MFI Pincer Search Trace

A1	A2	A3	A4	A5	A6	A7	A8	A9
1	0	0	0	1	1	0	1	0
0	1	0	1	0	0	0	1	0
0	0	0	1	1	0	1	0	0
0	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	0	0
0	1	1	1	0	0	0	0	0
0	1	0	0	0	1	1	0	1
0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	1	0
0	0	1	0	1	0	1	0	0
0	0	1	0	1	0	1	0	0
0	0	0	0	1	1	0	1	0
0	1	0	1	0	1	1	0	0
1	0	1	0	1	0	1	0	0
0	1	1	0	0	0	0	0	1

Pincer-Search Method

$L_0 := \varnothing$; $k := 1$; $C_1 := \{\{i\} \mid i \in I\}$; $S_0 = \varnothing$;
MFCS := $\{\{1,2, \ldots, n\}\}$; MFS := $\varnothing$;
do until $C_k = \varnothing$ and $S_{k-1} = \varnothing$
read database and count supports for C_k and MFCS;
MFS := MFS $\cup$ {frequent itemsets in MFCS};
S_k := {infrequent itemsets in C_k};
call *MFCS-gen* algorithm if $S_k \neq \varnothing$;
call *MFS-pruning* procedure;
generate candidates C_{k+1} from C_k; (similar to a priori's generate & prune)
if any frequent itemset in C_k is removed in MFS-pruning procedure
call the recovery procedure to recover candidates to C_{k+1};
call *MFCS prune procedure* to prune candidates in C_{k+1};
$k := k+1$;
return MFS

MFCS-gen

for all itemsets $s \in S_k$
for all itemsets $m \in$ MFCS
if s is a subset of m
MFCS := MFCS\{m};
for all items $e \in$ itemset s
if $m\backslash\{e\}$ is not a subset of any itemset in MFCS
MFCS := MFCS $\cup \{m\backslash\{e\}\}$;
return MFCS

for all itemsets $l \in C_k$

for all itemsets $m \in$ MFS

if the first $k-1$ items in l are also in m

/* suppose $m.\text{item}_j = l.\text{item}_{k-1}$ */

for i from $j+1$ to $|m|$

$C_{k+1} := C_{k+1} \cup \{\{l.\text{item}_1, l.\text{item}_2, \ldots, l.\text{item}_k, m.\text{item}_i\}\}$

MFS-Prune

for all itemsets c in C_k

if c is a subset of any itemset is the current MFS

delete c from C_k;

MFCS-Prune

for all itemsets c in C_{k+1}

if c is not a subset of any itemset in the current MFCS

delete c from C_{k+1};

STEP 1: $L_0 := \emptyset$; $k := 1$;

$C_1 := \{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}, \{7\}, \{8\}, \{9\}\}$

MFCS := {1, 2, 3, 4, 5, 6, 7, 8, 9}

MFS := ∅;

PASS ONE: Database is read to count the support as follows

{1} → 2, {2} → 6, {3} → 6, {4} → 4, {5} → 8, {6} → 5, {7} → 7, {8} → 4, {9} → 2

{1, 2, 3, 4, 5, 6, 7, 8, 9} → 0.

So MFCS := {1, 2, 3, 4, 5, 6, 7, 8, 9} and MFS := ∅;

$L_1 := \{\{2\}, \{3\}, \{4\}, \{5\}, \{6\}, \{7\}, \{8\}\}$

$S_1 := \{\{1\}, \{9\}\}$

At this stage we call the MFCS-gen to update MFCS.

For {1} in S_1 and for {1, 2, 3, 4, 5, 6, 7, 8, 9} in MFCS, we get the new element in MFCS as {2, 3, 4, 5, 6, 7, 8, 9}.

For {9} in S_1 and for {2, 3, 4, 5, 6, 7, 8, 9} in MFCS, we get the new element in MFCS as {2, 3, 4, 5, 6, 7, 8}.

