GUJARAT TECHNOLOGICAL UNIVERSITY

BE 6 SEMESTER – OLD PAPER – S22 TO S26 – ANSWER BANK

Subject Name & Code: Data Mining - 3160714

Chapter 1: Introduction to data mining (DM)

Repeated Questions:

1. Knowledge Discovery Process:

- **Q.** Describe the steps involved in data mining when viewed as a process of knowledge discovery / Explain KDD process. [Diagram required – refer to the original question paper] **(SUMMER 2023, 7 Marks; WINTER 2023, 7 Marks; SUMMER 2024, 4 Marks; WINTER 2025, 7 Marks; SUMMER 2026, 7 Marks)**

Answer:

KDD (Knowledge Discovery in Databases) Process:

- **Data Cleaning:** Remove noise and inconsistent data; handle missing values.
- **Data Integration:** Combine multiple heterogeneous data sources into a coherent data store.
- **Data Selection:** Retrieve only the data relevant to the analysis task from the database.
- **Data Transformation:** Transform data into appropriate forms for mining (e.g., normalization, aggregation, discretization).
- **Data Mining:** Apply intelligent methods to extract data patterns — this is the core step.
- **Pattern Evaluation:** Identify truly interesting patterns representing knowledge using interestingness measures.
- **Knowledge Presentation:** Visualize and present the mined knowledge to the user.

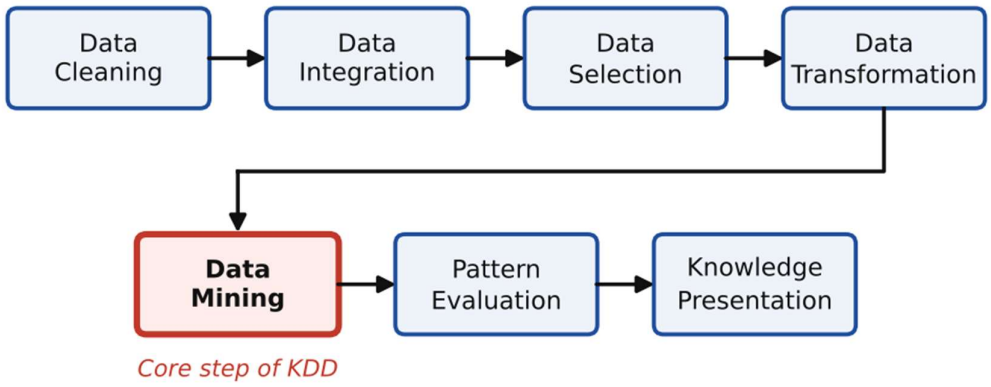


Fig. 1: KDD process (steps of knowledge discovery)

2. Data Mining Issues and Functionalities:

- **Q.** Explain the major issues in data mining / Explain major requirements and challenges in data mining. **(WINTER 2022, 7 Marks; WINTER 2023, 7 Marks; SUMMER 2024, 7 Marks; WINTER 2025, 4 Marks; SUMMER 2026, 7 Marks)**

Answer:

Major Issues in Data Mining:

- **Mining Different Kinds of Knowledge:** Supporting multiple functionalities (association, classification, clustering, etc.) within one unified framework.
- **Interactive Mining at Multiple Abstraction Levels:** Allowing users to interactively refine mining queries and view results at different granularities.
- **Incorporation of Background Knowledge:** Using domain knowledge and concept hierarchies to guide the mining process and improve result quality.
- **Efficiency and Scalability:** Algorithms must handle massive datasets efficiently within acceptable time limits.
- **Parallel, Distributed, and Incremental Mining:** Mining across distributed data sources and handling incremental database updates without full recomputation.
- **Diverse Data Types:** Handling relational, spatial, temporal, multimedia, text, and web data.
- **Noise, Uncertainty, and Incomplete Data:** Robustness against imperfect, missing, or noisy data.
- **Pattern Evaluation and Interestingness:** Filtering out uninteresting patterns and measuring pattern quality effectively.

- **Q.** Explain the data mining functionalities / List out the data mining functionality. **(WINTER 2022, 4 Marks; WINTER 2025, 3 Marks)**

Answer:

Data Mining Functionalities:

- **Concept/Class Description:** Characterization and discrimination of data classes.
- **Association Analysis:** Discovery of association rules showing attribute-value conditions that occur frequently together.
- **Classification and Prediction:** Finding models that describe and predict class labels or numerical values.
- **Cluster Analysis:** Grouping data into clusters based on similarity without predefined classes.
- **Outlier Analysis:** Identification of data objects that do not comply with the general behavior of the data.

- **Q.** Compare descriptive and predictive data mining. **(SUMMER 2022, 4 Marks; WINTER 2022, 3 Marks)**

Answer:

Aspect	Descriptive Data Mining	Predictive Data Mining
Purpose	Describes general properties of data	Predicts future trends/values
Approach	Characterizes data in a concise manner	Builds models to predict unknown values
Techniques	Clustering, association rules, summarization	Classification, regression, prediction
Output	Patterns, clusters, summaries	Predictive models, forecasts

---

3. Data Warehouse and OLAP:

- **Q.** Differentiate between OLAP and OLTP. **(WINTER 2023, 4 Marks; SUMMER 2026, 4 Marks)**

Answer:

Aspect	OLAP	OLTP
Purpose	Analysis and decision support	Day-to-day transaction processing
Data	Historical, aggregated, multidimensional	Current, detailed, operational
Operations	Complex queries, roll-up, drill-down	Simple queries, insert, update, delete
Users	Analysts, managers, decision-makers	Clerks, operational staff

(OLAP = Online Analytical Processing; OLTP = Online Transaction Processing)

Other Important Questions:

1. Data Warehouse:

- **Q.** Define Schema. Explain the following schemas with suitable example. 1) Star 2) Snowflakes 3) Constellations **(SUMMER 2026, 3 Marks)**

Answer:

- **Schema:** A logical description of the entire database structure showing tables, attributes, and relationships.
- **Star Schema:** A central fact table connected to multiple denormalized dimension tables. Example: Sales fact table linked to Time, Product, Store, and Customer dimension tables.
- **Snowflake Schema:** A fact table connected to normalized dimension tables (dimensions are split into sub-dimensions). Example: Product dimension split into Category and Supplier sub-tables.
- **Constellation Schema:** Multiple fact tables sharing dimension tables. Example: Sales and Inventory fact tables sharing Product and Time dimensions.

