Fouille de données spatiales
Approche basée sur la programmation logique inductive

Nadjim Chelghoum*, Karine Zeitouni**, Thierry Laugier*, Annie Fiandrino*, Lionel Loubersac*

* Institut Français de Recherche pour l'Exploitation de la Mer (IFREMER)
Laboratoire Environnement- Ressources du Languedoc Roussillon (LER/LR)
BP 171, Boulevard Jean Monnet,
34203 Sète Cedex
Tel/Fax (0)4 99 57 32 83/ (0)4 99 57 32 96
prénom.nom@ifremer.fr

** Laboratoire PRiSM, Université de Versailles
45, avenue des Etats-Unis, 78035 Versailles Cedex, France
Tel / Fax: (0)1 39 25 40 46 / (0)1 39 25 40 57
prénom.nom@prism.uvsq.fr

Résumé. Ce qui caractérise la fouille de données spatiales est la nécessité de prendre en compte les interactions des objets dans l'espace. Les méthodes classiques de fouille de données sont mal adaptées pour ce type d'analyse. Nous proposons dans cet article une approche basée sur la programmation logique inductive. Elle se base sur deux idées. La première consiste à matérialiser ces interactions spatiales dans des tables de distances, ramenant ainsi la fouille de données spatiales à la fouille de données multi-tables. La seconde transforme les données en logique du premier ordre et applique ensuite la programmation logique inductive. Cet article présentera cette approche. Il décrira son application à la classification supervisée par arbre de décision spatial. Il présentera aussi les expérimentations réalisées et les résultats obtenus sur l'analyse de la contamination des coquillages dans la lagune de Thau.

1 Positionnement du problème

La fouille de données spatiales (FDS) consiste à appliquer la fouille de données aux données spatiales (Zeitouni, 2000), (Han et Kamber, 2001), (Shashi et Sanjay, 2003). Contrairement aux données classiques, ces données sont de nature dépendantes (Anselin, 1989), (Longley et Goodchild, 1999) car tout phénomène spatial est influencé par son voisinage. Par exemple, la contamination des coquillages dans une lagune est influencée par les champs d'agriculture autours. Cette notion de dépendance entre les données est justifiée par une loi en géographie qui stipule que « *ce qui se passe dans une localité particulière dépend de ce qui se passe dans d'autres localités et ces interactions sont d'autant plus fortes que les localités concernées sont plus proches* » (Tobler, 1979). Du point de vue analyse, ceci revient à dire que l'analyse spatiale nécessite l'analyse des entités spatiales en fonction de leurs caractéristiques, des caractéristiques de leurs voisins et des relations spatiales avec ces voisins. Analyser ces données sans considérer cette spécificité génère des résultats incorrects (Anse-

lin et Griffith, 1988). Prendre en compte cette spécificité pose néanmoins deux problèmes. Le premier est l'inadaptation des méthodes existantes de fouille de données à ce type d'analyse. Le second est la complexité de calcul des relations spatiales.

1.1 Inadaptation des méthodes traditionnelles à la FDS

Aucune méthode d'analyse de données ne permet d'analyser les données dépendantes. En effet, la statistique et l'analyse de données, ainsi les méthodes de fouille de données traditionnelles supposent toutes les données indépendantes. De plus, aucune de ces méthodes ne peut interpréter les propriétés spatiales ni découvrir les liens entre objets spatiaux. Toutes traitent des données simples de type numérique ou chaîne de caractère. Certes, les systèmes d'information géographique (SIG) (Laurini et Thompson, 1994) et la statistique dite spatiale (Cressie, 1993), (Sanders, 1989), (Shaw et Wheeler, 1994) intègrent parfaitement les relations spatiales, mais ils restent encore limités dans l'analyse spatiale exploratoire. En fait, les SIG, même s'ils permettent d'interroger les données spatiales et de répondre à un certain calcul (ex : trouver la surface d'une lagune), ne permettent pas de découvrir des modèles, ni des règles ou de nouvelles connaissances cachées dans les bases de données spatiales. La statistique spatiale, même si elle est largement répandue et offre un grand nombre de techniques allant de la géostatistique à l'analyse globale et locale d'autocorrélation ou l'analyse de données multi-variées, reste le plus souvent confirmatoire, guidée par un expert, basée sur des données numériques et ne découvre pas de règles. De plus, l'analyse exploratoire de données multi-variées sous contrainte de contiguïté présente l'inconvénient de ne considérer que les relations entre objets d'une même table excluant les relations spatiales pouvant exister entre objets de tables différentes (Zeitouni, 2000). Par conséquent, ces méthodes ne sont pas adaptées à la fouille les données spatiales.

