# Complémentarité de deux méthodes de classification pour la construction de treillis de concepts à partir de données numériques

Mehdi Kaytoue [1] Zainab Assaghir [1,2] Nizar Messai [1] Amedeo Napoli [1]

[1] Laboratoire Lorrain de Recherche en Informatique et ses Applications (LORIA) – UMR 7503
Campus Scientifique - BP 239 - 54506 Vandoeuvre-lès-Nancy Cedex
[2] Institut National de la Recherche Agronomique (INRA)
28, rue de Herrlisheim, B.P. 20507 - 68021 Colmar

*kaytouem@loria.fr, zassaghir@colmar.inra.fr, messai@loria.fr, napoli@loria.fr*

## Résumé

*Ce papier porte sur l'application à des données numériques de deux méthodes de classification s'appuyant sur l'analyse formelle de concepts. La première méthode de classification utilise une similarité entre objets tandis que la seconde considère une "structure de patrons". Nous détaillons ces deux méthodes et mettons en évidence leur proximité. En fait, et c'est là l'objectif de ce papier, cette étude parallèle aide à comprendre comment traiter des données complexes avec des treillis de concepts. Nous expliquons comment la deuxième méthode obtient les mêmes résultats bénéficiant d'algorithmes efficaces tout en traitant le problème des valeurs manquantes. Une expérience sur des données réelles en agronomie nous sert de première validation pour l'aide à la décision de pratiques agricoles.*

## Mots Clef

Analyse formelle de concepts, classification, similarité, valeurs manquantes

## Abstract

*This paper holds on the application to numerical data of two methods based on formal concept analysis. One classification method uses a similarity on objects while the other considers so-called pattern structures. We detail both methods and show their proximity. We explain how the second can obtain same results with an existing framework and handle missing-values. An experiment on real-world data is carried out for helping agricultural practices decision.*

## Keywords

Formal concept analysis, classification, similarity, missing values

## 1 Introduction

De nombreux problèmes de classification se formalisent à l'aide de *contextes formels* mettant en relation des objets et leurs attributs [2, 8]. Un contexte formel est représenté par une table binaire où le fait qu'un attribut en colonne est attaché à un objet en ligne est renseigné par une croix $\times$ (table 2). Une *connexion de Galois* permet de classifier au sein de *concepts formels* des ensembles d'objets partageant un même ensemble maximal d'attributs et vice-versa. Ces concepts peuvent être ordonnés pour former un *treillis de concepts* dans le cadre de l'analyse formelle de concepts (AFC) [8]. Ainsi, l'AFC permet de construire une structure mathématique qui sert de support à la découverte de connaissances [19]. L'AFC est un formalisme bien établi, pourvu d'outils pour la construction, visualisation et interprétation des treillis de concepts [8, 14].

Dans de nombreux cas, les données ne sont pas binaires mais *complexes* et *hétérogènes* : un objet ne partage pas un attribut ou non, mais prend une ou plusieurs valeurs pour chaque attribut. Ces valeurs peuvent être des nombres, des graphes, des intervalles, ... Classifier un objet au sein d'un concept n'est pas direct : les données sont échelonnées (binarisées) [8]. Dans ce papier par exemple, nous considérons les objets décrits par un nombre ou un intervalle de nombres pour chaque attribut (table 1). Un échelonnage classique consiste à définir des règles du type : "un objet possède un attribut ssi sa valeur pour cet attribut est supérieure à un seuil donné". Cette règle permet par exemple de construire la table binaire 2. Encore, "un objet possède un attribut ssi sa valeur pour cet attribut est comprise dans un intervalle donné *a priori*" [12]. Cependant, un échelonnage n'est pas toujours fiable ni facile à définir [7, 11]. La plupart du temps, cette procédure dépend d'un expert du domaine concerné (si disponible) et conduit à une perte d'information. De plus, partitionner le domaine de valeurs des attributs est arbitraire et strict : une valeur est soit dans un intervalle, soit dans un autre.

Il est plus naturel de classifier un nouvel objet dans un groupe d'objets dont les valeurs d'attributs sont "proches". Cela permet d'éviter l'échelonnage et d'utiliser une *similarité* intuitive entre les objets complexes. Ainsi les auteurs de [16, 17] définissent l'AFC par similarité que nous dénommons FCAS dans ce papier, pour *FCA guided by Si-*

*milarity*. Ils proposent de considérer une similarité entre objets "numériques" pour mettre en œuvre le partage d'attributs et construire un treillis de concepts : deux valeurs d'attributs sont similaires ssi leur différence n'excède pas un seuil donné. Parce que cette relation de similarité n'est pas transitive [1], les auteurs montrent qu'elle pose un problème pour ordonner partiellement les concepts. Ils proposent donc de rendre la similarité transitive. C'est alors une similarité mutuelle ou deux à deux (*pairwise similarity* en anglais). Des exemples d'utilisation du treillis obtenu sont donnés dans [16, 17] pour la recherche et l'intégration de ressources génomiques sur le Web. Cependant, FCAS ne permet pas de traiter des données importantes comme le reconnaissent les auteurs.

Parallèlement, les auteurs de [11] proposent de munir les *structures de patrons*, définies dans [7], d'une algèbre d'intervalles pour construire un treillis de concepts à partir de données numériques. Nous appelons ce procédé PSI dans ce papier pour *pattern structures for intervals*. Une structure de patrons comprend un ensemble d'objets et leurs descriptions partiellement ordonnées à l'aide d'un opérateur dit de *similarité* [7]. D'une certaine manière, PSI "généralise" la notion d'attributs à des structures de valeurs complexes et le partage revient à vérifier dans quelle mesure deux objets ont des "descriptions similaires". Des travaux semblables existent pour des objets décrits par des formules logiques [4, 6], par des intervalles [9], et par des graphes [15]. Les structures de patrons sont définies en complète adéquation avec le formalisme de l'AFC et bénéficient ainsi d'outils existants et performants comme les nombreux algorithmes discutés dans [14].

En fait, et c'est là l'objectif de ce papier, l'étude parallèle des deux méthodes FCAS et PSI aide à comprendre les données complexes avec la construction de treillis de concepts. PSI est basé sur un formalisme en totale adéquation avec l'AFC et FCAS apporte des notions intuitives sur la similarité et l'interprétation des treillis résultants.

Le papier est organisé comme suit. Après l'introduction de l'AFC en section 2, nous présentons les deux méthodes FCAS et PSI sur un exemple commun (sections 3 et 4). Ensuite, la section 5 compare les deux méthodes, discute de la notion de similarité et montre comment les structures de patrons obtiennent le même résultat que la FCAS de manière plus efficace. Une seconde contribution permet de considérer les valeurs manquantes. Enfin, une expérience en agronomie vient conforter les notions présentées pour l'aide à la décision de pratiques agricoles.