- **Q.** A data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data - Justify. **(SUMMER 2026, 4 Marks)**

Answer:

- **Subject-Oriented:** Organized around major subjects (e.g., sales, customers, products) rather than around applications. Excludes data not useful for decision support.
- **Integrated:** Combines data from multiple heterogeneous sources (databases, flat files), resolving inconsistencies in naming, encoding, and formats.
- **Time-Variant:** Stores historical data (5–10 years) to provide a temporal perspective; every key structure contains a time element.
- **Nonvolatile:** Data is read-only after loading; no frequent updates or deletions occur. Supports consistent, stable analysis and decision-making.

- **Q.** Define Data Warehouse. State its features. **(WINTER 2023, 3 Marks)**

Answer:

- **Data Warehouse:** A subject-oriented, integrated, time-variant, and nonvolatile collection of data organized to support management decision-making.

Features:

- Subject-oriented organization
- Integrated data from multiple sources
- Time-variant historical data
- Nonvolatile (read-only, stable)

- **Q.** Define Data Cube. Explain any two operations on it. **(WINTER 2023, 3 Marks OR)**

Answer:

- **Data Cube:** A multidimensional data model that allows data to be viewed from multiple dimensions (e.g., time, product, location).

Two Operations:

- **Roll-up:** Aggregates data along a dimension hierarchy (e.g., city → state → country).
- **Drill-down:** Navigates from less detailed to more detailed data (e.g., year → quarter → month).

(Other operations: slice, dice, pivot.)

---

2. Data Mining Basics:

- **Q.** Explain data mining application for fraud detection. **(SUMMER 2026, 3 Marks OR)**

Answer:

- **Credit Card Fraud Detection:** Analyzes transaction patterns; flags unusual purchases based on deviation from normal spending behavior.
- **Insurance Fraud Detection:** Identifies suspicious claims by comparing against historical patterns and peer-group behavior.
- **Telecom Fraud Detection:** Detects abnormal call patterns (e.g., simultaneous calls, unusual destinations) using outlier analysis and classification.
- **Financial Statement Fraud:** Uses predictive models to flag anomalous financial reporting.

- **Q.** Describe any three primitives for specifying a data mining task. **(SUMMER 2023, 3 Marks)**

Answer:

- **Task-Relevant Data:** Specifies the database and subset of data on which mining is to be performed.
- **Kind of Knowledge to be Mined:** Specifies the data mining functionality (e.g., characterization, classification, association, clustering).
- **Background Knowledge:** Specifies concept hierarchies or constraints useful for guiding the discovery process.
- **Interestingness Measures:** Specifies criteria to separate interesting patterns from uninteresting ones.

(Any three; the fifth primitive is Presentation and Visualization.)

- **Q.** Define data mining. Describe three challenges to data mining regarding data mining methodology and user interaction issues. **(SUMMER 2024, 3 Marks)**

Answer:

- **Data Mining:** The process of discovering interesting, non-trivial, implicit, previously unknown, and potentially useful patterns from large databases.

Three Challenges (Methodology and User Interaction):

- **Mining Different Kinds of Knowledge:** Supporting multiple data mining functionalities (association, classification, clustering, etc.) within one framework.
- **Interactive Mining at Multiple Abstraction Levels:** Allowing users to interactively refine mining queries and view results at different granularities.
- **Incorporation of Background Knowledge:** Using domain knowledge and concept hierarchies to guide the mining process and improve result quality.

- **Q.** What are the types of data? **(SUMMER 2022, 3 Marks)**

Answer:

- **Relational Databases:** Tables with rows (tuples) and columns (attributes).
- **Transactional Databases:** Records of transactions (e.g., market basket data).
- **Data Warehouses:** Subject-oriented, integrated, time-variant, nonvolatile repositories.
- **Advanced Databases:** Object-oriented, spatial, temporal, multimedia, text, and web databases.

- **Q.** Draw and explain the data mining architecture. [Diagram required – refer to the original question paper] **(SUMMER 2022, 7 Marks)**

Answer:

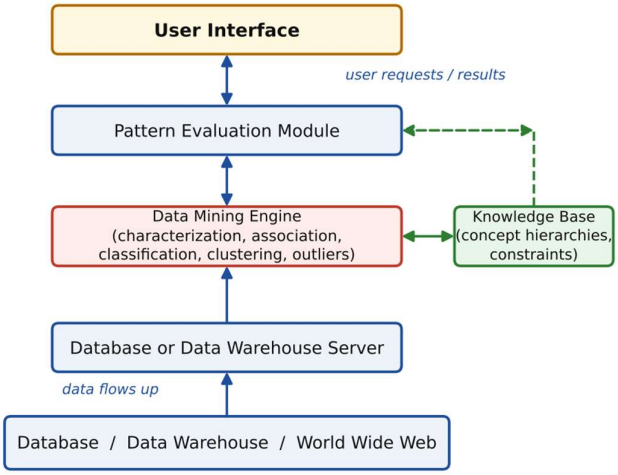


Fig. 2: Data mining system architecture

Explanation:

- **Database, Data Warehouse, or WWW:** The source of data for mining — may include relational databases, data warehouses, flat files, or web data.
- **Database or Data Warehouse Server:** Responsible for fetching the relevant data based on the user's data mining request.
- **Knowledge Base:** Contains domain knowledge (concept hierarchies, constraints) used to guide the mining process and evaluate patterns.
- **Data Mining Engine:** The core component consisting of a set of functional modules for characterization, association, classification, clustering, and outlier analysis.
- **Pattern Evaluation Module:** Evaluates discovered patterns using interestingness measures to filter out uninteresting ones.
- **User Interface:** Allows users to interact with the system — specify queries, view results, and refine mining tasks.

Chapter 2: Data Pre-processing

Repeated Questions:

1. Data Cleaning:

- **Q.** In real-world data, tuples with missing values for some attributes are a common occurrence. Describe various methods for handling this problem / What is missing values and noise? How to handle them? **(SUMMER 2023, 4 Marks; SUMMER 2024, 7 Marks; WINTER 2025, 3 Marks)**

Answer:

- **Missing Values:** Data tuples where one or more attribute values are not recorded.