1.2 Complexité du calcul des relations spatiales

Les relations spatiales représentent les liens de voisinage qui relient les objets spatiaux. Elles jouent un rôle important en géographie car elles mettent en évidence l'influence de voisinage. Elles peuvent être métriques, topologiques ou directionnelles (Egenhofer, 1991).
Le problème des relations spatiales est double. Le premier est que ces relations sont souvent implicites. Elles ne sont pas stockées dans les bases de données spatiales et sont résolues chaque fois qu'elles sont évoquées, nécessitant des calculs géométriques complexes et coûteux. Il faut donc optimiser leur calcul. Le deuxième problème est que les relations spatiales sont nombreuses, voir infinies (distance, inclusion, …). Par conséquent, le choix de la ″bonne″ relation spatiale à prendre en compte dans un processus de fouille est difficile. Les méthodes de fouille de données spatiales existantes (Ester et al, 1997), (Koperski et al, 1998, 1996), (Malerba et Lisi, 2001), (Ceci et al., 2004) se limitent toutes à des relations spatiales en nombre limité et choisies par un utilisateur métier. Ce choix devient difficile lorsque les relations potentiellement intéressantes sont multiples. Il faut donc trouver des méthodes qui permettent de choisir automatiquement ces ″bonnes″ relations spatiales.

Dans la suite de cet article, la section 2 donne quelques définitions préliminaires. La section 3 présente l'approche proposée. La section 4 décrit l'application de cette approche aux arbres de décision spatiaux. Les expérimentations et les résultats obtenus sur l'analyse de la contamination des coquillages dans la lagune de Thau sont présentés dans la section 5, suivis par une discussion et une conclusion.

2 Définitions préliminaires

Dans cette section, nous présenterons brièvement la programmation logique inductive et la méthode TILDE que nous utiliserons dans l'approche proposée. Un état de l'art plus détaillé sur la fouille de données spatiales et les relations spatiales est donné dans (Chelghoum, 2004), (Han et Kamber, 2001), (Shashi et Sanjay, 2003), (Zeitouni, 2000).

2.1 Programmation logique inductive

La dénomination "programmation logique inductive" (PLI) est due à Muggleton (Muggleton, 1991). Elle est née du croisement de l'apprentissage automatique et de la programmation logique. À l'inverse de la programmation logique déductive, qui dérive des conséquences à partir des théories, la programmation logique inductive a pour but de trouver des hypothèses H à partir d'un ensemble d'observations E. Il s'agit de synthétiser de nouvelles connaissances à partir d'observations et d'une base de connaissances. Elle réalise la même tâche que la fouille de données traditionnelle qui génère des hypothèses à partir de données. La différence est que, tandis que la fouille de données opère sur des données organisées dans une table et de format "attribut = valeur", la programmation logique inductive suppose que les données en entrée ainsi que les modèles extraits sont exprimés en logique du premier ordre - appelée aussi logique des prédicats – (Lavrac et Dzeeoski, 1994).

Formellement, la programmation logique inductive est décrite de la façon suivante (Lavrac et Dzeeoski, 1994) :
Entrées : Trois ensembles de clauses : B, P et N avec
- B : base de connaissances exprimées sous forme de clauses de Horn,
- P : exemples positifs exprimés sous forme de clauses de Horn,
- N : exemples négatifs exprimés sous forme de clauses de Horn.

Sortie : Trouver une hypothèse H sous forme d'un ensemble de clauses de Horn et telle que les propriétés suivantes soient le plus possible respectées :
- Complétude : $\forall e \in P, H \cup B \models e$
- Consistance : $\forall e \in N, H \cup B$ (non $\models$) e

On cherche donc à trouver une hypothèse H qui permet d'expliquer au mieux les exemples positifs, tout en rejetant au maximum les exemples négatifs. Cette recherche s'effectue par inversion du raisonnement déductif. Elle se base souvent sur la propositionnalisation (Kramer et al, 2001) ou sur l'adaptation des méthodes de fouille de données à la logique de premier ordre et sur l'utilisation d'autres techniques liées à la programmation logique comme la substitution, la spécialisation, la généralisation, l'unification et la résolution (Van Laer et De Raedt, 2000). La définition du langage, l'alphabet et les concepts utilisés en PLI est donnée dans (Lavrac et Dzeroski, 1994).

2.2 TILDE (Top- down Induction Logical DEcision tree)

La technique TILDE (Blockeel et De Raedt, 1998) est une méthode de classification par arbre de décision basée sur la logique de premier ordre. C'est une extension d'une méthode bien connue en fouille de données : C4.5 de Quinlan (Quinlan 1986). Elle génère un arbre de décision binaire conformément à la définition d'un arbre de décision logique (Blockeel et De Raedt, 1998). Pour la construction d'arbre de décision, elle applique le même principe que les techniques classiques : applications successives de critères de subdivision sur une population d'apprentissage afin d'aboutir à des sous populations qui maximisent les effectifs d'une

des classes. La prémisse de la règle de décision est la conjonction de littéraux. Du fait que le critère de subdivision se base sur un prédicat simple ou dérivé, TILDE peut prendre en compte des relations entre tables, des règles d'experts ou des prédicats exprimant une conjonction de prédicats simples. De ce fait, elle génère un arbre moins profond que C4.5. La figure ci-dessous décrit cet algorithme.

