Drawing a concept lattice diagram

Drawing a line diagram

Concept No.	(extent	,	intent)
1	$(\{T_4\}$	,	$\{a, b, c\})$
2	$(\{T_1, T_2, T_4, T_6\}$	,	$\{b\})$
3	$(\{T_3, T_4, T_6\}$	,	$\{c\})$
4	$(\{T_1, T_5\}$	,	$\{d\})$
5	$(\{T_2, T_7\}$	,	$\{e\})$
6	$(\emptyset$	,	$\{a, b, c, d, e\})$
7	$(\{T_4, T_6\}$	,	$\{b, c\})$
8	$(\{T_1\}$	,	$\{b, d\})$
9	$(\{T_2\}$	,	$\{b, e\})$
10	$(\{T_1, T_2, T_3, T_4, T_5, T_6, T_7\}$	,	$\emptyset)$

Neighbors

Let (A_1, B_1) and (A_2, B_2) be formal concepts of (G, M, I).

$(A_1, B_1) < (A_2, B_2)$

(A_1, B_1) is a proper subconcept of (A_2, B_2) if

- $(A_1, B_1) \leq (A_2, B_2)$ and
- $(A1\ , B1\ ) \neq (A2\ , B2\ )$.

$(A_1, B_1) \prec (A_2, B_2)$

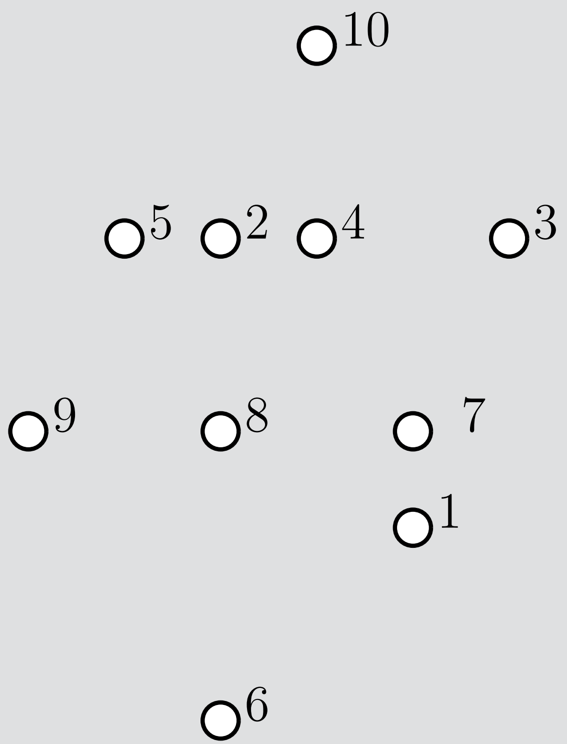
(A_1, B_1) is a lower neighbor of (A_2, B_2) if

- $(A_1, B_1) < (A_2, B_2)$ and
- $(A_1, B_1) < (A, B) < (A_2, B_2)$ for no formal concept (A, B).

(A_2, B_2) is then an upper neighbor of (A_1, B_1)

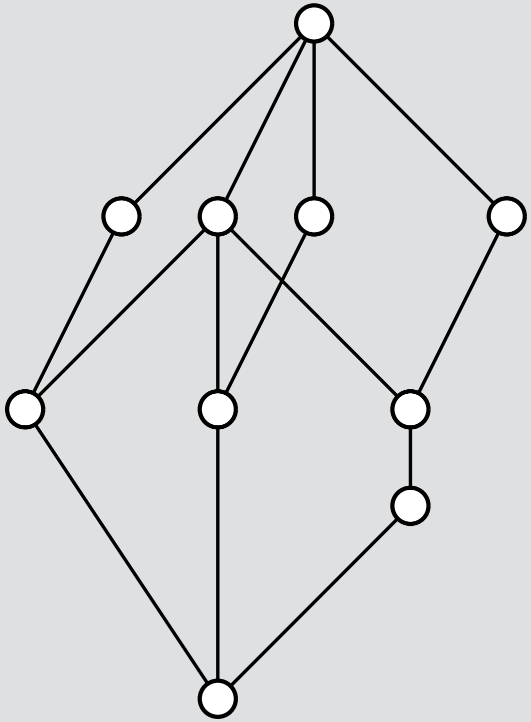
Drawing a line diagram

- Draw one circle for every formal concept.
- If $(A_1, B_1) < (A_2, B_2)$, a circle for (A_1, B_1) is positioned higher than that for (A_2, B_2).