# 2 Analyse formelle de concepts

Soient $G$ un ensemble d'objets, $M$ un ensemble d'attributs et $I$ une relation binaire définie sur le produit cartésien $G \times M$, i.e. $I \subseteq G \times M$. Le triplet $(G, M, I)$ est appelé *contexte formel* : pour un objet $g \in G$ et un attribut $m \in M$, $(g, m) \in I$ ou $gIm$ signifie que l'objet $g$ "possède" l'attribut $m$, ou que l'attribut $m$ est rattaché à $g$. La table 2 représente un contexte formel. Chaque ligne (resp. colonne) correspond à un objet de $G$ (resp. attribut de $M$). Ici l'objet $g_1$ possède l'attribut $m_1 \leq 8$, l'objet $g_2$ possède les attributs $m_1 \leq 8$ et $m_2 \leq 19$ . Les deux opérateurs de dérivation $(.)'$

$$A' = \{m \in M \mid \forall g \in A \subseteq G : gIm\},$$
$$B' = \{g \in G \mid \forall m \in B \subseteq M : gIm\}$$

définissent une *connexion de Galois* entre l'ensemble des parties de $G$, noté $2^G$, et l'ensemble des parties de $M$. Le premier opérateur $(.)'$ associe à un ensemble d'objets $A$ l'ensemble des attributs que les objets de $A$ possèdent en commun. Par exemple, $\{g_1\}' = \{m_1 \leq 8\}$. De manière duale, le second opérateur associe à un ensemble d'attributs $B$, l'ensemble des objets qui possèdent au moins tous les attributs de $B$. Par exemple, $\{m_1 \leq 8\}' = \{g_1, g_2\}$. Un couple $(A, B)$ avec $A' = B$ et $B' = A$ est un *concept formel*, par exemple $(\{g_1, g_2\}, \{m_1 \leq 8\})$. $A$ et $B$ sont respectivement appelés l'*extension* et l'*intension* du concept.

L'ensemble de tous les concepts du contexte $(G, M, I)$ est ordonné par la relation suivante : $(A_1, B_1) \leq (A_2, B_2) \Leftrightarrow A_1 \subseteq A_2$ $(\Leftrightarrow B_2 \subseteq B_1)$. Ainsi, $(A_1, B_1)$ est un sous-concept de $(A_2, B_2)$ et ce dernier est un super-concept de $(A_1, B_1)$. Muni de cet ordre, l'ensemble de tous les concepts forme un treillis complet appelé *treillis de concepts* du contexte $(G, M, I)$. La figure 1 représente le digramme du treillis de notre exemple. Cette représentation à *étiquetage réduit* ne présente chaque objet et attribut qu'une seule fois. Un concept est représenté par un nœud. Son extension (resp. intension) est composée de tous les objets (resp. attributs) qui sont associés à lui et à ses sous-concepts (resp. super-concepts).

# 3 AFC guidée par une similarité

FCAS est une méthode basée sur l'AFC qui permet de considérer une similarité entre objets pour construire une treillis de concepts sans échelonnage à partir d'un contexte multi-valué [16, 17]. La table 1 présente un contexte multi-valué. Il s'agit d'un quadruplet $(G, M, W, I)$ où $G$ et $M$ désignent encore l'ensemble des objets et des attributs, et $W$ désigne l'ensemble des valeurs d'attributs. Nous considérons les contextes où $W$ est composé uniquement d'intervalles ou de nombres. La relation $I$ est définie sur le produit Cartésien $G \times M \times W$. Par exemple, "l'objet $g_1$ prend la valeur $0.3$ pour l'attribut $m_3$" s'écrit : $(g_1, m_3, 0.3)$ ou encore $m_3(g_1) = 0.3$. Nous présentons ici d'abord une similarité intuitive entre intervalles et les problèmes qu'elle pose, puis la similarité par couples d'objets ou similarité d'objets deux à deux utilisée par la connexion de Galois qui permet la construction du treillis de concepts.

## 3.1 Similarité entre intervalles

En AFC classique, un attribut $m$ est partagé par un ensemble d'objets $A$ ssi chaque objet $g \in A$ possède l'attri-

1. Le terme de "similarité" est conservé ici pour des raisons "historiques" [16, 17]. Le terme de "tolérance" utilisé dans [13] conviendrait peut être mieux car il désigne une relation réflexive et symétrique seulement.

| | $m_1$ | $m_2$ | $m_3$ |
|---|---|---|---|
| $g_1$ | $[2,4]$ | $[25,29]$ | $0.3$ |
| $g_2$ | $[4,8]$ | $19$ | $0.1$ |
| $g_3$ | $[10,15]$ | $29$ | $0.5$ |
| $g_4$ | $[9,13]$ | $17$ | $0.5$ |
| $g_5$ | $[8,13]$ | $[17,19]$ | $0.3$ |
| $g_6$ | $[9,15]$ | $[14,19]$ | $[0.5,0.7]$ |

TABLE 1 – Données d'intervalles

| I | $m_1 \leq 8$ | $m_2 \leq 19$ | $m_3 \geq 0.5$ |
|---|---|---|---|
| $g_1$ | × | | |
| $g_2$ | × | × | |
| $g_3$ | | | × |
| $g_4$ | | × | × |
| $g_5$ | | × | |
| $g_6$ | | × | × |

TABLE 2 – Un contexte formel

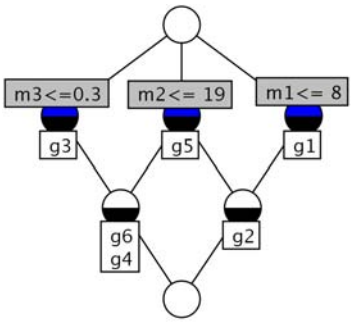


FIGURE 1 – Un treillis de concepts

but $m$. Lorsque les objets sont décrits par des intervalles, ce partage n'est pas si simple, il faut un "échelonnage" pour obtenir un contexte binaire. L'échelonnage consiste à découper un attribut en une famille d'attributs puis à cocher les cases correspondantes pour chaque objet. Un exemple donne la table 2 pour le contexte multi-valué de la table 1. Une autre façon consiste à considérer une similarité entre les nombres : deux valeurs sont similaires si leur différence n'est pas significative. Soit $\theta$ un seuil dit de similarité, deux intervalles $[\alpha_i, \beta_i]$ et $[\alpha_j, \beta_j]$ sont dits similaires ssi

$$[\alpha_i, \beta_i] \simeq_\theta [\alpha_j, \beta_j] \Leftrightarrow max(\beta_i, \beta_j) - min(\alpha_i, \alpha_j) \leq \theta.$$

Le seuil $\theta$ exprime la variation maximale autorisée entre deux valeurs similaires. Par exemple, quand $\theta = 6$, $[2,4] \simeq_\theta [4,8]$ mais $[2,4] \not\simeq_\theta [9,13]$. Quand $\theta = 11$ ces trois intervalles sont similaires. Il est important de remarquer que l'opérateur $\simeq_\theta$ n'est pas transitif et ne détermine ainsi pas de relation d'équivalence : avec $\theta = 9$, $[2,4] \simeq_\theta [4,8]$, $[4,8] \simeq_\theta [9,13]$ mais $[2,4] \not\simeq_\theta [9,13]$.

