Learning Spaces, and How to Build them

Jean-Paul Doignon

Université Libre de Bruxelles,
Deparment of Mathematics c.p. 216,
Bd du Triomphe, 1050 Brussels, Belgium.
doignon@ulb.ac.be

Abstract. In Knowledge Space Theory (KST), a knowledge structure encodes a body of information as a domain, consisting of all the relevant pieces of information, together with the collection of all possible states of knowledge, identified with specific subsets of the domain. Knowledge spaces and learning spaces are defined through pedagogically natural requirements on the collection of all states. We explain here several ways of building in practice such structures on a given domain. In passing we point out some connections linking KST with Formal Concept Analysis (FCA).

Keywords: knowledge space, learning space, `QUERY` routine, antimatroid, convex geometry, closure space, formal concept lattice

1 Introduction

In Knowledge Space Theory (KST) a 'knowledge structure' encodes a body of information as a 'domain' together with 'states of knowledge'. The domain is the set of all the relevant, elementary pieces of information. Each knowledge state is a subset of the domain, which contains all the items mastered at some time by some (hypothetical) individual. For example, the empty set and the domain itself represent respectively a completely ignorant and an omniscient students. We assume here that in any knowledge structure, the empty set is a state[1]. In general, there will be many more knowledge states; their collection captures the overall structure of the body of information. If Q is the domain and $\mathcal{K}$ the collection of states, the knowledge structure is the pair $(Q, \mathcal{K})$. An example with domain $Q = \{a, b, c, d\}$ is displayed in Figure 1: the boxes show the nine states forming $\mathcal{K}$, while the ascending lines indicate the covering relation among states.

Without further restrictions on the collection of states, knowledge structures are too poorly organized for the development of a useful theory. Fortunately, pedagogical considerations lead in a natural way to impose restrictions on the state collection. We now explain two natural requirements by looking at the knowledge structure $(Q, \mathcal{K})$ from Figure 1. The subset $\{c, d\}$ is a knowledge

[1] In KST, it is often required that the domain be also a state; we leave out this assumption here in order to ease in Section 3 the comparison with the closure spaces of FCA.

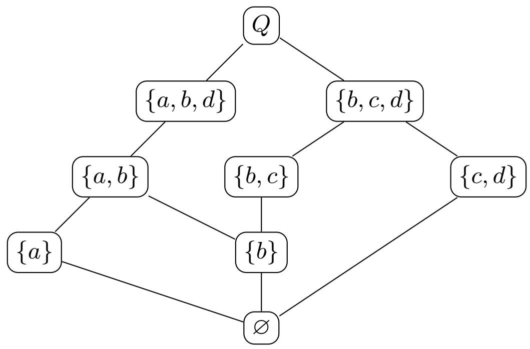

Fig. 1. An example of knowledge structure with domain $Q = \{a, b, c, d\}$ and nine knowledge states as shown.

state in $\mathcal{K}$, but there is no way for a student to acquire mastery of items c and d one after the other in any order (neither subset $\{c\}$ nor $\{d\}$ is a state in $\mathcal{K}$). This contradicts the (common) view that learning occurs progressively, that is one item at a time. For another singularity in the same knowledge structure $(Q, \mathcal{K})$, consider a student in state $\{b\}$. She may learn item a to reach state $\{a, b\}$. On the other hand, while in state $\{b\}$ she may rather learn item c first and reach state $\{b, c\}$; then, strangely enough, item a is not learnable anymore to her (because the subset $\{a, b, c\}$ is not a state in $\mathcal{K}$). The definition of a 'learning space' as a particular type of knowledge structure rules out the two strange situations that we just illustrated on Figure 1. It imposes the following two conditions on the states of a knowledge structure $(Q, \mathcal{K})$.

[A] ACCESSIBILITY. Any state K contains an item q such that $K \setminus \{q\}$ is again a state.

[LC] LEARNING CONSISTENCY. For any state K and items q, r, if $K \cup \{q\}$ and $K \cup \{r\}$ are states, then $K \cup \{q, r\}$ is also a state.

As we will explain in the next section, "learning space" happens to be just another name for "$\cup$-stable antimatroids".

Knowledge Space Theory (KST) is at the basis of the computer-assisted teaching system `ALEKS`. Developed since around 1995 in a start-up company[2] with the same name in Irvine (California), `ALEKS` is now quite successful with $1,300,000$ single users in 2013. A special feature of `ALEKS` is its assessment module, whose foundation relies on the concept of a learning space. We will not expose the assessment principles, but rather consider the question of how to build a learning space for a specific body of information—this enterprise is a preliminary for the implementation of an efficient assessment module.