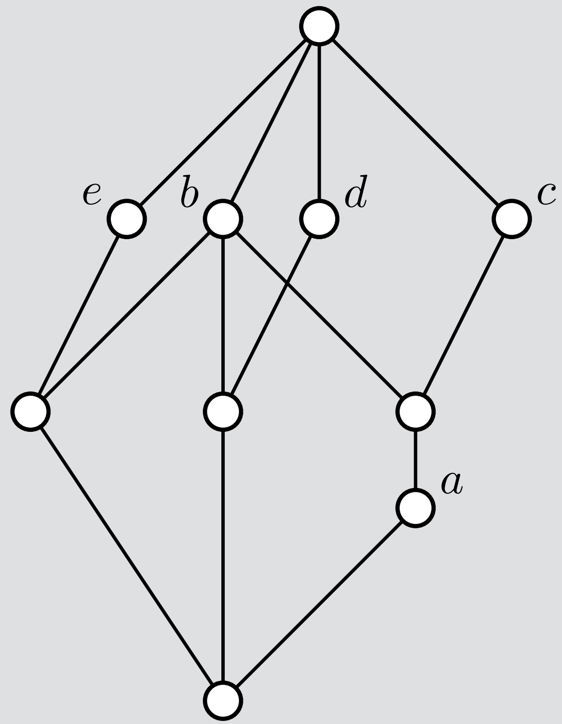
Drawing a line diagram

- Connect circles with their lower neighbors.



Drawing a line diagram

- Attach every attribute m to the circle representing the concept $(\{m\}', \{m\}'')$.



Drawing a line diagram

- Determine object concepts:

	Concept No.	(extent	,	intent)
	1	$(\{T_4\}$	,	$\{a, b, c\})$
	2	$(\{T_1, T_2, T_4, T_6\}$	,	$\{b\})$
	3	$(\{T_3, T_4, T_6\}$	,	$\{c\})$
	4	$(\{T_1, T_5\}$	,	$\{d\})$
	5	$(\{T_2, T_7\}$	,	$\{e\})$
	6	$(\emptyset$	,	$\{a, b, c, d, e\})$
	7	$(\{T_4, T_6\}$	,	$\{b, c\})$
T_1	8	$(\{T_1\}$	,	$\{b, d\})$
	9	$(\{T_2\}$	,	$\{b, e\})$
	10	$(\{T_1, T_2, T_3, T_4, T_5, T_6, T_7\}$	,	$\emptyset)$

Drawing a line diagram

- Determine object concepts:

	Concept No.	(extent	,	intent)
	1	$(\{T_4\}$	,	$\{a, b, c\})$
	2	$(\{T_1, T_2, T_4, T_6\}$	,	$\{b\})$
	3	$(\{T_3, T_4, T_6\}$	,	$\{c\})$
	4	$(\{T_1, T_5\}$	,	$\{d\})$
	5	$(\{T_2, T_7\}$	,	$\{e\})$
	6	$(\emptyset$	,	$\{a, b, c, d, e\})$
	7	$(\{T_4, T_6\}$	,	$\{b, c\})$
T_1	8	$(\{T_1\}$	,	$\{b, d\})$
T_2	9	$(\{T_2\}$	,	$\{b, e\})$
	10	$(\{T_1, T_2, T_3, T_4, T_5, T_6, T_7\}$	,	$\emptyset)$

Drawing a line diagram

- Determine object concepts:

	Concept No.	(extent	,	intent)
	1	$(\{T_4\}$	,	$\{a, b, c\})$
	2	$(\{T_1, T_2, T_4, T_6\}$	,	$\{b\})$
T_3	3	$(\{T_3, T_4, T_6\}$	,	$\{c\})$
	4	$(\{T_1, T_5\}$	,	$\{d\})$
	5	$(\{T_2, T_7\}$	,	$\{e\})$
	6	$(\emptyset$	,	$\{a, b, c, d, e\})$
	7	$(\{T_4, T_6\}$	,	$\{b, c\})$
T_1	8	$(\{T_1\}$	,	$\{b, d\})$
T_2	9	$(\{T_2\}$	,	$\{b, e\})$
	10	$(\{T_1, T_2, T_3, T_4, T_5, T_6, T_7\}$	,	$\emptyset)$

Drawing a line diagram

- Determine object concepts:

Concept No.		(extent ,	intent)
T_4	1	$(\{T_4\}$,	$\{a, b, c\})$
	2	$(\{T_1, T_2, T_4, T_6\}$,	$\{b\})$
T_3	3	$(\{T_3, T_4, T_6\}$,	$\{c\})$
	4	$(\{T_1, T_5\}$,	$\{d\})$
	5	$(\{T_2, T_7\}$,	$\{e\})$
	6	$(\emptyset$,	$\{a, b, c, d, e\})$
	7	$(\{T_4, T_6\}$,	$\{b, c\})$
T_1	8	$(\{T_1\}$,	$\{b, d\})$
T_2	9	$(\{T_2\}$,	$\{b, e\})$
	10	$(\{T_1, T_2, T_3, T_4, T_5, T_6, T_7\}$,	$\emptyset)$

Drawing a line diagram

- Determine object concepts:

	Concept No.	(extent	,	intent)
T_4	1	$(\{T_4\}$	,	$\{a,b,c\})$
	2	$(\{T_1,T_2,T_4,T_6\}$	,	$\{b\})$
T_3	3	$(\{T_3,T_4,T_6\}$	,	$\{c\})$
T_5	4	$(\{T_1,T_5\}$	,	$\{d\})$
	5	$(\{T_2,T_7\}$	,	$\{e\})$
	6	$(\emptyset$	,	$\{a,b,c,d,e\})$
	7	$(\{T_4,T_6\}$	,	$\{b,c\})$
T_1	8	$(\{T_1\}$	,	$\{b,d\})$
T_2	9	$(\{T_2\}$	,	$\{b,e\})$
	10	$(\{T_1,T_2,T_3,T_4,T_5,T_6,T_7\}$	,	$\emptyset)$

Drawing a line diagram

- Determine object concepts:

	Concept No.	(extent	,	intent)
T_4	1	$(\{T_4\}$	,	$\{a,b,c\})$
	2	$(\{T_1,T_2,T_4,T_6\}$	,	$\{b\})$
T_3	3	$(\{T_3,T_4,T_6\}$	,	$\{c\})$
T_5	4	$(\{T_1,T_5\}$	,	$\{d\})$
	5	$(\{T_2,T_7\}$	,	$\{e\})$
	6	$(\emptyset$	,	$\{a,b,c,d,e\})$
T_6	7	$(\{T_4,T_6\}$	,	$\{b,c\})$
T_1	8	$(\{T_1\}$	,	$\{b,d\})$
T_2	9	$(\{T_2\}$	,	$\{b,e\})$
	10	$(\{T_1,T_2,T_3,T_4,T_5,T_6,T_7\}$	,	$\emptyset)$