Paramètres en entrée : T : Arbre, E : ensemble d'exemples, B : base de connaissances,
Procédure Construire_Arbre (T, E, B, True) ;
/* La classe est un prédicat dans E. Initialement, T est vide est Q = true */
Paramètres en sortie : T : Arbre de décision binaire

Procédure Construire_Arbre
En entrée : N : nœud, E : Ensemble des exemples du nœud N, Q : prémisse du nœud
Si (E est suffisamment homogène) **alors**
1. K ← classe_majoritaire ; N : feuille (info (E)) ;

Sinon
2. L ← ensemble des spécialisations de Q dans E
3. Q_b ← la meilleure condition qui segmente E /*calculée suivant une heuristique : gain ratio */
4. Conj ← $Q_b \wedge Q$;
5. E1 = {e ∈ E/ e est vrai dans Conj}; E2 = {e ∈ E/ e est faux dans Conj};
6. Construire_Arbre (gauche, E1, B, Q_b};
7. Construire_Arbre (droit, E2, B, Q} ;
8. N = nœud (Conj, gauche, droit) ;

Fin si
En sortie : Arbre T ;

FIG. 1 - *Algorithme TILDE.*

Initialement, l'arbre est vide, la prémisse Q = true et toutes les observations E sont dans le nœud racine. On commence par vérifier si ce nœud racine est homogène ou pas. Le cas échéant, on déclare le nœud comme feuille (nœud saturé) et on récupère toutes les informations le concernant (ligne 1). Sinon, on calcule l'ensemble des spécialisations de la prémisse Q dans E (ligne 2). La spécialisation consiste à ajouter un littéral à la prémisse d'une clause ou à substituer une variable par un terme. Parmi ces spécialisations, on retient celle qui donne une meilleure segmentation de E. Cette meilleure segmentation est choisie en fonction d'un critère utilisé dans C4.5 basé sur le gain ratio (ligne 3). On ajoute cette spécialisation à la prémisse de Q et on divise le nœud père en deux nœuds : fils gauche qui contient les observations qui vérifient la condition et fils droit qui contient les observations qui ne vérifient pas la condition (ligne 5). On réitère la procédure de construction de l'arbre pour chacun des fils gauche et droit (ligne 6 et 7) et on insère le nœud parent dans l'arbre (ligne 8). Le processus s'arrête lorsque tous les nœuds sont saturés.

3 Approche proposée

Comme il a été souligné précédemment, la fouille de données spatiales utilise d'une manière intensive les relations spatiales car ces dernières mettent en évidence l'influence du voisinage entre les entités spatiales. Ces relations sont à l'origine implicites et nécessitent des jointures coûteuses sur des critères spatiaux pour être exhibées. Zeitouni et al. (Zeitouni et

al., 2000) proposent de les rendre explicites en utilisant l'index de jointure spatiale (cf. FIG. 2). L'idée est de pré calculer les relations spatiales exactes entre les localisations de deux collections d'objets spatiaux et de les stocker dans une table de type (objet1, relation spatiale, objet2). Nous proposons d'exploiter cette structure et de l'intégrer dans la fouille de données spatiales. Outre le fait que cette jointure via l'index est bien plus performante qu'une jointure spatiale, cette organisation relationnelle nous offre un grand avantage : elle ramène la fouille de données spatiales à la fouille de données multi-tables (Dzeroski et Lavrac, 2001). Afin de limiter le coût de construction de cet index, le calcul se limite à un périmètre de distance utile. Ce coût est celui de la jointure spatiale sur critère de distance, mais l'avantage est d'éviter ce coût lors des multiples résolutions ultérieures des relations spatiales.

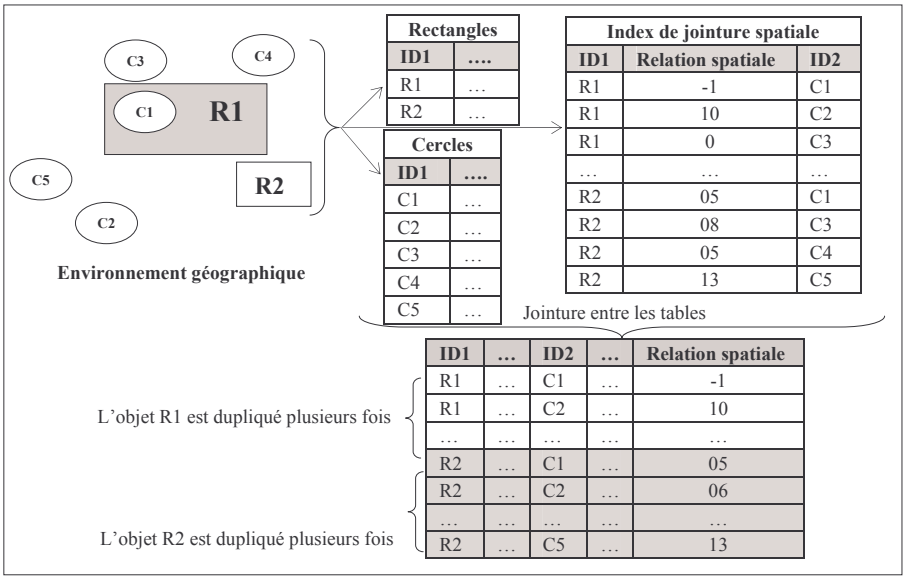

FIG. 2 - *Index de jointure spatiale (en haut) et problème de la jointure (en bas).*

