MDL for FCA: is there a place for background knowledge?

T. Makhalova[1,2] S. O. Kuznetsov[1] A. Napoli[2]

National Research University Higher School of Economics,
3 Kochnovsky Proezd, Moscow, Russia

LORIA, (CNRS – Inria – U. of Lorraine), BP 239
Vandœuvre-lès-Nancy, France

July 13, 2018

Outline

Motivation

- A wide range of application in Data Mining and Machine Learning.
- The exponential explosion of the number of concepts.

Concept Filtering. What Do We Want?

Requirements to the Filtering

- Interpretability. Why the concept has been selected?

Concept Filtering. What Do We Want?

Requirements to the Filtering

- Interpretability. Why the concept has been selected?
- Flexibility. Is it easy to compute a new subset with the adjusted requirements?

Concept Filtering. What Do We Want?

Requirements to the Filtering

- Interpretability. Why the concept has been selected?
- Flexibility. Is it easy to compute a new subset with the adjusted requirements?
- Low complexity. To get the result in an affordable time frames.

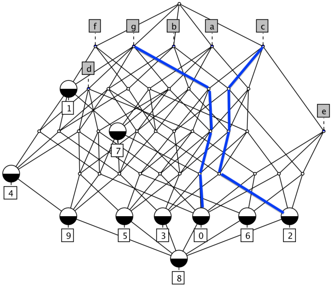
Concept Filtering. What Do We Want?

Requirements to the Filtering

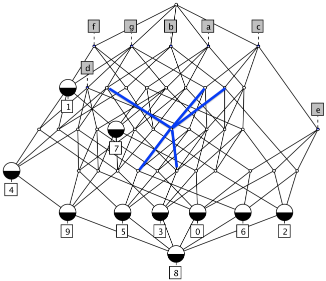
- Interpretability. Why the concept has been selected?
- Flexibility. Is it easy to compute a new subset with the adjusted requirements?
- Low complexity. To get the result in an affordable time frames.
- Background knowledge embedding. Is it easy to incorporate our assumption on interestingness?

Measure-Based Pattern Selection

- Meets requirements (e.g., well-separable, stable to noise, etc).
- Provides localized subsets of concepts:



long paths



concept neighborhoods

Formal Context and Its Coverings

	a	b	c	d	e
1	×	×	×		
2		×	×	×	×
3				×	×
4	×		×	×	×
5	×		×		

(1) A formal context

	a	b	c	d	e
1	×	×	×		
2		×	×	×	×
3				×	×
4	×		×	×	×
5	×		×		

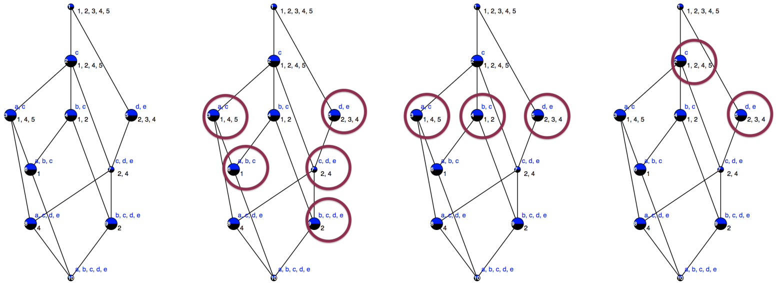
(2) Covering of objects with $\mathcal{S}_2 = \{\{abc\}, \{bcde\}, \{de\}, \{cde\}, \{ac\}\}$. RCR = 1.

	a	b	c	d	e
1	×	×	×		
2		×	×	×	×
3				×	×
4	×		×	×	×
5	×		×		

(3) Covering of objects with $\mathcal{S}_3 = \{\{bc\}, \{de\}, \{ac\}\}$. RCR = 1.

	a	b	c	d	e
1	×	×	×		
2		×	×	×	×
3				×	×
4	×		×	×	×
5	×		×		

(4) Covering of objects with $\mathcal{S}_4 = \{\{c\}, \{de\}\}$. RCR = 10/15.



