1 Apriori

Level 1 Count frequencies of individual items:
1:2, 2:2, 3:6, 4:6, 5:4, 6:7, 7:3, 8:4, ~~9:1~~, ~~10:1~~

Level 2 Count frequencies of pairs of frequent items:
~~1 2:0~~, ~~1 3:0~~, 1 4:2, ~~1 5:0~~, ~~1 6:0~~, ~~1 7:0~~, ~~1 8:0~~,
~~2 3:0~~, ~~2 4:0~~, ~~2 5:0~~, 2 6:2, 2 7:2, 2 8:2,
3 4:2, 3 5:4, 3 6:5, ~~3 7:1~~, 3 8:2,
~~4 5:0~~, ~~4 6:1~~, ~~4 7:0~~, ~~4 8:0~~,
5 6:4, ~~5 7:1~~, 5 8:2,
6 7:3, 6 8:4
7 8:2

Level 3 *Join* (candidate 3-itemsets):
2 6 7, 2 6 8, 2 7 8,
3 4 5, 3 4 6, 3 4 8, 3 5 6, 3 5 8, 3 6 8,
5 6 8,
6 7 8
Prune (subset pruning): 3 4 5 ('4 5' is infrequent), 3 4 6, 3 4 8
Count (count frequencies of the remaining itemsets):
2 6 7:2, 2 6 8:2, 2 7 8:2,
3 5 6:4, 3 5 8:2, 3 6 8:2,
5 6 8:2,
6 7 8:2 (all are frequent)

Level 4 *Join*: 2 6 7 8, 3 5 6 8
Prune: no pruning, all 3-subsets are frequent, e.g. one should check '2 6 7', '2 6 8', '2 7 8' and '6 7 8' for '2 6 7 8'
Count: 2 6 7 8:2, 3 5 6 8:2

Level 5 *Join*: Cannot generate any 5-itemsets $\Rightarrow$ the algorithm terminates.

All frequent itemsets, ordered by frequency descending:
$Freq = 7$: 6
$Freq = 6$: 3
$Freq = 5$: 3 6
$Freq = 4$: 3 5 6, 3 5, 5 6, 6 8, 4, 5, 8
$Freq = 3$: 6 7, 7
$Freq = 2$: 2 6 7 8, 3 5 6 8, 2 6 7, 2 6 8, 2 7 8, 3 5 8, 3 6 8, 5 6 8, 6 7 8, 1 4, 2 6, 2 7, 2 8, 3 4, 3 8, 5 8, 7 8, 1, 2

The dataset corresponds to the first 10 rows of `eda.csv`, where items are as follows:
1 - $hospital = sports$, 2 - $hospital = general$, 3 - $hospital = prenatal$, 4 - $gender = f$, 5 - $gender = m$, 6 - $blood\_test = t$, 7 - $ecg = t$, 8 -$ultrasound = t$, 9 - $mri = t$, 10 - $xray = t$

2 PCY

1. Perform the first pass of PCY.

   (a) Compute the support for each individual item.
   $sup(\{1\}) = 4$, $sup(\{2\}) = 6$, $sup(\{3\}) = 8$, $sup(\{4\}) = 8$, $sup(\{5\}) = 6$, $sup(\{6\}) = 4$.

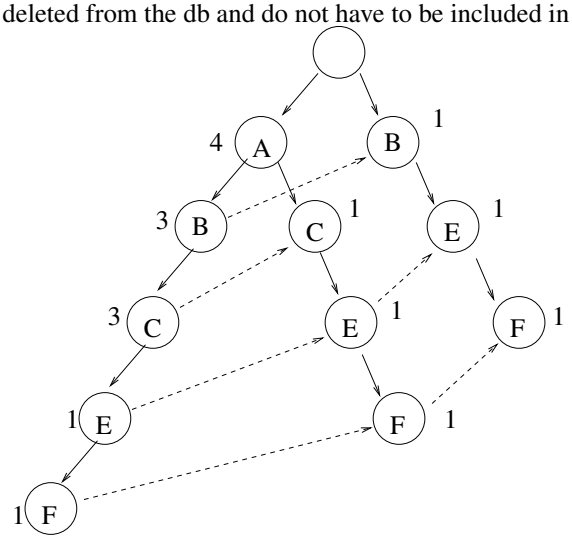
   (b) Which pairs hash to which buckets? Compute the frequencies of the buckets.
   $\{2,6\}$, $\{3,4\} \mapsto 1$ (bucket frequency $= 5$).
   $\{1,2\}$, $\{4,6\} \mapsto 2$ (5).
   $\{1,3\} \mapsto 3$ (3).
   $\{1,4\}$, $\{3,5\} \mapsto 4$ (6).
   $\{1,5\} \mapsto 5$ (1).
   $\{1,6\}$, $\{2,3\} \mapsto 6$ (3).
   $\{3,6\} \mapsto 7$ (2).
   $\{2,4\}$, $\{5,6\} \mapsto 8$ (6).
   $\{4,5\} \mapsto 9$ (3).
   $\{2,5\} \mapsto 10$ (2).
   Note that only the information on the right-hand side of arrows needs to be stored.

   (c) Which buckets are frequent?
   1, 2, 4, 8.

2. Generate all candidate pairs (2-itemsets): which ones are counted on the second pass of PCY?
   All individual items are frequent, hence all item pairs are valid candidates.
   It is only necessary to count frequencies for the ones that hash into frequent buckets:
   $\{2,6\}$, $\{3,4\}$, $\{1,2\}$, $\{4,6\}$, $\{1,4\}$, $\{3,5\}$, $\{2,4\}$, $\{5,6\}$.
   Note that neither $\{1,2\}$ nor $\{4,6\}$ is frequent, even though they hash into the frequent bucket 2.

3 FP Growth

Frequent **items**: A(4), B(4), C(4), ~~D(1)~~, E(3), F(3), ~~G(2)~~, ~~H(1)~~, ~~I(2)~~, ~~J(1)~~ (already ordered :)) – infrequent ones can be deleted from the db and do not have to be included in the FP-tree



4 Thought Question

Primary causes of pattern explosions include:

- *The nature of the problem*.
  If $\{A, B, C\}$ is frequent, then all its 7 supersets are necessarily returned.
- *Functional or statistical relations between attributes*.
  Dependencies between multiple variables might result in a large number of itemsets.
- *Locality of the support constraint*.
  A frequent itemset is always returned, independent of already returned itemsets.

Possible solutions are:

- *Top-k mining*, i.e. only returning k most frequent itemsets.
  Furthermore, frequency can be replaced with another interestingness measure.
- *Condensed representations*, lossless (closed itemsets) or lossy (maximal itemsets).
- *Pattern set mining*, i.e. introducing global constraints on the result sets to eliminate redundancy.
  For example, ensuring that covers of returned itemsets do not overlap. More advanced methods rely on heuristics rooted in probability theory, information theory, compression, etc.