Relational Data Analysis

SSIE 540 – Spring 2009

Formal Concept Analysis

Radim BELOHLAVEK

Dept. Systems Science & Industrial Engineering
Watson School of Engineering and Applied Science
Binghamton University – SUNY

Introduction to Formal Concept Analysis (FCA)

Introduction to Formal Concept Analysis

– Formal Concept Analysis (FCA) = method of analysis of tabular data (Rudolf Wille, TU Darmstadt),

– alternatively called: concept data analysis, concept lattices, Galois lattices, ...

– used for data mining, knowledge discovery, preprocessing data

– **input**: objects (rows) $\times$ attributes (columns) table

	y_1	y_2	y_3
x_1	1	1	1
x_2	1	0	1
x_3	0	1	1
...		...	

or

	y_1	y_2	y_3
x_1	X	X	X
x_2	X		X
x_3		X	X
...		...	

or $\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}$

Introduction to Formal Concept Analysis

– **output**:

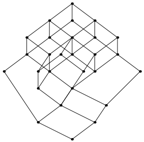
1. hierarchically ordered collection of clusters:
   – called concept lattice,
   – clusters are called formal concepts,
   – hierarchy = subconcept-superconcept,
2. data dependencies:
   – called attribute implications,
   – not all (would be redundant), only representative set

Output 1: Concept Lattices

input data:

	y_1	y_2	y_3
x_1	X	X	X
x_2	X		X
x_3		X	X

output concept lattice:



- concept lattice = hierarchically ordered set of clusters
- cluster (formal concept) = $\langle A, B \rangle$,
- A = collection of objects covered by cluster,
  B = collection of attributes covered by cluster,
- example of formal concept: $\langle \{x_1, x_2\}, \{y_1, y_3\} \rangle$,
- clusters = nodes in the Hasse diagram,
- Hasse diagram = represents partial order given by subconcept-superconcept hierarchy
- concept lattice = all potentially interesting concepts in data

Output 2: Attribute Implications

input data:

	y_1	y_2	y_3
x_1	X	X	X
x_2	X		X
x_3		X	X

attribute implications:

$A \Rightarrow B$ like
$\{y_2\} \Rightarrow \{y_3\}$, $\{y_1, y_2\} \Rightarrow \{y_3\}$,
but not $\{y_1\} \Rightarrow \{y_2\}$,

- attribute implication = particular data dependency,
- large number of attribute implications may be valid in given data,
- some of them redundant and thus not interesting ($\{y_2\} \Rightarrow \{y_2\}$),
- reasonably small non-redundant set of attribute dependencies (non-redundant basis),
- connections to other types of data dependencies (functional dependencies from relational databases, association rules).

History of FCA

- Port-Royal logic (traditional logic): formal notion of concept
  Arnauld A., Nicole P.: La logique ou l'art de penser, 1662 (Logic Or The Art Of Thinking, CUP, 2003):
  concept = extent (objects) + intent (attributes)
- G. Birkhoff (1940s): work on lattices and related mathematical structures, emphasizes applicational aspects of lattices in data analysis.
- Barbut M., Monjardet B.: Ordre et classification, algbre et combinatoire. Hachette, Paris, 1970.
- Wille R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In: I. Rival (Ed.): Ordered Sets. Reidel, Dordrecht, 1982, pp. 445–470.

Literature on FCA

books

- Ganter B., Wille R.: Formal Concept Analysis. Springer, 1999.
- Carpineto C., Romano G.: Concept Data Analysis. Wiley, 2004.

conferences

- ICFCA (Int. Conference of Formal Concept Analysis), Springer LNCS, `http://www.isima.fr/icfca07/`
- CLA (Concept Lattices and Their Applications), `http://www.lirmm.fr/cla07/index.htm`
- ICCS (Int. Conference on Conceptual Structures), Springer LNCS, `http://www.iccs.info/`
- conferences with focus on data analysis, information sciences, etc.

web

- keywords: formal concept analysis, concept lattice, attribute implication, concept data analysis, Galois lattice

Selected Applications of FCA

- clustering and classification (conceptual clustering),
- information retrieval, knowledge extraction (structured view on data, structured browsing),
- machine learning,
- software engineering
  - G. Snelting, F. Tip: Understanding class hierarchies using concept analysis. ACM Trans. Program. Lang. Syst. 22(3):540–582, May 2000.
  - U. Dekel, Y. Gill: Visualizing class interfaces with formal concept analysis. In OOPSLA'03, pp. 288–289, Anaheim, CA, October 2003.
- preprocessing method: e.g., Zaki M.: Mining non-redundant association rules. *Data Mining and Knowl. Disc.* **9**(2004), 223–248. closed frequent itemsets instead of frequent itemsets $\Rightarrow$ non-redundant association rules ($<<$ number)
- mathematics (new results in math. structures related to FCA)

State of the art of FCA

- Ganter, B., Stumme, G., Wille, R. (Eds.): Formal Concept Analysis Foundations and Applications. Springer, LNCS 3626, 2005,
- development of theoretical foundations,
- development of algorithms,
- applications: increasingly popular (information retrieval, software engineering, social networks, . . .),
- FCA as method of data preprocessing, interaction with other methods of data analysis,
- several software packages available.

Concept Lattices

What is a concept?

central notion in FCA = formal concept

but what is a concept? many approaches, including:

- psychology (approaches: classical, prototype, exemplar, knowledge)
  Murphy G. L.: The Big Book of Concepts. MIT Press, 2004.
  Margolis E., Laurence S.: Concepts: Core Readings. MIT Press, 1999.
- logic (rare, but Transparent Intensional Logic)
  Tichy P.: The Foundations of Frege's Logic. W. De Gryuter, 1988.
  Materna P.: Conceptual Systems. Logos Verlag, Berlin, 2004.
- artificial intelligence (frames, learning of concepts)
  Michalski, R. S., Bratko, I. and Kubat, M. (Eds.), Machine Learning and Data Mining: Methods and Applications, London, Wiley, 1998.
- conceptual graphs (Sowa)
  Sowa J. F.: Knowledge Representation: Logical, Philosophical, and Computational Foundations. Course Technology, 1999.
- "conceptual modeling", object-oriented paradigm, ...
- **traditional/Port-Royal logic**

Traditional (Port-Royal) view on concepts

The notion of a concept as used in FCA — inspired by Port-Royal logic (traditional logic):
Arnauld A., Nicole P.: La logique ou l'art de penser, 1662 (Logic Or The Art Of Thinking, CUP, 2003):

- **concept** (according to Port-Royal) **:=** **extent** + **intent**
  - **extent** = objects covered by concept
  - **intent** = attributes covered by concept
- **example: DOG** (extent = collection of all dogs (foxhound, poodle, ...), intent = {barks, has four limbs, has tail,...})
- **concept hierarchy**
  - subconcept/superconcept relation
  - DOG $\leq$ MAMMAL $\leq$ ANIMAL
  - **concept1=(extent1,intent1)** $\leq$ **concept2=(extent2,intent2)** $\Leftrightarrow$ extent1 $\subseteq$ extent2 ($\Leftrightarrow$ intent1 $\supseteq$ intent2)

Formal Contexts (Tables With Binary Attributes)

Definition (formal context (table with binary attributes))

A formal context is a triplet $\langle X, Y, I \rangle$ where X and Y are non-empty sets and I is a binary relation between X and Y, i.e., $I \subseteq X \times Y$.

- interpretation: X . . . set of objects, Y . . . set of attributes, $\langle x, y \rangle \in I$ . . . object x has attribute y
- formal context can be represented by table (table with binary attributes)
  $\langle x, y \rangle \in I$ . . . $\times$in table, $\langle x, y \rangle \notin I$ . . . blank in table,

I	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×		×	×
x_3		×	×	×
x_4		×	×	×
x_5	×			

Concept-forming Operators $^{\uparrow}$ and $^{\downarrow}$

Definition (concept-forming operators)

For a formal context $\langle X, Y, I\rangle$, operators $^{\uparrow}: 2^X \to 2^Y$ and $^{\downarrow}: 2^Y \to 2^X$ are defined for every $A \subseteq X$ and $B \subseteq Y$ by

$$
\begin{aligned}
A^{\uparrow} &= \{y \in Y \mid \text{for each } x \in A : \langle x, y\rangle \in I\}, \\
B^{\downarrow} &= \{x \in X \mid \text{for each } y \in B : \langle x, y\rangle \in I\}.
\end{aligned}
$$

- operator $^{\uparrow}$:
  assigns subsets of Y to subsets of X,
  $A^{\uparrow}$ ... set of all attributes shared by all objects from A,
- operator $^{\downarrow}$:
  assigns subsets of X to subsets of Y,
  $B^{\uparrow}$ ... set of all objects sharing all attributes from B.
- To emphasize that $^{\uparrow}$ and $^{\downarrow}$ are induced by $\langle X, Y, I\rangle$, we use $^{\uparrow_I}$ and $^{\downarrow_I}$.

Concept-forming Operators $^{\uparrow}$ and $^{\downarrow}$

Example (concept-forming operators)

For table

I	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×		×	×
x_3		×	×	×
x_4		×	×	×
x_5	×			

we have:

- $\{x_2\}^{\uparrow} = \{y_1, y_3, y_4\}$, $\{x_2, x_3\}^{\uparrow} = \{y_3, y_4\}$,
- $\{x_1, x_4, x_5\}^{\uparrow} = \emptyset$,
- $X^{\uparrow} = \emptyset$, $\emptyset^{\uparrow} = Y$,
- $\{y_1\}^{\downarrow} = \{x_1, x_2, x_5\}$, $\{y_1, y_2\}^{\downarrow} = \{x_1\}$,
- $\{y_2, y_3\}^{\downarrow} = \{x_1, x_3, x_4\}$, $\{y_2, y_3, y_4\}^{\downarrow} = \{x_1, x_3, x_4\}$,
- $\emptyset^{\downarrow} = X$, $Y^{\downarrow} = \{x_1\}$.

Formal Concepts

Definition (formal concept)

A formal concept in $\langle X, Y, I\rangle$ is a pair $\langle A, B\rangle$ of $A \subseteq X$ and $B \subseteq Y$ such that

$$A^{\uparrow} = B \text{ and } B^{\downarrow} = A.$$

- A ... extent of $\langle A, B\rangle$,
- B ... extent of $\langle A, B\rangle$,
- verbal description: $\langle A, B\rangle$ is a formal concept iff A contains just objects sharing all attributes from B and B contains just attributes shared by all objects from A,
- mathematical description: $\langle A, B\rangle$ is a formal concept iff $\langle A, B\rangle$ is a fixpoint of $\langle {}^{\uparrow}, {}^{\downarrow}\rangle$.

Formal Concepts

Example (formal concept)

For table

I	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×		×	×
x_3		×	×	×
x_4		×	×	×
x_5	×			

the highlighted rectangle represents formal concept $\langle A_1, B_1 \rangle = \langle \{x_1, x_2, x_3, x_4\}, \{y_3, y_4\} \rangle$ because

$$\{x_1, x_2, x_3, x_4\}^{\uparrow} = \{y_3, y_4\},$$

$$\{y_3, y_4\}^{\downarrow} = \{x_1, x_2, x_3, x_4\}.$$

Example (formal concept (cntd.))

But there are further formal concepts:

I	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×		×	×
x_3		×	×	×
x_4		×	×	×
x_5	×			

I	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×		×	×
x_3		×	×	×
x_4		×	×	×
x_5	×			

I	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×		×	×
x_3		×	×	×
x_4		×	×	×
x_5	×			

i.e., $\langle A_2, B_2\rangle = \langle\{x_1, x_3, x_4\}, \{y_2, y_3, y_4\}\rangle$,
$\langle A_3, B_3\rangle = \langle\{x_1, x_2\}, \{y_1, y_3, y_4\}\rangle$, $\langle A_4, B_4\rangle = \langle\{x_1, x_2, x_5\}, \{y_1\}\rangle$.

Subconcept-superconcept ordering

Definition (subconcept-superconcept ordering)

For formal concepts $\langle A_1, B_1\rangle$ and $\langle A_2, B_2\rangle$ of $\langle X, Y, I\rangle$, put

$$\langle A_1, B_1\rangle \leq \langle A_2, B_2\rangle \quad \text{iff} \quad A_1 \subseteq A_2 \text{ (iff } B_2 \subseteq B_1\text{)}.$$

- $\leq$ ... subconcept-superconcept ordering,
- $\langle A_1, B_1\rangle \leq \langle A_2, B_2\rangle$ ... $\langle A_1, B_1\rangle$ is more specific than $\langle A_2, B_2\rangle$ ($\langle A_2, B_2\rangle$ is more general),
- captures intuition behind DOG $\leq$ MAMMAL.