To give an example, suppose we have at hand 200 items which represent the topic of arithmetic at the ages 12–14. How can we build an adequate collection of knowledge states on these 200 items? The idea is to rely on the advice of experts

[2] The company was recently acquired by McGraw-Hill Education.

in the teaching of arithmetic, or as it is done today, on the huge database of past assessments of student mastery (in kind of a bootstrapping method, see details below in Section 4). Section 4 explains (at least the basic principles of) a general routine called `QUERY`. The `QUERY` routine emerges from work by Koppen and Doignon (1990) and Koppen (1993). Eppstein, Falmagne and Uzun (2009) were the first to apply it to build learning spaces. Then Falmagne and Doignon (2011) introduces another way of using the routine for the same goal. We sketch here a third way, maybe a more insightful one, of taking advantage of the `QUERY` routine.

In Section 3 we point out some links between KST and Formal Concept Analysis (FCA), thus complementing the works of Rusch and Wille (1996) and Spoto, Stefanutti and Vidotto (2010).

2 Learning spaces and Knowledge Spaces

We first provide the formal definitions of concepts met in the Introduction.

Definition 1. A *knowledge structure* $(Q, \mathcal{K})$ consists of a finite, nonempty set Q together with a collection $\mathcal{K}$ of subsets of Q. In the present text we make the only requirement $\varnothing \in \mathcal{K}$. The elements of the *domain* Q are *items*, those of $\mathcal{K}$ *(knowledge) states*.

The restriction to finite domains Q is made here because of our main goal—namely, the explanation of the `QUERY` routine.

Definition 2. A *learning space* $(Q, \mathcal{K})$ is a knowledge structure $(Q, \mathcal{K})$ in which the collection $\mathcal{K}$ of states satisfies[3] two conditions (as in the Introduction):

[A] Accessibility. Any state K contains an item q such that $K \setminus \{q\}$ is again a state:

$$\forall K \in \mathcal{K}, \exists q \in K : \; K \setminus \{q\} \in \mathcal{K}; \tag{1}$$

[LC] Learning consistency. For any state K and items q, r, if $K \cup \{q\}$ and $K \cup \{r\}$ are states, then $K \cup \{q, r\}$ is also a state:

$$\forall K \in \mathcal{K}, \forall q, r \in Q : \; (K \cup \{q\}, K \cup \{r\} \in \mathcal{K}) \implies K \cup \{q, r\} \in \mathcal{K}. \tag{2}$$

A large collection of learning spaces derives from ordered sets. Let $(Q, \preceq)$ be a *partially ordered set* (in other words, $\preceq$ is a reflexive, transitive and antisymmetric relation, or a *partial order*, on Q). Define a *state of* $\preceq$ to be any subset K of Q such that

$$\forall q, r \in Q : \; (q \preceq r \text{ and } r \in K) \implies q \in K.$$

As it is easily checked, the collection $\mathcal{L}$ of states of $\preceq$ contains $\varnothing$ and Q, and it satisfies [A] and [LC] in Definition 2. So $(Q, \mathcal{L})$ is a learning space, that we call the *ordinal space (derived from* $\preceq$*)*. Notice that the collection of states of an

[3] In an unusual way, we do not require $Q \in \mathcal{K}$—see Footnote 1.

ordinal space is stable under both union and intersection, in the sense that any union and intersection of states are again states.

There are many other characterizations of learning spaces. To state one, we recourse to the notion of 'wellgradedness'. In rough terms, a collection of subsets of a finite domain Q is well-graded when it is possible to move from any of its members to any other one by 'elementary' steps which, moreover, are in number equal to the 'distance' between the two members. Here, the distance means the 'symmetric-difference distance', and a step is elementary if it consists in either adding or deleting a single element.

Definition 3. The *(symmetric-difference) distance* between two subsets K and L of a finite domain Q is equal to $d(K, L) = |K \Delta L|$ (this indeed defines a distance d on the collection of subsets of Q). A collection $\mathcal{K}$ of subsets of a finite domain Q is *well-graded* when for any two members K, L of $\mathcal{K}$ with $d(K, L) = m$, there exist states K_1, K_2, ..., K_{m-1} in $\mathcal{K}$ such that, with $K_0 = K$ and $K_m = L$, there holds $d(K_{i-1}, K_i) = 1$ for $i = 1, 2, \ldots, m$.

The notion of wellgradedness plays a role also outside KST. For instance, Doignon and Falmagne (1997) show that the collection of all partial orders (resp. "interval orders", "semiorders") on a finite domain is well-graded. Returning to our present topic, we notice that the states of a learning space form a well-graded collection, and even more:

Proposition 1. *Let* $(Q, \mathcal{K})$ *be a knowledge structure. Then the two following conditions are equivalent:*

(i) $(Q, \mathcal{K})$ *is a learning space;*
(ii) the collection $\mathcal{K}$ *is well-graded and stable under union.*

Stability under union is an important property in KST. For instance, knowledge structures whose collection of states is stable under union are closely related to "AND/OR graphs"[4]. They will be central in Section 4.

Definition 4. A *knowledge space* $(Q, \mathcal{K})$ is a knowledge structure whose collection $\mathcal{K}$ of states is stable under union.

In any knowledge space $(Q, \mathcal{K})$, some states can be written as unions of other ones, while some states cannot. We now characterize the latter.

