Frequent Regular Itemset Mining

Salvatore Ruggieri

Dipartimento di Informatica, Università di Pisa,
Largo B. Pontecorvo 3, 56127 Pisa, Italy
`ruggieri@di.unipi.it`

ACM SIGKDD 2010 - Washington DC, USA

Motivation

Concise representations of frequent itemsets:

- alleviate the problems due to extracting, storing and post-processing a huge amount of frequent patterns.
  - closed, free ($+$ negative border), non-derivable, disjunctive, ...

Motivation

Concise representations of frequent itemsets:

- alleviate the problems due to extracting, storing and post-processing a huge amount of frequent patterns.
  - closed, free (+ negative border), non-derivable, disjunctive, ...
- through a compact, lossless representation, where itemsets whose support is derivable from others are pruned away

Motivation

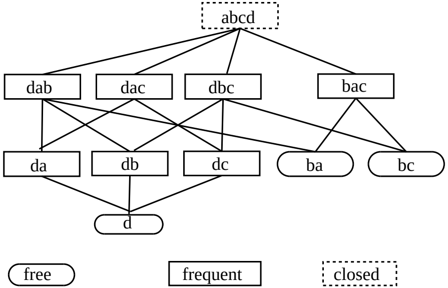
Concise representations of frequent itemsets:

- alleviate the problems due to extracting, storing and post-processing a huge amount of frequent patterns.
  - closed, free ($+$ negative border), non-derivable, disjunctive, ...
- through a compact, lossless representation, where itemsets whose support is derivable from others are pruned away
- *at the cost of sacrificing readability and direct interpretability by a data analyst!*

Motivation

tid	transaction
1	*abcde*
2	*abcd*
3	*b*
4	*ac*

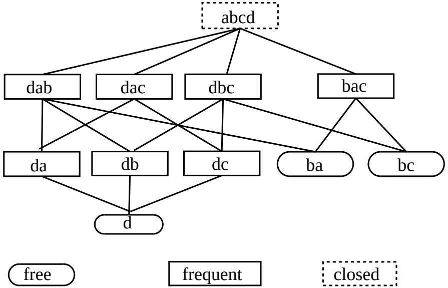
cover	support	closed	free
$\{1,2\}$	2	*abcd*	*d* *ba* *bc*
$\{1,2,3\}$	3	*b*	*b*
$\{1,2,4\}$	3	*ac*	*a* *c*



Motivation

tid	transaction
1	*abcde*
2	*abcd*
3	*b*
4	*ac*

cover	support	closed	free
$\{1,2\}$	2	*abcd*	*d ba bc*
$\{1,2,3\}$	3	*b*	*b*
$\{1,2,4\}$	3	*ac*	*a c*

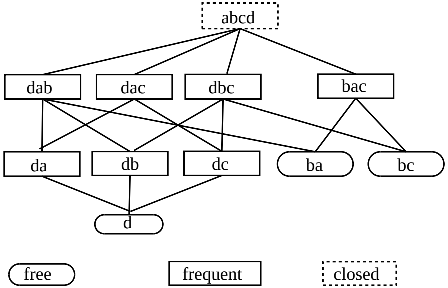


What itemsets are represented by *abcd*?

Motivation

tid	transaction
1	*abcde*
2	*abcd*
3	*b*
4	*ac*

cover	support	closed	free
$\{1,2\}$	2	*abcd*	*d* *ba* *bc*
$\{1,2,3\}$	3	*b*	*b*
$\{1,2,4\}$	3	*ac*	*a* *c*



What itemsets are represented by *abcd*?

$$Pow(abcd) \setminus \bigcup_{Y \in \mathcal{CS}, support(Y) > support(abcd)} Pow(Y)$$

Contribution

Problem: itemsets represented by a closed itemset (its semantics) are not derivable from it in isolation.

Contribution

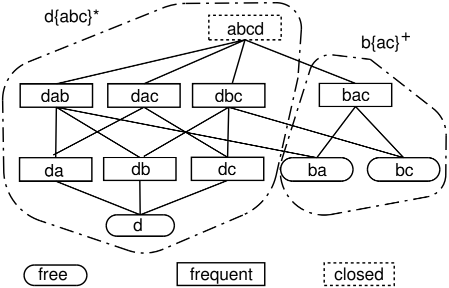
Problem: itemsets represented by a closed itemset (its semantics) are not derivable from it in isolation.

