1. What are the basic building block of data warehouse

Source Data Component :Source data entering the data warehouse can be classified into four categories:

- **Production Data**: Data from operational systems (e.g., ERP, CRM) that supports day-to-day business operations.
- **Internal Data**: Private data within the organization, including spreadsheets, reports, and department databases.
- **Archived Data**: Historical data from operational systems, stored for long-term reference.
- **External Data**: Data sourced externally, often for industry insights, trends, and statistics.

Data Staging Component :This is the preparation area where data extracted from various sources is cleaned, transformed, and formatted for storage.

- **Data Extraction**: Pulling data from various operational systems and external sources.
- **Data Transformation**: Involves cleaning (e.g., correcting errors), standardizing, integrating, and summarizing data to ensure consistency.
- **Data Loading**: Initial loading of large volumes of data into the data warehouse and subsequent incremental loads.

Data Storage Components :Data is stored in a split repository, optimized for efficient querying. Operational systems hold only current data, while the data warehouse stores historical, normalized data for analysis.

Information Delivery Component :Facilitates the transfer of data from the data warehouse to various destinations according to user requirements, often using scheduled deliveries.

Metadata Component :Metadata acts as a data dictionary, describing data structures, relationships, and definitions within the warehouse, ensuring consistency and clarity.

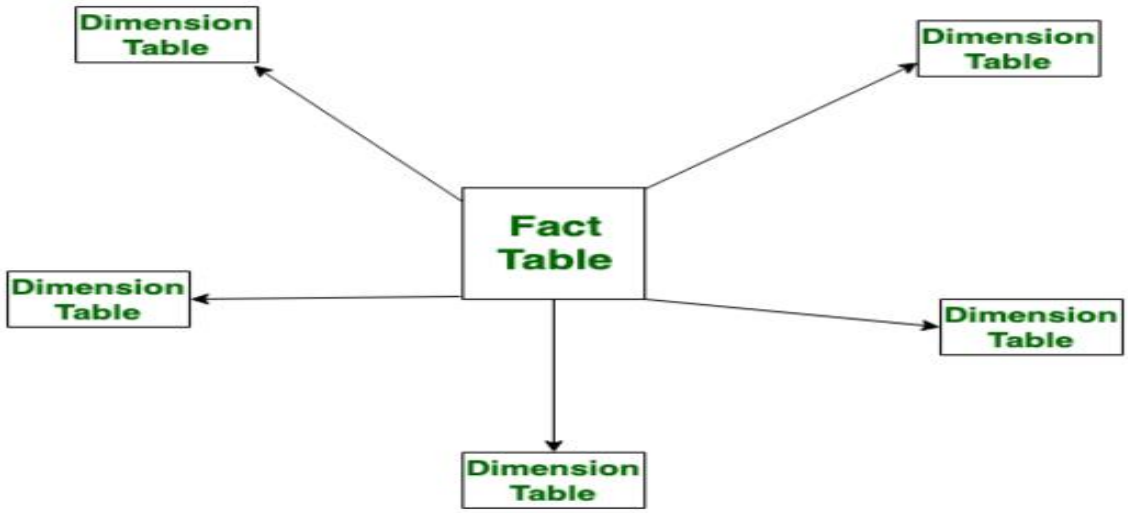
 Data Marts :Smaller, subject-specific subsets of data warehouses designed for particular user groups. These allow quicker and more focused reporting and querying.

 Management and Control Component :Coordinates the movement, transformation, and delivery of data. It ensures data is correctly transferred into the storage and available for clients as needed.