Definition 5. In a knowledge structure $(Q, \mathcal{K})$, a *clause for an item* q is any state which contains q and is minimal for the latter property. A *clause* is a state which is a clause for some item. The set of all clauses is denoted as $\mathcal{B}$.

Proposition 2. *In a knowledge space* $(Q, \mathcal{K})$*, any state is a union of clauses, but no clause can be written as a union of other states. Moreover, any collection* $\mathcal{A}$ *of states having the property that any state in* $\mathcal{K}$ *is a union of members of* $\mathcal{A}$ *must contain all the clauses, that is* $\mathcal{B} \subseteq \mathcal{A}$*.*

[4] AND/OR graphs generalize partially ordered sets in that each item may have several set of predecessors; for details, see Doignon and Falmagne (1999), Chapter 3.

Definition 6. In a knowledge space $(Q, \mathcal{K})$, the *base* is the collection $\mathcal{B}$ of all clauses.

The following two other characterizations of learning spaces are due to Koppen (1998).

Proposition 3. *For a knowledge space* $(Q, \mathcal{K})$*, the three following assertions are equivalent:*

(i) $(Q, \mathcal{K})$ *is a learning space;*
(ii) any clause is a clause for only one item;
(iii) for any two distinct items q*,* r*, the set of clauses for* q *differ from the set of clauses for* r*.*

The names we introduced in Definitions 1, 2 and 4 reflect the motivation of KST. We now point out the links with more classical, mathematical structures. To do so, we associate to any knowledge structure $(Q, \mathcal{K})$ its *dual* structure $(Q, \overline{\mathcal{K}})$, where

$$\overline{\mathcal{K}} \;=\; \{\, Q \setminus K \bigm| K \in \mathcal{K} \,\}.$$

Knowledge structures $(Q, \mathcal{K})$, if we remove the innocuous requirement $\varnothing \in \mathcal{K}$, are just "hypergraphs" (Berge, 1989). Knowledge spaces (with the requirement of stability under union) are exactly the duals of 'closure spaces' (see for instance Birkhoff, 1967; Buekenhout, 1967; van de Vel, 1993). Closure spaces play an important role in FCA as we will recall in Section 3 (Ganter and Wille, 1996).

Definition 7. A *closure space* $(Q, \mathcal{C})$ is a finite set Q with a collection $\mathcal{C}$ of subsets of Q which is closed under intersection and contains Q.

A closure space $(Q, \mathcal{C})$ corresponds to exactly one *closure operator on* Q, that is a map $2^Q \to 2^Q$ which is expansive, monotone and idempotent. To be precise, the closure operator of $(Q, \mathcal{C})$ is $2^Q \to 2^Q : A \to \overline{A} = \cap\{C \in \mathcal{C} \bigm| A \subseteq C\}$.

Learning spaces (characterized through well-gradedness and stability under union as in Proposition 1) are the $\cup$-stable antimatroids of Korte, Lovász and Schrader (1991) (except for the missing requirement $Q \in \mathcal{K}$—notice that by Proposition 1 any of our learning space has a maximum state containing all the other states, but this state may differ from Q). As a matter of fact, Proposition 1 can be found in Chapter III of the latter reference (see also Cosyn and Uzun, 2009). Notice that the duals of learning spaces are '$\cap$-stable antimatroids' in the sense of Edelman and Jamison (1985) (we give a definition which is different from, but equivalent to, the original one except that we admit the omission of $\varnothing$ from $\mathcal{C}$).

Definition 8. An *$\cap$-stable antimatroid* is a closure space $(Q, \mathcal{C})$ in which the collection $\mathcal{C}$ of closed sets satisfies

[E] Extendability. For any closed set C in $\mathcal{C}$, there exists an element p of $Q \setminus C$ such that $C \cup \{q\}$ is again a closed set:

$$\forall C \in \mathcal{C}, \exists p \in Q \setminus C : \; C \cup \{q\} \in \mathcal{C}.$$

Note that in any $\cap$-stable antimatroid $(Q, \mathcal{C})$, there is a minimum closed set contained in all closed sets (which may be empty or not). Another characterization of $\cap$-stable antimatroids is as follows, in terms of the closure $A \to \overline{A}$ (this is the original definition in Jamison, 1980, 1982, however modified here to allow for the possible omission of $\varnothing$ from $\mathcal{K}$).

