

Constraint-Based Mining of Closed Patterns in Noisy n -ary Relations

Loïc Cerf

Advisor: Pr. Jean-François Boulicaut

Université de Lyon, CNRS
INSA-Lyon, LIRIS, UMR 5205, F-69621, France


Context

From data to knowledge through patterns.

Motivating example: Vélo'v usage analysis



©2007 ebarrera and pug freak (from flickr®)

 These pictures are licensed under the Creative Commons Attribution Noncommercial ShareAlike 2.0 License.

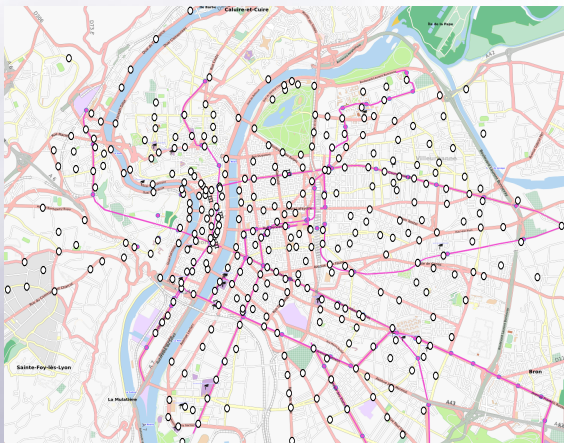
Example of data

More than 13.1 million rides

dep	arr	timestamp	...
1001	1001	2005-05-24 12:37:22	...
4005	4005	2005-05-25 09:06:00	...
⋮	⋮	⋮	⋮
6023	7034	2005-12-18 08:10:51	...
3038	2013	2005-12-18 08:14:38	...
⋮	⋮	⋮	⋮
10055	10012	2007-12-18 08:08:16	...
10080	10012	2007-12-18 08:09:37	...

Example of a relevant *local* pattern

During the week-ends,
from 3pm to 8pm:

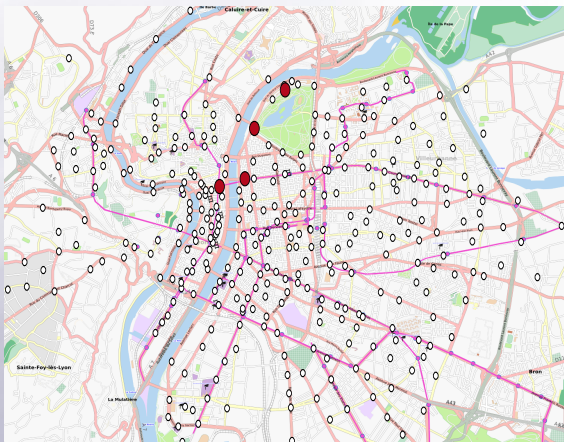


©2004-2010 OpenStreetMap contributors (created from OpenStreetMap project data)

 This map is licensed under Creative Commons Attribution ShareAlike 2.0 License.

Example of a relevant *local* pattern

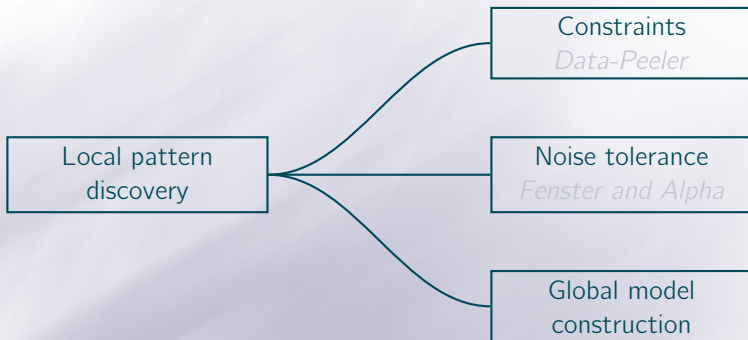
During the week-ends,
from 3pm to 8pm:



©2004-2010 OpenStreetMap contributors (created from OpenStreetMap project data)

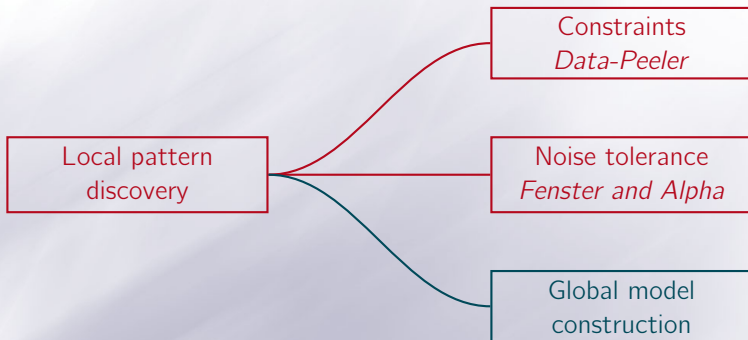
 This map is licensed under Creative Commons Attribution ShareAlike 2.0 License.

Data-Mining@LIRIS



Applications: Biological data analysis, dynamic graph analysis, etc.

Data-Mining@LIRIS



Applications: Biological data analysis, **dynamic graph analysis**, etc.

From binary relations...

Binary relations encode whether objects have properties.

A table can represent a binary relation with objects as row labels and properties as column labels:

	<i>A</i>	<i>B</i>	<i>C</i>
1	●	●	●
2	●		
3			●
4	●		●

Example of application

Basket analysis (relation customers \times items).

From binary relations...

Binary relations encode whether objects have properties.

A table can represent a binary relation with **objects** as row labels and properties as column labels:

	<i>A</i>	<i>B</i>	<i>C</i>
1	●	●	●
2	●		
3			●
4	●		●

Example of application

Basket analysis (relation **customers** \times **items**).

From binary relations...

Binary relations encode whether objects have properties.

A table can represent a binary relation with objects as row labels and **properties** as column labels:

	<i>A</i>	<i>B</i>	<i>C</i>
1	●	●	●
2	●		
3			●
4	●		●

Example of application

Basket analysis (relation customers \times **items**).

From binary relations...

Binary relations encode whether objects have properties.

A table can represent a binary relation with objects as row labels and properties as column labels:

	<i>A</i>	<i>B</i>	<i>C</i>
1	●	●	●
2	●		
3			●
4	●		●








Example of application

Basket analysis (relation customers \times items).

From binary relations...

Binary relations encode whether objects have properties.

A table can represent a binary relation with objects as row labels and properties as column labels:

	A	B	C
1			
2			
3			
4			

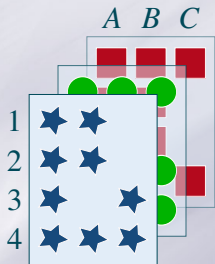
$$\mathcal{O} = \{1, 2, 3, 4\}$$

$$\mathcal{P} = \{A, B, C\}$$

$$\mathcal{R} = \{(1, A), (1, B), (1, C), (2, A), (3, C), (4, A), (4, C)\}$$

... to n -ary relations

N -ary relations allow to consider n dimensions of analysis ($n \geq 2$).

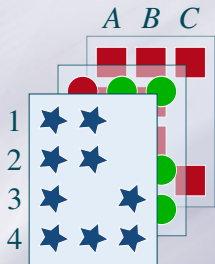


Example of application

Per-supermarket basket analysis (relation customers \times items \times supermarkets).

... to n -ary relations

N -ary relations allow to consider n dimensions of analysis ($n \geq 2$).



$$\mathcal{D}^1 = \{1, 2, 3, 4\}$$

$$\mathcal{D}^2 = \{A, B, C\}$$

$$\mathcal{D}^3 = \{\blacksquare, \bullet, \star\}$$

$$\mathcal{R} = \{(1, A, \blacksquare), (1, A, \bullet), (1, A, \star), (1, B, \blacksquare), (1, B, \bullet), (1, B, \star), (1, C, \blacksquare), (1, C, \bullet), (2, A, \blacksquare), (2, A, \bullet), (2, A, \star), (2, B, \blacksquare), (2, B, \star), (3, A, \star), (3, B, \blacksquare), (4, A, \bullet), (4, A, \star), (4, B, \star), (4, C, \blacksquare), (4, C, \bullet), (4, C, \star)\}$$

Mining local patterns in relations

Task

Discover relevant subspaces of the relation.

Declarative issue Defining what is a *relevant* local pattern;

Procedural issue Discovering them **all** in *huge* relations.

Mining local patterns in relations

Task

Discover relevant subspaces of the relation.

$$\{\phi \in L \mid \mathcal{C}_1(\phi, \mathcal{R}) \wedge \mathcal{C}_2(\phi, \mathcal{R}) \wedge \dots\}$$

where:

- ≡ L is the pattern space,
- ≡ \mathcal{R} is the relation,
- ≡ \mathcal{C}_i is a primitive constraint.