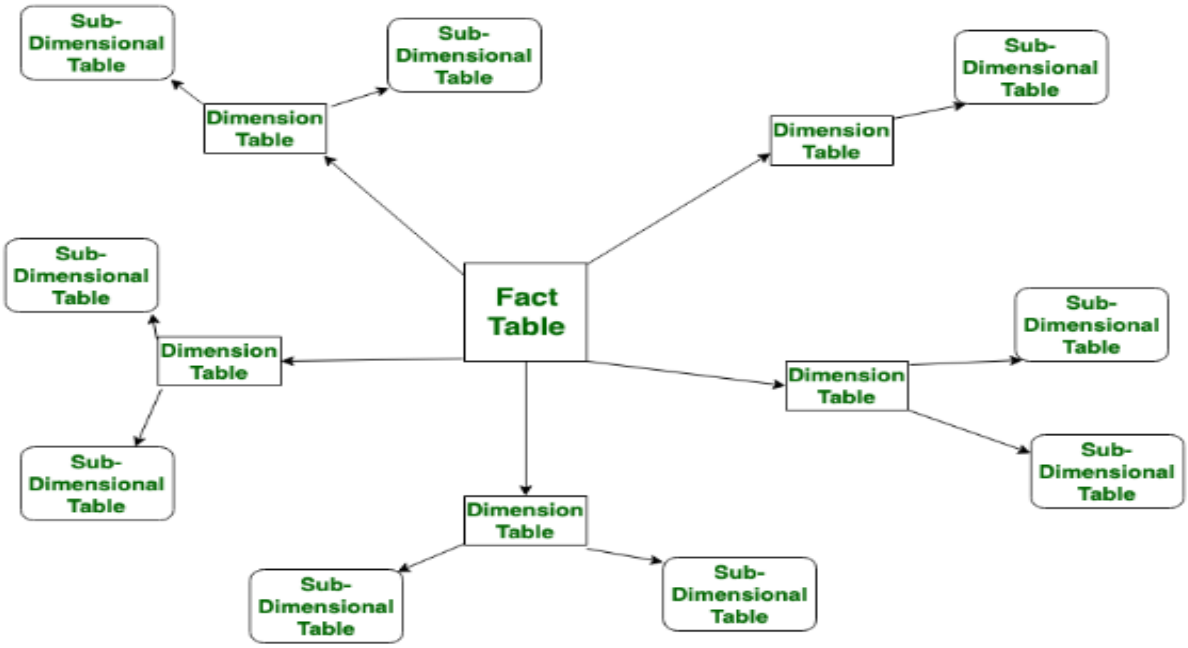
---

2. Explain star snowflake and fact consetallation schema for multi dimensional database.

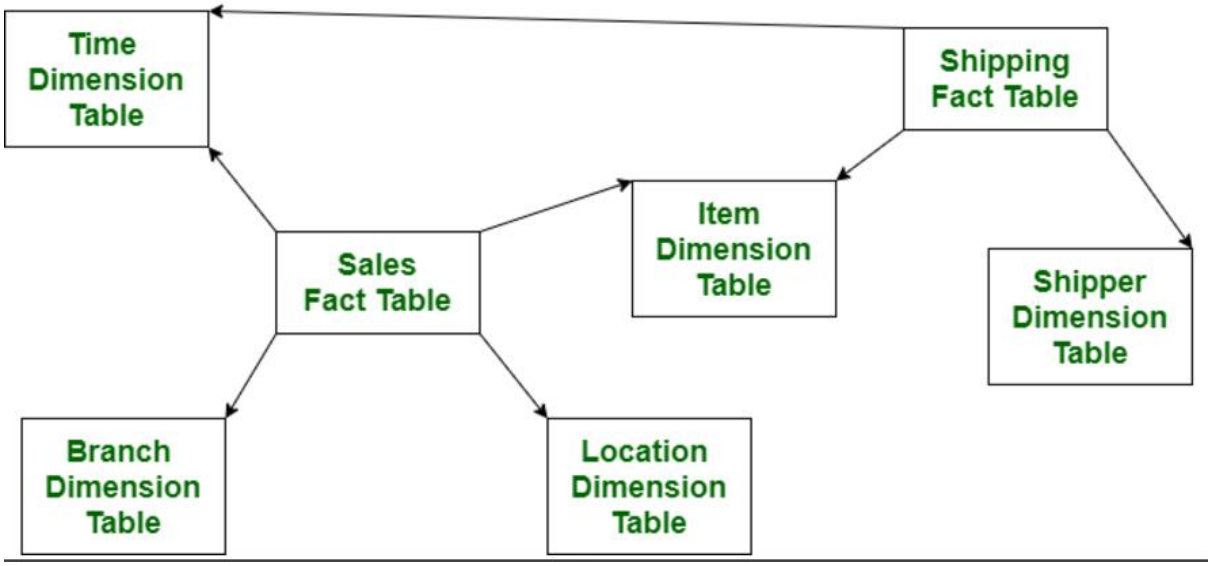
Star schema : is the type of multidimensional model that is used for data warehouse. In a star schema, The fact tables and the dimension tables are contained. In this schema, fewer foreign-key join is used. This schema forms a star with a fact table and dimension tables.



Snowflake Schema is also the type of multidimensional model which is used for data warehouse. In snowflake schema, The fact tables, dimension tables as well as sub dimension tables are contained. This schema forms a snowflake with fact tables, dimension tables as well as sub-dimension tables.



This schema is a type of multidimensional model. In this, dimension tables are shared by many fact tables. The fact constellation schema consists of more than one star schema at a time. Unlike the star schema, it is not easy to operate, as it has more joins between the tables. Unlike the Star schema, fact constellation schema uses heavily complex queries to access data from the database.



S. No.	TOP DOWN APPROACH	BOTTOM UP APPROACH
1.	In this approach We focus on breaking up the problem into smaller parts.	In bottom up approach, we solve smaller problems and integrate it as whole and complete the solution.
2.	Mainly used by structured programming language such as COBOL, Fortran, C, etc.	Mainly used by object oriented programming language such as C++, C#, Python.
3.	Each part is programmed separately therefore contain redundancy.	Redundancy is minimized by using data encapsulation and data hiding.
4.	In this the communications is less among modules.	In this module must have communication.
5.	It is used in debugging, module documentation, etc.	It is basically used in testing.
6.	In top down approach, decomposition takes place.	In bottom up approach composition takes place.
7.	In this top function of system might be hard to identify.	In this sometimes we can not build a program from the piece we have started.
8.	In this implementation details may differ.	This is not natural for people to assemble.

Advantages of top-down design

Data Marts are loaded from the data warehouses.

Developing new data mart from the data warehouse is very easy.

Disadvantages of top-down design

This technique is inflexible to changing departmental needs.

The cost of implementing the project is high.

Advantages of bottom-up design

Documents can be generated quickly.

The data warehouse can be extended to accommodate new business units.

It is just developing new data marts and then integrating with other data marts.

Disadvantages of bottom-up design

the locations of the data warehouse and the data marts are reversed in the bottom-up approach design.

---

3. WHAT IS METADATA IN DATA WAREHOUSE

Metadata in a Data Warehouse refers to data that describes other data. It is essentially the "data about the data" that helps users understand, manage, and navigate the data warehouse. Metadata provides crucial information about the structure, relationships, and properties of the data stored in the warehouse, making it easier to interpret and query the data. There are several types of metadata in a data warehouse:

1. **Business Metadata**:
    - Describes the data in business terms, such as what each field or metric represents in a way that business users can understand.
    - For example, definitions of key performance indicators (KPIs) or business rules.

2. **Technical Metadata**:
    - Provides detailed technical information about the data, such as data types, formats, table structures, column definitions, and relationships between tables.
    - It helps in understanding how data is stored and processed in the data warehouse.
3. **Operational Metadata**:
    - Describes the processes and operations related to data, including data lineage (the flow of data from source to destination), load history, refresh times, and data quality metrics.
    - This is useful for tracking data updates, transformations, and ensuring data integrity.
4. **Extraction and Transformation Metadata**:
    - Involves data transformations, including rules and logic used during data extraction, cleaning, and transformation in the ETL (Extract, Transform, Load) process.

Importance of Metadata:

- **Data Understanding**: It helps users and analysts understand the data without having to inspect the actual data sets.
- **Data Management**: Metadata helps in managing data quality, transformations, and integrations, ensuring consistency across the data warehouse.
- **Query Optimization**: It aids in optimizing data queries by providing information about indexes, table structures, and relationships.
- **Data Governance**: Metadata plays a key role in data governance by ensuring that data usage is transparent, well-documented, and compliant with organizational policies.

