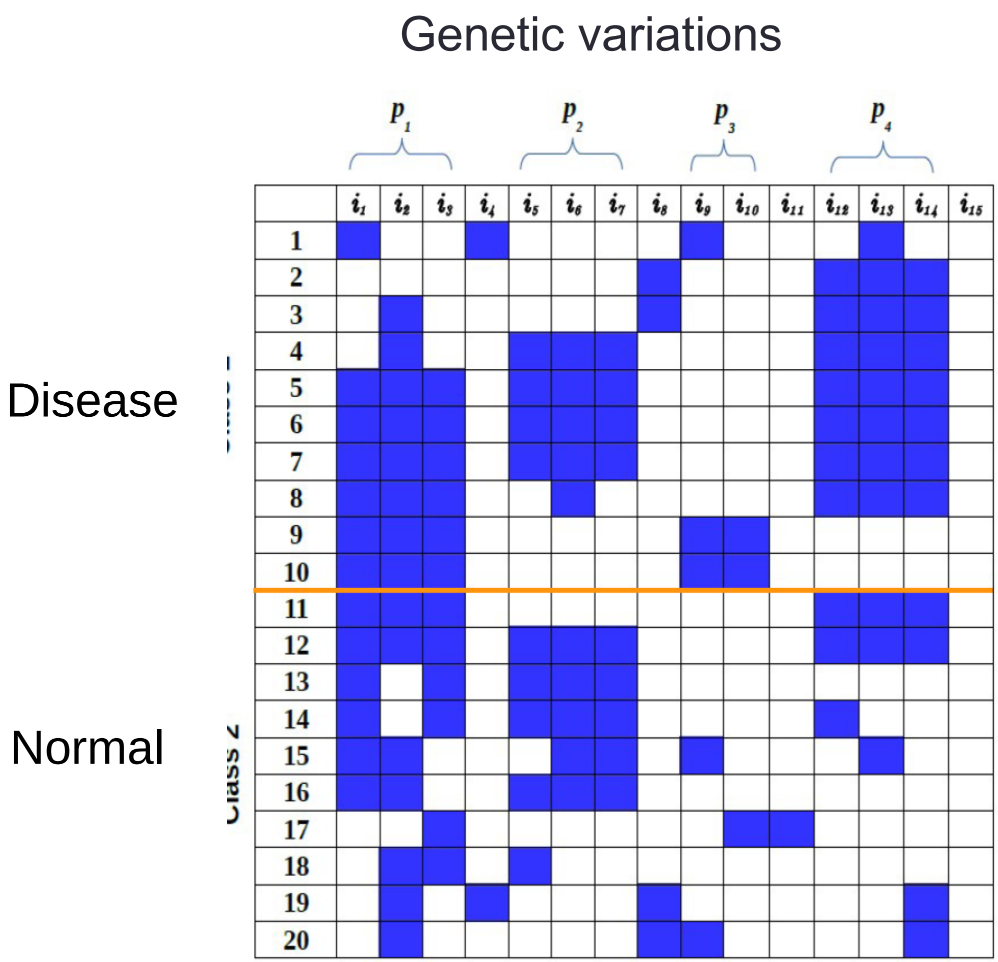
Statistically Significant Discriminative Pattern Search

Hoang-Son Pham[1], Gwendal Virlet[2], Dominique Lavenier[2], Alexandre Termier[2]

[1] ICTEAM-Université Catholique de Louvain, Belgium,
[2] Univ Rennes, Inria, CNRS, IRISA

21st International Conference on **Big Data Analytics and Knowledge Discovery**
August 27, 2019

Introduction



?

$\{i_1, i_2, i_3\}$: 6 diseases / 2 normal

$\{i_5, i_6, i_7\}$: 4 diseases / 4 normal

$\{i_{12}, i_{13}, i_{14}\}$: 7 diseases / 2 normal

HOW ?