Outline

- 1 Closed n -set
- 2 Enumeration
- 3 Constraints
- 4 Dynamic graph mining: a case study
- 5 Noise tolerance
- 6 More noise tolerance
- 7 Conclusion

$$\{\phi \in L \mid \mathcal{C}_{\text{connected}}(\phi, \mathcal{R}) \wedge \mathcal{C}_{\text{closed}}(\phi, \mathcal{R}) \wedge \dots\}$$

Closed 2-set: purpose

Task

From a binary relation, extract every *maximal* set of objects sharing the same *maximal* set of properties.

	A	B	C
1	●	●	●
2	●		
3			●
4	●		●

$$\mathcal{O} = \{1, 2, 3, 4\}$$

$$\mathcal{P} = \{A, B, C\}$$

$$\mathcal{R} = \{(1, A), (1, B), (1, C), (2, A), (3, C), (4, A), (4, C)\}$$

Closed 2-set: purpose

Task

From a binary relation, extract every *maximal* set of objects sharing the same *maximal* set of properties.

	A	B	C
1	●		●
2			
3			
4	●		●

$(\{1, 4\}, \{A, C\})$ is a closed 2-set.

Closed 2-set: purpose

Task

From a binary relation, extract every *maximal* set of objects sharing the same *maximal* set of properties.

	A	B	C
1	●	●	●
2	●		
3			●
4	●		●

$(\{1, 4\}, \{A, C\})$ is a closed 2-set.

Closed 2-set: purpose

Task

From a binary relation, extract every *maximal* set of objects sharing the same *maximal* set of properties.

	A	B	C
1	●		●
2			
3			
4	●		●

Example of application

Basket analysis (relation customers \times items).

Closed 2-set: definition

Let \mathcal{O} a finite set of objects, \mathcal{P} a finite set of properties and $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{P}$ a binary relation.

Definition

$\forall O \subseteq \mathcal{O}, \forall P \subseteq \mathcal{P}$, (O, P) is a closed 2-set if and only if:

$$\equiv \mathcal{C}_{\text{connected}}(O, P) \equiv O \times P \subseteq \mathcal{R}$$

$$\equiv \mathcal{C}_{\text{closed}}(O, P) \equiv \begin{cases} \forall o \in \mathcal{O} \setminus O, \neg \mathcal{C}_{\text{connected}}(O \cup \{o\}, P) \\ \forall p \in \mathcal{P} \setminus P, \neg \mathcal{C}_{\text{connected}}(O, P \cup \{p\}) \end{cases}$$

Alternative names: formal concepts, maximal bi-cliques, closed itemsets associated with their supporting sets.

Closed n -set: definition

Let $(\mathcal{D}^i)_{i=1..n}$ n finite sets and $\mathcal{R} \subseteq \times_{i=1..n} \mathcal{D}^i$ an n -ary relation.

Definition

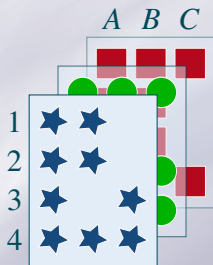
$\forall (X^1, \dots, X^n) \subseteq \times_{i=1..n} \mathcal{D}^i$, (X^1, \dots, X^n) is a closed n -set if and only if:

- ≡ $\mathcal{C}_{\text{connected}}(X^1, \dots, X^n) \equiv \times_{i=1..n} X^i \subseteq \mathcal{R}$
- ≡ $\mathcal{C}_{\text{closed}}(X^1, \dots, X^n) \equiv \forall i = 1..n, \forall x \in \mathcal{D}^i \setminus X^i,$
 $\neg \mathcal{C}_{\text{connected}}(X^1, \dots, X^i \cup \{x\}, \dots, X^n)$

Closed n -set: example

Task

From an n -ary relation, extract every n -tuple binding n sets, *maximal* for each of the attributes.



$$\mathcal{D}^1 = \{1, 2, 3, 4\}$$

$$\mathcal{D}^2 = \{A, B, C\}$$












$$\mathcal{D}^3 = \{\blacksquare, \bullet, \star\}$$

$$\mathcal{R} = \{(1, A, \blacksquare), (1, A, \bullet), (1, A, \star), (1, B, \blacksquare), (1, B, \bullet), (1, B, \star), (1, C, \blacksquare), (1, C, \bullet), (2, A, \blacksquare), (2, A, \bullet), (2, A, \star), (2, B, \blacksquare), (2, B, \star), (3, A, \star), (3, B, \blacksquare), (4, A, \bullet), (4, A, \star), (4, B, \star), (4, C, \blacksquare), (4, C, \bullet), (4, C, \star)\}$$

Closed n -set: example

Task

From an n -ary relation, extract every n -tuple binding n sets, *maximal* for each of the attributes.

	A	B	C
1			
2			
3			
4			

$$\mathcal{D}^1 = \{1, 2, 3, 4\}$$

$$\mathcal{D}^2 = \{A, B, C\}$$




$$\mathcal{D}^3 = \{\blacksquare, \bullet, \star\}$$

$$\mathcal{R} = \{(1, A, \blacksquare), (1, A, \bullet), (1, A, \star), (1, B, \blacksquare), (1, B, \bullet), (1, B, \star), (1, C, \blacksquare), (1, C, \bullet), (2, A, \blacksquare), (2, A, \bullet), (2, A, \star), (2, B, \blacksquare), (2, B, \star), (3, A, \star), (3, B, \blacksquare), (4, A, \bullet), (4, A, \star), (4, B, \star), (4, C, \blacksquare), (4, C, \bullet), (4, C, \star)\}$$

Closed n -set: example

Task

From an n -ary relation, extract every n -tuple binding n sets, *maximal* for each of the attributes.












	A	B	C
1			
2			
3			
4			

$(\{1\}, \{A, B, C\}, \{\text{red square}, \text{green circle}\})$ is a closed 3-set.

Closed n -set: example

Task

From an n -ary relation, extract every n -tuple binding n sets, *maximal* for each of the attributes.

	A	B	C
1			
2			
3			
4			

$(\{1\}, \{A, B, C\}, \{\text{■}, \text{●}\})$ is a closed 3-set.

Closed n -set: example

Task

From an n -ary relation, extract every n -tuple binding n sets, *maximal* for each of the attributes.

	A	B	C
1	■	■	■
2			
3			
4			

Example of application

Per-supermarket basket analysis (relation customers \times items \times supermarkets).

Outline

- 1 Closed n -set
- 2 Enumeration
- 3 Constraints
- 4 Dynamic graph mining: a case study
- 5 Noise tolerance
- 6 More noise tolerance
- 7 Conclusion

$$\{\phi \in L \mid \mathcal{C}_{\text{connected}}(\phi, \mathcal{R}) \wedge \mathcal{C}_{\text{closed}}(\phi, \mathcal{R}) \wedge \dots \}$$

Enumeration: binary case in, e. g., DMINER (2005)

With a binary relation, **choose one of the two attributes**, then, at every iteration:

1. choose one of its elements,
2. append it (or not) to those previously enumerated,
3. reduce the subset of the other attribute domain (*support*),
4. check that the previously refused elements cannot extend the pattern.

Galois connection property

Given any closed 2-set, its subset of **one** attribute domain is enough to deduce its subset of the other attribute domain.

Galois connection in practice

	A	B	C
1	●	●	●
2	●		
3			●
4	●		●

$(\{1, 4\}, \{A, C\})$ is a closed 2-set.



Galois connection in practice

	A	B	C
1	●		●
2			
3			
4	●		●

$(\{1, 4\}, \{A, C\})$ is a closed 2-set.

$\{AC\}$

$\{14\}$

Galois connection in practice

	A	B	C
1	●		●
2			
3			
4	●		●

$(\{1, 4\}, \{A, C\})$ is a closed 2-set.



Galois connection in practice

	A	B	C
1	●	●	●
2	●		
3			●
4	●		●

$(\{1, 4\}, \{A, C\})$ is a closed 2-set.



Galois connection in practice

	A	B	C
1	●	●	●
2	●		
3			●
4	●		●

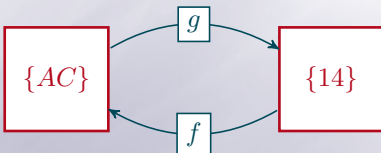
$(\{1, 4\}, \{A, C\})$ is a closed 2-set.














Galois connection in practice

	A	B	C
1	●	●	●
2	●		
3			●
4	●		●

$(\{1, 4\}, \{A, C\})$ is a closed 2-set.














Absence of a Galois connection in n -ary relations

	A	B	C
1			
2			
3			
4			

Absence of property

Given a closed n -set, its subset of one attribute domain generally is *not* enough to deduce its subsets of the other attribute domains.

Absence of a Galois connection in n -ary relations

	A	B	C
1			
2			
3			
4			

$(\{1, 4\}, \{A, C\}, \{\bullet\})$ is a closed 3-set.

Absence of a Galois connection in n -ary relations

	A	B	C
1	●		●
2			
3			
4	●		●












$(\{1, 4\}, \{A, C\}, \{\bullet\})$ is a closed 3-set.