## 3.2 Similarité entre objets

En FCAS, la similarité entre objets est définie ainsi.

- Deux objets $g_1$ et $g_2$ partagent un attribut $m$ ssi $m(g_1) \simeq_\theta m(g_2)$.
- Un ensemble d'objets $A \subseteq G$ partage un attribut $m$ si *toute paire d'objets* de $A$ partage $m$.
- Un ensemble d'objets $A \subseteq G$ partage un ensemble d'attributs $B \subseteq M$, ssi toute paire d'objets partage tous les attributs de $B$. Alors, $A$ est dit *valide* pour $B$.

Quand un ensemble d'objets partage un ensemble d'attributs, les objets sont similaires deux à deux pour cet attribut. Par exemple, avec $\theta = 6$ pour l'attribut $m_1$, $\theta = 4$ pour $m_2$, et $\theta = 0.2$ pour $m_3$, alors les objets de $\{g_3, g_4, g_6\}$ sont similaires deux à deux pour les attributs $m_1$ et $m_3$.
Une connexion de Galois peut être définie. Un opérateur associe à un ensemble d'objets les attributs qu'ils partagent et pour chacun de ces attributs l'intervalle qui caractérise la similarité entre les objets (voir ci-après). Cet opérateur retourne donc une liste de couples (attribut,intervalle). De manière duale, le second opérateur associe à une liste de couples l'ensemble des objets qui la partagent. Alors, les ensembles d'objets et leur description, fermés pour la connexion ainsi établie, forment des concepts formels.

## 3.3 Ensembles maximaux d'objets similaires

Comme c'est le cas en AFC standard, il faut trouver un ensemble d'objets maximal qui partage un ensemble d'attributs maximal. Pour exemple, $\{g_3, g_6\}$ est valide tout comme $\{g_3, g_4, g_6\}$ pour les mêmes attributs $m_1$ et $m_3$. C'est ce dernier ensemble qu'il faut conserver et qui va déterminer un concept formel. Pour calculer un ensemble d'objets valide et maximal, l'idée est pour un ensemble d'objets donné de (i) chercher tous les objets similaires, et (ii) de supprimer les paires d'objets non-similaires entre eux. Enfin, (iii) il est nécessaire pour la connexion de Galois de retourner la description de la similarité entre ces objets. (i) et (ii) se voient comme une fermeture en morphologie mathématique qui consiste en (i) une dilatation puis (ii) une érosion par un élément structurant qui caractérise $\theta$ [10].

**(i) Ensemble d'objets "atteignables".** L'ensemble des objets atteignables à partir d'un ensemble d'objets $A$ valide pour un attribut $m$ est donné par la fonction suivante.

$$AT(A, m, \theta) = \{g_i \in G \mid m(g_i) \simeq_\theta m(g), \forall g \in A\}$$

Le fait que $A$ doit être valide est expliqué ci-après. L'ensemble des objets atteignables à partir de $A$ pour un ensemble d'attributs est l'intersection des ensembles atteignables pour chaque attribut : $AT(A, B, \Theta) = \bigcap_{m \in B} AT(A, m, \theta_i)$. $\Theta$ est l'ensemble des $\theta$ pour chaque attribut, et $\theta_i \in \Theta$ est le seuil correspondant à l'attribut $m$. Par exemple, $AT(\{g_1\}, m_1, 11) = \{g_1, g_2, , g_4, g_5\}$ et $AT(\{g_1\}, \{m_1, m_3\}, \{11, 0.2\}) = \{g_1, g_2, g_4, g_5\}$.

**(ii) Ensemble valide maximal.** Puisque l'opérateur $\simeq_\theta$ n'est pas transitif, certains objets dans $AT(A, m, \theta)$ peuvent ne pas être similaires deux à deux. Une solution est de supprimer les paires d'éléments qui violent la transitivité. On obtient alors un ensemble maximal d'objets similaires deux à deux pour un attribut $m$,

$$ATvm(A, m, \theta) = AT(A, m) \backslash \{g_i, g_j \mid m(g_i) \not\simeq_\theta m(g_j)\}$$

Il n'est pas possible de supprimer un objet de $A$, c'est pourquoi $A$ doit être valide. Pour un ensemble d'attributs $B$, le calcul est $ATvm(A, B, \Theta) = \bigcap_{m \in B} ATvm(A, m, \theta_i)$.

Pour exemples, $ATvm(\{g_1\}, m_3, 0.2)$ = $\{g_1, g_5\}$ et $ATvm(\{g_1\}, \{m_1, m_3\}, \{11, 0.2\})$ = $\{g_1, g_5\}$.

**(iii) Description d'ensembles valides maximaux.** Le premier opérateur de la connexion de Galois retourne la description d'un ensemble d'objets. En FCAS, cette description est un ensemble de couples (nom d'attribut, intervalle de similarité). L'intervalle de similarité associé à $ATvm(A, m, \theta)$ est retourné par $Isim(A, m, \theta) = [min(\alpha_i), max(\beta_i)]$ avec $[\alpha_i, \beta_i] = m(g_i), g_i \in ATvm(A, m, \theta)$. Par exemple, $Isim(\{g_1, g_2\}, m_1, 6) = [2, 8]$. Quand $A$ ne partage pas $m$, alors $Isim(A, m) = \varnothing$.

