UNIT – V

DATA GENERALIZATION AND SUMMARIZATION

What is concept description? - Data Generalization and summarization-based characterization - Attribute relevance - class comparisons Association Rule Mining: Market basket analysis - basic concepts - Finding frequent item sets: Apriori algorithm - generating rules – Improved Apriori algorithm – IncrementalARM – Associative Classification – Rule Mining

WHAT IS CONCEPT DESCRIPTION?

Descriptive vs. predictive data mining

❖ **Descriptive mining:**

describes concepts or task-relevant data sets in concise, summarative, informative, discriminative forms

❖ **Predictive mining:** Based on data and analysis, constructs models for the database, and predicts the trend and properties of unknown data

❖ Concept description:

- Characterization: provides a concise and succinct **summarization** of the given collection of data
- Comparison: provides descriptions **comparing** two or more collections of data

WHAT IS CONCEPT DESCRIPTION?

❖A concept usually refers to a collection of data such as frequent_buyers, graduate_students etc.

❖As a data mining task, concept description is not a simple enumeration (number of things done one by one) of the data.

❖Concept description generates descriptions for characterization and Comparison of the data it is also called class description.

❖ Characterization provides a concise and brief summarization

While concept or class comparison (also known as discrimination) provides (inequity) comparing two or more collections of data.

Example

- Discriminations Given the ABC Company database, for example, examining individual customer transactions.
- Sales managers may prefer to view the data generalized to higher levels, such as summarized by customer groups according to geographic regions, frequency of purchases per group and customer income.

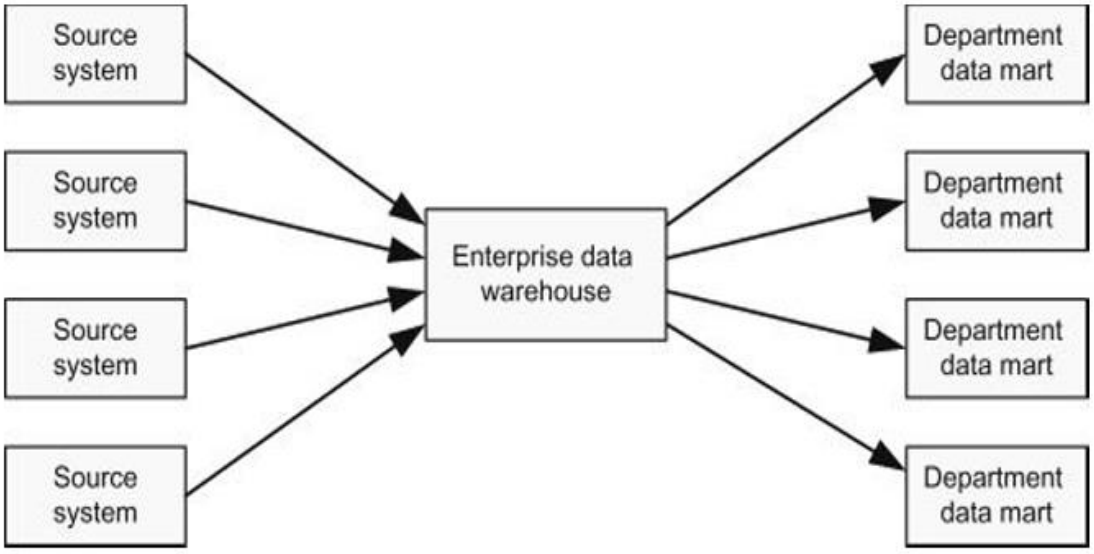
DATA GENERALIZATION AND SUMMARIZATION

❖ Data generalization

- A process which abstracts a large set of task-relevant data in a database from a low conceptual levels to higher ones.

❖ Approaches:

- Data cube approach(OLAP approach)
- Attribute-oriented induction approach



DATA GENERALIZATION

PRESENTATION OF GENERALIZED RESULTS

❖ **Generalized relation:**

- Relations where some or all attributes are generalized, with counts or other aggregation values accumulated.

❖ **Cross *tabulation:***

- *Mapping* results into cross tabulation form (similar to contingency tables).

❖ **Visualization techniques:**

Pie charts, bar charts, curves, cubes, and other visual forms.

❖ **Quantitative characteristic rules:**

- Mapping generalized result into characteristic rules with quantitative information associated with it

Presentation—Generalized Relation

location	item	sales (in million dollars)	count (in thousands)
Asia	TV	15	300
Europe	TV	12	250
North_America	TV	28	450
Asia	computer	120	1000
Europe	computer	150	1200
North_America	computer	200	1800

A generalized relation for the sales in 1997.

Class Characterization: An Example

Initial Relation

Name	Gender	Major	Birth-Place	Birth_date	Residence	Phone #	GPA
Jim Woodman	M	CS	Vancouver,BC, Canada	8-12-76	3511 Main St., Richmond	687-4598	3.67
Scott Lachance	M	CS	Montreal, Que, Canada	28-7-75	345 1st Ave., Richmond	253-9106	3.70
Laura Lee	F	Physics	Seattle, WA, USA	25-8-70	125 Austin Ave., Burnaby	420-5232	3.83
...	...	...	...	...	...	...	...
Removed	Retained	Sci,Eng, Bus	Country	Age range	City	Removed	Excl, VG,..