Absence of a Galois connection in n -ary relations

	A	B	C
1	●		●
2			
3			
4	●		●

$(\{1, 4\}, \{A, C\}, \{\bullet\})$ is a closed 3-set.

Absence of a Galois connection in n -ary relations

	A	B	C
1			
2			
3			
4			












$(\{1, 4\}, \{A, C\}, \{\bullet\})$ is a closed 3-set.

Absence of a Galois connection in n -ary relations

	A	B	C
1			
2			
3	★		★
4	★		★












$(\{3, 4\}, \{A, C\}, \{\star\})$ is another closed 3-set.

Absence of a Galois connection in n -ary relations

	A	B	C
1			
2			
3			
4			

$(\{3, 4\}, \{A, C\}, \{\star\})$ is another closed 3-set.

Absence of a Galois connection in n -ary relations

	A	B	C
1			
2			
3			
4			

$(\{1, 4\}, \{A, C\}, \{\bullet\})$ is a closed 3-set.

$$\equiv \{1, 4\}, \{A, C\} \rightarrow \{\bullet\}$$

$$\equiv \{1, 4\}, \{\bullet\} \rightarrow \{A, C\}$$

$$\equiv \{A, C\}, \{\bullet\} \rightarrow \{1, 4\}$$

N -ary relation and Galois connection

With an n -ary relation, although no Galois connection exists, the closed n -sets can be made up via n functions:

- ≡ from $2^{\cup_{i \neq 1} \mathcal{D}^i}$ to $2^{\mathcal{D}^1}$,
- ≡ from $2^{\cup_{i \neq 2} \mathcal{D}^i}$ to $2^{\mathcal{D}^2}$,
- ≡ ...
- ≡ from $2^{\cup_{i \neq n} \mathcal{D}^i}$ to $2^{\mathcal{D}^n}$.

The elements of $n - 1$ attributes must be enumerated so that the search space on the remaining attribute is reduced.

Enumeration: general case

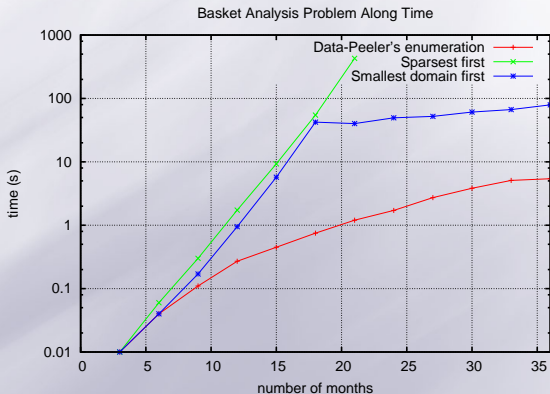
With an n -ary relation, at every iteration:

1. choose an element in the domain of **one of the n attributes**,
2. append it (or not) to those previously enumerated,
3. relating this element to those previously enumerated in $n - 2$ other attributes, reduce the search space on the remaining attribute,
4. check that the previously refused elements cannot extend the pattern.

This algorithm, DATA-PEELER (2008), is efficient if a good choice is made at Step 1 (goal: maximizing the search space reduction at Step 3).

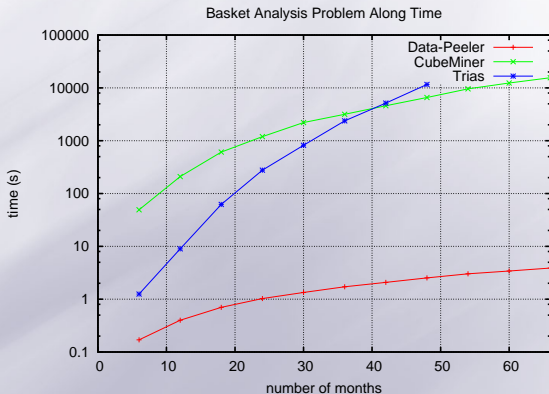
Enumeration: choice of the element

On synthetic *ternary* relations (generated by QUEST):



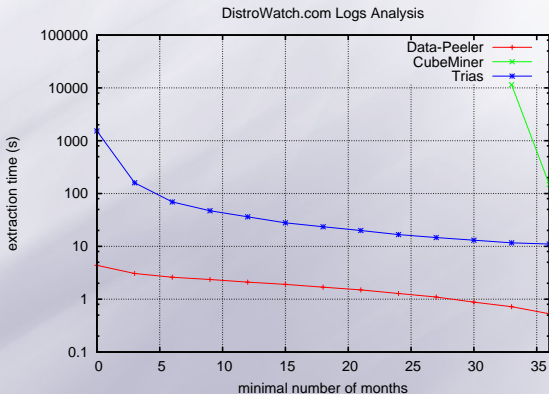
Comparison with CUBEMINER (2006) and TRIAS (2006)

On synthetic *ternary* relations (generated by QUEST):



Comparison with CUBEMINER (2006) and TRIAS (2006)

On a real-life *ternary* relation (derived from DistroWatch.com logs):



Outline

- 1 Closed n -set
- 2 Enumeration
- 3 Constraints**
- 4 Dynamic graph mining: a case study
- 5 Noise tolerance
- 6 More noise tolerance
- 7 Conclusion

$$\{\phi \in L \mid \mathcal{C}_{\text{connected}}(\phi, \mathcal{R}) \wedge \mathcal{C}_{\text{closed}}(\phi, \mathcal{R}) \wedge \dots\}$$

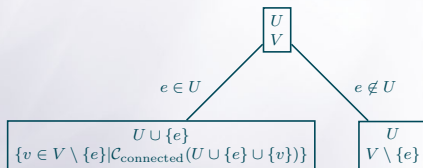
Constraints: why?

- Quality** Enhance the relevance of the extracted closed n -sets,
Efficiency Guide the search of these closed n -sets and make their extraction tractable in huge datasets.

$$\{\phi \in L \mid \mathcal{C}_{\text{connected}}(\phi, \mathcal{R}) \wedge \mathcal{C}_{\text{closed}}(\phi, \mathcal{R}) \wedge \dots\}$$

Constraints: from where?

The class of constraints
DATA-PEELER efficiently handles
comes from its enumeration
principles.



At every iteration (enumeration node), we can access:

- U the smallest closed n -set that may be recursively discovered from this enumeration node.
- $U \cup V$ the largest closed n -set that may be recursively discovered from this enumeration node.

Constraints: how?

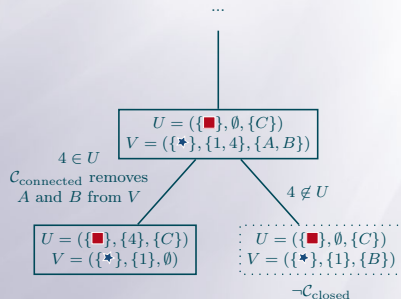
$$\{\phi \in L \mid \mathcal{C}_{\text{connected}}(\phi, \mathcal{R}) \wedge \mathcal{C}_{\text{closed}}(\phi, \mathcal{R}) \wedge \mathcal{C}_{5\text{-volume}}(\phi, \mathcal{R})\}$$

$$\equiv \begin{array}{l} \mathcal{C}_{5\text{-volume}}(X^1, X^2, X^3) \\ |X^1 \times X^2 \times X^3| \geq 5 \end{array}$$

Constraints: how?

$$\{\phi \in L \mid \mathcal{C}_{\text{connected}}(\phi, \mathcal{R}) \wedge \mathcal{C}_{\text{closed}}(\phi, \mathcal{R}) \wedge \mathcal{C}_{5\text{-volume}}(\phi, \mathcal{R})\}$$

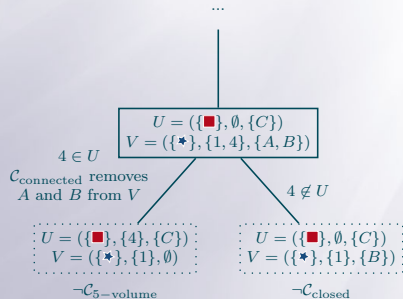
$$\begin{aligned} & \mathcal{C}_{5\text{-volume}}(X^1, X^2, X^3) \\ \equiv & |X^1 \times X^2 \times X^3| \geq 5 \end{aligned}$$



Constraints: how?

$$\{\phi \in L \mid \mathcal{C}_{\text{connected}}(\phi, \mathcal{R}) \wedge \mathcal{C}_{\text{closed}}(\phi, \mathcal{R}) \wedge \mathcal{C}_{5\text{-volume}}(\phi, \mathcal{R})\}$$

$$\begin{aligned} & \mathcal{C}_{5\text{-volume}}(X^1, X^2, X^3) \\ \equiv & |X^1 \times X^2 \times X^3| \geq 5 \end{aligned}$$



Constraints: which ones?

Monotone Does $U \cup V$ satisfy it?

Anti-monotone Does U satisfy it?

Piecewise (anti)-monotone Is it satisfied when:

- ≡ U replaces **the variable occurrences** satisfying the constraint when increasing (according to \subseteq),
- ≡ $U \cup V$ replaces **the variable occurrences** violating the constraint when increasing (according to \subseteq).