Désormais, tout problème de fouille de données spatial peut être réduit à un problème de fouille de données multi-tables et l'utilisation des relations spatiales devient alors possible car elles seront vues par les méthodes d'analyse comme un attribut à analyser à l'instar des autres attributs. Ainsi, le choix de la bonne relation spatiale peut être fait automatiquement par les méthodes d'analyse répondant ainsi au deuxième problème cité précédemment. Cependant, cette organisation multi-tables des données ne peut pas être directement analysée par les méthodes d'analyse de données car celles-ci considèrent que les données en entrée sont dans une table unique où chaque tuple constitue une observation à analyser et chaque colonne est une variable d'analyse. Il est possible de se ramener à une seule table en joignant les différentes tables initiales. Or, cette jointure peut dupliquer des tuples car les observations à analyser sont en liaison N-M avec les objets voisins (cf. FIG. 2). Ceci fausse les résultats des méthodes de fouille de données en raison du multiple comptage de ces observations. Par exemple, l'objet rectangle R1 (cf. fig. 2) se retrouve dupliqué autant de fois qu'il existe d'objets voisins C_i. Le même objet sera compté plusieurs fois et risque d'être classé dans différen-

tes classes si on applique un algorithme classique d'arbre décision et générera ainsi des règles non discriminantes. Les travaux existants contournent ce problème en généralisant les données dupliquées comme dans (Koperski et al., 1996).

Pour résoudre ce problème multi-tables, nous proposons une approche basée sur la programmation logique inductive (PLI). Elle consiste à transformer les données multi-tables en logique des prédicats et d'appliquer ensuite les méthodes de la PLI pour l'extraction des connaissances. Ceci nous permet de bénéficier des avancées de cette dernière en termes d'algorithmes, de simplicité de ses modèles et la possibilité d'intégrer lors de l'analyse des connaissances implicites. Un panorama d'algorithmes de PLI est présenté dans (Dzeroski et Lavrac, 2001). La transformation des données relationnelles en logique de 1er ordre se fait selon les règles ci-dessous (TAB. 1) et décrites dans (Dzeroski et Lavrac, 2001). La transformation de l'exemple de la FIG. 2 est donnée dans le TAB. 2.

Cette idée d'utiliser la programmation logique inductive pour la fouille de données spatiales est également utilisée par Malerba et al. (Malerba et Lisi, 2001), (Ceci et al., 2004) pour l'extraction des règles d'associations spatiales. Leur démarche consiste à adapter les algorithmes de Koperski et al. (Koperski et al., 1996) aux données spatiales exprimées en logique du premier ordre. L'avantage de ces travaux est qu'ils bénéficient du pouvoir expressif offert par la logique des prédicats. Cependant, comme la méthode de Koperski et al., ils n'explorent pas toutes les relations spatiales et toutes les distances possibles, car la relation spatiale est limitée à un prédicat, évalué à vrai ou faux pour une distance prédéfinie. De plus, ils optent pour la généralisation préalable des données, ce qui même parfois à une perte d'information.

- Chaque table T devient un prédicat P,
- Chaque attribut Att de la table T devient un argument *arg* du prédicat P,
- Chaque tuple (Att_1, …, Att_n) de T devient un fait ou un modèle P (arg_1, …., arg_n),

TAB. 1- *Règles de transformation des données en logique des prédicats.*

Transformation des données de l'exemple en logique du 1er ordre			
Begin(model (rectangle1)).	Cercle (C1, blanc, …).	**Begin**(model (rectangle2)).	Cercle (C1, blanc, …)
Rectangle (R1, grand, …).	Cercle (C2, blanc, …).	Rectangle (R2, petit, …).	Cercle (C2, blanc, …)
Index (R1, -1, C1).	Cercle (C3, blanc, …).	Index (R2, 05, C1).	Cercle (C3, blanc, …)
Index (R1, 10, C2).	Cercle (C4, blanc, …).	Index (R2, 06, C2).	Cercle (C4, blanc, …)
Index (R1, 0, C3).	Cercle (C5, blanc, …).	Index (R2, 08, C3).	Cercle (C5, blanc, …)
Index (R1, 12, C5).	**End**	Index (R2, 05, C4).	**End**

TAB. 2 - *Exemple de transformation des données en logique des prédicats.*

4 Application aux arbres de décision spatiaux

Un arbre de décision est un modèle de fouille de données représentant une structure de connaissances composée d'une séquence de règles de décision. Il a pour but de trouver les attributs explicatifs et les critères précis sur ces attributs donnant le meilleur classement vis-à-vis d'un attribut à expliquer. Il existe diverses méthodes d'arbres de décision (Zighed et Ricco, 2000). Le critère de subdivision est déterminé au niveau de l'attribut comme dans ID3 (Quinlan, 1986) et au niveau d'une valeur d'attribut comme dans CART (Breiman 1984).
L'extension des arbres de décision au spatial se traduit par la prise en compte, non seulement des propriétés des objets à analyser, mais aussi des propriétés des objets voisins et des liens de voisinage. On trouve plusieurs méthodes d'arbre de décision spatial (Ester et al, 1997),

(Koperski et Han, 1998), (Chelghoum et al, 2002), (Chelghoum et Zeitouni, 2004). La description et les limites de chacune de ces méthodes sont résumées dans (Chelghoum, 2004).
En suivant la démarche proposée en section 3, nous proposons une nouvelle méthode d'arbre de décision spatial baptisée S-TILDE (Spatial Top- down Induction Logical DEcision tree). C'est une extension de la méthode TILDE aux données spatiales. Cette extension se traduit par la modification du critère de division d'un nœud. Plus précisément, dans S-TILDE, ce critère intègre les propriétés d'objets voisins et leur relation spatiale avec l'objet à classer. La combinaison de ces propriétés et de la « bonne » relation spatiale sera considérée pour déterminer la meilleure partition. Le fils droit est le complément du fils gauche.

