Vers une discrétisation locale pour les treillis dichotomiques

N. Girard, K. Bertet et M. Visani

L3i - Laboratoire Informatique, Image et Interaction
Pôle Sciences et Technologie Avenue Michel Crépeau
17042 La Rochelle Cedex 1
{ngirar02, kbertet, mvisani}@univ-lr.fr

RÉSUMÉ. *Dans cet article, nous rappelons la méthode de classification supervisée Navigala, que nous avons développée pour de la reconnaissance de symboles détériorés. Elle repose sur une navigation dans un treillis de Galois similaire à une navigation dans un arbre de décision. Les treillis manipulés par Navigala sont des treillis dits dichotomiques, dont nous décrivons dans ce papier les propriétés et les liens structurels avec les arbres de décision. La construction du treillis de Galois oblige à une étape préalable de discrétisation des données continues (discrétisation globale), ce qui n'est généralement pas le cas de l'arbre de décision qui procède à cette discrétisation au cours de sa construction (discrétisation locale). Utilisée comme prétraitement, la discrétisation détermine les concepts et la taille du treillis, lorsque l'algorithme de génération est directement appliqué sur ces données discrétisées. Nous proposons donc un algorithme de discrétisation locale pour la construction du treillis dichotomique ce qui pourrait nous permettre de mettre en œuvre une méthode d'élagage en cours de génération et ainsi d'améliorer les performances du treillis et éviter le sur-apprentissage.*

MOTS-CLÉS : *Treillis de Galois ; treillis dichotomique ; classification ; arbre de classification ; reconnaissance de symboles bruités.*

1. Introduction

La reconnaissance d'objets dans des images repose généralement sur deux étapes principales : l'extraction de signatures et la classification supervisée. Nous nous intéressons à la partie classification supervisée de ce processus. Parmi les nombreuses approches de la littérature, les approches symboliques offrent de la lisibilité et présentent l'avantage d'être intuitives, ce qui permet une meilleure compréhension des données. Nous nous intéressons aux deux méthodes symboliques que sont l'arbre de décision et le treillis de Galois lorsqu'il est utilisé comme classifieur. Le treillis de Galois ou treillis des concepts utilisé depuis une vingtaine d'année en classification supervisée donne des résultats comparables aux méthodes standards. C'est un graphe dont les nœuds sont des concepts. En classification, il existe de nombreuses méthodes utilisant le treillis de Galois, la plupart pour sélectionner des concepts les plus pertinents pour la tâche de classification (généralement menée à l'aide d'un classifieur tel que les K-PPV ou le classifieur Bayésien) [MEP 05, OOS 88, SAH 95]. Nous avons développé la méthode Navigala [GUI 07] qui utilise quant à elle la structure complète du treillis pour reconnaître des images détériorées de symboles par navigation dans son diagramme de Hasse [1] à partir de la racine et de manière similaire à l'arbre de décision. Différemment de l'arbre, le treillis propose plusieurs chemins vers un concept donné, ce qui lui confère une meilleure robustesse vis-à-vis du bruit [GUI 06]. De par sa construction à partir de données continues nécessitant une discrétisation, Navigala manipule des treillis dits *dichotomiques* qui sont structurellement proches des arbres de décision. Tandis que pour l'arbre [BRE 84, QUI 86, RAK 05], la discrétisation des données s'effectue le plus souvent au fur et à mesure de la construction (discrétisation locale), la construction du treillis nécessite généralement une phase préalable de discrétisation (discrétisation globale), qui détermine complètement les concepts et la taille du treillis, lorsque l'algorithme de génération est directement appliqué sur ces données discrétisées. Dans cet article, nous proposons un algorithme de discrétisation locale, menée au fur et à mesure de

1. Le diagramme de Hasse du treillis de Galois est le graphe de sa réduction réflexive et transitive.

la construction du treillis, ce qui pourrait nous permettre de mettre en œuvre une méthode d'élagage en cours de génération et ainsi d'améliorer les performances du treillis et éviter le sur-apprentissage.
Ce papier est organisé comme suit. Dans la partie 2, nous décrivons la méthode Navigala ainsi que les treillis dichotomiques et leurs liens structurels avec les arbres de décision. Dans la partie 3 nous présentons notre algorithme.