---

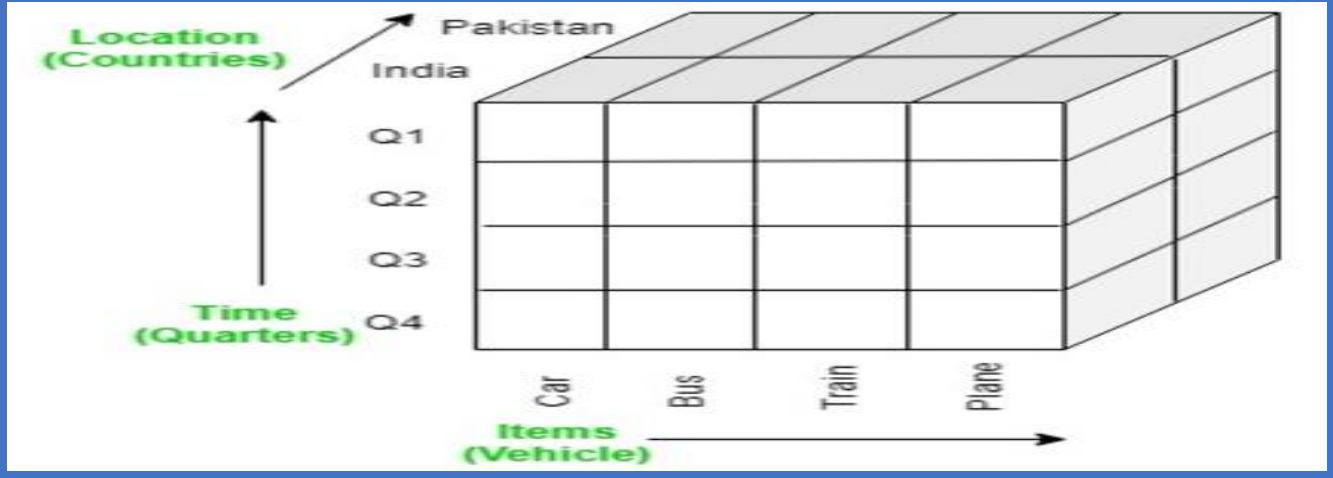
4. EXPLAIN OLAP OPERATION

OLAP stands for ***Online Analytical Processing*** Server. It is a software technology that allows users to analyze information from multiple database systems at the same time. It is based on multidimensional data model and allows the user to query on multi-dimensional data (eg. Delhi -> 2018 -> Sales data). OLAP databases are divided into one or more cubes and these cubes are known as *Hyper-cubes*.

There are five basic analytical operations that can be performed on an OLAP cube:

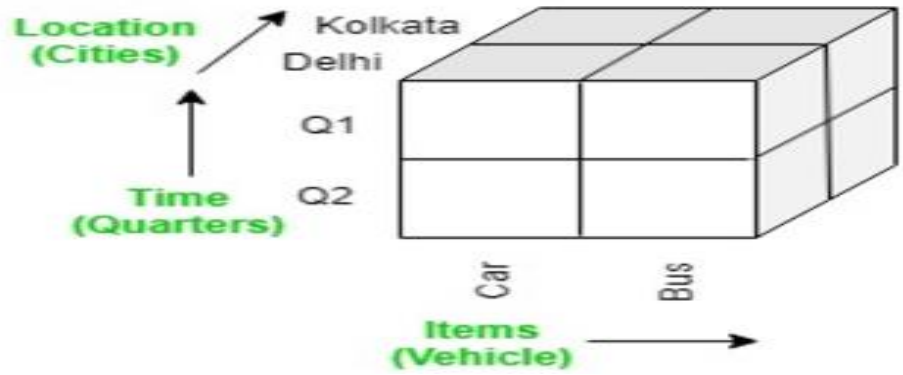
1. **Drill down:** In drill-down operation, the less detailed data is converted into highly detailed data. It can be done by:
    - Moving down in the concept hierarchy
    - Adding a new dimension

In the cube given in overview section, the drill down operation is performed by moving down in the concept hierarchy of *Time* dimension (Quarter -> Month).



Dice: It selects a sub-cube from the OLAP cube by selecting two or more dimensions. In the cube given in the overview section, a sub-cube is selected by selecting following dimensions with criteria:

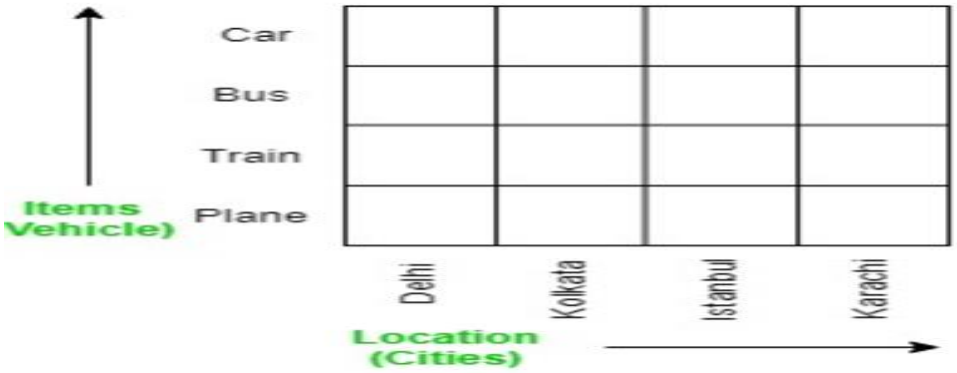
- Location = “Delhi” or “Kolkata”
- Time = “Q1” or “Q2”
- Item = “Car” or “Bus”



Slice: It selects a single dimension from the OLAP cube which results in a new sub-cube creation. In the cube given in the overview section, Slice is performed on the dimension Time = “Q1”.

Pivot: It is also known as *rotation* operation as it rotates the current view to get a new view of the representation. In the sub-cube obtained after the slice operation, performing pivot

operation gives a new view of it.