4.1 Description de l'algorithme S-TILDE proposé

L'algorithme (cf. figure 3) prend en entrée la table "*cible*" qui contient les objets à classer, la table "*voisin*" qui contient les objets voisins à prendre en compte, l'"*index de jointure spatial*", les attributs explicatifs qui peuvent provenir de la table cible ou de la table *"voisin"*, l'attribut à expliquer (classe) de la table cible, et enfin les conditions de saturation qui déterminent la poursuite du développement de l'arbre. Pour construire l'arbre, l'algorithme opère en deux phases. La première phase consiste à transformer les données en logique de premier ordre (étape 1 de la figure 3). La seconde applique la méthode TILDE adaptée aux données spatiales (étape 2 à 7).

L'étape de transformation des données s'effectue en respectant les règles présentées précédemment dans le TAB. 1. Ces règles sont générales et ne prennent pas en compte les spécificités de telle ou telle méthode. Dans le cas de S-TILDE, ces règles sont insuffisantes pour trois raisons. La première est qu'elles ne distinguent pas les valeurs de la classe. Nous proposons, pour y remédier, d'ajouter la règle **R1** suivante : « *chaque valeur de la classe devient un prédicat dont l'argument est l'identifiant de l'objet à classer* ». La deuxième raison est qu'elles ne peuvent tenir compte du critère de division d'un nœud défini plus haut et comprenant la combinaison d'une relation de voisinage et de propriétés du voisin. Pour ce faire, on définit la règle **R2** suivante « on substitue la transformation des tables "Index de jointure spatiale" et "*voisin*", par la génération de prédicats *″voisinage* (Id, relation spatiale, attributs du voisinage)*″* où Id est l'identifiant de l'objet à classer ». En effet, si la table cible dans S-TILDE décrit les objets identifiés à classer, ce qui importe pour l'analyse est le type de voisinage plutôt que l'identité du voisin. Enfin, il faut rajouter les règles du domaine. Ici, on sait qu'un objet V_j est à distance *Rel* de O_i l'est aussi à distance $r > Rel$. Par exemple, si un objet cercle est à moins de 50 cm d'un rectangle alors il est aussi à moins de 80 cm de ce rectangle. Pour tenir compte de cette caractéristique, nous proposons d'ajouter la règle **R3** suivante : « $Voisinage\ (id, Rel, X, Y, \ldots, Z) \wedge (Rel < r) \Rightarrow Voisinage\ (id, r, X, Y, \ldots, Z)$ ».

Transformation des données de l'exemple en logique du 1er ordre			
Begin(model (rectangle1)).	Voisinage (R1, 0,…).	Rectangle (R2, , …).	Voisinage (R2, 08,…).
Rectangle (R1, …).	Voisinage (R1, 02, …).	Petit (R2) « issue de R1 »	Voisinage (R2, 05,…).
Grand (R1) « issue de R1 »	Voisinage (R1, 12, …).	Voisinage (R2, 05, …).	Voisinage (R2, 13,…).
Voisinage (R1, -1, …).	**End**	Voisinage (R2, 06, …).	**End**
Voisinage (R1, 10, …).	**Begin**(model (rectangle2)).	Voisinage(R,T)∧(R < r)⇒Voisinage(r,T)	

TAB. 3 - *Exemple de données en entrée exprimée en logique de 1er ordre.*

A noter que cette première phase est effectuée simplement par une réécriture du résultat de la jointure des trois tables initiales. Son coût en entrée/sortie est celui de cette jointure augmenté de la lecture/écriture de son résultat.

Désormais, la construction de l'arbre se base sur des données exprimées en logique de 1er ordre. Le tableau ci-dessus exprime en logique de premier ordre l'exemple de la FIG. 2.

