The Apriori Algorithm: Basics

The Apriori Algorithm
It is an influential algorithm for mining frequent itemsets and using them for creating association rules

Key Concepts:
• Frequent Itemsets
• Apriori Property
The Apriori Algorithm: Basics

Key Concepts:
Frequent Itemsets

• The sets of item which has minimum support (denoted by $L_i$ for $i^{th}$-Itemset)

• Apriori Property
  • Any subset of frequent itemset must be frequent

• Join Operation
  • To find $C_k$, a set of candidate $k$-itemsets is generated by joining $L_{k-1}$ with itself.
The Apriori Algorithm in a Nutshell

- Apriori Algorithm finds the frequent itemsets
  i.e. the sets of items that have minimum support

It follows the Apriori Principle:

a subset of a frequent itemset must also be a frequent itemset

i.e., if \{A, B\} is a frequent itemset, both \{A\} and \{B\} should be a frequent itemset
The Apriori Algorithm in a Nutshell

- Apriori Algorithm

The algorithm Iteratively finds frequent itemsets with cardinality from 1 to $k$ (k-itemset)

- As the next step in the Apriori Process we use the frequent itemsets to generate association rules
The Apriori Algorithm: Pseudo code

- **Join Step:** $C_k$ is generated by **joining** $L_{k-1}$ with itself
- **Prune Step:** Any (k-1)-itemset that is not frequent cannot be a subset of a frequent k-itemset

**Pseudo-code:**

$C_k$: Candidate itemset of size k  
$L_k$: frequent itemset of size k

$L_1 = \{\text{frequent items}\}$;  
for $(k = 1; L_k \neq \emptyset; k++)$ do begin  
    $C_{k+1} = \text{candidates generated from } L_k$;  
    for each transaction $t$ in database do  
        increment the count of all candidates in $C_{k+1}$ that are contained in $t$  
    end  
    $L_{k+1} = \text{candidates in } C_{k+1} \text{ with min\_support}$  
end  
return $\bigcup_k L_k$;
The Apriori Algorithm: Example

Consider a database, $D$, consisting of 9 transactions.

Suppose min. support count required is 2 (i.e. $\text{min}_\text{sup} = \frac{2}{9} = 22\%$).

Let minimum confidence required is 70%.

We have to first find out the frequent itemset using Apriori algorithm.

Then, Association rules will be generated using min. support & min. confidence.

<table>
<thead>
<tr>
<th>TID</th>
<th>List of Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>T100</td>
<td>I1, I2, I5</td>
</tr>
<tr>
<td>T100</td>
<td>I2, I4</td>
</tr>
<tr>
<td>T100</td>
<td>I2, I3</td>
</tr>
<tr>
<td>T100</td>
<td>I1, I2, I4</td>
</tr>
<tr>
<td>T100</td>
<td>I1, I3</td>
</tr>
<tr>
<td>T100</td>
<td>I2, I3</td>
</tr>
<tr>
<td>T100</td>
<td>I1, I3</td>
</tr>
<tr>
<td>T100</td>
<td>I1, I2, I3, I5</td>
</tr>
<tr>
<td>T100</td>
<td>I1, I2, I3</td>
</tr>
</tbody>
</table>
Step 1: Generating 1-itemset Frequent Pattern

<table>
<thead>
<tr>
<th>Itemset</th>
<th>Sup.Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>{I1}</td>
<td>6</td>
</tr>
<tr>
<td>{I2}</td>
<td>7</td>
</tr>
<tr>
<td>{I3}</td>
<td>6</td>
</tr>
<tr>
<td>{I4}</td>
<td>2</td>
</tr>
<tr>
<td>{I5}</td>
<td>2</td>
</tr>
</tbody>
</table>

The set of frequent 1-itemsets, \(L_1\), consists of the candidate 1-itemsets satisfying minimum support.