Constraints: example

Let $\epsilon \in \mathbb{R}_+$, a user-defined parameter,

$$\mathcal{C}_{\text{square}}(X^1, X^2) \equiv \frac{|X^1|}{|X^2|} - \frac{|X^2|}{|X^1|} \leq \epsilon \wedge \frac{|X^2|}{|X^1|} - \frac{|X^1|}{|X^2|} \leq \epsilon$$

When increasing (according to the \subseteq order),

- ≡ the green occurrences of the variables tend to satisfy $\mathcal{C}_{\text{square}}$,
- ≡ the red occurrences of the variables tend to violate $\mathcal{C}_{\text{square}}$.

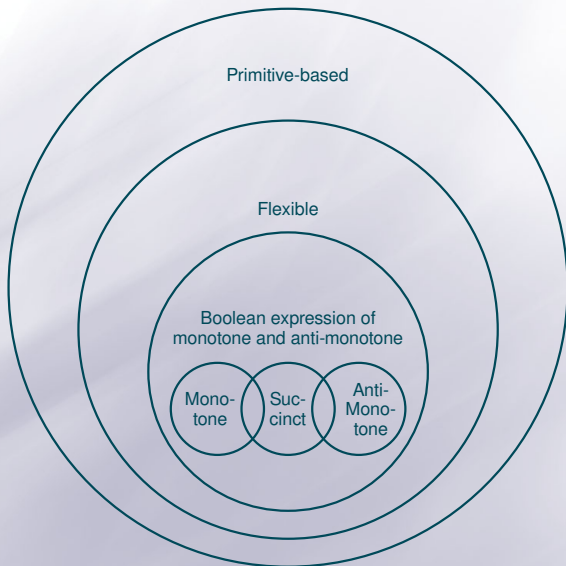
Constraints: example

Let $\epsilon \in \mathbb{R}_+$, a user-defined parameter,

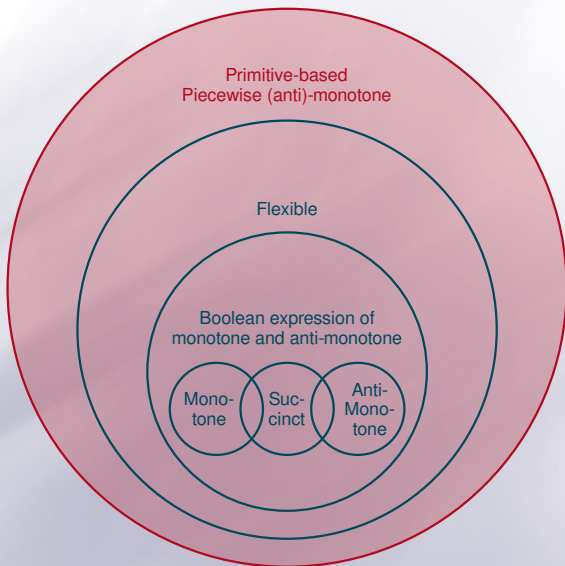
$$C_{\text{square}}(U, V) \equiv \frac{|U^1 \cup V^1|}{|U^2|} - \frac{|U^2|}{|U^1 \cup V^1|} \leq \epsilon \wedge \frac{|U^2 \cup V^2|}{|U^1|} - \frac{|U^1|}{|U^2 \cup V^2|} \leq \epsilon$$

- ≡ the **green** occurrences of the variables are replaced by the related U^i s,
- ≡ the **red** occurrences of the variables are replaced by the related $U^i \cup V^i$ s.

Constraints: how many?



Constraints: how many?



Outline

- 1 Closed n -set
- 2 Enumeration
- 3 Constraints
- 4 Dynamic graph mining: a case study
- 5 Noise tolerance
- 6 More noise tolerance
- 7 Conclusion

$$\{\phi \in L \mid \mathcal{C}_{\text{connected}}(\phi, \mathcal{G}) \wedge \mathcal{C}_{\text{closed}}(\phi, \mathcal{G}) \wedge \mathcal{C}_{\text{P(A)M}}(\phi, \mathcal{G})\}$$

Example of a dynamic graph

Vélo'v network usage

At a given day and a given one-hour period:



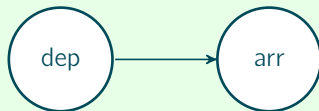
Some figures

- ≡ More than 13.1 million rides from May 2005 to December 2007
- ≡ 117,411 edges
- ≡ 327 (departure and arrival) stations
- ≡ 7 days
- ≡ 24 one-hour periods

Example of a dynamic graph

Vélo'v network usage

At a given day and a given one-hour period:



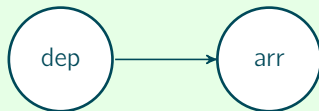
Some figures

- ≡ More than 13.1 million rides from May 2005 to December 2007
- ≡ 117,411 edges
- ≡ 327 (departure and arrival) stations
- ≡ 7 days
- ≡ 24 one-hour periods

Example of a dynamic graph

Vélo'v network usage

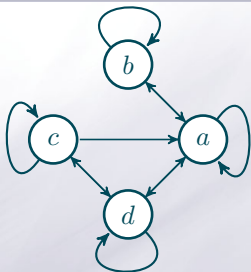
At a given day and a given one-hour period:



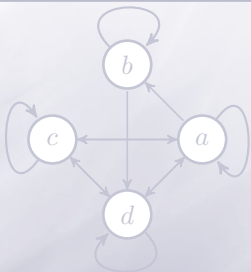
Some figures

- ≡ More than 13.1 million rides from May 2005 to December 2007
- ≡ 117,411 edges
- ≡ 327 (departure and arrival) stations
- ≡ 7 days
- ≡ 24 one-hour periods

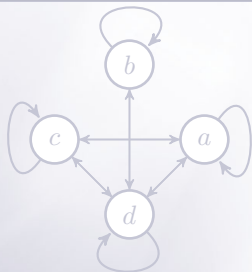
Link with closed n -set mining



(Sunday, [0am-1am])



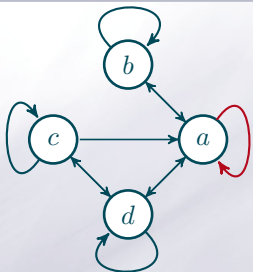
(Sunday, [1am-2am])



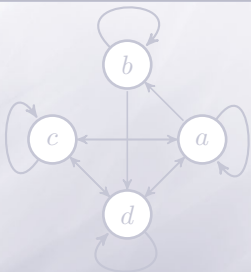
(Sunday, [2am-3am])

	a	b	c	d
a	★	★		★
b	★	★		
c	★		★	★
d	★		★	★

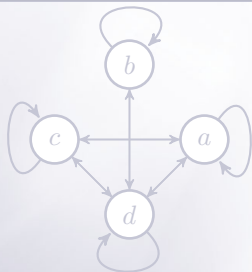
Link with closed n -set mining



(Sunday, [0am-1am])



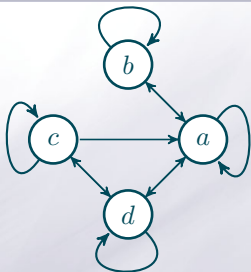
(Sunday, [1am-2am])



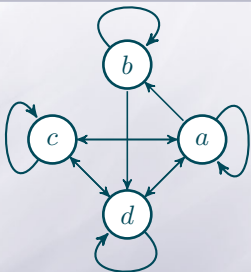
(Sunday, [2am-3am])

	a	b	c	d
a	★	★		★
b	★	★		
c	★		★	★
d	★		★	★

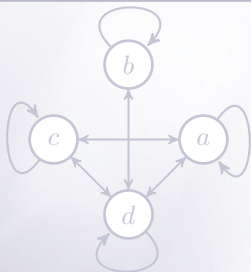
Link with closed n -set mining



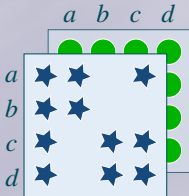
(Sunday, [0am–1am])



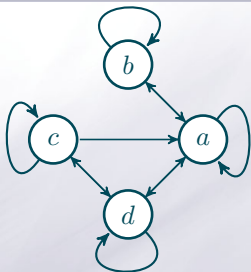
(Sunday, [1am–2am])



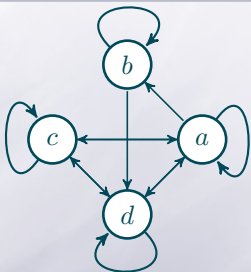
(Sunday, [2am–3am])



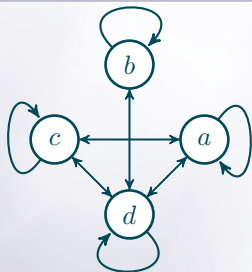
Link with closed n -set mining



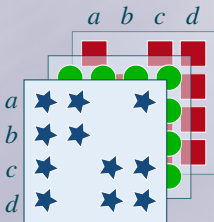
(Sunday, [0am-1am])