Contribution: an extension of itemsets, called regular, with an immediate semantics and interpretability, and a conciseness comparable to closed itemsets.

Contribution

Problem: itemsets represented by a closed itemset (its semantics) are not derivable from it in isolation.

Contribution: an extension of itemsets, called regular, with an immediate semantics and interpretability, and a conciseness comparable to closed itemsets.



Basic definitions

set of items $\mathcal{I}$

- transaction (tid, X) with $X \subseteq \mathcal{I}$
- $cover(I) = \{tid \mid (tid, X) \in \mathcal{D}, X \subseteq I\}$
- $support(I) = |cover(I)|$.
- frequent itemsets $\mathcal{F} = \{X \subseteq \mathcal{I} \mid support(X) \geq minsupp\}$.

Basic definitions

set of items $\mathcal{I}$

- transaction (tid, X) with $X \subseteq \mathcal{I}$
- $cover(I) = \{tid \mid (tid, X) \in \mathcal{D}, X \subseteq I\}$
- $support(I) = |cover(I)|$.
- frequent itemsets $\mathcal{F} = \{X \subseteq \mathcal{I} \mid support(X) \geq minsupp\}$.

θ-equivalence

- relation: $X \theta Y$ if $cover(X) = cover(Y)$.
- classes: $[X] = \{Y \subseteq \mathcal{I} \mid X \theta Y\}$.
- closed itemsets $Y \in \mathcal{CS}$ iff $\{Y\} = max[X]$ for some X.
- free itemsets $Y \in \mathcal{FS}$ iff $Y \in min[X]$ for some X.

Extended itemsets: syntax

The set $\mathcal{J}$ of *extended items* is defined as follows:

$$E ::= a \mid a? \mid \{a_1, \ldots, a_h\}^\star \mid \{a_1, \ldots, a_k\}^+$$

where a, a_i's are items, $h \geq 0$ and $k > 0$.

An *extended itemset* is a subset $R \subseteq \mathcal{J}$.

Ex. The intended meaning of $ab\{cd\}^\star$ is

$$\{ab, abc, abd, abcd\}$$

The intended meaning of $ab?\{cd\}^+$ is

$$\{ac, ad, acd, abc, abd, abcd\}$$

Extended itemsets: semantics

Semantics $s_e() : \mathcal{J} \rightarrow Pow(Pow(\mathcal{I}))$ for extended items :

$$\begin{array}{rcl} s_e(a) & = & \{\{a\}\} \\ s_e(a?) & = & \{\{a\}, \emptyset\} \\ s_e(\{a_1, \ldots, a_h\}^\star) & = & \{X \mid X \subseteq \{a_1, \ldots, a_h\}\} \\ s_e(\{a_1, \ldots, a_k\}^+) & = & \{X \mid X \subseteq \{a_1, \ldots, a_k\}, X \neq \emptyset\}. \end{array}$$

Semantics $s() : Pow(\mathcal{J}) \rightarrow Pow(Pow(\mathcal{I}))$ for extended itemsets:

$$s(e_1, \ldots, e_n) = \{ \cup_{i=1 \ldots n} X_i \mid X_i \in s(e_i), i = 1 \ldots n\}.$$

Extended itemsets: semantics

Semantics $s_e() : \mathcal{J} \rightarrow Pow(Pow(\mathcal{I}))$ for extended items :

$$
\begin{array}{rcl}
s_e(a) & = & \{\{a\}\} \\
s_e(a?) & = & \{\{a\}, \emptyset\} \\
s_e(\{a_1, \dots, a_h\}^\star) & = & \{X \mid X \subseteq \{a_1, \dots, a_h\}\} \\
s_e(\{a_1, \dots, a_k\}^+) & = & \{X \mid X \subseteq \{a_1, \dots, a_k\}, X \neq \emptyset\}.
\end{array}
$$

Semantics $s() : Pow(\mathcal{J}) \rightarrow Pow(Pow(\mathcal{I}))$ for extended itemsets:

$$s(e_1, \dots, e_n) = \{ \cup_{i=1 \dots n} X_i \mid X_i \in s(e_i), i = 1 \dots n\}.$$

$s()$ is and-compositional: the meaning of an extended itemset can be obtained by looking (only) at the meaning of its items!

