Revisiting Pattern Structure Projections

Aleksey Buzmakov, Sergei O. Kuznetsov, Amedeo Napoli

HAL Id: hal-01186719
https://hal.archives-ouvertes.fr/hal-01186719

Submitted on 25 Aug 2015

Revisiting Pattern Structure Projections*

Aleksey Buzmakov[1,2] Sergei O. Kuznetsov[2] Amedeo Napoli[1]

[1]LORIA (CNRS – Inria NGE – U. de Lorraine),
Vandœuvre-lès-Nancy, France
[2]National Research University Higher School of Economics,
Moscow, Russia
aleksey.buzmakov@inria.fr, skuznetsov@hse.ru,
amedeo.napoli@loria.fr

Abstract

Formal concept analysis (FCA) is a well-founded method for data analysis and has many applications in data mining. Pattern structures is an extension of FCA for dealing with complex data such as sequences or graphs. However the computational complexity of computing with pattern structures is high and projections of pattern structures were introduced for simplifying computation. In this paper we introduce o-projections of pattern structures, a generalization of projections which defines a wider class of projections preserving the properties of the original approach. Moreover, we show that o-projections form a semilattice and we discuss the correspondence between o-projections and the representation contexts of o-projected pattern structures.

KEYWORDS: formal concept analysis, pattern structures, representation contexts, projections

1 Introduction

A significant part of recorded data represents phenomena in a structured way, e.g., a molecule is better represented as a labeled graph than as a set of attributes. Pattern structures are an extension of FCA for dealing with such kind of data [4, 3, 2]. Such a pattern structure is defined by a set of objects, a set of descriptions associated with the set of objects, and a similarity operation on descriptions, matching a pair of descriptions to their common part. For instance, the set of objects can contain molecule names, the set of descriptions contains fragments of molecules, and the similarity operation taking two sets of graphs to a set of maximal common subgraphs. The similarity operation is a semilattice operation on the set of descriptions. It allows one to deal with data (objects

---

*The final publication is available at link.springer.com

1

and their descriptions) in a similar way as one deals with objects and their intents in standard FCA. Such kind of formalization allows one to describe many types of data, however processing can be computationally very demanding. For example, pattern structures on sets of graphs [3, 2, 8] is based on the operation of finding maximal common subgraphs for a set of graphs, which is #P-hard.

To deal with this complexity and to have a possibility to process most of the data, projections of pattern structures were introduced [3]. Projections are special mathematical functions on the set of descriptions that simplify the descriptions of objects. This approach reduces the number of concepts in the pattern lattice corresponding to a pattern structure. However, it does not impact the computational worst-case complexity of the similarity operation. Moreover, it cannot remove concepts of special kinds from the "middle" of the semilattice which can be important in some practical cases, e.g., concepts containing too small graphs can be considered useless but they cannot be removed with projections. For example, in [1] concepts having intents that include short sequences of patient hospitalisations have little sense. Hence, short sequences could be "removed" from the intent, but the descriptions of objects, i.e., patients, usually include only one long sequence and should not be changed.

In this paper we introduce *o-projections* of pattern structures, a generalization of projections of pattern structures, that allow one to reduce the computational complexity of similarity operations. They also allow one to remove certain kinds of descriptions in the "middle" of the semilattice while the descriptions of the objects can be preserved. By introducing o-projections of pattern structures, we correct also some formal problems of projections of pattern structures, which will be discussed later.

The main difference between o-projections and projections is that in o-projected pattern structures we modify the semilattice of descriptions, while in the case of projected pattern structures we can modify only the descriptions of single objects. It should be noticed that most of the properties of projections are valid for o-projections. However, the relation between representation contexts, a reduction from pattern structures to FCA, and projections is quite different from the relation between representation contexts and o-projections. The introduction and study of this difference is one of the main contributions of this work. In addition we have discovered the fact that the set of o-projections of a pattern structure forms a semilattice. From a practical point of view it allows one to apply a set of independent o-projections, e.g., o-projections obtained from several experts, to a pattern structure.

This work further develops the methodology introduced in [1], where it was applied for the analysis of sequential pattern structures by introducing projections that remove irrelevant concepts.

The rest of the paper is organized as follows. In Section 2 we introduce the definitions of a pattern structure, representation context of a pattern structure, and discuss how one can compute with pattern structures along the lines of FCA. Section 3 introduces projections and o-projections of a pattern structure, defines the partial order on o-projections and shows that this order is a semilattice. At the end of this section the relation between o-projections and representation

contexts of o-projected pattern structure is discussed. Finally, we conclude the paper and discuss furture work.

2   Pattern Structures

In FCA a formal context (G, M, I), where G is a set of objects, M is a set of attributes, and $I \subseteq G \times M$ is a binary relation between G and M, is taken to a concept lattice $\mathfrak{L}(G, M, I)$ [4]. For non-binary data, such as sequences or graphs, lattices can be constructed in the same way using pattern structures [3].

Definition 1. *A pattern structure $\mathbb{P}$ is a triple $(G, (D, \sqcap), \delta)$, where G, D are sets, called the set of objects and the set of descriptions, and $\delta : G \to D$ maps an object to a description. Respectively, $(D, \sqcap)$ is a meet-semilattice on D w.r.t. $\sqcap$, called similarity operation such that $\delta(G) := \{\delta(g) \mid g \in G\}$ generates a complete subsemilattice $(D_\delta, \sqcap)$ of $(D, \sqcap)$.*

For illustration, let us represent standard FCA in terms of pattern structures. The set of objects G is preserved, the semilattice of descriptions is $(\wp(M), \cap)$, where $\wp(M)$ denotes the powerset of the set of attributes M, a description is a subset of attributes and $\cap$ is the set-theoretic intersection. If $x = \{a, b, c\}$ and $y = \{a, c, d\}$ then $x \sqcap y = x \cap y = \{a, c\}$, and $\delta : G \to \wp(M)$ is given by $\delta(g) = \{m \in M \mid (g, m) \in I\}$.

Note that Definition 1 has an important partial case where $(D, \sqcap)$ is a complete meet-semilattice. In this case the semilattice $(D_\delta, \sqcap)$ is necessarily complete. First, in practical applications one often needs finite lattices, which are always complete. Second, in many practical cases one can easily extend an incomplete semilattice to a complete one by introducing some extra elements. For example, given an incomplete semilattice w.r.t containment order on the interval (a, b), one can add a and b to obtain the interval $[a, b]$, which is a complete semilattice. In this paper some of the statements hold only for the partial case of $(D, \sqcap)$ being a complete meet-semilattice.