Example

Consider formal concepts from the previous example:
$\langle A_1, B_1\rangle = \langle\{x_1, x_2, x_3, x_4\}, \{y_3, y_4\}\rangle$, $\langle A_2, B_2\rangle = \langle\{x_1, x_3, x_4\}, \{y_2, y_3, y_4\}\rangle$, $\langle A_3, B_3\rangle = \langle\{x_1, x_2\}, \{y_1, y_3, y_4\}\rangle$, $\langle A_4, B_4\rangle = \langle\{x_1, x_2, x_5\}, \{y_1\}\rangle$. Then:
$\langle A_3, B_3\rangle \leq \langle A_1, B_1\rangle$, $\langle A_3, B_3\rangle \leq \langle A_2, B_2\rangle$, $\langle A_3, B_3\rangle \leq \langle A_4, B_4\rangle$,
$\langle A_2, B_2\rangle \leq \langle A_1, B_1\rangle$, $\langle A_1, B_1\rangle || \langle A_4, B_4\rangle$, $\langle A_2, B_2\rangle || \langle A_4, B_4\rangle$.

Concept Lattice

Definition (concept lattice)

Denote by $\mathcal{B}(X, Y, I)$ the collection of all formal concepts of $\langle X, Y, I\rangle$, i.e.

$$\mathcal{B}(X, Y, I) = \{\langle A, B\rangle \in 2^X \times 2^Y \mid A^{\uparrow} = B, B^{\downarrow} = A\}.$$

$\mathcal{B}(X, Y, I)$ equipped with the subconcept-superconcept ordering $\leq$ is called a concept lattice of $\langle X, Y, I\rangle$.

- $\mathcal{B}(X, Y, I)$ represents all (potentially interesting) clusters which are "hidden" in data $\langle X, Y, I\rangle$.
- We will see that $\langle \mathcal{B}(X, Y, I), \leq\rangle$ is indeed a lattice later.

Denote
$\mathrm{Ext}(X, Y, I) = \{A \in 2^X \mid \langle A, B\rangle \in \mathcal{B}(X, Y, I) \text{ for some } B\}$
(extents of concepts)
$\mathrm{Int}(X, Y, I) = \{B \in 2^Y \mid \langle A, B\rangle \in \mathcal{B}(X, Y, I) \text{ for some } A\}$
(intents of concepts)

Concept Lattice – Example

input data (Ganter, Wille: Formal Concept Analysis. Springer, 1999):

		a	b	c	d	e	f	g	h	i
leech	1	×	×					×		
bream	2	×	×					×	×	
frog	3	×	×	×				×	×	
dog	4	×		×				×	×	×
spike-weed	5	×	×		×		×			
reed	6	×	×	×	×		×			
bean	7	×		×	×	×				
maize	8	×		×	×		×			

a: needs water to live, b: lives in water,

c: lives on land, d: needs chlorophyll to produce food,

e: two seed leaves, f: one seed leaf,

g: can move around, h: has limbs,

i: suckles its offspring.

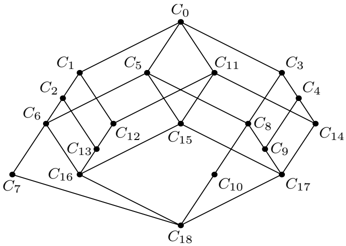
		a	b	c	d	e	f	g	h	i
leech	1	×	×					×		
bream	2	×	×					×	×	
frog	3	×	×	×				×	×	
dog	4	×		×				×	×	×
spike-weed	5	×	×		×		×			
reed	6	×	×	×	×		×			
bean	7	×		×	×	×				
maize	8	×		×	×		×			

formal concepts:

$C_0 = \langle\{1,2,3,4,5,6,7,8\},\{a\}\rangle,\ C_1 = \langle\{1,2,3,4\},\{a,g\}\rangle,$
$C_2 = \langle\{2,3,4\},\{a,g,h\}\rangle,\ C_3 = \langle\{5,6,7,8\},\{a,d\}\rangle,$
$C_4 = \langle\{5,6,8\},\{a,d,f\}\rangle,\ C_5 = \langle\{3,4,6,7,8\},\{a,c\}\rangle,$
$C_6 = \langle\{3,4\},\{a,c,g,h\}\rangle,\ C_7 = \langle\{4\},\{a,c,g,h,i\}\rangle,$
$C_8 = \langle\{6,7,8\},\{a,c,d\}\rangle,\ C_9 = \langle\{6,8\},\{a,c,d,f\}\rangle,$
$C_{10} = \langle\{7\},\{a,c,d,e\}\rangle,\ C_{11} = \langle\{1,2,3,5,6\},\{a,b\}\rangle,$
$C_{12} = \langle\{1,2,3\},\{a,b,g\}\rangle,\ C_{13} = \langle\{2,3\},\{a,b,g,h\}\rangle,$
$C_{14} = \langle\{5,6\},\{a,b,d,f\}\rangle,\ C_{15} = \langle\{3,6\},\{a,b,c\}\rangle,$
$C_{16} = \langle\{3\},\{a,b,c,g,h\}\rangle,\ C_{17} = \langle\{6\},\{a,b,c,d,f\}\rangle,$
$C_{18} = \langle\{\},\{a,b,c,d,e,f,g,h,i\}\rangle.$

		a	b	c	d	e	f	g	h	i
leech	1	×	×					×		
bream	2	×	×					×	×	
frog	3	×	×	×				×	×	
dog	4	×		×				×	×	×
spike-weed	5	×	×		×		×			
reed	6	×	×	×	×		×			
bean	7	×		×	×	×				
maize	8	×		×	×		×			

concept lattice:



$C_0 = \langle\{1,2,3,4,5,6,7,8\},\{a\}\rangle$, $C_1 = \langle\{1,2,3,4\},\{a,g\}\rangle$,
$C_2 = \langle\{2,3,4\},\{a,g,h\}\rangle$, $C_3 = \langle\{5,6,7,8\},\{a,d\}\rangle$,
$C_4 = \langle\{5,6,8\},\{a,d,f\}\rangle$, $C_5 = \langle\{3,4,6,7,8\},\{a,c\}\rangle$,
$C_6 = \langle\{3,4\},\{a,c,g,h\}\rangle$, $C_7 = \langle\{4\},\{a,c,g,h,i\}\rangle$,
$C_8 = \langle\{6,7,8\},\{a,c,d\}\rangle$, $C_9 = \langle\{6,8\},\{a,c,d,f\}\rangle$,
$C_{10} = \langle\{7\},\{a,c,d,e\}\rangle$, $C_{11} = \langle\{1,2,3,5,6\},\{a,b\}\rangle$,
$C_{12} = \langle\{1,2,3\},\{a,b,g\}\rangle$, $C_{13} = \langle\{2,3\},\{a,b,g,h\}\rangle$,
$C_{14} = \langle\{5,6\},\{a,b,d,f\}\rangle$, $C_{15} = \langle\{3,6\},\{a,b,c\}\rangle$,
$C_{16} = \langle\{3\},\{a,b,c,g,h\}\rangle$, $C_{17} = \langle\{6\},\{a,b,c,d,f\}\rangle$,
$C_{18} = \langle\{\},\{a,b,c,d,e,f,g,h,i\}\rangle$.

Formal concepts as maximal rectangles

I	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×		×	×
x_3		×	×	×
x_4		×	×	×
x_5	×			

Definition (rectangles in $\langle X, Y, I\rangle$)

A rectangle in $\langle X, Y, I\rangle$ is a pair $\langle A, B\rangle$ such that $A \times B \subseteq I$, i.e.: for each $x \in A$ and $y \in B$ we have $\langle x, y\rangle \in I$. For rectangles $\langle A_1, B_1\rangle$ and $\langle A_2, B_2\rangle$, put $\langle A_1, B_1\rangle \sqsubseteq \langle A_2, B_2\rangle$ iff $A_1 \subseteq A_2$ and $B_1 \subseteq B_2$.

Example

In the table above, $\langle\{x_1, x_2, x_3\}, \{y_3, y_4\}\rangle$ is a rectangle which is not maximal w.r.t. $\sqsubseteq$. $\langle\{x_1, x_2, x_3, x_4\}, \{y_3, y_4\}\rangle$ is a rectangle which is maximal w.r.t. $\sqsubseteq$.

Formal concepts as maximal rectangles

Theorem (formal concepts as maximal rectangles)

$\langle A, B\rangle$ is a formal concept of $\langle X, Y, I\rangle$ iff $\langle A, B\rangle$ is a maximal rectangle in $\langle X, Y, I\rangle$.

Proof.

"$\Rightarrow$":

"$\Leftarrow$":

$\square$

"Geometrical reasoning" in FCA based on rectangles is important.

Mathematical structures related to FCA

- Galois connections,
- closure operators,
- fixed points of Galois connections and closure operators.

These structure are referred to as closure structures.

Galois connections

Definition (Galois connection)

A Galois connection between sets X and Y is a pair $\langle f, g\rangle$ of $f : 2^X \to 2^Y$ and $g : 2^Y \to 2^X$ satisfying for $A, A_1, A_2 \subseteq X$, $B, B_1, B_2 \subseteq Y$:

$$A_1 \subseteq A_2 \Rightarrow f(A_2) \subseteq f(A_1), \tag{1}$$
$$B_1 \subseteq B_2 \Rightarrow g(B_2) \subseteq g(B_1), \tag{2}$$
$$A \subseteq g(f(A)), \tag{3}$$
$$B \subseteq f(g(B)). \tag{4}$$

Definition (fixpoints of Galois connections)

For a Galois connection $\langle f, g\rangle$ between sets X and Y, the set

$$\mathrm{fix}(\langle f, g\rangle) = \{\langle A, B\rangle \in 2^X \times 2^Y \mid f(A) = B, g(B) = A\}$$

is called a set of fixpoints of $\langle f, g\rangle$.

Galois connections

Theorem (arrow operators form a Galois connection)

For a formal context $\langle X, Y, I\rangle$, the pair $\langle {}^{\uparrow_I}, {}^{\downarrow_I}\rangle$ of operators induced by $\langle X, Y, I\rangle$ is a Galois connection between X and Y.

Proof.

□

Lemma (chaining of Galois connection)

For a Galois connection $\langle f, g\rangle$ between X and Y we have
$f(A) = f(g(f(A)))$ *and* $g(B) = g(f(g(B)))$ *for any* $A \subseteq X$ *and* $B \subseteq Y$.

Proof.

We prove only $f(A) = f(g(f(A)))$, $g(B) = g(f(g(B)))$ is dual:
"$\subseteq$":
$f(A) \subseteq f(g(f(A)))$ follows from (4) by putting $B = f(A)$.
"$\supseteq$":
Since $A \subseteq g(f(A))$ by (3), we get $f(A) \supseteq f(g(f(A)))$ by application of (1). □

Closure operators

Definition (closure operator)

A closure operator on a set X is a mapping $C : 2^X \to 2^X$ satisfying for each $A, A_1, A_2 \subseteq X$

$$A \subseteq C(A), \quad (5)$$

$$A_1 \subseteq A_2 \Rightarrow C(A_1) \subseteq C(A_2), \quad (6)$$

$$C(A) = C(C(A)). \quad (7)$$

Definition (fixpoints of closure operators)

For a closure operator $C : 2^X \to 2^X$, the set

$$\mathrm{fix}(C) = \{A \subseteq X \mid C(A) = A\}$$

is called a set of fixpoints of C.

Closure operators

Theorem (from Galois connection to closure operators)

If $\langle f, g\rangle$ is a Galois connection between X and Y then $C_X = f \circ g$ is a closure operator on X and $C_Y = g \circ f$ is a closure operator on Y.

Proof.

We show that $f \circ g : 2^X \to 2^X$ is a closure operator on X:
(5) is $A \subseteq g(f(A))$ which is true by definition of a Galois connection.
(6): $A_1 \subseteq A_2$ impies $f(A_2) \subseteq f(A_1)$ which implies $g(f(A_1)) \subseteq g(f(A_2))$.
(7): Since $f(A) = f(g(f(A)))$, we get $g(f(A)) = g(f(g(f(A))))$. □

Theorem (extents and intents)

$$\begin{aligned}\mathrm{Ext}(X, Y, I) &= \{B^{\downarrow} \mid B \subseteq Y\},\\ \mathrm{Int}(X, Y, I) &= \{A^{\uparrow} \mid A \subseteq X\}.\end{aligned}$$

Proof.

We prove only the part for $\mathrm{Ext}(X, Y, I)$, part for $\mathrm{Int}(X, Y, I)$ is dual.
"$\subseteq$": If $A \in \mathrm{Ext}(X, Y, I)$, then $\langle A, B\rangle$ is a formal concept for some $B \subseteq Y$. By definition, $A = B^{\downarrow}$, i.e. $A \in \{B^{\downarrow} \mid B \subseteq Y\}$.
"$\supseteq$": Let $A \in \{B^{\downarrow} \mid B \subseteq Y\}$, i.e. $A = B^{\downarrow}$ for some B. Then $\langle A, A^{\uparrow}\rangle$ is a formal concept. Namely, $A^{\uparrow\downarrow} = B^{\downarrow\uparrow\downarrow} = B^{\downarrow} = A$ by chaining, and $A^{\uparrow} = A^{\uparrow}$ for free. That is, A is the extent of a formal concept $\langle A, A^{\uparrow}\rangle$, whence $A \in \mathrm{Ext}(X, Y, I)$. □

Theorem (least extent containing A, least intent containing B)

The least extent containing $A \subseteq X$ is $A^{\uparrow\downarrow}$. The least intent containing $B \subseteq Y$ is $B^{\downarrow\uparrow}$.

Proof.