- search patterns : data mining
- evaluate quality : statistic

Problem definition

Tids	Items										Class
1	a	b	c			f			i	j	1
2	a	b	c		e		g		i		1
3	a	b	c			f		h		j	1
4		b		d	e		g		i	j	1
5				d		f	g	h	i	j	1
6		b	c		e		g	h		j	0
7	a	b	c			f	g	h			0
8		b	c	d	e			h	i		0
9	a			d	e		g	h		j	0

Support of the pattern p in class i:

$$sup(p, D_i) = \frac{|D_i(p)|}{|D_i|}$$

Negative support:

$$\overline{\text{sup}}(p, D_i) = 1 - sup(p, D_i)$$

Discriminative score measures:

Support difference:

$$SD(p, D) = sup(p, D_1) - sup(p, D_2)$$

Grow rate supports:

$$GR(p, D) = \frac{sup(p, D_1)}{sup(p, D_2)}$$

Odds ratio supports:

$$ORS(p, D) = \frac{sup(p, D_1)/\overline{\text{sup}}(p, D_1)}{sup(p, D_2)/\overline{\text{sup}}(p, D_2)}$$

Problem definition

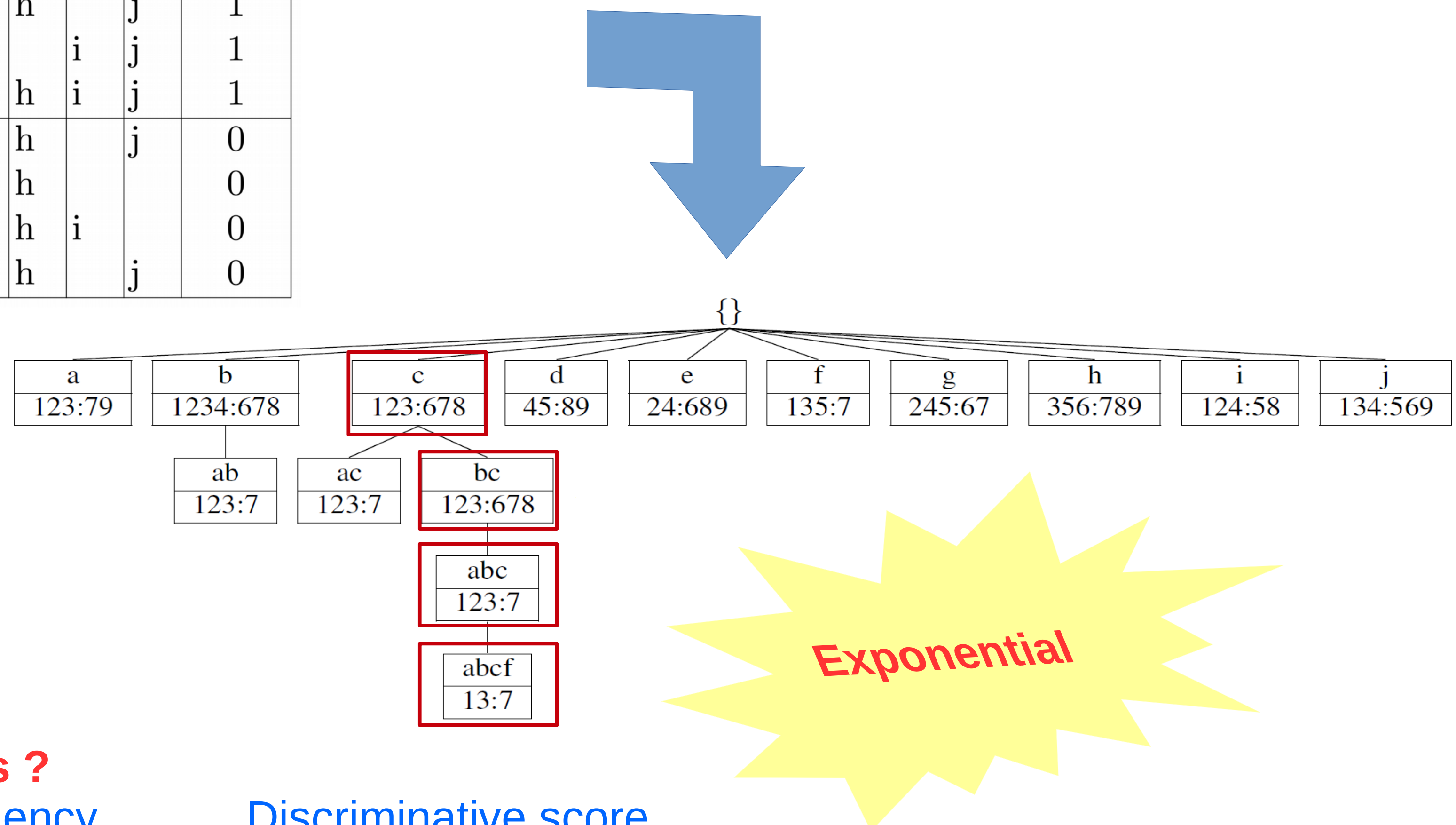
- Confidence interval:
  - a range of values (Lower confidence interval – Upper confidence interval)
  - used to evaluate the statistical significance of a result

