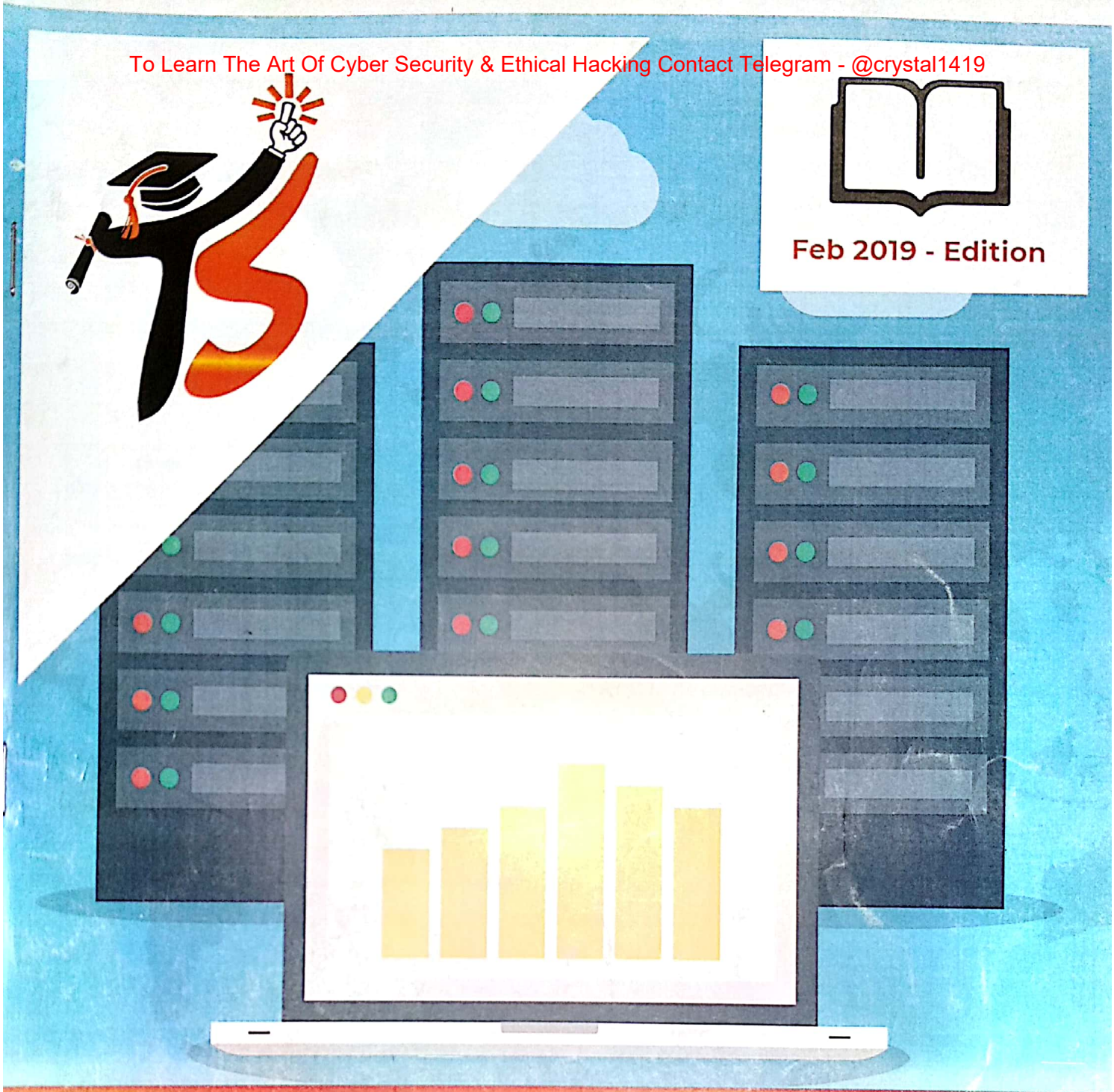
Feb 2019 - Edition



DATA WAREHOUSING & MINING

(BE - COMPUTER)

8 SEM

(As per Revised Syllabus w.e.f 2015-2016)

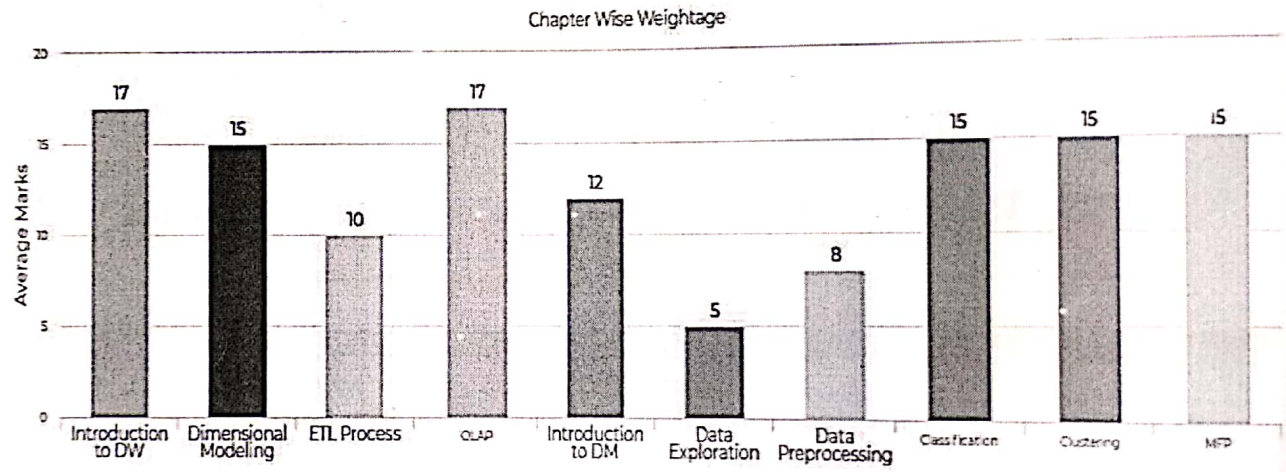
Syllabus:

Exam	TT-1	TT-2	AVG	Term Work	Oral/Practical	End of Exam	Total
Marks	20	20	20	25	25	80	150

#	Module	Details Contents	Page No.
1	Introduction to Data Warehousing	The Need for Data Warehousing; Increasing Demand for Strategic Information; Inability of Past Decision Support System; Operational V/s Decisional Support System; Data Warehouse Defined; Benefits of Data Warehousing; Features of a Data Warehouse; The Information Flow Mechanism; Role of Metadata; Classification of Metadata; Data Warehouse Architecture; Different Types of Architecture; Data Warehouse and Data Marts; Data Warehousing Design Strategies.	01
2	Dimensional Modeling	Data Warehouse Modeling Vs Operational Database Modeling; Dimensional Model Vs ER Model; Features of a Good Dimensional Model; The Star Schema; How Does a Query Execute? The Snowflake Schema; Fact Tables and Dimension Tables; The Factless Fact Table; Updates To Dimension Tables: Slowly Changing Dimensions, Type 1 Changes, Type 2 Changes, Type 3 Changes, Large Dimension Tables, Rapidly Changing or Large Slowly Changing Dimensions, Junk Dimensions, Keys in the Data Warehouse Schema, Primary Keys, Surrogate Keys & Foreign Keys; Aggregate Tables; Fact Constellation Schema or Families of Star.	08
3	ETL Process	Challenges in ETL Functions; Data Extraction; Identification of Data Sources; Extracting Data: Immediate Data Extraction, Deferred Data Extraction; Data Transformation: Tasks Involved in Data Transformation, Data Loading: Techniques of Data Loading, Loading the Fact Tables and Dimension Tables Data Quality; Issues in Data Cleansing.	20
4	Online Analytical Processing (OLAP)	Need for Online Analytical Processing; OLTP V/s OLAP; OLAP and Multidimensional Analysis; Hypercube; OLAP Operations in Multidimensional Data Model; OLAP Models: MOLAP, ROLAP, HOLAP, DOLAP.	24
5	Introduction to data mining	What is Data Mining; Knowledge Discovery in Database (KDD), What can be Data to be Mined, Related Concept to Data Mining, Data Mining Technique, Application and Issues in Data Mining.	38
6	Data Exploration	Types of Attributes; Statistical Description of Data; Data Visualization; Measuring similarity and dissimilarity.	44
7	Data Preprocessing	Why Preprocessing? Data Cleaning; Data Integration; Data Reduction: Attribute subset selection, 8Histograms, Clustering and Sampling; Data Transformation & Data Discretization: Normalization, Binning, Histogram Analysis and Concept hierarchy generation.	47
8	Classification	Basic Concepts; Classification methods: Decision Tree Induction: Attribute Selection Measures, Tree pruning. Bayesian Classification: Naïve Bayes' Classifier.	50

		Prediction: Structure of regression models; Simple linear regression, Multiple linear regression. Model Evaluation & Selection: Accuracy and Error measures, Holdout, Random Sampling, Cross Validation, Bootstrap; Comparing Classifier performance using ROC Curves. Combining Classifiers: Bagging, Boosting, Random Forests.	
9	Clustering	What is clustering? Types of data, Partitioning Methods (K-Means, KMedoids)Hierarchical Methods(Agglomerative , Divisive, BRICH), Density Based Methods (DBSCAN, OPTICS)	59
10	Mining Frequent Pattern and Association Rule	Market Basket Analysis, Frequent Itemsets, Closed Itemsets, and Association Rules; Frequent Pattern Mining, Efficient and Scalable Frequent Itemset Mining Methods....	71

Chapter wise Weightage:



Marks Distribution:

#	MAY - 16	DEC - 16	MAY - 17	DEC - 17	MAY – 18	DEC - 18
1.	15	15	15	20	20	15
2.	15	15	20	15	10	15
3.	10	15	10	05	10	10
4.	20	20	10	20	10	20
5.	10	15	05	10	15	15
6.	-	-	10	05	05	10
7.	10	-	10	10	10	05
8.	15	25	10	10	20	10
9.	10	10	20	15	15	10
10.	20	10	15	15	15	15
Repeated Marks	**-**	**05**	**35**	**60**	**80**	**90**

CHAP - 1: INTRODUCTION TO DATA WAREHOUSING

Q1. Differentiate Data Warehouse Vs Data Mart.

Ans: **[5M – Dec16 & May17]**

DIFFERENTIATE BETWEEN DATA WAREHOUSE AND DATA MART:

Table 1.1 shows the difference between Data Warehouse and Data Mart.

Table 1.1

Parameters	Data Warehouse	Data Mart
Scope	Enterprise Level.	Department Level.
Approach	Top – Down Approach is used.	Bottom – Up Approach is used.
Centralized & Planned	Yes	No
Size	100 GB to 1 TB.	< 100 GB.
Initial effort, cost, Risk	Higher.	Lower.
Data Sources Used	Many Data Sources are required.	Few Data Sources are required.
Nature	Highly Flexible.	It is restrictive.
Implementation Time Required	Implementation takes Months to Year.	Implementation is done usually in months.
Subjects	Multiple Subjects.	Single Subject.
Data Available	Data is historical, detailed and summarized.	Data consists of some history, detailed and summarized.

Q2. Operational Vs. Decisional Support System.