The Galois connection for a pattern structure $(G, (D, \sqcap), \delta)$, relating sets of objects and descriptions, is defined as follows:

$$A^\diamond := \bigsqcap_{g \in A} \delta(g), \qquad\qquad \text{for } A \subseteq G$$

$$d^\diamond := \{g \in G \mid d \sqsubseteq \delta(g)\}, \qquad\qquad \text{for } d \in D$$

Given a subset of objects A, $A^\diamond$ returns the description which is common to all objects in A. Given a description d, $d^\diamond$ is the set of all objects whose description subsumes d. The natural partial order (or subsumption order between descriptions) $\sqsubseteq$ on D is defined w.r.t. the similarity operation $\sqcap$: $c \sqsubseteq d \Leftrightarrow c \sqcap d = c$ (in this case we say that c is subsumed by d). In the case of standard FCA the natural partial order corresponds to the set-theoretical inclusion order, i.e., for two sets of attributes x and y $x \sqsubseteq y \Leftrightarrow x \subseteq y$.
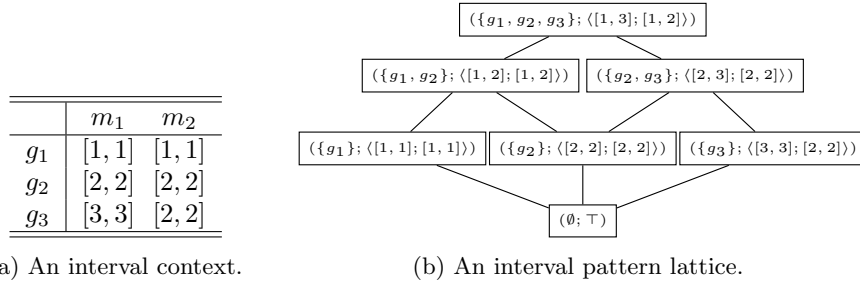
3

	m_1	m_2
g_1	$[1,1]$	$[1,1]$
g_2	$[2,2]$	$[2,2]$
g_3	$[3,3]$	$[2,2]$

(a) An interval context.   (b) An interval pattern lattice.

Figure 1: An interval pattern structure and the corresponding lattice.

Definition 2. *A pattern concept of a pattern structure* $(G, (D, \sqcap), \delta)$ *is a pair* (A, d)*, where* $A \subseteq G$ *and* $d \in D$ *such that* $A^\diamond = d$ *and* $d^\diamond = A$*; A is called the pattern extent and d is called the pattern intent.*

As in standard FCA, a pattern concept corresponds to the maximal set of objects A whose description subsumes the description d, where d is the maximal common description of objects in A. The set of all pattern concepts is partially ordered w.r.t. inclusion of extents or, dually, w.r.t. subsumption of pattern intents within a concept lattice, these two antiisomorphic orders making a lattice, called pattern lattice.

2.1 Running Example

The authors of [6] have used interval pattern structures for gene expression analysis. Let us consider an example of such pattern structures. In Figure 1a an interval context is shown. It has three objects and two attributes. Every attribute shows the interval of values the attribute can have. If we have two objects, then a numerical attribute can have all values from the interval of this attribute in the first object and from the interval of this attribute of the second object. Consequently, the similarity between two intervals can be defined as a convex hull of the intervals, i.e. $[a, b] \sqcap [c, d] = [\min(a, c), \max(b, d)]$. Then, given two tuples of intervals, the similarity between these tuples is computed as a component-wise similarity between intervals.

In this example, we have the pattern structure $(G, (D, \sqcap), \delta)$, where $G = \{g_1, g_2, g_3\}$, the set D is the set of all possible interval pairs with the similarity operation described above, and δ is given by the context in Figure 1a, i.e., $\delta(g_1) = \langle [1, 1]; [1, 1] \rangle$ and $\delta(g_1) \sqcap \delta(g_2) = \langle [1, 2]; [1, 2] \rangle$.

Figure 1b shows the pattern lattice of the interval context in Figure 1a. One can check that the extents and the intents in this lattice are connected by means of the Galois connection given above. The partial order in the semilattice of intervals is given by "the smaller the interval, the larger the description with this interval", i.e., the former description gives more certainty about the values than the latter.

4

2.2 Representation Context of a Pattern Structure

Note that any pattern structure can be represented by a formal context with the concept lattice isomorphic to the lattice of the pattern structure. Below we introduce a representation context of a pattern structure and its properties in the line of [3].

Given a pattern structure $(G, (D, \sqcap), \delta)$, we denote by $D_\delta \subseteq D$ the set of all intents of the concept lattice, i.e., $D_\delta = \{d \in D \mid (\exists X \subseteq G) \bigsqcap_{g \in X} \delta(g) = d\}$. Since $(D_\delta, \sqcap)$ is a complete subsemilattice of $(D, \sqcap)$, for $X \subseteq D$ a join operation $\sqcup$ can be defined as follows:

$$\bigsqcup X = \bigsqcap \{d \in D_\delta \mid (\forall x \in X) x \sqsubseteq d\}.$$

Given this join operation, $(D_\delta, \sqcap, \sqcup)$ is a complete lattice. We say that a set $M \subseteq D$ is $\sqcup$-dense for $(D_\delta, \sqcap)$ if every element in D_δ is of the form $\sqcup X$ for some $X \subseteq M$. For example, $M = D_\delta$ is always $\sqcup$-dense for D_δ.

Definition 3. *Given a pattern structure $\mathbb{P} = (G, (D, \sqcap), \delta)$ and a set $M \subseteq D$ $\sqcup$-dense in D_δ, a formal context (G, M, I) is called the representation context of $\mathbb{P}$, if I is given by $I = \{(g, m) \in G \times M \mid m \sqsubseteq \delta(g)\}$. The representation context of $\mathbb{P}$ is denoted by $\mathbb{R}(\mathbb{P})$.*

The next theorem establishes a bijection between the pattern concepts in the lattice of pattern structure $\mathbb{P}$ and the concepts in the lattice of the representation context $\mathbb{R}(\mathbb{P})$. Here, the ideal of element $d \in D$ is denoted by $\downarrow d = \{e \in D \mid e \sqsubseteq d\}$.