### 3.4 Construction du treillis

Dans [16], il est montré que les deux opérateurs suivants forment une connexion de Galois entre $2^G$ et l'ensemble partiellement ordonné $(M \times \mathcal{I}_\Theta, \sqsubseteq)$. $\mathcal{I}_\Theta$ est l'ensemble de tous les intervalles maximaux (que retourne la fonction Isim). $\sqsubseteq$ ordonne les couples (nom d'attribut,intervalle) par inclusion d'intervalles de même attribut. Soient $A \subseteq G$ un ensemble d'objets et $B \subseteq M \times \mathcal{I}_\Theta$,

$$A^\uparrow = \{(m, Isim(A, m)) \in M \times \mathcal{I}_\Theta \mid Isim(A, m) \neq \varnothing\}$$
$$B^\downarrow = ATvm(\{g \in G \mid \forall (m, [\alpha, \beta]) \in B,\ m(g) \simeq_\theta [\alpha, \beta]\}, B)$$

$A^\uparrow$ retourne l'ensemble des attributs partagés par les objets de $A$ avec pour chacun un intervalle décrivant leur similarité (couples). $B^\downarrow$ retourne l'ensemble des objets partageant tous les attributs de $B$. Illustrons ces opérateurs avec $\theta = 6$, $\theta = 4$ et $\theta = 0.2$ pour les attributs $m_1$, $m_2$ and $m_3$ :

$$\{g_3, g_6\}^\uparrow = \{(m_1, [9, 15]), (m_3, [0.5, 0.7])\}$$
$$\{(m_1, [9, 15]), (m_3, [0.5, 0.7])\}^\downarrow = \{g_3, g_4, g_6\}.$$

$(A, B) = (\{(m_1, [9, 15]), (m_3, [0.5, 0.7])\}, \{g_3, g_4, g_6\})$ est un concept car $A^\uparrow = B$ et $A = B^\downarrow$. L'ensemble de tous les concepts est classiquement ordonné par $(A_1, B_1) \leq (A_2, B_2) \Leftrightarrow A_1 \subseteq A_2 (\Leftrightarrow B_2 \sqsubseteq B_1)$ et génère un treillis complet, dont un exemple est détaillé à la section 6 à partir de données du monde réel.

## 4 Structure de patrons et intervalles

Cette section présente une méthode de construction de treillis de concepts sans échelonnage [11]. Classiquement, la connexion de Galois met en relation les éléments du treillis des objets $(2^G, \subseteq)$ et du treillis des attributs $(2^M, \subseteq)$, et vice-versa. Ces treillis sont des ensembles partiellement ordonnés. Ainsi, pour construire un treillis de concepts où les objets ne sont pas décrits par des attributs binaires mais par des descriptions complexes (graphes, intervalles, ...), il faut définir un ordre partiel de ces descriptions. C'est là une des idées principales des *structures de patrons* définies par [7] formalisant des objets dans $G$ et leurs descriptions appelées *patrons* dans $D$. Les patrons de $D$ sont ordonnés au sein d'un inf.-demi-treillis $(D, \sqcap)$. En effet, en AFC classique, dans le treillis des attributs $(2^M, \subseteq)$, $\forall N, O \subseteq M$, alors $N \subseteq O \Leftrightarrow N \cap O = N$, par exemple avec $M = \{a, b, c\}$, $\{a, b\} \subseteq \{a, b, c\} \Leftrightarrow \{a, b\} \cap \{a, b, c\} = \{a, b\}$. L'intersection ensembliste $\cap$ possède les propriétés d'un *infimum* $\sqcap$ dans un demi-treillis. C'est de la même façon que sont ordonnés les patrons (ou patrons structurels) : soient deux patrons $c, d \in D$ alors $c \sqsubseteq d \Leftrightarrow c \sqcap d = c$. $\sqcap$ est aussi appelé opérateur de similarité, car il retourne un élément représentant la similarité "maximale" de ses arguments. Ainsi, un patron se définit comme un élément d'un inf.-demi-treillis. Une connexion de Galois générale établit une correspondance entre $(2^G, \subseteq)$ et $(D, \sqsubseteq)$, et la construction du treillis de concepts suit alors l'AFC classique [7]. Ci-dessous, nous montrons comment considérer les intervalles et les vecteurs d'intervalles comme des patrons.

### 4.1 Similarité entre intervalles

Les intervalles sont ordonnés dans un inf.-demi-treillis et sont considérés comme des patrons. L'*infimum* de deux intervalles $[a_1, b_1]$ et $[a_2, b_2]$, avec $a_1, b_1, a_2, b_2 \in \mathbb{R}$, est donné par $[a_1, b_1] \sqcap [a_2, b_2] = [min(a_1, a_2), max(b_1, b_2)]$. Cela signifie que l'*infimum* de plusieurs intervalles est le plus petit intervalle qui les contient tous. Les intervalles sont ordonnés par la relation de subsomption suivante.

$$\begin{aligned} & [a_1, b_1] \sqsubseteq [a_2, b_2] \\ \Leftrightarrow\ & [min(a_1, a_2), max(b_1, b_2)] = [a_1, b_1] \\ \Leftrightarrow\ & a_1 \leq a_2 \quad \text{et} \quad b_1 \geq b_2. \end{aligned}$$

Contrairement à l'intuition, les intervalles subsument les intervalles plus grands qui les contiennent. Par exemple, $[2, 4] \sqcap [4, 8] = [2, 8]$, d'où $[2, 8] \sqsubseteq [4, 8]$ et $[2, 8] \sqsubseteq [2, 4]$.

### 4.2 Similarité entre objets

Dans la table 1, les objets sont décrits par des vecteurs de valeurs. De manière générale, un *vecteur d'intervalles* est un vecteur à $p$ dimensions, composé uniquement de $p$ intervalles. Lorsque $e$ et $f$ sont des vecteurs d'intervalles, nous écrivons $e = \langle [a_i, b_i] \rangle_{i \in [1,p]}$ et $f = \langle [c_i, d_i] \rangle_{i \in [1,p]}$. L'*infimum* de deux vecteurs d'intervalles représente la similarité entre deux objets et est défini comme suit :

$$e \sqcap f \quad = \quad \langle [a_i, b_i] \sqcap [c_i, d_i] \rangle_{i \in [1,p]}$$

Les vecteurs doivent tous être de même taille. Un ordre canonique sur les composantes (ou dimensions) est défini. Cette définition implique que le calcul de $\sqcap$ pour des vecteurs d'intervalles est réalisé par le calcul de $\sqcap$ entre les intervalles de chaque dimension. Alors, les vecteurs d'intervalles sont partiellement ordonnés par :

$$e \sqsubseteq f \quad \Leftrightarrow \quad [a_i, b_i] \sqsubseteq [c_i, d_i],\ \forall i \in [1, p]$$

Ainsi, chaque intervalle $[a_i, b_i]$ de $e$ est subsumé par l'intervalle correspondant $[c_i, d_i]$ de $f$. Par exemple $\langle [2, 4], [2, 6] \rangle \sqsubseteq \langle [4, 4], [3, 4] \rangle$ car $[2, 4] \sqsubseteq [4, 4]$ et $[2, 6] \sqsubseteq [3, 4]$.