Ans: **[5M | May16]**

COMPARISON BETWEEN OPERATIONAL SYSTEM AND DECISIONAL SUPPORT SYSTEM:

Table 1.2 shows the difference between Operational System and Decisional Support System.

Table 1.2

Operational System	Decisional Support System
It is Application Oriented.	It is Subject Oriented.
It uses Detailed Data.	It uses Summarized Data.
It contains isolated data.	It contains integrated data.
It is used to run business.	It is used to analyze business.
It is performance sensitive.	It is not performance sensitive.
There is no data redundancy.	There is data redundancy.
It has repetitive access.	It has Adhoc access.
It has up to date data.	It has snapshot of data.
Database size is 100 MB – 100 GB.	Database size is 100 GB – few TB.
Only few records can be accessed at a time.	Large volume of data can be accessed at a time.

Q3. Architecture of a typical DW system.

Q4. Illustrate the architecture of a typical DW system.

Q5. Explain Data Warehouse Architecture in detail

Ans: **[5 – 10M | May16, May17, Dec16]**

DATA WAREHOUSE:

1. A data warehouse is a **Relational Database**.
2. It was defined by **Bill Inmon** in 1990.
3. It is a (usually huge) collection of data.
4. It is used primarily in decision making processes.
5. It is integrated, subject-oriented, time-variant and non-volatile collection of data.
6. It is designed for query and analysis rather than for transaction processing.
7. It is constructed by **integrating data** from multiple heterogeneous sources.
8. Data Warehouse is a system used for **reporting** and **data analysis**.
9. It is considered as a core component of **business intelligence**.

ARCHITECTURE OF TYPICAL DATA WAREHOUSE:

Figure 1.1 shows Typical Data Warehouse Architecture.

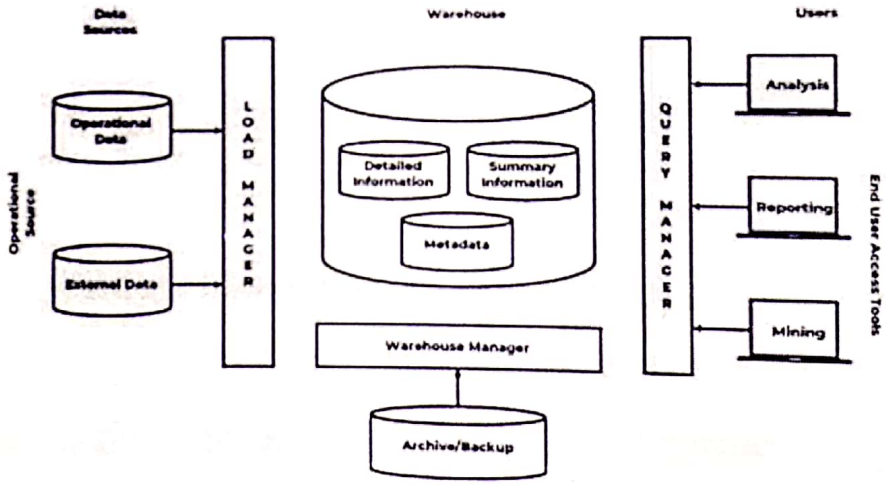


Figure 1.1: Typical Data Warehouse Architecture.

Data Warehouse Architecture consist of following components:

Operational Source:

1. Operational Source is a data source which consists of **Operational Data** and **External Data**.
2. Data can come from Relational DBMS like **Oracle, Informix**.

Load Manager:

1. Load Manager performs all the operations required to **Extract and Load Data**.
2. The size and complexity of the load manager varies between specific solutions from one data warehouse to other.

Warehouse Manager:

1. Warehouse Manager is responsible for the **warehouse management process**.
2. Operations performed by warehouse manager includes:
   a. Analysis of Data.
   b. Transformation and Merging.
   c. Generation of Aggregation.
   d. Backing up and archiving of data.
   e. De-normalization.

Query Manager:

1. Query Manager performs all the operations associated with management of user queries.
2. The complexity of a query manager is determined by facilities provided by the end users access tools and database.

Detailed Data:

1. It is used to store all the **detailed data in the database schema**.
2. Detailed Data is loaded into the data warehouse to supplement the aggregated data.

Summarized Data:

1. Summarized Data is a part of data warehouse that **stores predefined aggregations**.
2. These aggregations are generated by the warehouse manager.

Archive and Backup Data:

1. The Detailed and Summarized Data are stored for the purpose of archiving and backup.
2. The data is transferred to storage archives such as magnetic tapes or optical disks.

Metadata:

1. Metadata is basically **Data stored above Data**.
2. It is used for extraction and loading process, warehouse management process and query management process.

End User Access Tools:

1. End User Access Tools consists of **Analysis, Reporting and Mining**.
2. The users interacts with warehouse using end user access tools.

Q6. Differentiate top-down and bottom-up approaches for building data warehouse. Discuss the merits and limitations of each approach.

Q7. Data warehouse design strategies

Q8. Discuss Data Warehouse design strategies in detail?

Ans: **[5 – 10M | Dec17, May18 & Dec18]**

DATA WAREHOUSE:

1. A data warehouse is a **Relational Database**.
2. It was defined by **Bill Inmon** in 1990.

3. It is designed for query and analysis rather than for transaction processing.
4. It is integrated, subject-oriented, time-variant and non-volatile collection of data.

TOP DOWN APPROACH:

1. Figure 1.2 shows the Top Down Approach for Data Warehouse.
2. In this approach, the data flow begins with **data extraction** from the operational data sources.
3. This data is then loaded into **staging area**.
4. It is then transferred to **Operational Data Store (ODS)**.
5. Sometimes the ODS step is skip, if it is replication of the operational databases.
6. Data is also loaded into data warehouse in a parallel process to avoid extracting it from the ODS.
7. Then the data mart is loaded with the data.
8. And finally OLAP environment is available to the users.

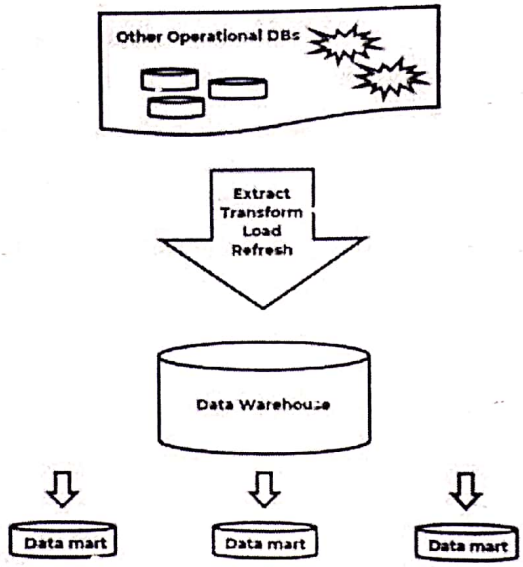


Figure 1.2: Top Down Approach.

ADVANTAGES:

1. The data is centralized.
2. Results can be obtained quickly.

DISADVANTAGES:

1. Time consuming process.
2. Failure risk is very high.

BOTTOM UP APPROACH:

1. Figure 1.3 shows the Bottom Up Approach for Data Warehouse.
2. The position of data warehouse and data mart are reversed in bottom up approach.
3. The data flow begins with **extraction of data** from operational databases into the staging area.
4. Data is then loaded into **Operational Data Store (ODS)**.

5. The data in ODS is appended to or replaced by the fresh data being loaded.
6. Once the ODS is refreshed the current data is once again extracted into the staging area.
7. It is then processed to fit into the data mart structure.
8. The data from the data mart is then extracted to the staging area aggregated, summarized and so on.
9. It is then loaded into the data warehouse.
10. Finally it is made available to the end user for analysis.

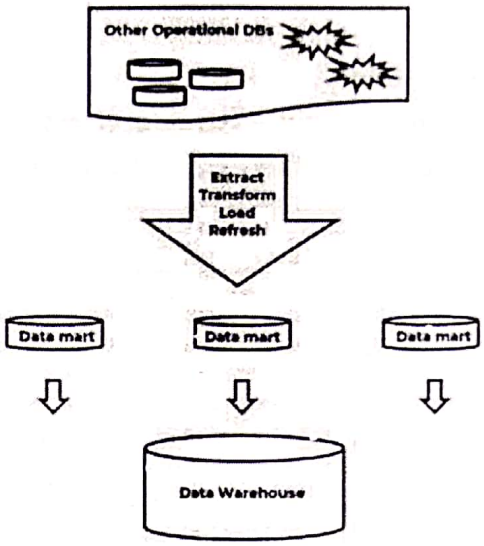


Figure 1.3: Bottom Up Approach.

ADVANTAGES:

1. Data Marts can be delivered more quickly.
2. Risk of failure is low.

DISADVANTAGES:

1. Redundancy of data in data mart.
2. It preserve inconsistent and incompatible data.

Q9. What is meant by metadata in the context of a Data warehouse? Explain the different types of Meta data stored in a data warehouse. Illustrate with a suitable example.

Q10. Metadata in Data Warehouse.

Q11. Define Metadata. Discuss the types of Metadata stored in a data warehouse. Illustrate with an example.

Q12. Role of metadata

Q13. Meta data with example

Ans: **[5 – 10M | Dec16, May17, Dec17, May18 & Dec18]**

METADATA:

1. Metadata is simply defined as **Data about Data**.
2. The data that is used to represent other data is known as **metadata**.

3. For **example**, the index of a book serves as a metadata for the contents in the book.
4. In other words, we can say that metadata is the **summarized data** that leads us to detailed data.
5. Metadata is the control panel to the data warehouse.
6. It is data that describes the data warehousing and business intelligence system:

DATA WAREHOUSE METADATA:

1. In terms of data warehouse, we can define metadata as follows:
   a. Meta Data is the **road-map** to a data warehouse.
   b. Meta Data in a data warehouse defines the **warehouse objects**.
   c. Metadata acts as a **directory**. This directory helps the decision support system to locate the contents of a data warehouse.
2. Data warehousing has specific metadata requirements.
3. Metadata that describes tables typically includes:
   a. Physical Name.
   b. Logical Name.
   c. Type: Fact, Dimension, Bridge.
   d. Role: Legacy, OLTP, Stage.
   e. DBMS: DB2, Informix, MS SQL Server, Oracle, Sybase.
   f. Location.
   g. Definition.
   h. Notes
4. Metadata describes columns within tables:
   a. Physical Name.
   b. Logical Name.
   c. Order in Table.
   d. Datatype & Length.
   e. Decimal Positions.
   f. Nullable/Required & Default Value.
   g. Edit Rules.
   h. Definition.
   i. Notes

ROLES OF METADATA:

1. The following figure 1.4 shows the roles of metadata.
2. Roles of metadata includes:
   a. It is used for query tools.
   b. It is used in extraction and cleansing tools.
   c. It is used in reporting tools.
   d. It is used in transformation tools.
   e. It plays an important role in loading functions.

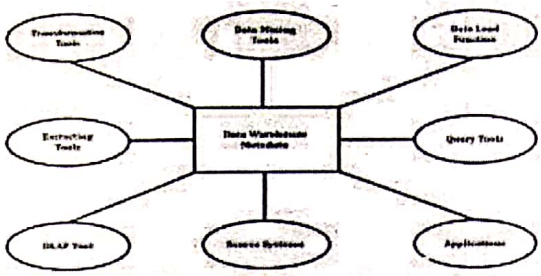


Figure 1.4: Roles of Metadata.