We generate the candidate itemsets

$C_2 := \{$ {2,3}, {2,4}, {2,5}, {2,6}, {2,7}, {2,8}, {3,4}, {3,5}, {3,6}, {3,7}, {3,8},

C_2 := { {2,3}, {2,4}, {2,5}, {2,6}, {2,7}, {2,8}, {3,4}, {3,5}, {3,6}, {3,7}, {3,8}, {4,5}, {4,6}, {4,7}, {4,8}, {5,6}, {5,7}, {5,8}, {6,7}, {6,8}, {7,8} }

PASS TWO: read the database to count the support of elements in C_2 and MFCS as given below:

{2,3}→3, {2,4}→3, {2,5}→0, {2,6}→2, {2,7}→2, {2,8}→1, {3,4}→1, {3,5}→3, {3,6}→0, {3,7}→3, {3,8}→0, {4,5}→1, {4,6}→1, {4,7}→2, {4,8}→1, {5,6}→3, {5,7}→5, {5,8}→2, {6,7}→3, {6,8}→2, {7,8}→0

{2, 3, 4, 5, 6, 7, 8}→0.

MFS:= ∅;

L_2 := { {2,3}, {2,4}, {3,5}, {3,7}, {5,6}, {5,7}, {6,7} }

S_2 := { {2,5}, {2,6}, {2, 7} {2,8}, {3,4}, {3,6}, {3,8}, {4,5}, {4,6}, {4,7}, {4,8}, {5,8}, {6,8}, {7,8} }

For {2,5} in S_2 and for {2, 3, 4, 5, 6, 7, 8} in MFCS, we get the new elements in MFCS as {3, 4, 5, 6, 7, 8} and {2, 3, 4, 6, 7, 8}

For {2,6} in S_2 and for {3, 4, 5, 6, 7, 8} in MFCS, since {2,6} is not contained in this element of MFCS and hence, no action.

For {2, 3, 4, 6, 7, 8} we get two new elements in MFCS in place of {2, 3, 4, 6, 7, 8} as {3, 4, 6, 7, 8} and {2, 3, 4, 7, 8}. Since {3, 4, 6, 7, 8} is already contained in an element of MFCS, it is excluded from MFCS.

So at this stage MFCS := {{ 3, 4, 5, 6, 7, 8}, {2, 3, 4, 7, 8}}.

For {2,7} in S_2, we get

MFCS := {{3, 4, 5, 6, 7, 8}, {2, 3, 4, 8}}.

For {2,8} in S_2, we get

MFCS := {{3, 4, 5, 6, 7, 8}, {2, 3, 4}}.

For {3,4} in S_2, we get

MFCS := {{3, 5, 6, 7, 8}, {4, 5, 6, 7, 8}, {2, 3}, {2, 4}}.

For {3,6} in S_2, we get

MFCS := {{3, 5, 7, 8}, {4, 5, 6, 7, 8}, {2, 3}, {2, 4}}.

For {3,8} in S_2, we get

MFCS := {{3, 5, 7}, {4, 5, 6, 7, 8}, {2, 3}, {2, 4}}.

For {4,5} in S_2, we get

MFCS := {{3, 5, 7}, {5, 6, 7, 8}, {4, 6, 7, 8}, {2, 3}, {2, 4}}.

For {4,6} in S_2, we get

MFCS := {{3, 5, 7}, {5, 6, 7, 8}, {4, 7, 8}, {2, 3}, {2, 4}}.

For {4,7} in S_2, we get

MFCS := {{3, 5, 7}, {5, 6, 7, 8}, {4, 8}, {2, 3}, {2, 4}}.

For {4,8} in S_2, we get

MFCS := {{3, 5, 7}, {5, 6, 7, 8}, {2, 3}, {2, 4}}.

For {5,8} in S_2, we get

MFCS := {{3, 5, 7}, {6, 7, 8}, {5, 6, 7}, {2, 3}, {2, 4}}.

For {6,8} in S_2, we get

MFCS := {{7, 8}, {3, 5, 7}, {5, 6, 7}, {2, 3}, {2, 4}}.

For {7,8} in S_2, we get

MFCS := {{8}, {3, 5, 7}, {5, 6, 7}, {2, 3}, {2, 4}}.

We generate the candidate sets as

C_3 := {{2, 3, 4}, {3, 5, 7}, {5, 6, 7}}

In the pruning stage the itemsets {2, 3, 4} are pruned from C_3 and hence,

C_3 := {{3, 5, 7}, {5, 6, 7}}

At this stage we make one more pass of the database to count the supports of {{3, 5, 7}, {5, 6, 7}}.