Outline

Introduction. Pattern Mining Problem

Minimal Description Length. Basic Notions

MDL in Practice: Compression under Constraints

Experiments

Minimal Description Length. Basic Notions

The main principle: the best set of patterns is the set that best compresses the database [Vreeken et al., 2011].
Objective:

$$L(D, CT) = L(D \mid CT) + L(CT \mid D),$$

where $L(D \mid CT)$ is the length of the dataset encoded with the code table CT and $L(CT \mid D)$ is the length of the code table CT computed w.r.t. D.

MDL. Basic Notions

- **Code table**: a set of selected patterns with their encoding lengths.
- **Encoding length**: new length that "compresses", i.e. the most frequently used ones have the shortest encoding length.
- **Disjoint covering**: principle of compression by patterns.

MDL. Basic Notions

- **Code table**: a set of selected patterns with their encoding lengths.
- **Encoding length**: new length that "compresses", i.e. the most frequently used ones have the shortest encoding length.
- **Disjoint covering**: principle of compression by patterns.

✓	$m_1 m_2 m_3 m_4 m_5$ $m_1 m_2 m_3$ $m_4 m_5$
✗	$m_1 m_2 m_3 m_4 m_5$ $m_1 m_2 m_3$ $m_3 m_4 m_5$
✗	$m_1 m_2 m_3 m_4 m_5$ $m_1 m_2 m_3$ $m_3 m_4$

MDL. Basic Notions

Example

CT i	len	usage (freq.)
m_3		4
m_1		3
m_2		2
m_4		1
m_6		1
m_7		1
m_8		1
m_9		1
m_5		

Data with covering	Encoded data
$(m_1)(m_2)(m_3)(m_6)$	, , ,
$(m_1)(m_2)(m_3)(m_7)$	, , ,
$(m_1)(m_3)(m_8)$	, ,
$(m_3)(m_4)(m_9)$	, ,

$$L(D, CT) = L(CT \mid D) + L(D \mid CT);$$

$$L(CT \mid D) = \sum_{i \in CT} code(i) + len(i); \qquad L(D \mid CT) = \sum_{d \in D} \sum_{i \in cover(d)} len(i)$$

MDL. Unsupervised Settings

Main Steps

- Compute ordered candidate set.

Example. All frequent patterns sorted by length, frequency

Step 0			
CT		**Data with covering**	**Candidate set, area**
i	**u**		
m_3	4	$(m_1)(m_2)(m_3)(m_6)$	$m_1 m_2 m_3, 6$
m_1	3	$(m_1)(m_2)(m_3)(m_7)$	$m_1 m_3, 6$
m_2	2	$(m_1)(m_3)(m_8)$	$m_1 m_2 m_3 m_6, 4$
m_4	1	$(m_3)(m_4)(m_9)$	$m_1 m_2 m_3 m_7, 4$
m_6-m_9	1		$m_1 m_3 m_8, 3$
m_5	0		$m_3 m_4 m_9, 3$

MDL. Unsupervised Settings

Main Steps

- Compute ordered candidate set.
- Cover greedily the given data.

Example. All frequent patterns sorted by length, frequency

Step 0			
CT		**Data with covering**	**Candidate set, area**
i	**u**		
m_3	4	$(m_1)(m_2)(m_3)(m_6)$	$m_1 m_2 m_3, 6$
m_1	3	$(m_1)(m_2)(m_3)(m_7)$	$m_1 m_3, 6$
m_2	2	$(m_1)(m_3)(m_8)$	$m_1 m_2 m_3 m_6, 4$
m_4	1	$(m_3)(m_4)(m_9)$	$m_1 m_2 m_3 m_7, 4$
m_6-m_9	1		$m_1 m_3 m_8, 3$
m_5	0		$m_3 m_4 m_9, 3$

MDL. Unsupervised Settings

Main Steps

- Compute ordered candidate set.
- Cover greedily the given data.