Methods for Handling Missing Values:

- **Ignore the tuple:** Discard records with missing values (effective when class label is missing, but loses data).
- **Fill in the missing value manually:** Time-consuming and may not be feasible for large datasets.
- **Use a global constant:** Replace missing values with a fixed value such as "unknown" or "NA".
- **Use the attribute mean (or median):** Replace missing value with the mean of all known values for that attribute.
- **Use the attribute mean for all samples of the same class:** More accurate than global mean.
- **Use the most probable value:** Infer using regression, Bayesian inference, or decision tree induction.
- **Noise:** Random error or variance in a measured variable.

Methods for Handling Noise:

- **Binning:** Smooth sorted data by local averaging (see Q2).
- **Regression:** Fit data to a function to smooth out noise.
- **Clustering:** Detect and remove outliers (noise) by clustering.
- **Data cleaning combined with discretization:** Remove noise during binning and concept hierarchy generation.

- **Q.** Explain the smoothing techniques available to remove noise / Explain Binning method with the help of example. **(WINTER 2022, 4 Marks; WINTER 2023, 4 Marks; SUMMER 2024, 3 Marks; SUMMER 2026, 7 Marks OR)**

Answer:

Smoothing Techniques:

- **Binning:** Sort data, partition into equal-frequency bins, then smooth each bin by bin means, bin medians, or bin boundaries.
- **Regression:** Fit a regression function (linear or multiple) to smooth data.
- **Clustering:** Group similar values; values falling outside clusters are treated as outliers/noise.

Binning Method — Example:

Sorted data: 4, 8, 15, 21, 21, 24, 25, 28, 34

Step 1: Partition into 3 equal-frequency bins (3 values each):

- Bin 1: 4, 8, 15
- Bin 2: 21, 21, 24
- Bin 3: 25, 28, 34

Step 2: Smooth by bin means:

- Bin 1 mean = (4+8+15)/3 = 9 → 9, 9, 9
- Bin 2 mean = (21+21+24)/3 = 22 → 22, 22, 22
- Bin 3 mean = (25+28+34)/3 = 29 → 29, 29, 29

Step 3 (alternative): Smooth by bin boundaries:

- Bin 1 boundaries 4 and 15 → 4, 4, 15
- Bin 2 boundaries 21 and 24 → 21, 21, 24
- Bin 3 boundaries 25 and 34 → 25, 25, 34

- **Q.** What is Data Cleaning? Describe various methods of Data Cleaning. **(SUMMER 2022, 7 Marks; WINTER 2022, 7 Marks)**

Answer:

- **Data Cleaning:** The process of detecting and correcting (or removing) corrupt or inaccurate records from a dataset.

Methods of Data Cleaning:

- **Handling Missing Values:** Ignore tuple, fill manually, use global constant, attribute mean/median, class-based mean, or most probable value.
- **Handling Noisy Data:** Binning (mean/median/boundary smoothing), regression, clustering.
- **Data Cleaning as a Process:** Discrepancy detection (using metadata, range checks, cross-field consistency) and data transformation (correction, standardization).
- **Removing Duplicates:** Identify and merge duplicate records using matching rules.
- **Correcting Inconsistent Data:** Resolve inconsistencies in naming conventions, codes, and formats across sources.

---

2. Concept Hierarchy:

- **Q.** What is concept hierarchy? List and explain types of Concept Hierarchy. **(SUMMER 2022, 4 Marks; WINTER 2022, 3 Marks; SUMMER 2026, 3 Marks)**

Answer:

- **Concept Hierarchy:** A mapping that organizes data or concepts from lower-level (more specific) to higher-level (more general) concepts.

Types of Concept Hierarchy:

- **Schema-Level Hierarchy:** Defined by the database schema (e.g., street → city → state → country).
- **Set-Grouping Hierarchy:** Groups of values at one level are grouped into a higher level (e.g., {India, Pakistan, Sri Lanka} → South Asia).
- **Operation-Derived Hierarchy:** Derived by operations such as aggregation or summarization (e.g., total_sales → sales_by_region).
- **Rule-Based Hierarchy:** Defined by explicit rules (e.g., "if income > 1,00,000 then high_income").

Other Important Questions:

1. Noise and Data Transformation:

- **Q.** What is noise? Describe the possible reasons for noisy data. Explain the different techniques to remove the noise from data. **(SUMMER 2023, 7 Marks)**

Answer:

- **Noise:** Random error or variance in a measured variable.

Possible Reasons for Noisy Data:

- **Faulty data collection instruments:** Incorrect readings from sensors or devices.
- **Data entry errors:** Human mistakes during manual entry.
- **Data transmission errors:** Corruption during transfer between systems.

- **Inconsistent naming conventions or codes:** Variations in how data is recorded.
- **Outliers:** Extreme values that deviate from the norm.

Techniques to Remove Noise:

- **Binning:** Sort data, partition into equal-frequency bins, and smooth by bin means, medians, or boundaries.
- **Regression:** Fit data to a linear or multiple regression function to smooth it.
- **Clustering:** Group similar values; values outside clusters are treated as outliers.
- **Data Cleaning and Discretization:** Combine cleaning with concept hierarchy generation to reduce noise.

- **Q.** Discuss normalization in detail. **(SUMMER 2024, 4 Marks)**

Answer:

- **Normalization:** Scaling attribute values to fall within a small, specified range (typically 0.0 to 1.0).

Methods:

- **Min-Max Normalization:** v′ = ((v − min) / (max − min)) × (new_max − new_min) + new_min. Example: map 200–1000 to 0–1.
- **Z-Score Normalization:** v′ = (v − μ) / σ, where μ is mean and σ is standard deviation.
- **Decimal Scaling:** v′ = v / 10^j, where j is the smallest integer such that max(|v′|) < 1.