Proposition 4. *A closure space $(Q, \mathcal{C})$ is an $\cap$-stable antimatroid if and only if, for any closed set C in $\mathcal{C}$ and distinct elements p, q in Q:*

$$\Big(p \in \overline{C \cup \{q\}} \text{ and } q \notin C\Big) \quad \Longrightarrow \quad q \notin \overline{C \cup \{p\}}.$$

3 Knowledge Space Theory and Formal Concept Analysis

As is well known, closure spaces are useful in FCA. Given a context (G, M, I) (thus I is a relation from G to M), define the two mappings

$$2^G \to 2^M : \; A \to A' = \{m \in M \,|\, \forall a \in A : a \; I \; m\}, \tag{3}$$

$$2^M \to 2^G : \; B \to B' = \{g \in G \,|\, \forall b \in B : g \; I \; b\}. \tag{4}$$

Then

$$2^G \to 2^G : \; A \to A'', \tag{5}$$

$$2^M \to 2^M : \; B \to B'' \tag{6}$$

are both closure operators. Their collections of closed sets,

$$\mathcal{C} \; = \; \{A \in 2^G \,|\, A = A''\}, \tag{7}$$

$$\mathcal{D} \; = \; \{B \in 2^G \,|\, B = B''\} \tag{8}$$

produce closure spaces, respectively $(G, \mathcal{C})$ and $(M, \mathcal{D})$. The two collections are put in one-to-one correspondence by the two mutually reciprocal, bijective mappings

$$\mathcal{C} \to \mathcal{D} : \; A \to A', \tag{9}$$

$$\mathcal{D} \to \mathcal{C} : \; B \to B'. \tag{10}$$

Moreover, these mappings are inclusion-reversing, so that the two partially ordered sets $(\mathcal{C}, \subseteq)$ and $(\mathcal{D}, \subseteq)$ are anti-isomorphic. In FCA, a pair (A, B) with $A \in 2^G$, $B \in 2^M$, $A' = B$ and $B' = A$ is a *concept* with *extent* A and *intent* B.

In line with Section 2, we may ask under which condition on the context (G, M, I) the closure spaces $(G, \mathcal{C})$ and/or $(M, \mathcal{D})$ are $\cap$-stable antimatroids. Although Ganter and Wille (1996) do not mention the word "antimatroid", precise answers in lattice-theoretic terms appear in their Theorem 44. Here we directly derive from Proposition 3 similar answers in set-theoretic terms. Let us first look at an example.

Example 1. Let $G = \{a, b\}$, $M = \{m, n, p\}$, and I be the relation described by the following boolean table:

	m	n	p
a	1	1	0
b	0	0	1

Figure 2 shows the resulting two closure spaces $(G, \mathcal{C})$ and $(M, \mathcal{D})$. Notice that

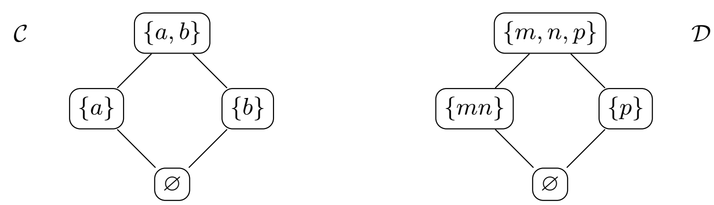

Fig. 2. The two closure spaces for the context in Example 1.

$(G, \mathcal{C})$ is an $\cap$-stable antimatroid, while $(M, \mathcal{D})$ is not because $\mathcal{D}$ is not extendible in view of $\{p\}$.

Definition 9. In a context (G, M, I), an attribute m in M is a *demarcator at an object* g when m' does not contain g and is maximal among all the b', for b in M, which share this property. A *demarcator* is a demarcator at some object.

In the notation of Ganter and Wille (1996), the attribute m is a demarcator at the object g exactly when $g \nearrow m$; also, a demarcator is an attribute m such that the concept (m', m'') is $\wedge$-irreducible. Demarcators are dual to clauses (Definition 5). We now rephrase Proposition 2 and 3 in their dual versions.

Proposition 5. *For a context (G, M, I), any closed set in G is an intersection of demarcators. Moreover, the collection of demarcators is contained in any subset N of M having the property that any closed set C in G is the intersection of a subcollection of $\{n' \mid n \in N\}$.*

Proposition 6. *Given a context (G, M, I), the following three assertions are equivalent:*

(i) *the closure space $(G, \mathcal{C})$ is an $\cap$-stable antimatroid;*
(ii) *any demarcator is a demarcator at only one object;*
(iii) *for any two distinct objects g, h, the set of demarcators at g differ from the set of demarcators at h.*

There is of course a similar criterion for $(M, \mathcal{D})$ to be an $\cap$-stable antimatroid. The demarcators m', for $m \in M$, are crucial here because any closed set in G is an intersection of such subsets (Proposition 5). They are also useful in relation with a quasi order that we now define on the collection of all relations from the finite set G to the finite set M.