Theorem 1 (Theorem 1 from [3]). *Let $\mathbb{P} = (G, (D, \sqcap), \delta)$ be a pattern structure and let $\mathbb{R}(\mathbb{P}) = (G, M, I)$ be a representation context of $\mathbb{P}$. Then for any $A \subseteq G$, $B \subseteq M$, and $d \in D$ the following conditions are equivalent:*

1. *(A, d) is a pattern concept of $\mathbb{P}$ and $B = \downarrow d \cap M$.*

2. *(A, B) is a formal concept of $\mathbb{R}(\mathbb{P})$ and $d = \bigsqcup B$.*

Example 1. *A representation context for the pattern structure given in Figure 1 can be given by the set M where every element $m \in M$ is of the form $\langle [-\infty, a]; [-\infty, +\infty] \rangle$ or $\langle [-\infty, +\infty]; [b, +\infty] \rangle$, and $a, b \in \{1, 2, 3\}$.*

In fact, the element $\langle [-\infty, +\infty]; [a, +\infty] \rangle$ corresponds to the attribute 'm$_2 \geq$ a' in the case of the interordinal scaling [4] of numerical data. Another representation context can be constructed from the intents of join-irreducible concepts of the lattice in Figure 1b. These two representation contexts of the pattern structure related to Figure 1 are shown in Figures 2a and 2b. It can be seen that the resulting lattices, e.g., the lattice in Figure 2c, are isomorphic to the lattice in Figure 1b.

It should be noticed that in some cases the representation context is hard to compute. For example, in case of numerical data with the set of all values

	⟨[3,+∞]; [−∞,+∞]⟩	⟨[2,+∞]; [−∞,+∞]⟩	⟨[−∞,1]; [−∞,+∞]⟩	⟨[−∞,2]; [−∞,+∞]⟩	⟨[−∞,+∞]; [2,∞]⟩	⟨[−∞,+∞]; [−∞,1]⟩
	$m_1 \geq 3$	$m_1 \geq 2$	$m_1 \leq 1$	$m_1 \leq 2$	$m_2 \geq 2$	$m_2 \leq 1$
g_1			x	x		x
g_2		x		x	x	
g_3	x	x			x	

	⟨[1,3]; [1,2]⟩	⟨[3,3]; [2,2]⟩	⟨[1,2]; [1,2]⟩	⟨[2,3]; [2,2]⟩	⟨[1,3]; [1,2]⟩
	a_1	a_2	a_3	a_4	a_5
g_1	x		x		x
g_2	x		x	x	x
g_3	x	x		x	x

(a) Representation context corresponding to interordinal scaling.

(b) Another possible representation context.
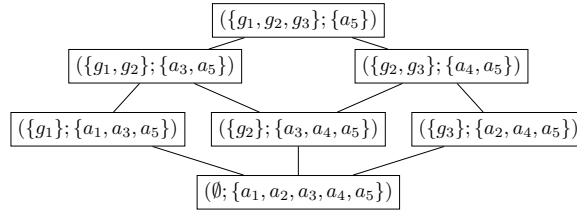
(c) A concept lattice for the context if Figure 2b.

Figure 2: Possible representation contexts for the pattern structure in Figure 1 and the concept lattice for the context in Figure 2b.

W, to construct representation context, one needs to create $2|W| + 1$ binary attributes, which can be much more than the number of original real-valued attributes. The authors of [6] have shown that pattern structures provide more efficient computations than the equivalent approach based on FCA and scaling, which can be considered as a way to build representation context of interval pattern structures, e.g., see Figure 2a.

In case of graph data the set of attributes of the representation context consists of all subgraphs of the original graph descriptions, which is hard to compute [8].

2.3 Computation of Pattern Lattices

Nearly any algorithm for computing concept lattices from contexts can be adapted to compute pattern lattices from pattern structures. To adapt an algorithm, every set intersection operation on attributes is replaced by the semilattice operation $\sqcap$ on corresponding patterns, and every subset checking is replaced by the semilattice order $\sqsubseteq$ checking, in particular, all $(\cdot)'$ operations are replaced by $(\cdot)^\diamond$. For example, let us consider a modified version of Close-by-One (CbO) algorithm [7].

Algorithm 1 shows the listing of the modified part of CbO. Here the canonical

```
1  Function CloseByOne(Ext, Int)
       Data: ℙ = (G, (D, ⊓), δ), the extent Ext and the intent Int of a
             concept.
       Result: All canonical ancestors of (Ext, Int) in the concept lattice.
2      foreach S ⊆ G, S ≻ Ext do
3          NewInt ⟵ ⊓ δ(g) ;                    /* ⊓ - the similarity */
                      g∈S
4          NewExt ⟵ {g ∈ G | NewInt ⊑ δ(g)} ;            /* ⊑ - the
           subsumption */
5          if IsCanonicExtension(Ext, NewExt) then
6              SaveConcept((NewExt, NewInt));
7              CloseByOne(NewExt,NewInt);
8  /* Looking for all concepts of the concept lattice            */
9  CloseByOne(∅, ⊤);
```

Algorithm 1: The version of the Close-by-One algorithm computing the
pattern lattice of a pattern structure ℙ.

extension `IsCanonicExtension` and canonical order $\succ$ are defined on the set of
objects and hence are the same as in [7]. We can see that only lines 3 and 4 are
modified. In these lines the set intersection operation and the subset relation
checking are replaced by the corresponding operators of a pattern structure.
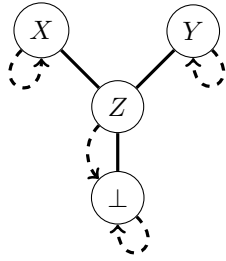
3 Revised Projections of Pattern Structures

Pattern structures are hard to process due to the large number of pattern con-
cepts in the pattern lattice and the algorithmic complexity of the similarity
operation $\sqcap$. Projections of pattern structures "simplify" to some degree the
computation and allow one to work with "simpler" descriptions. In fact, a pro-
jection can be considered as a mapping for pruning descriptions with certain
mathematical properties. These properties ensure that a projection of a semi-
lattice is a semilattice and that the concepts of a projected[1] pattern structure
are related to the concepts of the original pattern structure [3].