### 4.3 Construction du treillis

Le triplet $(G, (D, \sqcap), \delta)$ détermine une *structure de patrons* où $G$ est un ensemble d'objets, $(D, \sqcap)$ est un inf.-demi-treillis de patrons et $\delta$ est une fonction qui associe

à tout objet $g \in G$ sa description $\delta(g)$ dans $D$ [7]. Les nombres sont des intervalles et les vecteurs d'intervalles sont des patrons partiellement ordonnés au sein d'un inf.-demi-treillis : les données de la table 1 sont un exemple structure de patrons avec $G = \{g_1, \ldots, g_5\}$ et $D$ est un ensemble de vecteurs d'intervalles à trois composantes. Chaque composante correspond à un attribut ou colonne de la table. Par exemple, la description de l'objet $g_1$ est $\delta(g_1) = \langle [2,4], [25,29], [0.3,0.3] \rangle$.
Dans ce cadre, la connexion de Galois est composée de deux opérateurs. Le premier retourne la similarité d'un ensemble d'objets, i.e. l'infimum de leur description. Par exemple,

$$\begin{aligned}
&\{g_3, g_6\}^{\square} \\
&= \textstyle\bigsqcap_{g \in \{g_3, g_6\}} \delta(g) \\
&= \delta(g_3) \sqcap \delta(g_6) \\
&= \langle [10,15], [29,29], [0.5,0.5] \rangle \sqcap \langle [9,15], [14,19], [0.5,0.7] \rangle \\
&= \langle [10,15] \sqcap [9,15], [29,29] \sqcap [14,19], [0.5,0.5] \sqcap [0.5,0.7] \rangle \\
&= \langle [9,15], [14,29], [0.5,0.7] \rangle
\end{aligned}$$

Le second retourne l'ensemble maximal d'objets dont la similarité est représentée par une description en argument :

$$\begin{aligned}
&\langle [9,15], [14,29], [0.5,0.7] \rangle^{\square} \\
&= \{g \in G \mid \langle [9,15], [14,29], [0.5,0.7] \rangle \sqsubseteq \delta(g)\} \\
&= \{g_3, g_4, g_6\}
\end{aligned}$$

Ainsi, le couple $(A, d) = (\{g_3, g_4, g_6\}, \langle [9,15], [14,29], [0.5,0.7] \rangle)$ est tel que $A^{\square} = d$ et $A = d^{\square}$ : c'est un concept pour PSI. L'ensemble des concepts est ordonné de la même manière qu'en AFC par $(A_1, d_1) \leq (A_2, d_2) \Leftrightarrow A_1 \subseteq A_2$ ($\Leftrightarrow d_2 \sqsubseteq d_1$), un exemple de treillis est donné plus loin.

# 5 Complémentarité des méthodes

Nous avons détaillé deux méthodes pour la construction d'un treillis de concepts à partir de données numériques ou intervalles. Dans cette section, nous essayons de dégager les relations plutôt étroites qui existent entre les deux méthodes, relations qui ne sont pas très visibles à première vue. Nous montrons que FCAS est une méthode plus spécifique que PSI, mais aussi que PSI peut arriver au même résultat en s'appuyant sur un formalisme mathématique bien établi. Nous étudions ensuite l'idée d'objets similaires deux à deux dans le cadre des patrons structurels mais aussi le problème des données manquantes.

## 5.1 Premiers éléments

La construction du treillis repose pour les deux méthodes sur la connexion de Galois entre deux ensembles ordonnés. L'un est $(2^G, \subseteq)$ c.-à-d. l'ensemble des parties de $G$ ordonné par l'inclusion ensembliste. L'autre est un ensemble partiellement ordonné de descriptions : pour FCAS, il s'agit d'un ensemble de couples avec le nom de l'attribut et un intervalle de similarité associé ; pour PSI, il s'agit d'un vecteur d'intervalles à taille fixe dont les composantes suivent un ordre canonique. Dans les deux cas, les intervalles sont ordonnées par l'inclusion d'intervalles.
Le premier opérateur de la connexion de Galois en FCAS associe à un ensemble d'objets l'ensemble des attributs qu'ils partagent. Tout d'abord, tous les objets similaires deux à deux sont recherchés, puis $Isim$ retourne l'intervalle de similarité maximal pour chaque attribut partagé. Avec PSI, l'opérateur de similarité réalise une tâche similaire puisqu'il retourne la similarité d'un ensemble d'objets à la façon de l'opérateur $Isim$. C'est sur $\sqcap$ qu'il faut agir pour prendre en compte des types de similarité différents. Ainsi, $\sqcap$ peut se voir comme un *opérateur noyau* : il retourne un élément représentant une similarité entre ses arguments [7, 18].
Le second opérateur de la connexion de Galois en FCAS donne pour une description, ou ensemble de couples $(m, [a, b])$ avec $m \in M$ et $a, b \in \mathbb{R}$ l'ensemble des objets qui partagent ses attributs. PSI réalise un travail similaire à partir d'un vecteur d'intervalles. Puisque les deux opérateurs de la connexion de Galois sont par définition en dualité et les opérateurs pour PSI se définissent par rapport à $\sqsubseteq$ et $\sqcap$, prendre en compte une similarité comme dans FCAS avec PSI nécessite une redéfinition de $\sqcap$, ce que nous montrons ci-après. Nous procédons en deux étapes : en considérant le seuil de similarité $\theta$ (5.2), puis la similarité deux à deux (5.3).

## 5.2 Similarité entre patrons

Étant donné deux objets $g$ et $h$, la similarité entre $g$ et $h$ est donnée par $\delta(\{g, h\}) = \delta(g) \sqcap \delta(h)$. Ces trois descriptions sont ordonnées au sein d'un inf.-demi-treillis. Ce dernier est défini par l'existence pour chaque paire d'éléments qui le compose d'un infimum. Ainsi, tous les objets d'un ensemble quelconque sont "similaires" et leur "niveau" de similarité dépend du niveau de leur infimum dans le demi-treillis. Alors, considérer deux objets comme différents implique d'étendre la définition de l'infimum et d'introduire un élément * qui dénote la dissimilarité entre deux objets. L'élément * est subsumé par tout patron dans le demi-treillis de patrons, $* \sqcap d = * \iff * \sqsubseteq d$, et deux patrons $c$ et $d$ sont similaires ssi $c \sqcap d \neq *$.
Cette définition générale s'illustre avec les intervalles pour considérer la notion de similarité entre intervalles introduite en section 3.1. Soient $\theta \in \mathbb{R}$ un seuil de similarité et $[a, b]$ et $[c, d]$ deux intervalles, alors leur infimum "contraint par $\theta$" est donné par $[a, b] \sqcap_\theta [c, d] = [min(a, c), max(b, d)]$ si $max(b, d) - min(a, c) \leq \theta$, et $[a, b] \sqcap_\theta [c, d] = *$ sinon. On a bien $* \sqcap_\theta [a, b] = * \Leftrightarrow * \sqsubseteq_\theta [a, b]$ et le lien avec FCAS devient :

$$\begin{aligned}
[a, b] \sqcap_\theta [c, d] = [min(a, c), max(b, d)] &\Leftrightarrow [a, b] \simeq_\theta [c, d] \\
\text{et } [a, b] \sqcap_\theta [c, d] = * &\Leftrightarrow [a, b] \not\simeq_\theta [c, d]
\end{aligned}$$