Regular itemsets

Ex. Let $\mathcal{D} = \{(1, ab), (2, a)\}$, and $R = ab?$. We have: $s(R) = \{a, ab\}$ and

$$cover(a) = \{1, 2\} \neq \{1\} = cover(ab)$$

Regular itemsets

Ex. Let $\mathcal{D} = \{(1, ab), (2, a)\}$, and $R = ab?$. We have: $s(R) = \{a, ab\}$ and

$$cover(a) = \{1, 2\} \neq \{1\} = cover(ab)$$

Extended itemsets are relevant to the FIM problem only when they denote itemsets with a common cover.

Regular itemsets

Ex. Let $\mathcal{D} = \{(1, ab), (2, a)\}$, and $R = ab?$. We have:
$s(R) = \{a, ab\}$ and

$$cover(a) = \{1, 2\} \neq \{1\} = cover(ab)$$

Extended itemsets are relevant to the FIM problem only when they denote itemsets with a common cover.

Def. An extended itemset R is said *regular* if
for every $X, Y \in s(R)$ we have that $cover(X) = cover(Y)$.

Regular itemsets

Ex. Let $\mathcal{D} = \{(1, ab), (2, a)\}$, and $R = ab?$. We have:
$s(R) = \{a, ab\}$ and

$$cover(a) = \{1, 2\} \neq \{1\} = cover(ab)$$

Extended itemsets are relevant to the FIM problem only when they denote itemsets with a common cover.

Def. An extended itemset R is said *regular* if
for every $X, Y \in s(R)$ we have that $cover(X) = cover(Y)$.

Other equivalent formulations:

- if $s(R) \subseteq [X]$ for some itemset X,
- if for every $X, Y \in s(R)$, $support(X) = support(Y)$.

Regular itemsets: concise representation

For a regular itemset R, we define

$$\mathit{cover}(R) = \mathit{cover}(X) \quad \text{and} \quad \mathit{support}(R) = |\mathit{cover}(R)|$$

where X is any element in $s(R)$.

Regular itemsets: concise representation

For a regular itemset R, we define

$$cover(R) = cover(X) \quad \text{and} \quad support(R) = |cover(R)|$$

where X is any element in $s(R)$.

Def. A finite set of regular itemsets $\mathcal{R}$ is a concise repr. of $\mathcal{F}$ if:

(a) $\cup_{R \in \mathcal{R}} s(R) = \mathcal{F}$, and

(b) for every pair $R_1 \neq R_2 \in \mathcal{R}$, $s(R_1) \cap s(R_2) = \emptyset$.

Regular itemsets: concise representation

For a regular itemset R, we define

$$cover(R) = cover(X) \quad \text{and} \quad support(R) = |cover(R)|$$

where X is any element in $s(R)$.

Def. A finite set of regular itemsets $\mathcal{R}$ is a concise repr. of $\mathcal{F}$ if:

(a) $\cup_{R \in \mathcal{R}} s(R) = \mathcal{F}$, and

(b) for every pair $R_1 \neq R_2 \in \mathcal{R}$, $s(R_1) \cap s(R_2) = \emptyset$.

How large is a concise representation $\mathcal{R}$?

Regular itemsets: concise representation

For a regular itemset R, we define

$$cover(R) = cover(X) \quad \text{and} \quad support(R) = |cover(R)|$$

where X is any element in $s(R)$.

Def. A finite set of regular itemsets $\mathcal{R}$ is a concise repr. of $\mathcal{F}$ if:

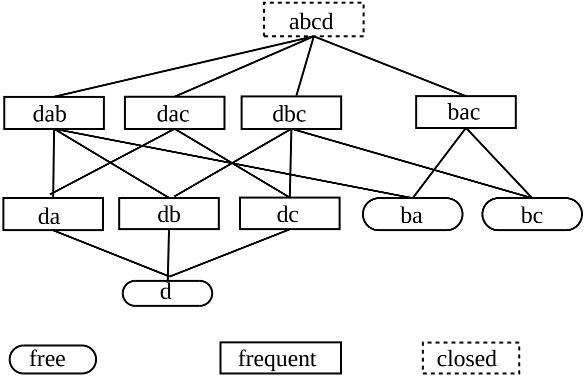
(a) $\cup_{R \in \mathcal{R}} s(R) = \mathcal{F}$, and

(b) for every pair $R_1 \neq R_2 \in \mathcal{R}$, $s(R_1) \cap s(R_2) = \emptyset$.

