu/~mlearn/MLRepository.html, 1996.

[Michalski, 1969] R. S. Michalski. On the quasi-minimal solution of the general covering problem. Dans *Proceedings of the First International Symposium on Information Processing*, pages 125–128, Bled, Yugoslavia, 1969.

[Michalski, 1983] R. S. Michalski. A theory and methodology of inductive learning. Dans *Machine Learning: An artificial intelligence approach*, rédacteurs R. S. Michalski, J. G. Carbonell et T. M. Mitchell, San Mateo, CA, 1983. Morgan Kaufmann.

[Minker, 1982] Jack Minker. On indefinite databases and the closed world assumption. Dans *Proceedings of the Sixth Conference on Automated Deduction (CADE'82)*, pages 292–308, Berlin, Germany, 1982. Lecture Notes in Computer Science 138, Springer-Verlag.

[Mitchell, 1982] T. M. Mitchell. Generalization as search. *Artificial Intelligence*, 18:203–226, 1982.

[Mitchell, 1997] Tom M. Mitchell. *Machine Learning.* McGraw-Hill, 1997.

[Monteanu, 1996] Paul Monteanu. *Extraction de connaissances dans les bases de données Parole : Apport de l'apprentissage symbolique.* Thèse de doctorat, Institut National Polytechnique de Grenoble, 6 décembre 1996.

[Muggleton et DeRaedt, 1994] Stephen Muggleton et Luc DeRaedt. Inductive logic programming: Theory and methods. *Journal of Logic Programming*, 19:629–679, 1994.

[Muggleton, 1995] Stephen Muggleton. Inverse entailment and progol. *New Generation Computing*, 13(3-4):245–286, 1995.

[Nédellec *et al.*, 1996] C. Nédellec, C. Rouveirol, H. Adé, F. Bergadano et Tausend B. Declarative bias in inductive logic programming. Dans *Advances in Inductive Logic Programming*, rédacteur Raedt de L., pages 82–103. IOS Press, 1996.

[Niblett, 1988] T. Niblett. A study of generalisation in logic programs. Dans *Proceedings of the Third European Working Session on Learning (EWSL'88)*, pages 131–138, Glasgow, 1988.

[Nicod, 1961] Jean Nicod. *Le problème logique de l'induction.* 2^{nd} éd, 1961.

[Nock et Gascuel, 1995] Richard Nock et Olivier Gascuel. On learning decision committees. Dans *Proceedings of the Twelfth International Conference on Machine Learning (ICML'95)*, pages 413–420. Morgan Kaufmann, 1995.

[Plotkin, 1970a] Gordon D. Plotkin. A further note on inductive generalization. *Machine Intelligence*, 6:101–124, 1970.

[Plotkin, 1970b] Gordon D. Plotkin. A note on inductive generalization. *Machine Intelligence*, 5:153–163, 1970.

[Quinlan et Cameron-Jones, 1993] J. R. Quinlan et R. M. Cameron-Jones. Foil: A midterm report. Dans *Proceedings of the Fifth European Conference on Machine Learning (ECML'93), Lecture Notes in Artificial Intelligence 667*, rédacteur P. B. Brazdil, Vienna, Austria, 1993. Springer-Verlag.

[Quinlan, 1986] John Ross Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986.

[Quinlan, 1990] John Ross Quinlan. Learning logical definitions from relations. *Machine Learning*, 5:239–266, 1990.

[Quinlan, 1993a] John Ross Quinlan. *C4.5: Programs for machine learning.* Series in machine learning. Morgan Kaufmann, 1993.

[Quinlan, 1993b] John Ross Quinlan. Combining instance-based et model-based learning. Dans *Proceedings of the Tenth International Conference on Machine Learning (ICML'93)*, pages 236–243, Amherst, MA, 1993. Morgan Kaufmann.

[Reiter, 1978] Raymond Reiter. On closed-world data base. Dans *Proceedings of the symposium on logic and data bases*, rédacteurs Hervé Gallaire et Jack Minker, pages 55–76, New York, 1978. Plenum Press.

[Rendell et Cho, 1990] Larry Rendell et Howard Cho. Empirical learning as a function of concept character. *Machine Learning*, 5:267–298, 1990.

[Rumelhart *et al.*, 1986] D. E. Rumelhart, J. L. McClelland et the PDP Research Group. *Parallel distributed processing: Explorations in the microstructure of cognition (Vol. 1).* MIT Press, Cambridge, MA, 1986.