For extents:

1. $A^{\uparrow\downarrow}$ is an extent (by previous theorem).
2. If C is an extent such that $A \subseteq C$, then $A^{\uparrow\downarrow} \subseteq C^{\uparrow\downarrow}$ because $^{\uparrow\downarrow}$ is a closure operator. Therefore, $A^{\uparrow\downarrow}$ is the least extent containing A. □

Extents, intents, concept lattice

Theorem

For any formal context $\langle X, Y, I\rangle$:

$$\begin{aligned}
\mathrm{Ext}(X,Y,I) &= \mathrm{fix}({}^{\uparrow\downarrow}),\\
\mathrm{Int}(X,Y,I) &= \mathrm{fix}({}^{\downarrow\uparrow}),\\
\mathcal{B}(X,Y,I) &= \{\langle A, A^{\uparrow}\rangle \mid A \in \mathrm{Ext}(X,Y,I)\},\\
\mathcal{B}(X,Y,I) &= \{\langle B^{\downarrow}, B\rangle \mid B \in \mathrm{Int}(X,Y,I)\}.
\end{aligned}$$

Proof.

For $\mathrm{Ext}(X,Y,I)$:
We need to show that A is an extent iff $A = A^{\uparrow\downarrow}$.
"$\Rightarrow$": If A is an extent then for the corresponding formal concept $\langle A, B\rangle$ we have $B = A^{\uparrow}$ and $A = B^{\downarrow} = A^{\uparrow\downarrow}$. Hence, $A = A^{\uparrow\downarrow}$.
"$\Leftarrow$": If $A = A^{\uparrow\downarrow}$ then $\langle A, A^{\uparrow}\rangle$ is a formal concept. Namely, denoting $\langle A, B\rangle = \langle A, A^{\uparrow}\rangle$, we have both $A^{\uparrow} = B$ and $B^{\downarrow} = A^{\uparrow\downarrow} = A$. Therefore, A is an extent.

Extents, intents, concept lattice

cntd.

For $\mathcal{B}(X, Y, I) = \{\langle A, A^{\uparrow}\rangle \mid A \in \mathrm{Ext}(X, Y, I)\}$:
If $\langle A, B\rangle \in \mathcal{B}(X, Y, I)$ then $B = A^{\uparrow}$ and, obviously, $A \in \mathrm{Ext}(X, Y, I)$.
If $A \in \mathrm{Ext}(X, Y, I)$ then $A = A^{\uparrow\downarrow}$ (above claim) and, therefore,
$\langle A, A^{\uparrow}\rangle \in \mathcal{B}(X, Y, I)$. □

remark

The previous theorem says:
In order to obtain $\mathcal{B}(X, Y, I)$, we can:

1. compute $\mathrm{Ext}(X, Y, I)$,
2. for each $A \in \mathrm{Ext}(X, Y, I)$, output $\langle A, A^{\uparrow}\rangle$.

Concise definition of Galois connections

There is a single condition which is equivalent to conditions (1)–(4) from definition of Galois connection:

Theorem

$\langle f, g\rangle$ is a Galois connection between X and Y iff for every $A \subseteq X$ and $B \subseteq Y$:

$$A \subseteq g(B) \quad \textit{iff} \quad B \subseteq f(A) \tag{8}$$

Proof.

"$\Rightarrow$":

Let $\langle f, g\rangle$ be a Galois connection.

If $A \subseteq g(B)$ then $f(g(B)) \subseteq f(A)$ and since $B \subseteq f(g(B))$, we get $B \subseteq f(A)$. In similar way, $B \subseteq f(A)$ implies $A \subseteq g(B)$.

□

Concise definition of Galois connections

cntd.

"$\Leftarrow$":

Let $A \subseteq g(B)$ iff $B \subseteq f(A)$. We check that $\langle f, g \rangle$ is a Galois connection. Due to duality, it suffices to check (a) $A \subseteq g(f(A))$, and (b) $A_1 \subseteq A_2$ implies $f(A_2) \subseteq f(A_1)$.

(a): Due to our assumption, $A \subseteq g(f(A))$ is equivalent to $f(A) \subseteq f(A)$ which is evidently true.

(b): Let $A_1 \subseteq A_2$. Due to (a), we have $A_2 \subseteq g(f(A_2))$, therefore $A_1 \subseteq g(f(A_2))$. Using assumption, the latter is equivalent to $f(A_2) \subseteq f(A_1)$. □

Galois connections, and union and intersection

Theorem

$\langle f, g\rangle$ is a Galois connection between X and Y then for $A_j \subseteq X$, $j \in J$, and $B_j \subseteq Y$, $j \in J$ we have

$$f(\bigcup_{j\in J} A_j) = \bigcap_{j\in J} f(A_j), \quad (9)$$

$$g(\bigcup_{j\in J} B_j) = \bigcap_{j\in J} g(B_j). \quad (10)$$

Proof.

(9):
For any $D \subseteq Y$: $D \subseteq f(\bigcup_{j\in J} A_j)$ iff $\bigcup_{j\in J} A_j \subseteq g(D)$ iff for each $j \in J$: $A_j \subseteq g(D)$ iff for each $j \in J$: $D \subseteq f(A_j)$ iff $D \subseteq \bigcap_{j\in J} f(A_j)$.
Since D is arbitrary, it follows that $f(\bigcup_{j\in J} A_j) = \bigcap_{j\in J} f(A_j)$.
(10): dual. □

Each Galois connection is induced by a binary relation

Theorem

Let $\langle f, g\rangle$ be a Galois connection between X and Y. Consider a formal context $\langle X, Y, I\rangle$ such that I is defined by

$$\langle x, y\rangle \in I \quad \text{iff} \quad y \in f(\{x\}) \qquad \text{or, equivalently, iff } x \in g(\{y\}), \tag{11}$$

for each $x \in X$ and $y \in Y$. Then $\langle {}^{\uparrow_I}, {}^{\downarrow_I}\rangle = \langle f, g\rangle$, i.e., the arrow operators $\langle {}^{\uparrow_I}, {}^{\downarrow_I}\rangle$ induced by $\langle X, Y, I\rangle$ coincide with $\langle f, g\rangle$.

Proof.

First, we show $y \in f(\{x\})$ iff $x \in g(\{y\})$:
From $y \in f(\{x\})$ we get $\{y\} \subseteq f(\{x\})$ from which, using (8), we get $\{x\} \subseteq g(\{y\})$, i.e. $x \in g(\{y\})$.
In a similar way, $x \in g(\{y\})$ implies $y \in f(\{x\})$. This establishes $y \in f(\{x\})$ iff $x \in g(\{y\})$.

Each Galois connection is induced by a binary relation

cntd.

Now, using (9), for each $A \subseteq X$ we have

$$\begin{aligned} f(A) &= f(\cup_{x\in A}\{x\}) = \cap_{x\in A} f(\{x\}) = \\ &= \cap_{x\in A}\{y \in Y \mid y \in f(\{x\})\} = \cap_{x\in A}\{y \in Y \mid \langle x, y\rangle \in I\} = \\ &= \{y \in Y \mid \text{ for each } x \in A : \langle x, y\rangle \in I\} = A^{\uparrow_I}. \end{aligned}$$

Dually, for $B \subseteq Y$ we get $g(B) = B^{\downarrow_I}$. □

remarks

- Relation I induced from $\langle f, g\rangle$ by (11) will be denoted by $I_{\langle f,g\rangle}$.
- Therefore, we have established two mappings:
  $I \mapsto \langle {}^{\uparrow_I}, {}^{\downarrow_I}\rangle$ assigns a Galois connection to a binary relation I.
  $\langle {}^{\uparrow}, {}^{\downarrow}\rangle \mapsto I_{\langle \uparrow,\downarrow\rangle}$ assigns a binary relation to a Galois connection.

Representation theorem for Galois connections

Theorem (representation theorem)

$I \mapsto \langle {}^{\uparrow_I}, {}^{\downarrow_I} \rangle$ and $\langle {}^{\uparrow}, {}^{\downarrow} \rangle \mapsto I_{\langle \uparrow, \downarrow \rangle}$ are mutually inverse mappings between the set of all binary relations between X and Y and the set of all Galois connections between X and Y.

Proof.

Using the results established above, it remains to check that $I = I_{\langle \uparrow_I, \downarrow_I \rangle}$: We have

$$\langle x, y \rangle \in I_{\langle \uparrow_I, \downarrow_I \rangle} \text{ iff } y \in \{x\}^{\uparrow_I} \text{ iff } \langle x, y \rangle \in I,$$

finishing the proof. □

remarks

In particular, previous theorem assures that (1)–(4) fully describe all the properties of our arrow operators induced by data $\langle X, Y, I \rangle$.

Duality between extents and intents

Having established properties of $\langle ^{\uparrow}, ^{\downarrow} \rangle$, we can see the duality relationship between extents and intents:

Theorem

For $\langle A_1, B_1 \rangle, \langle A_2, B_2 \rangle \in \mathcal{B}(X, Y, I)$,

$$A_1 \subseteq A_2 \quad \textit{iff} \quad B_2 \subseteq B_1. \qquad (12)$$

Proof.

By assumption, $A_i = B_i^{\downarrow}$ and $B_i = A_i^{\uparrow}$. Therefore, using (1) and (2), we get $A_1 \subseteq A_2$ implies $A_2^{\uparrow} \subseteq A_1^{\uparrow}$, i.e., $B_2 \subseteq B_1$, which implies $B_1^{\downarrow} \subseteq B_2^{\downarrow}$, i.e. $A_1 \subseteq A_2$. □

Therefore, the definition of a partial order $\leq$ on $\mathcal{B}(X, Y, I)$ is correct.

Duality between extents and intents

Theorem (extents, intents, and formal concepts)

1. $\langle \mathrm{Ext}(X, Y, I), \subseteq \rangle$ *and* $\langle \mathrm{Int}(X, Y, I), \subseteq \rangle$ *are partially ordered sets.*
2. $\langle \mathrm{Ext}(X, Y, I), \subseteq \rangle$ *and* $\langle \mathrm{Int}(X, Y, I), \subseteq \rangle$ *are dually isomorphic, i.e., there is a mapping* $f : \mathrm{Ext}(X, Y, I) \to \mathrm{Int}(X, Y, I)$ *satisfying* $A_1 \subseteq A_2$ *iff* $f(A_2) \subseteq f(A_1)$.
3. $\langle \mathcal{B}(X, Y, I), \leq \rangle$ *is isomorphic to* $\langle \mathrm{Ext}(X, Y, I), \subseteq \rangle$.
4. $\langle \mathcal{B}(X, Y, I), \leq \rangle$ *is dually isomorphic to* $\langle \mathrm{Int}(X, Y, I), \subseteq \rangle$.

Proof.

1.: Obvious because $\mathrm{Ext}(X, Y, I)$ is a collection of subsets of X and $\subseteq$ is set inclusion. Same for $\mathrm{Int}(X, Y, I)$.
2.: Just take $f = {}^{\uparrow}$ and use previous results.
3.: Obviously, mapping $\langle A, B \rangle \mapsto A$ is the required isomorphism.
4.: Mapping $\langle A, B \rangle \mapsto B$ is the required dual isomorphism. □

Hierarchical structure of concept lattices

We know that $\mathcal{B}(X, Y, I)$ (set of all formal concepts) equipped with $\leq$ (subconcept-superconcept hierarchy) is a partially ordered set. Now, the question is:

What is the structure of $\langle \mathcal{B}(X, Y, I), \leq \rangle$?

It turns out that $\langle \mathcal{B}(X, Y, I), \leq \rangle$ is a complete lattice (we will see this as a part of Main theorem of concept lattices).

concept lattice $\approx$ complete conceptual hierarchy

The fact that $\langle \mathcal{B}(X, Y, I), \leq \rangle$ is a lattice is a "welcome property". Namely, it says that for any collection $K \subseteq \mathcal{B}(X, Y, I)$ of formal concepts, $\mathcal{B}(X, Y, I)$ contains both the "direct generalization" $\bigvee K$ of concepts from K (supremum of K), and the "direct specialization" $\bigvee K$ of concepts from K (infimum of K). In this sense, $\langle \mathcal{B}(X, Y, I), \leq \rangle$ is a complete conceptual hierarchy.

Now: details to Main theorem of concept lattices.

Theorem (system of fixpoints of closure operators)

For a closure operator C on X, the partially ordered set $\langle \mathrm{fix}(C), \subseteq \rangle$ of fixpoints of C is a complete lattice with infima and suprema given by

$$\bigwedge_{j\in J} A_j = \bigcap_{j\in J} A_j, \tag{13}$$

$$\bigvee_{j\in J} A_j = C(\bigcup_{j\in J} A_j). \tag{14}$$

Proof.

Evidently, $\langle \mathrm{fix}(C), \subseteq \rangle$ is a partially ordered set.
(13): First, we check that for $A_j \in \mathrm{fix}(C)$ we have $\bigcap_{j\in J} A_j \in \mathrm{fix}(C)$ (intersection of fixpoints is a fixpoint). We need to check $\bigcap_{j\in J} A_j = C(\bigcap_{j\in J} A_j)$.
"$\subseteq$": $\bigcap_{j\in J} A_j \subseteq C(\bigcap_{j\in J} A_j)$ is obvious (property of closure operators).
"$\supseteq$": We have $C(\bigcap_{j\in J} A_j) \subseteq \bigcap_{j\in J} A_j$ iff for each $j \in J$ we have $C(\bigcap_{j\in J} A_j) \subseteq A_j$ which is true. Indeed, we have $\bigcap_{j\in J} A_j \subseteq A_j$ from which we get $C(\bigcap_{j\in J} A_j) \subseteq C(A_j) = A_j$.

cntd.