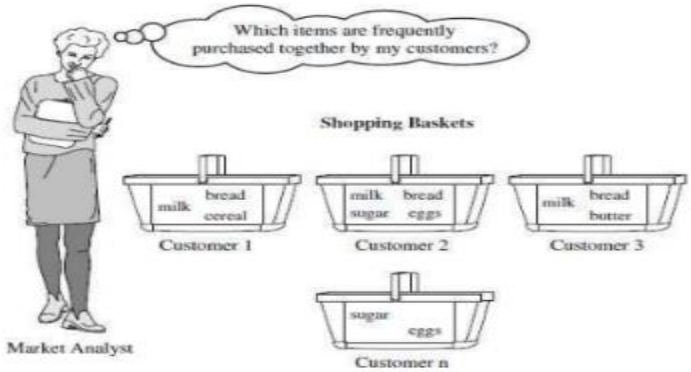
Prime Generalized Relation

Gender	Major	Birth_region	Age_range	Residence	GPA	Count
M	Science	Canada	20-25	Richmond	Very-good	16
F	Science	Foreign	25-30	Burnaby	Excellent	22
...	...	...	...	...	...	...

Gender \ Birth_Region	Canada	Foreign	Total
M	16	14	30
F	10	22	32
Total	26	36	62

MARKET BASKET ANALYSIS

Market Basket Analysis is a modeling technique based upon the theory that if you **buy a certain group of items**, you are more (or less) likely **to buy another group of items**. For example, if you are in an English pub and you buy a pint of beer and don't buy a bar meal, you are more likely to buy crisps (US. chips) at the same time than somebody who didn't buy beer.



❖ The set of items a customer buys is referred to as an itemset, and market basket analysis seeks to find relationships between purchases.

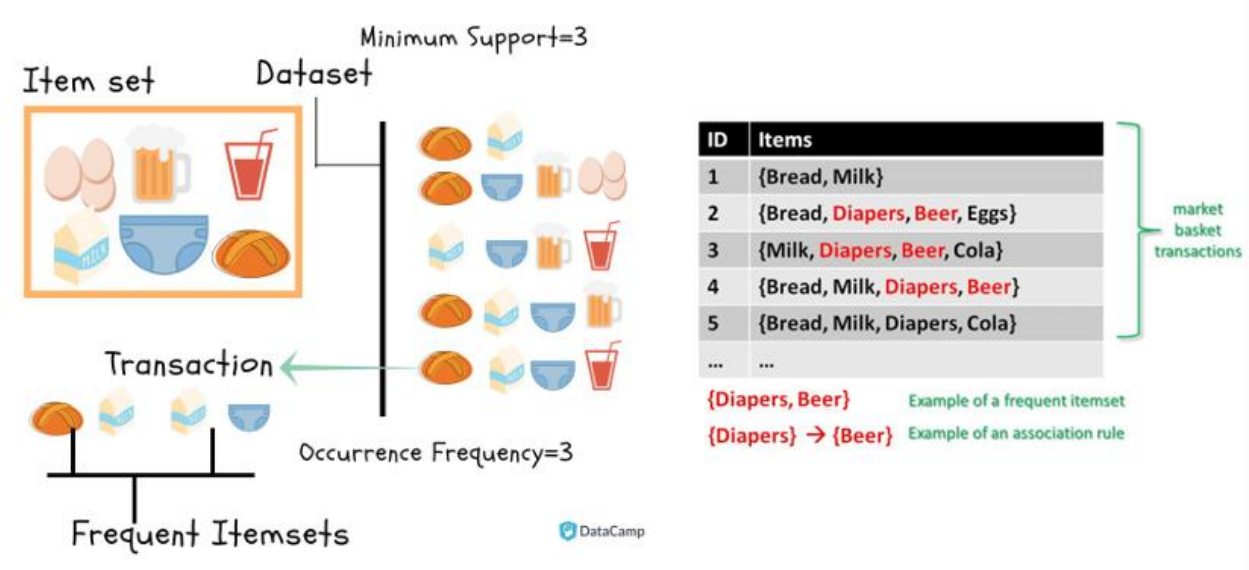
❖ Typically the relationship will be in the form of a rule:

- IF {beer, no bar meal} THEN {crisps}.

The probability that a customer will buy beer without a bar meal (i.e. that the antecedent is true) is referred to as the **support** for the rule. The conditional probability that a customer will purchase crisps is referred to as the **confidence**.

❖ The algorithms for performing market basket analysis are fairly straightforward (Berry and Linhoff is a reasonable introductory resource for this). The complexities mainly arise in exploiting taxonomies, avoiding combinatorial explosions (a supermarket may stock 10,000 or more line items), and dealing with the large amounts of transaction data that may be available.

❖ A major difficulty is that a large number of the rules found may be trivial for anyone familiar with the business. Although the volume of data has been reduced, we are still asking the user to find a needle in a haystack. Requiring rules to have a high minimum support level and a high confidence level risks missing any exploitable result we might have found.



MARKET BASKET ANALYSIS

How Association Rules are Evaluated

The strength and reliability of an association rule are measured using three key metrics.