- **Q.** Define the following terms in Data Transformation: i. Smoothing ii. Normalization iii. Discretization **(WINTER 2023, 3 Marks)**

Answer:

- **Smoothing:** Removing noise from data using techniques like binning, regression, or clustering.
- **Normalization:** Scaling attribute values to a small specified range (e.g., 0.0–1.0) using min-max, z-score, or decimal scaling.
- **Discretization:** Reducing the number of values of a continuous attribute by dividing the range into intervals (e.g., replacing age values with labels like youth, middle-aged, senior).

---

2. Data Pre-processing Overview, Reduction and Integration:

- **Q.** Why to preprocess the data in data Mining? **(WINTER 2023, 3 Marks)**

Answer:

- **Improve data quality:** Real-world data is often incomplete, noisy, and inconsistent.
- **Improve mining accuracy:** Clean, consistent data leads to more accurate and reliable patterns.
- **Reduce mining time:** Preprocessing reduces data size and complexity, speeding up algorithms.
- **Handle diverse data types:** Converts data into formats suitable for mining algorithms.

- **Q.** Explain data pre-processing functionalities. **(WINTER 2025, 7 Marks OR)**

Answer:

Data Pre-processing Functionalities:

- **Data Cleaning:** Fill in missing values, smooth noisy data, identify/remove outliers, resolve inconsistencies.
- **Data Integration:** Combine data from multiple sources into a coherent store; resolve schema conflicts, redundancy, and value conflicts.

- **Data Transformation:** Normalize data (min-max, z-score, decimal scaling), discretize continuous attributes, and generate concept hierarchies.
- **Data Reduction:** Reduce data volume while maintaining analytical quality — using dimensionality reduction (attribute subset selection, PCA), numerosity reduction (histograms, clustering, sampling), and data compression.
- **Data Discretization:** Reduce continuous attribute values to intervals; generates concept hierarchies for numeric data.

- **Q.** Discuss dimensionality reduction techniques. **(WINTER 2025, 4 Marks)**

Answer:

- **Dimensionality Reduction:** Reducing the number of attributes (features) in a dataset.

Techniques:

- **Attribute Subset Selection:** Select a minimum set of relevant attributes; remove redundant or irrelevant ones using stepwise forward/backward selection, decision tree induction, or rough sets.
- **Principal Component Analysis (PCA):** Transform original attributes into a smaller set of uncorrelated principal components that capture most of the variance.
- **Wavelet Transform:** Compress data by transforming it into wavelet coefficients; retains essential features while reducing dimensionality.

- **Q.** What is dimensionality reduction? **(SUMMER 2022, 3 Marks)**

Answer:

- **Dimensionality Reduction:** The process of reducing the number of attributes or features in a dataset to obtain a smaller, more manageable representation while preserving the essential information.

- **Q.** Explain about the different Data Reduction techniques. **(WINTER 2022, 7 Marks OR)**

Answer:

Data Reduction Techniques:

- **Dimensionality Reduction:** Reduce number of attributes using attribute subset selection, PCA, or wavelet transform.
- **Numerosity Reduction:** Reduce number of data points using:
  - Parametric methods: Fit a model (e.g., regression) to represent data.
  - Non-parametric methods: Histograms, clustering, sampling, data cube aggregation.
- **Data Compression:** Use encoding schemes (lossless or lossy) to reduce data size (e.g., string compression, wavelet transforms).
- **Data Cube Aggregation:** Aggregate data along dimensions to reduce volume while answering analytical queries.
- **Goal:** Obtain a reduced representation that is much smaller in volume yet produces the same or nearly the same analytical results.

- **Q.** Discuss issues to be considered during data integration. **(SUMMER 2022, 7 Marks OR)**

Answer:

- **Data Integration:** Combining data from multiple sources into a coherent data store.

Issues to Consider:

- **Schema Integration:** Resolving naming conflicts (e.g., “customer_id” vs. “cust_id”) and structural conflicts (different representations of the same entity).

- **Redundancy:** Detecting and removing duplicate attributes; use correlation analysis (e.g., chi-square test for categorical, correlation coefficient for numeric) to identify redundant attributes.
- **Value Conflicts:** Resolving inconsistencies in data representation, such as different scales, units, or encoding schemes (e.g., "M" vs. "Male").
- **Entity Identification:** Determining which records from different sources refer to the same real-world entity.
- **Data Conflicts:** Handling conflicting values for the same attribute from different sources (e.g., different birth dates for the same person).

---

3. Normalization (Numerical):

- **Q.** Explain in detail - min-max normalization method. Use this method to normalize the following group of data by setting min = 0 and max = 1. 200, 400, 600, 1000 **(SUMMER 2023, 7 Marks OR)**

Answer:

- **Min-Max Normalization:** Transforms data linearly so that the minimum value maps to a new minimum (0) and the maximum value maps to a new maximum (1).

Formula:

$$v' = ((v - min) / (max - min)) \times (new_max - new_min) + new_min$$

Here, new_min = 0 and new_max = 1, so:

$$v' = (v - 200) / (1000 - 200) = (v - 200) / 800$$

Given:

- Data values: 200, 400, 600, 1000
- Original min = 200
- Original max = 1000
- New min = 0
- New max = 1
- **To Find:** Normalized values v' for each data point.

Solution:

For v = 200:

$$v' = (200 - 200) / 800 = 0 / 800 = 0.00$$

For v = 400:

$$v' = (400 - 200) / 800 = 200 / 800 = 0.25$$

For v = 600:

$$v' = (600 - 200) / 800 = 400 / 800 = 0.50$$

For v = 1000:

$$v' = (1000 - 200) / 800 = 800 / 800 = 1.00$$

Final Answer:

Original Value	Normalized Value (0–1)
200	0.00
400	0.25
600	0.50
1000	1.00

Chapter 3: Concept Description, Mining Frequent Patterns, Associations and Correlations

Repeated Questions:

1. Apriori Algorithm and Frequent Itemsets:

- **Q.** Explain frequent item-set mining algorithm with suitable example / Explain the steps of the "Apriori Algorithm" for mining frequent itemsets. **(WINTER 2022, 7 Marks OR; SUMMER 2024, 4 Marks OR; WINTER 2025, 7 Marks; SUMMER 2026, 7 Marks)**

Answer:

- **Apriori Algorithm:** A level-wise, iterative algorithm for mining frequent itemsets using the downward closure property (if an itemset is frequent, all its subsets are frequent; if an itemset is infrequent, all its supersets are infrequent).