Now, since $\bigcap_{j\in J} A_j \in \operatorname{fix}(C)$, it is clear that $\bigcap_{j\in J} A_j$ is the infimum of A_j's: first, $\bigcap_{j\in J} A_j$ is less of equal to every A_j; second, $\bigcap_{j\in J} A_j$ is greater or equal to any $A \in \operatorname{fix}(C)$ which is less or equal to all A_j's; that is, $\bigcap_{j\in J} A_j$ is the greatest element of the lower cone of $\{A_j \mid j \in J\}$).

(14): We verify $\bigvee_{j\in J} A_j = C(\bigcup_{j\in J} A_j)$. Note first that since $\bigvee_{j\in J} A_j$ is a fixpoint of C, we have $\bigvee_{j\in J} A_j = C(\bigvee_{j\in J} A_j)$.
"$\subseteq$": $C(\bigcup_{j\in J} A_j)$ is a fixpoint which is greater or equal to every A_j, and so $C(\bigcup_{j\in J} A_j)$ must be greater or equal to the supremum $\bigvee_{j\in J} A_j$, i.e. $\bigvee_{j\in J} A_j \subseteq C(\bigcup_{j\in J} A_j)$.
"$\supseteq$": Since $\bigvee_{j\in J} A_j \supseteq A_j$ for any $j \in J$, we get $\bigvee_{j\in J} A_j \supseteq \bigcup_{j\in J} A_j$, and so $\bigvee_{j\in J} A_j = C(\bigvee_{j\in J} A_j) \supseteq C(\bigcup_{j\in J} A_j)$.

To sum up, $\bigvee_{j\in J} A_j = C(\bigcup_{j\in J} A_j)$. □

Theorem (Main theorem of concept lattices, Wille (1982))

(1) $\mathcal{B}(X, Y, I)$ is a complete lattice with infima and suprema given by

$$\bigwedge_{j \in J} \langle A_j, B_j \rangle = \langle \bigcap_{j \in J} A_j, (\bigcup_{j \in J} B_j)^{\downarrow\uparrow} \rangle \ , \bigvee_{j \in J} \langle A_j, B_j \rangle = \langle (\bigcup_{j \in J} A_j)^{\uparrow\downarrow}, \bigcap_{j \in J} B_j \rangle \ . \quad (15)$$

(2) Moreover, an arbitrary complete lattice $\mathbf{V} = (V, \leq)$ is isomorphic to $\mathcal{B}(X, Y, I)$ iff there are mappings $\gamma : X \to V$, $\mu : Y \to V$ such that

(i) $\gamma(X)$ *is* $\bigvee$*-dense in* V*,* $\mu(Y)$ *is* $\bigwedge$*-dense in* V*;*

(ii) $\gamma(x) \leq \mu(y)$ *iff* $\langle x, y \rangle \in I$*.*

remark

(1) $K \subseteq V$ is supremally dense in V iff for each $v \in V$ there exists $K' \subseteq K$ such that $v = \bigvee K'$ (i.e., every element v of V is a supremum of some elements of K).
Dually for infimal density of K in V (every element v of V is an infimum of some elements of K).

(2) Supremally (infimally) dense sets canbe considered building blocks of V

Proof.

Proof for (1) only. We check $\bigwedge_{j\in J}\langle A_j, B_j\rangle = \langle\bigcap_{j\in J} A_j, (\bigcup_{j\in J} B_j)^{\downarrow\uparrow}\rangle$:
First, $\langle \mathrm{Ext}(X,Y,I),\subseteq\rangle = \langle \mathrm{fix}(^{\uparrow\downarrow}),\subseteq\rangle$ and $\langle \mathrm{Int}(X,Y,I),\subseteq\rangle = \langle \mathrm{fix}(^{\downarrow\uparrow}),\subseteq\rangle$.
That is, $\mathrm{Ext}(X,Y,I)$ and $\mathrm{Int}(X,Y,I)$ are systems of fixpoints of closure operators, and therefore, suprema and infima in $\mathrm{Ext}(X,Y,I)$ and $\mathrm{Int}(X,Y,I)$ obey the formulas from previous theorem.

Second, recall that $\langle \mathcal{B}(X,Y,I),\leq\rangle$ is isomorphic to $\langle \mathrm{Ext}(X,Y,I),\subseteq\rangle$ and dually isomorphic to $\langle \mathrm{Int}(X,Y,I),\subseteq\rangle$.
Therefore, infima in $\mathcal{B}(X,Y,I)$ correspond to infima in $\mathrm{Ext}(X,Y,I)$ and to suprema in $\mathrm{Int}(X,Y,I)$.
That is, since $\bigwedge_{j\in J}\langle A_j, B_j\rangle$ is the infimum of $\langle A_j, B_j\rangle$'s in $\langle \mathcal{B}(X,Y,I),\leq\rangle$:
The extent of $\bigwedge_{j\in J}\langle A_j, B_j\rangle$ is the infimum of A_j's in $\langle \mathrm{Ext}(X,Y,I),\subseteq\rangle$ which is, according to (13), $\bigcap_{j\in J} A_j$. The intent of $\bigwedge_{j\in J}\langle A_j, B_j\rangle$ is the supremum of B_j's in $\langle \mathrm{Int}(X,Y,I),\subseteq\rangle$ which is, according to (14), $(\bigcup_{j\in J} B_j)^{\downarrow\uparrow}$. We just proved

$$\bigwedge_{j\in J}\langle A_j, B_j\rangle = \langle\bigcap_{j\in J} A_j, (\bigcup_{j\in J} B_j)^{\downarrow\uparrow}\rangle.$$

Checking the formula for $\bigvee_{j\in J}\langle A_j, B_j\rangle$ is dual. □

γ and μ in part (2) of Main theorem

Consider part (2) and take $V := \mathcal{B}(X, Y, I)$. Since $\mathcal{B}(X, Y, I)$ is isomorphic to $\mathcal{B}(X, Y, I)$, there exist mappings

$$\gamma : X \to \mathcal{B}(X, Y, I) \text{ and } \mu : Y \to \mathcal{B}(X, Y, I)$$

satisfying properties from part (2). How do mappings γ and μ work?

$$\begin{aligned} \gamma(x) &= \langle \{x\}^{\uparrow\downarrow}, \{x\}^{\uparrow} \rangle \ldots \text{object concept of } x, \\ \mu(y) &= \langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle \ldots \text{attribute concept of } y. \end{aligned}$$

Then: (i) says that each $\langle A, B \rangle \in \mathcal{B}(X, Y, I)$ is a supremum of some objects concepts (and, infimum of some attribute concepts). This is true since

$$\langle A, B \rangle = \bigvee_{x \in A} \langle \{x\}^{\uparrow\downarrow}, \{x\}^{\uparrow} \rangle \text{ and } \langle A, B \rangle = \bigwedge_{y \in B} \langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle.$$

(ii) is true, too: $\gamma(x) \leq \mu(y)$ iff $\{x\}^{\uparrow\downarrow} \subseteq \{y\}^{\downarrow}$ iff $\{y\} \subseteq \{x\}^{\uparrow\downarrow\uparrow} = \{x\}^{\uparrow}$ iff $\langle x, y \rangle \in I$.

What does Main theorem say?

Part (1): $\mathcal{B}(X, Y, I)$ is a lattice + description of infima and suprema.
Part (2): way to label a concept lattice so that no information is lost.

labeling of Hasse diagrams of concept lattices

$\gamma(x) = \langle \{x\}^{\uparrow\downarrow}, \{x\}^{\uparrow} \rangle$ ... object concept of x –labeled by x,
$\mu(y) = \langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle$ ... attribute concept of y – labeled by y.

How do we see extents and intents in a labeled Hasse diagram?

extents and intents in labeled Hasse diagram

Consider formal concept $\langle A, B \rangle$ corresponding to node c of a labeled diagram of concept lattice $\mathcal{B}(X, Y, I)$. What is then extent and the intent of $\langle A, B \rangle$?

$x \in A$ iff node with label x lies on a path going from c downwards,

$y \in B$ iff node with label y lies on a path going from c upwards.

Labeling of diagrams of concept lattices

Example

(1) Draw a labeled Hasse diagram of concept lattice associated to formal context

I	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×		×	×
x_3		×	×	×
x_4		×	×	×
x_5	×			

(2) Is every formal concept either an object concept or an attribute concept? Can a formal concept be both an object concept and an attribute concept?

Exercise

Label the Hasse diagram from the organisms vs. their properties example.

Labeling of diagrams of concept lattices

Example

Draw a labeled Hasse diagram of concept lattice associated to formal context

I	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×		×	×
x_3		×	×	×
x_4		×	×	×
x_5	×			

$\mathcal{B}(X, Y, I)$ consists of: $\langle\{x_1\}, Y\rangle$, $\langle\{x_1, x_2\}, \{y_1, y_3, y_4\}\rangle$, $\langle\{x_1, x_3, x_4\}, \{y_2, y_3, y_4\}\rangle$, $\langle\{x_1, x_2, x_3, x_4\}, \{y_3, y_4\}\rangle$, $\langle\{x_1, x_2, x_5\}, \{y_1\}\rangle$, $\langle X, \emptyset\rangle$.

Clarified and reduced formal contexts

Definition (clarified context)

A formal context $\langle X, Y, I\rangle$ is called clarified if the corresponding table does neither contain identical rows nor identical columns.

That is, if $\langle X, Y, I\rangle$ is clarified then
$\{x_1\}^{\uparrow} = \{x_2\}^{\uparrow}$ implies $x_1 = x_2$ for every $x_1, x_2 \in X$;
$\{y_1\}^{\downarrow} = \{y_2\}^{\downarrow}$ implies $y_1 = y_2$ for every $y_1, y_2 \in Y$.
clarification: removal of identical rows and columns (only one of several identical rows/columns is left)

Example

The formal context on the right results by clarification from the formal context on the left.

I	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×		×	×
x_3		×	×	×
x_4		×	×	×
x_5	×			

I	y_1	y_2	y_3
x_1	×	×	×
x_2	×		×
x_3		×	×
x_5	×		

Clarified and reduced formal contexts

Theorem

If $\langle X_1, Y_1, I_1\rangle$ *is a clarified context resulting from* $\langle X_2, Y_2, I_2\rangle$ *by clarification, then* $\mathcal{B}(X_1, Y_1, I_1)$ *is isomorphic to* $\mathcal{B}(X_2, Y_2, I_2)$*.*

Proof.

Let $\langle X_2, Y_2, I_2\rangle$ contain x_1, x_2 s.t. $\{x_1\}^{\uparrow} = \{x_2\}^{\uparrow}$ (identical rows). Let $\langle X_1, Y_1, I_1\rangle$ result from $\langle X_2, Y_2, I_2\rangle$ by removing x_2 (i.e., $X_1 = X_2 - \{x_2\}$, $Y_1 = Y_2$). An isomorphism $f : \mathcal{B}(X_1, Y_1, I_1) \to \mathcal{B}(X_2, Y_2, I_2)$ is given by

$$f(\langle A_1, B_1\rangle) = \langle A_2, B_2\rangle$$

where $B_1 = B_2$ and

$$A_2 = \begin{cases} A_1 & \text{if } x_1 \notin A_1, \\ A_1 \cup \{x_2\} & \text{if } x_1 \in A_1. \end{cases}$$

□

Clarified and reduced formal contexts

cntd.

Namely, one can easily see that $\langle A_1, B_1\rangle$ is a formal concept of $\mathcal{B}(X_1, Y_1, I_1)$ iff $f(\langle A_1, B_1\rangle)$ is a formal concept of $\mathcal{B}(X_2, Y_2, I_2)$ and that for formal concepts $\langle A_1, B_1\rangle, \langle C_1, D_1\rangle$ of $\mathcal{B}(X_1, Y_1, I_1)$ we have

$$\langle A_1, B_1\rangle \leq \langle C_1, D_1\rangle \text{ iff } f(\langle A_1, B_1\rangle) \leq f(\langle C_1, D_1\rangle).$$

Therefore, $\mathcal{B}(X_1, Y_1, I_1)$ is isomorphic to $\mathcal{B}(X_2, Y_2, I_2)$. This justifies the claim for removing one (identical) row. The same is true for removing one column. Repeated application gives the theorem. □

Example

Find the isomorphism between concept lattices of formal contexts from the previous example.

Clarified and reduced formal contexts

Another way to simplify the input formal context: removing reducible objects and attributes

Example

Draw concept lattices of the following formal contexts:

I	y_1	y_2	y_3
x_1			×
x_2	×	×	×
x_3	×		

I	y_1	y_3
x_1		×
x_2	×	×
x_3	×	

Why are they isomorphic?

Hint: $y_2 =$ intersection of y_1 and y_3 (i.e., $\{y_2\}^{\downarrow} = \{y_1\}^{\downarrow} \cap \{y_3\}^{\downarrow}$).