Support and Confidence for Itemset A and B are represented by formulas:

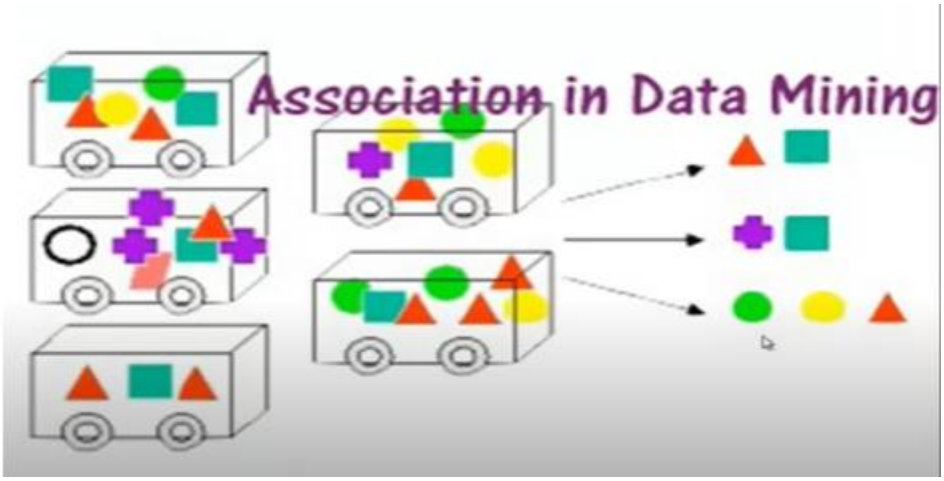
$$\text{Support (A)} = \frac{\text{Number of transaction in which A appears}}{\text{Total number of transactions}}$$

$$\text{Confidence (A}\rightarrow\text{B)} = \frac{\text{Support(AUB)}}{\text{Support(A)}}$$

Lift (A,B) = C(A→B)/S(B)

ASSOCIATION RULE MINING

Association rule learning is a rule-based machine learning method for discovering interesting relations between variables in large databases. It is intended to identify strong rules discovered in databases using some measures of interestingness



❖ Learning of Association rules is used to find relationships between attributes in large databases. An association rule, A=> B, will be of the form" for a set of transactions, some value of itemset A determines the values of itemset B under the condition in which minimum support and confidence are met".

❖ Support and Confidence can be represented by the following example:

```
Bread=> butter [support=2%, confidence-60%]
```

❖ The above statement is an example of an association rule. This means that there is a 2% transaction that bought bread and butter together and there are 60% of customers who bought bread as well as butter.
ASSOCIATION RULE MINING

Support and Confidence for Itemset A and B are represented by formulas:

$$\text{Support (A)} = \frac{\text{Number of transaction in which A appears}}{\text{Total number of transactions}}$$

$$\text{Confidence } (A \rightarrow B) = \frac{\text{Support(AUB)}}{\text{Support(A)}}$$

$$Lift(A \Rightarrow B) = \frac{Confidence\,(A \Rightarrow B)}{Expected\ Confidence\,(A \Rightarrow B)} = \frac{Confidence\,(A \Rightarrow B)}{P(B)} = \frac{P(A \cap B)}{P(A) \cdot P(B)} = \frac{Confidence\,(A \Rightarrow B)}{Support\,(B)}$$

Association rule mining consists of 2 steps:
1. Find all the frequent itemsets.
2. Generate association rules from the above frequent itemsets.

TID	Items
1	Bread, Peanuts, Milk, Fruit, Jam
2	Bread, Jam, Soda, Chips, Milk, Fruit
3	Steak, Jam, Soda, Chips, Bread
4	Jam, Soda, Peanuts, Milk, Fruit
5	Jam, Soda, Chips, Milk, Bread
6	Fruit, Soda, Chips, Milk
7	Fruit, Soda, Peanuts, Milk
8	Fruit, Peanuts, Cheese, Yogurt

Examples

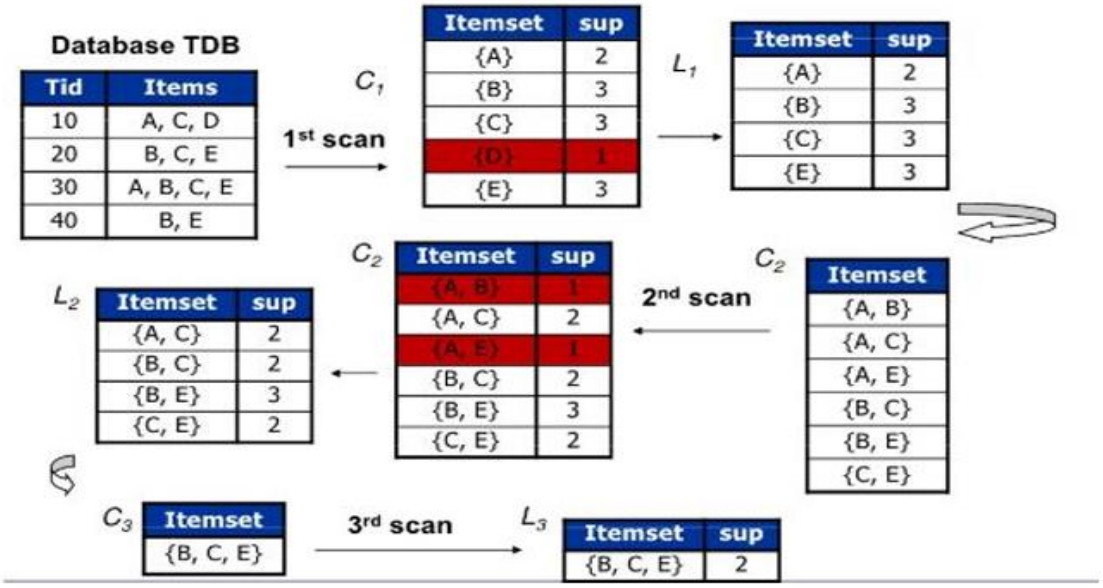
{bread} ⇒ {milk}

{soda} ⇒ {chips}

{bread} ⇒ {jam}

- Given a set of transactions, find rules that will predict the occurrence of an item based on the occurrences of other items in the transaction

EXAMPLE OF ASSOCIATION RULE



FINDING FREQUENT ITEM SETS:

APRIORI ALGORITHM:

Apriori Algorithm :-

Apriori Algorithm is an influential algorithm for mining frequent itemsets for Boolean Association rules. The name of the algorithm is based on the fact that the algorithm uses prior knowledge of frequent itemset properties.