Definition 10. Let $\mathcal{I}$ be the collection of all relations from the finite set G to the finite set M. Define a relation $\preccurlyeq$ on $\mathcal{I}$ by letting, for I, $J \in \mathcal{I}$:

$$I \preccurlyeq J \quad \text{when} \quad \forall m \in M, \exists m_1, m_2, \ldots, m_k \in M, \forall g \in G : \tag{11}$$

$$g\ I\ m \iff g\ J\ m_1, g\ J\ m_2, \ldots, g\ J\ m_k. \tag{12}$$

In other words, $I \preccurlyeq J$ exactly when each attribute extent m' w.r.t I (for each m in M) is an intersection of some attribute extents n' w.r.t. J (where the n's are in M); notice that an equivalent definition of $\preccurlyeq$ results when "$\forall m \in M$" is replaced with "for any demarcator m w.r.t. I" and "$\exists m_1, m_2, \ldots, m_k \in M$" with "there exist demarcators $m_1, m_2, \ldots, m_k$ w.r.t. J". It is easily checked that $\preccurlyeq$ is a quasi order on $\mathcal{I}$ (that is, $\preccurlyeq$ is a reflexive and transitive relation on $\mathcal{I}$). Moreover, two relations I and J in $\mathcal{I}$ are equivalent ($I \preccurlyeq J$ and $J \preccurlyeq I$) exactly if they have the same collection of demarcator extents. The definition of $\preccurlyeq$ is tailored for delivering the following straightforward result.

Proposition 7. *For I a relation from G to M (that is, $I \in \mathcal{I}$), denote by $\mathcal{C}_I$ the closed sets in G of the context (G, M, I). Then for I, J in $\mathcal{I}$*

$$I \preccurlyeq J \qquad \iff \qquad \mathcal{C}_I \subseteq \mathcal{C}_J.$$

Thus I and J are equivalent in $\preccurlyeq$ exactly if they produce the same concept lattice.

4 The `QUERY` routine to build a knowledge space

Suppose we have all the items in an area of knowledge. How can we then build an adequate collection of (potential) knowledge states? In the first steps of the application of KST, relevant information came from experts in the area. A computer routine displays 'queries' on the screen, and collects experts' answers on the keyboard. A crucial feature of the `QUERY` routine is its ability to infer additional information from previous answers (it thus avoids setting forth too many queries). It is useful as well in current use of KST, where queries are addressed to a database of past assessment sessions rather than to human experts. There is a bootstrapping method at work here. The very first stage relies on a very crude collection $\mathcal{K}$ of potential knowledge states (for instance, if the domain Q is not too large, all of its subsets are in the initial collection $\mathcal{K}$ of states). The database records the assessments based on $\mathcal{K}$. Then a call of the `QUERY` routine results (as we explain below) in the deletion of subsets from the collection $\mathcal{K}$. Next, the database records further assessments based on the new collection of states. The `QUERY` routine can then take advantage of the new assessment history, and again reduce the collection of states. There can be many repetitions of the assessment/`QUERY` sequence.

We focus here the exposition on the `QUERY` routine itself, however we leave many details aside. For instance, the two-stage process usually first works only with small subsets of the domain (say, with only 6 items). The information found about the states within the parallel subdomains delivers an initial list of (not too many) potential states on the whole domain. Then the two-stage process works on the full domain (for more about this, see Subsection 0.10.2 of Doignon and Falmagne, 2015).

In the present section, we explain how the original `QUERY` routine produces a knowledge space, that is a collection of states closed under union. In the next section, we explain how to adapt the routine in order that it produce a learning space (that is, a $\cup$-stable antimatroid): there, we want the collection of states to be not only $\cup$-closed, but also accessible (Definition 2).

A typical query to an expert or the database takes the following form, for some subset A of Q and some item q in Q:

> *Suppose that a student under examination has just provided wrong answers to all the items in A.*
> *Is it practically certain that this student will also fail item q?*
>
> *(Assume that the conditions are ideal in the sense that in the formulation of student answers, careless errors and lucky guesses are excluded.)*

We denote the above query by (A, q). A positive answer to query (A, q) rules out subsets from being potential knowledge states. Indeed, if there were a state L with $A \cap L = \varnothing$ and $q \in L$, then the expert answer could not be positive. Thus, assuming that before query (A, q) the available collection of states is $\mathcal{F}$, upon a positive answer to query (A, q) we may delete from $\mathcal{F}$ all the elements of the collection

$$\mathcal{D}_{\mathcal{F}}(A, q) \quad = \quad \{L \in \mathcal{F} \,\big|\, A \cap L = \varnothing \text{ and } q \in L\}.$$

If the goal is to build a knowledge space, the next proposition shows that we may safely perform the deletion for as many queries as we want—if the initial collection is itself a knowledge space (it could be for instance $(Q, 2^Q)$).