TYPES OF METADATA:

Metadata in a data warehouse fall into three major categories as shown in figure 1.5.

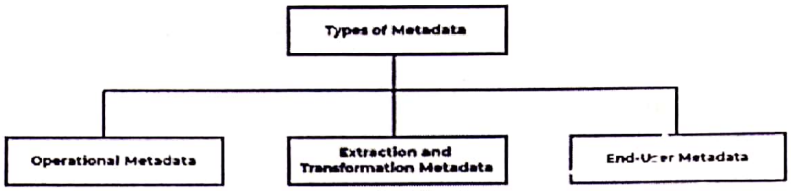


Figure 1.5: Types of Metadata.

I) Operational Metadata:

1. In Data Warehouse, Data comes from **several operational systems** of the enterprise.
2. Different source systems contain **different data structures**.
3. The data elements selected for the data warehouse have various field lengths and data types.
4. So the information of operational data source is given by **Operational Metadata**.

II) Extraction and Transformation Metadata:

1. Extraction and transformation metadata contains the information about **extraction of data** from heterogeneous source system.
2. It also contains the information about **data transformation in data staging area**.

III) End-User Metadata:

1. The end-user metadata is the **navigational map** of the data warehouse.
2. It enables the end-users to find information from the data warehouse.
3. The end-user metadata allows the end-users to use their own business terminology and look for information in those ways in which they normally think of the business.

EXAMPLE OF METADATA: (Customer Sales Data Warehouse)

Entity Name: Customer.
Alias Name: Account, Client.
Definitions: A Person that purchases the product.
Source Systems: Online Sales.
Responsible User: Angel Priya.
Data Quality Reviewed: 07-Dec-2018.

CHAP – 2 | DIMENSIONAL MODELING

Q1. Factless Fact Table **[5M | May16]**

Ans:

FACT TABLE:

1. Fact Table is a collection of **facts and measures.**
2. It is located at the center of a **star schema** or a **snowflake schema** surrounded by dimension tables.

FACTLESS FACT TABLE:

1. A Factless fact table is fact table that does not contain fact i.e. measures.
2. They contain only **dimensional keys**.
3. Factless Fact Table captures events that happen only at **information level**.
4. It is used to capture the **many-to-many relationships** between dimensions.
5. Factless fact tables are used for **tracking a process or collecting stats**.

TYPES OF FACTLESS FACT TABLE:

As shown in figure 2.1, there are two types of factless fact tables: those that describe events, and those that describe conditions.

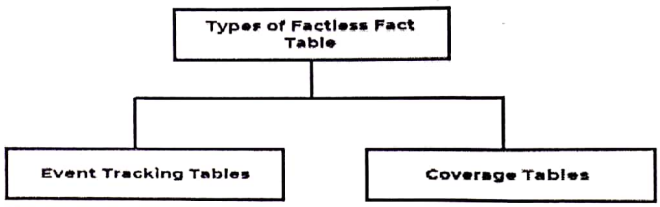


Figure 2.1: Types of Factless Fact Table.

Event Tracking Tables:

1. Event Tracking Tables is used to track the event of interest.
2. Many event-tracking tables in dimensional data warehouses turn out to be Factless.

Coverage Tables:

1. Coverage Tables was defined by Ralph.
2. It is used to support negative analysis report.

EXAMPLE OF FACTLESS FACT TABLE:

1. Tracking student attendance.
2. List of people for the web click.

Q2. Fact Constellation

Ans:

[5M | Dec17]

FACT CONSTELLATION:

1. Fact constellation is a measure of **online analytical processing**.
2. It is a collection of multiple fact tables sharing dimension tables.
3. It is viewed as a **collection of stars**.

4. This is an improvement over Star schema.
5. A fact constellation has **multiple fact tables**.
6. It is also known as **Galaxy Schema**.
7. This schema is more complex than star or snowflake schema.
8. For each star schema or snowflake schema it is possible to construct a fact constellation schema.
9. The main disadvantage of the fact constellation schema is a more complicated design.
10. It is hard to manage and support fact constellation schema.
11. The schema of this type should only be used for applications that need a high level of sophistication.
12. Figure 2.2 shows the representation of fact constellation.

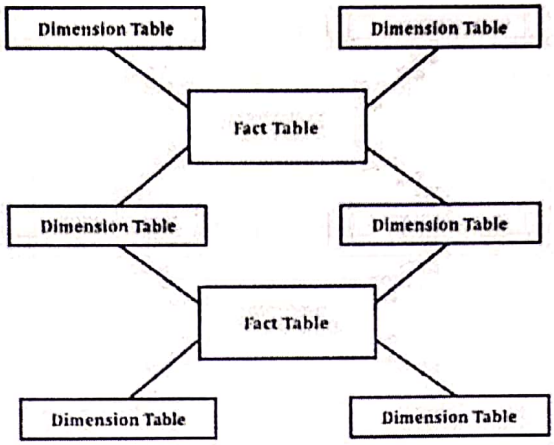


Figure 2.2: Fact Constellation.

Q3. Updates to Dimension tables.

Q4. Explain Updates to dimension tables in detail

Ans: **[5 - 10M | May16, Dec17 & Dec18]**

DIMENSION TABLE:

1. A dimension table is a table in a **star schema** of a data warehouse.
2. A dimension table stores attributes, or dimensions, that describe the objects in a fact table.

UPDATES TO DIMENSIONS TABLES:

1. Over the time, every day as more and more sales take place, more and more rows get added to the fact table.
2. Updations due to change in fact table happens very rarely.
3. Compared to the fact table, the dimension tables are more **stable** and **less volatile**.
4. Dimension table changes due to **change in attributes** themselves but not because of increase in number of rows.
5. Types of changes that affect dimension tables are as follows:

I) Slowly Changing Dimensions:

1. Dimensions are generally constant over time, but if not constant then it may change slowly.
2. **Example:** Customer ID of the record remain same but the marital status or location of customer may change over time.

3. There are three different types:
   a. **Type 1 Change:** It is related to **correction of errors** in source systems and changes are not preserved.
   b. **Type 2 Change:** It is related to the **true changes** in source systems and changes are preserved.
   c. **Type 3 Change:** It is related to **tentative changes** in the source systems and changes are preserved.

II) Large Dimension Tables:

1. Large Dimensions tables are **very deep and wide**.
2. Deep means it has large numbers of rows.
3. Wide means it may have many attributes or columns.
4. To handle large dimensions table, we can divide large dimension into some mini dimensions based on the interest.

III) Rapidly Changing Dimensions:

1. If the dimension table changes rapidly then break the dimension table into one or more smaller dimension tables.
2. Move the rapidly changing attributes in another dimension table and leave the original dimension table with slowly changing attributes.

IV) Junk Dimensions:

1. Some textual data or flags cannot be the significant fields in major dimensions of source legacy systems.
2. Although it cannot be discarded.
3. So create a single **Junk dimension** and keep all meaningful text and flags into it.
4. Such junk dimensions are useful to fire the quires based on flags or text values.

Q5. For a Super market chain, consider the following dimensions namely product, store, time & promotion. The schema contains a central fact table for sales.

i. Design star schema for the above application.

ii. Calculate the maximum number of base fact tables records for warehouse with the following values given below:

- **Time period – 5 Years.**
- **Store – 300 stores reporting daily sales.**
- **Product – 40,000 products in each store (about 4000 sell in each store daily)**

[10M | May16]

Ans:

STAR SCHEMA:

1. Star Schema is the most popular schema design for a Data Warehouse.
2. In data warehousing and business intelligence, a star schema is the simplest form of a dimensional model
3. It is called a star schema because the diagram resembles a star, with points radiating from a center.
4. The center of the star consists of **fact table** and the points of the star are the **dimension tables**.

5. Usually the fact tables in a star schema are in **third normal form (3NF)** whereas dimensional tables are **de-normalized**.
6. Each dimension in a star schema is represented with only one-dimension table.
7. This dimension table contains the set of attributes.
8. For example, the location dimension table may contains the attribute set like location_key, street, city, province_or_state, country.

STAR SCHEMA FOR SUPER MARKET CHAIN:

Figure 2.3 shows the Star Schema for Super Market Chain.

Fact Table: Sales.

Dimension Table: Product, Store, Time and promotion.

There are 40,000 Products, 300 Stores and Time period of 5 Years.

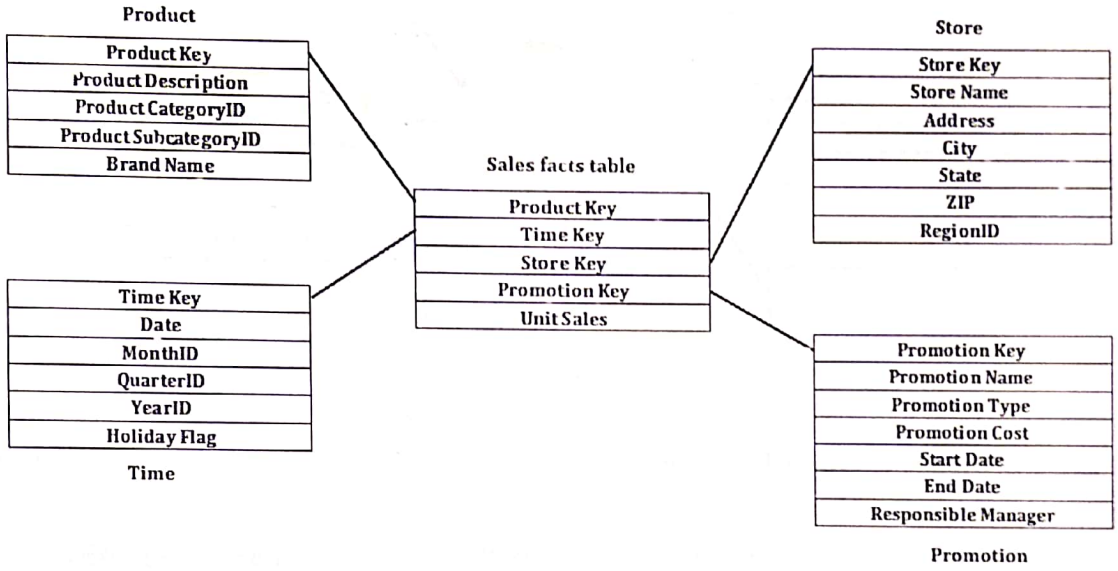


Figure 2.3: Sales Promotion Star Schema.

Maximum No. of fact table records:

Time Period = 5 Years x 365 Days
= 1825

No. of Stores = 300

Each Stores daily sale = 4000

Promotion = 1

Maximum No. of Fact Table Records = Time Period x No. of Stores x Daily Sale x Promotion.
= 1825 x 300 x 4000 x 1
= 2,190,000,000

Maximum No. of Fact Table Records = 2 Billion.

Q6. What is dimensional modelling? Design the data warehouse for wholesale furniture Company. The data warehouse has to allow analyzing the company's situation at least with respect to the Furniture, Customer and Time. More ever; the company needs to analyze: The furniture with respect to its type, category and material. The customers with respect to their spatial location, by considering at least cities, regions and states. The company is interested in learning the quantity, income and discount of its sales.