It uses iterative approach known as level wise search where k-itemsets used to explore (k+1) itemsets.

The set of frequent 1-itemsets is found. This set is denoted by L_1.

L_1 is used to find L_2, the set of frequent 2-itemsets, which is used to find L_3 and so on and it continues until no frequent k-itemsets can be found. the finding of each L_k requires one full scan of database

In order to use Apriori property, all non-empty subsets of a frequent itemset must also be frequent. If an itemset I doesnot satisfy the minimum support threshold min-sup, then I is not frequent.

$P(I) < min\text{-}sup.$

If an itemset item A is added to the itemset I, then the resulting itemset is [P(I ∪ A) < min_sup]

Apriori property follows two steps:

① Join step
② Prune step

① Join step :-

To find L_k, a set of candidate k-itemsets is generated by joining L_{k-1} with itself i.e., $L_{k-1} \bowtie L_{k-1}$ and $l_1, l_2 \in L_{k-1}$

Notation :- $l_i[j]$ where j is j^{th} item in l_i

Formula :-

$$L_{k-1} \bowtie L_{k-1} = (l_1[1] = l_2[1]) \wedge (l_1[2] = l_2[2]) \wedge \ldots \wedge (l_1[k-2] = l_2[k-2]) \wedge (l_1[k-1] < l_2[k-1])$$

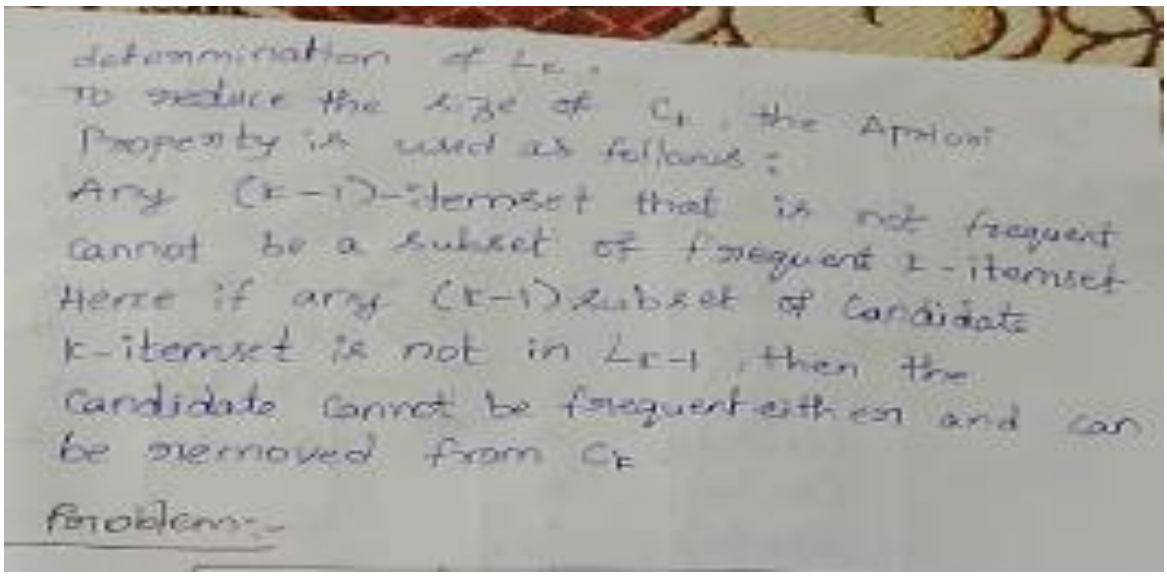
The condition $l_1[k-1] < l_2[k-1]$ ensures that no duplicates are generated