[Rymon, 1992] Ron Rymon. Search through systematic set enumeration. Dans *Proceedings of the Third International Conference on Knowledge Representation and Reasoning (KR'92)*, pages 539–550, Cambridge, Massachusetts, USA, 1992. Morgan Kaufmann.

[Rymon, 1993] Ron Rymon. An SE-tree based characterization of the induction problem. Dans *Proceedings of the Tenth International Conference on Machine Learning (ICML'93)*, pages 268–275, Amherst, Massachusetts, USA, 1993.

[Rymon, 1994] Ron Rymon. On kernel rules and prime implicants. Dans *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI'94)*, pages 181–186, Seattle, Washington, USA, 1994.

[Rymon, 1996a] Ron Rymon. SE-Learn home page. http://www.isp.pitt.edu/~rymon/SE-Learn.html, 1996.

[Rymon, 1996b] Ron Rymon. SE-trees outperform decision trees in noisy domains. Dans *Proceedings of the Second International Conference on Knowledge Discovery in Databases (KDD'96)*, rédacteurs E. Simoudis, J. Han et U. Fayyad, pages 331–334, Portland, Oregon, USA, 1996. AAAI Press.

[Sahami, 1995] Mehran Sahami. Learning classification rules using lattices. Dans *Proceedings of the Seventh European Conference on Machine Learning, LNCS 912*, rédacteurs Nada Lavrac et Stefan Wrobel, pages 343–346, Heraclion, Crete, Greece, 1995.

[Salzberg, 1991] Steven Salzberg. A nearest hyperrectangle learning method. *Machine Learning*, 6:251–276, 1991.

[Saporta, 1990] Gilbert Saporta. *Probabilités, analyse des données et statistique.* Technip, 1990.

[Schaffer, 1994] Cullen Schaffer. A conservation law for generalization performance. Dans *Proceedings of the Eleventh International Conference on Machine Learning (ICML'94)*, pages 259–265, New Brunswick, NJ, 1994. Morgan Kaufmann.

[Sebag et Rouveirol, 1997] Michèle Sebag et Céline Rouveirol. Tractable induction and classification in first-order logic via stochastic matching. Dans *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI'97)*, pages 888–893, Nagoya, Japan, 1997.

[Sebag et Schoenauer, 1994] Michèle Sebag et Marc Schoenauer. A rule-based similarity measure. Dans *Proceedings of the First European Workshop on Topics in Case-Based Reasoning, Lectunre Notes on Artificial Intelligence 837*, rédacteurs Stefan Wess, Klaus-Dieter Althoff et Michael Richter, pages 119–131, Berlin, 1994. Springer.

[Sebag, 1994] Michèle Sebag. Using constraints to building version spaces. Dans *Proceedings of the Sixth European Conference on Machine Learning (ECML'94), Lecture Notes in Artificial Intelligence 784*, rédacteurs Francesco Bergadano et Luc DeRaedt, Catania, Italy, 1994. Springer Verlag.

[Sebag, 1996] Michèle Sebag. Delaying the choice of bias: A disjunctive version space approach. Dans *Proceedings of the Thirteenth International Conference on Machine Learning (ICML'96)*, rédacteur L. Saitta, pages 444–452, Bari, Italy, 1996. Morgan Kaufmann.

[Sebban et Zighed, 1996] Marc Sebban et Djamel Abdelkaber Zighed. Test de séparabilité dans IRp : application à la modélisation de la marche. Dans *Proceedings ARAE*, 30-31 mai 1996.

[Shepherdson, 1988] John C. Shepherdson. Negation in logic programming. Dans *Foundations of deductive databases and logic programming*, rédacteur Jack Minker, pages 19–89, 1988.

[Slagle *et al.*, 1970] J. R. Slagle, C. L. Chang et R. C. T. Lee. A new algorithm for generating prime implicants. *IEEE Transactions on Computers*, C-19(4):304–310, 1970.

[Stanfill et Waltz, 1986] C. Stanfill et D. Waltz. Toward memory based reasoning. *Communications of the ACM*, 29:1213–1228, 1986.

[Thayse *et al.*, 1989] André Thayse, Pascal Gribomont, Guy Hulin, Alain Pirotte, Dominique Roelants, Dominique Snyers, Marc Vauclair, Paul Gochet, Pierre Wolper, Eric Grégoire et Philippe Delsarte. *Approche Logique de l'Intelligence Artificielle*, volume 2. De la logique modale à la logique des bases de données. Dunod informatique, 1989.

[Utgoff, 1986] Paul E. Utgoff. Shift of bias for inductive concept learning. Dans *Machine learning: An artificial intelligence approach*, rédacteurs R. S. Michalski, J.G. Carbonell et T. M. Mitchell. Morgan Kaufmann, 1986.