Example. Try to cover by disjoint patterns:

Step 1			
CT		**Data with covering**	**Candidate set, area**
i	**u**		
$m_1 m_2 m_3$	2	$(m_1 m_2 m_3)(m_6)$	$m_1 m_3 m_8, 3$
m_3	2	$(m_1 m_2 m_3)(m_7)$	$m_3 m_4 m_9, 3$
m_1, m_4	1	$(m_1)(m_3)(m_8)$	$m_1 m_3, 2$
m_6-m_9	1	$(m_3)(m_4)(m_9)$	
m_2, m_5	0		

MDL. Unsupervised Settings

Main Steps

- Compute ordered candidate set.
- Cover greedily the given data.

Example. Try to cover by disjoint patterns:

Step 1			
CT		**Data with covering**	**Candidate set, area**
i	**u**		
$m_1 m_2 m_3$	2	$(m_1 m_2 m_3)(m_6)$	$m_1 m_3 m_8, 3$
m_3	2	$(m_1 m_2 m_3)(m_7)$	$m_3 m_4 m_9, 3$
m_1, m_4	1	$(m_1)(m_3)(m_8)$	$m_1 m_3, 2$
m_6-m_9	1	$(m_3)(m_4)(m_9)$	
m_2, m_5	0		

MDL. Unsupervised Settings

Main Steps

- Compute ordered candidate set.
- Cover greedily the given data.

Example. Try to cover by disjoint patterns:

Step 2			
CT		**Data with covering**	**Candidate set, area**
i	**u**		
$m_1 m_2 m_3$	2	$(m_1 m_2 m_3)(m_6)$	$m_3 m_4 m_9$, 3
$m_1 m_3 m_8$	1	$(m_1 m_2 m_3)(m_7)$	
m_3, m_4	1	$(m_1 m_3 m_8)$	
m_1, m_2, m_5, m_8	0	$(m_3)(m_4)(m_9)$	

MDL. Unsupervised Settings

Main Steps

- Compute ordered candidate set.
- Cover greedily the given data.

Example. Try to cover by disjoint patterns:

Step 2			
CT		**Data with covering**	**Candidate set, area**
i	**u**		
$m_1 m_2 m_3$	2	$(m_1 m_2 m_3)(m_6)$	$m_3 m_4 m_9$, 3
$m_1 m_3 m_8$	1	$(m_1 m_2 m_3)(m_7)$	
m_3, m_4	1	$(m_1 m_3 m_8)$	
m_1, m_2, m_5, m_8	0	$(m_3)(m_4)(m_9)$	

MDL. Unsupervised Settings

Main Steps

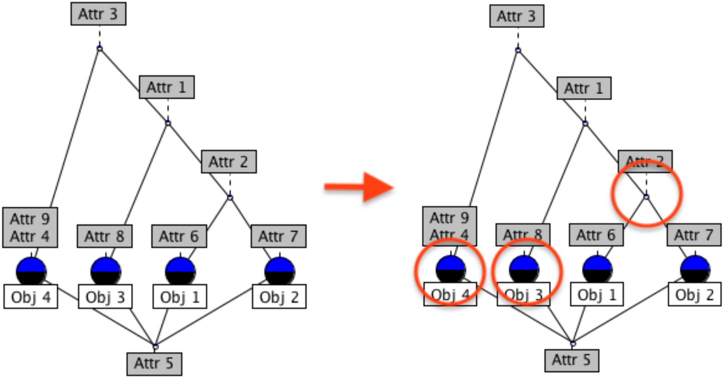
- Compute ordered candidate set.
- Cover greedily the given data.

Example. Try to cover by disjoint patterns:

Step 3			
CT		**Data with covering**	**Candidate set, area**
i	**u**		
$m_1 m_2 m_3$	2	$(m_1 m_2 m_3)(m_6)$	
$m_1 m_3 m_8$	1	$(m_1 m_2 m_3)(m_7)$	
$m_3 m_4 m_9$	1	$(m_1 m_3 m_8)$	
m_6, m_7	1	$(m_3 m_4 m_9)$	
m_1-m_5	0		
m_8-m_9	0		

MDL. Unsupervised Settings

From the candidate set to the code table



MDL in Practice: Compression under Constraints

MDL:

- threshold-free selection;
- variable patterns.

Measure-based selection:

- background knowledge (constraints) embedding.