[10M | May17]

Ans:

DIMENSIONAL MODELLING:

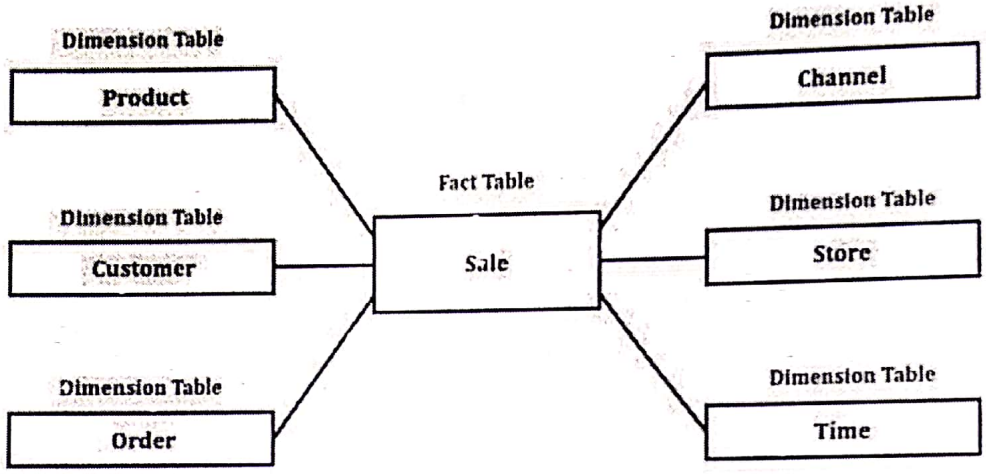


Figure 2.4: Dimensional Model.

1. Dimensional modeling is a set of techniques and concepts used in data warehouse design.
2. Dimensional modeling is a **design technique** for databases intended to support end-user queries in a data warehouse.
3. Dimensional modeling always uses the concepts of facts (measures), and dimensions (context).
4. Facts are typically (but not always) numeric values that can be aggregated.
5. Dimensions are groups of hierarchies and descriptors that define the facts.
6. For example, sales amount is a fact; timestamp, product, register#, store#, etc. are elements of dimensions.
7. Dimensional models are built by business process area, e.g. store sales, inventory, claims, etc.
8. This modeling technique provides high performance for queries and analysis.
9. Figure 2.4 represent Dimensional Model.

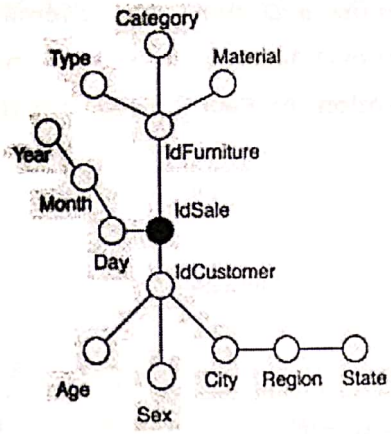
DATA WAREHOUSE DESIGN FOR A WHOLESALE FURNITURE COMPANY:

I) Identify facts, dimensions and measures:

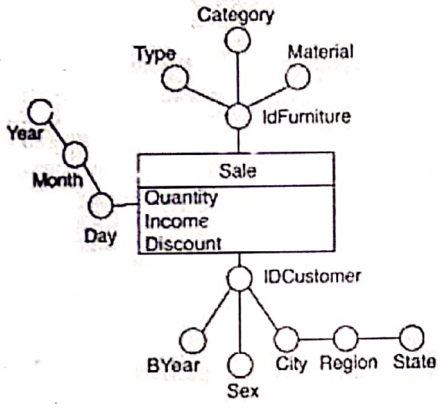
1. **Fact:** Sales.
2. **Measures:** Quantity, Income and Discount.
3. **Dimensions:**
   a. Furniture (Type, Category, Material)
   b. Customer (Age, Sex, City → Region → State)
   c. Time (Day → Month → Year)

II) For each fact produce the attribute tree and fact schema.

1. **Attribute Tree:**



2. **Fact Schema:**



III) Design the star or snowflake schema:

Figure 2.5 shows Star Schema for Furniture Data warehouse.

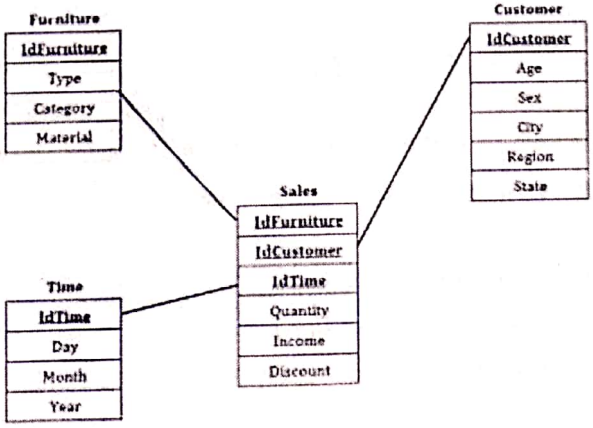


Figure 2.5: Data Warehouse for Furniture Company (Star Schema)

Q7. Consider following dimensions for a Hypermarket chain: Product, Store, Time and Promotion. With respect to this business scenario, answer the following questions. Clearly state any reasonable assumptions you make. Design a star schema. Whether the star schema can be converted to snowflake schema? Justify your answer and draw snowflake schema for the data warehouse (clearly mention the Fact table(s), Dimension table(s), their attributes and measures)

[10M | May16]

Ans:

STAR SCHEMA:

Refer Star Schema Section.

STAR SCHEMA FOR HYPER MARKET CHAIN:

Figure 2.6 shows the Star Schema for Hyper Market Chain.

Fact Table: Sales.

Dimension Table: Product, Store, Time and promotion.

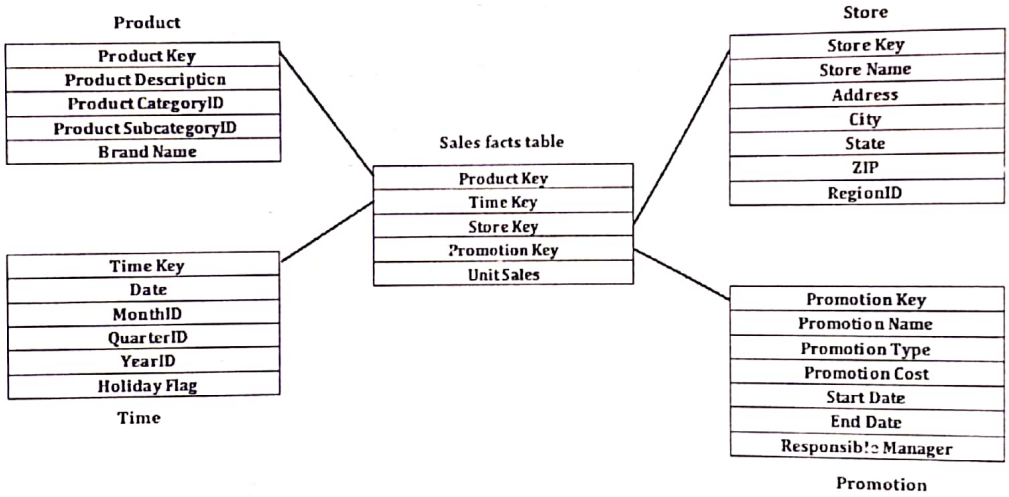


Figure 2.6: Sales Promotion Star Schema.

Whether the star schema can be converted to snowflake schema?

Yes the above star schema can be converted to snowflake schema by considering the following assumptions:

1. Product can be classified into category and subcategory.
2. Store belongs to a region, and a region dimension is not added in star schema.
3. Time Dimensions can be further divided into Month, Quarter and Year.
4. Promotion can be further classified into types.

SNOWFLAKE SCHEMA:

1. The snowflake schema is an **extension** of the star schema.
2. Snowflake schema consists of a fact table surrounded by multiple dimension tables which can be connected to other dimension tables via **many-to-one relationship**.

3. Snowflake schema is more **complex** than a star schema in term of the data model.
4. This schema resembles a snowflake, therefore, it is called **Snowflake Schema**.
5. A snowflake schema is designed from star schema by further normalizing dimension tables to eliminate data redundancy.
6. Therefore in the snowflake schema, instead of having big dimension tables connected to a fact table, we have a group of multiple dimension tables.
7. In the snowflake schema, dimension tables are normally in the third normal form (3NF).
8. The snowflake schema helps save storage however it increases the number of dimension tables.
9. Figure 2.7 shows the Snowflake Schema for Hyper Market Chain.

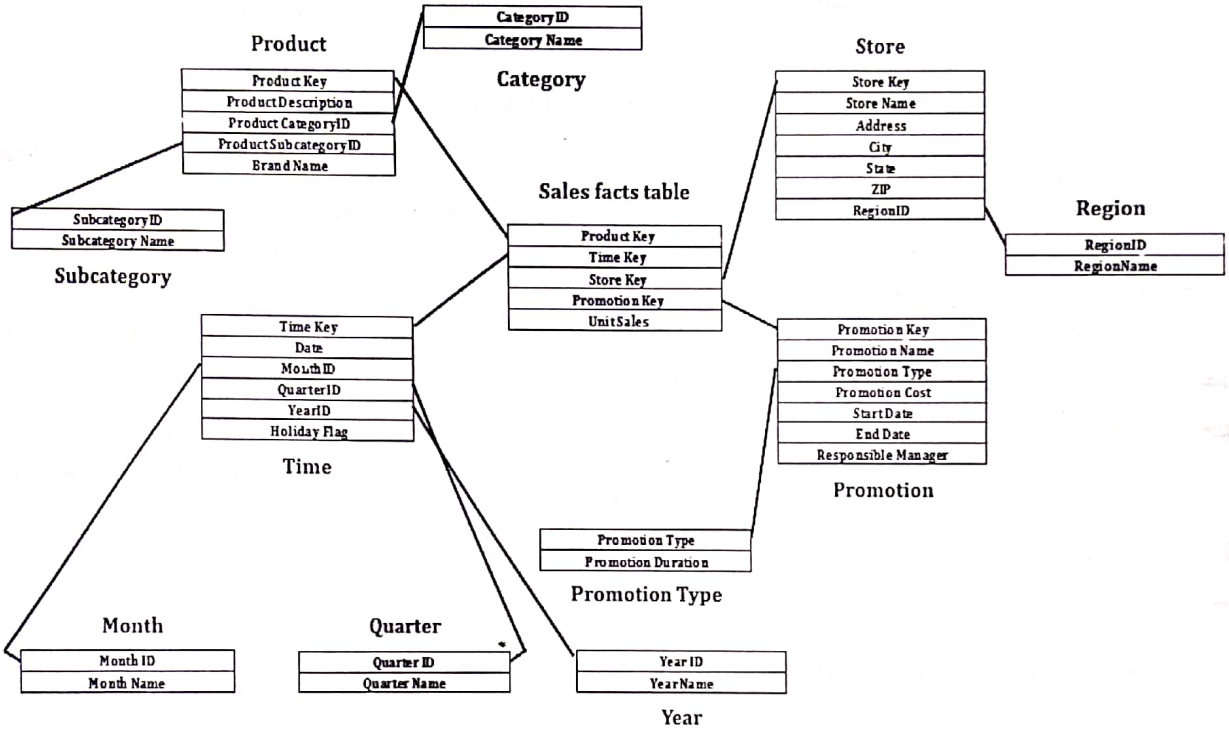


Figure 2.7: Sales Promotion Snowflake Schema.