In the first iteration of the algorithm, each item is a member of the set of candidates.
Step 2: Generating 2-itemset Frequent Pattern

- To **discover** the set of frequent 2-itemsets, $L_2$, the algorithm uses $L_1 \text{Join } L_1$ to generate a candidate set of 2-itemsets, $C_2$

- **Next**, the transactions in $D$ are scanned and the **support count** for each candidate itemset in $C_2$ is accumulated (as shown in the middle table)
Step 2: Generating 2-itemset Frequent Pattern

- 2-itemsets, $L_2$, is then determined, consisting of those candidate 2-itemsets in $C_2$ having minimum support

- Note: We haven’t used Apriori Property yet
**Step 2: Generating 2-itemset Frequent Pattern**

<table>
<thead>
<tr>
<th>Itemset</th>
<th>Sup. Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>{I1, I2}</td>
<td>4</td>
</tr>
<tr>
<td>{I1, I3}</td>
<td>4</td>
</tr>
<tr>
<td>{I1, I4}</td>
<td>1</td>
</tr>
<tr>
<td>{I1, I5}</td>
<td>2</td>
</tr>
<tr>
<td>{I2, I3}</td>
<td>4</td>
</tr>
<tr>
<td>{I2, I4}</td>
<td>2</td>
</tr>
<tr>
<td>{I2, I5}</td>
<td>2</td>
</tr>
<tr>
<td>{I3, I4}</td>
<td>0</td>
</tr>
<tr>
<td>{I3, I5}</td>
<td>1</td>
</tr>
<tr>
<td>{I4, I5}</td>
<td>0</td>
</tr>
</tbody>
</table>

- **Generate** $C_2$ candidates from $L_1$
- **Scan** $D$ for count of each candidate
- **Compare** candidate support count with minimum support count

$C_2$ candidates from $L_1$

<table>
<thead>
<tr>
<th>Itemset</th>
<th>Sup. Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>{I1, I2}</td>
<td>4</td>
</tr>
<tr>
<td>{I1, I3}</td>
<td>4</td>
</tr>
<tr>
<td>{I1, I5}</td>
<td>2</td>
</tr>
<tr>
<td>{I2, I3}</td>
<td>4</td>
</tr>
<tr>
<td>{I2, I4}</td>
<td>2</td>
</tr>
<tr>
<td>{I2, I5}</td>
<td>2</td>
</tr>
</tbody>
</table>

$L_2$
Step 3: Generating 3-itemset Frequent Pattern

In order to find $C_3$, we first compute $L_2 \ Join \ L_2$

$C_3 = L_2 \ Join \ L_2 = \{\{I_1, I_2, I_3\}, \{I_1, I_2, I_5\}, \{I_1, I_3, I_5\}, \{I_2, I_3, I_4\}, \{I_2, I_3, I_5\}, \{I_2, I_4, I_5\}\}.$

Now, Join step is complete and Prune step will be used to reduce the size of $C_3$.

Prune step helps to avoid heavy computation due to large $C_k$. 
Step 3: Generating 3-itemset Frequent Pattern

- **Apriori property** says that all subsets of a frequent itemset must also be frequent.

- \[ C_3 = L_2 \text{ Join } L_2 = \{\{I_1, I_2, I_3\}, \{I_1, I_2, I_5\}, \{I_1, I_3, I_5\}, \{I_2, I_3, I_4\}, \{I_2, I_3, I_5\}, \{I_2, I_4, I_5\}\} \]

- We determine now which of candidates in \( C_3 \) can and which can not possibly be frequent.
- Take \{I_1, I_2, I_3\}
- The 2-item subsets of it are \{I_1, I_2\}, \{I_1, I_3\}, \{I_2, I_3\}.
  - All of them are members of \( L_2 \).
  - We keep \{I_1, I_2, I_3\} in \( C_3 \).
Step 3: Generating 3-itemset Frequent Pattern

- Lets take \( \{I_2, I_3, I_5\} \)
- The 2-item subsets are \( \{I_2, I_3\}, \{I_2, I_5\}, \{I_3, I_5\} \)
- But \( \{I_3, I_5\} \) is not a member of \( L_2 \) and hence it is not frequent violating Apriori Property
- Thus we remove \( \{I_2, I_3, I_5\} \) from \( C_3 \)