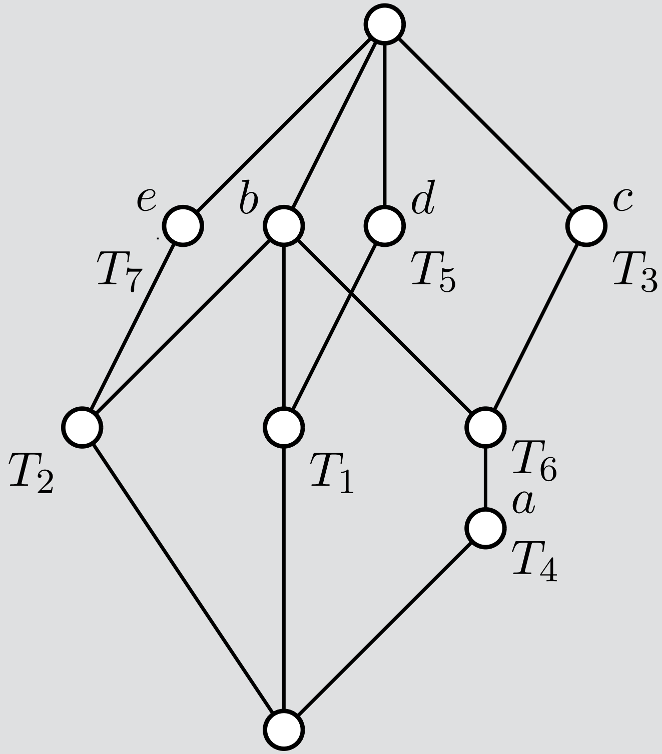
Drawing a line diagram

- Determine object concepts:

	Concept No.	(extent	,	intent)
T_4	1	$(\{T_4\}$	,	$\{a,b,c\})$
	2	$(\{T_1,T_2,T_4,T_6\}$	,	$\{b\})$
T_3	3	$(\{T_3,T_4,T_6\}$	,	$\{c\})$
T_5	4	$(\{T_1,T_5\}$	,	$\{d\})$
T_7	5	$(\{T_2,T_7\}$	,	$\{e\})$
	6	$(\emptyset$	,	$\{a,b,c,d,e\})$
T_6	7	$(\{T_4,T_6\}$	,	$\{b,c\})$
T_1	8	$(\{T_1\}$	,	$\{b,d\})$
T_2	9	$(\{T_2\}$	,	$\{b,e\})$
	10	$(\{T_1,T_2,T_3,T_4,T_5,T_6,T_7\}$	,	$\emptyset)$

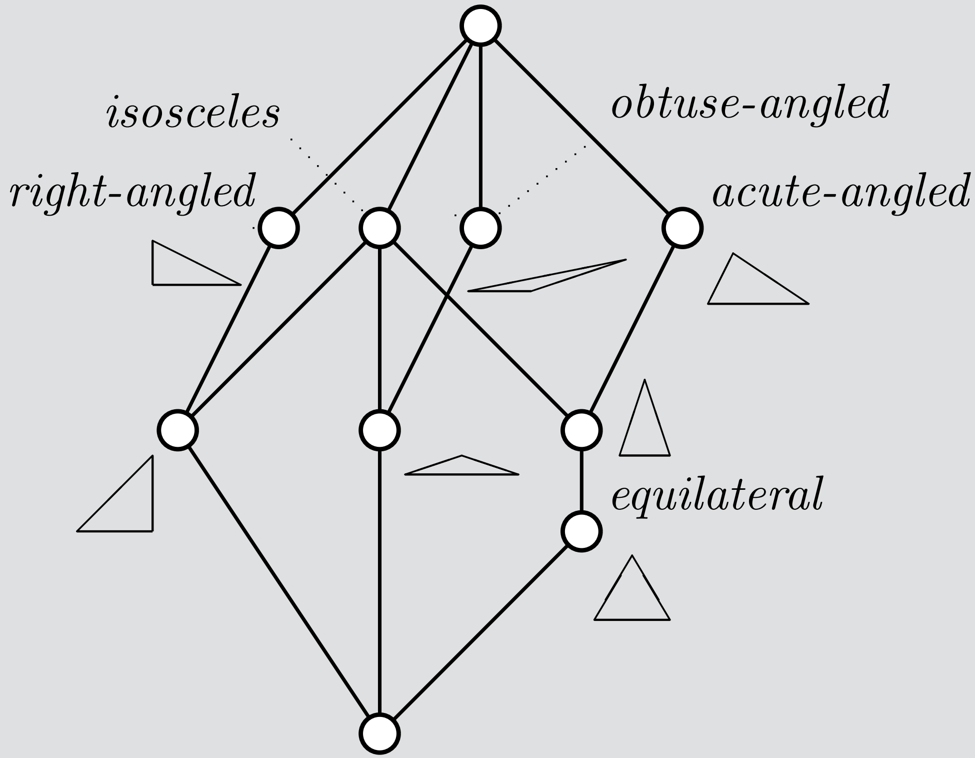
Drawing a line diagram

- Attach every object g to the circle representing the concept $(\{g\}'', \{g\}')$.