Q8. A manufacturing company has a huge sales network. To control the sales, it is divided into regions. Each region has multiple zones. Each zone has different cities at different granularity levels of region and to count no. of products sold. Design a star schema by considering granularity levels for region, sales person and time. Convert the star schema to snow flake schema.

Ans: **[10M | Dec17]**

STAR SCHEMA:

Refer Star Schema Section.

STAR SCHEMA FOR MANUFACTURING COMPANY:

Figure 2.8 shows the Star Schema for manufacturing company.

Fact Table: Sales.

Dimension Table: Sales_Person, Location, Time and Product.

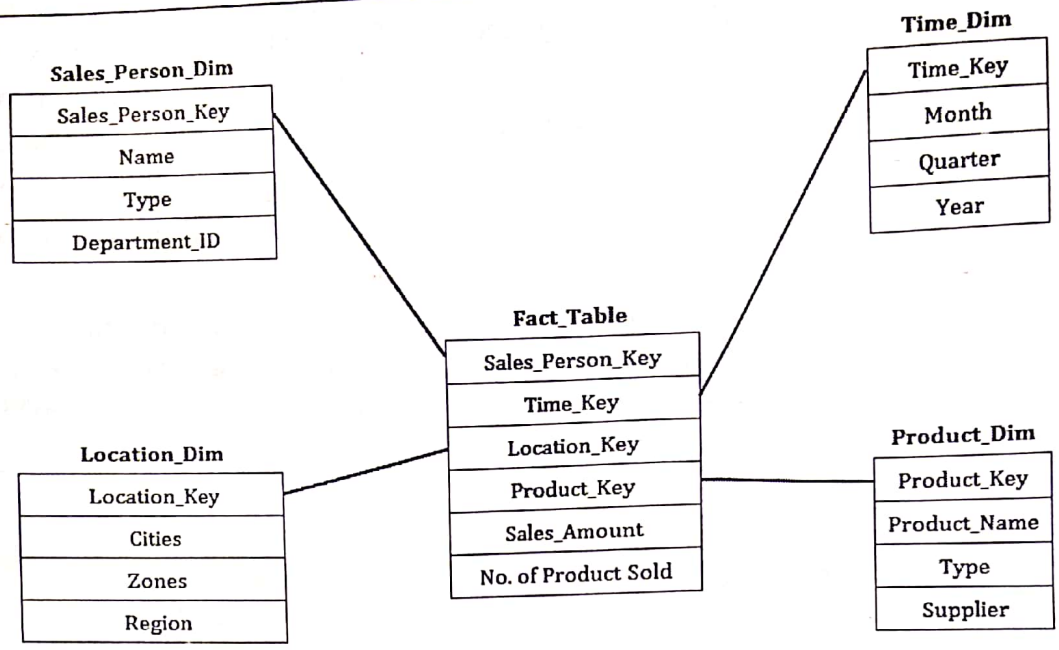


Figure 2.8: Manufacturing Company Star Schema.

STAR SCHEMA TO SNOWFLAKE SCHEMA:

Above star schema can be converted to snowflake schema by considering the following assumptions:

1. Month, Quarter & Year can be further divided.
2. Region can be further divided.
3. Department can be further divided.

SNOWFLAKE SCHEMA:

Refer Q5. Snowflake Schema Section.

SNOWFLAKE SCHEMA FOR MANUFACTURING COMPANY:

Figure 2.9 shows the Snowflake Schema for Manufacturing Company.

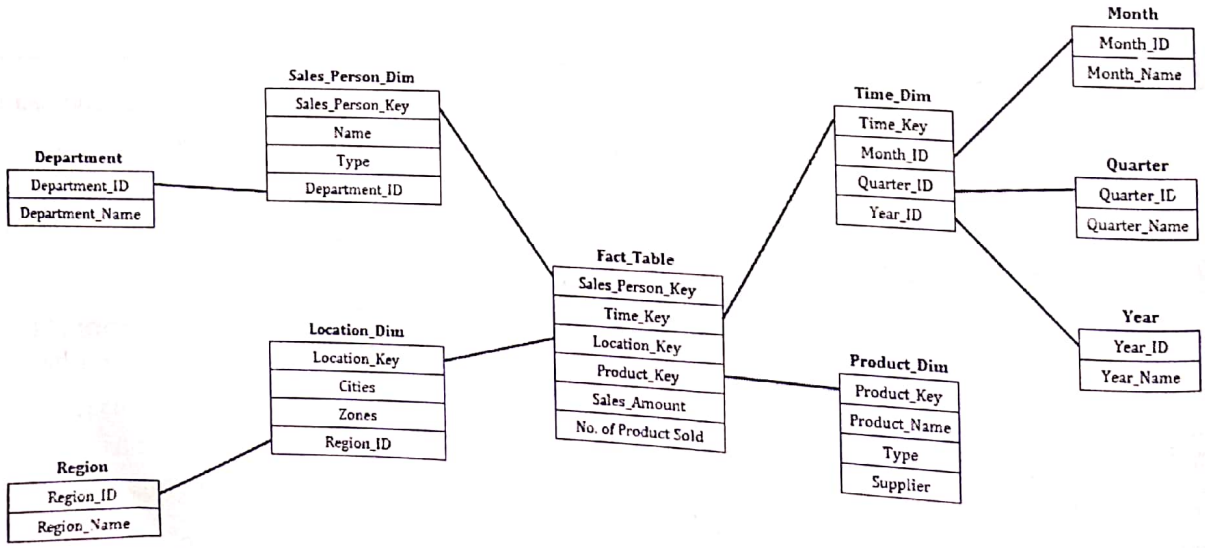


Figure 2.9: Manufacturing Company Snowflake Schema.

Q9. **(i) Design star & snowflake schema for "Hotel Occupancy" considering dimensions like Time, Hotel, Room, etc.**

(ii) Calculate the maximum number of base fact table records for the values given below: Time period: 5 years, Hotels: 150, Rooms: 750 rooms in each Hotel (about 400 occupied in each hotel daily).

Q10. **Information requirements are recorded for "Hotel occupancy" considering dimensions like Hotel, Room and Time. Few Facts recorded are vacant rooms, occupied rooms, number of occupants, etc. Answer the following questions for this problem:**

(i) Design the star schema.

(ii) Can you convert this star schema to a snowflake schema? If yes, justify and draw the snowflake schema

[10M | May18 & Dec18]

Ans:

STAR SCHEMA:

1. Star Schema is the most popular schema design for a Data Warehouse.
2. In data warehousing and business intelligence, a star schema is the simplest form of a dimensional model.
3. It is called a star schema because the diagram resembles a star, with points radiating from a center.
4. The center of the star consists of **fact table** and the points of the star are the **dimension tables**.
5. Usually the fact tables in a star schema are in **third normal form (3NF)** whereas dimensional tables are **de-normalized**.
6. Each dimension in a star schema is represented with only **one-dimension table**.
7. This dimension table contains the set of attributes.
8. For example, the time dimension table may contains the attribute set like date, month, quarter etc.

STAR SCHEMA FOR HOTEL OCCUPANCY:

Figure 2.10 shows the Star Schema for Hotel Occupancy.

Fact Table: Reservation.

Dimension Table: Time, Hotel, Room and Customer.

There are 150 Hotels, 750 Rooms and Time period of 5 Years.

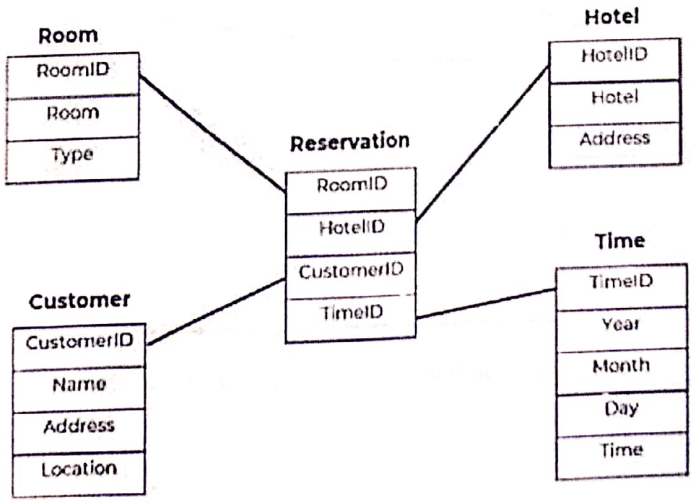


Figure 2.10: Star Schema for Hotel Occupancy.

Maximum No. of fact table records:

Time Period = 5 Years x 365 Days
= 1825

No. of Hotels = 150

Each Hotel daily occupancy = 400

Occupancy = 1

Maximum No. of Fact Table Records = Time Period x No. of Hotels x Daily Occupancy x Occupancy.
= 1825 x 150 x 400 x 1
= 109500000

Maximum No. of Fact Table Records = 109500000.

STAR SCHEMA TO SNOWFLAKE SCHEMA:

Above star schema can be converted to snowflake schema by considering the following assumptions:

1. Room Type can be further divided.
2. Address can be further divided.

SNOWFLAKE SCHEMA:

Refer Q5. Snowflake Schema Section.

SNOWFLAKE SCHEMA FOR HOTEL OCCUPANCY:

Figure 2.11 shows the Snowflake Schema for Hotel Occupancy.

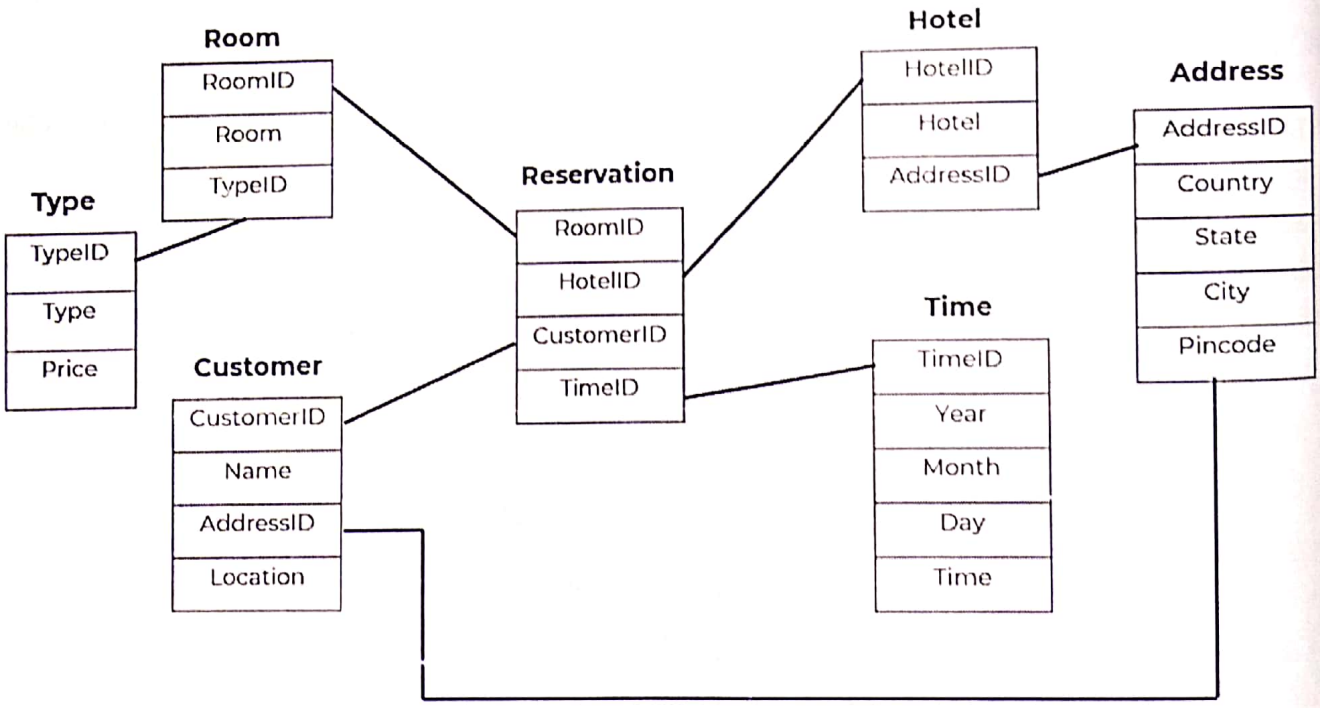


Figure 2.11: Snowflake Schema for Hotel Occupancy.