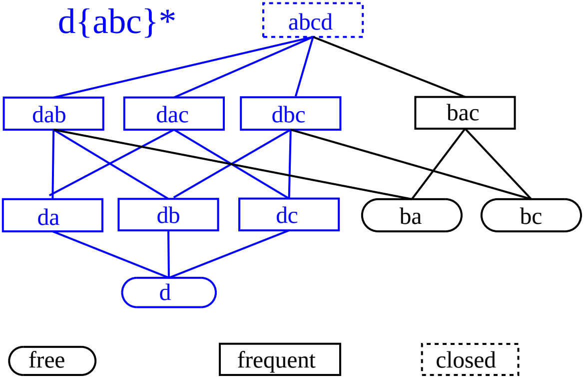
How large is a concise representation $\mathcal{R}$?

$$|\mathcal{CS}| \leq |\mathcal{R}|, \text{ but, in practice, } |\mathcal{CS}| \approx |\mathcal{R}|$$

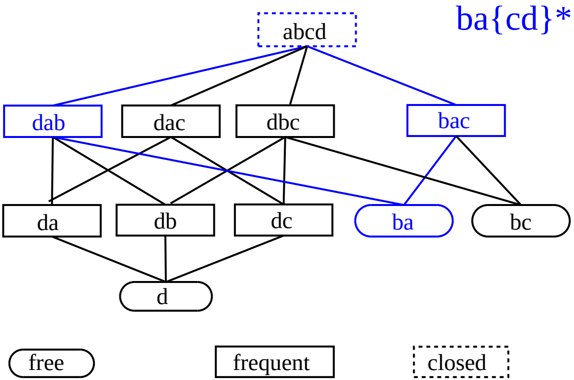
Towards mining a concise representation



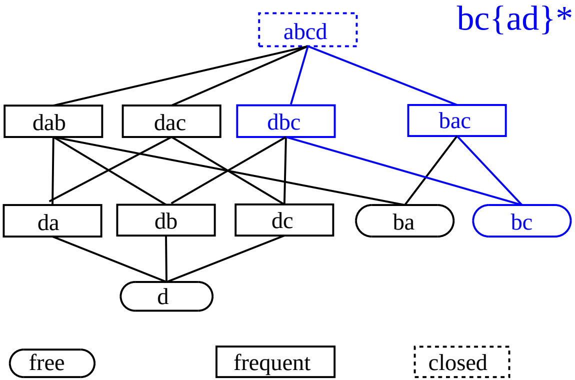
Towards mining a concise representation



Towards mining a concise representation



Towards mining a concise representation



Towards mining a concise representation

The (semantics of the) extended itemsets

$$d\{abc\}^\star \quad ba\{cd\}^\star \quad bc\{ad\}^\star$$

are not pair-wise disjoint!

Towards mining a concise representation

The (semantics of the) extended itemsets

$$d\{abc\}^{\star} \quad ba\{cd\}^{\star} \quad bc\{ad\}^{\star}$$

are not pair-wise disjoint!

We would like to express

$$s(ba\{cd\}^{\star}) \setminus s(d\{abc\}^{\star})$$

and

$$s(bc\{ad\}^{\star}) \setminus (s(d\{abc\}^{\star}) \cup s(bc\{ad\}^{\star}))$$

Non-compositional itemsets

Non-compositional items are extended items plus:

$$E \ ::= \{a_1, \ldots, a_h\}^-$$

where $h \geq 0$, with the following semantics:

$$s_e(\{a_1, \ldots, a_h\}^-) \quad = \quad \{X \mid X \subset \{a_1, \ldots, a_h\}\}.$$

Since we do expect $b \notin s(b\{b\}^-)$, we define:

$$s'(e_1, \ldots, e_n) = \{X \in s(e_1, \ldots, e_n) \mid X \cap Y \subset Y \\ \text{for every } e_i \text{ of the form } Y^-\}.$$

$s'()$ is not and-compositional.

Mining a concise representation

Let C be a closed itemset, and $X_1, \ldots, X_n$ be its free itemsets.

