

TDT4300 Datawarehouse and datamining

Association analysis
Assignment 2

Group: 99

Name: Erik Turøy Midtun

Submission date: February 17, 2021

1. Apriori Algorithm

| TID | Items |
|-----|------------|
| T1 | H, B, K |
| T2 | H, B |
| T3 | H, C, I |
| T4 | C, I |
| T5 | I, K |
| T6 | H, C, I, U |

Task a)

Show thoroughly the steps on how the frequent itemsets are generated:

| Item | Support count |
|------|---------------|
| B | 2 |
| C | 3 |
| H | 4 |
| I | 4 |
| K | 2 |
| U | 1 |

| Itemset | Support count |
|---------|---------------|
| {B,C} | 0 |
| {B,H} | 2 |
| {B,I} | 0 |
| {B,K} | 1 |
| {C,H} | 2 |
| {C,I} | 3 |
| {C,K} | 0 |
| {H,I} | 2 |
| {H,K} | 1 |
| {I,K} | 1 |

| Itemset | Support count |
|----------|---------------|
| {B,H,C} | 0 |
| {B,H,I} | 0 |
| {B,C,I} | 0 |
| {C,H, I} | 2 |

Frequent itemsets: {C,H, I}, {H,I}, {C,I}, {C,H}, {B,H}, {K}, {I}, {H}, {C}, {B}

(b)

We first find all 2^k-2 candidates from {H,C,I}:

HC->I HI->C CI->H C->HI H->CI I->HC

Then we calculate the confidence for all of them, and since all candidates include {H,C, I} we use the support count of 2 as the numerator from the confidence formula:

$$C(\{A\} \rightarrow \{B\}) = \frac{\sigma(\{A\} \cup \{B\})}{\sigma(\{A\})} \quad \text{where } \sigma \text{ is the support count}$$

| Candidate (\rightarrow) | | Support | Confidence | Accepted (threshold 0.6) |
|-----------------------------|----|---------|------------|--------------------------|
| HC | I | 2 | 1 | Yes |
| HI | C | 2 | 1 | Yes |
| CI | H | 3 | 0.66 | Yes |
| C | HI | 3 | 0.66 | Yes |
| H | CI | 4 | 0.5 | No |
| I | HC | 4 | 0.5 | No |

This gives us the following 4 association rules:

1. $\{H, C\} \rightarrow \{I\}$
2. $\{H, I\} \rightarrow \{C\}$
3. $\{C, I\} \rightarrow \{H\}$
4. $\{C\} \rightarrow \{H, I\}$

2. FP-Growth Algorithm

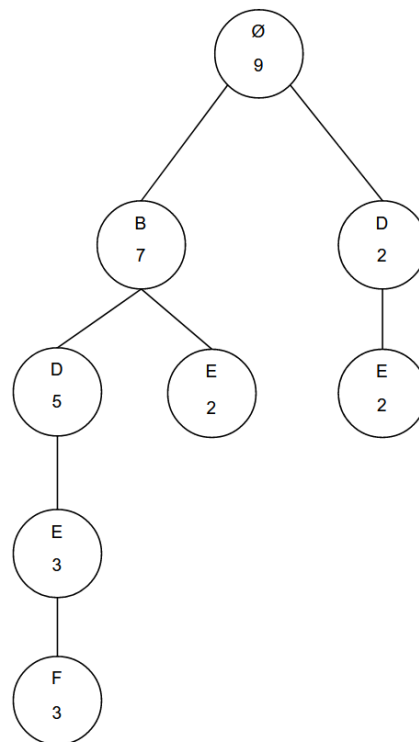
We find all the 1 itemset supports counts and sort descending. Then we remove the infrequent items marked in gray.

| Item | Support count |
|------|---------------|
| B | 7 |
| D | 7 |
| E | 7 |
| F | 3 |
| A | 1 |
| C | 1 |
| G | 1 |
| H | 1 |
| I | 1 |
| J | 1 |