Q11. Differentiate between Star Schema and Snowflake Schema.

Ans:

COMPARISON BETWEEN STAR SCHEMA AND SNOWFLAKE SCHEMA:

Table 2.1 shows the comparison between Star Schema and Snowflake Schema.

Table 2.1

Point	Star Schema	Snowflake Schema
Data model	Top down approach.	Bottom up approach.
When to use	When dimension table contains less number of rows, we can choose Star schema.	When dimension table is relatively big in size, snow flaking is better as it reduces space.
Dimension table	A star schema contains only single dimension table for each dimension.	A snowflake schema may have more than one dimension table for each dimension.
Joins	Fewer Joins.	Higher number of Joins.
Type of Data Warehouse	Good for data marts with simple relationships (1:1 or 1:many)	Good to use for data warehouse core to simplify complex relationships (many: many)
Query Performance	Less number of foreign keys and hence shorter query execution time (faster)	More foreign keys and hence longer query execution time (slower)
Ease of Use	Lower query complexity and easy to understand.	More complex queries and hence less easy to understand.
Ease of maintenance / change	Has redundant data and hence less easy to maintain/change.	No redundancy, so snowflake schemas are easier to maintain and change.
Normalization/ De-Normalization	Both Dimension and Fact Tables are in De-Normalized form.	Dimension Tables are in Normalized form but Fact Table is in De-Normalized form.

CHAP – 3 | ETL PROCESS

Q1. Describe the steps of ETL Process.

Q2. Explain ETL of data warehousing in details?

Q3. ETL Process.

Q4. Discuss the process of extraction, transformation and loading with a neat and labelled diagram

[10M | May16, May17, Dec17, May18 & Dec18]

Ans:

ETL:

1. ETL Stands for **Extract, Transform** and **Load**.
2. ETL covers a process of how the data are loaded from the source system to the data warehouse.
3. It is a process in data warehousing responsible for **pulling data out** of the source systems and **placing it** into a data warehouse.
4. ETL helps organizations to make meaningful, data-driven decisions by interpreting and transforming enormous amounts of structured and unstructured data.

ETL PROCESS:

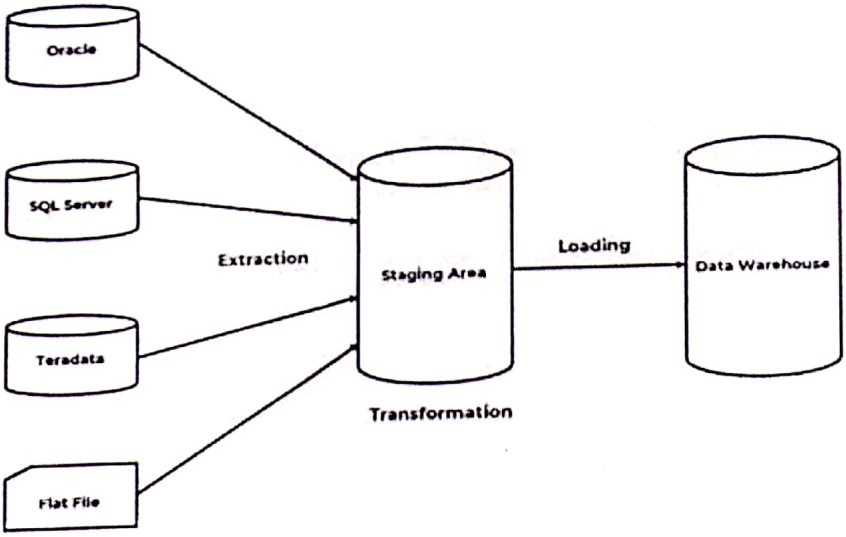


Figure 3.1: ETL Process.

1. Figure 3.1 represents the ETL Process.
2. ETL process is the way to move and prepare data for data analysis.
3. ETL process involves the following tasks:

I) Extracting the data from different sources:

1. This is the first step in ETL process.
2. Different data sources can be **RDBMS** or **files** like CSV, JSON, and XML etc.
3. In this step, Data is **extracted** from source system.
4. Data is also made accessible for further processing.
5. The main objective of the extraction step is to **retrieve all required data** from source system.

6. The extraction step should be designed in a way that it does not negatively affect the source system.
7. Most data projects consolidate data from different source systems.
8. Each separate source uses a different format.
9. Common data-source formats include RDBMS, XML (like CSV, JSON).
10. Thus the extraction process must convert the data into a format suitable for further transformation.

II) Transforming the data:

1. This may involve **cleaning, filtering, validating and applying business rules**.
2. In this step, certain rules are applied on the extracted data.
3. The main aim of this step is to **load the data** to the target database in a cleaned and general format.
4. This is because when the data is collected from different sources each source will have their own standards.
5. For **example** if we have two different data sources A and B.
6. In source A, date format is like dd/mm/yyyy, and in source B, it is yyyy-mm-dd.
7. In the transforming step we convert these dates to a general format.
8. The other things that are carried out in this step are:
    a. **Cleaning** (e.g. "Male" to "M" and "Female" to "F" etc.).
    b. **Filtering** (e.g. selecting only certain columns to load).
    c. **Enriching** (e.g. Full name to First Name , Middle Name , Last Name).
    d. **Splitting** a column into multiple columns and vice versa.
    e. **Joining** together data from multiple sources.
9. In some cases data does not need any transformations and here the data is said to be **"rich data"** or **"direct move"** or **"pass through"** data.

III) Loading:

1. This is the **final step in the ETL process**.
2. In this step, the extracted data and transformed data is **loaded** to the target database.
3. In order to make data load efficient, it is necessary to **index the database** and disable constraints before loading the data.
4. All the three steps in the ETL process can be run parallel.
5. Data extraction takes time and so the second step of transformation process is executed simultaneously.
6. This prepares data for the third step of loading.
7. As soon as some data is ready, it is loaded without waiting for completion of the previous steps.

Q5. In what way ETL cycle can be used in typical data warehouse, explain with suitable instance.

Ans: **[10M | Dec16]**

ETL:

1. ETL Stands for **Extract, Transform** and **Load**.
2. ETL covers a process of how the data are loaded from the source system to the data warehouse.
3. It is a process in data warehousing responsible for **pulling data out** of the source systems and **placing it into a data warehouse**.

4. ETL helps organizations to make meaningful, data-driven decisions by interpreting and transforming enormous amounts of structured and unstructured data.

ETL Cycle:

A typical ETL lifecycle consists of the following 10 steps of execution.

1. Initiation of cycle.
2. Building reference data.
3. Extracting data from different sources.
4. Validation of data.
5. Transforming data.
6. Staging of data.
7. Generation of audit reports.
8. Publishing data.
9. Archiving.
10. Cleanup.

ETL Process:

Refer ETF Section.

USES OF ETL CYCLE IN TYPICAL DATA WAREHOUSE:

1. ETL is the most important step in data warehousing.
2. Data warehousing brings data from different sources onto a single platform and in a single format.
3. So ETL makes analysis of the data easier and effective.
4. ETL is required in taking management decisions.
5. It is used in designing strategies and future plans.

Q6. Data Quality

Ans:

[5M | Dec16]

DATA QUALITY:

1. Data quality can simply be described as a **fitness for use of data**.
2. To be more specific every portion of data has to be accurate to clearly represent the value of itself.
3. **Data cleansing** may be required in order to ensure data quality.
4. Some of the reasons for Dirty Data are listed as follows:
   a. Dummy Values.
   b. Absence of Data.
   c. Non-Unique Identifiers.
   d. Cryptic Data.
   e. Multi-purpose Fields.

DATA QUALITY CYCLE:

Figure 3.2 shows the data quality cycle.

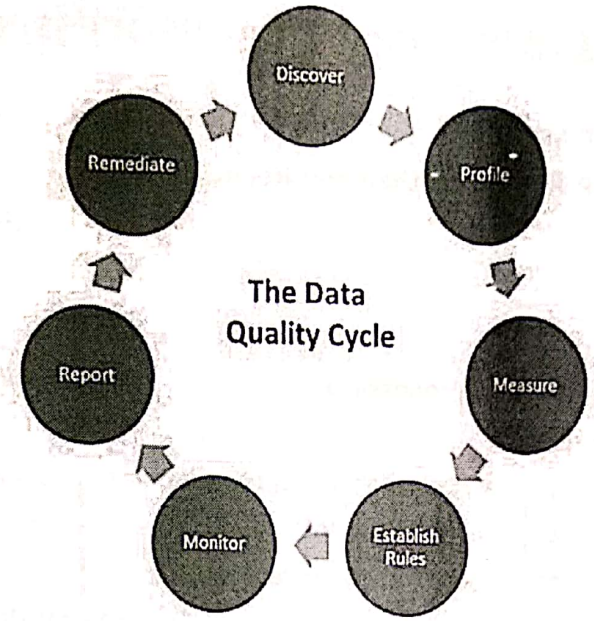


Figure 3.2: Data Quality Cycle.

COMPONENTS OF DATA QUALITY CYCLE INCLUDES:

1. **Data Discovery:** It is the process of finding, gathering, organizing and reporting metadata about data.
2. **Data Profiling:** It is the process of analyzing data in detail, comparing the data to its metadata, calculating data statistics and reporting the measures of quality for the data.
3. **Data Quality Rules:** Based on the business requirements for each Data Quality measure, the data quality rules are made.
4. **Data Quality Monitoring:** It is the process of monitoring of Data Quality, based on the results of executing the Data Quality rules.
5. **Data Quality Reporting:** Dashboards and scorecards are used to report Data Quality measures.
6. **Data Remediation:** It is the ongoing correction of Data Quality exceptions and issues as they are reported.

CHAP - 4: ONLINE ANALYTICAL PROCESSING (OLAP)

Q1. Discuss various OLAP Models.