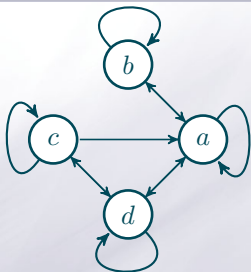
(Sunday, [1am-2am])



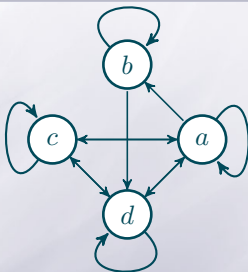
(Sunday, [2am-3am])



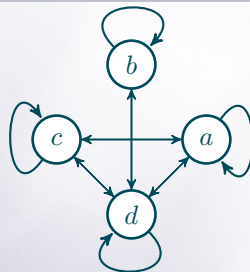
Link with closed n -set mining



(Sunday, [0am-1am])



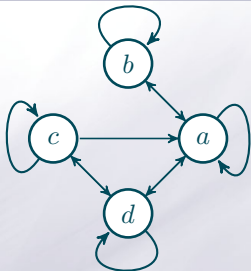
(Sunday, [1am-2am])



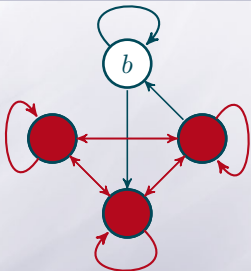
(Sunday, [2am-3am])

	a	b	c	d
a				
b				
c				
d				

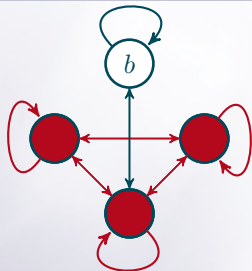
Link with closed n -set mining



(Sunday, [0am–1am])



(Sunday, [1am–2am])



(Sunday, [2am–3am])

	a	b	c	d
a	■		■	■
b				
c	■		■	■
d	■		■	■

Two relevance constraints for dynamic graph mining

Symmetry constraint

A 3-set (T, N^1, N^2) is symmetric $\Leftrightarrow N^1 \subseteq N^2 \wedge N^2 \subseteq N^1$.

With this additional constraint, only *maximal cross-graph cliques* are extracted.

δ -contiguity constraint

Let $\delta \in \mathbb{R}_+$, a user-defined parameter. A 3-set (T, N^1, N^2) is δ -contiguous $\Leftrightarrow \forall t \in [\min(T), \max(T)], \exists t' \in T$ s.t. $|t - t'| \leq \delta$.

Two relevance constraints for dynamic graph mining

Symmetry constraint

A 3-set (T, N^1, N^2) is symmetric $\Leftrightarrow N^1 \subseteq N^2 \wedge N^2 \subseteq N^1$.

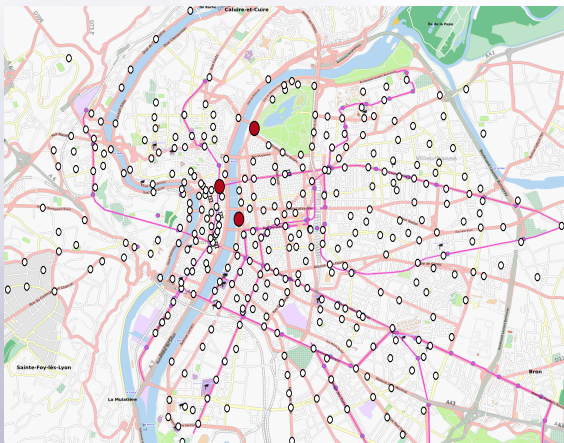
With this additional constraint, only *maximal cross-graph cliques* are extracted.

δ -contiguity constraint

Let $\delta \in \mathbb{R}_+$, a user-defined parameter. A 3-set (T, N^1, N^2) is δ -contiguous $\Leftrightarrow \forall t \in [\min(T), \max(T)], \exists t' \in T$ s.t. $|t - t'| \leq \delta$.

Discovered closed 4-set

During the week-ends,
from 5pm to 8pm:



©2004-2010 OpenStreetMap contributors (created from OpenStreetMap project data)

 This map is licensed under Creative Commons Attribution ShareAlike 2.0 License.

Outline

- 1 Closed n -set
- 2 Enumeration
- 3 Constraints
- 4 Dynamic graph mining: a case study
- 5 Noise tolerance**
- 6 More noise tolerance
- 7 Conclusion

$$\{\phi \in L \mid \mathcal{C}_{\text{connected}}(\phi, \mathcal{R}) \wedge \mathcal{C}_{\text{closed}}(\phi, \mathcal{R}) \wedge \dots\}$$

Impact of noise

False-positive noise Unwanted tuples are inserted.

False-negative noise Wanted tuples are erased.

We define the *noise level* as the probability, for any tuple, to be affected by the noise (uniform distribution).

Closed 2-sets in *noisy* binary relations

Let a 3×3 closed 2-set and a false positive tuple.

	A	B	C
1	●	●	●
2	●	●	●
3	●	●	●
4			

The closed 2-set:

$(\{1, 2, 3\}, \{A, B, C\})$

and another one:

$(\{1, 2, 3, 4\}, \{C\})$

Additional constraints allow to remove the supernumerary pattern.

Closed 2-sets in *noisy* binary relations

Let a 3×3 closed 2-set and a false positive tuple.

	A	B	C
1	●	●	●
2	●	●	●
3	●	●	●
4			●

The closed 2-set:

$(\{1, 2, 3\}, \{A, B, C\})$

and another one:

$(\{1, 2, 3, 4\}, \{C\})$

Additional constraints allow to remove the supernumerary pattern.

Closed 2-sets in *noisy* binary relations

Let a 3×3 closed 2-set and a false positive tuple.

	A	B	C
1	●	●	●
2	●	●	●
3	●	●	●
4			●

The closed 2-set:

$(\{1, 2, 3\}, \{A, B, C\})$

and another one:

$(\{1, 2, 3, 4\}, \{C\})$

Additional constraints allow to remove the supernumerary pattern.

Closed 2-sets in *noisy* binary relations

Let a 3×3 closed 2-set and a false negative tuple (noise level: $\frac{1}{3^2} = 0.11$).

	A	B	C
1	●	●	●
2	●	●	●
3	●	●	
4			

2 fragments of the pattern to discover:

($\{1, 2, 3\}, \{A, B\}$)

($\{1, 2\}, \{A, B, C\}$)

Need for tolerating false negative tuples.

Closed 2-sets in *noisy* binary relations

Let a 3×3 closed 2-set and a false negative tuple (noise level: $\frac{1}{3^2} = 0.11$).

	A	B	C
1	●	●	●
2	●	●	●
3	●	●	
4			

2 **fragments** of the pattern to discover:










($\{1, 2, 3\}, \{A, B\}$)

($\{1, 2\}, \{A, B, C\}$)

Need for tolerating false negative tuples.

Closed n -set in *noisy* n -ary relations

Let a $3 \times 3 \times 3$ closed 3-set and a false negative tuple 3-set (noise level: $\frac{1}{3^3} = 0.037$).










	A	B	C
1			
2			
3			
4			

1 closed 3-set:

$(\{1, 2, 3\}, \{A, B, C\}, \{\blacksquare, \bullet, \star\})$

Closed n -set in *noisy* n -ary relations

Let a $3 \times 3 \times 3$ closed 3-set and a false negative tuple 3-set (noise level: $\frac{1}{3^3} = 0.037$).

	A	B	C
1			
2			
3			
4			

3 fragments of the pattern to discover:

$(\{1, 2, 3\}, \{A, B, C\}, \{\blacksquare, \bullet\})$










$(\{1, 2, 3\}, \{A, B\}, \{\blacksquare, \bullet, \star\})$

$(\{1, 2\}, \{A, B, C\}, \{\blacksquare, \bullet, \star\})$

With a higher arity, a greater need for tolerating false negative tuples.

Closed n -set in *noisy* n -ary relations

Let a $3 \times 3 \times 3$ closed 3-set and a false negative tuple 3-set (noise level: $\frac{1}{3^3} = 0.037$).

	A	B	C
1			
2			
3			
4			

3 fragments of the pattern to discover:

$(\{1, 2, 3\}, \{A, B, C\}, \{\blacksquare, \bullet\})$

$(\{1, 2, 3\}, \{A, B\}, \{\blacksquare, \bullet, \star\})$

$(\{1, 2\}, \{A, B, C\}, \{\blacksquare, \bullet, \star\})$

With a higher arity, a greater need for tolerating false negative tuples.

Closed ET- n -set: definition

Let $(\mathcal{D}^i)_{i=1..n}$ n finite sets and $\mathcal{R} \subseteq \times_{i=1..n} \mathcal{D}^i$ an n -ary relation. Let $\epsilon = (\epsilon^i)_{i=1..n} \in \mathbb{N}^n$ n noise tolerance thresholds.