Clarified and reduced formal contexts

Definition (reducible objects and attributes)

For a formal context $\langle X, Y, I\rangle$, an attribute $y \in Y$ is called reducible iff there is $Y' \subset Y$ with $y \notin Y'$ such that

$$\{y\}^{\downarrow} = \bigcap_{z \in Y'} \{z\}^{\downarrow},$$

i.e., the column corresponding to y is the intersection of columns corresponding to zs from Y'. An object $x \in X$ is called reducible iff there is $X' \subset X$ with $x \notin X'$ such that

$$\{x\}^{\uparrow} = \bigcap_{z \in X'} \{z\}^{\uparrow},$$

i.e., the row corresponding to x is the intersection of rows corresponding to zs from X'.

Clarified and reduced formal contexts

- y_2 from the previous example is reducible ($Y' = \{y_1, y_3\}$).
- Analogy: If a (real-valued attribute) y is a linear combination of other attributes, it can be removed (caution: this depends on what we do with the attributes). Intersection = particular attribute combination.
- (Non-)reducibility in $\langle X, Y, I \rangle$ is connected to so-called $\bigwedge$-(ir)reducibility and $\bigvee$-(ir)reducibility in $\mathcal{B}(X, Y, I)$.
- In a complete lattice $\langle V, \leq \rangle$, $v \in V$ is called $\bigwedge$-irreducible if there is no $U \subset V$ with $v \notin U$ s.t. $v = \bigwedge U$. Dually for $\bigvee$-irreducibility.
- Determine all $\bigwedge$-irreducible elements in $\langle 2^{\{a,b,c\}}, \subseteq \rangle$, in a "pentagon", and in a 4-element chain.
- Verify that in a finite lattice $\langle V, \leq \rangle$: v is $\bigwedge$-irreducible iff v is covered by exactly one element of V; v is $\bigvee$-irreducible iff v covers exactly one element of V.

Clarified and reduced formal contexts

- easily from definition: y is reducible iff there is $Y' \subset Y$ with $y \notin Y'$ s.t.

$$\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle = \bigwedge_{z \in Y'} \langle \{z\}^{\downarrow}, \{z\}^{\downarrow\uparrow} \rangle. \qquad (16)$$

- Let $\langle X, Y, I \rangle$ be clarified. Then in (16), for each $z \in Y'$: $\{y\}^{\downarrow} \neq \{z\}^{\downarrow}$, and so, $\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle \neq \langle \{z\}^{\downarrow}, \{z\}^{\downarrow\uparrow} \rangle$. Thus: y is reducible iff $\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle$ is an infimum of attribute concepts different from $\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle$. Now, since every concept $\langle A, B \rangle$ is an infimum of some attribute concepts (attribute concepts are $\bigwedge$-dense), we get that y is not reducible iff $\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle$ is $\bigwedge$-irreducible in $\mathcal{B}(X, Y, I)$.
- Therefore, if $\langle X, Y, I \rangle$ is clarified, y is not reducible iff $\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle$ is $\bigwedge$-irreducible.

Clarified and reduced formal contexts

- Suppose $\langle X, Y, I \rangle$ is not clarified due to $\{y\}^{\downarrow} = \{z\}^{\downarrow}$ for some $z \neq y$. Then y is reducible by definition (just put $Y' = \{z\}$ in the definition). Still, it can happen that $\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle$ is $\bigwedge$-irreducible and it can happen that y is $\bigwedge$-reducible, see the next example.
- Example. Two non-clarified contexts. Left: y_2 reducible and $\langle \{y_2\}^{\downarrow}, \{y_2\}^{\downarrow\uparrow} \rangle$ $\bigwedge$-reducible. Right: y_2 reducible but $\langle \{y_2\}^{\downarrow}, \{y_2\}^{\downarrow\uparrow} \rangle$ $\bigwedge$-irreducible.

I	y_1	y_2	y_3	y_4
x_1			×	
x_2	×	×	×	×
x_3	×	×	×	×
x_4	×			

I	y_1	y_2	y_3	y_4	y_5
x_1	×		×		
x_2		×		×	
x_3	×	×	×	×	
x_4	×		×		

- The same for reducibility of objects: If $\langle X, Y, I \rangle$ is clarified, then x is not reducible iff $\langle \{x\}^{\uparrow\downarrow}, \{x\}^{\uparrow} \rangle$ is $\bigvee$-irreducible in $\mathcal{B}(X, Y, I)$.
- Therefore, it is convenient to consider reducibility on clarified contexts (then, reducibility of objects and attributes corresponds to $\bigvee$- and $\bigwedge$-reducibility of object concepts and attribute concepts).

Theorem

Let $y \in Y$ be reducible in $\langle X, Y, I \rangle$. Then $\mathcal{B}(X, Y - \{y\}, J)$ is isomorphic to $\mathcal{B}(X, Y, I)$ where $J = I \cap (X \times (Y - \{y\}))$ is the restriction of I to $X \times Y - \{y\}$, i.e., $\langle X, Y - \{y\}, J \rangle$ results by removing column y from $\langle X, Y, I \rangle$.

Proof.

Follows from part (2) of Main theorem of concept lattices:
Namely, $\mathcal{B}(X, Y - \{y\}, J)$ is isomorphic to $\mathcal{B}(X, Y, I)$ iff there are mappings $\gamma : X \to \mathcal{B}(X, Y, I)$ and $\mu : Y - \{y\} \to \mathcal{B}(X, Y, I)$ such that (a) $\gamma(X)$ is $\bigvee$-dense in $\mathcal{B}(X, Y, I)$, (b) $\mu(Y - \{y\})$ is $\bigwedge$-dense in $\mathcal{B}(X, Y, I)$, and (c) $\gamma(x) \leq \mu(z)$ iff $\langle x, z \rangle \in J$. If we define $\gamma(x)$ and $\mu(z)$ to be the object and attribute concept of $\mathcal{B}(X, Y, I)$ corresponding to x and z, respectively, then:
(a) is evident.
(c) is satisfied because for $z \in Y - \{z\}$ we have $\langle x, z \rangle \in J$ iff $\langle x, z \rangle \in I$ (J is a restriction of I).

□

cntd.

(b): We need to show that each $\langle A, B\rangle \in \mathcal{B}(X, Y, I)$ is an infimum of attribute concepts different from $\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow}\rangle$. But this is true because y is reducible: Namely, if $\langle A, B\rangle \in \mathcal{B}(X, Y, I)$ is the infimum of attribute concepts which include $\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow}\rangle$, then we may replace $\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow}\rangle$ by the attribute concepts $\langle \{z\}^{\downarrow}, \{z\}^{\downarrow\uparrow}\rangle$, $z \in Y'$ (cf. definition of reducible attribute), of which $\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow}\rangle$ is the infimum. □

Definition (reduced formal context)

$\langle X, Y, I\rangle$ is

- row reduced if no object $x \in X$ is reducible,
- column reduced if no attribute $y \in Y$ is reducible,
- reduced if it is both row reduced and column reduced.

- By above observation: If $\langle X, Y, I\rangle$ is not clarified, then either some object is reducible (if there are identical rows) or some attribute is reducible (if there are identical columns). Therefore, if $\langle X, Y, I\rangle$ is reduced, it is clarified.
- The relationship between reducibility of objects/attributes and $\bigvee$- and $\bigwedge$-reducibility of object/attribute concepts gives:

observation

A clarified $\langle X, Y, I\rangle$ is

- row reduced iff every object concept is $\bigvee$-irreducible,
- column reduced iff every attribute concept is $\bigwedge$-irreducible.

Reducing formal context by arrow relations

How to find out which objects and attributes are reducible?

Definition (arrow relations)

For $\langle X, Y, I\rangle$, define relations $\nearrow$, $\swarrow$, and $\updownarrow$ between X and Y by

- $x \swarrow y$ iff $\langle x, y\rangle \notin I$ and if $\{x\}^\uparrow \subset \{x_1\}^\uparrow$ then $\langle x_1, y\rangle \in I$.
- $x \nearrow y$ iff $\langle x, y\rangle \notin I$ and if $\{y\}^\downarrow \subset \{y_1\}^\downarrow$ then $\langle x, y_1\rangle \in I$.
- $x \updownarrow y$ iff $x \swarrow y$ and $x \nearrow y$.

Therefore, if $\langle x, y\rangle \in I$ then none of $x \swarrow y$, $x \nearrow y$, $x \updownarrow y$ occurs. The arrow relations can therefore be entered in the table of $\langle X, Y, I\rangle$ such as

I	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×	×		
x_3		×	×	×
x_4		×		
x_5		×	×	

I	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×	×	$\updownarrow$	$\swarrow$
x_3	$\updownarrow$	×	×	×
x_4	$\nearrow$	×	$\nearrow$	
x_5	$\nearrow$	×	×	$\updownarrow$

Reducing formal context by arrow relations

Theorem (arrow relations and reducibility)

For any $\langle X, Y, I\rangle$, $x \in X$, $y \in Y$*:*

- $\langle \{x\}^{\uparrow\downarrow}, \{x\}^{\uparrow}\rangle$ *is* $\bigvee$*-irreducible iff there is* $y \in Y$ *s.t.* $x \swarrow y$*;*
- $\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow}\rangle$ *is* $\bigwedge$*-irreducible iff there is* $x \in Y$ *s.t.* $x \nearrow y$*.*

Proof.

Due to duality, we verify $\bigwedge$-irreducibility:
$x \nearrow y$ IFF
$x \not\in \{y\}^{\downarrow}$ and for every y_1 with $\{y\}^{\downarrow} \subset \{y_1\}^{\downarrow}$ we have $x \in \{y_1\}^{\downarrow}$ IFF
$\{y\}^{\downarrow} \subset \bigcap_{y_1 : \{y\}^{\downarrow} \subset \{y_1\}^{\downarrow}}$ IFF
$\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow}\rangle$ is not an infimum of other attribute concepts IFF
$\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow}\rangle$ is $\bigwedge$-irreducible. □

Reducing formal context by arrow relations

Problem:
INPUT: (arbitrary) formal context $\langle X_1, Y_1, I_1 \rangle$
OUTPUT: a reduced context $\langle X_2, Y_2, I_2 \rangle$

Algorithm:

1. clarify $\langle X_1, Y_1, I_1 \rangle$ to get a clarified context $\langle X_3, Y_3, I_3 \rangle$ (removing identical rows and columns),
2. compute arrow relations $\swarrow$ and $\nearrow$ for $\langle X_3, Y_3, I_3 \rangle$,
3. obtain $\langle X_2, Y_2, I_2 \rangle$ from $\langle X_3, Y_3, I_3 \rangle$ by removing objects x from X_3 for which there is no $y \in Y_3$ with $x \swarrow y$, and attributes y from Y_3 for which there is no $x \in X_3$ with $x \nearrow y$. That is:
$X_2 = X_3 - \{x \mid \text{ there is no } y \in Y_3 \text{ s. t. } x \swarrow y\}$,
$Y_2 = Y_3 - \{y \mid \text{ there is no } x \in X_3 \text{ s. t. } x \nearrow y\}$,
$I_2 = I_3 \cap (X_2 \times Y_2)$.

Reducing formal context by arrow relations

Example (arrow relations)

Compute arrow relations $\swarrow$, $\nearrow$, $\updownarrow$ for the following formal context:

I_1	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×	×		
x_3		×	×	×
x_4		×		
x_5		×	×	

Start with $\nearrow$. We need to go through cells in the table not containing $\times$ and decide whether $\nearrow$ applies.

The first such cell corresponds to $\langle x_2, y_3\rangle$. By definition, $x_2 \nearrow y_3$ iff for each $y \in Y$ such that $\{y_3\}^{\downarrow} \subset \{y\}^{\downarrow}$ we have $x_2 \in \{y\}^{\downarrow}$. The only such y is y_2 for which we have $x_2 \in \{y_2\}^{\downarrow}$, hence $x_2 \nearrow y_3$.

And so on up to $\langle x_5, y_4\rangle$ for which we get $x_5 \nearrow y_4$.

Reducing formal context by arrow relations

Example (arrow relations cntd.)

Compute arrow relations $\swarrow$, $\nearrow$, $\updownarrow$ for the following formal context:

I_1	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×	×		
x_3		×	×	×
x_4		×		
x_5		×	×	

Continue with $\swarrow$. Go through cells in the table not containing $\times$ and decide whether $\swarrow$ applies. The first such cell corresponds to $\langle x_2, y_3\rangle$. By definition, $x_2 \swarrow y_3$ iff for each $x \in X$ such that $\{x_2\}^{\uparrow} \subset \{x\}^{\uparrow}$ we have $y_3 \in \{x\}^{\uparrow}$. The only such x is x_1 for which we have $y_3 \in \{x_1\}^{\uparrow}$, hence $x_2 \swarrow y_3$.

And so on up to $\langle x_5, y_4\rangle$ for which we get $x_5 \swarrow y_4$.