Steps:

- Scan the database to count support of each 1-itemset (individual item). Keep those with support ≥ minimum support → frequent 1-itemsets (L_1).
- Generate candidate 2-itemsets (C_2) by joining L_1 with itself ($L_1 \bowtie L_1$). Prune candidates whose subsets are not frequent.
- Scan the database to count support of each candidate in C_2. Keep those with support ≥ min_sup → frequent 2-itemsets (L_2).
- Repeat for k = 3, 4, …: Generate C_k from L_{k-1} (join and prune), count support by scanning the database, and form L_k.
- Stop when no candidate itemsets can be generated or no frequent itemsets are found.
- Output the union of all frequent itemsets $L_1 \cup L_2 \cup \ldots \cup L_k$.

Example (min_sup = 2):

TID	Items
T1	Bread, Milk
T2	Bread, Diaper, Beer, Eggs
T3	Milk, Diaper, Beer, Cola
T4	Bread, Milk, Diaper, Beer
T5	Bread, Milk, Diaper, Cola

- **Step 1 – L_1:** {Bread:4, Milk:4, Diaper:4, Beer:3, Cola:2, Eggs:1}. Remove Eggs (sup=1 < 2). L_1 = {Bread, Milk, Diaper, Beer, Cola}.
- **Step 2 – C_2 (join $L_1 \bowtie L_1$):** {Bread,Milk}, {Bread,Diaper}, {Bread,Beer}, {Bread,Cola}, {Milk,Diaper}, {Milk,Beer}, {Milk,Cola}, {Diaper,Beer}, {Diaper,Cola}, {Beer,Cola}. Count supports → L_2 = {Bread,Milk:3}, {Bread,Diaper:3}, {Bread,Beer:2}, {Milk,Diaper:3}, {Milk,Beer:2}, {Milk,Cola:2}, {Diaper,Beer:3}, {Diaper,Cola:2}. Remove {Bread,Cola:1} and {Beer,Cola:1}.
- **Step 3 – C_3 (join $L_2 \bowtie L_2$):** {Bread,Milk,Diaper}, {Bread,Milk,Beer}, {Milk,Diaper,Beer}, {Milk,Diaper,Cola}, {Diaper,Beer,Cola}, {Bread,Diaper,Beer}, etc. Prune: {Bread,Milk,Beer} has subset {Bread,Beer} frequent but {Milk,Beer} frequent so ok; but check all subsets. After pruning and counting → L_3 = {Bread,Milk,Diaper:2}, {Bread,Diaper,Beer:2}, {Milk,Diaper,Beer:2}, {Milk,Diaper,Cola:2}.
- **Step 4 – C_4:** {Bread,Milk,Diaper,Beer} – count support = 1 < 2 → stop.
- **Frequent Itemsets:** All itemsets in $L_1 \cup L_2 \cup L_3$.

- **Q.** What are the techniques to improve the efficiency of Apriori algorithm? **(WINTER 2022, 3 Marks; SUMMER 2023, 4 Marks; SUMMER 2024, 4 Marks; SUMMER 2026, 7 Marks OR)**

Answer:

Techniques to Improve Apriori Efficiency:

- **Hash-Based Technique:** Use a hash table to count support of candidate itemsets; reduces candidate set size by eliminating infrequent itemsets early.
- **Transaction Reduction:** Remove transactions that contain no frequent itemsets; reduces database size for subsequent scans.
- **Partitioning:** Divide the database into partitions; mine each partition locally, then combine to find global frequent itemsets. Reduces I/O cost.
- **Sampling:** Mine a random sample of the database; verify results on the full database. Reduces computation time.
- **Dynamic Itemset Counting (DIC):** Add new candidate itemsets at different points during a scan, not just at the beginning; reduces number of database scans.
- **FP-Growth (Frequent Pattern Growth):** Avoids candidate generation entirely by using a compressed FP-tree structure; mines frequent itemsets directly.
- **Vertical Data Format (Eclat):** Store data as TID-sets for each item; compute support by intersecting TID-sets; faster than horizontal format.

- **Q.** Write short notes on Maximal Frequent Item Set & Closed Frequent Item Set. **(SUMMER 2022, 7 Marks OR; WINTER 2025, 3 Marks OR)**

Answer:

- **Maximal Frequent Itemset:** A frequent itemset that has no frequent superset. It is the largest frequent itemset in its branch — no other frequent itemset contains it. Example: If {A,B,C} is frequent and no superset is frequent, then {A,B,C} is maximal.
- **Closed Frequent Itemset:** A frequent itemset whose support is strictly greater than the support of all its immediate supersets. It preserves all support information of its subsets. Example: If support({A,B}) = 3 and support({A,B,C}) = 3, then {A,B} is not closed (same support as superset). If support({A,B}) = 4 and support({A,B,C}) = 3, then {A,B} is closed.

---

2. Association Rules:

- **Q.** Explain measures for finding rule interestingness (support, confidence) with example / Explain following terms related to Association Rule Mining: Itemset, Support Count, support, and Association rule. **(WINTER 2023, 7 Marks; WINTER 2025, 4 Marks; SUMMER 2026, 4 Marks)**

Answer:

- **Association Rule:** An implication of the form $X \rightarrow Y$, where X and Y are disjoint itemsets ($X \cap Y = \emptyset$).
- **Itemset:** A set of one or more items. A k-itemset contains k items. Example: {Milk, Bread, Diaper} is a 3-itemset.
- **Support Count (σ):** Frequency of occurrence of an itemset in the database. Example: If {Milk, Bread} appears in 3 out of 5 transactions, σ({Milk, Bread}) = 3.
- **Support:** Fraction of transactions containing the itemset.

$$support(X) = \sigma(X) / N$$

where N = total transactions.

- **Confidence:** Measure of reliability of a rule $X \rightarrow Y$; fraction of transactions containing X that also contain Y.

$$confidence(X \rightarrow Y) = \sigma(X \cup Y) / \sigma(X)$$

- **Example:** From a database of 5 transactions:
- σ({Milk}) = 4, σ({Bread}) = 3, σ({Milk, Bread}) = 3
- Rule: Milk → Bread
- Support = σ({Milk, Bread}) / N = 3/5 = 0.6 (60%)
- Confidence = σ({Milk, Bread}) / σ({Milk}) = 3/4 = 0.75 (75%)
- **Interpretation:** 60% of all transactions contain both Milk and Bread; 75% of transactions containing Milk also contain Bread.