Drawing a line diagram

The result:



A naive algorithm for enumerating closed sets

Computing closed sets

Given a closure operator

$$A \mapsto A''$$

on *M*, compute all closed sets $A = A''$.

Naïve algorithm

for all $A \subseteq M$ **do**
compute A'' and check if the result is already listed

Must compute A'' for 2^n subsets $A \subseteq M$. $n = |M|$
For each computed subset, must check if it is in a list of size $O(2^n)$.

Representing sets by bit vectors

Representing sets by bit vectors

Let M be a finite linearly ordered set:

$$M = \{m_1 < m_2 < \ \dots \ < m_2\}$$

Every subset $S \subseteq M$ can be described by its characteristic vector

$\varepsilon_S\colon \{1, 2, \dots, k\} \to \{0,1\}$, given by

$$\varepsilon_s(i) := \begin{cases} 1 & \text{if } m_i \in S \\ 0 & \text{if } m_i \notin S \end{cases}$$

Example

$$M = \{a < b < c < d < e < f < g\}$$

The characteristic vector of $\{a, c, d, f\}$ is 1011010:

×	·	×	×	·	×	·

Lectic order

Let $A, B \subseteq M$. We say that A is lectically smaller than B and write $A < B$ if

$$\exists i\big(i \in B,\ i \notin A,\ \forall j < i(j \in A \Leftrightarrow j \in B)\big).$$

Example

$$\{a, c, e, f\} < \{a, c, d, f\}$$

×	·	×	·	×	×	·

$$\updownarrow$$

×	·	×	×	·	×	·

Lectic order

We write $A <_m B$ if

$$m \in B, \quad m \notin A, \quad \forall n < m(n \in A \Leftrightarrow n \in B).$$

Proposition. For $A, B \subseteq M$:

1. $A < B$ if and only if $A <_m B$ for some $m \in M$.
2. If $A <_m B$ and $A <_n C$ with $m < n$, then $C <_m B$.

Closures in lectic order

Computing closed sets

For $A \subseteq M$ and $m_i \in M$, define

$$A \oplus m_i := \big((A \cap \{m_1, \dots, m_{i-1}\}) \cup \{m_i\}\big)''.$$

For example, let $A = \{a, c, d, f\}$ and $m_i = e$.

$$\begin{array}{r c}
 & \begin{array}{ccccccc} & & & & \downarrow & & \end{array} \\
\{a, c, d, f\} = & \begin{array}{|c|c|c|c|c|c|c|} \hline \times & \cdot & \times & \times & \cdot & \times & \cdot \\ \hline \end{array} \\
 & \begin{array}{ccccccc} & & & & \downarrow & & \end{array} \\
\{a, c, d, f\} \cap \{a, b, c, d\} = & \begin{array}{|c|c|c|c|c|c|c|} \hline \times & \cdot & \times & \times & \cdot & \cdot & \cdot \\ \hline \end{array} \\
 & \begin{array}{ccccccc} & & & & \downarrow & & \end{array} \\
\{a, c, d, f\} \oplus e = & \left(\begin{array}{|c|c|c|c|c|c|c|} \hline \times & \cdot & \times & \times & \times & \cdot & \cdot \\ \hline \end{array}\right)''
\end{array}$$

$$A \oplus e = \{a, c, d, e\}''$$

Computing closed sets

Theorem

The smallest closed set larger than a given set $A \subset M$ with respect to the lectic order is

$$A \oplus m_i,$$

m_i being the largest element of M with $A <_{m_i} A \oplus m_i$.