2. Contexte

2.1. *La méthode Navigala*

La méthode de classification supervisée Navigala a été développée pour la reconnaissance d'images de symboles issus de documents techniques. Cependant, il s'agit d'une méthode pouvant être utilisée dans un contexte plus large de classification supervisée d'objets décrits par des vecteurs numériques de taille fixe. On y retrouve les trois étapes classiques de *préparation des données, d'apprentissage supervisé et de classement de nouveaux exemples*.
Les signatures extraites des images sont des vecteurs numériques (ie à chaque image correspond un vecteur de caractéristiques numériques) de *taille fixée*. Ces vecteurs sont stockés dans une table de données regroupant les images, leurs caractéristiques et leur classe. La préparation des données consiste en une discrétisation des données continues qui permet de créer des intervalles disjoints de valeurs. La table discrète est ensuite binarisée. Ainsi pour une caractéristique donnée issue de la signature, chaque objet n'est associé qu'à un seul intervalle (appelé attribut) issu de cette caractéristique. Le *critère d'arrêt* de la discrétisation est la séparation des classes, chaque classe se différenciant alors des autres par au moins un attribut (sauf dans le cas où les signatures de deux objets appartenant à des classes différentes sont identiques, cette séparation ne pouvant alors évidemment pas être atteinte). Les caractéristiques non discrétisées au cours du traitement ne sont pas intégrées dans la table binaire, lui conférant ainsi une propriété de réduction encore appelée *sélection de caractéristiques*. Cette discrétisation se faisant en amont de la construction du classifieur, il s'agit d'une *discrétisation globale*. Dans le cas de Navigala, trois *critères de coupe* supervisés ou non ont été testés : la distance maximum, l'entropie, le critère de Hotelling [GUI 07]. Les expérimentations ont montré que le critère de Hotelling était le plus efficace dans notre contexte applicatif où la dispersion à l'intérieur des classes peut être importante (présence de bruit). La table binaire obtenue à l'issue de la phase de discrétisation des données continues est appelée *contexte*. Le contexte se définit par un triplet $(O, I, (f, g))$ où O est un ensemble d'objets, I est un ensemble d'attributs et (f, g) est une correspondance de Galois [2] entre objets et attributs. Le treillis de Galois est constitué d'un ensemble K de *concepts formels* muni d'une *relation d'ordre* [3] $\leq$:

- Un *concept formel* est un sous-ensemble maximal d'objets associés à un même sous-ensemble maximal d'attributs : $\forall A \in O$ et $\forall B \in I$ le couple (A, B) est un concept formel $\Leftrightarrow f(A) = B$ et $g(B) = A$.
- La *relation d'ordre* $\leq$ est définie pour deux concepts (A_1, B_1), (A_2, B_2) par :
  $(A_1, B_1) \leq (A_2, B_2) \Leftrightarrow A_1 \supseteq A_2 \Leftrightarrow B_1 \subseteq B_2$.

Ainsi défini, le *treillis de Galois* $(K, \leq)$ possède la propriété de treillis, c'est à dire que pour deux concepts de K, il existe un unique plus petit successeur commun et un unique plus grand prédécesseur commun. Il possède donc un *concept minimum* noté $\perp = (O, f(O))$ et un *concept maximum* noté $\top = (g(I), I)$. On distingue les *concepts finaux* qui sont les concepts vérifiant un critère de pureté concernant les classes des objets qui les composent et qui sont étiquetés par la classe majoritaire parmi ces objets. Notons que plusieurs concepts finaux peuvent être associés à la même classe.
Le classement d'un nouvel exemple se fait par navigation dans le diagramme de Hasse du treillis, à partir du concept minimum et jusqu'à un concept final par validation d'intervalles, comme lors de la navigation dans un arbre de décision. Le nouvel objet sera donc classé dans la classe-étiquette du concept final ainsi atteint. L'avantage de la structure de treillis par rapport à l'arbre de classification est la multiplicité des chemins menant à un même concept final, ce qui lui confère une meilleure robustesse vis à vis du bruit.

2. f associe à un ensemble d'objets leurs attributs communs, g associe à un ensemble d'attributs les objets qui possèdent ces attributs
3. Une relation d'ordre est une relation transitive, antisymétrique et réflexive

2.2. *Les treillis dichotomiques et les arbres de décision*

Lorsque chaque objet est décrit par un vecteur de caractéristiques, la phase de discrétisation permet d'obtenir une table binaire vérifiant une propriété d'exclusivité mutuelle entre attributs. En particulier, les intervalles créés lors de la discrétisation d'une même caractéristique sont disjoints, donc mutuellement exclusifs. A partir de cette propriété de la table binaire nous définissons le treillis dichotomique associé à cette table.