---

5. What is Dimensional Modeling?

Dimensional modeling represents data with a cube operation, making more suitable logical data representation with OLAP data management. The perception of Dimensional Modeling was developed by **Ralph Kimball** and is consist of **"fact"** and **"dimension"** tables.

- In dimensional modeling, the transaction record is divided into either **"facts,"** which are frequently numerical transaction data, or **"dimensions,"** which are the reference information that gives context to the facts. For example, a sale transaction can be damage into facts such as the number of products ordered and the price paid for the products, and into dimensions such as order date, user name, product number, order ship-to, and bill-to locations, and salesman responsible for receiving the order.

Advantages of Dimensional Modeling

Following are the benefits of dimensional modeling are:

- Dimensional modeling is simple
- Dimensional modeling promotes data quality:
- Performance optimization is possible through aggregates:

Disadvantages of Dimensional Modeling

To maintain the integrity of fact and dimensions, loading the data warehouses with a record from various operational systems is complicated.
It is severe to modify the data warehouse operation if the organization adopting the dimensional technique changes the method in which it does business.

---

6. TECHNIUES OF DATA LOADINNG

The data warehouse is structured by the integration of data from different sources. Several factors separate the data warehouse from the operational database. Since the two systems provide vastly different functionality and require different types of data, it is necessary to keep the data database separate from the operational database. A data warehouse is an exchequer of acquaintance gathered from multiple sources, picked under a unified schema, and usually residing on a single site. A data warehouse is built through the process of data cleaning, data integration, data transformation, data loading, and periodic data refresh.

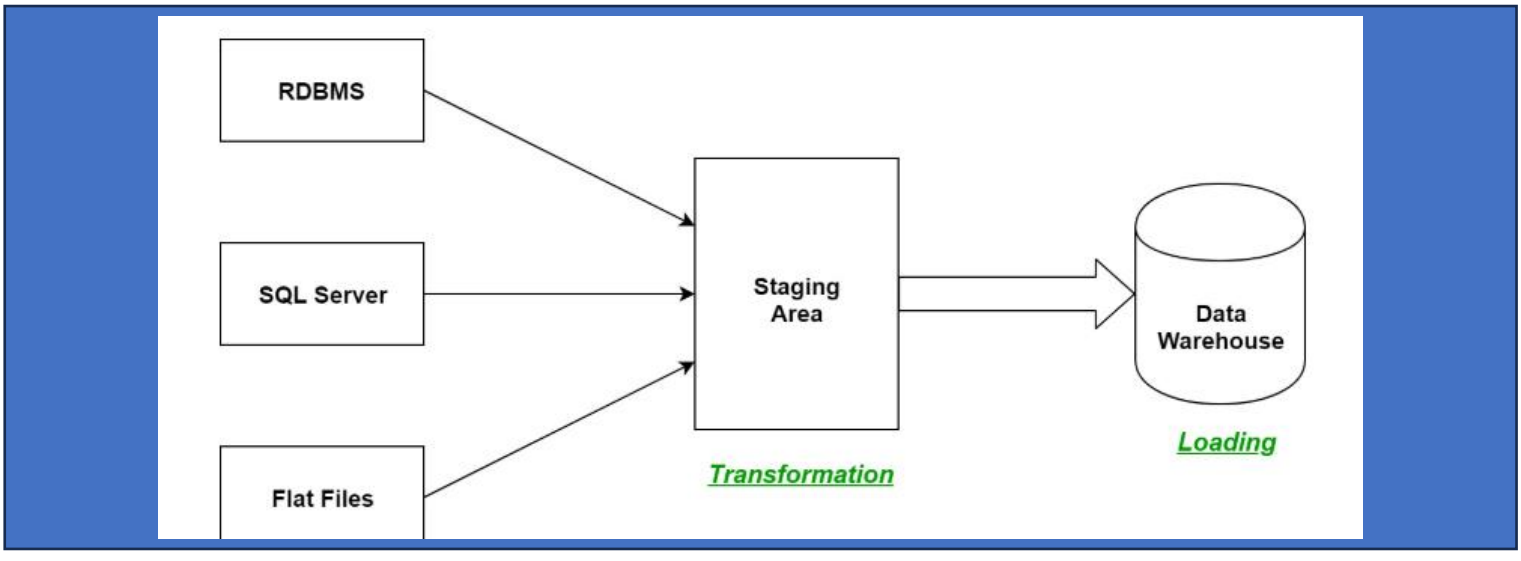
7. Illustrate major steps in ETL process.

ETL stands for Extract, Transform, Load and it is a process used in data warehousing to extract data from various sources, transform it into a format suitable for loading into a data warehouse, and then load it into the warehouse. The process of ETL can be broken down into the following three stages:

Extract: The first stage in the ETL process is to extract data from various sources such as transactional systems, spreadsheets, and flat files. This step involves reading data from the source systems and storing it in a staging area.

Transform: In this stage, the extracted data is transformed into a format that is suitable for loading into the data warehouse. This may involve cleaning and validating the data, converting data types, combining data from multiple sources, and creating new data fields.

Load: After the data is transformed, it is loaded into the data warehouse. This step involves creating the physical data structures and loading the data into the warehouse.



Extraction: The first step of the ETL process is extraction. In this step, data from various source systems is extracted which can be in various formats like relational databases, No SQL, XML, and flat files into the staging area. It is important to extract the data from various source systems and store it into the staging area first and not directly into the data warehouse because the extracted data is in various formats and can be corrupted also. Hence loading it directly into the data warehouse may damage it and rollback will be much more difficult. Therefore, this is one of the most important steps of ETL process.

Transformation: The second step of the ETL process is transformation. In this step, a set of rules or functions are applied on the extracted data to convert it into a single standard format. It may involve following processes/tasks:

- Filtering – loading only certain attributes into the data warehouse.
- Cleaning – filling up the NULL values with some default values, mapping U.S.A, United States, and America into USA, etc.
- Joining – joining multiple attributes into one.
- Splitting – splitting a single attribute into multiple attributes.
- Sorting – sorting tuples on the basis of some attribute (generally key-attribute). **Loading:** The third and final step of the ETL process is loading. In this step, the transformed data is finally loaded into the data warehouse. Sometimes the data is updated by loading into the data warehouse very frequently and sometimes it is done after longer but regular intervals.