Dans le cas où les patrons sont des vecteurs d'intervalles, les définitions de leur infimum et du test de subsomption utilisent l'opérateur contraint $\sqcap_\theta$. Un exemple de concept est alors $(\{g_3, g_4, g_6\}, \langle [9,15], *, [0.5,0.7] \rangle)$ : les objets

de l'extension sont dits similaires pour le premier et le troisième attribut. Le concept équivalent pour FCAS est $(\{g_3, g_4, g_6\}, \{(m_1, [9, 15]), (m_3, [0.5, 0.7])\})$ : seuls les attributs partagés sont décrits.
Ainsi, FCAS utilise un vocabulaire plus intuitif : il est plus évident de dire, "deux objets sont similaires ssi ils partagent des attributs" (FCAS) plutôt que "deux objets sont similaires ssi l'infimum de leur valeur pour au moins une composante de leur description n'est pas *" (IPS).

## 5.3 Patrons pour la similarité deux à deux

Si l'opérateur de similarité contraint $\sqcap_\theta$ évite de construire des intervalles violant le seuil $\theta$ comme en section 3.1, le problème est alors le même qu'en section 3.3 : il faut s'assurer qu'un intervalle est maximal, c.-à-d. qu'il représente la similarité d'un ensemble maximal d'objets similaires deux à deux.

**(i) Boule de patrons dans un demi-treillis.** Soit un demi-treillis d'intervalles $(D, \sqcap_\theta)$. L'ensemble des patrons similaires à un élément $d \in D$ est défini par une boule de centre $d$ et de rayon $\theta$ : $B(d, \theta) = \{e \in D \mid e \simeq_\theta d\}$. Cette définition repose sur le fait que deux patrons sont similaires si leur infimum n'est pas *. Par exemple, $B([0.1, 0.1], 0.2)$ = $\{[0.1, 0.1], [0.3, 0.3]\}$ si l'on considère l'attribut $m_3$ de la table 1. Les éléments de la boule $B(d, \theta)$ pour un attribut $m$ sont les descriptions de ceux de l'ensemble retourné par $AT(A, m)$ quand $A = g$ et $m(g) = d$ en FCAS.

**(ii) Intervalles maximaux représentant une similarité deux à deux.** Il s'agit maintenant de supprimer les éléments de la boule qui violent une similarité deux à deux. Intuitivement, cela implique de construire l'intervalle retourné par $Isim$ décrivant les objets de $ATvm$ en FCAS. En termes de structures de patrons, il s'agit de remplacer un patron $d$ par l'infimum de tous les intervalles $e$ de la boule $B(d, \theta)$ qui ne sont dissimilaires avec aucun autre élément $e'$ de la boule, i.e. ne violant pas la similarité par paire. La fonction $\psi$ calcule cet intervalle de la façon suivante.

$$\psi(d, \theta) = \bigsqcap_{e \in B(d,\theta)} e \sqcap_\theta d$$

en vérifiant que $\nexists e' \in B(d, \theta)$ avec $e \sqcap_\theta e' = *$. Dans l'exemple, $\psi([0.1, 0.1]) = [0.1, 0.1] \sqcap [0.3, 0.3] = [0.1, 0.3]$, pour l'attribut $m_3$ et $\theta = 0.2$. $\psi$ est une fonction qui associe à tout $d \in D$ un élément $\psi(d) \in (D, \sqcap_\theta)$ tel que $\psi(d) \sqsubseteq d$. En effet, $\psi(d)$ est l'infimum de $d$ et de tous ses intervalles similaires deux à deux. $\psi(d) \sqsubseteq d$ signifie que $\psi$ est contractive : c'est une *projection* dans le demi-treillis $(D, \sqcap_\theta)$ car aussi monotone et idempotente. De plus, toute projection d'un inf.-demi-treillis $(D, \sqcap)$ est stable pour l'infimum [7]. Ainsi, les éléments peuvent être projetés par une étape de pré-traitement avant la construction du treillis. Cette étape remplace chaque intervalle par son projeté, et réduit ainsi le nombre de valeurs différentes dans les données. Cela implique de meilleures propriétés pour le calcul : [11] montrent que ce nombre influence le

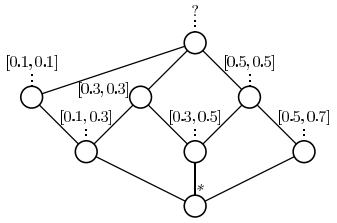


FIGURE 2 – Treillis de patrons avec les élements * et ?

plus le temps de calcul ainsi que le nombre de concepts du treillis. La FCAS ne suggère pas une telle opération, et $Isim$ est calculé chaque fois que les opérateurs de la connexion de Galois sont utilisés. Par manque de place, plus de détails sur la complexité de l'adaptation d'algorithmes standards d'AFC, et leur passage à l'échelle ne peuvent être donnés ici, mais de nombreux éléments se trouvent dans [7, 11, 14].

## 5.4 Considérer les valeurs manquantes

Considérer des valeurs manquantes dans les deux approches nécessite de les insérer dans un ordre partiel. Deux possibilités sont immédiates avec les structures de patrons : la valeur manquante (i) *subsume* ou (ii) *est subsumée* par toutes les autres. En termes de FCAS, cela se formule ainsi : la valeur manquante est (i) *similaire* ou (ii) *dissimilaire* avec toutes les autres. Les éléments du paragraphe précédent nous invitent à utiliser le formalisme des structures de patrons dans la suite.