Définition 1 *Un treillis est dit* dichotomique *lorsqu'il est défini pour une table où il est toujours possible d'associer à un attribut binaire x un ensemble non vide $\overline{X}$ d'attributs binaires (avec $x \notin \overline{X}$) tel que les attributs de $\{x\} \cup \overline{X}$ soient mutuellement exclusifs.*

Comme cela a été démontré dans [GUI 08], les treillis dichotomiques sont sup-*pseudo-complémentés*[4] alors que les treillis sup-pseudo-complémentés ne sont pas toujours dichotomiques. De plus, lorsque chaque objet est associé à un vecteur de caractéristiques de même longueur, comme dans Navigala, les treillis dichotomiques possèdent la propriété de *co-atomisticité*[5]. Les treillis dichotomiques possèdent des liens structurels forts avec les arbres de décision :

1. Tout arbre de décision est inclus dans le treillis dichotomique, lorsque ces deux structures sont construites à partir des mêmes attributs binaires.

2. Tout treillis dichotomique est la fusion de tous les arbres de décision possibles lorsque ces structures sont construites à partir des mêmes attributs binaires.

Ainsi, lorsque les arbres de décision et le treillis de Galois sont définis à partir de la même table discrétisée, ils ont des liens structurels forts. Mais généralement, la discrétisation globale de la table des données n'est nécessaire que pour la construction du treillis de Galois, car la plupart des arbres de décision procèdent à la discrétisation des données au cours de leur construction (*discrétisation locale*). La discrétisation locale consiste à choisir en chaque nœud la segmentation optimale localement, qui permettra de discriminer au mieux les objets du nœud courant selon leur classe. Nous proposons dans la partie 3 un algorithme de construction du treillis à partir de données continues par discrétisation locale, ce qui pourrait nous permettre de mettre en œuvre une méthode d'élagage en cours de génération et ainsi d'améliorer les performances du treillis et éviter le sur-apprentissage.

3. Algorithme de discrétisation locale

Nous proposons de façon similaire à l'arbre de décision l'algorithme 1 de discrétisation locale pour la construction d'un treillis de Galois à la fois dichotomique et co-atomistique (ie. issu de données décrites par des vecteurs numériques de même longueur).
L'étape d'initialisation génère, pour chaque caractéristique de la table, un intervalle (appelé attribut) contenant l'ensemble des valeurs observées. La table discrète ainsi composée d'intervalles est binarisée. On initialise l'ensemble des concepts finaux CF avec le concept minimum $\perp$.
Comme pour la division d'un nœud ne vérifiant pas le critère d'arrêt dans l'arbre de décision, nous sélectionnons parmi les attributs B_i des co-atomes (A_i,B_i) de CF ne vérifiant pas le critère d'arrêt S, l'intervalle I et son point de coupe c_i selon le critère de coupe C (C pouvant être par exemple le critère de Hotelling). Ceci nous permet de segmenter I pour obtenir I_1 et I_2 deux intervalles disjoints, puis de remplacer dans la table de données I par I_1 et I_2. L'ensemble CF des concepts finaux du treillis associé à T est ensuite calculé pour pouvoir réitérer ce processus jusqu'à ce que tous les concepts finaux contenus dans CF vérifient un critère d'arrêt S similaire à celui qui peut être mis en œuvre pour les arbres (généralement mesure de pureté ou nombre minimum d'objets dans chacun des concepts de CF). Le treillis étant co-atomistique, les concepts finaux sont les co-atomes et s'obtiennent en calculant les prédécesseurs immédiats du concept maximum $\top$ avec par exemple une adaptation de la fonction successeurs immédiats de Bordat. Il n'est donc pas nécessaire de calculer le treillis à chaque itération.