Reducing formal context by arrow relations

Example (arrow relations cntd. – result)

Compute arrow relations $\swarrow$, $\nearrow$, $\updownarrow$ for the following formal context (left):

I_1	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×	×		
x_3		×	×	×
x_4		×		
x_5		×	×	

I_1	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×	×	$\updownarrow$	$\swarrow$
x_3	$\updownarrow$	×	×	×
x_4	$\nearrow$	×	$\nearrow$	
x_5	$\nearrow$	×	×	$\updownarrow$

The arrow relations are indicated in the right table. Therefore, the corresponding reduced context is

I_2	y_1	y_3	y_4
x_2	×		
x_3		×	×
x_5		×	

Reducing formal context by arrow relations

For a complete lattice $\langle V, \leq \rangle$ and $v \in V$, denote

$$v_* = \bigvee_{u \in V, u < v} u,$$
$$v^* = \bigwedge_{u \in V, v < u} u.$$

exercise

- Show that $x \swarrow y$ iff $\langle \{x\}^{\uparrow\downarrow}, \{x\}^{\uparrow} \rangle \vee \langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle = \langle \{x\}^{\uparrow\downarrow}, \{x\}^{\uparrow} \rangle_* < \langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle$,
- Show that $x \nearrow y$ iff $\langle \{x\}^{\uparrow\downarrow}, \{x\}^{\uparrow} \rangle \wedge \langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle = \langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle^* > \langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow} \rangle$.

Reducing formal context by arrow relations

Let $\langle X_1, Y_1, I_1\rangle$ be clarified, $X_2 \subseteq X_1$ and $Y_2 \subseteq Y_1$ be sets of irreducible objects and attributes, respectively, let $I_2 = I_1 \cap (X_2 \times Y_2)$ (restriction of I_1 to irreducible objects and attributes).

How can we obtain from concepts of $\mathcal{B}(X_1, Y_1, I_1)$ from those of $\mathcal{B}(X_2, Y_2, I_2)$? Answer is based on:

1. $\langle A_1, B_1\rangle \mapsto \langle A_1 \cap X_2, B_1 \cap Y_2\rangle$ is an isomorphism from $\mathcal{B}(X_1, Y_1, I_1)$ on $\mathcal{B}(X_2, Y_2, I_2)$.
2. therefore, each extent A_2 of $\mathcal{B}(X_2, Y_2, I_2)$ is of the form $A_2 = A_1 \cap X_2$ where A_1 is an extent of $\mathcal{B}(X_1, Y_1, I_1)$ (same for intents).
3. for $x \in X_1$: $x \in A_1$ iff $\{x\}^{\uparrow\downarrow} \cap X_2 \subseteq A_1 \cap X_2$,
   for $y \in Y_1$: $y \in B_1$ iff $\{y\}^{\downarrow\uparrow} \cap Y_2 \subseteq B_1 \cap Y_2$.

Here, $^\uparrow$ and $^\downarrow$ are operators induced by $\langle X_1, Y_1, I_1\rangle$.

Therefore, given $\langle A_2, B_2\rangle \in \mathcal{B}(X_2, Y_2, I_2)$, the corresponding $\langle A_1, B_1\rangle \in \mathcal{B}(X_1, Y_1, I_1)$ is given by

$$A_1 = A_2 \cup \{x \in X_1 - X_2 \mid \{x\}^{\uparrow\downarrow} \cap X_2 \subseteq A_2\}, \quad (17)$$

$$B_1 = B_2 \cup \{y \in Y_1 - Y_2 \mid \{y\}^{\downarrow\uparrow} \cap Y_2 \subseteq B_2\}. \quad (18)$$

Reducing formal context by arrow relations

Example

Left is a clarified formal context $\langle X_1, Y_1, I_1 \rangle$, right is a reduced context $\langle X_2, Y_2, I_2 \rangle$ (see previous example).

I_1	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×	×		
x_3		×	×	×
x_4		×		
x_5		×	×	

I_2	y_1	y_3	y_4
x_2	×		
x_3		×	×
x_5		×	

Determine $\mathcal{B}(X_1, Y_1, I_1)$ by first computing $\mathcal{B}(X_2, Y_2, I_2)$ and then using the method from the previous slide to obtain concepts $\mathcal{B}(X_1, Y_1, I_1)$ from the corresponding concepts from $\mathcal{B}(X_2, Y_2, I_2)$.

Example (cntd.)

I_1	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×	×		
x_3		×	×	×
x_4		×		
x_5		×	×	

I_2	y_1	y_3	y_4
x_2	×		
x_3		×	×
x_5		×	

$\mathcal{B}(X_2, Y_2, I_2)$ consists of:
$\langle \emptyset, Y_2 \rangle$, $\langle \{x_2\}, \{y_1\} \rangle$, $\langle \{x_3\}, \{y_3, y_4\} \rangle$, $\langle \{x_3, x_5\}, \{y_3\} \rangle$, $\langle X_2, \emptyset \rangle$.
We need to go through all $\langle A_2, B_2 \rangle \in \mathcal{B}(X_2, Y_2, I_2)$ and determine the corresponding $\langle A_1, B_1 \rangle \in \mathcal{B}(X_1, Y_1, I_1)$ using (17) and (18). Note: $X_1 - X_2 = \{x_1, x_4\}$, $Y_1 - Y_2 = \{y_2\}$.

1. for $\langle A_2, B_2 \rangle = \langle \emptyset, Y_2 \rangle$ we have
   $\{x_1\}^{\uparrow\downarrow} \cap X_2 = \{x_1\} \cap X_2 = \emptyset \subseteq A_2$,
   $\{x_4\}^{\uparrow\downarrow} \cap X_2 = X_1 \cap X_2 = X_2 \not\subseteq A_2$,
   hence $A_1 = A_2 \cup \{x_1\} = \{x_1\}$, and
   $\{y_2\}^{\downarrow\uparrow} \cap Y_2 = \{y_2\} \cap Y_2 = \emptyset \subseteq B_2$,
   hence $B_1 = B_2 \cup \{y_2\} = Y_1$. So, $\langle A_1, B_1 \rangle = \langle \{x_1\}, Y_1 \rangle$.

Example (cntd.)

I_1	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×	×		
x_3		×	×	×
x_4		×		
x_5		×	×	

I_2	y_1	y_3	y_4
x_2	×		
x_3		×	×
x_5		×	

2. for $\langle A_2, B_2\rangle = \langle\{x_2\}, \{y_1\}\rangle$ we have
   $\{x_1\}^{\uparrow\downarrow} \cap X_2 = \emptyset \subseteq A_2$, $\{x_4\}^{\uparrow\downarrow} \cap X_2 = X_2 \not\subseteq A_2$,
   hence $A_1 = A_2 \cup \{x_1\} = \{x_1, x_2\}$, and
   $\{y_2\}^{\downarrow\uparrow} \cap Y_2 = \{y_2\} \cap Y_2 = \emptyset \subseteq B_2$,
   hence $B_1 = B_2 \cup \{y_2\} = \{y_1, y_2\}$. So, $\langle A_1, B_1\rangle = \langle\{x_1, x_2\}, \{y_1, y_2\}\rangle$.
3. for $\langle A_2, B_2\rangle = \langle\{x_3\}, \{y_3, y_4\}\rangle$ we have
   $\{x_1\}^{\uparrow\downarrow} \cap X_2 = \emptyset \subseteq A_2$, $\{x_4\}^{\uparrow\downarrow} \cap X_2 = X_2 \not\subseteq A_2$,
   hence $A_1 = A_2 \cup \{x_1\} = \{x_1, x_3\}$, and
   $\{y_2\}^{\downarrow\uparrow} \cap Y_2 = \{y_2\} \cap Y_2 = \emptyset \subseteq B_2$,
   hence $B_1 = B_2 \cup \{y_2\} = \{y_2, y_3, y_4\}$. So,
   $\langle A_1, B_1\rangle = \langle\{x_1, x_3\}, \{y_2, y_3, y_4\}\rangle$.

Example (cntd.)

I_1	y_1	y_2	y_3	y_4
x_1	×	×	×	×
x_2	×	×		
x_3		×	×	×
x_4		×		
x_5		×	×	

I_2	y_1	y_3	y_4
x_2	×		
x_3		×	×
x_5		×	

4. for $\langle A_2, B_2\rangle = \langle\{x_3, x_5\}, \{y_3\}\rangle$ we have
   $\{x_1\}^{\uparrow\downarrow} \cap X_2 = \emptyset \subseteq A_2$, $\{x_4\}^{\uparrow\downarrow} \cap X_2 = X_2 \not\subseteq A_2$,
   hence $A_1 = A_2 \cup \{x_1\} = \{x_1, x_3, x_5\}$, and
   $\{y_2\}^{\downarrow\uparrow} \cap Y_2 = \{y_2\} \cap Y_2 = \emptyset \subseteq B_2$,
   hence $B_1 = B_2 \cup \{y_2\} = \{y_2, y_3\}$. So,
   $\langle A_1, B_1\rangle = \langle\{x_1, x_3, x_5\}, \{y_2, y_3\}\rangle$.
5. for $\langle A_2, B_2\rangle = \langle X_2, \emptyset\rangle$ we have
   $\{x_1\}^{\uparrow\downarrow} \cap X_2 = \emptyset \subseteq A_2$, $\{x_4\}^{\uparrow\downarrow} \cap X_2 = X_2 \subseteq A_2$,
   hence $A_1 = A_2 \cup \{x_1, x_4\} = X_1$, and
   $\{y_2\}^{\downarrow\uparrow} \cap Y_2 = \{y_2\} \cap Y_2 = \emptyset \subseteq B_2$,
   hence $B_1 = B_2 \cup \{y_2\} = \{y_2\}$. So, $\langle A_1, B_1\rangle = \langle X_1, \{y_2\}\rangle$.

Clarification and reduction

exercise

Determine a reduced context from the following formal context. Use the reduced context to compute $\mathcal{B}(X, Y, I)$.

I	y_1	y_2	y_3	y_4	y_5
x_1					
x_2		×		×	
x_3		×	×	×	
x_4		×		×	×
x_5		×	×		
x_6		×	×	×	
x_7	×	×	×		

Hint: First clarify, then compute arrow relations.

Algorithms for computing concept lattices

problem:
INPUT: formal context $\langle X, Y, I\rangle$,
OUTPUT: concept lattice $\mathcal{B}(X, Y, I)$ (possibly plus $\leq$)

- Sometimes one needs to compute the set $\mathcal{B}(X, Y, I)$ of formal concepts only.
- Sometimes one needs to compute both the set $\mathcal{B}(X, Y, I)$ and the conceptual hierarchy $\leq$. $\leq$ can be computed from $\mathcal{B}(X, Y, I)$ by definition of $\leq$. But this is not efficient. Algorithms exist which can compute $\mathcal{B}(X, Y, I)$ and $\leq$ simultaneously, which is more efficient (faster) than first computing $\mathcal{B}(X, Y, I)$ and then computing $\leq$.

survey: Kuznetsov S. O., Obiedkov S. A.: Comparing performance of algorithms for generating concept lattices. *J. Experimental & Theoretical Artificial Intelligence* **14**(2003), 189–216.

We will introduce:

- Ganter's NextClosure algorithm (computes $\mathcal{B}(X, Y, I)$),
- Lindig's UpperNeighbor algorithm (computes $\mathcal{B}(X, Y, I)$ and $\leq$).

NextClosure Algorithm

- author: Bernhard Ganter (1987)
- input: formal context $\langle X, Y, I\rangle$,
- output: $\mathrm{Int}(X, Y, I)$ ... all intents (dually, $\mathrm{Ext}(X, Y, I)$ ... all extents),
- list all intents (or extents) in lexicographic order,
- note that $\mathcal{B}(X, Y, I)$ can be reconstructed from $\mathrm{Int}(X, Y, I)$ due to

$$\mathcal{B}(X, Y, I) = \{\langle B^{\downarrow}, B\rangle \mid B \in \mathrm{Int}(X, Y, I)\},$$

- one of most popular algorithms, easy to implement,
- we present NextClosure for intents.

NextClosure Algorithm

suppose $Y = \{1, \ldots, n\}$
(that is, we denote attributes by positive integers, this way, we fix an ordering of attributes)

Definition (lexicographic ordering of sets of attributes)

For $A, B \subseteq Y$, $i \in \{1, \ldots, n\}$ put

$$A <_i B \quad \text{iff} \quad i \in B - A \text{ a } A \cap \{1, \ldots, i-1\} = B \cap \{1, \ldots, i-1\},$$
$$A < B \quad \text{iff} \quad A <_i B \text{ for some } i.$$

Note: $<$ ...lexicographic ordering (thus, every two distinct sets $A, B \subseteq$ are comparable).

For $i = 1$, we put $\{1, \ldots, i-1\} = \emptyset$.

One may think of $B \subseteq Y$ in terms of its characteristic vector. For $Y = \{1, 2, 3, 4, 5, 6, 7\}$ and $B = \{1, 3, 4, 6\}$, the characteristic vector of B is 1011010.