Then we sort the items for each transaction in descending order of support without the infrequent items

| TID | items |
|-----|------------|
| T1 | b, e |
| T2 | b, d |
| T3 | b, d, e, f |
| T4 | d, e |
| T5 | d, e |
| T6 | b, d |
| T7 | b, d, e, f |
| T8 | b, d, e, f |
| T9 | b, e |

From this table of sorted transactions, we generate a frequent pattern tree:

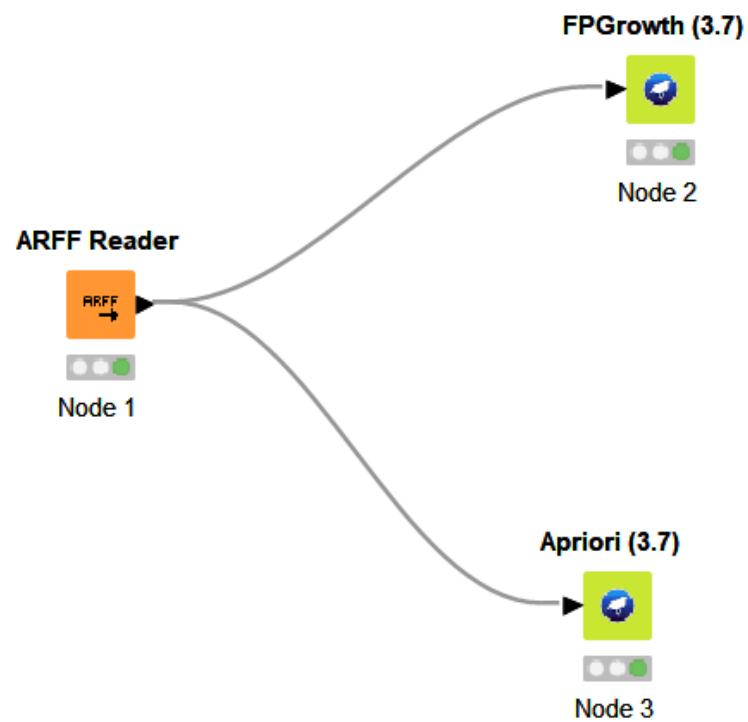


Minimum support count is 2

| Items | Conditional pattern base | Conditional FP tree | Frequent Patterns |
|-------|--------------------------|---------------------|---|
| F | BDE:3 | BDE:3 | {FB}:3, {FD}:3, {FE}:3, {FBD}:3, {FBE}:3, {FDE}:3, {FBDE}:3 |
| E | D:2, B:2, BD:3 | B:5, BD:3. D:2 | {ED}:2, {EB}:5, {ED}:3, {EBD}:3 |
| D | B:5 | B:5 | {DB}:5 |
| B | - | - | - |

3. KNIME

I added an ARFF reader and fed its output to the FPGrowth and Apriori Nodes as shown in the following KNIME Workflow