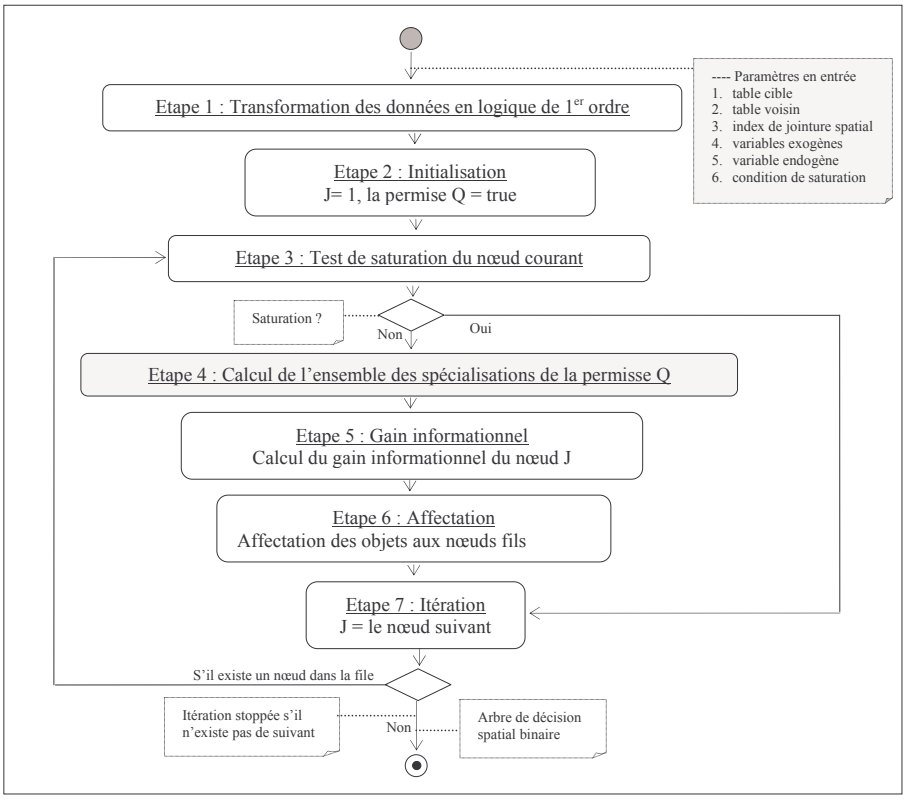

FIG. 3 - *L'algorithme S-TILDE.*

Le principe de construction de l'arbre reste le même que celui de TILDE. Initialement, l'arbre contient un seul nœud racine contenant toutes les observations E et la prémisse « Q = true » (étape 2 de la FIG. 3). On commence par vérifier si ce nœud racine est saturé (étape 3). Si c'est le cas, le développement de l'arbre est stoppé. Sinon, on calcule l'ensemble des spécialisations de la prémisse Q et le gain informationnel de chacune (étape 4 et 5). On retient la spécialisation qui retourne la meilleure valeur du gain informationnel et on divise le nœud courant en deux nœuds : fils gauche et fils droit. On affecte les observations E aux nœuds fils gauche ou fils droit selon qu'elles vérifient ou pas la condition de segmentation. On réitère ce processus sur tous les nœuds jusqu'à saturation de tous les nœuds (étape 7).

5 Expérimentations et résultats

C'est dans le cas de l'analyse de la contamination des coquillages dans la lagune de Thau que nous avons testé notre approche. Vu sa position d'interface entre terre et mer, la lagune de Thau a vu la qualité de son eau menacée par les apports des bassins versants (agricoles,

industriels, urbains, …) perturbant ainsi son activité conchylicole. Les fortes pressions anthropiques auxquelles elle est soumise conduisent régulièrement à des crises et à des contaminations microbiologiques ou chimiques des coquillages réduisant ainsi son activité économique et détruisant souvent l'intégralité du cheptel conchylicole. Notre objectif est d'identifier et de prédire cette contamination des coquillages connaissant la description de la lagune et de son voisinage géographique. Ce qui revient à appliquer la classification supervisée par arbre de décision spatial.

5.1 Test et résultats

L'analyse part d'une base de donnée réelle fournie par Ifremer[1], relevée sur le terrain et décrivant la lagune de Thau et son voisinage géographique. On y trouve des renseignements liés aux points de prélèvement dans la lagune comme la profondeur, la température de l'eau, le degré de salinité ou des renseignements liés aux bassins versants (urbain, industriel,…).
Un exemple de résultat est donné dans la FIG. 4. Il est obtenu en utilisant ACE system[2]. Les paramètres en entrée de ce test sont résumés dans la table TAB. 4.

PARAMETRES EN ENTREE		
Objets à analyser	Prélèvement (2128 prélèvements)	
Objets voisins	Bassins versants (30 bassins)	
Prédicats explicatifs	Voisinage (distance, Libelle)	
Classe	- Contaminé	- Non contaminé
Critères de saturation	Confidence >= 0.25	

TAB. 4 - *Paramètres en entrée du test.*

Dans cet arbre (cf. fig. 4), le fils gauche de la racine correspond aux prélèvements près des bassins versants agricoles (distance≤1788m) affecté à la classe non contaminé. L'affectation à une classe signifie que celle-ci est plus fréquente dans le nœud que dans l'ensemble des prélèvements effectués (ici 169/196). Le fils droit est le complément de fils gauche. Il est segmenté à son tour par une partie près des bassins urbains (distance≤3332m) affectée à la classe contaminé et une partie loin des basins versants urbains et ainsi de suite. Le développement de l'arbre s'arrête lorsque le noeud est saturé.

Arbre de décision

```
voisinage (1788, agricole)?
+--yes:[non contaminé] [169/196]
+--no: voisinage (3332, urbain)?
   +--yes: [contaminé] [9/47]
   +--no: voisinage (2137, agricole)?
      +--yes: [contaminé] [27/170]
      +--no: [non contaminé] [1592/1715]
```

Son équivalent en prolog

```
1. Class ([non contaminé]):- voisinage (1788, agricole)!. (≈53.3%)
2. Class ([contaminé]) :- voisinage (3332, urbain)!. (≈56%)
3. Class ([contaminé]) :- voisinage (2137, agricole)!. (≈50%)
4. Class ([noncontaminee]). (≈ 52%)
```

[1] Institut Français de Recherche pour l'Exploitation de la Mer (http://www.ifremer.fr)
[2] http://www.cs.kuleuven.ac.be/~dtai/ACE/

Remarque : l'ordre de l'écriture des règles est important. Pour déterminer la classe d'un nouvel exemple, on teste d'abord la règle 1 ensuite, en cas d'échec, on teste la règle 2. En cas d'échec avec les règles 1 et 2, on teste la Règle 3 et ainsi de suite.