NextClosure Algorithm

Example

Let $Y = \{1, 2, 3, 4, 5, 6\}$, consider sets $\{1\}$, $\{2\}$, $\{2, 3\}$, $\{3, 4, 5\}$, $\{3, 6\}$, $\{1, 4, 5\}$. We have

- $\{2\} <_1 \{1\}$ because $1 \in \{1\} - \{2\} = \{1\}$ and $A \cap \emptyset = B \cap \emptyset$. Characteristic vectors: $010000 <_1 100000$.
- $\{3, 6\} <_4 \{3, 4, 5\}$ because $4 \in \{3, 4, 5\} - \{3, 6\} = \{4, 5\}$ and $A \cap \{1, 2, 3\} = B \cap \{1, 2, 3\}$. Characteristic vectors: $001001 <_4 001110$.
- All sets ordered lexicographically: $\{3, 6\} <_4 \{3, 4, 5\} <_2 \{2\} <_3 \{2, 3\} <_1 \{1\} <_4 \{1, 4, 5\}$. Characteristic vectors: $001001 <_4 001110 <_2 010000 <_3 011000 <_1 100000 <_4 100110$.

Note: if $B_1 \subset B_2$ then $B_1 < B_2$.

NextClosure Algorithm

Definition

For $A \subseteq Y$, $i \in \{1, \ldots, n\}$, put

$$A \oplus i := ((A \cap \{1, \ldots, i-1\}) \cup \{i\})^{\downarrow\uparrow}.$$

Example

I	1	2	3	4
x_1	×	×		×
x_2	×	×	×	×
x_3		×		

- $A = \{1, 3\}$, $i = 2$.
  $A \oplus i = ((\{1, 3\} \cap \{1, 2\}) \cup \{2\})^{\downarrow\uparrow} = (\{1\} \cup \{2\})^{\downarrow\uparrow} = \{1, 2\}^{\downarrow\uparrow} = \{1, 2, 4\}$.
- $A = \{2\}$, $i = 1$.
  $A \oplus i = ((\{2\} \cap \emptyset) \cup \{1\})^{\downarrow\uparrow} = \{1\}^{\downarrow\uparrow} = \{1, 2, 4\}$.

Lemma

For any $B, D, D_1, D_2 \subseteq Y$:

(1) *If* $B <_i D_1$, $B <_j D_2$, *and* $i < j$ *then* $D_2 <_i D_1$;

(2) *if* $i \notin B$ *then* $B < B \oplus i$;

(3) *if* $B <_i D$ *and* $D = D^{\downarrow\uparrow}$ *then* $B \oplus i \subseteq D$;

(4) *if* $B <_i D$ *and* $D = D^{\downarrow\uparrow}$ *then* $B <_i B \oplus i$.

Proof.

(1) by easy inspection.
(2) is true because $B \cap \{1, \ldots, i-1\} \subseteq B \oplus i \cap \{1, \ldots, i-1\}$ and $i \in (B \oplus i) - B$.
(3) Putting $C_1 = B \cap \{1, \ldots, i-1\}$ and $C_2 = \{i\}$ we have $C_1 \cup C_2 \subseteq D$, and so $B \oplus i = (C_1 \cup C_2)^{\downarrow\uparrow} \subseteq D^{\downarrow\uparrow} = D$.
(4) By assumption, $B \cap \{1, \ldots, i-1\} = D \cap \{1, \ldots, i-1\}$. Furthermore, (3) yields $B \oplus i \subseteq D$ and so $B \cap \{1, \ldots, i-1\} \supseteq B \oplus i \cap \{1, \ldots, i-1\}$. On the other hand, $B \oplus i \cap \{1, \ldots, i-1\} \supseteq (B \cap \{1, \ldots, i-1\})^{\downarrow\uparrow} \cap \{1, \ldots, i-1\} \supseteq B \cap \{1, \ldots, i-1\}$. Therefore, $B \cap \{1, \ldots, i-1\} = B \oplus i \cap \{1, \ldots, i-1\}$. Finally, $i \in B \oplus i$.

NextClosure Algorithm

Theorem (lexicographic successor)

The least intent B^+ *greater (w.r.t.* $<$*) than* $B \subseteq Y$ *is given by*

$$B^+ = B \oplus i$$

where i *is the greatest one with* $B <_i B \oplus i$*.*

Proof.

Let B^+ be the least intent greater than B (w.r.t. $<$). We have $B < B^+$ and thus $B <_i B^+$ for some i such that $i \in B^+$. By Lemma (4), $B <_i B \oplus i$, i.e. $B < B \oplus i$. Lemma (3) yields $B \oplus i \leq B^+$ which gives $B^+ = B \oplus i$ since B^+ is the least intent with $B < B^+$. It remains to show that i is the greatest one satisfying $B <_i B \oplus i$. Suppose $B <_k B \oplus k$ for $k > i$. By Lemma (1), $B \oplus k <_i B \oplus i$ which is a contradiction to $B \oplus i = B^+ < B \oplus k$ (B^+ is the least intent greater than B and so $B^+ < B \oplus k$). Therefore we have $k = i$. □

pseudo-code of NextClosure algorithm:

```
A:=∅↓↑; (leastIntent)
store(A);
while not(A=Y) do
    A:=A+;
    store(A);
endwhile.
```

complexity: time complexity of computing A^+ is $O(|X| \cdot |Y|^2)$: complexity of computing $C^\uparrow$ is $O(|X| \cdot |Y|)$, for $D^\downarrow$ it is $O(|X| \cdot |Y|)$, thus for $D^{\downarrow\uparrow}$ it is $O(|X| \cdot |Y|)$; complexity of computing $A \oplus i$ is thus $O(|X| \cdot |Y|)$; to get A^+ we need to compute $A \oplus i$ $|Y|$-times in the worst case. As a result, complexity of computing A^+ is $O(|X| \cdot |Y|^2)$.

time complexity of NextClosure is $O(|X| \cdot |Y|^2 \cdot |\mathcal{B}(X, Y, I)|)$

$\Rightarrow$ **polynomial time delay complexity** (Johnson D. S., Yannakakis M., Papadimitrou C. H.: On generating all maximal independent sets. *Inf. Processing Letters* **27**(1988), 129–133.): going from A to A^+ in a polynomial time = NextClosure has polynomial time delay complexity

Note! Almost **no space requirements**. But: NextClosure does not directly give information about $\leq$.

Example (NextClosure Algorithm – simulation)

Simulate NextClosure algorithm on the following example.

I	1	2	3
x_1	×	×	×
x_2	×		×
x_3		×	×
x_4	×		

1. $A = \emptyset^{\downarrow\uparrow} = \emptyset$.
2. Next, we are looking for A^+, i.e. $\emptyset^+$, which is $A \oplus i$ s.t. i is the largest one with $A <_i A \oplus i$. We proceed for $i = 3, 2, 1$ and test whether $A <_i A \oplus i$:
   - $i = 3$: $A \oplus i = \{3\}^{\downarrow\uparrow} = \{3\}$ and $\emptyset <_3 \{3\} = A \oplus i$, therefore $A^+ = \{3\}$.
3. Next, $\{3\}^+$:
   - $i = 3$: $A \oplus i = \{3\}^{\downarrow\uparrow} = \{3\}$ and $\{3\} \not<_3 \{3\} = A \oplus i$, therefore we proceed for $i = 2$.
   - $i = 2$: $A \oplus i = \{2\}^{\downarrow\uparrow} = \{2, 3\}$ and $\{3\} <_2 \{2, 3\} = A \oplus i$, therefore $A^+ = \{2, 3\}$.

Example (cntd.)

4. Next, $\{2,3\}^+$:
   - $i = 3$: $A \oplus i = \{2,3\}^{\downarrow\uparrow} = \{2,3\}$ and $\{2,3\} \not<_3 \{2,3\} = A \oplus i$, therefore we proceed for $i = 2$.
   - $i = 2$: $A \oplus i = \{2\}^{\downarrow\uparrow} = \{2,3\}$ and $\{2,3\} \not<_2 \{2,3\} = A \oplus i$, therefore we proceed for $i = 1$.
   - $i = 1$: $A \oplus i = \{1\}^{\downarrow\uparrow} = \{1\}$ and $\{2,3\} <_1 \{1\} = A \oplus i$, therefore we $A^+ = \{1\}$.
5. Next, $\{1\}^+$:
   - $i = 3$: $A \oplus i = \{1,3\}^{\downarrow\uparrow} = \{1,3\}$ and $\{1\} <_3 \{1,3\} = A \oplus i$, therefore $A^+ = \{1,3\}$.
6. Next, $\{1,3\}^+$:
   - $i = 3$: $A \oplus i = \{1,3\}^{\downarrow\uparrow} = \{1,3\}$ and $\{1,3\} \not<_3 \{1,3\} = A \oplus i$, therefore we proceed for $i = 2$.
   - $i = 2$: $A \oplus i = \{1,2\}^{\downarrow\uparrow} = \{1,2,3\}$ and $\{1,3\} <_2 \{1,2,3\} = A \oplus i$, therefore $A^+ = \{1,2,3\} = Y$.

Therefore, the intents from $\mathrm{Int}(X, Y, I)$, ordered lexicographically, are: $\emptyset < \{3\} < \{2,3\} < \{1\} < \{1,3\} < \{1,2,3\}$.

Example (cntd.)

I	1	2	3
x_1	×	×	×
x_2	×		×
x_3		×	×
x_4	×		

$\mathrm{Int}(X, Y, I) = \{\emptyset, \{3\}, \{2,3\}, \{1\}, \{1,3\}, \{1,2,3\}\}$.

From this list, we can get the corresponding extents:
$X = \emptyset^{\downarrow}$, $\{x_1, x_2, x_3\} = \{3\}^{\downarrow}$, $\{x_1, x_3\} = \{2,3\}^{\downarrow}$, $\{x_1, x_3, x_4\} = \{1\}^{\downarrow}$, $\{x_1, x_2\} = \{1,3\}^{\downarrow}$, $\{x_1\} = \{1,2,3\}^{\downarrow}$.
Therefore, $\mathcal{B}(X, Y, I)$ consists of: $\langle\{x_1\}, \{1,2,3\}\rangle$, $\langle\{x_1, x_2\}, \{1,3\}\rangle$, $\langle\{x_1, x_3\}, \{2,3\}\rangle$, $\langle\{x_1, x_2, x_3\}, \{3\}\rangle$, $\langle\{x_1, x_2, x_4\}, \{1\}\rangle$, $\langle\{x_1, x_2, x_3, x_4\}, \emptyset\rangle$.

NextClosure Algorithm

- If $^{\downarrow\uparrow}$ is replaced by an arbitrary closure operator C, NextClosure computes all fixpoints of C. This is easy to see: all that matters in the proofs of Theorem and Lemma justifying correctness of NextClosure, is that $^{\downarrow\uparrow}$ is a closure operator.
- Therefore, NextClosure is essentially an algorithm for computing all fixpoints of a given closure operator C.
- Computational complexity of NextClosure depends on computational complexity of computing $C(A)$ (computing closure of arbitrary set A).

UpperNeighbor Algorithm

- author: Christian Lindig (Fast Concept Analysis, 2000)
- input: formal context $\langle X, Y, I \rangle$,
- output: $\mathcal{B}(X, Y, I)$ and $\leq$
- idea:
  1. start with the least formal concept $\langle \emptyset^{\uparrow\downarrow}, \emptyset^{\uparrow} \rangle$,
  2. for each $\langle A, B \rangle$ generate all its upper neighbors (and store the necessary information)
  3. go to the next concept.
- Details can be found at `http://www.st.cs.uni-sb.de/~lindig/papers/fast-ca/iccs-lindig.pdf`
- Crucial point: how to compute upper neighbors of a given $\langle A, B \rangle$.

UpperNeighbor Algorithm

Theorem (upper neighbors of formal concept)

If $\langle A, B\rangle \in \mathcal{B}(X, Y, I)$ is not the largest concept then $(A \cup \{x\})^{\uparrow\downarrow}$, with $x \in X - A$, is an extent of an upper neighbor of $\langle A, B\rangle$ iff for each $z \in (A \cup \{x\})^{\uparrow\downarrow} - A$ we have $(A \cup \{x\})^{\uparrow\downarrow} = (A \cup \{z\})^{\uparrow\downarrow}$.

Remark

In general, for $x \in X - A$, $(A \cup \{x\})^{\uparrow\downarrow}$ need not be an extent of an upper neighbor of $\langle A, B\rangle$. Find an example.

UpperNeighbor Algorithm

pseudo-code of UpperNeighbor procedure:

min:= $X - A$;
neighbors:= $\emptyset$;
for $x \in X - A$ do
  $B_1 := (A \cup \{x\})^{\uparrow}$; $A_1 := B_1^{\downarrow}$;
  if (min $\cap((A_1 - A) - \{x\}) = \emptyset$) then
   neighbors:=neighbors $\cup\{(A_1, B_1)\}$
  else min:=min $-\{x\}$;
enddo.

complexity: polynomial time delay with delay $O(|X|^2 \cdot |Y|)$ (same as NextClosure – version for extents)

Example (UpperNeighbor – simulation)

I	1	2	3
x_1	×	×	×
x_2	×		×
x_3		×	×
x_4	×		

Determine all upper neighbors of the least concept $\langle A, B\rangle = \langle \emptyset^{\uparrow\downarrow}, \emptyset^{\uparrow}\rangle = \langle \{x_1\}, \{1, 2, 3\}\rangle$.