MDL in Practice: Compression under Constraints

Proposed Approach: MDL Perspective

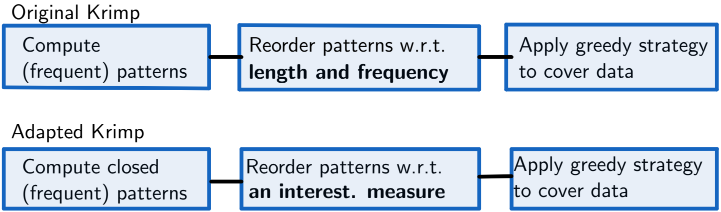


Figure: The workflow for pattern mining by the original Krimp and its adapted version.

MDL in Practice: Compression under Constraints

Proposed Approach: Measure-Based Selection Perspective

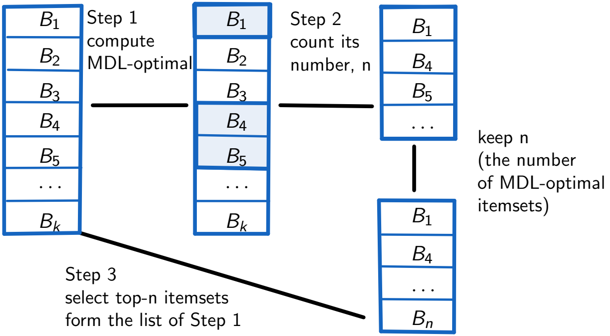


Figure: The principle of computing MDL-optimal and top-n sets of patterns

Compression

Reduction in The Number of Concepts

dataset	nmb. of obj.	nmb. of attr.	nmb. of concepts	Number of MDL-optimal					
				area fr_lift	area len_fr	area len_lift	len fr	len lift	lift len
breast	699	16	702	36.0	32.2	20.4	37.3	37.3	33.5
car	1 728	25	12 420	868.4	849.2	138.6	714.6	847.7	698.3
ecoli	336	29	690	58.8	55.9	16.4	64.9	65.6	55.9
iris	150	19	183	31.1	28.9	12.9	34.8	34.6	26.3
led7	3 200	24	3 808	108.0	118.3	64.2	108.7	108.7	130.3
pima	768	38	2 769	110.1	106.3	35.9	120.6	112.1	101.7

Non-redundancy

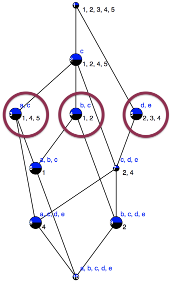
Distance to the 1st nearest neighbor

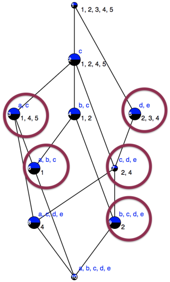
$\mathcal{S}_3$	binary representation (abcde)	nearest neighbor	Euclidean distance
bc	01100	ac	$\sqrt{2}$
de	00011	bc(ac)	2
ac	10100	bc	$\sqrt{2}$

The average distance is $(2 + 2\sqrt{2})/3$.

$\mathcal{S}_2$	binary representation (abcde)	nearest neighbor	Euclidean distance
bcde	01111	cde	$\sqrt{2}$
cde	00111	bcde (de)	$\sqrt{2}$
abc	11100	ac	$\sqrt{2}$
ac	10100	abc	$\sqrt{2}$
de	00011	cde	$\sqrt{2}$

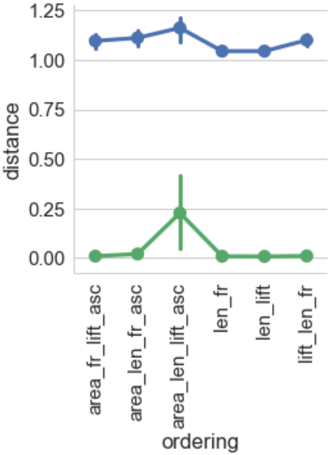
The average distance is $\sqrt{2}$.





Euclidean distances to the 1st nearest neighbors. The average distance for $\mathcal{S}_3$ is longer then for $\mathcal{S}_2$, thus $\mathcal{S}_3$ contains more diverse patterns.