Proposition 8. *For any knowledge space $(Q, \mathcal{K})$ and any query (A, q), the collection $\mathcal{K} \setminus \mathcal{D}_{\mathcal{K}}(A, q)$ is stable under union, so that $(Q, \mathcal{K} \setminus \mathcal{D}_{\mathcal{K}}(A, q))$ is again a knowledge space.*

Now we briefly explain how inferences can be made from collected answers to past queries. Suppose first that we limit ourselves to queries (A, q) having $|A| = 1$. In other words, we try to uncover a (binary) relation R on Q: a positive answer to query $(\{q\}, r)$ acknowledges $q\ R\ r$, a negative answer entails (not $q\ R\ r$). The latent relation R, to be uncovered, captures prerequisites among the items. To be precise, for items q and r, we have $q\ R\ r$ when q is a prerequisite for r (in the sense that any knowledge state containing r also contains q). This is reminiscent of the definition of an ordinal space (see after Definition 2): because it is natural to assume that R is a partial order on Q, all the states compatible with the answers to queries form the ordinal space derived from R. The transitivity of the relation R makes it possible to infer additional information from answers to past queries. Indeed, if queries $(\{q\}, r)$ and $(\{r\}, s)$ were positively answered, then we may infer that query $(\{q\}, s$ will also be—and we may refrain from asking the latter query. Another case: if query $(\{q\}, s)$ receives a negative answer and query $(\{q\}, r)$ a positive one, then we may infer that query $(\{r\}, s)$ will receive a negative answer—and again we do not need asking the latter query. Similarly, query $(\{q\}, s)$ negatively answered, query $(\{r\}, s)$ positively answered entail that query $(\{q\}, r)$ will be negatively answered. There are similar inferences that can be made when no restriction is made on the size of A in queries (A, q); we refer the reader to Koppen (1998) for a detailed exposition.

In view of Proposition 8, the application of the `QUERY` routine, starting from any knowledge space, results after each query in the production of a knowledge space. But are we sure that the routine will uncover a latent knowledge space, if the expert answers are coherent with such a space? The answer is in the affirmative.

Proposition 9. *Suppose the* `QUERY` *routine starts with the initial knowledge space* $(Q, 2^Q)$*, and that the answers of the expert to queries are always compatible with a latent knowledge space* $(Q, \mathcal{L})$*. Then at any step, the knowledge space* $(Q, \mathcal{K})$ *built by the routine satisfies* $\mathcal{K} \subseteq \mathcal{L}$*. Moreover, if* $\mathcal{K} \subset \mathcal{L}$*, there is some query* (A, q) *such that* $\mathcal{D}_{\mathcal{K}}(A, q)$ *is nonempty. Hence, after having collected or inferred the answers to all possible queries, the routine produces the latent knowledge space* $(Q, \mathcal{L})$*.*

5 Adapting the `QUERY` routine to build a learning space

In this section, we assume again that all the items forming the domain Q are available. Our goal this time is to build a learning space, not just a knowledge space as in the previous section. A solution proposed by Eppstein, Falmagne and Uzun (2009) works in two steps: first use the `QUERY` routine to build a knowledge space $(Q, \mathcal{K})$, then add states until $\mathcal{K}$ becomes a learning space. Note that the second step involves arbitrariness in the choice of the additional states. One reason lies in the following observation: given a knowledge space $(Q, \mathcal{K})$, the collection of all learning spaces $(Q, \mathcal{L})$ such that $\mathcal{K} \subseteq \mathcal{L}$ contains in general several minimal elements (w.r.t. inclusion).

Example 2. For $Q = \{a, b\}$ and $\mathcal{K} = \{\varnothing\}$ (the smallest possible knowledge space on $\{a, b\}$), there are three learning spaces $(Q, \mathcal{L})$ such that $\mathcal{K} \subseteq \mathcal{L}$. Their collections $\mathcal{L}$ of states are $\{\varnothing, \{a\}, Q\}$, $\{\varnothing, \{b\}, Q\}$ and $\{\varnothing, \{a\}, \{b\}, Q\}$. Two of them are minimal.

Here is a fundamental difference between knowledge spaces and learning spaces: if $(Q, \mathcal{K}_1)$ and $(Q, \mathcal{K}_2)$ are knowledge spaces on the same domain, their intersection $(Q, \mathcal{K}_1 \cap \mathcal{K}_2)$ is again a knowledge space, while the similar assertion for learning spaces does not hold.

Falmagne and Doignon (2011) describes another solution to the problem of building a learning space, which adapts the `QUERY` routine according to the following general principle. Start from an initial learning space (for instance $(Q, 2^Q)$) and successively collect responses to queries; when a query receives a positive answer, delete subsets which cannot be states only if the resulting space is again a learning space—if it is not, keep the information provided by the positive answer for possible, later use. The resulting 'adapted `QUERY`' routine performs well, in particular it uncovers the latent learning space governing the expert answers—if there is any such latent space (see Proposition 12 below). To provide more details on the adapted `QUERY` routine, we need two more notions from KST.

Definition 11. Let $(Q, \mathcal{K})$ be a knowledge structure, and K be a state in $\mathcal{K}$. The *inner fringe* of K is

$$K^{\mathcal{I}} \;=\; \{\, q \in K \,|\, K \setminus \{q\} \in \mathcal{K} \,\}.$$

The *outer fringe* of K is

$$K^{\mathcal{O}} \;=\; \{\, q \in Q \setminus K \,|\, K \cup \{q\} \in \mathcal{K} \,\}.$$

The outer fringe of the state K contains the items which a student in state K is ready to learn. In a general knowledge structure, both the inner and outer fringes can be empty. Accessibility (Definition 2) entails that the inner fringe of any nonempty state is nonempty, while extendability (Definition 8) entails that the outer fringe of any state different from the domain is nonempty.

