



Universiteit Utrecht

[Faculty of Science
Information and Computing Sciences]

Pattern Subset Selection

Jilles Vreeken
Algorithmic Data Analysis group
Universiteit Utrecht

The Pattern Mining Question

- For a database db
 - a pattern language \mathcal{P} and
 - a set of constraints \mathcal{C}
- Find the set of patterns $\mathcal{S} \subseteq \mathcal{P}$ such that
 - each $p \in \mathcal{P}$ satisfies each $c \in \mathcal{C}$ on db
 - \mathcal{S} is maximal
- That is, we want *all* patterns that satisfy the constraints



Careful what you wish for...

- The pattern explosion:
 - High thresholds: few, well-known patterns
 - Low thresholds: a gazillion patterns
- Many patterns seem redundant
- The set of patterns is unstable
 - Small data change: different results



Boiling them Down

There are many proposals to deal with these problems:

- Condensed representations
 - The complete set of patterns can be reconstructed
 - E.g. closed patterns, non-derivable patterns
- Quality measures, such as lift
 - Based on the intended application
- All these approaches consider *individual* patterns



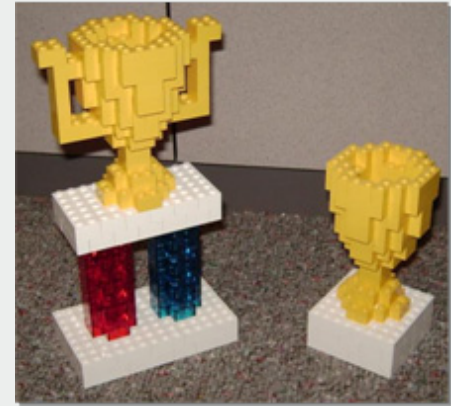
Nothing but the Best

- The root of all this evil is:
 - We ask for *all* patterns that satisfy some constraints
 - While, at the same time, we want a small set
- In other words, we want a *set of patterns* such that:
 - All members of the set satisfy the constraints
 - The set is *optimal* with regard to some criterion



Optimal? Come Again?

- You mean, the pattern set that best...
 - describes the (distribution of the) data,
 - separates the data into its classes,
 - covers the 1's with fewest patterns,
 - splits the data into non-overlapping parts,
 - complies with a number of constraints on the pattern set,
 - or, what exactly?



Crossroads

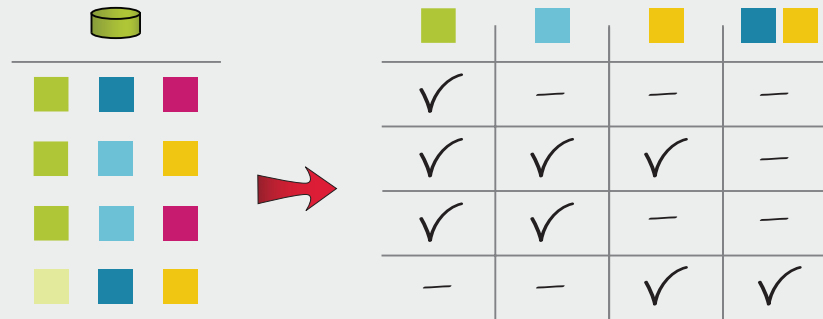
- View A: Lossy
 - Only the 'best' patterns, ignore 'noise'
 - Don't have to cover every row
 - Don't have to cover every attribute

- View 1: Lossless
 - All of the data is potentially interesting
 - Every row fully covered
 - Very exploratory in nature



Regarding Patterns as Features

- Patterns match a data row, or they don't
 - It's a binary feature!
 - Pattern subset selection resembles feature selection








- Many selection criteria make sense, e.g.
 - Maximise accuracy: identify classes
 - Maximise entropy: carve up the data space



Selecting from Pattern Feature Space

■ Maximally Informative k -Itemsets

- k patterns such that their joint entropy is maximal
- Miki's – Knobbe & Ho, KDD'06

			 	$\{\text{green}, \text{blue}, \text{yellow}\}$	$\{\text{blue}, \text{yellow}\}$
✓	–	–	–		
✓	✓	✓	–	$\frac{0}{1} \mid \frac{3}{0}$	$\frac{1}{1} \mid \frac{1}{1}$
✓	✓	–	–		
–	–	✓	✓	bad	good

■ More abstract: criterion as parameter

- Exhaustive:
Pattern Teams - Knobbe & Ho, PKDD'06
- Greedy:
Chosen Few - Bringmann & Zimmermann, ICDM'07



Some Lossy Discussion

- Freedom!
 - Independent of pattern syntax
 - Criterion can be chosen: accuracy, mutual information, joint entropy, ...
- Very strong reduction!
- Pattern set cannot explain everything
 - Important interactions may be missed
- Search either exhaustive or very greedy



Full Coverage

■ Pattern sets that cover all of the data

- Syntax matters!
- 0/1 data, itemsets

■ 'Colouring' the database:

A	B	C	D	E	F	G	H	Patterns
Orange	Orange	Light Green	Light Green	Blue	Light Green	Light Green	Light Green	Orange AB
Orange	Orange				Light Blue		Light Yellow	Light Green C D F G H
Orange	Orange	Light Green	Light Green		Light Green	Light Green	Light Green	Magenta B D E G
Orange	Orange	Light Green	Light Green		Light Green	Light Green	Light Green	Light Yellow H
	Magenta		Magenta	Magenta	Light Blue	Magenta		Blue E
	Magenta		Magenta	Magenta		Magenta	Light Yellow	Light Blue F



Selection in Data Space

- Finding large areas of 1s
 - Tiling – Geerts, Goethals & Mielikäinen, DS'04
 - NP-hard, good & fast approximation

- Finding the best data description
 - KRIMP – Siebes, Vreeken & van Leeuwen, SDM'06
 - MDL (compression) considers quality *and* complexity
 - For discrete data: itemsets, sequences, graphs



Discussing Lossless Pattern Sets

Wow!

- High quality data descriptions
- Global models naturally formed

However...

- Results in more patterns than lossy
- Computationally complex



Bridging the Gap?

■ Grand Unifying Theory?

- Major stumbling block: pattern usage
- Conceptual bridge difficult to build



■ General framework holds promise

- Constraint-based Pattern Set Mining, De Raedt & Zimmermann, SDM'07
- Pattern set properties depend on the user



the Future, and so on

- Pandora's box only just been opened
 - Lots of ground to be covered
 - Feature selection as Pattern selection?
- Scale remains an issue
 - Industrial strength techniques and implementations: millions of transactions, billions of patterns
- Upcoming:
 - How do different selection criteria perform?
 - Which, and how much data?
 - Keeping the user in the selection loop



Thank you for your attention!

■ Any questions?



The Bigger Picture

Approach	Lossy	Optimal	Data	0/1 Symmetric	# Patterns	Overfitting Sensitive?
Miki's	Yes	Yes	Any	Yes	k (2~5)	Yes
Pattern Teams	Yes	Yes	Any	Yes	k (3)	Yes
Chosen Few	Yes	No	Any	Yes	10~100s	Yes
Krimp	No	No	0/1	No	100~1000s	No
Tiling	Can	No	0/1	No	k to 100s	Yes
CB Pattern Sets	Might	Yes	Any	Can	Depends	Perhaps