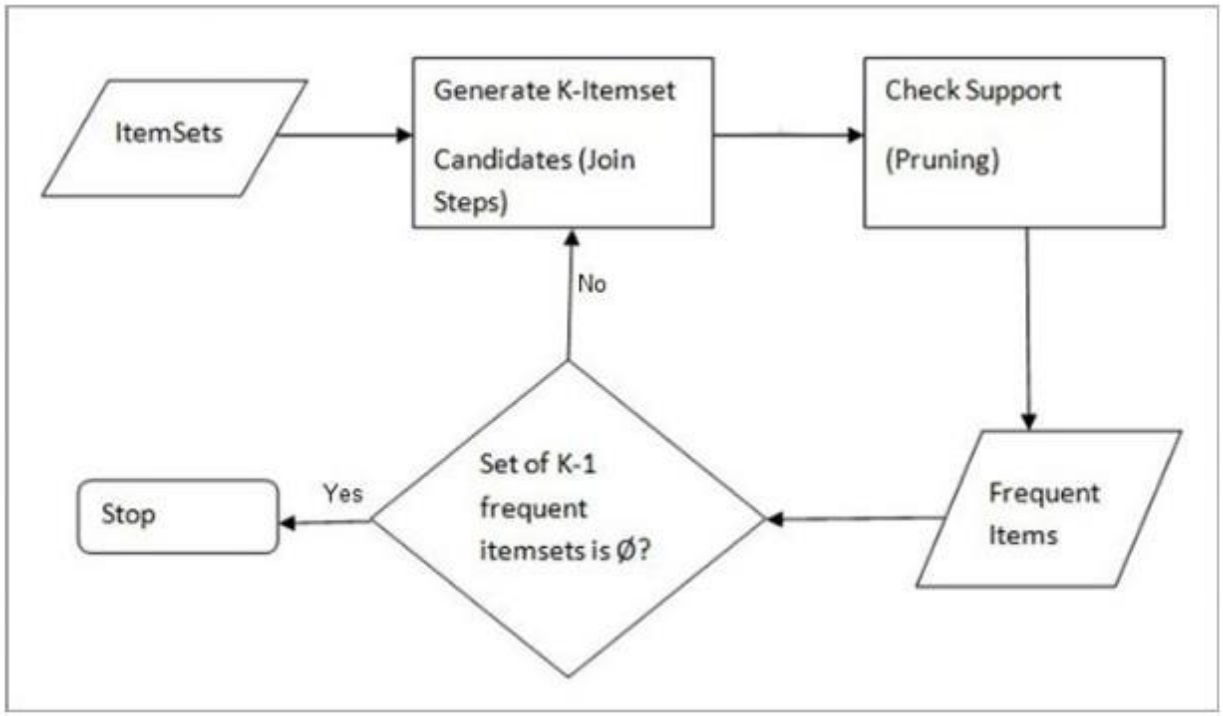
Resulting itemset :-

$l_1[1] l_1[2] \ldots l_1[k-1] l_2[k-1]$

② Prune step :-

C_k is superset of L_k.
That is members may or may not be frequent, but all frequent k-itemsets are included in C_k would result in



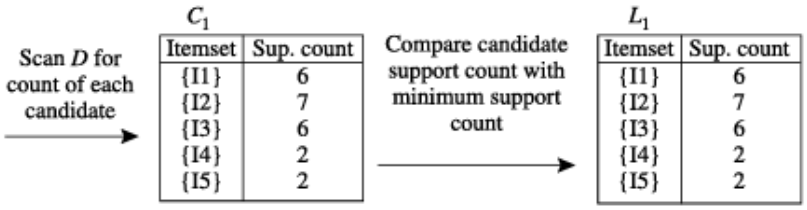


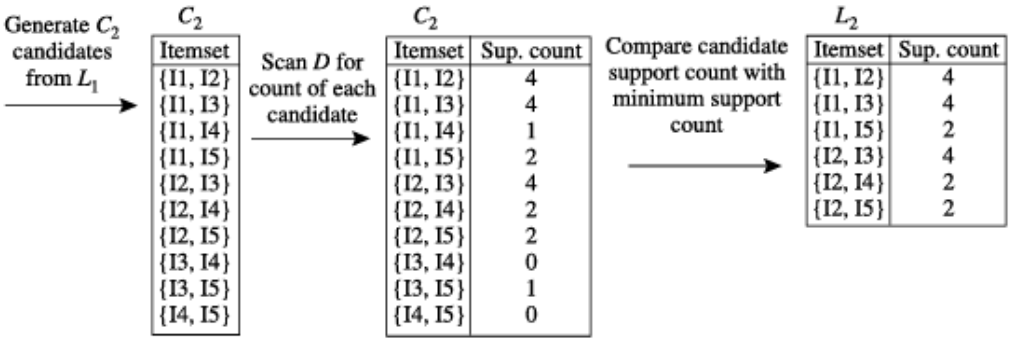
Example of Apriori: Support threshold=50%, Confidence= 60%

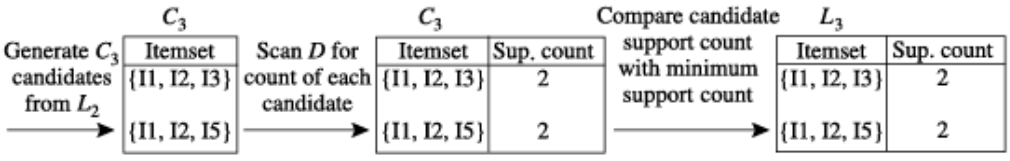
EXAMPLE:1

Transactional Data for an *AllElectronics* Branch

TID	*List of item_IDs*
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3







EXAMPLE:2

Transaction	List of items
T1	I1,I2,I3
T2	I2,I3,I4
T3	I4,I5
T4	I1,I2,I4
T5	I1,I2,I3,I5
T6	I1,I2,I3,I4

Solution:

Support threshold=50% => 0.5*6= 3 => min_sup=3

1. Count Of Each Item

TABLE-2

Item	Count
I1	4
I2	5
I3	4
I4	4
I5	2

2. Prune Step: TABLE -2 shows that I5 item does not meet min_sup=3, thus it is deleted, only I1, I2, I3, I4 meet min_sup count.

TABLE-3

Item	Count
I1	4
I2	5
I3	4
I4	4

3. Join Step: Form 2-itemset. From **TABLE-1** find out the occurrences of 2-itemset.

TABLE-4

Item	Count
I1,I2	4
I1,I3	3
I1,I4	2
I2,I3	4
I2,I4	3
I3,I4	2

4. Prune Step: TABLE -4 shows that item set {I1, I4} and {I3, I4} does not meet min_sup, thus it is deleted.

TABLE-5

Item	Count
I1,I2	4
I1,I3	3
I2,I3	4
I2,I4	3

5. Join and Prune Step: Form 3-itemset. From the **TABLE- 1** find out occurrences of 3-itemset. From **TABLE-5**, find out the 2-itemset subsets which support min_sup.