[Watanabe, 1965] S. Watanabe. A mathematical explication of inductive inference. Dans *Colloque sur les fondements des mathématiques*, pages 67–107, Paris, 1965.

[Wettschereck et Dietterich, 1995] Dietrich Wettschereck et Thomas G. Dietterich. An experimental comparison of the nearest-neighbor and nearest-hyperrectangle algorithms. *Machine Learning*, 19:5–27, 1995.

[Wille, 1982] Rudolf Wille. Restructuring lattice theory. Dans *Ordered sets*, rédacteur Ivan Rival, pages 445–470. D. Reidel, 1982.

Index

Nom : **LACHICHE**

Prénom : **Nicolas**

DOCTORAT de l'UNIVERSITE HENRI POINCARE, NANCY-I

en INFORMATIQUE

VU, APPROUVÉ ET PERMIS D'IMPRIMER

Nancy, le 17 NOV 1997 n° 121

Le Président de l'Université

UNIVERSITÉ HENRI POINCARÉ, NANCY 1 ★ LE PRESIDENT
J.P. FINANCE

3C-F9

UNIVERSITÉ HENRI POINCARÉ · NANCY 1
24-30, rue Lionnois B.P.3069 54013 Nancy cedex
Tél : 83 32 81 81 Fax : 83 32 95 90
Adresse électronique : _____ @presn1.u-nancy.fr

Abstract

Confirmatory generalisation consists in determining the most general laws confirmed by a set of observations. Confirmatory induction is based on the similarity assumption: unknown individuals behave like known individuals. We show that a circumscription of individuals is more appropriate to model this assumption than a circumscription of properties. Compared to existing approaches, the model we propose can produce more general clauses and avoids the production of unwanted generalisations. We specialise this model defined in first-order logic to attribute-value languages and show that it comes down, in this case, to the calculation of prime implicates.

Considering the problem of classification of objects from examples, we show that the minimal consistent and relevant rules are not, in general, confirmatory generalisations. We propose a new classification technique, called scope classification, which consists of building the set of examples from which a consistent and relevant rule can be built, so it is an instance-based approach of rule-based classification. We present several adaptations of the logical basements of the scope classification to better deal with real data. Scope classification is also extended to instances generalised into rules. The generalisation strategies we introduce, and especially the search of the neighbours, clearly differ from those of existing techniques. We show that, whereas hypotheses built by the scope classification and by the disjunctive version space differ, both techniques leads to the same classification. Though, in addition to a more efficient implementation, our original point of view allows us to propose some developments specific to a rule-based approach. Our system obtains on average a better accuracy and a similar execution time to some of the most used instance-based or rule-based systems on an usual set of benchmarks.

Keywords: Machine learning, confirmatory induction, classification.

Résumé

La généralisation confirmatoire consiste à déterminer les lois les plus générales confirmées par un ensemble d'observations. L'induction confirmatoire repose sur l'hypothèse des similarités : les individus inconnus se comportent comme les individus connus. Nous montrons qu'une circonscription des individus modélise mieux cette hypothèse qu'une circonscription des propriétés. Nous proposons un modèle capable d'obtenir des clauses plus générales et d'éviter des généralisations indésirables par rapport aux approches existantes. Ce modèle défini en logique des prédicats se spécialise dans le cas propositionnel en un calcul des premiers impliqués.

Considérant le problème de la classification, ou classement, d'objets à partir d'exemples, nous montrons que les règles minimales cohérentes et pertinentes ne sont pas, dans le cas général, des généralisations confirmatoires. Nous proposons une nouvelle technique de classification, dite par portée, qui consiste à chercher l'ensemble des exemples à partir desquels une règle cohérente et pertinente peut être construite. Elle a donc une approche à base d'instances de la classification à base de règles. Nous présentons un ensemble d'adaptations des fondements logiques de la classification par portée aux données réelles. Nous étendons également notre technique aux instances généralisées en règles. Les stratégies de généralisation que nous proposons, et surtout la recherche des voisins diffèrent nettement de celles existantes. Nous montrons que, bien que les hypothèses construites diffèrent, notre approche et celle de l'espace des versions disjonctif conduisent au même classement. Notre point de vue original permet cependant de proposer, au delà d'une implantation plus efficace, des développements propres à une approche à base de règles. Notre approche a en moyenne une meilleure précision et un temps d'exécution semblable à ceux des approches les plus utilisées à base d'instances ou à base de règles sur des ensembles de test usuels.

Mots-clés: Apprentissage automatique, induction confirmatoire, classification.