**(i) Une valeur manquante comme le supremum de tous les éléments.** Cette approche est la plus intuitive. Puisque l'on ne connaît pas la véritable valeur, il peut s'agir de n'importe quelle autre : la valeur manquante, notée ?, subsume toutes les autres. Alors on peut ne plus restreindre $D$ à un demi-treillis $(D, \sqcap)$, mais l'étendre à un treillis $(D, \sqcap, \sqcup)$ de patrons tel que $? \in D$. $\sqcup$ est l'opération duale de $\sqcap$ et est appelée *supremum*. Cela requiert (a) de définir $\sqcup$ pour les patrons considérés, ainsi que (b) $\sqcap$ lorsqu'un argument est ?. Pour (b), puisque l'on veut $d \sqsubseteq ?$ alors, $d \sqcap ? = d$ et de façon duale $d \sqcup ? = ?$. Pour (a), l'opération est à définir pour le type de patrons en question mais reste duale de $\sqcap$. Dans le cas des intervalles, on a $[a, b] \sqcup [c, d] = [max(a, c), min(b, d)]$ ssi $min(b, d) \geq max(a, c)$ et $[a, b] \sqcup [c, d] = ?$ sinon.

**(i) Une valeur manquante comme l'infimum de tous les éléments.** Partir du constat qu'une valeur manquante n'est similaire à aucune autre (à part une autre valeur manquante) peut aussi être utile (voir section 6 pour une utilisation sur des données réelles). Cela souligne le fait que la valeur n'est pas renseignée, et qu'il ne faut pas la considérer comme une valeur inconnue. En fait, cette valeur peut être représentée par l'élément * introduit plus haut. En effet, rappelons que * représente la dissimilarité de la description d'objets, et que * est subsumé par tout autre valeur.

Nous avons montré comment considérer des valeurs manquantes avec les structures de patrons, et leur interprétation en terme de FCAS. Un exemple de treillis de patrons considérant les éléments additionnels * et ? est donné à la figure 2. Les algorithmes classiques de FCA adaptés pour les structures de patrons (comme dans [11]) ne nécessitent aucune modification outre la considération des éléments * et ? lors du calcul de $\sqcap$. * permet de considérer des valeurs non renseignées que l'on ne désire pas confondre avec d'autres. ? représente une valeur manquante qui peut se substituer à n'importe quelle autre. Dans les deux cas, cela offre des perspectives pour l'utilisation de treillis de concepts pour l'analyse de données complexes et lacunaires. Un exemple est donné dans ci-après.

# 6 Une application en agronomie

Cette section montre comment l'interprétation d'un treillis de concepts, construit à partir de données lacunaires de type intervalle et considérant une similarité d'objets deux à deux, aide à la décision de pratiques agricoles.

**Données et problème.** Étant donné les caractéristiques d'un pesticide et les propriétés d'un champ, les agronomes calculent des indicateurs qui reflètent l'impact du pesticide sur l'environnement. Par exemple, l'indicateur $\mathrm{I}_{gro}$ renseigne du risque qu'un produit atteigne les eaux souterraines [1]. Les propriétés d'un champ sont directement mesurées par les agronomes. Les caractéristiques d'un pesticide sont renseignées par différentes sources d'expertise comme des bases de données, des livres, et bien sûr les agronomes eux-mêmes. Elles peuvent être organisées comme dans la table 3. Ici, 12 sources donnent les valeurs de 3 caractéristiques de la *cypermethrine*. Parfois les sources fournissent la même valeur, parfois non. Par conséquent, le problème des agronomes est de choisir une "bonne valeur" pour chaque caractéristique, argument de l'indicateur $\mathrm{I}_{gro}$. Nous considérons la situation réelle qui suit. Les propriétés d'un champ sont données par les variables *position* $= 0.5$ et *risque de lessivage* $= 0.75$, toutes deux estimées par un expert dans l'intervalle $[0, 1]$. La première caractérise le type et la couverture du sol ainsi que la date d'application du pesticide, tandis que la seconde estime le pourcentage de produit qui atteint les eaux souterraines. On désire calculer l'indicateur $\mathrm{I}_{gro}$ pour savoir s'il est dangereux d'appliquer la cypermethrine sur le sol considéré : il faut choisir une valeur pour $DT50$, $koc$, et $ADI$, dans la table 3. L'explication de ces variables se trouve dans [1] et n'est pas directement utile pour la compréhension de ce qui suit. Chaque objet de la table 3 est décrit par un vecteur d'intervalles, par exemple $BUS^{\square}$ = $\langle[30, 30],[10000, 10000],[0.05, 0.05]\rangle$ ce qui en fait une structure de patrons. Les valeurs manquantes sont remplacées par * : l'expert n'a pas fourni d'information. Nous montrons comment un treillis de concepts permet de sélectionner au mieux une valeur pour chaque variable de l'indicateur $\mathrm{I}_{gro}$.

**Paramètres et méthode.** L'expert choisit un seuil $\theta$ pour chaque attribut : pour $DT50$, $\theta = 100$, pour $koc$, $\theta = 150\,000$ et pour $ADI$, $\theta = 0$. En effet, les pesticides avec une valeur $DT50$ en dessous de 100 sont considérés comme peu néfastes pour les eaux souterraines [3]. Tout intervalle dont la taille dépasse 100 est remplacé par *. La valeur 150000 est choisie comme seuil pour $koc$. Cette valeur permet de considérer toutes les valeurs entre 2000 et 160000 similaires. En fait, un produit avec une valeur de $koc$ supérieure à 500 est fortement retenu par le sol. Enfin, toutes les valeurs disponibles pour $ADI$ sont les mêmes, il n'y a pas lieu de considérer de similarité pour cet attribut et $\theta = 0$. Le treillis de concepts obtenu est donné par la figure 3 avec 11 concepts. L'extension d'un concept se lit de manière usuelle, comme en AFC classique. L'intension se lit un peu différemment : elle est composée pour chaque attribut du label "le plus bas" au dessus du concept considéré. Dans l'exemple, le concept le plus en bas à gauche est $(\{BUS\}, \{(DT50, 30), (koc, [3684, 144652])\})$, et son super concept le plus à gauche est $(\{BUS, BUK\}, \{(DT50, [5, 82]), (koc, [3684, 144652])\})$. Ici, les intensions d'un concept renseignent la valeur d'un attribut si elle est différente de *, c.-à-d. que les objets partagent cet attribut. L'interprétation est la suivante. Plus haut est un concept dans le treillis, plus grands sont les intervalles de son intension, moins nombreux sont les attributs de son intension, et plus nombreux sont les objets de son extension. Pour chaque concept, les objets sont similaires au regard des $\theta$ choisis : les concepts les plus hauts donnent un consensus idéal pour les caractéristiques décrites par l'intension.

