Apriori Algorithm

Key Points

- **Definition**: Apriori is an algorithm used for mining frequent itemsets and learning association rules in transactional databases. It is one of the earliest and most widely used algorithms for this purpose.
- **Purpose**: To identify frequently occurring itemsets in large datasets and use these itemsets to generate association rules.
- **Advantages**:
  - **Simplicity**: Conceptually simple and easy to understand.
  - **Rule Generation**: Directly generates association rules from frequent itemsets.
- **Drawbacks**:
  - **Candidate Generation**: Generates a large number of candidate itemsets, which can be computationally expensive.
  - **Multiple Passes**: Requires multiple passes over the dataset to count item frequencies, which can be inefficient for very large datasets.
- **Key Concepts**:
  - **Support**: The frequency of an itemset appearing in the database.
  - **Confidence**: The likelihood that an item B is bought when item A is bought (i.e., P(B|A)).
  - **Lift**: The ratio of the observed support to the expected support if A and B were independent.

Algorithm

1. **Generate Frequent Itemsets**:
   - **Scan Dataset**:
     - Scan the dataset to count the support for each item and identify frequent items based on a minimum support threshold.
   - **Generate Candidates**:
     - Generate candidate itemsets of size k by combining frequent itemsets of size k-1. Ensure that all subsets of these candidate itemsets are frequent.
   - **Prune Non-Frequent Candidates**:
     - Count the support for each candidate itemset and prune those that do not meet the minimum support threshold.
2. **Repeat**:
   - **Increment k**:
     - Increase the size of itemsets (k) and repeat the candidate generation and pruning process until no more frequent itemsets can be found.
3. **Generate Association Rules** (if needed):
   - **Rule Generation**:
     - For each frequent itemset, generate association rules that satisfy a minimum confidence threshold. This involves splitting the itemset into antecedent and consequent parts and calculating their confidence.
4. **Evaluate Rules**:

- **Measure Quality**:
  - Evaluate the generated rules using metrics like support, confidence, and lift to ensure they are interesting and actionable.