- according to 1., and 2., $min := \{x_2, x_3, x_4\}$, $neighbors := \emptyset$.
- run loop 3.–8. for $x \in \{x_2, x_3, x_4\}$.
- for $x = x_2$:
  - 4. $B_1 = \{x_1, x_2\}^{\uparrow} = \{1, 3\}$, $A_1 = B_1^{\downarrow} = \{x_1, x_2\}$.
  - 5. $min \cap ((A_1 - A) - \{x\}) = \{x_2, x_3, x_4\} \cap ((\{x_1, x_2\} - \{x_1\}) - \{x_2\}) = \{x_2, x_3, x_4\} \cap \emptyset = \emptyset$, therefore $neighbors := \{\langle\{x_1, x_2\}, \{1, 3\}\rangle\}$.
- for $x = x_3$:
  - 4. $B_1 = \{x_1, x_3\}^{\uparrow} = \{2, 3\}$, $A_1 = B_1^{\downarrow} = \{x_1, x_3\}$.
  - 5. $min \cap ((A_1 - A) - \{x\}) = \{x_2, x_3, x_4\} \cap ((\{x_1, x_3\} - \{x_1\}) - \{x_3\}) = \{x_2, x_3, x_4\} \cap \emptyset = \emptyset$, therefore $neighbors := \{\langle\{x_1, x_2\}, \{1, 3\}\rangle, \langle\{x_1, x_3\}, \{2, 3\}\rangle\}$.

Example (UpperNeighbor – simulation)

I	1	2	3
x_1	×	×	×
x_2	×		×
x_3		×	×
x_4	×		

- for $x = x_4$:
  - 4. $B_1 = \{x_1, x_4\}^{\uparrow} = \{1\}$, $A_1 = B_1^{\downarrow} = \{x_1, x_2, x_4\}$.
  - 5. $min \cap ((A_1 - A) - \{x\}) = \{x_2, x_3, x_4\} \cap ((\{x_1, x_2, x_4\} - \{x_1\}) - \{x_4\}) = \{x_2, x_3, x_4\} \cap \{x_2\} = \{x_2\}$, therefore *neighbors* does not change and we proceed with 7. and set $min := min - \{x_4\} = \{x_2, x_3\}$.
- loop 3.–8. ends, result is $neighbors = \{\langle\{x_1, x_2\}, \{1, 3\}\rangle, \langle\{x_1, x_3\}, \{2, 3\}\rangle\}$.

This is correct since $\mathcal{B}(X, Y, I)$ consists of $\langle\{x_1\}, \{1, 2, 3\}\rangle$, $\langle\{x_1, x_2\}, \{1, 3\}\rangle$, $\langle\{x_1, x_3\}, \{2, 3\}\rangle$, $\langle\{x_1, x_2, x_3\}, \{3\}\rangle$, $\langle\{x_1, x_2, x_4\}, \{1\}\rangle$, $\langle\{x_1, x_2, x_3, x_4\}, \emptyset\rangle$.

Many-valued contexts and conceptual scaling

– many-valued formal contexts = tables like

	age	education	symptom
Alice	23	BS	1
Boris	30	MS	0
Cyril	31	PhD	1
David	43	MS	0
Ellen	24	PhD	1
Fred	64	MS	0
George	30	Bc	0

– how to use FCA to such data? $\Rightarrow$ conceptual scaling

– conceptual scaling = transformation of many-valued formal contexts to ordinary formal contexts such as

Many-valued contexts and conceptual scaling

	a_y	a_m	a_o	e_{BS}	e_{MS}	e_{PhD}	symptom
Alice	1	0	0	1	0	0	1
Boris	1	0	0	0	1	0	0
Cyril	0	1	0	0	0	1	1
David	0	1	0	0	1	0	0
Ellen	1	0	0	0	0	1	1
Fred	0	0	1	0	1	0	0
George	1	0	0	1	0	0	0

– new attributes introduced:

a_y ... young, a_m ... middle-aged, a_o ... old, e_{BS} ... highest education BS, e_{MS} ... highest education MS, e_{PhD} ... highest education PhD.

– After scaling, the data can be processed by means of FCA.

– Scaling needs to be done with assistance of a user:
  - what kind of new attributes to introduce?
  - how many? (rule: the more, the larger the concept lattice)
  - how to scale? (nominal scaling, ordinal scaling, other types)

Many-valued contexts and conceptual scaling

Definition (many-valued context)

A many-valued context (data table with general attributes) is a tuple $\mathcal{D} = \langle X, Y, W, I \rangle$ where X is a non-empty finite set of objects, Y is a finite set of (many-valued) attributes, W is a set of values, and I is a ternary relation between X, Y, and W, i.e., $I \subseteq X \times Y \times W$, such that

$$\langle x, y, w \rangle \in I \text{ and } \langle x, y, v \rangle \in I \text{ imply } w = v.$$

remark

(1) A many-valued context can be thought of as representing a table with rows corresponding to $x \in X$, columns corresponding to $y \in Y$, and table entries at the intersection of row x and column y containing values $w \in W$ provided $\langle x, y, w \rangle \in I$ and containing blanks if there is no $w \in W$ with $\langle x, y, w \rangle \in I$.

Many-valued contexts and conceptual scaling

remark (cntd.)

(2) One can see that each $y \in Y$ can be considered a partial function from X to W. Therefore, we often write

$$y(x) = w \text{ instead of } \langle x, y, w \rangle \in I.$$

A set

$$\mathrm{dom}(y) = \{x \in X \mid \langle x, y, w \rangle \in I \text{ for some } w \in W\}$$

is called a domain of y. Attribute $y \in Y$ is called complete if $\mathrm{dom}(y) = X$, i.e. if the table contains some value in every row in the column corresponding to y. A many-valued context is called complete if each of its attributes is complete.

Many-valued contexts and conceptual scaling

remark (cntd.)

(3) From the point of view of theory of relational databases, a complete many-valued context is essentially a relation over a relation scheme Y. Namely, each $y \in Y$ can be considered an attribute in the sense of relational databases and putting

$$D_y = \{w \mid \langle x, y, w \rangle \in I \text{ for some } x \in X\},$$

D_y is a domain for y.
(4) We consider only complete many-valued contexts.

Example (many-valued context)

	age	education	symptom
Alice	23	BS	1
Boris	30	MS	0
Cyril	31	PhD	1
David	43	MS	0
Ellen	24	PhD	1
Fred	64	MS	0
George	30	Bc	0

represents a many-valued context $\langle X, Y, W, I\rangle$ with

- $X = \{$Alice, Boris, ..., George$\}$,
- $Y = \{$age, education, symptom$\}$,
- $W = \{0,1, \ldots, 150$, BS, MS, PhD, $0,1\}$,
- $\langle$Alice, age, 23$\rangle \in I$, $\langle$Alice, education, BS$\rangle \in I$, ..., $\langle$George, symptom, 0$\rangle \in I$.
- Using the above convention, we have age(Alice)=23, education(Alice)=BS, symptom(George)=0.

Many-valued contexts and conceptual scaling

Definition (scale)

Let $\langle X, Y, W, I\rangle$ be a many-valued context. A scale for attribute $y \in Y$ is a formal context (data table) $\mathbb{S}_y = \langle X_y, Y_y, I_y\rangle$ such that $D_y \subseteq X_y$. Objects $w \in X_y$ are called scale values, attributes of Y_y are called scale attributes.

Example (scale)

	e_{BS}	e_{MS}	e_{PhD}
BS	1	0	0
MS	0	1	0
PhD	0	0	1

is a scale for attribute $y =$education. Here, $\mathbb{S}_y = \langle X_y, Y_y, I_y\rangle$, $X_y = \{$BS, MS, PhD$\}$, $Y_y = \{\mathrm{e}_{BS}, \mathrm{e}_{MS}, \mathrm{e}_{PhD}\}$, I_y is given by the above table.

Many-valued contexts and conceptual scaling

Example (scale)

	a_y	a_m	a_o
0	1	0	0
⋮	1	0	0
30	1	0	0
31	0	1	0
⋮	0	1	0
60	0	1	0
61	0	0	1
⋮	0	0	1
150	0	0	1

	a_y	a_m	a_o
0–30	1	0	0
31–60	0	1	0
61–150	0	0	1

is a scale for attribute age (right table is a shorthand version of left table). Here, $\mathbb{S}_y = \langle X_y, Y_y, I_y \rangle$, $X_y = \{0, \dots, 150\}$, $Y_y = \{a_y, a_m, a_o\}$, I_y is given by the above table.

Many-valued contexts and conceptual scaling

Example (scale - granularity)

A different scale for attribute age is.

	a_{vy}	a_y	a_m	a_o	a_{vo}
0–25	1	0	0	0	0
26–35	0	1	0	0	0
36–55	0	0	1	0	0
56–75	0	0	0	1	0
76–150	0	0	0	0	1

a_{vy} . . . very young, a_y . . . young, a_m . . . middle aged, a_o . . . old, a_{vo} . . . very old.
The choice is made by a user and depends on his/her desired level of granularity (precision).

Scale defines the meaning of a scale attributes from Y_y. Two most important types are:

- nominal scale: values of attribute y are not ordered in any natural way (y is a nominal variable) or we do not want to take this ordering into consideration,
- ordinal scale: values of attribute y are ordered (y is an ordinal variable).

Example (nominal and ordinal scales)

Left: nominal scale for y =education. Right: ordinal scale for y =education with BS $<$ MS $<$ PhD.

	e_{BS}	e_{MS}	e_{PhD}
BS	1	0	0
MS	0	1	0
PhD	0	0	1

	e_{BS}	e_{MS}	e_{PhD}
BS	1	0	0
MS	1	1	0
PhD	1	1	1

For nominal scale: e_{MS} applies to individuals with highest degree MS
For ordinal scale: e_{MS} applies to individuals with degree at least MS (MS or higher)

Many-valued contexts and conceptual scaling

Assume $Y_{y_1} \cap Y_{y_2} = \emptyset$ for different $y_1, y_2 \in Y$.

Definition (plain scaling)

For a many-valued context $\mathcal{D} = \langle X, Y, W, I \rangle$ (as above), scales $\mathbb{S}_y$ ($y \in Y$), the derived formal context (w.r.t. plain scaling) is $\langle X, Z, J \rangle$ with attributes defined by

- $Z = \bigcup_{y \in Y} Y_y$,
- $\langle x, z \rangle \in J$ iff $y(x) = w$ and $\langle w, z \rangle \in I_y$.

Meaning of $\langle X, Y, W, I \rangle \mapsto \langle X, Z, J \rangle$:

- objects of the derived context are the same as of the original many-valued context;
- each column representing an attribute y is replaced by columns representing scale attributes $z \in Y_y$;
- attribute value $y(x)$ is replaced by the row of scale context $\mathbb{S}_y$.

Example

Formal context and nominal scales for age and education:

	age	education	symptom
Alice	23	BS	1
Boris	30	MS	0
Cyril	31	PhD	1
David	43	MS	0
Ellen	24	PhD	1
Fred	64	MS	0
George	30	Bc	0

	a_y	a_m	a_o
0–30	1	0	0
31–60	0	1	0
61–150	0	0	1

	e_{BS}	e_{MS}	e_{PhD}
BS	1	0	0
MS	0	1	0
PhD	0	0	1

Example

Derived formal context:

	a_y	a_m	a_o	e_{BS}	e_{MS}	e_{PhD}	symptom
Alice	1	0	0	1	0	0	1
Boris	1	0	0	0	1	0	0
Cyril	0	1	0	0	0	1	1
David	0	1	0	0	1	0	0
Ellen	1	0	0	0	0	1	1
Fred	0	0	1	0	1	0	0
George	1	0	0	1	0	0	0

Example

Formal context and nominal scale for age and ordinal scale for education:

	age	education	symptom
Alice	23	BS	1
Boris	30	MS	0
Cyril	31	PhD	1
David	43	MS	0
Ellen	24	PhD	1
Fred	64	MS	0
George	30	Bc	0

	a_y	a_m	a_o
0–30	1	0	0
31–60	0	1	0
61–150	0	0	1

	e_{BS}	e_{MS}	e_{PhD}
BS	1	0	0
MS	1	1	0
PhD	1	1	1

Example

Derived formal context:

	a_y	a_m	a_o	e_{BS}	e_{MS}	e_{PhD}	symptom
Alice	1	0	0	1	0	0	1
Boris	1	0	0	1	1	0	0
Cyril	0	1	0	1	1	1	1
David	0	1	0	1	1	0	0
Ellen	1	0	0	1	1	1	1
Fred	0	0	1	1	1	0	0
George	1	0	0	1	0	0	0

Example

- In the examples of derived formal context, what scale was used for attribute symptom?:

	symptom
0	
1	×

or (different notation)

	symptom
0	0
1	1

What is the impact of using nominal scale vs. ordinal scale? Compare concept lattices of two derived contexts, one one using nominal scale, the other using ordinal scale.

	education
Alice	BS
Boris	MS
Cyril	PhD
David	MS
Ellen	PhD
Fred	MS
George	BS

	e_{BS}	e_{MS}	e_{PhD}
Alice	1	0	0
Boris	0	1	0
Cyril	0	0	1
David	0	1	0
Ellen	0	0	1
Fred	0	1	0
George	1	0	0

	e_{BS}	e_{MS}	e_{PhD}
Alice	1	0	0
Boris	1	1	0
Cyril	1	1	1
David	1	1	0
Ellen	1	1	1
Fred	1	1	0
George	1	0	0