**Premiers résultats et discussion.** Nous extrayons du treillis les valeurs partagées par le plus de sources pour le calcul d'$\mathrm{I}_{gro}$, c.-à-d. $koc = [3684, 144652]$, $ADI = [0.05, 0.05]$ et $DT50 = [5, 82]$, voir [1] pour le détail de ce calcul. L'indicateur $\mathrm{I}_{gro}$ prend ses valeurs dans l'intervalle $[0, 10]$, où une valeur inférieure à 7 signifie qu'il est dangereux d'appliquer le pesticide, et qu'il faut donc revoir la pratique agricole. Dans notre cas, les valeurs extraites du treillis permettent de calculer la valeur 10. En d'autres termes, cela signifie qu'il n'est pas dangereux d'appliquer le pesticide au champ considéré. Cela correspond parfaitement aux observations des experts : après application sur le champ, la *cypermethrine* n'a pas été trouvée dans les eaux souterraines [3]. L'intérêt du treillis de concepts est qu'il permet de regrouper des sources qui partagent certains attributs seulement et que les valeurs extraites ne sont pas forcément dans les données. Dans certains cas, l'extraction des valeurs à partir du treillis n'est pas si simple et l'expert peut ne pas être d'accord. Alors la navigation dans le treillis lui apporte un ensemble ordonné de valeurs partagées par un nombre décroissant de sources, l'aidant ainsi à la décision.

# 7 Conclusion

Cet article propose la synthèse de deux méthodes de classification de données complexes par treillis, et discute parti-

| | DT50 jours | koc L/kg | ADI mg/kg.jour |
|---|---|---|---|
| BUS | 30 | 10000 | |
| PM10 | 5 | | 0.05 |
| PM11 | 5 | | 0.05 |
| INRA | | | 0.05 |
| Dabène | [7,82] | [2000,160000] | 0.05 |
| ARSf | [7,82] | [5800,160000] | |
| ARSl | [6,60] | [5800,160000] | |
| Com96 | [7,82] | [2000,160000] | 0.05 |
| RIVM | [61,119] | 3684 | |
| BUK | [7,70] | 19433 | |
| AGXf | [14,199] | [26492,144652] | 0.05 |
| AGXl | [31,125] | [26492,144652] | 0.05 |

TABLE 3 – Caractéristiques du pesticide *cypermethrine*

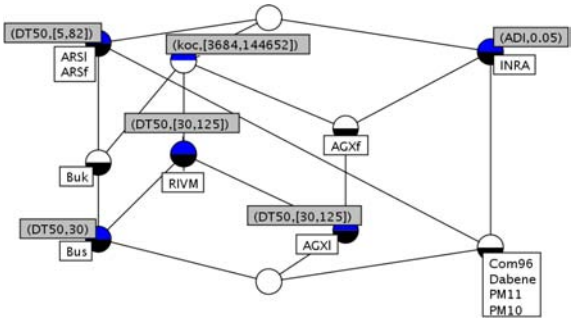


FIGURE 3 – Treillis de concepts construit à partir de la table 3

culièrement de la notion de similarité (deux à deux) et des valeurs manquantes. FCAS utilise un vocabulaire et une représentation des treillis intuitifs. IPS repose sur un formalisme mathématique existant, performant et reconnu. Les pistes de recherches sont les suivantes. Le choix des paramètres $\theta$ pose problème lorsque peu d'information sur les attributs est connue, ou que ceux-ci sont nombreux : une étude de leur variation va être menée. De nombreuses données sont bruitées : la tolérance aux fautes doit être considérée. Enfin, des liens forts s'établissent avec les méthodes de classification floues et la théorie des possibilité [5].

# Références


[1] Z. Assaghir, P. Girardin, and A. Napoli. Fuzzy logic approach to represent and propagate imprecision in agri-environmental indicator assessment. In *EUSFLAT*, 2009.

[2] M. Barbut and B. Monjardet. *Ordre et Classification, Algèbre et Combinatoire*. Hachette, Paris, 1970.

[3] C. Bockstaller, P. Girardin, and H. van der Werf. Use of agro-ecological indicators for the evaluation of farming systems. *European Journal of Agronomy*, 7(1-3) :261–270, 1997.

[4] L. Chaudron and N. Maille. Generalized formal concept analysis. In *ICCS*, volume 1867 of *LNCS*, pages 357–370. Springer, 2000.

[5] D. Dubois, F. D. de Saint-Cyr, and H. Prade. A possibility-theoretic view of formal concept analysis. *Fundamenta Informaticae*, 75(1-4) :195–213, 2007.

[6] S. Ferré and O. Ridoux. A Logical Generalization of Formal Concept Analysis. In *ICCS*, LNCS 1867, 2000.

[7] B. Ganter and S. O. Kuznetsov. Pattern Structures and Their Projections. In *ICCS*, pages 129–142, 2001.

[8] B. Ganter and R. Wille. *Formal Concept Analysis*. Springer, mathematical foundations edition, 1999.

[9] G.Polaillon. Pyramidal classification for interval data using galois lattice reduction. *Analysis of Symbolic Data ? Exploratory Methods for Extracting Statistical Information from Complex Data*, 2000.

[10] A. Haas, G. Matheron, and J. Serra. Morphologie mathématique et granulométries en place. Technical report, Annales de Mines, 1967.

[11] M. Kaytoue, S. Duplessis, S. O. Kuznetsov, and A. Napoli. Two FCA-Based Methods for Mining Gene Expression Data. In *Formal Concept Analysis*, LNCS 5548, pages 251–266, 2009.

[12] M. Kaytoue, S. Duplessis, and A. Napoli. Using formal concept analysis for the extraction of groups of co-expressed genes. In *MCO*, CCIS 14, pages 439–449, 2008.

[13] S. O. Kuznetsov. Galois connections in data analysis : Contributions from the soviet era and modern russian research. In *ICFCA*, pages 196–225, 2005.

[14] S. O. Kuznetsov and S. A. Obiedkov. Comparing Performance of Algorithms for Generating Concept Lattices. *Journal of Experimental and Theoretical Artificial Intelligence*, 14 :189–216, 2002.

[15] M. Liquiere. Some Links Between Formal Concept Analysis and Graph Mining. In *Mining Graph Data*. Wiley & Sons, 2006.

[16] N. Messai. *Analyse de concepts formels guidée par des connaissances du domaine – Application á la découverte de ressources génomiques sur le Web*. Thèse d'informatique, Université Henri Poincaré (Nancy 1), 2009.

[17] N. Messai, M. Devignes, A. Napoli, and M. Smail-Tabbone. Many-valued concept lattices for conceptual clustering and information retrieval. In *European Conference on Artificial Intelligence*, 2008.

[18] L. D. Raedt. *Logical and Relational Learning*, chapter 9 : Kernels and Distances for Strucutred Data, pages 289–324. 2008.

[19] R. Wille. Why can concept lattices support knowledge discovery in databases ? *J. Exp. Theor. Artif. Intell.*, 14(2-3) :81–92, 2002.