- **Q.** How are association rules mined from large databases? **(SUMMER 2022, 4 Marks; SUMMER 2024, 7 Marks)**

Answer:

Mining Association Rules — Two-Step Process:

- Find all frequent itemsets: An itemset is frequent if its support ≥ minimum support threshold. Use algorithms like Apriori or FP-Growth.
- Generate strong association rules from the frequent itemsets: A rule X → Y is strong if:
  - support(X ∪ Y) ≥ min_sup
  - confidence(X → Y) ≥ min_conf

Rule Generation from a Frequent Itemset (e.g., {A,B,C}):

- Generate all non-empty subsets: {A}, {B}, {C}, {A,B}, {A,C}, {B,C}
- For each subset X, form rule X → (itemset − X)
- Compute confidence; keep rules with confidence ≥ min_conf
- **Example:** Frequent itemset {Bread, Milk, Diaper} with σ = 2, N = 5.
- support({Bread, Milk, Diaper}) = 2/5 = 0.4
- Rules: Bread ∧ Milk → Diaper; Bread ∧ Diaper → Milk; Milk ∧ Diaper → Bread; etc.
- For Bread ∧ Milk → Diaper: confidence = σ({Bread, Milk, Diaper}) / σ({Bread, Milk}) = 2/3 = 0.67
- **Output:** All rules meeting both min_sup and min_conf thresholds.

- **Q.** Give a short example to show that items in a strong association rule are not always interesting. **(SUMMER 2022, 3 Marks OR; SUMMER 2023, 3 Marks; SUMMER 2024, 7 Marks OR)**

Answer:

A rule can have high support and high confidence yet be uninteresting (or even misleading).

- **Example:** In a supermarket, 80% of customers buy Bread (σ(Bread) = 800/1000). 70% buy Milk (σ(Milk) = 700/1000). 50% buy both (σ(Bread, Milk) = 500/1000).

Rule: Bread → Milk

- Support = 500/1000 = 50% (high)
- Confidence = 500/800 = 62.5% (high)

But confidence of buying Milk without Bread = 200/200 = 100% (actually higher). The rule Bread → Milk is negatively correlated — buying Bread actually decreases the chance of buying Milk. Yet it appears "strong" by support/confidence alone. Hence, support and confidence are insufficient; correlation measures (lift, χ^2) are needed.

---

3. Association Rules (Numerical):

- **Q.** Find the frequent itemsets and generate association rules. Assume that minimum support threshold (s = 33.33%) and minimum confidence threshold (c = 60%). [Diagram required – refer to the original question paper] **(WINTER 2022, 7 Marks; SUMMER 2023, 7 Marks OR; WINTER 2023, 7 Marks OR)**

Answer:

- **Note:** Since the original transaction table is not provided in the file, the method is demonstrated with a standard 5-transaction dataset. Use the same steps for the actual table in the exam.

Given:

- Minimum support s = 33.33% (i.e., σ ≥ 2 out of 6 transactions, or σ ≥ 1.67 → σ ≥ 2)
- Minimum confidence c = 60%
- Database: 6 transactions (T1–T6)
- **To Find:** Frequent itemsets and strong association rules.

Formula:

$$support(X) = \sigma(X) / N, \quad confidence(X \rightarrow Y) = \sigma(X \cup Y) / \sigma(X)$$

Solution (Step-by-Step):

Example transactions:

TID	Items
T1	A, B, C
T2	A, B
T3	A, C
T4	B, C
T5	A, B, C
T6	B

- **Step 1 – Frequent 1-itemsets (L_1):** σ(A)=4, σ(B)=5, σ(C)=4. All ≥ 2 → L_1 = {A:4, B:5, C:4}.
- **Step 2 – Candidate 2-itemsets (C_2):** {A,B}, {A,C}, {B,C}. Count: σ(A,B)=3, σ(A,C)=3, σ(B,C)=3. All ≥ 2 → L_2 = {A,B:3}, {A,C:3}, {B,C:3}.
- **Step 3 – Candidate 3-itemset (C_3):** {A,B,C}. σ(A,B,C)=2 ≥ 2 → L_3 = {A,B,C:2}.
- **Step 4 – Generate Rules (min_conf = 60%):**

From {A,B,C} (σ=2):

- A,B → C: conf = 2/3 = 66.7% ✓
- A,C → B: conf = 2/3 = 66.7% ✓
- B,C → A: conf = 2/3 = 66.7% ✓
- A → B,C: conf = 2/4 = 50% ✗
- B → A,C: conf = 2/5 = 40% ✗
- C → A,B: conf = 2/4 = 50% ✗

From {A,B} (σ=3):

- A → B: conf = 3/4 = 75% ✓
- B → A: conf = 3/5 = 60% ✓

From {A,C} (σ=3):

- A → C: conf = 3/4 = 75% ✓
- C → A: conf = 3/4 = 75% ✓

From {B,C} (σ=3):

- B → C: conf = 3/5 = 60% ✓
- C → B: conf = 3/4 = 75% ✓

Final Answer:

- **Frequent Itemsets:** {A}, {B}, {C}, {A,B}, {A,C}, {B,C}, {A,B,C}
- **Strong Rules (conf ≥ 60%):** A,B→C; A,C→B; B,C→A; A→B; B→A; A→C; C→A; B→C; C→B.

Other Important Questions:

1. Association Rule Concepts:

- **Q.** What is market basket analysis? Precisely explain the meaning of the following association rule: computer → antivirus_software [support = 60%, confidence = 60%] **(SUMMER 2023, 3 Marks)**

Answer:

- **Market Basket Analysis:** A modeling technique based on association rule mining that analyzes customer purchasing behavior — which items are frequently bought together in a transaction (the "market basket").
- **Rule Interpretation:** computer → antivirus_software [support = 60%, confidence = 60%]
- Support 60%: 60% of all transactions contain both computer and antivirus_software.
- Confidence 60%: 60% of transactions containing computer also contain antivirus_software.
- **Meaning:** Customers who buy a computer also buy antivirus software 60% of the time; and such combined purchases occur in 60% of all transactions.