Q2. Discuss various OLAP Models and their architecture. **[10M | May16, Dec17 & Dec18**

Ans:

OLAP:

1. OLAP Stands for **Online Analytical Processing**.
2. It was defined by **OLAP Council**.
3. OLAP is an approach to answering multi-dimensional analytical (MDA) queries.
4. It is based on the **multidimensional data model**.
5. OLAP is part of **business intelligence**.
6. It allows managers, and analysts to get an insight of the information through fast, consistent, and interactive access to information.

OLAP MODELS:

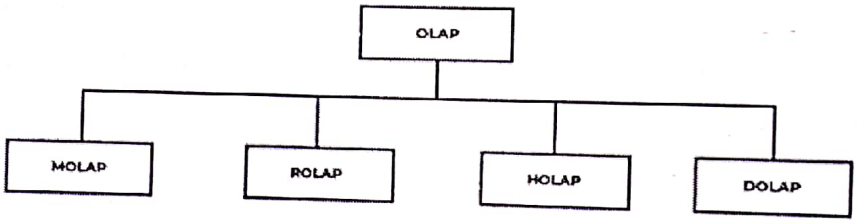


Figure 4.1: OLAP Models.

I) MOLAP:

1. MOLAP Stands for **Multi-dimensional OLAP**.
2. It is the classic form of OLAP.
3. In MOLAP, data is stored in a **multidimensional cube**.
4. It uses array-based multidimensional storage engines.
5. The storage is not in the relational database, but in proprietary formats.
6. Figure 4.2 shows MOLAP Architecture / Process.

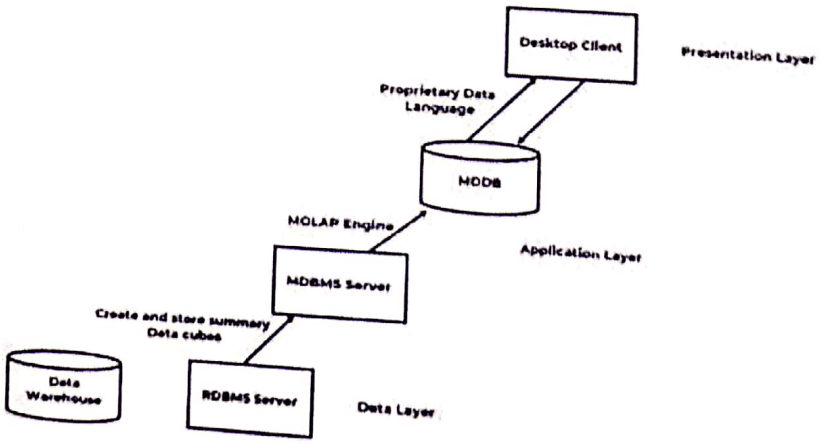


Figure 4.2: MOLAP Architecture/Process.

Advantages:

1. It can perform complex calculations.
2. It has excellent performance.

Disadvantages:

1. It can handle limited amount of data.
2. It requires additional investment.

II) ROLAP:

1. ROLAP Stands for **Relational OLAP**.
2. ROLAP uses relational or extended relational DBMS.
3. ROLAP servers are placed between **relational back-end server** and **client front-end** tools.
4. ROLAP is used for following:
   a. For implementation of aggregation navigation logic.
   b. Optimization for each DBMS back end.
   c. Additional tools and services.
5. Figure 4.3 shows ROLAP Architecture / Process.

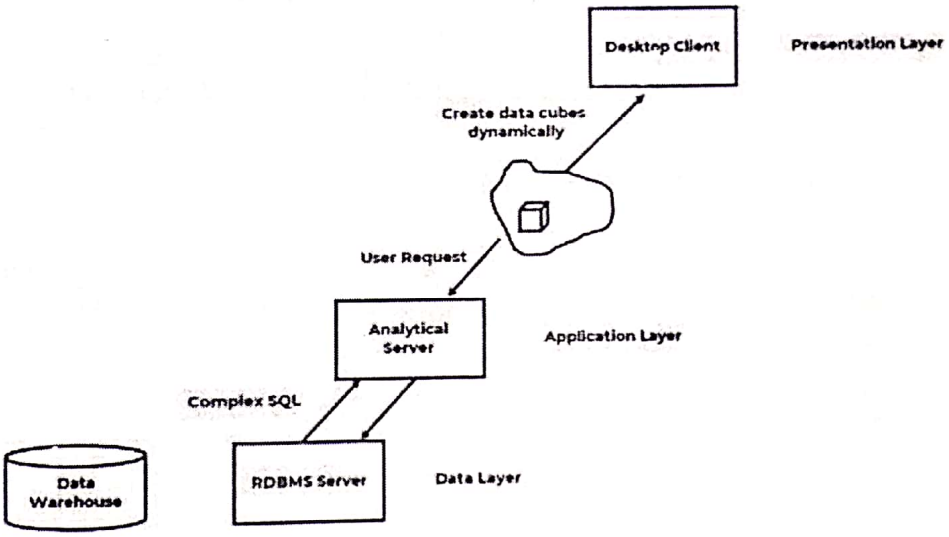


Figure 4.3: ROLAP Architecture / Process.

Advantages:

1. It has higher scalability.
2. It can handle large amount of data.

Disadvantages:

1. Performance is slow.
2. Limited SQL Functionality.

III) HOLAP:

1. HOLAP Stands for **Hybrid OLAP**.
2. Hybrid OLAP is a combination of both ROLAP and MOLAP.
3. It offers **higher scalability of ROLAP** and **faster computation of MOLAP**.
4. HOLAP servers allows to store the large data volumes of detailed information.

IV) DOLAP:

1. DOLAP Stands for **Desktop OLAP**.
2. It is variation of ROLAP.
3. DOLAP requires only DOLAP software to be present on machine.
4. It offers **portability to the users**.

Q3. Indexing OLAP Data. **[5M | Dec16]**

Ans:

INDEXING OLAP DATA:

1. Indexing is used to quickly locate data without having to search every row in a database.
2. Indexing provides the basis for both **rapid random lookups** and **efficient access of ordered records**.
3. Indexing OLAP Data includes **Bitmap Index and Join Indices**.

I) Bitmap Index:

1. Bitmap Index is the **index on particular column**.
2. Each value in the column has a **bit vector**.
3. The length of the bit vector is number of records in the base table.
4. The i^{th} bit is set, if the i^{th} row of the base table has the value for the indexed column.
5. It is not suitable for high cardinality domains.
6. **Example** of Bitmap Index is shown in figure 4.4.

Base Table		
Customer	Region	Type
C1	Asia	Retail
C2	Europe	Dealer
C3	Asia	Dealer
C4	America	Retail
C5	Europe	Dealer

Index on Region			
RecID	Asia	Europe	America
1	1	0	0
2	0	1	0
3	1	0	0
4	0	0	1
5	0	1	0

Index on Type		
RecID	Retail	Dealer
1	1	0
2	0	1
3	0	1
4	1	0
5	0	1

Figure 4.4: Example of Bitmap Index.

II) Join Indices:

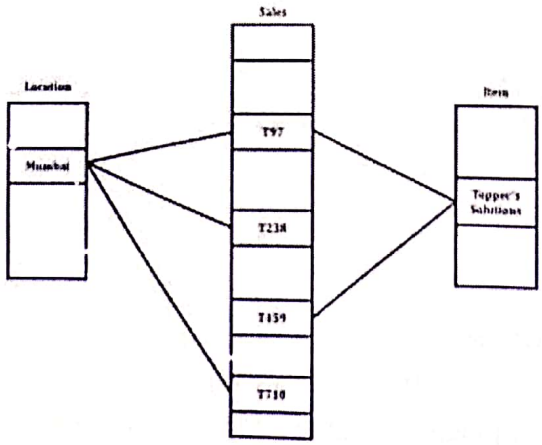


Figure 4.5: Example of Join Indices.

1. Traditional indices map the values to a **list of record ids**.
2. But Join Indices map the values of the dimensions of star schema to rows in the fact table.
3. Join Indices is: **JI (R-id, S-id) where R (R-id, ...) >< S (S-id, ...)**
4. Join indices can span multiple dimensions.
5. Figure 4.5 shows the Example of Join Indices.
6. **Fact Table:** Sales and **Dimension Tables:** Location and Item.

Q4. We would like to view sales data of a company with respect to three dimensions namely Location, Item and Time. Represent the sales data in the form of a 3-D data cube for the above and perform Roll up, Drill down, Slice and Dice OLAP operations on the above data cube and Illustrate.

Ans: **[10M | May16]**

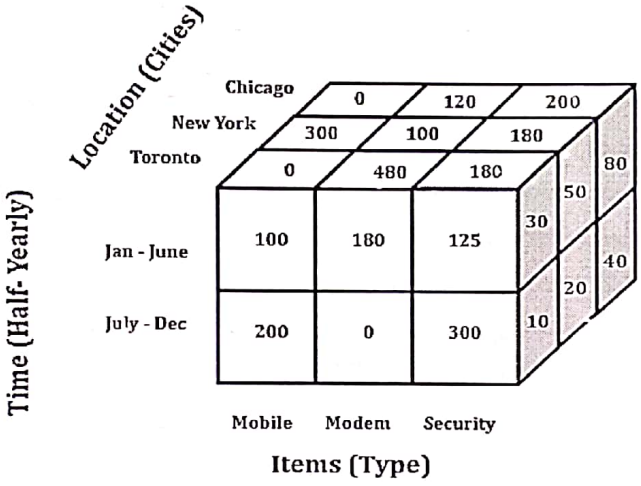


Figure 4.6: OLAP Cube.

1. OLAP Operations are implemented to retrieve the information from data warehouse into OLAP multi-dimensional databases.
2. Since OLAP servers are based on multidimensional view of data, So OLAP operations are performed in multidimensional data.
3. Figure 4.6 represent the cube which will be used for OLAP Operations.

OLAP OPERATIONS:

I) Roll-up:

1. Roll-up performs aggregation on a data cube in any of the following ways:
   a. By climbing up a concept hierarchy for a dimension.
   b. By dimension reduction.
2. The following figure 4.7 illustrates how roll-up works.
3. When roll-up is performed, one or more dimensions from the data cube are removed.
4. We will perform roll up on location dimension.

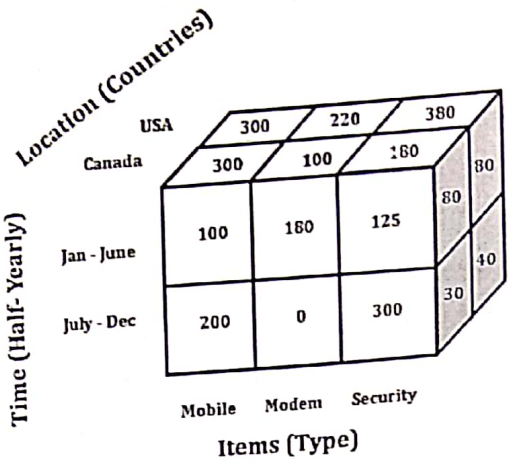


Figure 4.7: Roll-up Operation on Location Dimension.