Proof. Let B be the lectically smallest closed set after A.

- $A <_{m_i} B$ for some $m_i \in M$
- $(A \cap \{m_1, \dots, m_{i-1}\}) \cup \{m_i\} \subseteq B$
- $A < A \oplus m_i \leq B$
- $A \oplus m_i = B$
- If $m_j < m_i$, $A <_{m_i} A \oplus m_i$, and $A <_{m_j} A \oplus m_j$, then $A \oplus m_i < A \oplus m_j$ (by an earlier Proposition).

Computing closed sets

Algorithm 1 $\textsc{All Closures}(M, '')$: Generating all closed sets

Input: A closure operator $X \mapsto X''$ on a finite linearly ordered set M.
Output: All closed sets in lectic order.

$A := \textsc{First Closure}('')$ {see Algorithm 2}
while $A \neq \bot$ **do**
  output A
  $A := \textsc{Next Closure}(A, M, '')$ {see Algorithm 3}

Computing closed sets

Algorithm 1 ALL CLOSURES(M, $''$): Generating all closed sets

Input: A closure operator $X \mapsto X''$ on a finite linearly ordered set M.
Output: All closed sets in lectic order.

$A :=$ FIRST CLOSURE($''$) {see Algorithm 2}
while $A \neq \perp$ **do**
  output A
  $A :=$ NEXT CLOSURE(A, M, $''$) {see Algorithm 3}

Algorithm 2 FIRST CLOSURE($''$)

Input: A closure operator $X \mapsto X''$ on a finite linearly ordered set.
Output: The lectically first closed set, i.e., the closure of the empty set.

return $\emptyset''$

Computing closed sets

Algorithm 3 NEXT CLOSURE(A, M, $''$)

Input: A closure operator $X \mapsto X''$ on a finite linearly ordered set M and a subset $A \subseteq M$.
Output: The lectically next closed set after A if it exists; $\perp$, otherwise.

```
for all m ∈ M in reverse order do
    if m ∈ A then
        A := A \ {m}
    else
        B := (A ∪ {m})''
        if B \ A contains no element < m then
            return B
return ⊥
```

Computing the closure under implications

Closure(X, $\mathcal{L}$)

Input: An attribute set $X \subseteq M$ and a set $\mathcal{L}$ of implications over M.

Output: The closure of X w.r.t. implications in $\mathcal{L}$.

repeat
 $stable :=$ **true**
 for all $A \to B \in \mathcal{L}$ **do**
  if $A \subseteq X$ **then**
   $X := X \cup B$
   $stable :=$ **false**
   $\mathcal{L} := \mathcal{L} \setminus \{A \to B\}$
until $stable$
return X

The algorithm is quadratic in the number of implications in the worst case.

Worst case for CLOSURE(X, $\mathcal{L}$)

- $X = \{1\}$
- $\mathcal{L} = \{\{i\} \to \{i+1\} \mid i \in \mathbb{N}, 0 < i < n\}$ for some n

Example ($n = 5$)

$\mathcal{L} = [\{4\} \to \{5\}, \quad \{3\} \to \{4\}, \quad \{2\} \to \{3\}, \quad \{1\} \to \{2\}]$

After each iteration of the outer loop:

1. $X = \{1, 2\}$
2. $X = \{1, 2, 3\}$
3. $X = \{1, 2, 3, 4\}$
4. $X = \{1, 2, 3, 4, 5\}$
5. $X = \{1, 2, 3, 4, 5\}$

$n(n+1)/2$ iterations of the **for all** loop, each requiring $O(n)$ time

LinClosure(X, $\mathcal{L}$)

LinClosure is a linear-time algorithm for the same problem:

Input: An attribute set $X \subseteq M$ and a set $\mathcal{L}$ of implications over M.
Output: The closure of X w.r.t. implications in $\mathcal{L}$.

Works in time linear in the combined size of the implications in $\mathcal{L}$.

Beeri, C., Bernstein, P.:
Computational problems related to the design of normal form relational schemas. ACM TODS **4**(1), 30–59 (1979)