Statistically significant discriminative pattern

a pattern satisfies both discriminative threshold (alpha) and confidence interval (beta)

Classical enumeration strategy

Tids	Items										Class
1	a	b	c			f			i	j	1
2	a	b	c		e		g		i		1
3	a	b	c			f		h		j	1
4		b		d	e		g		i	j	1
5				d		f	g	h	i	j	1
6		b	c		e		g	h		j	0
7	a	b	c			f	g	h			0
8		b	c	d	e			h	i		0
9	a			d	e		g	h		j	0



Pruning strategies ?

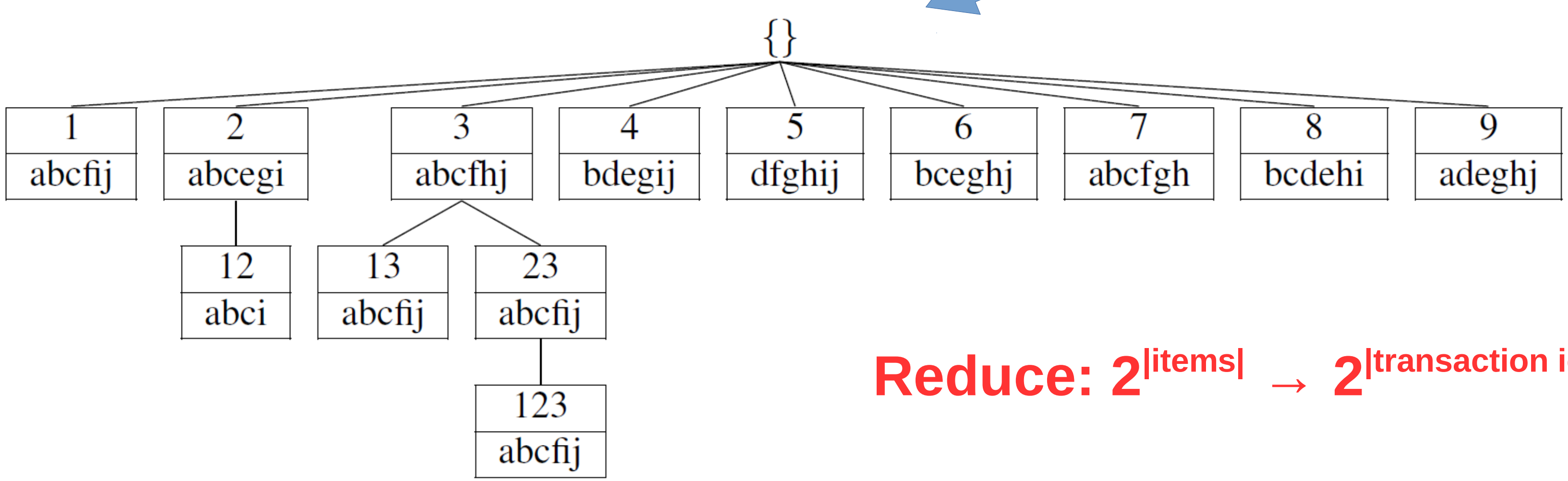
Itemset	Frequency	Discriminative score
c	*6*	*OR = 0.5*
bc	*6*	*OR = 0.5*
abc	*4*	*OR = 4.5*
abcf	*3*	*OR = 2.0*
Anti-monotonic ?	***YES***	***NO!***