4. Un treillis est sup-pseudo-complémentés lorsque pour tout concept (A, B), il existe toujours un *concept complémentaire* (A', B') tel que : $(A, B) \vee (A', B') = \top = (\emptyset, I)$

5. les concepts finaux sont des co-atomes, les co-atomes d'un treillis sont les concepts dont le plus petit successeur commun est l'élément maximum $\top$

Algorithme 1 : Construction d'un treillis par discrétisation locale

Entrées :
- Ensemble de données $(O_i, V_{ij})_{i\in\{1...n\},j\in\{1...p\}}$, chacun des n objets O_i étant décrit par un vecteur de p caractéristiques $(V_{ij})_{j\in\{1...p\}}$
- $\mathcal{S}$: critère d'arrêt et $\mathcal{C}$: critère de coupe

Sorties : TG : treillis de Galois de la table discrétisée

Initialiser une table binaire T avec :
- sur chaque ligne : un objet O_i
- sur chaque colonne : l'intervalle I_j contenant toutes les valeurs observées d'une caractéristique V_j. Chacun des intervalles I_j devient alors un attribut binaire, partagé par tous les objets O_i

Initialiser CF avec $\perp$;

tant que $\exists(A, B) \in CF$ *tel que* $!S((A, B))$ **faire**

  Sélectionner selon $\mathcal{C}$ le point de coupe optimum c_{I^*} associé à un intervalle optimum I^* parmi les attributs B_k des concepts $(A_k, B_k) \in CF$ tel que $!S((A_k, B_k))$;
  Découper I^* en deux intervalles disjoints I_1 et I_2 selon c_{I^*} ;
  Remplacer I^* par I_1 et I_2 dans T, puis mettre à jour T en conséquence ;
  CF = co-atomes du treillis associé à T = prédécesseurs immédiat du concept maximum ;

Calculer le treillis TG de T ; **retourner** TG ;

4. Conclusion et Perspectives

Après avoir introduit le contexte de cette étude et décrit la méthode Navigala que nous avons développée, cet article présente les treillis dichotomiques, leurs propriétés et leurs liens structurels forts avec les arbres de décision. Puis un algorithme de discrétisation locale pour la construction du treillis de Galois est proposé. Cet algorithme est inspiré du traitement des données continues lors de la construction de la plupart des arbres de décision. Avec cet algorithme nous construisons un treillis de Galois complet avec une discrétisation locale. Nous sommes en train de mettre en œuvre un protocole expérimental associé à cet algorithme, et espérons avoir rapidement des résultats expérimentaux plus poussés. Une première perspective serait de procéder à l'élagage en cours de génération du treillis ainsi construit pour en améliorer les performances. Une deuxième perspective consisterait en une génération incrémentale de l'ensemble des co-atomes.

5. Bibliographie

[BRE 84] BREIMAN L., FRIEDMAN J., OLSHEN R., STONE C., *Classification and regression trees*, Wadsworth Inc., 1984.

[GUI 06] GUILLAS S., BERTET K., OGIER J.-M., A Generic Description of the Concept Lattices' Classifier : Application to Symbol Recognition, vol. 3926, 2006, p. 47-60, Lecture Notes in Computer Science, Revised and extended version of paper first presented at Sixth IAPR International Workshop on Graphics Recognition (GREC'05).

[GUI 07] GUILLAS S., Reconnaissance d'objets graphiques détériorés : approche fondée sur un treillis de Galois, Thèse de doctorat, Université de La Rochelle, 2007.

[GUI 08] GUILLAS S., BERTET K., VISANI M., OGIER J.-M., GIRARD N., Some Links Between Decision Tree and Dichotomic Lattice, *Proceedings of the Sixth International Conference on Concept Lattices and Their Applications*, CLA 2008, October 2008, p. 193-205.

[MEP 05] MEPHU-NGUIFO E., NJIWOUA P., Treillis des concepts et classification supervisée, *Technique et Science Informatiques, RSTI*, vol. 24, n^o 4, 2005, p. 449-488, Hermès - Lavoisier, Paris, France.

[OOS 88] OOSTHUIZEN G., The use of a Lattice in Knowledge Processing, PhD thesis, University of Strathclyde, Glasgow, 1988.

[QUI 86] QUINLAN J., Induction of Decision Trees, *Machine Learning*, vol. 1, 1986.

[RAK 05] RAKOTOMALALA R., Arbres de Décision, *Revue MODULAD*, vol. 33, 2005.

[SAH 95] SAHAMI M., Learning Classification Rules Using Lattices, LAVRAC N., WROBEL S., Eds., *Proceedings of European Conference on Machine Learning, ECML'95*, Heraclion, Crete, Greece, April 1995, p. 343-346.