In this section we introduce o-projected pattern structures ("o" coming from
"order"), i.e. a revision of projected pattern structures in accordance with [1].
We discuss the properties of o-projected pattern structures and relate them to
the projected pattern structures from [3]. The notion of (o-)projected pattern
structure is based on a kernel operator (a projection).

Definition 4 ([3]). *A projection $\psi : D \to D$ is a kernel (interior) operator
on the partial order $(D, \sqsubseteq)$, i.e. it is (1) monotone $(x \sqsubseteq y \Rightarrow \psi(x) \sqsubseteq \psi(y))$,
(2) contractive $(\psi(x) \sqsubseteq x)$ and (3) idempotent $(\psi(\psi(x)) = \psi(x))$.*

---

[1] We use the expression "a projected pattern structure" instead of "a projection of a
pattern structure" to distinguish between projection as an operator ψ and as the result of
applying the operator to a pattern structure.

$$D = \{x, y, z, \bot\}$$

$$\psi : x \mapsto x, y \mapsto y,$$
$$z \mapsto \bot, \bot \mapsto \bot$$

$$\psi(x \sqcap y) = \psi(z) = \bot \neq$$
$$\neq z = \psi(x) \sqcap \psi(y)$$

Figure 3: Contrexample to Proposition 1 from [3].

Given a projection ψ we say that the fixed point of ψ is the set of all elements from D such that they are mapped to themselves by ψ. The fixed point of ψ is denoted by $\psi(D) = \{d \in D \mid \psi(d) = d\}$. Note that, if $\psi(d) \neq d$, then there is no other $\tilde{d}$ such that $\psi(\tilde{d}) = d$ because of idempotency. Hence, any element outside the fixed point of the projection ψ is pruned.

3.1 Definition of Projected Pattern Structures

Let us first consider the projected pattern structure w.r.t. a projection ψ according to [3]. Given a pattern structure $\mathbb{P} = (G, (D, \sqcap), \delta)$ and a projection ψ on D, the projected pattern structure is defined as $(G, (D, \sqcap), \psi \circ \delta)$. As we can see, a projection only changes the descriptions of the objects but not the underlying semilattice $(D, \sqcap)$. There are two problems with this definition of the projected pattern structures. First, it is necessary to restrict the class of projections given by Definition 4 in order to ensure the property $\psi(x \sqcap y) = \psi(x) \sqcap \psi(y)$. Second, the complexity of computing $\sqcap$ can be very high, but with this kind of projected patten structures we cannot decrease the algorithmic complexity. Below we discuss these two points.

In [3] (Proposition 1) the following property of the projection operator is discussed: given a semilattice $(D, \sqcap)$ and a projection ψ on D, for any two elements x and y from D one has $\psi(x \sqcap y) = \psi(x) \sqcap \psi(y)$. Let us consider the example in Figure 3 with the meet-semilattice $D = \{x, y, z, \bot\}$ given by its diagram and the projection ψ given by the dotted lines. It is easy to see that $\psi(x \sqcap y) = \bot \neq z = \psi(x) \sqcap \psi(y)$. One way of solving this problem is to give additional conditions on projection ψ that would imply the required property. An important example is the following condition: for all $x, y \in D$ if $x < y$ and $\psi(y) = y$, then $\psi(x) = x$. This kind of solution respects the intuition behind the definition of the projected pattern structure in [3], according to which the initial descriptions of objects are changed, but the similarity operation $\sqcap$ is not changed.

Another way of solving the problem above is to generalize the definition of the projected pattern structure, and we proceed in this way in the next section,

by allowing to modify the similarity operation on descriptions.

3.2 Definition of o-projected Pattern Structures

Below we propose a definition of o-projected pattern structures by means of a kernel operator ψ. The definition takes into account the problems discussed above. In the o-projected pattern structure we substitute the semilattice of descriptions by its suborder (the letter "o" comes from "order") with another similarity operation, which can be different from the initial one.

Let us first note that, given a meet-semilattice D and a kernel operator ψ, the fixed point $\psi(D)$ is a semilattice w.r.t. to the natural order on D.

Theorem 2. *Given a semilattice $(D, \sqcap)$ and a kernel operator ψ, the fixed point $(\psi(D), \sqcap_\psi)$ is a semilattice w.r.t. the natural order on $(D, \sqcap)$, i.e., $d_1 \sqsubseteq d_2 \Leftrightarrow d_1 \sqcap d_2 = d_1$. If $\sqcap X$ exists for a set $X \subseteq D$, then $\bigsqcap_\psi \psi(x)$ exists and is given by*

$$\bigsqcap_{\substack{\psi \\ x \in X}} \psi(x) = \psi\left(\bigsqcap_{x \in X} x\right) \tag{1}$$

Proof. Let us denote $d = \bigsqcap_{x \in X} x$. Since $(\forall x \in X) d \sqsubseteq x$, one has $(\forall x \in X) \psi(d) \sqsubseteq \psi(x)$. Let us show that for any $p \in \psi(D)$, i.e. $\psi(p) = p$ such that $(\forall x \in X) p \sqsubseteq \psi(x)$, we have $p \sqsubseteq \psi(d)$, i.e., that $\psi(d) = \bigsqcap_{\substack{\psi \\ x \in X}} \psi(x)$.

Since $(\forall x \in X) p \sqsubseteq \psi(x)$ then $(\forall x \in X) p \sqsubseteq x$. Since $d = \bigsqcap_{x \in X} x$, one has $p \sqsubseteq d$. Thus, $p = \psi(p) \sqsubseteq \psi(d)$ and $\psi(d)$ is the minimum of the set $\psi(X)$, i.e. $\psi(D)$ is a semilattice and the Eq. (1) holds. $\square$

Corollary 1. *Given a complete subsemilattice $\tilde{D}$ of $(D, \sqcap)$ and a kernel operator ψ on D, the image of $\tilde{D}$ is a complete subsemilattice $\psi(\tilde{D})$ of the fixed point $(\psi(D), \sqcap_\psi)$.*

Since according to Theorem 2 $\psi(D)$ is a semilattice and according to Corollary 1 $\psi(D_\delta)$ is a complete semilattice, we can define an o-projected pattern structure as a pattern structure with $\psi(D)$ as a semilattice.