Better enumeration strategy

Transaction ids	Items										Class
1	a	b	c			f			i	j	1
2	a	b	c		e		g		i		1
3	a	b	c			f		h		j	1
4		b		d	e		g		i	j	1
5				d		f	g	h	i	j	1
6		b	c		e		g	h		j	0
7	a	b	c			f	g	h			0
8		b	c	d	e			h	i		0
9	a			d	e		g	h		j	0

Transposition

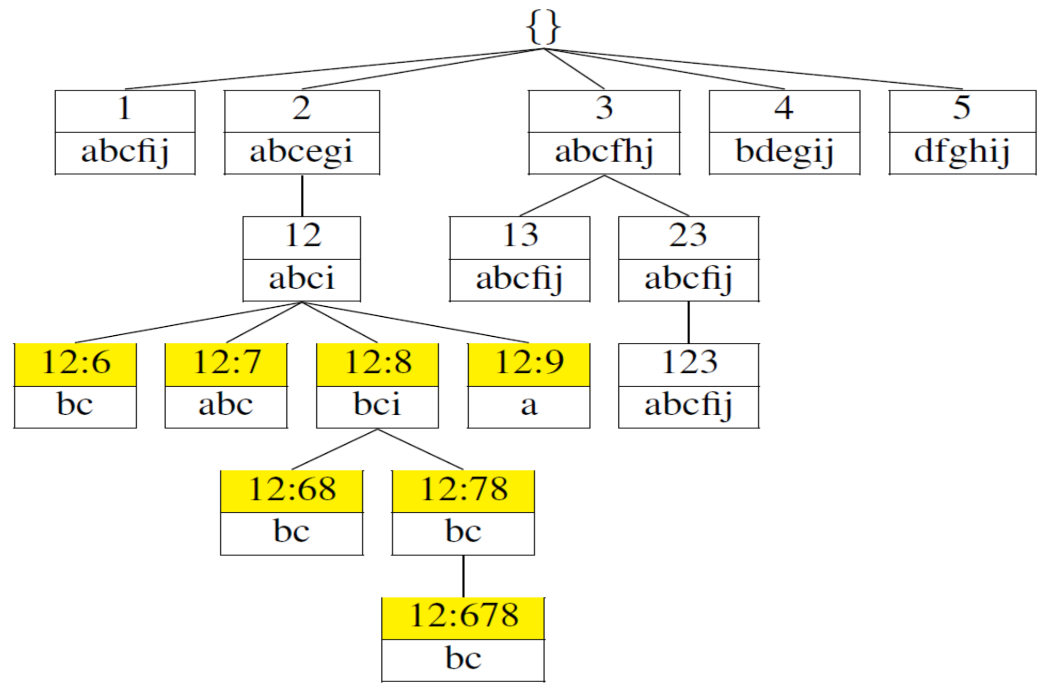
Items	Transaction ids								
a	1	2	3				7		9
b	1	2	3	4		6	7	8	
c	1	2	3			6	7	8	
d				4	5			8	9
e		2		4		6		8	9
f	1		3		5		7		
g		2		4	5	6	7		9
h			3		5	6	7	8	9
i	1	2		4	5			8	
j	1		3	4	5	6			9
class	1	1	1	1	1	0	0	0	0



Reduce: $2^{|\text{items}|} \rightarrow 2^{|\text{transaction ids}|}$

Better enumeration strategy + pruning

Items	Transaction ids								
a	1	2	3				7		9
b	1	2	3	4		6	7	8	
c	1	2	3			6	7	8	
d				4	5			8	9
e		2		4		6		8	9
f	1		3		5		7		
g		2		4	5	6	7		9
h			3		5	6	7	8	9
i	1	2		4	5			8	
j	1		3	4	5	6			9
class	1	1	1	1	1	0	0	0	0