Definition

$\forall (X^1, \dots, X^n) \subseteq \times_{i=1..n} \mathcal{D}^i$, (X^1, \dots, X^n) is a closed ET- n -set if and only if:

- $\equiv \mathcal{C}_{\epsilon}$ -connected $(X^1, \dots, X^n) \equiv \forall i = 1..n, \forall x \in X^i$,
 $|\mathcal{R} \cap (X^1 \times \dots \times \{x\} \times \dots \times X^n)| \leq \epsilon^i$
- $\equiv \mathcal{C}_{\epsilon}$ -closed $(X^1, \dots, X^n) \equiv \forall i = 1..n, \forall x \in D^i \setminus X^i$,
 $\neg \mathcal{C}_{\epsilon}$ -connected $(X^1, \dots, X^i \cup \{x\}, \dots, X^n)$

If $\forall i = 1..n, \epsilon^i = 0$, a closed ET- n -set is a closed n -set.

Closed ET- n -set: example

	A	B	C
1			
2			
3			
4			

$$\mathcal{D}^1 = \{1, 2, 3, 4\}$$

$$\mathcal{D}^2 = \{A, B, C\}$$

$$\mathcal{D}^3 = \{\blacksquare, \bullet, \star\}$$





$$\mathcal{R} = \{(1, A, \blacksquare), (1, A, \bullet), (1, A, \star), (1, B, \blacksquare), (1, B, \bullet), (1, B, \star), (1, C, \blacksquare), (1, C, \bullet), (2, A, \blacksquare), (2, A, \bullet), (2, A, \star), (2, B, \blacksquare), (2, B, \star), (3, A, \star), (3, B, \blacksquare), (4, A, \bullet), (4, A, \star), (4, B, \star), (4, C, \blacksquare), (4, C, \bullet), (4, C, \star)\}$$

$$\epsilon^1 = 1$$

$$\epsilon^2 = 1$$












$$\epsilon^3 = 1$$

Closed ET- n -set: example

	A	B	C
1			
2			
3			
4			

$(\{1, 4\}, \{A, C\}, \{\text{■}, \text{●}, \text{★}\})$ is a closed ET-3-set.

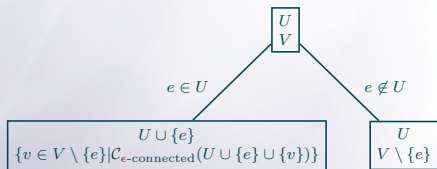
Closed ET- n -set: example

	A	B	C
1			
2			
3			
4			

$(\{1, 4\}, \{A, C\}, \{\text{■}, \text{●}, \text{★}\})$ is a closed ET-3-set.

Closed ET- n -set enumeration

Same enumeration principles as
DATA-PEELER.



FENSTER: an efficient algorithm based on counters.

Global quality of a closed ET- n -set collection

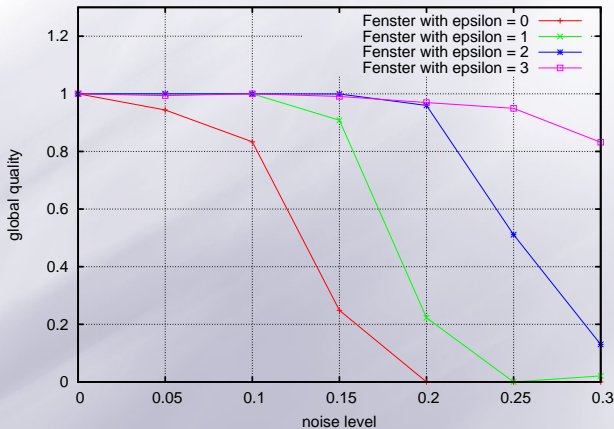
In a $32 \times 32 \times 32$ space, let $\mathcal{R}_{\text{hidden}}$ the tuples covered by four $8 \times 8 \times 8$ (hidden) patterns. Let \mathcal{E} the tuples covered by the collection of closed 3-sets extracted under minimal size constraints of 4 elements per attribute.

Definition

We define the *global quality* of \mathcal{E} as:

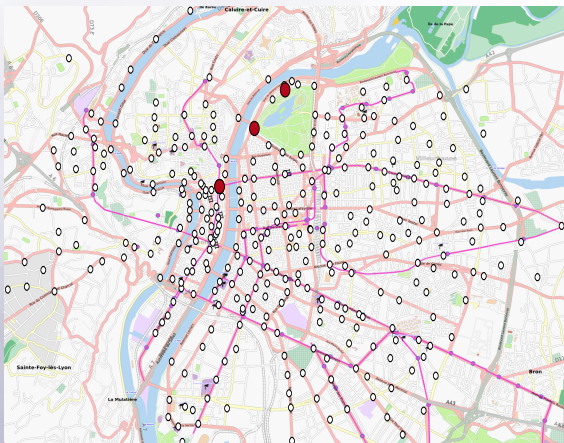
$$\frac{|\mathcal{R}_{\text{hidden}} \cap \mathcal{E}|}{|\mathcal{R}_{\text{hidden}} \cup \mathcal{E}|}$$

Closed ET- n -sets global quality



Discovered closed ET-4-set

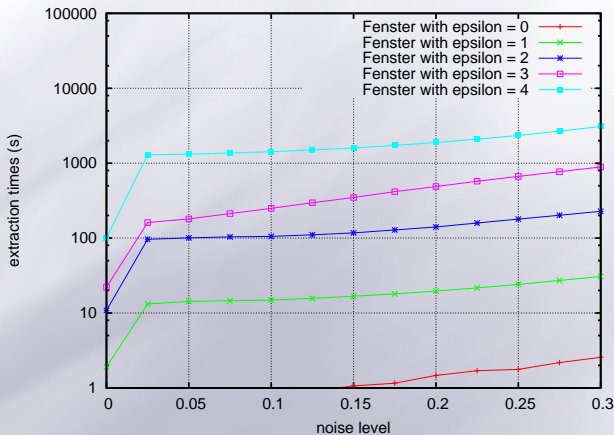
During the week-ends,
from 4pm to 8pm:



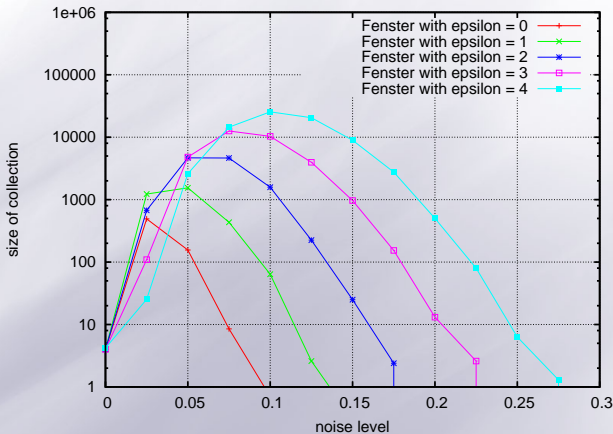
©2004-2010 OpenStreetMap contributors (created from OpenStreetMap project data)

 This map is licensed under Creative Commons Attribution ShareAlike 2.0 License.

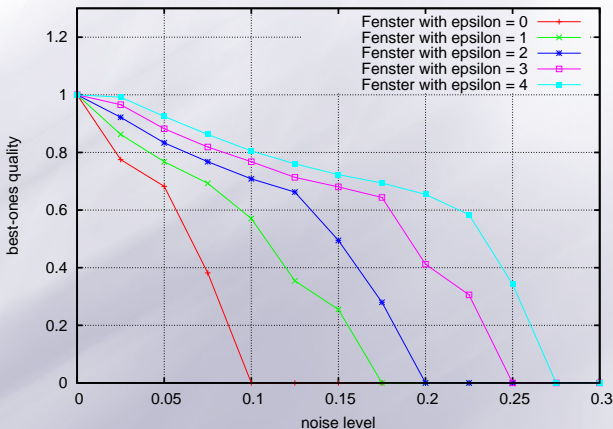
Closed ET- n -set extraction times



Size of closed ET- n -set collections



Closed-ET- n -set best-ones quality



Absolute vs. relative tolerance to noise

A relative tolerance to noise looks more natural but:

- ≡ the closedness constraint does not support a lossless condensation of all patterns;
- ≡ the search-space cannot be pruned as much as with an absolute tolerance to noise.

Outline

- 1 Closed n -set
- 2 Enumeration
- 3 Constraints
- 4 Dynamic graph mining: a case study
- 5 Noise tolerance
- 6 More noise tolerance**
- 7 Conclusion

$$\{\phi \in L \mid \mathcal{C}_{\text{ET-connected}}(\phi, \mathcal{R}) \wedge \mathcal{C}_{\text{ET-closed}}(\phi, \mathcal{R}) \wedge \dots\}$$

+ heuristic post-process

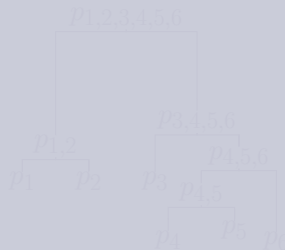
Global vs best-ones quality

The closed ET- n -sets are:

- ≡ numerous fragments of the hidden patterns,
- ≡ globally relevant.