Module 2

8. Describe any five issues of data mining

1. Data Quality: Poor quality data, such as missing values, noisy data, and inconsistencies, can lead to inaccurate results. Ensuring high-quality data is essential for meaningful analysis.
2. Scalability: As the size of datasets increases, data mining algorithms may become inefficient or slow. Handling large volumes of data requires optimized algorithms and more computational power.

3. Privacy Concerns: Mining sensitive or personal data can lead to privacy violations. Proper data anonymization and protection are required to address privacy concerns.

4. Data Integration: Data often comes from multiple sources, which may have different formats and structures. Integrating this data into a unified form can be challenging.

5. Interpretability: Complex data mining models, such as neural networks, may be difficult to interpret. This lack of transparency can hinder trust in the results, especially in critical areas like healthcare or finance.

---

9. Explain different data visualization techniques.

Data visualization plays a crucial role in transforming large, complex datasets into understandable and actionable insights. By mapping data to graphical representations, users can gain deeper insights into the underlying patterns, trends, structures, and relationships within the data. Various visualization methods help explore these insights, leading to a better understanding of the data's significance.

Key Objectives of Data Visualization:

1. **Insight Discovery**: Data is mapped to graphical primitives (e.g., dots, lines) to help users visualize and understand complex datasets.
2. **Qualitative Overview**: Provides a visual overview of large datasets, highlighting trends and patterns.
3. **Pattern and Relationship Detection**: Helps search for patterns, trends, structures, and irregularities within the data.
4. **Quantitative Analysis Preparation**: Identifies interesting regions or parameters for deeper quantitative analysis.
5. **Proof of Data Representation**: Visualizes computer-generated data representations for better comprehension.

Pixel-Oriented Visualization Technique Definition: These techniques represent multidimensional data by mapping each dimension to a pixel, providing a visual overview of the dataset.

Mapping Values to Pixels: Each data point is represented as a pixel, with its color reflecting the corresponding attribute value (e.g., age, income).

Space Filling with Circle Segments: Pixels are arranged in circular segments to save space and enhance visualization of complex data relationships.

Geometric Projection Visualization Techniques Definition: These techniques visualize multidimensional data by applying geometric transformations and projections to represent the data in 2D or 3D space.

Direct Visualization: Displays data directly in scatterplots or scatterplot matrices to visualize relationships between variables.

Scatterplot Matrices: A matrix of scatterplots visualizes pairwise relationships between attributes of the dataset.

Landscapes: Transforms data into a 2D spatial representation, maintaining the data's characteristics while giving a perspective view.

Parallel Coordinates: Uses multiple equidistant axes, with each axis representing an attribute, and connects data points as lines across these axes to explore relationships.

Icon-Based Visualization Techniques Definition: These techniques use icons or images to represent data attributes, making it easier to visualize multivariate data with visual patterns.

Chernoff Faces: Attributes are mapped to facial features (e.g., eyebrow slant, eye size) to represent multivariate data visually.

Stick Figures: Data attributes are represented by the length and angle of limbs in a stick figure.

Shape and Color Coding: Icons are shaped or colored in specific ways to represent different data attributes or categories.

Hierarchical Visualization Techniques

Definition: These techniques partition data into subspaces or hierarchical structures to reveal relationships and patterns within complex datasets.

Dimensional Stacking: Partitions multidimensional data into 2D subspaces and stacks them for easier analysis of attribute values.

Worlds-within-Worlds: Fixes certain parameters and visualizes other dimensions as nested "worlds" within these fixed dimensions.

Tree-Map: Divides the screen into hierarchical regions, with the x and y axes representing attribute values to visualize hierarchical data.

InfoCube: Uses semi-transparent cubes to represent hierarchical data in 3D, with outer cubes representing higher-level data.

Cone Trees: A 3D tree structure where nodes are visualized as cones, useful for displaying hierarchical data interactively.

Visualizing Complex Data and Relationships

Definition: These methods focus on visualizing non-numerical data and complex relationships like social networks or user-generated content.

Non-Numerical Data: **Tag Clouds** represent user-generated content (such as keywords) with size and color indicating their importance or frequency.

Social Networks: Represents relationships in social networks by visualizing nodes (individuals or entities) and edges (their connections), helping to explore social structures and interactions.

---

10. Explain in brief what is data discretization and concept hierarchy generation

Data Discretization Data Discretization is the process of converting continuous data (numeric values) into discrete categories or intervals. This is useful in data mining, especially when algorithms work better with categorical data rather than continuous variables. The goal is to simplify data while retaining important patterns for analysis.

Example:

Converting age into age groups:

• Continuous data: 25, 30, 35, 40, 45

• Discretized data:

o 20-30: "Young"

o 31-40: "Middle-aged"

o 41-50: "Older"

Concept Hierarchy Generation Concept Hierarchy Generation involves organizing data into a hierarchy of concepts or levels of abstraction. It helps in representing data at multiple levels, from the most general to the most specific, making it easier to understand and analyze patterns.

Example: • Product Hierarchy:

o General level: "Electronics"

o More specific: "Mobile Phones"

o Most specific: "Smartphones"Concept hierarchies are helpful in data mining tasks such as clustering and classification, where data needs to be grouped or classified at various levels of detail.

---

11. K-means Clustering algorithm

K-Means is an unsupervised machine learning algorithm used to group data into K distinct clusters based on their features. It minimizes the variance within each cluster Steps of the

K-Means Algorithm:

1. Initialization:Choose the number of clusters (KKK) and randomly initialize KKK cluster centroids.

2. Assignment Step:Assign each data point to the cluster whose centroid is nearest (measured by a distance metric like Euclidean distance).