We can see for itemset {I1, I2, I3} subsets, {I1, I2}, {I1, I3}, {I2, I3} are occurring in **TABLE-5** thus {I1, I2, I3} is frequent.

We can see for itemset {I1, I2, I4} subsets, {I1, I2}, {I1, I4}, {I2, I4}, {I1, I4} is not frequent, as it is not occurring in **TABLE-5** thus {I1, I2, I4} is not frequent, hence it is deleted.

TABLE-6

Item
I1,I2,I3
I1,I2,I4
I1,I3,I4
I2,I3,I4

Only {I1, I2, I3} is frequent.

6. Generate Association Rules: From the frequent itemset discovered above the association could be:

{I1, I2} => {I3}

Confidence = support {I1, I2, I3} / support {I1, I2} = (3/ 4)* 100 = 75%

{I1, I3} => {I2}

Confidence = support {I1, I2, I3} / support {I1, I3} = (3/ 3)* 100 = 100%

{I2, I3} => {I1}

Confidence = support {I1, I2, I3} / support {I2, I3} = (3/ 4)* 100 = 75%

{I1} => {I2, I3}

Confidence = support {I1, I2, I3} / support {I1} = (3/ 4)* 100 = 75%

{I2} => {I1, I3}

Confidence = support {I1, I2, I3} / support {I2 = (3/ 5)* 100 = 60%

{I3} => {I1, I2}

Confidence = support {I1, I2, I3} / support {I3} = (3/ 4)* 100 = 75%

This shows that all the above association rules are strong if minimum confidence threshold is 60%.

- 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

```
L1 = {frequent items};
for (k = 1; Lk !=∅; k++) do begin
   Ck+1 = candidates generated from Lk;
   for each transaction t in database do
         increment the count of all candidates in Ck+1
         that are contained in t
   Lk+1 = candidates in Ck+1 with min_support
   end
return ∪k Lk;
```

Association Rule Mining (Refer class work notes)

IMPROVED APRIORI ALGORITHM

Many methods are available for improving the efficiency of the algorithm.

1. **Hash-Based Technique**: This method uses a hash-based structure called a hash table for generating the k-itemsets and its corresponding count. It uses a hash function for generating the table.
2. **Transaction Reduction**: This method reduces the number of transactions scanning in iterations. The transactions which do not contain frequent items are marked or removed.
3. **Partitioning**: This method requires only two database scans to mine the frequent itemsets. It says that for any itemset to be potentially frequent in the database, it should be frequent in at least one of the partitions of the database.

4. **Sampling:** This method picks a random sample S from Database D and then searches for frequent itemset in S. It may be possible to lose a global frequent itemset. This can be reduced by lowering the min_sup.

5. **Dynamic Itemset Counting:** This technique can add new candidate itemsets at any marked start point of the database during the scanning of the database.

APPLICATION OF ALGORITHM

Some fields where Apriori is used:

1. **In Education Field:** Extracting association rules in data mining of admitted students through characteristics and specialties.
2. **In the Medical field:** For example Analysis of the patient's database.
3. **In Forestry:** Analysis of probability and intensity of forest fire with the forest fire data.
4. Apriori is used by many companies like Amazon in the by Google for the auto-complete feature.

INCREMENTAL ARM

❖ It is noted that analysis of past transaction data can provide very valuable information on customer buying behavior, and thus **improve the quality of business decisions.**

❖ With the increasing use of the record-based databases whose data is being continuously added, updated, deleted etc.

❖ Examples of such applications include Web log records, stock market data, grocery sales data, transactions in e-commerce, and daily weather/traffic records etc.

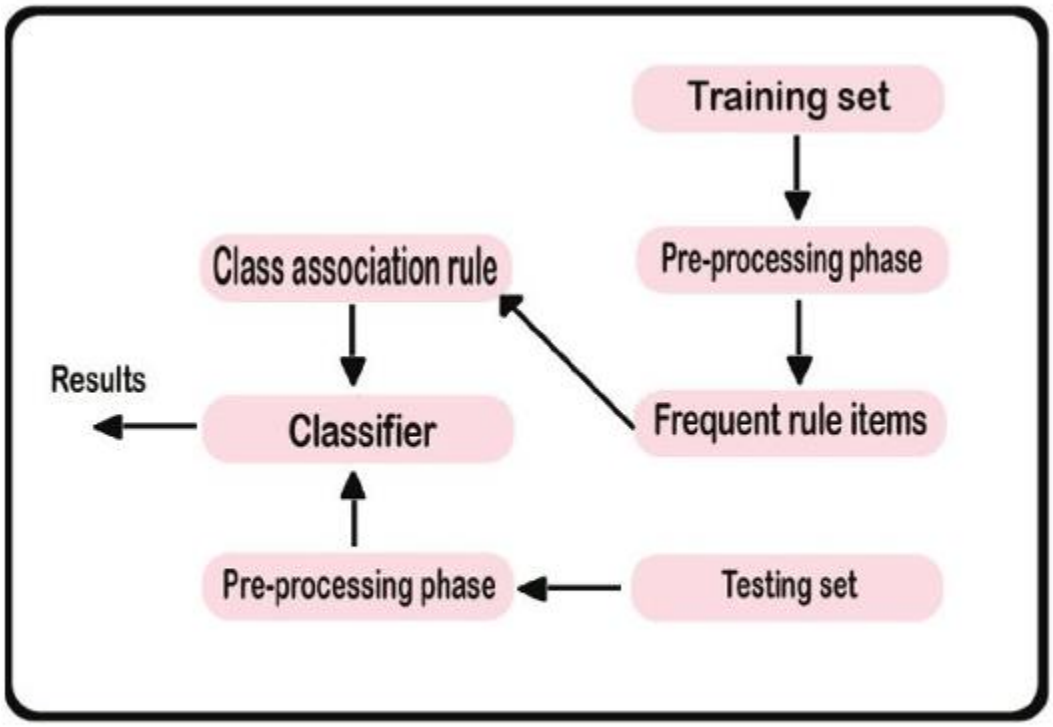
❖ In many applications, we would like to mine the transaction database for a fixed amount of most recent data (say, data in the last 12 months).