A concise representation of $[C]$ is provided by

$$N_i = X_i, X_1^-, \ldots, X_{i-1}^-, (C \setminus X_i)^\star$$

for $i = 1 \ldots n$.

Mining a concise representation

Let C be a closed itemset, and $X_1, \ldots, X_n$ be its free itemsets.

A concise representation of $[C]$ is provided by

$$N_i = X_i, X_1^-, \ldots, X_{i-1}^-, (C \setminus X_i)^\star$$

for $i = 1 \ldots n$.

$$d\{abc\}^\star \quad ba\{d\}^-\{cd\}^\star \quad bc\{d\}^-\{ba\}^-\{ad\}^\star$$

Mining a concise representation

Let C be a closed itemset, and $X_1, \ldots, X_n$ be its free itemsets.

A concise representation of $[C]$ is provided by

$$N_i = X_i, X_1^-, \ldots, X_{i-1}^-, (C \setminus X_i)^\star$$

for $i = 1 \ldots n$.

$$d\{abc\}^\star \quad ba\{d\}^-\{cd\}^\star \quad bc\{d\}^-\{ba\}^-\{ad\}^\star$$

Next problem: rewrite $N_1, \ldots, N_n$ into a set of equivalent pair-wise disjoint regular itemsets.

Mining a concise representation

$$\frac{ba\{d\}^{-}\{cd\}^{\star}}{bac?}\,\mathbf{S4}$$

Mining a concise representation

$$\frac{ba\{d\}^{-}\{cd\}^{\star}}{bac?}\,\mathbf{S4}$$

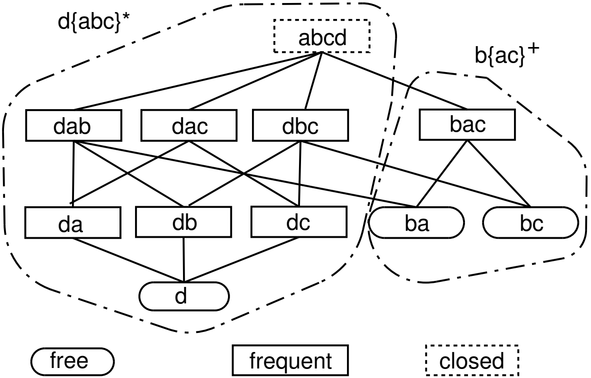
$$\cfrac{\cfrac{\cfrac{bc\{d\}^{-}\{ba\}^{-}\{ad\}^{\star}}{bc\{d\}^{-}\{a\}^{-}\{ad\}^{\star}}\,\mathbf{S1}}{bc\{a\}^{-}a?}\,\mathbf{S4}}{bc}\,\mathbf{S4}$$

Mining a concise representation

$$\frac{ba\{d\}^{-}\{cd\}^{\star}}{bac?}\textbf{S4} \qquad \dfrac{\dfrac{\dfrac{bc\{d\}^{-}\{ba\}^{-}\{ad\}^{\star}}{bc\{d\}^{-}\{a\}^{-}\{ad\}^{\star}}\textbf{S1}}{bc\{a\}^{-}a?}\textbf{S4}}{bc}\textbf{S4}$$

$$\frac{bac? \quad bc}{b\{ac\}^{+}}\textbf{M3}$$

Towards mining a concise representation



Splitting rules

$$\frac{R, X, Y^- \quad Y \cap X \neq \emptyset}{R, X, (Y \setminus X)^-}\textbf{S1} \qquad \frac{R, X, Z^\star \quad Z \cap X \neq \emptyset}{R, X, (Z \setminus X)^\star}\textbf{S2}$$

$$\frac{R, \emptyset^-}{\textbf{fail}}\textbf{S3} \qquad \frac{R, \{a\}^- \quad a \notin R}{R \setminus \{a?\}[\{a, X\}^- \to X^\star]}\textbf{S4}$$

$$\frac{R, \{a, Y\}^- \quad a \notin R \quad Y \neq \emptyset}{R \setminus \{a?\}[\{a, X\}^- \to X^\star], Y^\star \quad R \setminus \{a?\}[\{a, X\}^- \to X^-], a, Y^-}\textbf{S5}$$

Rewritings implemented as procedure **Covering** (see paper).