3. Update Step:Recalculate the centroids of the clusters by taking the mean of all data points assigned to that cluster.

4. Repeat:Repeat steps 2 and 3 until the centroids no longer change significantly or the maximum number of iterations is reached.

Example: Consider a dataset with points distributed in a 2D space.

1. Choose K=2K = 2K=2 clusters and randomly place two centroids.

2. Assign each point to the nearest centroid.

3. Recalculate the centroid for each cluster by finding the average position of points in that cluster.

4. Repeat until centroids stabilize

Advantages of K-Means:

1. Simple and Fast:Easy to implement and computationally efficient for large datasets.

2. Scalable:Works well with large datasets when the number of clusters (KKK) is small.

3. Flexible:Applicable to a wide variety of data types and domains.

4. Interpretability: Results are easy to understand and visualize.

Limitations of K-Means:

Choice of KKK:The algorithm requires the number of clusters (KKK) to be predefined, which can be difficult to determine.

12. EXPLAIN KDD PROCESS

KDD Process (Knowledge Discovery in Databases) The KDD process refers to extracting meaningful patterns, trends, or insights from large datasets. It is an iterative and multi-step process involving data preparation, transformation, and mining to derive useful knowledge.

Steps in the KDD Process:

1. Data Selection: o Identify and retrieve relevant data from a large database. o Ensure the data aligns with the objectives of the analysis.
2. Data Preprocessing: o Clean the data by handling missing values, outliers, and noise. o Reduce inconsistencies and ensure data quality.
3. Data Transformation: o Transform raw data into a suitable format. o This step may involve normalization, aggregation, or feature extraction.
4. Data Mining: o Apply algorithms to identify patterns, correlations, or trends. o Techniques include clustering, classification, regression, and association rule mining.
5. Pattern Evaluation: o Interpret and validate the patterns to ensure they are significant and actionable. o Use domain knowledge to assess relevance.
6. Knowledge Representation: o Present the discovered knowledge in a user-friendly format, such as reports, charts, or dashboards.

o Importance of KDD: • Helps in deriving actionable insights from vast amounts of data.

• Aids decision-making in fields like business, healthcare, and research.

• Supports predictive analysis and trend detection. The KDD process ensures a structured and efficient approach to transforming raw data into valuable knowledge.

13. EXPLAIN DATA PRE_PROCESSING

Data Preprocessing is the process of cleaning and organizing raw data into a usable format. This step is essential because real-world data is often incomplete, inconsistent, noisy, or contains irrelevant information. Data preprocessing ensures that the data is in the correct form for analysis or machine learning algorithms, improving the quality of the results.

Steps Involved in Data Preprocessing:

1. Data Cleaning: The primary goal of data cleaning is to handle missing, inconsistent, or erroneous data. Common tasks include:

o Handling Missing Data: Replace missing values with a default value (e.g., mean, median, or mode) or remove records with missing values.

o Handling Noise: Smooth out noisy data by techniques like binning or regression.

2.Data Integration: When data comes from multiple sources, integration involves combining the data into a unified view. This may include:

---

o Merging datasets from different databases.

o Resolving conflicts in the data from different sources (e.g., differing formats).

3.Data Transformation: This step involves converting data into an appropriate format or structure for analysis. Common transformations include:

o Normalization: Scaling data so it fits within a specific range (e.g., 0 to 1).

o Standardization: Rescaling data so it has a mean of 0 and a standard deviation of 1.

o Aggregation: Summarizing data (e.g., summing up sales by month).

1. Data Reduction: Reducing the size of data while maintaining its essential characteristics. Techniques include: o Dimensionality Reduction: Reducing the number of features using methods like Principal Component Analysis (PCA).

o Data Compression: Using algorithms to compress data while retaining important information.

5. Data Discretization: Converting continuous data into discrete categories. For example, converting ages into age groups (e.g., 0-18, 19-35, 36-60, 60+).

6. Feature Selection: Choosing the most relevant features (variables) to improve the performance of machine learning models. This step helps remove irrelevant or redundant data.

7. Encoding: Converting categorical data into a numerical format, especially for machine learning algorithms that work with numerical inputs. Common methods include:

o Label Encoding: Converting categories into integer labels.

o One-Hot Encoding: Creating binary columns for each category in a categorical feature.

---

14. DISUSS DIFFERNCE TYPES OF ATRIBUTES

Attribute (or dimensions, features, variables): a data field, representing a characteristic or feature of a data object.

■ E.g., customer _ID, name, address

\■ Types:

■ Nominal

■ Binary

■ Numeric: quantitative, Interval-scaled ,Ratio-scaled

Nominal: categories, states, or “names of things”

Hair_color = {auburn, black, blond, brown, grey, red, white}

■ marital status, occupation, ID numbers, zip codes

Binary

- Nominal attribute with only 2 states (0 and 1)
- Symmetric binary: both outcomes equally important
- e.g., gender
- Asymmetric binary: outcomes not equally important.
- e.g., medical test (positive vs. negative)
- Convention: assign 1 to most important outcome (e.g., HIV positive)

■ Ordinal

- Values have a meaningful order (ranking) but magnitude between successive values is not known.
- Size = {small, medium, large}, grades, army rankings 8 Numeric Attribute Types
- Quantity (integer or real-valued)

■ Interval

- Measured on a scale of equal-sized units
- Values have order
- E.g., temperature in C̊or F̊, calendar dates
- No true zero-point

■ Ratio

- Inherent zero-point
- We can speak of values as being an order of magnitude larger than the unit of measurement (10 K̊is twice as high as 5 K̊).
- e.g., temperature in Kelvin, length, counts, monetary quantities

■ Discrete Attribute

- Has only a finite or countably infinite set of values

- E.g., zip codes, profession, or the set of words in a collection of documents
- Sometimes, represented as integer variables
- Note: Binary attributes are a special case of discrete attributes

■ Continuous Attribute

- Has real numbers as attribute values
- E.g., temperature, height, or weight
- Practically, real values can only be measured and represented using a finite number of digits Continuous attributes are ty
- practically represented as floating-point variables

Module 3

15. Explain how navies bay classification predicaions and discuss the naïve assumption in naïve bays

Naive Bayes is a probabilistic classification algorithm based on Bayes' Theorem. It is called "naive" because it assumes that the features in the data are independent, which is often not true in real-world scenarios.

