

**Algorithm: Apriori.** Find frequent itemsets using an iterative level-wise approach based on candidate generation.

**Input:**

- $D$ , a database of transactions;
- $min\_sup$ , the minimum support count threshold.

**Output:**  $L$ , frequent itemsets in  $D$ .

**Method:**

```

(1)  $L_1 = \text{find\_frequent\_1-itemsets}(D)$ ;
(2) for  $(k = 2; L_{k-1} \neq \emptyset; k++)$  {
(3)    $C_k = \text{apriori\_gen}(L_{k-1})$ ;
(4)   for each transaction  $t \in D$  { // scan  $D$  for counts
(5)      $C_t = \text{subset}(C_k, t)$ ; // get the subsets of  $t$  that are candidates
(6)     for each candidate  $c \in C_t$ 
(7)        $c.\text{count}++$ ;
(8)   }
(9)    $L_k = \{c \in C_k \mid c.\text{count} \geq min\_sup\}$ 
(10) }
(11) return  $L = \cup_k L_k$ ;

procedure  $\text{apriori\_gen}(L_{k-1} \text{ : frequent } (k-1)\text{-itemsets})$ 
(1) for each itemset  $l_1 \in L_{k-1}$ 
(2)   for each itemset  $l_2 \in L_{k-1}$ 
(3)     if  $(l_1[1] = l_2[1]) \wedge (l_1[2] = l_2[2]) \wedge \dots \wedge (l_1[k-2] = l_2[k-2]) \wedge (l_1[k-1] < l_2[k-1])$  then {
(4)        $c = l_1 \bowtie l_2$ ; // join step: generate candidates
(5)       if  $\text{has\_infrequent\_subset}(c, L_{k-1})$  then
(6)         delete  $c$ ; // prune step: remove unfruitful candidate
(7)       else add  $c$  to  $C_k$ ;
(8)     }
(9) return  $C_k$ ;

procedure  $\text{has\_infrequent\_subset}(c \text{ : candidate } k\text{-itemset};$ 
    $L_{k-1} \text{ : frequent } (k-1)\text{-itemsets})$  // use prior knowledge
(1) for each  $(k-1)$ -subset  $s$  of  $c$ 
(2)   if  $s \notin L_{k-1}$  then
(3)     return TRUE;
(4) return FALSE;

```

**Figure 5.4** The Apriori algorithm for discovering frequent itemsets for mining Boolean association rules.

## 5.2.2 Generating Association Rules from Frequent Itemsets

Given a frequent itemset  $X$ , we can generate association rules by dividing  $X$  into two nonempty subsets  $A$  and  $B$  and then forming the rule  $A \Rightarrow B$ . For example, if  $X = \{a, b, c, d\}$  is a frequent itemset, we can generate the rule  $\{a, b\} \Rightarrow \{c, d\}$  by letting  $A = \{a, b\}$  and  $B = \{c, d\}$ . The support of the rule  $A \Rightarrow B$  is the support of  $X$ . The confidence of the rule  $A \Rightarrow B$  is the ratio of the support of  $X$  to the support of  $A$ . For example, if the support of  $X$  is 10 and the support of  $A$  is 5, then the confidence of the rule  $A \Rightarrow B$  is  $10/5 = 2$ .