All 2-item subsets of \( \{I_1, I_2, I_5\} \) members of \( L_2 \)
Therefore \( C_3 = \{\{I_1, I_2, I_3\}, \{I_1, I_2, I_5\}\} \)

- Now, the transactions in \( D \) are scanned in order to determine \( L_3 \), consisting of those candidates 3-itemsets in \( C_3 \) having minimum support and we get that

\[ L_3 = \{\{I_1, I_2, I_3\}, \{I_1, I_2, I_5\}\} \]
**Step 4: Generating 4-itemset Frequent Pattern**

- The algorithm uses $L_3 \text{Join} L_3$ to generate a candidate set of 4-itemsets, $C_4$
- $C_4 = L_3 \text{Join} L_3 = \{\{I1, I2, I3, I5\}\}$
- This itemset $\{\{I1, I2, I3, I5\}\}$ is pruned since its subset $\{\{I2, I3, I5\}\}$ is not frequent.
- Thus, $C_4 = \emptyset$ and algorithm terminates.

- **What’s Next?**
  Obtained frequent itemsets are to be used to generate strong association rules
  (where strong association rules are rules that satisfy both minimum support and minimum confidence)
Step 5: Generating Association Rules from Frequent Itemsets

Procedure:

• For each frequent itemset \( I \), generate the set of all nonempty subsets of \( I \)
  • For every nonempty subset \( S \) of \( I \),
  • output the rule \( S \rightarrow I - S \)
  • if support_count\( (I) \) / support_count\( (S) \) \( \geq \) min_conf
  • where min_conf is minimum confidence threshold.

Example

We obtained the set of all frequent itemsets
\[ L = \{\{I1\}, \{I2\}, \{I3\}, \{I4\}, \{I5\}, \{I1,I2\}, \{I1,I3\}, \{I1,I5\}, \{I2,I3\}, \{I2,I4\}, \{I2,I5\}, \{I1,I2,I3\}, \{I1,I2,I5\}\} \]

• Lets take for example \( I = \{I1,I2,I5\}\)
Step 5: Generating Association Rules from Frequent Itemsets

- Lets take $I = \{I1, I2, I5\}$
  - Its all nonempty subsets are $\{I1, I2\}, \{I1, I5\}, \{I2, I5\}, \{I1\}, \{I2\}, \{I5\}$
  - Let minimum confidence threshold be, say 70%
- The resulting association rules are shown below, each listed with its confidence.
  - $R1: I1 \land I2 \rightarrow I5$
    - Confidence = $\frac{sc\{I1, I2, I5\}}{sc\{I1, I2\}} = \frac{2}{4} = 50\%$
    - $R1$ is Rejected.
  - $R2: I1 \land I5 \rightarrow I2$
    - Confidence = $\frac{sc\{I1, I2, I5\}}{sc\{I1, I5\}} = \frac{2}{2} = 100\%$
    - $R2$ is Selected.
  - $R3: I2 \land I5 \rightarrow I1$
    - Confidence = $\frac{sc\{I1, I2, I5\}}{sc\{I2, I5\}} = \frac{2}{2} = 100\%$
    - $R3$ is Selected.
Step 5: Generating Association Rules from Frequent Itemsets

- **R4: I1 \(\rightarrow\) I2 ^ I5**
  - Confidence = \(\frac{\text{sc}\{I1,I2,I5\}}{\text{sc}\{I1\}} = \frac{2}{6} = 33\%\)
  - R4 is rejected.

- **R5: I2 \(\rightarrow\) I1 ^ I5**
  - Confidence = \(\frac{\text{sc}\{I1,I2,I5\}}{|I2|} = \frac{2}{7} = 29\%\)
  - R5 is rejected.

- **R6: I5 \(\rightarrow\) I1 ^ I2**
  - Confidence = \(\frac{\text{sc}\{I1,I2,I5\}}{|I5|} = \frac{2}{2} = 100\%\)
  - R6 is Selected

- We have found three strong association rules