Proposition 10. *In a learning space $(Q, \mathcal{L})$, no two states in $\mathcal{L}$ have the same pair of inner and outer fringes.*

In other words, a state in a learning space $(Q, \mathcal{L})$ is determined by its two fringes (and the availability of $\mathcal{L}$).

The design of the *adapted* `QUERY` routine relies on the following property. Remember that $\mathcal{D}_{\mathcal{K}}(A, q)$ is the collection of subsets ruled out by a positively answered query (A, q):

$$\mathcal{D}_{\mathcal{F}}(A, q) \quad = \quad \{L \in \mathcal{F} \,|\, A \cap L = \varnothing \text{ and } q \in L\}.$$

Proposition 11. *For any learning space $(Q, \mathcal{L})$ and any query (A, q), the collection $\mathcal{L} \setminus \mathcal{D}_{\mathcal{L}}(A, q)$ gives a learning space $\cancel{(Q, \mathcal{L})}$ if and only if there is no state K in $\mathcal{L}$ such that $|K^{\mathcal{I}}| = 1$, $A \cap K = K^{\mathcal{I}}$ and $q \in K$.*

The adapted `QUERY` routine relies on Proposition 11 to test whether a positive answer to the query (A, q) may be safely applied—that is, to test whether the knowledge space resulting from the deletion of the sets forming $\mathcal{D}_{\mathcal{F}}(A, q)$ is again a learning space. We refer the reader to Falmagne and Doignon (2011) for a full description of the adapted `QUERY` routine, but state here one of its fundamental properties.

Proposition 12. *If $\mathcal{L}$ is a latent learning space and the query answers are truthful with respect to $\mathcal{L}$, then the adapted* `QUERY` *routine will ultimately uncover $\mathcal{L}$.*

The proof of Proposition 12 relies on results of Edelman and Jamison (1985) and Caspard and Monjardet (2004) about the collection of all antimatroids on a given set.

We now briefly sketch a third way of using the `QUERY` routine for building a learning space. A fundamental property of the collection of all learning spaces on a domain Q is that it forms a $\vee$-semilattice w.r.t. inclusion (Caspard and Monjardet, 2004). Let us denote by $\mathbf{L}$ the family of all collections $\mathcal{L}$ of subsets of Q such that $(Q, \mathcal{L})$ is a learning space. If $\mathcal{L}_1$ and $\mathcal{L}_2$ are in $\mathbf{L}$, then among all the collections $\mathcal{L}$ in $\mathbf{L}$ such that $\mathcal{L}_1 \subseteq \mathcal{L}$ and $\mathcal{L}_2 \subseteq \mathcal{L}$, there is a smallest one for inclusion, namely their *least upper bound*

$$\mathcal{L}_1 \vee \mathcal{L}_2 \;=\; \{L_1 \cup L_2 \,\big|\, L_1 \in \mathcal{L}_1,\, L_2 \in \mathcal{L}_2\}. \tag{13}$$

Notice that the partially ordered set $(\mathbf{L}, \subseteq)$ has no minimum element; its minimal elements are all the *full chains*, that is the collections $\{L_0, L_1, \dots, L_m\}$ of subsets of Q such that $L_0 \subset L_1 \subset \cdots \subset L_m$ and $m = |Q|$. We may turn $(\mathbf{L}, \subseteq)$ into a lattice[5] by adding to $\mathbf{L}$ a new element $\perp$ which becomes the minimum (we assume $\perp \subseteq \mathcal{L}$ for any $\mathcal{L}$ in $\mathbf{L} \cup \{\perp\}$). Then any two elements $\mathcal{L}_1$ and $\mathcal{L}_2$ in $\mathbf{L} \cup \{\perp\}$ have a *greatest lower bound*

$$\mathcal{L}_1 \wedge \mathcal{L}_2 \;=\; \bigvee \left\{\mathcal{L} \in \mathbf{L} \cup \{\perp\} \,\big|\, \mathcal{L} \subseteq \mathcal{L}_1,\, \mathcal{L} \subseteq \mathcal{L}_2 \right\}. \tag{14}$$

Now when the `QUERY` routine has $\mathcal{L}$ as its actual collection of states, with $\mathcal{L} \in \mathbf{L}$, and it receives a positive answer to the query (A, q), we know that only the subsets in $\mathcal{L}^* = \mathcal{L} \setminus \mathcal{D}_{\mathcal{F}}(A, q)$ should remain in the new collection of possible states. The latter collection $\mathcal{L}^*$ always provides a knowledge space on Q (Proposition 8), but in general not a learning space (Proposition 11). There are two cases which our *adjusted* `QUERY` *routine* must handle:

(i) $\mathcal{L}^*$ contains some element $\mathcal{F}$ of $\mathbf{L}$; then, in view of the definition of $\vee$ above, it contains for sure the learning space

$$\bigvee \left\{\mathcal{F} \in \mathbf{L} \,\big|\, \mathcal{F} \subseteq \mathcal{L}^* \right\}.$$

We then instruct the adjusted `QUERY` routine to replace $\mathcal{L}$ with the latter learning space.