Idea: ALPHA

1. Hierarchical agglomeration of the closed ET- n -sets,
2. Selection of the relevant agglomerated patterns.



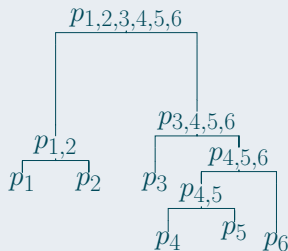
Global vs best-ones quality

The closed ET- n -sets are:

- ≡ numerous fragments of the hidden patterns,
- ≡ globally relevant.

Idea: ALPHA

1. Hierarchical agglomeration of the closed ET- n -sets,
2. Selection of the relevant agglomerated patterns.



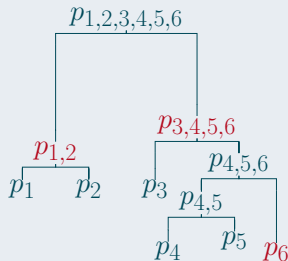
Global vs best-ones quality

The closed ET- n -sets are:

- ≡ numerous fragments of the hidden patterns,
- ≡ globally relevant.

Idea: ALPHA

1. Hierarchical agglomeration of the closed ET- n -sets,
2. Selection of the relevant agglomerated patterns.



Agglomeration operator \sqcup

Definition

Given two n -sets $X = (X^1, \dots, X^n)$ and $Y = (Y^1, \dots, Y^n)$,
 $X \sqcup Y = (X^1 \cup Y^1, \dots, X^n \cup Y^n)$.

	A	B	C	D	E	
1	●	●	●		●	$\mathcal{O} = \{1, 2, 3, 4, 5\}$ $\mathcal{P} = \{A, B, C, D, E\}$ $\mathcal{R} = \{(1, A), (1, B), (1, C), (1, E),$ $(2, A), (2, B), (2, C), (2, D),$ $(3, A), (3, B), (3, C), (3, D),$ $(3, E), (4, A), (4, C), (4, D),$ $(4, E), (5, B), (5, C), (5, D),$ $(5, E)\}$
2	●	●	●	●		
3	●	●	●	●	●	
4	●		●	●	●	
5		●	●	●	●	

Agglomeration operator \sqcup

Definition

Given two n -sets $X = (X^1, \dots, X^n)$ and $Y = (Y^1, \dots, Y^n)$,
 $X \sqcup Y = (X^1 \cup Y^1, \dots, X^n \cup Y^n)$.

	A	B	C	D	E
1	●	●	●		●
2	●	●	●	●	
3	●	●	●	●	●
4	●		●	●	●
5		●	●	●	●

$(\{1, 2, 3\}, \{A, B, C\})$ is a closed 2-set.

Agglomeration operator \sqcup

Definition

Given two n -sets $X = (X^1, \dots, X^n)$ and $Y = (Y^1, \dots, Y^n)$,
 $X \sqcup Y = (X^1 \cup Y^1, \dots, X^n \cup Y^n)$.

	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>
1	●	●	●		●
2	●	●	●	●	
3	●	●	●	●	●
4	●		●	●	●
5		●	●	●	●

$(\{3, 4, 5\}, \{C, D, E\})$ is a closed 2-set.

Agglomeration operator \sqcup

Definition

Given two n -sets $X = (X^1, \dots, X^n)$ and $Y = (Y^1, \dots, Y^n)$,
 $X \sqcup Y = (X^1 \cup Y^1, \dots, X^n \cup Y^n)$.

	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	
1	●	●	●	○	●	\sqcup $\frac{(\{1, 2, 3\}, \{A, B, C\}) \quad (\{3, 4, 5\}, \{C, D, E\})}{= (\{1, 2, 3, 4, 5\}, \{A, B, C, D, E\})}$
2	●	●	●	●	○	
3	●	●	●	●	●	
4	●	○	●	●	●	
5	○	●	●	●	●	

Metric: intuition

	A	B	C	D	E
1	●	●	●		●
2	●	●	●	●	
3	●	●	●	●	●
4	●		●	●	●
5		●	●	●	●

	A	B	C	D	E
1	●	●	●		
2	●	●	●		
3	●	●	●	●	●
4			●	●	●
5			●	●	●

$(\{1, 2, 3\}, \{A, B, C\})$ and $(\{3, 4, 5\}, \{C, D, E\})$ are “closer” in the relation on the left.

- ≡ $\frac{1}{5}$ absent n -tuple is tolerated;
- ≡ $\frac{2}{5}$ absent n -tuples are tolerated.

Metric: intuition

	A	B	C	D	E
1	●	●	●	○	●
2	●	●	●	●	
3	●	●	●	●	●
4	●		●	●	●
5		●	●	●	●

	A	B	C	D	E
1	●	●	●	○	○
2	●	●	●		
3	●	●	●	●	●
4			●	●	●
5			●	●	●

On element 1,

- ≡ $\frac{1}{5}$ absent n -tuple is tolerated;
- ≡ $\frac{2}{5}$ absent n -tuples are tolerated.

Metric: intuition

	A	B	C	D	E
1	●	●	●		●
2	●	●	●	●	
3	●	●	●	●	●
4	●		●	●	●
5	○	●	●	●	●

	A	B	C	D	E
1	●	●	●		
2	●	●	●		
3	●	●	●	●	●
4	○		●	●	●
5	○		●	●	●

On element A ,

- ≡ $\frac{1}{5}$ absent n -tuple is tolerated;
- ≡ $\frac{2}{5}$ absent n -tuples are tolerated.

Metric: definition

Definition (Intrinsic distance d)

Given a weight function $w : \cup_{i=1}^n D^i \rightarrow \mathbb{R}_+$ (greater weights mean less tolerance to noise) and an n -set $X = (X^1, \dots, X^n)$,

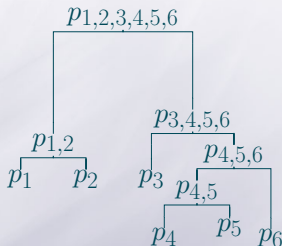
$$d(X) = \max_{i=1}^n \left(\max_{x \in X^i} \left(w(x) \frac{|K \setminus \mathcal{R}|}{|K|} \right) \right)$$

where $K = X^1 \times \dots \times X^{i-1} \times \{x\} \times X^{i+1} \times \dots \times X^n$.

Definition (Metric)

The distance between two n -sets X and Y is $d(X \sqcup Y)$.

Relevance measure: intuition

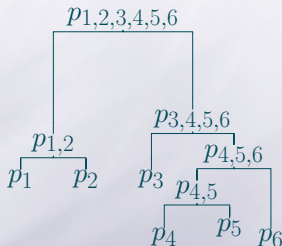


Local pattern

David Hand wrote:

A local pattern is a data vector serving to describe an anomalously high local density of data points.

Relevance measure: intuition

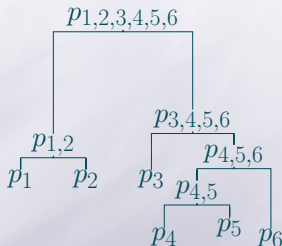


Local pattern

David Hand wrote:

*A local pattern is a data vector serving to describe an anomalously **high local density** of data points.*

Relevance measure: intuition



Local pattern

David Hand wrote:

*A local pattern is a data vector serving to describe an **anomalously** high local density of data points.*

Relevance measure r : definition

Definition

Given an n -set X and its parent $X \sqcup Y$ after hierarchical agglomeration, $r(X) = d(X \sqcup Y) - \max(d(X), d(Y)) - d(X)$

Relevance measure r : definition

Definition

Given an n -set X and its parent $X \sqcup Y$ after hierarchical agglomeration, $r(X) = d(X \sqcup Y) - \max(d(X), d(Y)) - d(X)$

Relevance measure r : definition

Definition

Given an n -set X and its parent $X \sqcup Y$ after hierarchical agglomeration, $r(X) = d(X \sqcup Y) - \max(d(X), d(Y)) - d(X)$

Selection of the relevant agglomerated patterns

Idea

Somehow preserve the completeness of the initial extraction.

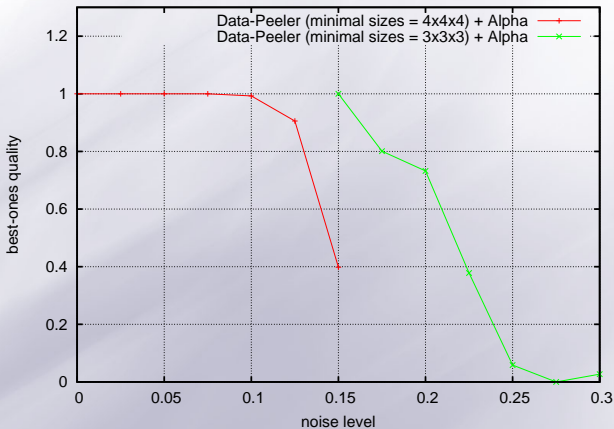
Global constraint

Every extracted closed ET- n -set is covered by at least one output pattern.