Non-redundancy

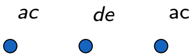


Distance to the 1NN

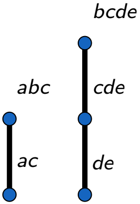
On the X-axis is different orderings of patterns, on the Y-axis is the values of the listed above non-redundancy parameters for MDL-optimal set (blue) and top-n (green) set of the same size.

Non-redundancy

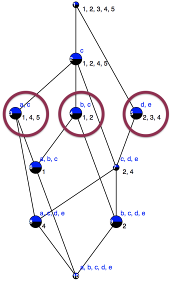
Average length of the longest paths built from partially ordered itemsets

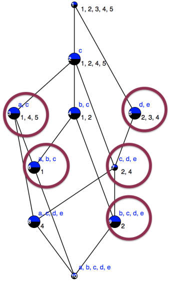


The average length
of the longest paths is 1.



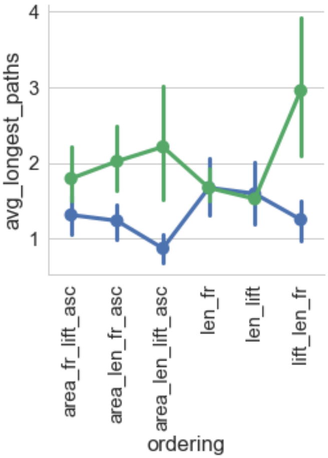
The average length
of the longest paths is 2.5.





The longest paths built on partially ordered patterns (by inclusion).

Non-redundancy

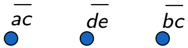


The average path lengths

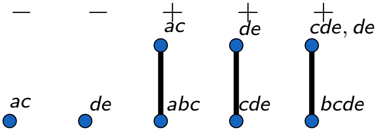
On the X-axis is different orderings of patterns, on the Y-axis is the values of the listed above non-redundancy parameters for MDL-optimal set (blue) and top-n (green) set of the same size.

Non-redundancy

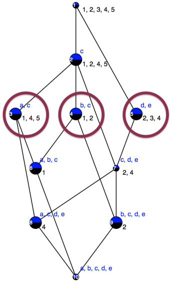
Average number of itemsets with parents (more general itemsets)

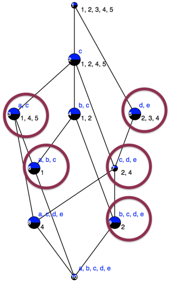


The rate of pattern with children is 0.

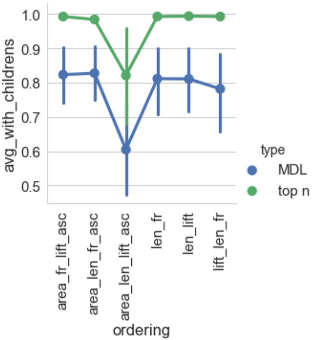


The rate of pattern with children is 3/5.





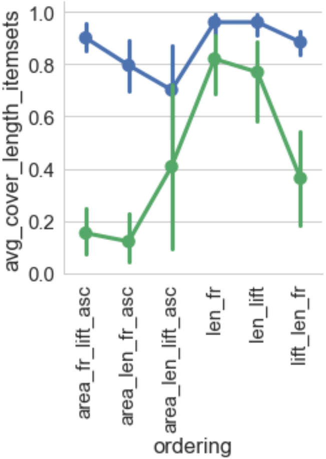
Non-redundancy



Rate of children with parents.

On the X-axis is different orderings of patterns, on the Y-axis is the values of the listed above non-redundancy parameters for MDL-optimal set (blue) and top-n (green) set of the same size.

Data Coverage



the average rate of crosses covered by patterns

On the X-axis is different orderings of patterns, on the Y-axis is the values of the listed above non-redundancy parameters for MDL-optimal set (blue) and top-n (green) set of the same size.

Conclusion

A new approach *"implementation of the MDL principle under constrains"* or *"embedding of background knowledge (on interestingness) into MDL"* has been proposed.

The approach:

- threshold-free;
- allows for selection of a small set of patterns having desired properties;
- patterns are diverse and varied, they cover almost all attributes of objects.

Thank you for your attention.