Definition 5. *Given a pattern structure $\mathbb{P} = (G, (D, \sqcap), \delta)$ and a kernel operator ψ on D, the o-projected pattern structure $\psi(\mathbb{P})$ is a pattern structure $(G, (\psi(D), \sqcap_\psi), \psi \circ \delta)$, where $\psi(D) = \{d \in D \mid \psi(d) = d\}$ and $\forall x, y \in D, x \sqcap_\psi y := \psi(x \sqcap y)$.*

In the o-projected pattern structure the kernel operator ψ modifies not only the descriptions of the objects, but also the semilattice operation, i.e., the semilattice $(\psi(D), \sqcap_\psi)$ is not necessarily a subsemilattice of $(D, \sqcap)$ and so it is not always true that $x \sqcap y = x \sqcap_\psi y$ in D.

Example 2. *Let us define an o-projection for the interval pattern structure from Subsection 2.1. Let us suppose that the aggregated size of a pattern, i.e.,*

9

the sum of the lengths of the intervals in the pattern, should be less than 2. First, we should define the corresponding kernel operator $\psi : D \to D$. Thus, if an aggregated length of a pattern p is less than 2, then $\psi(p) := p$, otherwise $\psi(p) := \perp = \langle [-\infty, +\infty]; [-\infty, +\infty] \rangle$. For instance, $\psi(\langle [1,1]; [1,1] \rangle) = \langle [1,1]; [1,1] \rangle$, while $\psi(\langle [1,2]; [1,2] \rangle) = \langle [-\infty, +\infty]; [-\infty, +\infty] \rangle$, because it has two intervals of length 1, i.e., the aggregated size is equal to 2.

Let us consider the o-projected interval pattern structure $(G, (\psi(D), \sqcap_\psi), \psi \circ \delta)$. It is clear that $\psi \circ \delta = \delta$, thus this o-projected interval pattern structure cannot be expressed as a projected pattern structure.

The concepts of a pattern structure and a projected pattern structure are connected through Proposition 1. This proposition can be found in [3], but thanks to Theorem 2, it is also valid in our case.

Proposition 1. *Given a pattern structure $\mathbb{P} = (G, (D, \sqcap), \delta)$ and a kernel operator ψ on D:*

1. *if A is an extent in $\psi(\mathbb{P})$, then A is also an extent in $\mathbb{P}$.*

2. *if d is an intent in $\mathbb{P}$, then $\psi(d)$ is also an intent in $\psi(\mathbb{P})$.*

It is easy to see that the other propositions from [3] concerning projected pattern structures hold for the o-projected pattern structures as well. Below we cite Proposition 3 from [3] that relates implications in a pattern structure and those in an o-projected pattern structure. We skip the propositions related to supervised classification with projected pattern structures by means of hypotheses, because it is out of the scope of this paper. However, they are valid in the case of o-projected pattern structures and can be proven with the help of Theorem 2.

Proposition 2 (Proposition 3 from [3])**.** *Let $a, b \in D$. If $\psi(a) \to \psi(b)$ and $\psi(b) = b$ then $a \to b$, where $x \to y \Leftrightarrow$ for all $g \in G$ ($x \sqsubseteq \delta(g)$ implies $y \sqsubseteq \delta(g)$)*

3.3 Order of Projections

In this subsection we limit ourselves to the practically important case when a set of descriptions is a complete semilattice. We can consider projections as a means of description pruning in $(D, \sqcap)$. Indeed, given a semilattice $(D, \sqcap)$ and a projection ψ on this semilattice, the set D can be divided into two sets $D = \{d \in D \mid \psi(d) = d\} \cup \{d \in D \mid \psi(d) \neq d\}$, i.e., the fixed point of ψ and the rest. It can be seen that the intents of the o-projected pattern structure $\psi((G, (D, \sqcap), \delta))$ are in the fixed point of ψ, i.e., all elements of the form $\psi(d) \neq d$ are discarded. We recall that by $\psi(D) = \{d \in D \mid \psi(d) = d\}$ we denote the fixed point of ψ. *But under which condition do we have that for any $D_1 \subset D_2$ there is a projection ψ of D_2 such that $\psi(D_2) = D_1$?* The following theorem gives necessary and sufficient conditions for such a property.

Theorem 3. *Given a complete semilattice $(D, \wedge)$, with the natural order $\leq$, and $D_s \subseteq D$, there is a projection $\psi : D \to D$ such that $\psi(D) = D_s$, if and only*

if $\perp \in D_s$ *and for any* $X \subseteq D_s \subseteq D$, *one has* $\bigvee X \in D_s$, *where* $\perp := \bigwedge D$ *and* $\bigvee X = \bigwedge \{d \in D \mid (\forall x \in X) d \geq x\}$.

Proof. 1. Given a projection ψ such that $\psi(D) = D_s$, $\perp \in D_s$ because of contractivity of ψ, i.e., $\psi(\perp) = \perp$. Let us suppose that there is a set $X \subseteq D_s$, i.e., $(\forall x \in X)\psi(x) = x$ such that $\psi(\bigvee X) \neq \bigvee X$. Then, $(\forall x \in X)(x < \bigvee X \underset{\text{monotonicity}}{\Rightarrow} x \leq \psi(\bigvee X) \underset{\text{contractivity}}{<} \bigvee X)$. It is a contradiction, since $\bigvee X$ is the supremum of X. Hence for any $X \subseteq D_s$ we have $\psi(\bigvee X) = \bigvee X$.

2. Given $D_s \subseteq D$ such that $\perp \in D_s$ and for any $X \subseteq D_s$, one has $\bigvee X \in D_s$, let us construct the corresponding projection ψ. First, $\psi(d \in D_s) := d$ and for all $d \in D \backslash D_s$ we should have $\psi(d) \neq d$. For an element $d \in D \backslash D_s$, let us consider the set $S_d = \{x \in D_s \mid x < d\}$, which is not an empty set since $\perp \in D_s$. We know that $\bigvee S_d \in D_s$ and by definition of $\bigvee$ we have $\bigvee S_d < d$. Then we set $\psi(d) := \bigvee S_d$.

Let us show that the function ψ is a projection of D. Idempotency and contractivity are satisfied by the construction of ψ. Let us check monotonicity. Let us take any $a, b \in D$ such that $a > b$. Then, if $\psi(a) = a$, then $\psi(a) = a > b \geq \psi(b)$, i.e., the monotonicity holds. If $\psi(a) \neq a$, then $\psi(a) = \bigvee S_a$ by construction. Hence, if $\psi(b) = b$, then $b \in S_a$, i.e., $\psi(a) \geq \psi(b)$. Finally, if $\psi(b) \neq b$, then $S_b \subseteq S_a$, because if $d \in S_b$, i.e., $d < b$, then $d < b < a$, i.e. $d \in S_a$. In this case, $\psi(a) = \bigvee S_a \geq \bigvee S_b = \psi(b)$.