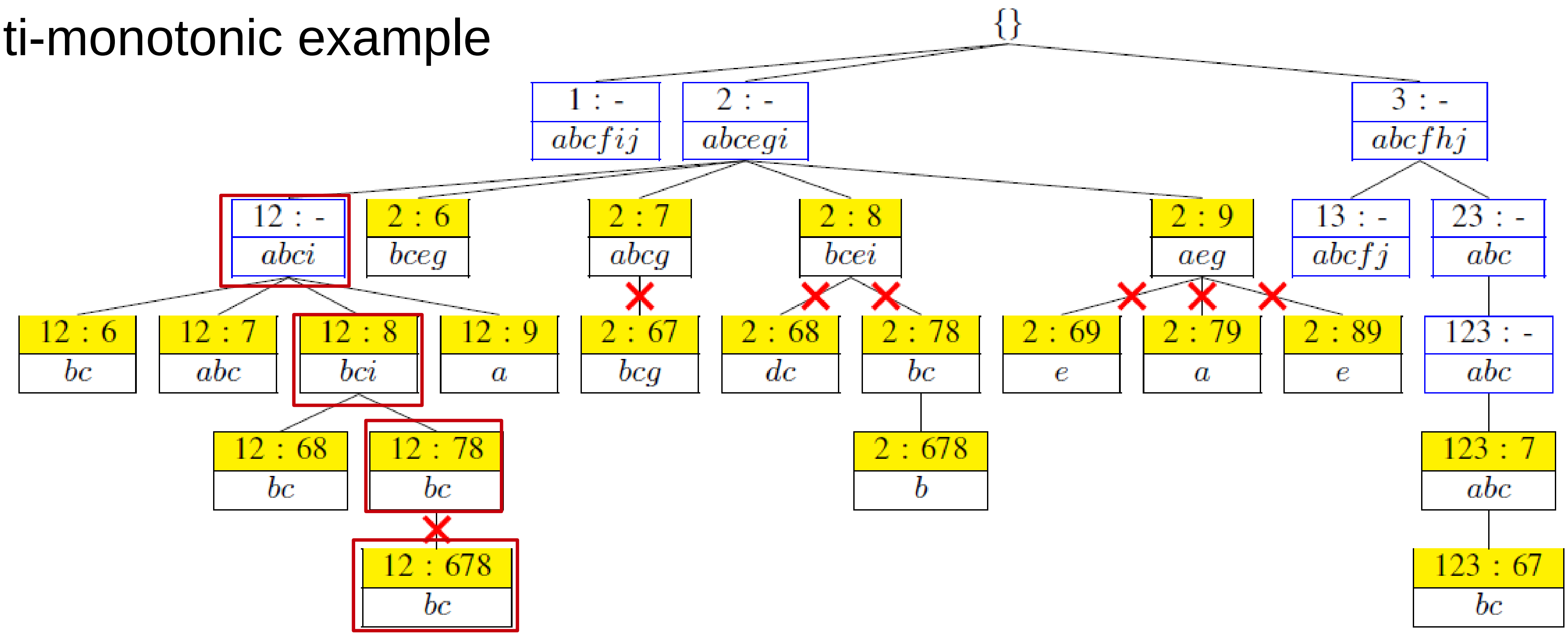
Important contribution:

discriminative score measures and confidence interval are anti-monotonic on a branch

→ search discriminative patterns more effectively (with pruning)

Enumeration strategy

- Anti-monotonic example



Threshold = 1	*Tidset*	*Itemset*	*Discriminative score*
-	12 : -	abci	OR = +∞
YES	12 : 8	bci	OR = 2*3 / 3*1 = 2
NO	12 : 78	bc	OR = 2*2 / 3*2 = 0.66
Pruned	12 : 678	bc	OR = 2*1 / 3*3 = 0.22