(ii) $\mathcal{L}^*$ does not contain any element from $\mathbf{L}$. We then keep $\mathcal{L}$ as the actual collection of states.

Although the best way to implement Step (i) in the adjusted `QUERY` routine is still under investigation, it is clear that the outcome of the adjusted `QUERY` routine is always a learning space. We also have a result similar to Proposition 12 (which was about the adapted `QUERY` routine).

[5] This is the usual way of turning a $\vee$-semilattice into a lattice.

Proposition 13. *If $\mathcal{L}$ is a latent learning space and the query answers are truthful with respect to $\mathcal{L}$, then the adjusted* `QUERY` *routine will ultimately uncover $\mathcal{L}$.*

Notice that in the setting of Proposition 13, Case (ii) never occurs in the execution of the routine. We leave for further work a comparison of the performances of the three ways of using the `QUERY` routine to uncover a latent learning space.

Bibliography

Berge, Cl. 1989. *Hypergraphs. Combinatorics of finite sets.* Transl. from the French. Amsterdam: North-Holland.

Birkhoff, G. 1967. *Lattice Theory.* Providence, R.I.: American Mathematical Society.

Buekenhout, F. 1967. Espaces à fermeture. *Bull. Soc. Math. Belg.*, **19**, 147–178.

Caspard, N., and Monjardet, B. 2004. Some lattices of closure systems on a finite set. *Discrete Math. Theor. Comput. Sci.*, **6**(2), 163–190 (electronic).

Cosyn, E., and Uzun, H.B. 2009. Note on two necessary and sufficient axioms for a well-graded knowledge space. *J. Math. Psych.*, **53**, 40–42.

Doignon, J.-P., and Falmagne, J.-Cl. 1997. Well-graded families of relations. *Discrete Math.*, **173**, 35–44.

Doignon, J.-P., and Falmagne, J.-Cl. 1999. *Knowledge Spaces.* Berlin: Springer-Verlag.

Doignon, J.-P., and Falmagne, J.-Cl. In press. Knowledge Spaces and Learning Spaces. To appear in: W. H. Batchelder, H. Colonius, E. N. Dzhafarov, J. Myung (Eds.), *New Handbook of Mathematical Psychology.*

Edelman, P.H., and Jamison, R.E. 1985. The theory of convex geometries. *Geom. Dedicata*, **19**, 247–270.

Eppstein, D., Falmagne, J.-Cl., and Uzun, H.B. 2009. On verifying and engineering the wellgradedness of a union-closed family. *J. Math. Psych.*, **53**, 34–39.

Falmagne, J.-C., and Doignon, J.-P. 2011. *Learning Spaces.* Berlin: Springer-Verlag.

Ganter, B., and Wille, R. 1996. *Formale Begriffsanalyse: Mathematische Grundlagen.* Berlin-Heidelber: Springer-Verlag. English translation by C. Franske: *Formal Concept Analysis: Mathematical Foundations*, Springer-Verlag, 1998.

Jamison, R.E. 1980. Copoints in antimatroids. In *Proceedings of the Eleventh Southeastern Conference on Combinatorics, Graph Theory and Computing (Florida Atlantic Univ., Boca Raton, Fla., 1980)*, Vol. II. *Congr. Numer.*, **29**, 535–544.

Jamison, R.E. 1982. A perspective on abstract convexity: classifying alignments by varieties. Pages 113–150 of: *Convexity and Related Combinatorial Geometry (Norman, Okla., 1980).* Lecture Notes in Pure and Appl. Math., vol. 76. New York: Dekker.

Koppen, M. 1993. Extracting human expertise for constructing knowledge spaces: An algorithm. *J. Math. Psych.*, **37**, 1–20.

Koppen, M. 1998. On alternative representations for knowledge spaces. *Math. Social Sci.*, **36**, 127–143.

Koppen, M., and Doignon, J.-P. 1990. How to build a knowledge space by querying an expert. *J. Math. Psych.*, **34**, 311–331.

Korte, B., Lovász, L., and Schrader, R. 1991. *Greedoids.* Algorithms and Combinatorics, vol. 4. Berlin: Springer-Verlag.

Rusch, A. and Wille, R. 1996. Knowledge spaces and formal concept analysis. In H.-H. Bock and W. Polasek (Eds), *Data Analysis and Information Systems.* Studies in Classification, Data Analysis, and Knowledge Organization, Berlin Heidelberg: Springer-Verlag.

Spoto, A., Stefanutti, L. and Vidotto, G. 2010. Knowledge space theory, formal concept analysis, and computerized psychological assessment. *Behavior Res. Meth.,* **42**, *342–350.*

van de Vel, M.L.J. 1993. *Theory of convex structures.* Amsterdam: North-Holland.