❖ Mining is not a one-time operation, a naive approach to solve the incremental mining problem is to re-run the mining algorithm on the updated database.

ASSOCIATIVE CLASSIFICATION

❖ Associative classification (AC) is a branch of a wide area of scientific study known as data mining. Associative classification makes use of association rule mining for extracting efficient rules, which can precisely generalize the training data set, in the rule discovery process.

❖ An associative classifier (AC) is a kind of supervised learning model that uses association rules to assign a target value. The term associative classification was coined by Bing Liu et al., in which the authors defined a model made of rules "whose right-hand side are restricted to the classification class attribute".



DATA FLOW DIAGRAM OF ASSOCIATIVE CLASSIFICATION

Association Mining

• Association rule mining: – Finding frequent patterns, associations, correlations, or causal structures among sets of items or objects in transaction databases, relational databases, and other information repositories.

• Applications: – Basket data analysis, cross-marketing, catalog design, loss-leader analysis, clustering, classification, etc.

• Examples.

- Rule form: "Body ® Head [support, confidence]".
- buys(x, "diapers") ® buys(x, "beers") [0.5%, 60%]
- major(x, "CS") ^ takes(x, "DB") ® grade(x, "A") [1%, 75%]

Association Rule: Basic Concepts

Given: (1) database of transactions, (2) each transaction is a list of items (purchased by a customer in a visit)

• Find: all rules that correlate the presence of one set of items with that of another set of items

– E.g., 98% of people who purchase tires and auto accessories also get automotive services done

• Applications
– *==> Maintenance Agreement (What the store should do to boost Maintenance Agreement sales)
– Home Electronics *==> (What other products should the store stocks up?)
– Attached mailing in direct marketing – Detecting “ping-pong”ing of patients, faulty “collisions”
Rule Measures: Support and Confidence
• Find all the rules X & Y => Z with minimum confidence and support
– support, s, probability that a transaction contains {X 4 Y 4 Z}
– confidence, c, conditional probability that a transaction having {X 4 Y} also contains Z
Let minimum support 50%, and minimum confidence 50%, we have
– A => C (50%, 66.6%)
– C=> A (50%, 100%)

Transaction ID	Items Bought
2000	A,B,C
1000	A,C
4000	A,D
5000	B,E,F

Association Rule Mining: A Road Map B,E,F
• Boolean vs. quantitative associations (Based on the types of values handled)
– buys(x, “SQLServer”) ^ buys(x, “DMBook”) ® buys(x, “DBMiner”) [0.2%, 60%]
– age(x, “30..39”) ^ income(x, “42..48K”) ® buys(x, “PC”) [1%, 75%]
• Single dimension vs. multiple dimensional associations (see ex. Above)
• Single level vs. multiple-level analysis
– What brands of beers are associated with what brands of diapers?
• Various extensions
– Correlation, causality analysis
• Association does not necessarily imply correlation or causality
– Maxpatterns and closed itemsets
– Constraints enforced
• E.g., small sales (sum < 100) trigger big buys (sum > 1,000)?

Apriori Algorithm and Its Role in Associative Classification

In associative classification, the Apriori algorithm is a key method that is essential for identifying popular item sets. The method finds itemsets that meet a minimal support criterion via an iterative technique, creating strong correlations between qualities. Its main function in associative categorization is to produce a set of frequent item sets from which association rules may be derived.

Utilizing the "apriori property," which stipulates that any non? frequent itemset must have non? frequent subsets, the method effectively prunes the search space.

Association Rule Mining is a powerful technique used to uncover meaningful relationships between variables within large datasets. They are designed to discover “if-then” patterns, providing insights into how data items are related and frequently occur together. These rules are particularly useful in identifying correlations and dependencies, enabling data-driven decision-making.

For instance, in a retail dataset, an association rule might identify that “if a customer buys bread, they are likely to buy butter”. Such insights help businesses improve cross-selling strategies, inventory management, and customer satisfaction.

Key Components of Association Rules

Antecedent: The “if” part of the rule, representing the condition.
Example: A customer buys bread.
Consequent: The “then” part of the rule, representing the outcome.
Example: The customer also buys butter.
Association rules are derived through algorithms that evaluate the frequency and strength of these relationships. They use metrics like support, confidence, and lift to measure the relevance and reliability of discovered patterns. These rules have applications in various fields, such as retail, healthcare, and marketing, where analyzing customer behavior or trends is critical for success.

Rule Evaluation Metrics

Association rules are evaluated using key metrics that determine their relevance, strength, and reliability. These metrics include support, confidence, and lift, which quantify the frequency and strength of relationships between data items.