Anti-monotonicity formally proved in document

SSDPS algorithm

Algorithm 1: SSDPS algorithm

---

input : D, α, β

output: a set of statistically significant discriminative patterns

Mining closed frequent pattern in the 1^{st} class by using LCM algorithm's principles

for *each closed pattern found in the* 1^{st} *class* **do**

  expand it with the transaction id in the 2^{nd} class

  if *the new pattern satisfies the given thresholds* **then**

    calculate the closure extension

    if *the pattern has extension* **then**

      continue expand the new pattern

    else

      print the pattern

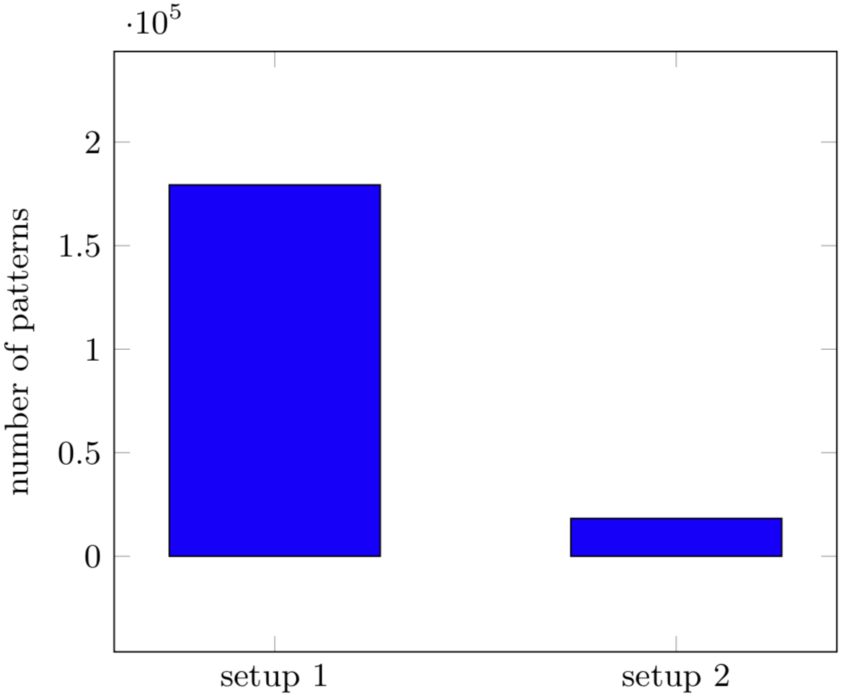
  else

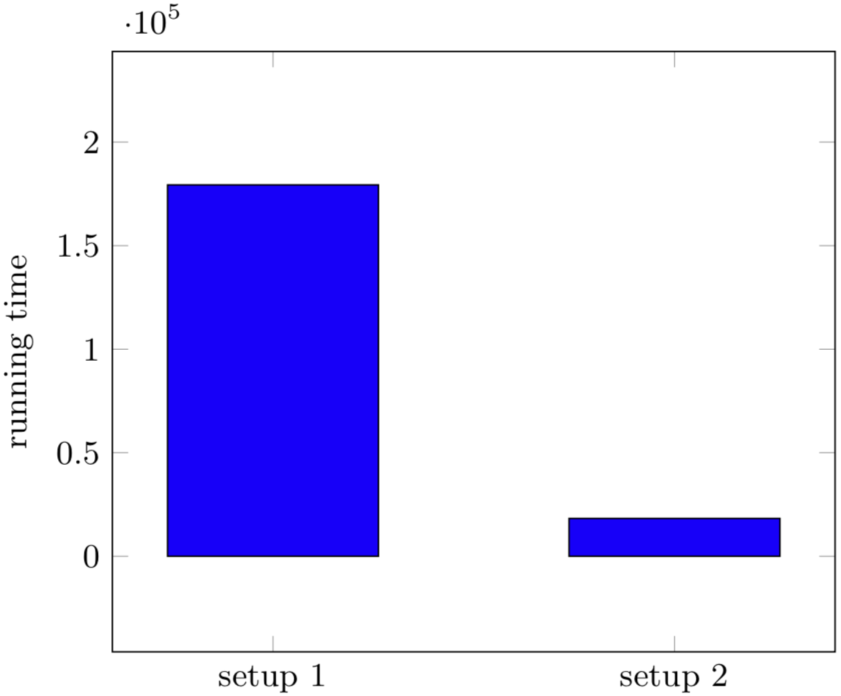
    prune

Experimental results

Simulated data: 100 transactions (50 cases, 50 controls), 260 items

- Pruning evaluation

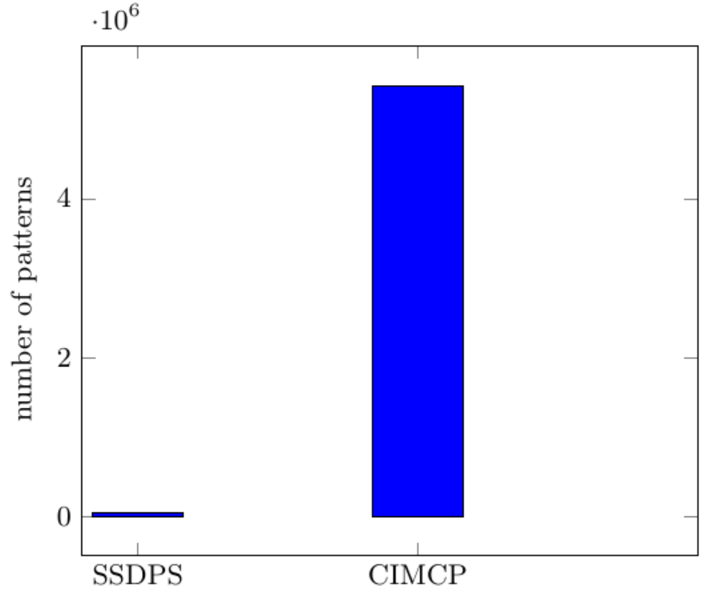


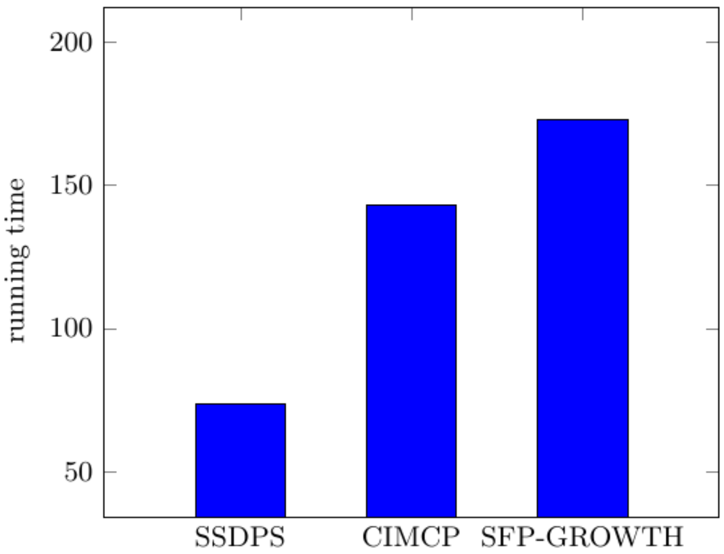


Setup 1: OR = 2; Setup 2: OR = 2, LCI_OR = 2

Experimental results

- Compare to other algorithms





Algorithms	**Measure**	**Threshold**	**#Patterns**	**Time(seconds)**
SSDPS	OR, LCI_ORS	$\alpha = 2$, $\beta = 2$	49,807	73.69
CIMCP	Chi-square	2	5,403,688	143
SFP-GROWTH	-log(p_value)	3	*	$>$ 172 (out of memory)

Conclusion

- Proposed a novel enumeration strategy in which discriminative measures and confidence interval are used as anti-monotonic property.

- Perspectives:
  - Heuristic search
  - Integrate domain knowledge in data mining
  - Apply other techniques to further remove uninteresting patterns

Reviewers’ comments

- Technical details:
  - Why do we use Galois connection ?
  - What are the different discriminative measures for ?
  - Parameters used in the algorithm, i.e., alpha and beta ?
- Experimental results:
  - Difficult to see the influence of the choice of measures
- Related work: lack of exposition

THANK YOU VERY MUCH

FOR YOUR LISTENING