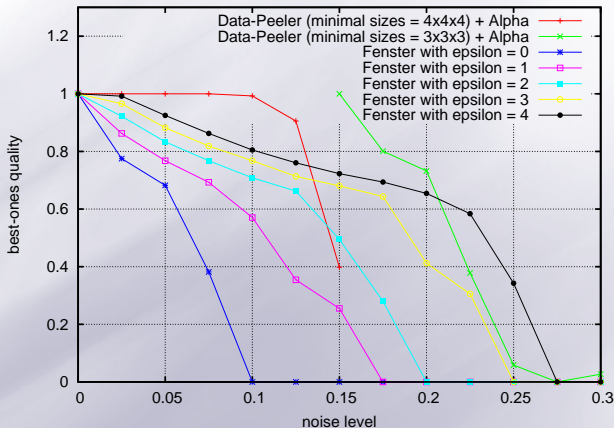
Procedure

Read the agglomerated patterns by decreasing relevance order and stop once the previously read patterns cover every extracted closed ET- n -set.

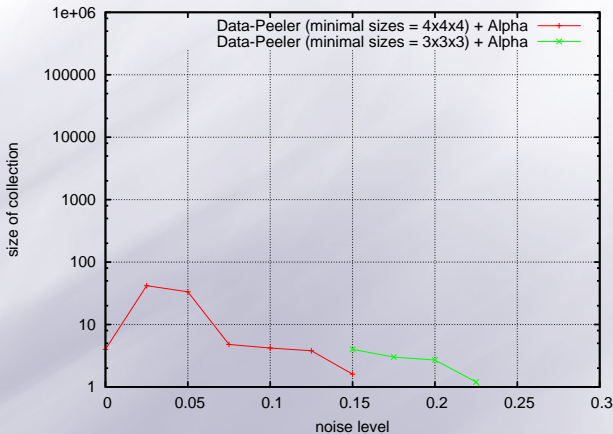
Alpha: best-ones quality



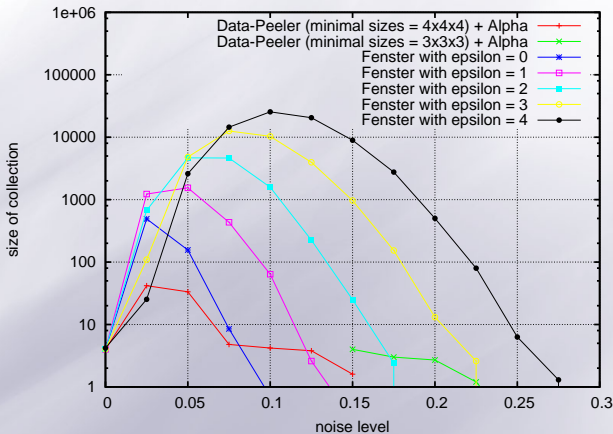
Alpha: best-ones quality



Alpha: size of collection

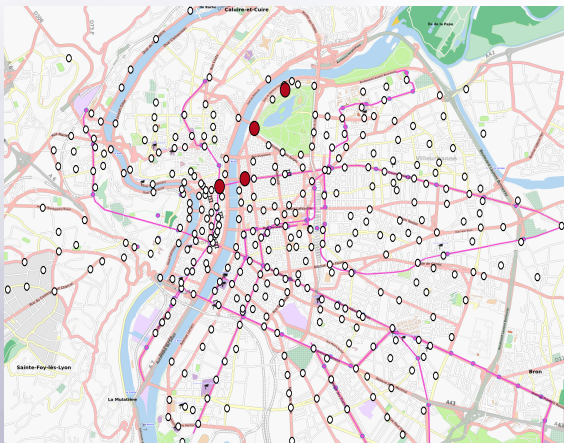


Alpha: size of collection



Example of a relevant pattern

During the week-ends,
from 3pm to 8pm:



©2004-2010 OpenStreetMap contributors (created from OpenStreetMap project data)

 This map is licensed under Creative Commons Attribution ShareAlike 2.0 License.

Outline

- 1 Closed n -set
- 2 Enumeration
- 3 Constraints
- 4 Dynamic graph mining: a case study
- 5 Noise tolerance
- 6 More noise tolerance
- 7 Conclusion**

Summary: DATA-PEELER

DATA-PEELER (2008):

- ≡ generalizes closed 2-set mining towards n -ary relations,
- ≡ extracts complete collections of closed n -sets,
- ≡ does not favor any attribute when enumerating the elements,
- ≡ can take advantage of a broad class of relevance constraints,
- ≡ in particular, can be specialized in dynamic graph mining.

Collaboration with Jérémy Besson and Céline Robardet.

Cerf et al.: SIAM SDM '08, ACM Trans. KDD in 2009, ISMIS '09.

Summary: FENSTER

FENSTER (2009):

- ≡ generalizes DATA-PEELER towards noise absolute tolerance,
- ≡ extracts complete collections of closed ET- n -sets,
- ≡ provides high global qualities,
- ≡ provides, in a reasonable time, low best-ones qualities,
- ≡ returns too many closed ET- n -sets.

Collaboration with Jérémy Besson.

Cerf et al.: EGC '09.

Summary: ALPHA

ALPHA (2009):

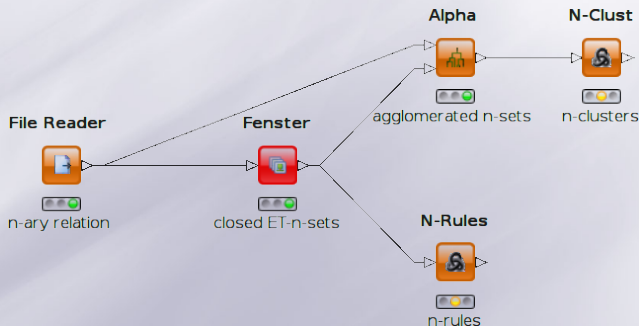
- ≡ complements FENSTER to relatively tolerate more noise,
- ≡ hierarchically agglomerates the closed ET- n -sets,
- ≡ somehow preserves the initial completeness,
- ≡ provides high best-ones qualities,
- ≡ returns small collections of patterns.

Collaboration with Pierre-Nicolas Mougel.

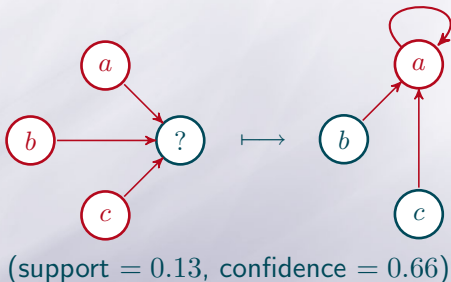
Cerf et al.: ACM CIKM '09.

Around n -ary relation mining

- ≡ Design of pre- and post-processes;
- ≡ Integration in the KNIME platform.



A rule in a dynamic graph



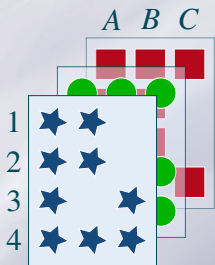
Collaboration with Thi Kim Ngan Nguyen and Marc Plantevit.

Nguyen et al.: BDA '10, DyNaK '10.

Theory and application of n -ary relation mining

Generalize From binary relations to n -ary relations to **probabilistic datasets**;

Specialize From monotonicity/anti-monotonicity to piecewise (anti)-monotonicity to **actionability**.



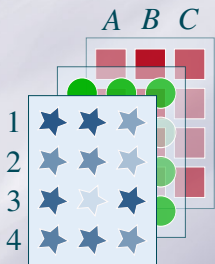
Vélo'v usage analysis

- ≡ Patterns discriminating the week-ends;
- ≡ Weak geographic spreading;
- ≡ Vélo'v in the multimodal transportation;
- ≡ ...

Theory and application of n -ary relation mining

Generalize From binary relations to n -ary relations to **probabilistic datasets**;

Specialize From monotonicity/anti-monotonicity to piecewise (anti)-monotonicity to **actionability**.



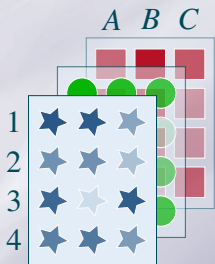
Vélo'v usage analysis

- ≡ Patterns discriminating the week-ends;
- ≡ Weak geographic spreading;
- ≡ Vélo'v in the multimodal transportation;
- ≡ ...

Theory and application of n -ary relation mining

Generalize From binary relations to n -ary relations to **probabilistic datasets**;

Specialize From monotonicity/anti-monotonicity to piecewise (anti)-monotonicity to **actionability**.



Vélo'v usage analysis

- ≡ Patterns discriminating the week-ends;
- ≡ Weak geographic spreading;
- ≡ Vélo'v in the multimodal transportation;
- ≡ ...