1. Support

Support measures how frequently an itemset (both antecedent and consequent) appears in the dataset. It provides an indication of how common a particular association is.

$$\text{Support} = \frac{\text{Transactions containing both antecedent and consequent}}{\text{Total transactions}}$$

Formula:

Example: If bread and butter appear together in 100 out of 1,000 transactions, the support is:

$$\text{Support} = \frac{100}{1000} = 0.10\,(10\%)$$

A higher support value indicates a more frequently occurring pattern in the dataset.

2. Confidence

Confidence measures the likelihood of the consequent occurring given that the antecedent has already occurred. It evaluates the reliability of the rule.

Formula:

$$\text{Confidence} = \frac{\text{Support of antecedent and consequent}}{\text{Support of antecedent}}$$

Example: If 70% of customers who buy bread also buy butter, the confidence is:

$$\text{Confidence} = 70\% = 0.70$$

Higher confidence suggests a stronger relationship between the antecedent and consequent.

3. Lift

Lift measures the strength of an association compared to its random occurrence in the dataset. It identifies how much more likely the antecedent and consequent are to appear together than independently.

Formula:

$$\text{Lift} = \frac{\text{Confidence}}{\text{Support of consequent}}$$

Example: A lift value greater than 1 indicates a strong positive association, while a value equal to 1 suggests no association. For instance, if the lift is 1.5, it means the antecedent makes the consequent 1.5 times more likely.

How Does Association Rule Learning Work?

Association rule learning is a multi-step process designed to identify meaningful patterns and relationships in large datasets. It involves two main stages:

Identifying Frequent Itemsets: The process begins by identifying frequent itemsets—combinations of items that appear together in transactions with a frequency above a predefined threshold. Metrics like support are used to measure how often these itemsets occur in the dataset. For example, a frequent itemset might reveal that bread and butter are purchased together in 10% of transactions.

Generating Association Rules: Once frequent itemsets are identified, association rules are generated. These rules take the form of if-then statements that describe relationships between items (e.g., "If a customer buys bread, they are likely to buy butter"). Metrics such as confidence and lift are applied to evaluate the strength and reliability of these rules.

Iterative Refinement

The process is iterative, with thresholds for support and confidence adjusted to refine the rules. This ensures that only the most significant and actionable rules are selected. For instance, a rule with low confidence may be excluded from further analysis.

Through this systematic approach, association rule learning uncovers valuable insights from raw data, enabling organizations to make data-driven decisions.

Types of Association Rule Learning Algorithms

Several algorithms are used for association rule learning, each with unique strengths and applications. The three most commonly used algorithms are:

1. Apriori Algorithm

The Apriori algorithm employs a breadth-first search approach to identify frequent itemsets. It relies on the principle that all subsets of a frequent itemset must also be frequent, reducing the search space.

Advantage: Simple to implement and effective for small datasets with low dimensionality.

Limitation: Performance degrades significantly with large or dense datasets due to repeated scanning of the database.

2. Eclat Algorithm

The Eclat algorithm uses a depth-first search strategy to discover frequent itemsets. Instead of scanning the database multiple times, it represents transactions as vertical itemsets and directly computes intersections.

Advantage: Efficient for datasets with sparse data or where there are fewer frequent itemsets.

3. FP-Growth Algorithm

The FP-Growth (Frequent Pattern Growth) algorithm leverages a prefix-tree structure called the FP-tree to represent transactional data compactly. Unlike Apriori, it avoids generating candidate itemsets explicitly, making it faster and more efficient.

Advantage: Significantly faster and more memory-efficient than Apriori, especially for large datasets.

Applications of Association Rules

Association rules are widely applied across various industries to uncover patterns and relationships in data, enabling better decision-making and operational efficiency.

1.Retail and Market Basket Analysis: Retailers use association rules to identify frequently purchased product combinations, helping them optimize store layouts or create product bundles to increase sales.
Example: A supermarket discovers that customers who buy bread often purchase butter and jam, leading to strategic placement of these items together.

2.Healthcare: In healthcare, association rules help discover co-occurrence patterns in symptoms, aiding in diagnostic processes and treatment plans.
Example: Identifying that patients with high blood pressure often have a higher risk of developing diabetes can guide preventative care strategies.

3.E-Commerce and Recommendation Systems: E-commerce platforms leverage association rules to build recommendation systems that enhance user experiences and drive sales.
Example: Amazon's "Customers who bought this also bought" feature suggests complementary products, boosting cross-selling opportunities.

4.Fraud Detection: Association rules are used in financial services to identify unusual patterns in transaction data, which can help detect fraudulent activities.
Example: Flagging transactions that deviate significantly from established spending patterns for further investigation.

Example of Association Rules

Consider a small transaction dataset where customers purchase items like bread, butter, and milk.

Dataset Example:

Transaction ID	Items Purchased
1	Bread, Butter
2	Bread, Milk
3	Bread, Butter, Milk
4	Milk
5	Bread, Butter

Rule Discovery Process:

Rule Example: "If bread is purchased, then butter is likely to be purchased."

1. **Support Calculation:**
   Support = Transactions containing both bread and butter ÷ Total transactions

$$\text{Support} = \frac{3}{5} = 0.6\,(60\%)$$

2. **Confidence Calculatic**
   Confidence = Support of bread and butter ÷ Support of bread

$$\text{Confidence} = \frac{3}{4} = 0.75\,(75\%)$$

3. **Lift Calculation:**
   Lift = Confidence ÷ Support of butter

$$\text{Lift} = \frac{0.75}{0.6} = 1.25$$

A lift value greater than 1 indicates a positive association between bread and butter.

This example demonstrates how association rules are derived and evaluated, providing actionable insights from transactional data.