Splitting rules

$$\frac{cd\{ab\}^{-}\{ab\}^{\star}}{}$$

Splitting rules

First partition $s'(cd\{ab\}^{-}\{ab\}^{\star}) \cap \{X \subseteq \mathcal{I} \mid a \notin X\}$

$$\dfrac{cd\{ab\}^{-}\{ab\}^{\star}}{cdb?}\;\mathbf{S5}$$

Splitting rules

Second partition $s'(cd\{ab\}^{-}\{ab\}^{\star}) \cap \{X \subseteq \mathcal{I} \mid a \in X\}$

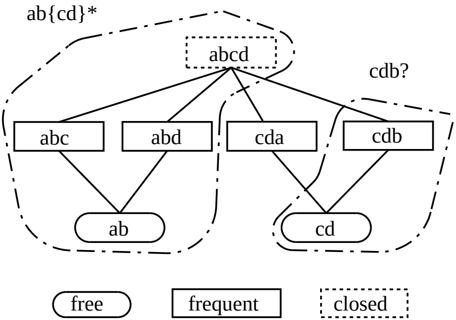
$$\dfrac{cd\{ab\}^{-}\{ab\}^{\star}}{cdb? \qquad \dfrac{cda\{b\}^{-}b?}{}}\,\mathbf{S5}$$

Splitting rules

$$\dfrac{cd\{ab\}^{-}\{ab\}^{\star}}{cdb? \qquad \dfrac{cda\{b\}^{-}b?}{cda}\,\mathbf{S4}}\,\mathbf{S5}$$

Splitting rules

$$\dfrac{cd\{ab\}^{-}\{ab\}^{\star}}{cdb? \qquad \dfrac{cda\{b\}^{-}b?}{cda}\,\mathbf{S4}}\,\mathbf{S5}$$



Merging rules

$$\frac{R \quad R, a}{R, a?}\,\mathbf{M1} \qquad \frac{R \quad R, Y^{+}}{R, Y^{\star}}\,\mathbf{M2}$$

$$\frac{R, b, a? \quad R, a}{R, \{a, b\}^{+}}\,\mathbf{M3} \qquad \frac{R, Y^{+}, a? \quad R, a}{R, \{a, Y\}^{+}}\,\mathbf{M4}$$

$$\frac{R, Y^{+} \quad R, a, Y^{\star}}{R, \{a, Y\}^{+}}\,\mathbf{M5} \qquad \frac{R, Y^{+} \quad R, Z^{+}, Y^{\star}}{R, \{Z, Y\}^{+}}\,\mathbf{M6}$$

Rewritings implemented as procedure **Merging** (see paper).

Frequent Regular Itemsets Mining

Algorithm RegularMine

Input: a transactional database $\mathcal{D}$
Output: a set $\mathcal{R}_{out}$ of frequent regular itemsets that is a concise representation of frequent itemsets

extract frequent closed itemsets $\mathcal{CS}$ from $\mathcal{D}$
and, for each $C \in \mathcal{CS}$, the free sets in $[C]$
$\mathcal{R}_{out} \leftarrow \emptyset$
for every $C \in \mathcal{CS}$ **do**
let $X_1, \ldots, X_n$ be the free sets in $[C]$ ordered w.r.t. $\preceq$
$\mathcal{R} = \cup_{i=1 \ldots n} Covering(X_i, X_1^-, \ldots, X_{i-1}^-, C^\star)$ // rules $S1 - S5$
$\mathcal{R}_{out} \leftarrow \mathcal{R}_{out} \cup Merging(\mathcal{R})$ // rules $M1 - M6$
end for

Nondeterminism

Nondeterminism I: The splitting and merging rules are non-deterministic. The procedures **Covering** and **Merging** adopt a few heuristics to drive the rewriting.

Nondeterminism

Nondeterminism I: The splitting and merging rules are non-deterministic. The procedures **Covering** and **Merging** adopt a few heuristics to drive the rewriting.

Nondeterminism II: The order $X_1, \ldots, X_n$ affects the (size of the) output.