$\square$

Corollary 2. *Given a complete semilattice* $(D, \wedge)$, *with the natural order* $\leq$, *and a subset* $D_s \subseteq D$ *such that* $\perp \in D_s$ *and for any* $X \subseteq D_s$, *one has* $\bigvee X \in D_s$, *the poset* $(D_s, \leq)$ *is a complete semilattice.*

Proof. According to Theorem 3 there is a projection $\psi : D \to D$ such that $\psi(D) = D_s$. Then, according to Theorem 2 D_s is a semilattice. $\square$

Since a projection of D can be considered as a mapping with the fixed point $\psi(D)$, we can introduce an order w.r.t. this fixed point.

Definition 6. *Given a complete semilattice* $(D, \sqcap)$ *and two projections* ψ_1 *and* ψ_2 *in* D, *we say that* $\psi_1 \leq \psi_2$ *if* $\psi_1(D) \subseteq \psi_2(D)$.

However in some cases, it is more convenient to order projections w.r.t. a superposition of projections or their "generality".

Definition 7. *Given a complete semilattice* $(D, \sqcap)$ *and two projections* ψ_1 *and* ψ_2 *in* D, *we say that* $\psi_1 \leq \psi_2$ *if there is a projection* $\psi : \psi_2(D) \to \psi_2(D)$ *such that* $\psi_1 = \psi \circ \psi_2$.

It can be seen that these two definitions yield the same ordering.

Proposition 3. *Definitions 6 and 7 are equivalent.*

Proof.   1. Let $\psi_1 = \psi \circ \psi_2$. Since ψ is a projection in $\psi_2(D)$, then $\psi_1(D) = \psi(\psi_2(D)) \subseteq \psi_2(D)$.

2. Let $\psi_1(D) \subseteq \psi_2(D)$. Let us denote by $(\cdot)_1$ and $(\cdot)_2$ the operations in $(\psi_1(D), \sqcap_{\psi_1})$ and $(\psi_2(D), \sqcap_{\psi_2})$, respectively, and let us denote $D_i = \psi_i(D)$ the fixed points of ψ_i, where $i \in \{1, 2\}$.

Let us build $\psi : D_2 \to D_1$ equal to ψ_1 in D_2, i.e., for all $d \in D_2$ we set $\psi(d) := \psi_1(d)$. Since ψ_1 is a projection in D, ψ is a projection in D_2 (the natural order is the same). Since D_1 is the fixed point of ψ_1 then $\psi_1(D_2) \subseteq D_1$. However, since $D_1 \subseteq D_2$ and $\psi_1(D_1) = D_1$ then $\psi_1(D_2) = D_1$, i.e., there is a projection ψ such that $\psi_1 = \psi \circ \psi_2$. $\qquad\square$

Example 3. *Let us return to Example 2. We change the threshold for the aggregated size. In Example 2 it was set to 2 ($\psi_{al=2}$), but we can change it to 5 ($\psi_{al=5}$) or 10 ($\psi_{al=10}$). The higher the threshold, the more possible descriptions are projected to themselves, i.e., belong to the fixed point of the projection. Thus, we have $\psi_{al=2} \leq \psi_{al=5} \leq \psi_{al=10}$.*

Thanks to Proposition 1 it can be seen that, given a pattern structure $\mathbb{P}$, if we have two projections $\psi_1 \leq \psi_2$, then the set of pattern extents of $\psi_1(\mathbb{P})$ is a subset of the set of pattern extents of $\psi_2(\mathbb{P})$, i.e., the smaller the projection, the smaller the number of concepts in the corresponding projected pattern structure.

Now it can be seen that projections actually form a semilattice with respect to the previously defined order.

Proposition 4. *Projections of a complete semilattice $(D, \sqcap)$ ordered by Definition 6 or 7 form a semilattice $(\mathbb{F}, \wedge)$, where the semilattice operation between $\psi_1, \psi_2 \in \mathbb{F}$ is given by $\psi_1 \wedge \psi_2 = \psi_3$ iff $\psi_3(D) = \psi_1(D) \cap \psi_2(D)$.*

Proof. It follows from the definitions that if for any ψ_1 and ψ_2 the projection ψ_3 exists, then projections of D form a semilattice. Let us describe the corresponding ψ_3.

Let us denote $D_1 = \psi_1(D)$ and $D_2 = \psi_2(D)$ and $D_3 = D_1 \cap D_2$. Let us suppose that there exist $x, y \in D_3$ such that $x \sqcup y \notin D_3$. But as $D_3 \subseteq D_1$ and $D_3 \subseteq D_2$, then, since ψ_1 is a projection of D and ψ_2 is a projection of D, we have $x \sqcup y \in D_1$ and $x \sqcup y \in D_2$, i.e., $x \sqcup y \in D_1 \cap D_2 = D_3$. Thus, $(\forall x, y \in D_3) x \sqcup y \in D_3$. Then, according to Theorem 3 there is a projection ψ_3 such that $\psi_3(D) = D_3$. $\qquad\square$

3.4   Analogue of Theorem II for Revised Projections

An important question is *how a projection changes the representation context of a pattern structure?* We limit the discussion of this question for the case when a set of description D is a complete semilattice. In [3] the authors describe this change by means of Theorem 2. The formulation of this theorem was corrected in [5]. Below we give the corrected version of the theorem.

Theorem 4 (Theorem 2 from [3]). *For two pattern structures $(G, (D, \sqcap), \delta_1)$ and $(G, (D, \sqcap), \delta_2)$ the following statements are equivalent:*

1. $\delta_2 = \psi \circ \delta_1$ *for some ψ on $(D, \sqcap)$.*

2. $(\forall g \in G)(\delta_2(g) \sqsubseteq \delta_1(g))$ *and there is a representation context (G, M, I) of $(G, (D, \sqcap), \delta_1)$ and some $N \subseteq M$ such that $(G, N, I \cap (G \times N))$ is a representation context of $(G, (D, \sqcap), \delta_2)$.*

In Theorem 4 one compares two pattern structures that differ in mapping functions. However, in the o-projected pattern structures we can modify the lattice structure itself. *How can we adjust the formulation of Theorem 4 in such a way that it can be applied to revised projections?* First, we should notice that in a pattern structure and in an o-projected pattern structure the set of objects is preserved. Second, the minimal representation context of a pattern structure can have less attributes than the minimal representation context of an o-projected pattern structure, as shown in Example 4.