Screenshot of the KNIME workflow

```
Weka Node View - 3:3 - Apriori (3.7)
File
Apriori
=====
Minimum support: 0.75 (7 instances)
Minimum metric <confidence>: 0.8
Number of cycles performed: 5

Generated sets of large itemsets:

Size of set of large itemsets L(1): 4

Size of set of large itemsets L(2): 4

Size of set of large itemsets L(3): 1

Best rules found:

1. G=t 8 ==> C=t 8 <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
2. B=t 7 ==> C=t 7 <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
3. B=t 7 ==> G=t 7 <conf:(1)> lift:(1.25) lev:(0.14) [1] conv:(1.4)
4. H=t 7 ==> C=t 7 <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
5. B=t G=t 7 ==> C=t 7 <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
6. B=t C=t 7 ==> G=t 7 <conf:(1)> lift:(1.25) lev:(0.14) [1] conv:(1.4)
7. B=t 7 ==> C=t G=t 7 <conf:(1)> lift:(1.25) lev:(0.14) [1] conv:(1.4)
8. G=t 8 ==> B=t 7 <conf:(0.88)> lift:(1.25) lev:(0.14) [1] conv:(1.2)
9. C=t G=t 8 ==> B=t 7 <conf:(0.88)> lift:(1.25) lev:(0.14) [1] conv:(1.2)
10. G=t 8 ==> B=t C=t 7 <conf:(0.88)> lift:(1.25) lev:(0.14) [1] conv:(1.2)
```

```
Weka Node View - 3:2 - FPGrowth (3.7)
File
FPGrowth found 10 rules (displaying top 10)

1. [G=t]: 8 ==> [C=t]: 8 <conf:(1)> lift:(1) lev:(0) conv:(0)
2. [H=t]: 7 ==> [C=t]: 7 <conf:(1)> lift:(1) lev:(0) conv:(0)
3. [B=t]: 7 ==> [C=t]: 7 <conf:(1)> lift:(1) lev:(0) conv:(0)
4. [B=t]: 7 ==> [G=t]: 7 <conf:(1)> lift:(1.25) lev:(0.14) conv:(1.4)
5. [B=t]: 7 ==> [C=t, G=t]: 7 <conf:(1)> lift:(1.25) lev:(0.14) conv:(1.4)
6. [C=t, B=t]: 7 ==> [G=t]: 7 <conf:(1)> lift:(1.25) lev:(0.14) conv:(1.4)
7. [G=t, B=t]: 7 ==> [C=t]: 7 <conf:(1)> lift:(1) lev:(0) conv:(0)
8. [G=t]: 8 ==> [B=t]: 7 <conf:(0.88)> lift:(1.25) lev:(0.14) conv:(1.2)
9. [G=t]: 8 ==> [C=t, B=t]: 7 <conf:(0.88)> lift:(1.25) lev:(0.14) conv:(1.2)
10. [C=t, G=t]: 8 ==> [B=t]: 7 <conf:(0.88)> lift:(1.25) lev:(0.14) conv:(1.2)
```

Output from the Apriori and FPGrowth nodes

4. Compact Representation of Frequent Itemsets

| Closed Frequent Itemsets | Support count |
|--------------------------|---------------|
| {b} | 10 |
| {d} | 13 |
| {a, d} | 11 |
| {b, d} | 7 |
| {b, e} | 8 |
| {d, e} | 6 |
| {a, b, e} | 7 |
| {a, c, d} | 6 |
| {b, d, e} | 4 |
| {a, c, d, e} | 5 |

Using Algorithm 6.4:

For K = 4:

{a, c, d, e}: support = 5

For K = 3

Find all subsets from {a, c, d, e} and add closed frequent itemsets of length 3:

| itemset | Max of itemset | Support count |
|---------|----------------|---------------|
| {a,b,e} | | 7 |
| {a,c,d} | | 6 |
| {a,c,e} | {a,c,d,e}:5 | 5 |
| {a,d,e} | {a,c,d,e}:5 | 5 |
| {b,d,e} | | 4 |
| {c,d,e} | {a,c,d,e}:5 | 5 |

For K = 2

| itemset | Max of itemsets | Support count |
|---------|---------------------------------|---------------|
| {a,c} | {a,c,e}:5,{a,c,d}:6 | 6 |
| {a,b} | {a,b,e}:7 | 7 |
| {a,d} | | 11 |
| {a,e} | {a,b,e}:7, {a,c,e}:5, {a,d,e}:5 | 7 |
| {b,d} | | 7 |
| {b,e} | | 8 |
| {c,d} | {a,c,d}:6, {c,d,e}:5 | 6 |
| {c,e} | {a,c,e}:5, {c,d,e}:5 | 5 |
| {d,e} | | 6 |

For K=1

| itemset | Max of itemsets | Support count |
|---------|------------------------------------|---------------|
| {a} | {a,c}:6, {a,b}:7, {a,d}:11, | 11 |
| {b} | | 10 |
| {c} | {a,c}:6, {c,d}:6, {c,e}:5 | 6 |
| {d} | | 13 |
| {e} | {c,e}:5, {a,e}:7, {b,e}:8, {d,e}:6 | 8 |

All the listed itemsets are frequent itemsets