Nondeterminism

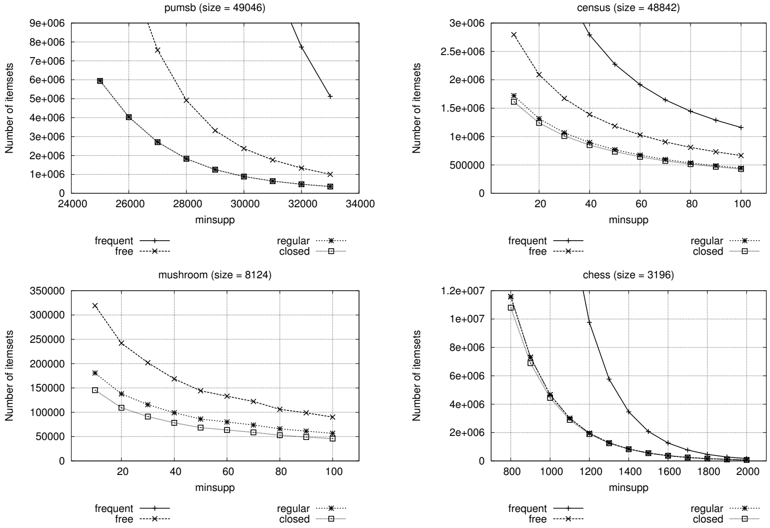
Nondeterminism I: The splitting and merging rules are non-deterministic. The procedures **Covering** and **Merging** adopt a few heuristics to drive the rewriting.

Nondeterminism II: The order $X_1, \ldots, X_n$ affects the (size of the) output.

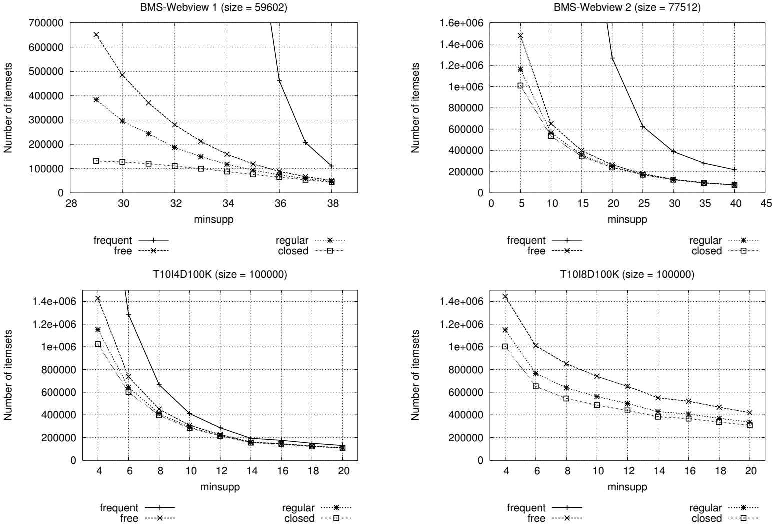
We (experimentally) resort to [Dong et al. 2005]:

Def. $X_i \preceq X_j$ iff $|X_i| < |X_j|$ or, $|X_i| = |X_j|$ and $X_i \preceq_{lex} X_j$ where $\preceq_{lex}$ is a lexicographic order induced by a total order items.

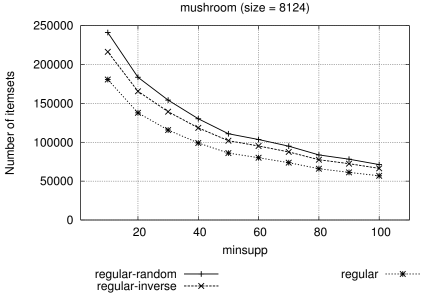
Experimental results: dense datasets

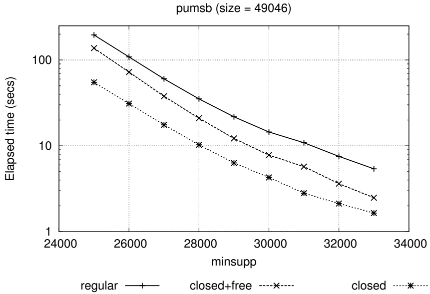


Experimental results: sparse datasets



Experimental results: orderings and execution time





Conclusion and future work

Contribution:

- regular itemsets as an easy-to-understand concise representation
- **RegularMine** to mine frequent regular itemsets

Conclusion and future work

Contribution:

- regular itemsets as an easy-to-understand concise representation
- **RegularMine** to mine frequent regular itemsets

Future work:

- pushing **RegularMine** inside closed and free itemsets extraction
- use of regular itemsets in non-redundant association rules and in case studies