Example 4. *Let $M = \{a, b, c\}$ and the description semilattice be $D = (2^M, \cap)$. Let $\psi : 2^M \to 2^M$ be the following mapping: $\psi(\{a\}) = \emptyset$ and for any $A \neq \{a\}$ we put $\psi(A) = A$. This projection is visualised in Figure 4a by dashed arrows. Let us consider the following pattern structure $(\{g_1, g_2, g_3\}, (2^M, \cap), \{g_1 \mapsto \{a, b\}, g_2 \mapsto \{a, c\}, g_3 \mapsto \{b, c\}\}$.*
The minimal representation context of this pattern structure contains 3 attributes $M = \{a, b, c\}$, while the minimal representation context of the o-projected pattern structure contains 4 attributes $M_\psi = \{b, c, ab, ac\}$. The corresponding contexts are shown in Figures 4b and 4c.

We can see that to introduce the "revised Theorem 2" from [3] we have to define a special relation between contexts.

Definition 8. *Given two contexts $\mathbb{K}_1 = (G, M_1, I_1)$ and $\mathbb{K}_2 = (G, M_2, I_2)$, $\mathbb{K}_1$ is said to be simpler than $\mathbb{K}_2$, denoted by $\mathbb{K}_1 \leq_S \mathbb{K}_2$, if for any $m_{1,i} \in M_1$ there is a set $B_2 \subseteq M_2$ such that $(\{m_{1,i}\})^1 = (B_2)^2$. Here by $(\cdot)^1$ and $(\cdot)^2$ we denote the derivation operators in the contexts $\mathbb{K}_1$ and $\mathbb{K}_2$, respectively.*

Example 5. *The context in Figure 4c is smaller w.r.t. Definition 8 than the context in Figure 4b because every column of the context in Figure 4c is the intersection of a subset of columns of the context in Figure 4b.*

This relation between contexts is a preorder. Indeed, it is reflexive, transitive, but not necessarily antisymmetric: given two contexts $\mathbb{K}_1$ and $\mathbb{K}_2$, if $\mathbb{K}_1$ and $\mathbb{K}_2$ have the same closure system of attributes, i.e., the same set of intents in the concept lattice, then according to the definition $\mathbb{K}_1 \leq_S \mathbb{K}_2$ and $\mathbb{K}_1 \geq_S \mathbb{K}_2$. However, we can consider only the context with the minimal number of attributes in the class of equivalence, i.e., the attribute-reduced context. For simplicity in the rest of the paper we consider only attribute-reduced contexts.

This definition of the simplicity order on contexts can be related to context bonds [4] in the following way. Three formal contexts $\mathbb{K}_i = (G_i, M_i, I_i)$ form a
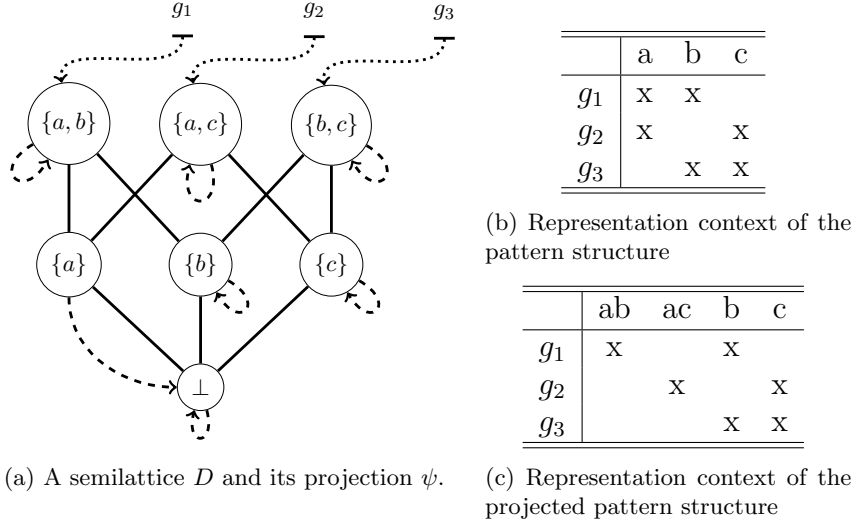
(a) A semilattice D and its projection ψ.

	a	b	c
g_1	X	X	
g_2	X		X
g_3		X	X

(b) Representation context of the pattern structure

	ab	ac	b	c
g_1	X		X	
g_2		X		X
g_3			X	X

(c) Representation context of the projected pattern structure

Figure 4: An example of a projection that can increase the number of attributes in the minimal represenation context.

bond if $\mathbb{K}_1 \leq_S \mathbb{K}_2$ and $\mathbb{K}_2^T \leq_S \mathbb{K}_3^T$, where $\mathbb{K}^T = (M, G, I^T)$. Simplicity order can also be considered as a generalization of "closed-relation-of" order between contexts:

Definition 9 (Definition 50 from [4])**.** *A binary relation $J \subseteq I$ is called a **closed relation** of the context (G, M, I) if every concept of the context (G, M, J) is also a concept of (G, M, I).*

From Definitions 8 and 9 it can be seen that if J is a closed relation of (G, M, I), then $(G, M, J) \leq_S (G, M, I)$, but not always in the other direction. The following theorem gives a relation between kernel operators of D and the change in the representation context of o-projected pattern structures.

Theorem 5. *Given a pattern structure $\mathbb{P} = (G, (D, \sqcap), \delta)$ such that $(D, \sqcap)$ is a complete semilattice the following holds:*

1. *for any projection ψ of D we have $\mathbb{R}(\psi(\mathbb{P})) \leq_S \mathbb{R}(\mathbb{P})$.*

2. *for any context $\mathbb{K} = (G, M, I)$ such that $\mathbb{K} \leq_S \mathbb{R}(\mathbb{P})$, there is a projection ψ of D such that $\mathbb{K}$ is a representation context of $\psi(\mathbb{P})$.*

Proof. 1. The first statement follows from the fact that any extent of $\psi(\mathbb{P})$ is an extent of $\mathbb{P}$ (Proposition 1).