How Naive Bayes Makes Predictions

1. Bayes' Theorem: The algorithm uses Bayes' theorem to calculate the probability of a class (CCC) given the features (XXX) of a data point

$$P(C|X) = \frac{P(X|C) \cdot P(C)}{P(X)}$$

- $P(C|X)$: Probability of the class given the features (posterior probability).
- $P(X|C)$: Probability of the features given the class (likelihood).
- $P(C)$: Prior probability of the class.
- $P(X)$: Prior probability of the features.

1. Prediction: For a new instance, the algorithm calculates P(C|X)P(C|X)P(C|X) for each class and assigns the class with the highest posterior probability. The "Naive" Assumption

• The algorithm assumes that all features are independent of each other.

• In reality, this assumption is rarely true, as features often exhibit some level of correlation.

• Despite this, Naive Bayes often performs well in practice, especially for high-dimensional datasets.

Advantages of Naive Bayes:

1. Simple and easy to implement.
2. Efficient for large datasets.
3. Works well for text classification problems

Limitations:

1. Relies on the unrealistic independence assumption.
2. Struggles with correlated features

---

2. Describe in detail about how to evaluate accuracy of the classifier.

■ Holdout method

- Given data is randomly partitioned into two independent sets
- Training set (e.g., 2/3) for model construction
- Test set (e.g., 1/3) for accuracy estimation
- Random sampling: a variation of holdout
- Repeat holdout k times, accuracy = avg. of the accuracies obtained

■ Cross-validation (k-fold, where k = 10 is most popular)

- Randomly partition the data into k mutually exclusive subsets, each approximately equal size
- At i-th iteration, use Di as test set and others as training set
- Leave-one-out: k folds where k = # of tuples, for small sized data

- *Stratified cross-validation*: folds are stratified so that class dist. in each fold is approx. the same as that in the initial data

■ Bootstrap
- Works well with small data sets
- Samples the given training tuples uniformly with replacement
- i.e., each time a tuple is selected, it is equally likely to be selected again and re-added to the training set
- Several bootstrap methods, and a common one is .632 boostrap
- A data set with d tuples is sampled d times, with replacement, resulting in a training set of d samples. The data tuples that did not make it into the training set end up forming the test set. About 63.2% of the original data end up in the bootstrap, and the remaining 36.8% form the test set (since (1 – 1/d)d ≈ e-1 = 0.368)
- Repeat the sampling procedure k times, overall accuracy of the model

$$Acc(M) = \frac{1}{k}\sum_{i=1}^{k}(0.632 \times Acc(M_i)_{test\_set} + 0.368 \times Acc(M_i)_{train\_set})$$

---

3. Discuss metric for evaluating classifier performance

■ Classifier Accuracy, or recognition rate: percentage of test set tuples that are correctly classified Accuracy = (TP + TN)/All

■ Error rate: 1 – accuracy, or Error rate = (FP + FN)/All

■ Class Imbalance Problem:
- One class may be rare, e.g. fraud, or HIV-positive

- Significant majority of the negative class and minority of the positive class

■ Sensitivity:

- True Positive recognition rate
- Sensitivity = TP/P

■ Specificity:

- True Negative recognition rate
- Specificity = TN/N

■ Precision: exactness – what % of tuples that the classifier labeled as positive are actually positive

$$precision = \frac{TP}{TP + FP}$$

Recall: completeness – what % of positive tuples did the classifier label as positive?

- Perfect score is 1.0
- Inverse relationship between precision & recall
- $$recall = \frac{TP}{TP + FN}$$

F measure (F1 or F-score): harmonic mean of precision and recall,

$$F = \frac{2 \times precision \times recall}{precision + recall}$$

Fß : weighted measure of precision and recall ■ assigns ß times as much weight to recall as to precision

$$F_\beta = \frac{(1 + \beta^2) \times precision \times recall}{\beta^2 \times precision + recall}$$

Module 5

4. Explain concepts of market basket with example

Market Basket Analysis (MBA) is a technique used in data mining to analyze and uncover relationships between products purchased together by customers. It involves identifying patterns in transactional data to understand which items are frequently bought together. The goal is to find associations between different products, which can be used to optimize product placement, promotions, and cross-selling strategies.

How Market Basket Analysis Works:

Data Collection: Collect transactional data, such as shopping carts or sales receipts

• Pattern Discovery: Identify associations between products using algorithms like Apriori or FP-Growth.

Association Rule Generation: Generate rules that show how frequently certain items are bought together, such as "If a customer buys Product A, they are likely to buy Product B."

Example of Market Basket Analysis:

Consider a retail store with the following transactional data:

• Transaction 1: {Milk, Bread, Butter}

• Transaction 2: {Milk, Bread}

• Transaction 3: {Milk, Butter}

• Transaction 4: {Bread, Butter}

• After performing market basket analysis, we may find that there is a high association between Milk and Bread, and Butter is often bought with Milk and Bread.

Generated Rules: 1. Rule 1: {Milk} → {Bread} (If a customer buys Milk, they are likely to buy Bread)

2. Rule 2: {Milk, Bread} → {Butter} (If a customer buys Milk and Bread, they are likely to buy Butter)

3. **Explain Multilevel Association Rules Mining and Multidimensional Association Rules Mining with examples.**

• Multidimensional Association Rule Mining is the process of discovering association rules from data that is organized across multiple dimensions or attributes. The data is often represented in a multidimensional database, such as OLAP (Online Analytical Processing) cubes, where the data is structured across several attributes or dimensions.

• Steps:

1. Data Representation at Multiple Levels: Data is represented at different granularities (e.g., product category vs. specific product).
2. Mining Rules: Association rules are mined at each level. More general rules are formed at higher levels, while more specific rules are mined at lower levels.