FIG. 4 - *Arbre de décision spatial.*

Ces règles nous laissent penser que les bassins versants agricoles ne sont pas la cause de la contamination de la lagune, car dans leurs alentours, il y a relativement plus de prélèvements non contaminés. A l'inverse, il est possible que les bassins versants urbains soient la cause de la contamination car il y a relativement plus de prélèvements contaminés dans autour d'eux. Ces nouvelles connaissances restent à valider par des experts du domaine.

5.2 Visualisation cartographique

Le but ultime de ce processus est de localiser sur les cartes les zones correspondant aux règles découvertes. Ici, cela permettrait par exemple de mettre en évidence le lien entre les prélèvements contaminés et un périmètre de 3332m autour des embouchures des bassins urbains (règle 2 de la figure FIG. 4).

La carte ci-contre montre les points de prélèvements contaminés et non contaminés et les alentours des bassins versants agricoles (à distance de 1788m) et urbains (à 3332m). Les alentours des bassins agricoles et urbains sont présentés sur la carte par des zones tampon (resp. des cercles pleins et vides). Les points de prélèvements contaminés et non contaminée sont présentés respectivement par des points blancs et noirs. D'après les calculs faits par l'algorithme, on sait que la proportion des prélèvements contaminés est plus importante autour des bassins urbains (cercles vides). Par conséquent, l'hypothèse du lien entre les bassins urbains et la contamination doit être étudiée. Des règles de ce type auraient été difficile à découvrir visuellement à cause de la superposition des points de relevé - chaque point sur la carte couvre approximativement 270 prélèvements-.

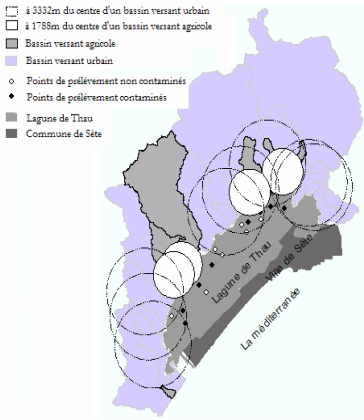

FIG. 5 - *Cartographie des règles 1 et 2*

6 Conclusion

La principale spécificité de la fouille de données spatiales est qu'elle intègre, lors de l'analyse, les relations spatiales. Pour la mise en œuvre de ses méthodes, nous avons proposé, dans cet article, une démarche en deux étapes. La première consiste à matérialiser ces relations spatiales et à les stocker dans des index de jointure spatiale ramenant ainsi la fouille de données spatiales à la fouille de données multi-tables. La deuxième transforme les données multi-tables en logique du premier ordre et applique ensuite la programmation logique inductive. Cette démarche est prometteuse. Ses avantages sont d'une part, d'utiliser n'importe quel algorithme et outil de PLI et de bénéficier du pouvoir expressif qu'offre cette dernière et d'une autre part, d'intégrer lors de l'analyse, les relations spatiales et les proprié-

tés du voisinage. L'application de cette démarche à la méthode d'arbre de décision spatial a été décrite dans cet article. L'originalité de cette méthode est qu'elle nous permet, d'une part, de prendre en compte la description du voisinage et les relations spatiales qui relient les objets, et d'une autre part, d'effectuer un choix automatique de la ″bonne″ relation spatiale. Comparée à l'approche préposée dans (Chelghoum et Zeitouni, 2004), l'inconvénient de cette méthode est celui inhérent à la PLI quant aux faibles performances d'exécution lorsque le volume de données augmente. Cela reste à confirmer sur des jeux de tests synthétiques.

En perspective, nous orienterons nos recherches vers l'intégration des méta-données dans le processus de fouille de données spatiales. En fait, au fil du temps, des données décrivant nos données sont générées. Celles-ci renferment des informations pouvant être utiles à l'analyse. Il est donc intéressant de les intégrer dans le processus d'extraction des connaissances. Des pistes sont d'ores et déjà envisagées. Forte de son pouvoir expressif, la programmation logique inductive nous servira comme un tremplin pour accomplir cette tâche.

Références

Anselin L., D.A. Griffith, Do spatial effects really matter in regression analysis? *Regional Science Association 65*, 11-34. (1988).

Anselin L., What is special about spatial data? Alternative perspectives on spatial data analysis, Technical paper 89-4. Santa Barbara, NCGIA 1989.

Breiman L., J.H. Friedman, R.A. Olshen, C.J. Stone, Classification and Regression Trees, Ed: Wadsworth & Brooks. Montery, California, 1984.

Blockeel H., L. De Raedt, Top-Down induction of first order logical decision trees, *Artificial intelligence*, 102(2-2)/ 285-297, 1998.

Chelghoum N., Fouille de données spatiales. Un problème de fouille de données multi-tables, *Rapport de thèse*, Université de Versailles, 2004 (www.prism.uvsq.fr/~nchelg)

Chelghoum N., K. Zeitouni, Mise en oeuvre des méthodes du data mining spatial. Alternatives et performances, 4èmes Journées d'Extraction et de Gestion des Connaissances, EGC 2004, Clermont-Ferrand, France, 211-217, 20-23 Janvier 2004.