2. Given a pattern structure $\mathbb{P}$ and a context $\mathbb{K}$ such that $\mathbb{K} \leq_S \mathbb{R}(\mathbb{P})$, let us define the set $D_M = \{d \in D \mid (\exists m \in M)(m')^\diamond = d\}$ (notice that for $\mathbb{K}$ and $\mathbb{P}$ there is the same set G, thus, given $A \subseteq G$, both A' and

14

$A^\diamond$ are defined in $\mathbb{K}$ and $\mathbb{P}$ correspondingly). Since $\mathbb{K} \leq_S \mathbb{R}(\mathbb{P})$, m' is an extent of $\mathbb{P}$. Thus, we can see that there is a bijection between D_M and M given by $m' = d^\diamond$. We denote this bijection by $f(m) = d$, i.e. $f(m) = d \Leftrightarrow m' = d^\diamond$. Correspondingly, given a subset $N \subseteq M$, we denote by $f(N) = \{d \in D_M \mid f^{-1}(d) \in N\}$, i.e., $f(M) = D_M$.

Let us define $D_\psi = \{d \in D \mid (\exists X \subseteq D_M) \bigsqcup X = d\}$. According to Theorem 3 there is a projection ψ such that $D_\psi = \psi(D)$.

Let us consider the o-projected pattern structure $\psi(\mathbb{P})$. The set D_M is $\sqcup$-dense for $\psi(D)$, i.e., the context (G, D_M, I_{D_M}), where $(g, d) \in I_{D_M} \Leftrightarrow \psi \circ \delta(g) \sqsupseteq d$, is a representation context of $\psi(\mathbb{P})$. There is the bijection between D_M and M. Let us show that the relation I is similar to the relation I_M, i.e., $(g, m) \in I \Leftrightarrow (g, f(m)) \in I_{D_M}$.

It can be seen that for all $g \in G$ and all $d \in f(g')$, we get $\psi \circ \delta(g) \sqsupseteq d$, because for any $d \in f(g')$ we have $g \in d^\diamond$. Moreover, for any $\tilde{d} \in D \setminus f(g')$ we have $d \not\sqsupseteq \psi \circ \delta(g)$. Thus, the context $\mathbb{K}$ and the context (G, D_M, I_{D_M}) are similar, and hence for any context $\mathbb{K} \leq_S \mathbb{R}(\mathbb{P})$ there is a projection such that $\mathbb{K}$ is a representation context of $\psi(\mathbb{P})$.

$\square$

4 Conclusion

In this paper we have introduced o-projections of pattern structures that are based on kernel operators $\psi : D \to D$. O-projections are a generalization of projections of pattern structures and allow one to change the semilattice of descriptions in o-projected pattern structures. Thus, the complexity of similarity (semilattice) operation can be reduced. Moreover, O-projections also correct a formal problem of projections.

We have shown that o-projections form a semilattice. This can be important when several independent o-projections are applied to a pattern structure. For example, if projections are discussed with several experts it may happen that several types of projections should be combined. In the case of several independent projections we know that there is the only one o-projection w.r.t. the semilattice of o-projections that is a combination of these projections.

Finally, we have shown that the representation context of an o-projected pattern structure can have more attributes than the representation context of the pattern structure itself. To describe this change in the representation context after o-projection we have introduced a new order on contexts, with the use of which we have described the way the representation context can change.

An important direction of the future work is to formalize *transformations* of pattern structures, i.e., special homomorphisms between the semilattice of descriptions D and a different semilattice D_1. In particular, it allows one to formalize the mappings of the form $\psi : D \to \mathbb{R}$, an instance of which are kernel functions used in Support Vector Machines (SVM).

References

[1] Aleksey Buzmakov, Elias Egho, Nicolas Jay, Sergei O. Kuznetsov, Amedeo Napoli, and Chedy Raïssi. On Projections of Sequential Pattern Structures (with an application on care trajectories). In *Proc. 10th Int. Conf. Concept Lattices Their Appl.*, pages 199—-208, La Rochelle, France, 2013.

[2] Bernhard Ganter, Peter A. Grigoriev, Sergei O. Kuznetsov, and Mikhail V. Samokhin. Concept-Based Data Mining with Scaled Labeled Graphs. In KarlErich Wolff, HeatherD. Pfeiffer, and HarryS. Delugach, editors, *Concept. Struct. Work SE - 6*, volume 3127 of *Lecture Notes in Computer Science*, pages 94–108. Springer Berlin Heidelberg, 2004.

[3] Bernhard Ganter and Sergei O. Kuznetsov. Pattern Structures and Their Projections. In Harry S. Delugach and Gerd Stumme, editors, *Concept. Struct. Broadening Base*, volume 2120 of *Lecture Notes in Computer Science*, pages 129–142. Springer Berlin Heidelberg, 2001.

[4] Bernhard Ganter and Rudolf Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer, 1st edition, 1999.

[5] Tim B. Kaiser and Stefan E. Schmidt. Some Remarks on the Relation between Annotated Ordered Sets and Pattern Structures. In Sergei O. Kuznetsov, DebaP. Mandal, MalayK. Kundu, and SankarK. Pal, editors, *Pattern Recognit. Mach. Intell. SE - 9*, volume 6744 of *Lecture Notes in Computer Science*, pages 43–48. Springer Berlin Heidelberg, 2011.

[6] Mehdi Kaytoue, Sergei O. Kuznetsov, Amedeo Napoli, and Sébastien Duplessis. Mining gene expression data with pattern structures in formal concept analysis. *Inf. Sci. (Ny).*, 181(10):1989 – 2001, 2011.

[7] Sergei O. Kuznetsov. A fast algorithm for computing all intersections of objects from an arbitrary semilattice. *Nauchno-Tekhnicheskaya Informatsiya Seriya 2 (Autom. Doc. Mathem. Ling.)*, (1):17–20, 1993.

[8] Sergei O. Kuznetsov and Mikhail V. Samokhin. Learning Closed Sets of Labeled Graphs for Chemical Applications. In Stefan Kramer and Bernhard Pfahringer, editors, *Inductive Log. Program. SE - 12*, volume 3625 of *Lecture Notes in Computer Science*, pages 190–208. Springer Berlin Heidelberg, lecture no edition, 2005.