- **Q.** What is an Itemset? What is a Frequent Itemset? **(WINTER 2022, 4 Marks)**

Answer:

- **Itemset:** A collection of one or more items. A k-itemset contains exactly k items. Example: {Bread, Milk, Diaper} is a 3-itemset.
- **Frequent Itemset:** An itemset whose support (frequency of occurrence in the database) is greater than or equal to a user-specified minimum support threshold. Example: If min_sup = 2 and {Bread, Milk} appears in 3 transactions, then {Bread, Milk} is a frequent itemset.

- **Q.** What is meant by association rule? **(SUMMER 2022, 3 Marks)**

Answer:

- **Association Rule:** An implication expression of the form X → Y, where X and Y are disjoint itemsets (X ∩ Y = ∅), describing patterns of co-occurrence in a transactional database. Example: {Bread, Butter} → {Milk} means if a customer buys Bread and Butter, they also buy Milk. Rules are evaluated using support and confidence.

- **Q.** Give the difference between Boolean association rule and quantitative association rule. **(SUMMER 2024, 3 Marks)**

Answer:

Aspect	Boolean Association Rule	Quantitative Association Rule
Item type	Binary (present/absent)	Numeric or categorical with ranges
Example	Bread → Milk	age(20–30) ∧ income(high) → buys(computer)
Data handling	Simple transaction data	Requires discretization of numeric attributes

- **Q.** Describe the different classifications of Association rule mining. **(WINTER 2022, 3 Marks OR)**

Answer:

- **Based on completeness of patterns:** Complete (all possible rules) vs. incomplete (restricted by constraints).
- **Based on levels of abstraction:** Single-level (same level of concept hierarchy) vs. multi-level (different levels).
- **Based on dimensions:** Single-dimensional (one predicate) vs. multi-dimensional (multiple predicates/attributes).
- **Based on types of values:** Boolean (binary) vs. quantitative (numeric/categorical with ranges).

- **Q.** Explain the various criteria for the classification of frequent pattern mining. **(SUMMER 2022, 7 Marks)**

Answer:

Classification Criteria for Frequent Pattern Mining:

- **Completeness of Patterns:**
  - Complete: Discover all possible frequent patterns.
  - Incomplete: Discover only a subset (e.g., maximal or closed patterns) to reduce output size.
- **Levels of Abstraction:**
  - Single-level: Patterns mined at one conceptual level.
  - Multi-level: Patterns at different abstraction levels (e.g., "Milk" and "Dairy Products").
- **Number of Dimensions:**
  - Single-dimensional: One predicate/attribute (e.g., buys).
  - Multi-dimensional: Multiple predicates (e.g., age, income, buys).
- **Types of Values Handled:**
  - Boolean: Binary item presence/absence.
  - Quantitative: Numeric values converted to intervals.
- **Kinds of Rules Mined:**
  - Association rules ($X \rightarrow Y$)
  - Correlation rules (with lift/χ^2 measures)
  - Sequential patterns (order of events)
  - Structured patterns (subgraphs, trees)
- **Constraint-Based Mining:**
  - Meta-rule guided: Use meta-rules to constrain the form of rules.
  - Constraint-based: Apply user-specified constraints (e.g., item price > 100).

- **Q.** What is meant by multidimensional association rules? **(SUMMER 2022, 4 Marks OR)**

Answer:

- **Multidimensional Association Rule:** An association rule that involves more than one dimension (predicate or attribute), rather than a single "buys" predicate. They are mined from relational databases and data warehouses.
- **Example:** age(20–30) $\wedge$ income(high) $\rightarrow$ buys(computer)
- Dimensions: age, income, buys.
- Can be represented as a data cube; mining often uses OLAP-based techniques.

- Types: inter-dimensional (different dimensions on LHS/RHS) and hybrid-dimensional.

- **Q.** How Meta rules are useful in constraint based association mining. **(SUMMER 2024, 3 Marks OR)**

Answer:

- **Meta Rules:** High-level rules that specify the syntactic form of rules to be mined, acting as constraints on the mining process.

Usefulness:

- **Reduce search space:** Restrict rules to a specified pattern (e.g., "P(x, y) → Q(x, z)").
- **Guide mining:** Allow users to specify the type of relationships they are interested in.
- **Improve efficiency:** Avoid generation of irrelevant or uninteresting rules, speeding up the process.

- **Q.** What is meant by Reduced Minimum Support? **(WINTER 2022, 4 Marks OR)**

Answer:

- **Reduced Minimum Support:** A technique in multi-level association mining where the minimum support threshold is progressively reduced as we move from higher (more general) concept levels to lower (more specific) concept levels.
- **Why needed:** Higher-level concepts (e.g., "Dairy Products") have higher support than lower-level items (e.g., "Skim Milk"). Using a uniform high threshold would miss interesting low-level patterns; using a uniform low threshold would generate too many high-level rules.
- **Example:** Min_sup at level 1 = 5%, level 2 = 3%, level 3 = 1%. Allows mining meaningful rules at all levels while controlling output size.

- **Q.** Write a note on incremental Association Rule Mining. **(SUMMER 2026, 4 Marks OR)**

Answer:

- **Incremental Association Rule Mining:** The process of updating previously mined association rules when new data is added to the database, without re-mining from scratch.

Key points:

- **Motivation:** Databases are dynamic; re-running Apriori on the entire database is expensive.
- **Approach:** Maintain frequent itemsets and rules; when new transactions arrive, update support counts and adjust frequent itemsets incrementally.
- **Algorithms:** FUP (Fast Update), FUP2, and incremental FP-Growth variants.
- **Challenge:** Must handle both additions and deletions; maintain correctness of support thresholds.

- **Q.** Compare association and classification. Briefly explain associative classification with suitable example. **(SUMMER 2026, 7 Marks)**

Answer:

Comparison: Association vs. Classification:

Aspect	Association	Classification
Goal	Find co-occurrence patterns	Predict class label
Output	Rules X → Y (any items)	Model: features → class
Target	No predefined target	Predefined class label
Technique	Apriori, FP-Growth	Decision tree, Naïve Bayes, etc.