Chelghoum N., K. Zeitouni, Spatial data mining implementation-Alternatives and performances. VI Brazilian symposium on Geoinformatics (GEOINFO'2004), Campos do Jordao, Brazil, November 22-24, 2004.

Chelghoum N., K. Zeitouni, A. Boulmakoul, A decision tree for multi-layered spatial data, In 10th International Symposium on Spatial Data Handling (SDH), Edition Springer, 1-10, Ottawa, Canada, , July 8-12 2002.

Cressie N.A.C, *Statistics for spatial data*, Edition Wiley, New York, 1993.

Dzeroski S., N. Lavrac, *Relational Data Mining*, Springer, 2001.

Egenhofer M.J., Reasoning about Binary Topological Relations, Proc. 2nd Int. Symp. on Large Spatial Databases, 143-160, Zurich, Switzerland, 1991.

Ester M., H.P. Kriegel, J. Sander, Spatial Data Mining: A Database Approach, In proceedings of 5th Symposium on Spatial Databases, Berlin, Germany, 1997.

Han J., M. Kamber, Data Mining. Concepts and Techniques, Academic Press Ed. 2001.

Koperski K., J. Han, N. Stefanovic, An Efficient Two-Step Method for Classification of Spatial Data, In proceedings of International Symposium on Spatial Data Handling (SDH'98), p. 45-54, Vancouver, Canada, July 1998.

Koperski K., J. Adhikary, J. Han, Knowledge Discovery in Spatial Databases: Progress and Challenges, Proc. SIGMOD Workshop on Research Issues in Data Mining and Knowl-

edge Discovery Technical Report 96-08, University of British Columbia, Vancouver, Canada, 1996.
Kramer S., N. Lavrac, P. Flach, Propositionalization approaches to relational data mining, in relational data mining, Dzeroski S., Springer Edition, p- 262- 291, 2001.
Laurini R., D. Thompson, Fundamentals of Spatial Information Systems, Academic Press, London, UK, 3rd printing, 1994.
Longley P.A., M.F. Goodchild, D.J Maguire, D.W Rhind, *Geographical Information Systems, Principles and Technical Issues*, John Wiley & Sons, Inc., 2nd Edition, 1999.
Malerba D., F.A. Lisi, An ILP Method for Spatial Association Rule Mining. In A. Knobbe and D. van der Wallen (Eds.), Notes of the ECML/PKDD 2001 Workshop on Multi-Relational Data Mining, 18-29, Germany Freiburg, 2001.
Ceci M., A. Appice, D. Malerba, Spatial Associative Classification at Different Levels of Granularity: A Probabilistic Approach, in J.-F. Boulicaut, F. Esposito, F. Giannotti, & D. Pedreschi (Eds.), Knowledge Discovery in Databases: PKDD 2004, Lecture Notes in Artificial Intelligence, 3202, 99-111, Springer, Berlin, Germany, 2004.
Muggleton S., *Inductive Logic Programming*, New Generation Computing Edition, 295-318, 1991.
Lavrac N., S. Dzeroski, *Inductive logic programming. Techniques and applications*. Edition Ellis Horwood, 3-38, New York, 1994.
Quinlan J.R., Induction of Decision Trees, *Machine Learning (1)*, 82 - 106, 1986.
Quinlan J.R., *C4.5: Programs for machine learning*, Morgan Kaufmann, 1993.
Sanders L., *L'analyse statistique des données en géographie*, GIP Reclus, 1989.
Shashi S., C. Sanjay, *Spatial Databases: A Tour*, Prentice Hall, 2003.
Shaw G., D. Wheeler, *Statistical Techniques in Geographical Analysis*, Edition David Fulton, London, 1994.
Tobler W.R., Cellular geography, In Gale S, Olsson G, In Phylosophy in Geography Edition, Dortrecht, Reidel, 379-86, 1979.
Van Laer W., L. De Raedt, How to Upgrade Propositional Learners to First Order Logic: a Case Study, 2000.
Zeitouni K., *Le data mining spatial*, Numéro spécial, Revue internationale de géomatique, Edition Hermès, Vol 9, 4 (99), Avril 2000.
Zeitouni K., L. Yeh, M.A. Aufaure, Join indices as a tool for spatial data mining, Int. Workshop on Temporal, Spatial and Spatio-Temporal Data Mining, In Artificial Intelligence n° 2007, Springer, 102-114, Lyon, France, September 12-16, 2000.
Zighed A., R. Ricco, *Graphes d'induction - Apprentissage et Data Mining*, Edition Hermès Sciences, 2000.

Abstract.

Spatial data mining requires the analysis of the interactions in space. The conventional data mining algorithms do not support well this type of analysis. We present in this paper an approach based on inductive logic programming (ILP). It is based on two ideas. The first one consists in materializing these interactions using distance tables, so that the spatial data mining problem is reduced to relational data mining problem. The second consists in transforming data into first order logic, and then applying the inductive logic programming methods. This paper details this approach, and describes its application to the supervised classification by spatial decision tree. It shows also some experimentation results in the shellfish contamination analysis in Thau lagoon.