• Example:

o Level 1: "If a customer buys Red T-shirt, they are likely to buy Blue Jeans."

o Level 2: "If a customer buys Clothing, they are likely to buy Men's Wear."

Multidimensional Association Rules Mining

Multidimensional Association Rule Mining uncovers patterns across multiple dimensions or attributes, such as product, time, and location.

• Steps:

o Data Representation: Data is organized across multiple dimensions (e.g., product, time, and location).

o Mining Rules: Patterns are discovered by considering combinations of these dimensions.

• Example:

o "If a customer buys Smartphone in Winter at Store A, they are likely to buy a Phone Cover."

Module 6

Explain Page Rank algorithm with example.

• The PageRank algorithm is a method used by search engines like Google to rank web pages based on their importance. It evaluates the importance of a page by analyzing the links between pages.

Key Concepts:

1. Link Importance:

o A page is considered important if many other important pages link to it.

o Links from highly-ranked pages carry more weight than links from low-ranked pages.

2. Damping Factor:

o Users do not always follow links; they may jump to a random page. This is modeled by a damping factor (typically 0.85), which accounts for random jumps.

3. Iteration:

o The PageRank values are calculated repeatedly until they stabilize, ensuring accurate rankings.

Example:

• Consider 3 pages: A, B, and C:

• Page A links to B and C.

• Page B links to C.

• Page C links to A.

• Process:

1. Initially, all pages are given equal importance (e.g., 1 point each).
2. Pages distribute their rank to the pages they link to.
    - If A links to B and C, it divides its rank equally between them.
3. Over several iterations, ranks are recalculated based on the ranks received from other pages.

What is web mining and its types (web content mining structure mining and web usage mining) in detail with applications

Web Mining is the process of Data Mining techniques to automatically discover and extract information from Web documents and services. The main purpose of web mining is to discover useful information from the World Wide Web and its usage patterns.

Web Structure Mining :Web structure mining is the application of discovering structure information from the web. The structure of the web graph consists of web pages as nodes, and hyperlinks as edges connecting related pages. Structure mining basically shows the structured summary of a particular website. It identifies relationship between web pages linked by information or direct link connection. To determine the connection between two commercial websites, Web structure mining can be very useful.

- Approaches Used in Web Structure Mining:
- 1. Link Analysis: This approach examines the hyperlinks between web pages. By analyzing how pages are linked to each other, it helps determine the relevance and importance of web pages. Popular algorithms include:
- o PageRank: Measures the importance of a page based on the number and quality of incoming links.
- o HITS (Hyperlink-Induced Topic Search): Identifies authoritative and hub pages by considering both inbound and outbound links.

2.Web Graph Construction: Construct a graph representing web pages as nodes and hyperlinks as edges. By analyzing the structure of this graph, crawlers and search engines can prioritize the most relevant pages and follow effective paths for crawling

3. Clustering of Web Pages: Group similar web pages based on their link structure. This helps search engines categorize pages and improve the efficiency of retrieving related pages.

4. Anchor Text Mining: Analyze the text surrounding the hyperlinks (anchor text) to understand the context of linked pages. This provides additional semantic information about the pages and helps improve the accuracy of search engine results.

5. Social Network Analysis: Apply network analysis techniques to identify communities and trends in web structure. This can help web crawlers and search engines better understand the structure and relevance of content based on community links.

- Information retrieval in social networks.
- To find out the relevance of each web page.
- Measuring the completeness of Websites.
- Used in Search engines to find the relevant information.

Web usage mining

Web Usage Mining Web Usage Mining is the process of extracting useful information from the web logs (e.g., server logs, clickstream data) to understand the behavior of users on a website. It analyzes how users interact with web pages, including their navigation patterns, clicks, search queries, and browsing sequences. The primary goal of web usage mining is to uncover patterns in user behavior that can help improve website design, enhance user experience, and personalize content.

Process of Web Usage Mining:

1. Data Collection: Data is collected from web logs, including information like pages visited, time spent on each page, and user clicks.
2. Preprocessing: The raw web logs are cleaned and filtered to remove irrelevant information, such as bots and spam data
3. Pattern Discovery: Techniques such as clustering, association rule mining, and sequential pattern mining are used to discover patterns in user behavior.
4. Pattern Analysis: The discovered patterns are analyzed to derive useful insights for website improvement.

Applications of Web Usage Mining:

1. Personalized Recommendations: Web usage mining helps websites recommend personalized content (e.g., products, articles, or videos) based on user behavior. By analyzing users' past interactions, the website can suggest items that are likely to interest the user.
2. Website Optimization: By understanding user navigation patterns, web usage mining helps improve website design, streamline user interfaces, and optimize content placement. For instance, if users frequently abandon a particular page,

1. Explain CLARANS extension in web mining

CLARANS (Clustering Large Applications based on RANdomized Search) is an extension of the K-means clustering algorithm used in web mining. It is specifically designed to handle large datasets and is more efficient than traditional K-means.

CLARANS works by randomly sampling the dataset and searching for the best clustering solution within the sample. This randomization process helps to avoid getting stuck in local optima, which can be a problem in traditional K-means.

Here's how CLARANS works:

1. **Initialization**: CLARANS randomly selects a number of data points as initial medoids (representative points) for each cluster.
2. **Search**: CLARANS performs a randomized search to find the best medoids. It randomly selects a data point from the current medoids and evaluates the cost of swapping it with another non-medoid point. If the cost is lower, the swap is accepted and the new point becomes a medoid. This process is repeated until no further improvements can be made.
3. **Sampling**: CLARANS repeats the search process multiple times with different random samples of the dataset. Each sample is called a local search. The best clustering solution found across all local searches is selected as the final result.

The CLARANS extension in web mining is particularly useful for clustering large web datasets. It can help identify groups of similar web pages, detect anomalies or outliers, and improve search engine performance by organizing web pages into meaningful clusters.

In summary, CLARANS is an extension of the K-means algorithm that uses randomized search and sampling techniques to efficiently cluster large web datasets. It helps overcome the limitations of traditional K-means and is widely used in web mining applications.