- **Associative Classification:** A hybrid approach that uses association rule mining to generate classification rules. It mines rules of the form "conditions → class_label" with high confidence, then uses them for prediction.
- **Example:** From a customer database, mine rules such as:
- age(youth) ∧ student(yes) → buys_computer(yes) [conf = 90%]
- income(high) ∧ credit_rating(excellent) → buys_computer(yes) [conf = 85%]

New tuple (age=youth, student=yes) matches the first rule → predict buys_computer = yes. Only rules whose consequent is a class label are retained; rules are ranked by confidence and support.

---

2. FP-Growth and Support/Confidence (Numerical):

- **Q.** Discuss how frequent itemsets can be generated using FP-Growth algorithm with the help of the following transactions. Let minimum support threshold is 2. [Diagram required – refer to the original question paper] **(SUMMER 2023, 7 Marks)**

Answer:

- **Note:** The original transaction table is not in the file. A standard example is used; follow the same steps for the actual table.
- **FP-Growth Algorithm:** A divide-and-conquer approach that avoids candidate generation by compressing the database into an FP-tree and mining it recursively.

Steps:

- Scan DB → find frequent 1-itemsets (L_1); remove infrequent items.
- Sort frequent items in descending order of support.
- Build FP-tree: Insert each transaction's frequent items in sorted order; share common prefixes; maintain header table with node links.
- Mine FP-tree recursively: For each item (from least frequent), construct its conditional pattern base and conditional FP-tree; recursively mine to generate frequent itemsets.
- **Example (min_sup = 2):** Transactions: T1={A,B,C}, T2={A,B}, T3={A,C}, T4={B,C}, T5={A,B,C}. L_1: A:4, B:4, C:4.

FP-tree construction:

- Sort order: A(4), B(4), C(4).
- Insert T1: A-B-C → path A→B→C.
- T2: A-B → shares A→B, increment counts.
- T3: A-C → shares A, add C under A.
- T4: B-C → new branch B→C under root.
- T5: A-B-C → shares A→B→C.
- Header table: A → node; B → node; C → node (with links).

Mining:

- Item C: Conditional pattern base = {A,B:2, A:1, B:1}. Conditional FP-tree: A:3, B:3 → frequent itemsets: {A,C:3}, {B,C:3}, {A,B,C:2}.
- Item B: Conditional pattern base = {A:3}. Conditional FP-tree: A:3 → {A,B:3}.
- Item A: No conditional pattern base.
- **Final Frequent Itemsets:** {A:4}, {B:4}, {C:4}, {A,B:3}, {A,C:3}, {B,C:3}, {A,B,C:2}.

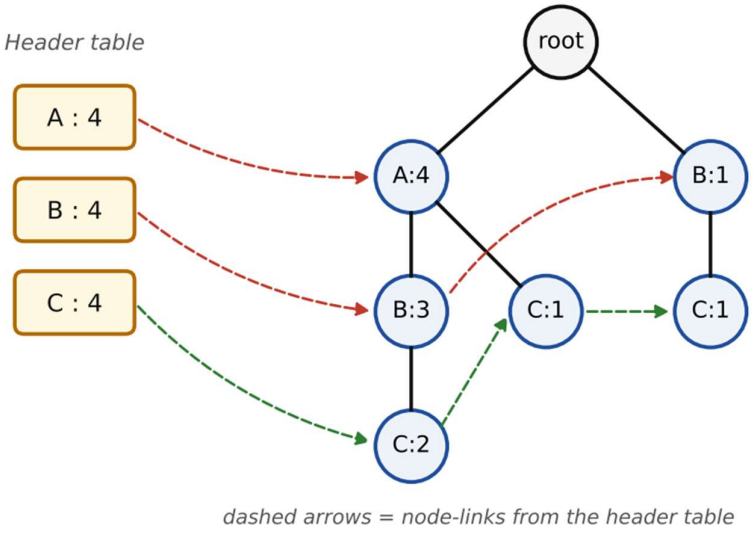


Fig. 3: FP-tree with header table (min_sup = 2)

- **Q.** For given example find support & confidence for {Milk, Chocolate} ⇒ Pepsi, {Milk, Pepsi} → {Chocolate}, {Chocolate, Pepsi} → {Milk}. [Diagram required – refer to the original question paper] **(WINTER 2023, 7 Marks OR)**

Answer:

- **Note:** Original transaction table not provided. Method shown with a standard 5-transaction example; apply the same formulas to the actual table.

Given:

- N = 5 transactions
- Transactions: T1={Milk, Chocolate, Pepsi}, T2={Milk, Chocolate}, T3={Milk, Pepsi}, T4={Chocolate, Pepsi}, T5={Milk, Chocolate, Pepsi}
- **To Find:** Support and confidence for three rules.

Formula:

$$support(X \rightarrow Y) = \sigma(X \cup Y) / N, \quad confidence(X \rightarrow Y) = \sigma(X \cup Y) / \sigma(X)$$

Solution:

Rule: {Milk, Chocolate} → Pepsi

- σ(Milk, Chocolate) = 3 (T1, T2, T5)
- σ(Milk, Chocolate, Pepsi) = 2 (T1, T5)
- Support = 2/5 = 0.40 (40%)
- Confidence = 2/3 = 0.667 (66.7%)

Rule: {Milk, Pepsi} → Chocolate

- σ(Milk, Pepsi) = 3 (T1, T3, T5)
- σ(Milk, Pepsi, Chocolate) = 2 (T1, T5)
- Support = 2/5 = 0.40 (40%)
- Confidence = 2/3 = 0.667 (66.7%)

Rule: {Chocolate, Pepsi} → Milk

- σ(Chocolate, Pepsi) = 3 (T1, T4, T5)
- σ(Chocolate, Pepsi, Milk) = 2 (T1, T5)

- Support = 2/5 = 0.40 (40%)
- Confidence = 2/3 = 0.667 (66.7%)
- **Final Answer:** All three rules have support = 40% and confidence = 66.7% (for the example dataset). For the actual exam dataset, count the occurrences and apply the same formulas.