II) Drill-down:

1. Drill-down is the reverse operation of roll-up.
2. It is performed by either of the following ways:
   a. By stepping down a concept hierarchy for a dimension.
   b. By introducing a new dimension.
3. The following figure 4.8 illustrates how drill-down works.
4. When drill-down is performed, one or more dimensions from the data cube are added.
5. It navigates the data from less detailed data to highly detailed data.

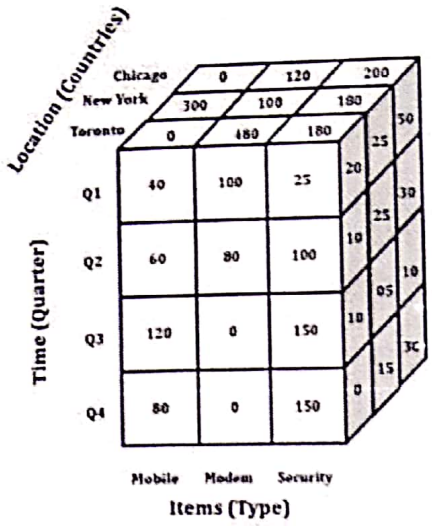


Figure 4.8: Drill Down Operation on Time Dimension.

III) Slice:

1. The slice operation selects one particular dimension from a given cube and **provides a new sub-cube**
2. Consider the following figure 4.9 that shows how slice works.
3. Here Slice is performed for the dimension "time" using the criterion time = "Jan - June".
4. It will form a new sub-cube by selecting one or more dimensions.

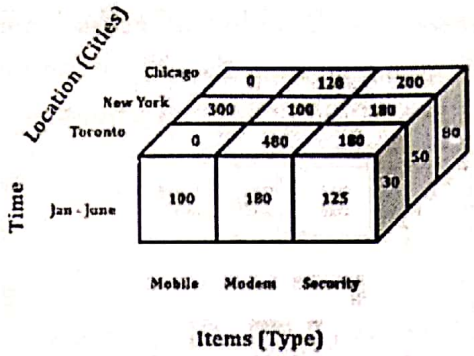


Figure 4.9: Slice Operation on Time Dimension.

IV) Dice:

1. Dice selects two or more dimensions from a given cube and provides a new sub-cube.
2. Consider the following figure 4.10 that shows the dice operation.
3. The dice operation on the cube based on the following selection criteria involves three dimensions.
   a. Location = "Toronto" and "New York"
   b. Time = "Jan-June" and "July-Dec"
   c. Item =" Mobile" and "Security"

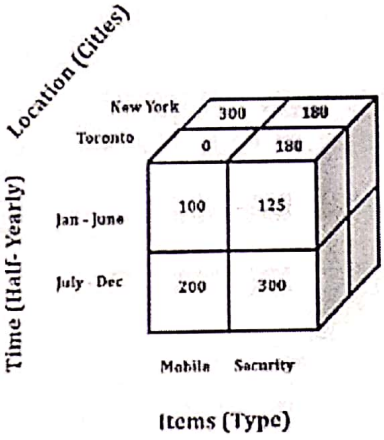


Figure 4.10: Dice Operation.

V) Pivot:

1. The pivot operation is also known as **rotation**.
2. It rotates the data axes in view in order to provide an alternative presentation of data.
3. Consider the following figure 4.11 that shows the pivot operation.
4. In this the item and location axes in 2-D slice are rotated.

Chicago	0	120	200
New York	300	100	180
Toronto	0	480	180
	Mobile	**Modem**	**Security**

Figure 4.11: Pivot Operation.

Q5. The college wants to record the Marks for the courses completed by students using the dimensions: i) Course, ii) Student, iii) Time & a measure Aggregate marks.
Create a Cube and describe following OLAP operations: (i) Slice (ii) Dice (iii) Roll up (iv)Drill down (v) Pivot.

Ans: **[10M | May17, May18 & Dec18]**

OLAP:

1. OLAP Stands for **Online Analytical Processing**.
2. It was defined by **OLAP Council**.
3. OLAP is an approach to answering multi-dimensional analytical (MDA) queries.
4. It is based on the **multidimensional data model**.
5. OLAP is part of **business intelligence**.
6. It allows managers, and analysts to get an insight of the information through fast, consistent, and interactive access to information.

OLAP CUBE:

1. There are four tables, out of 3 dimension tables and 1 fact table.
2. Dimension Tables:
   a. Student: (SID, Name, Phone, Location, Pin)
   b. Course: (CID, Name, Phone, State, City, Pin)
   c. Time: (TID, Day, Month, Quarter, Year)
3. Fact Table:
   a. College: (SID, CID, TID, Aggregate Marks)
4. Figure 4.12 shows cube of marks obtained by students.

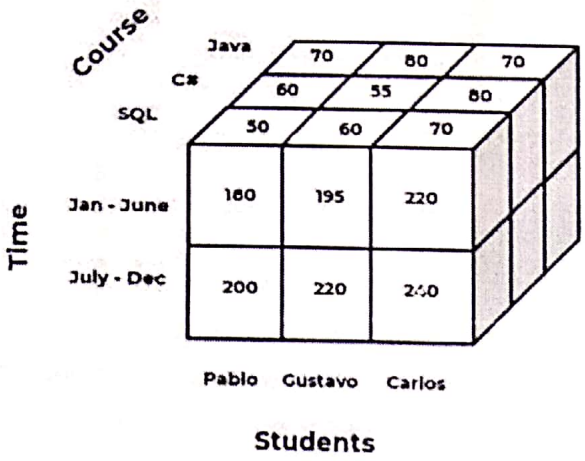


Figure 4.12: Cube of marks obtained by students.

OLAP OPERATIONS:

I) Roll-up:

1. Roll-up performs aggregation on a data cube in any of the following ways:
   a. By climbing up a concept hierarchy for a dimension.
   b. By dimension reduction.
2. The following figure 4.13 illustrates how roll-up works.

3. When roll-up is performed, one or more dimensions from the data cube are removed.
4. We will perform roll up on course dimension.

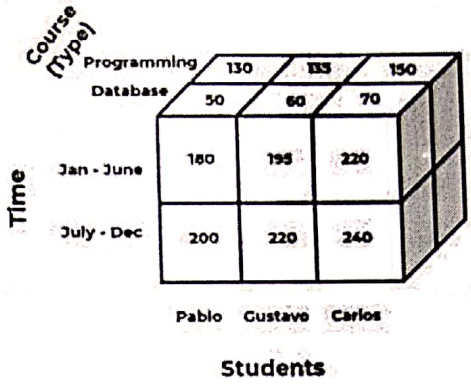


Figure 4.13: Roll-up Operation on Course Dimension.

II) Drill-down:

1. Drill-down is the reverse operation of roll-up.
2. It is performed by either of the following ways:
   a. By stepping down a concept hierarchy for a dimension.
   b. By introducing a new dimension.
3. The following figure 4.14 illustrates how drill-down works.
4. When drill-down is performed, one or more dimensions from the data cube are added.
5. It navigates the data from less detailed data to highly detailed data.

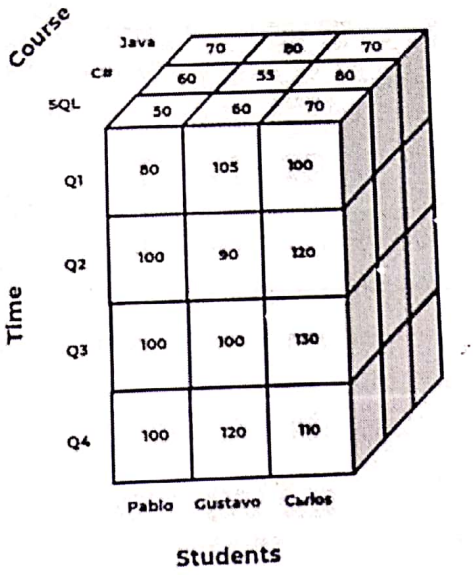


Figure 4.14: Drill Down Operation on Time Dimension.

III) Slice:

1. The slice operation selects one particular dimension from a given cube and provides a new sub-cube.
2. Consider the following figure 4.15 that shows how slice works.
3. Here Slice is performed for the dimension "time" using the criterion time = "Jan - June".
4. It will form a new sub-cube by selecting one or more dimensions.

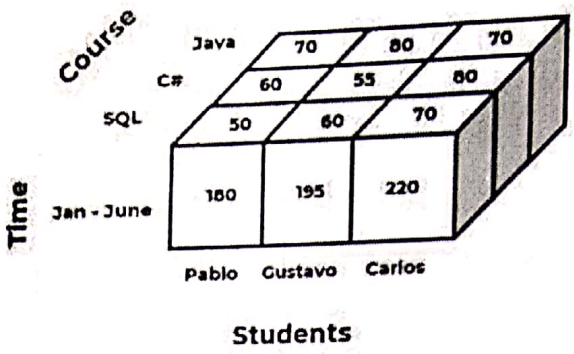


Figure 4.15: Slice Operation on Time Dimension.

IV) Dice:

1. Dice selects two or more dimensions from a given cube and provides a new sub-cube.
2. Consider the following figure 4.16 that shows the dice operation.

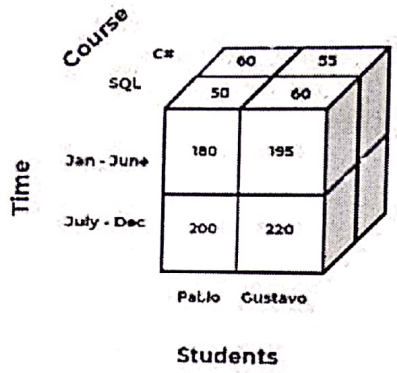


Figure 4.16: Dice Operation.

3. The dice operation on the cube based on the following selection criteria involves three dimensions.
   a. Course = "C#" and "SQL"
   b. Time = "Jan-June" and "July-Dec"
   c. Students ="Pablo" and "Gustavo"

V) Pivot:

1. The pivot operation is also known as **rotation**.
2. It rotates the data axes in view in order to provide an alternative presentation of data.
3. Consider the following figure 4.17 that shows the pivot operation.
4. In this the item and location axes in 2-D slice are rotated.

Java	70	80	70
C#	60	55	80
SQL	50	60	70
	Pablo	**Gustavo**	**Carlos**

Figure 4.17: Pivot Operation.

Q6. Consider a data warehouse for a hospital where there are three dimension

a) Doctor b) Patient c) Time

Consider two measures i) Count ii) Charge, where charge is the fee that the doctor charges a patient for a visit. For the above example create a cube and illustrate the following OLAP operations.

1) Rollup 2) Drill Down 3) Slice 4) Dice 5) Pivot.

Ans:

[10M | Dec17]

OLAP OPERATIONS:

1. OLAP Operations are implemented to retrieve the information from data warehouse into OLAP multi-dimensional databases.
2. Since OLAP servers are based on multidimensional view of data, So OLAP operations are performed in multidimensional data.

CUBE:

1. There are four tables, out of 3 dimension tables and 1 fact table.
2. Dimension Tables:
   a. Doctor: (DID, Name, Phone, Location, Pin)
   b. Patient: (PID, Name, Phone, State, City, Pin)
   c. Time: (TID, Day, Month, Quarter, Year)
3. Fact Table:
   a. Hospital: (DID, PID, TID, Count, Charge)
4. Figure 4.18 shows cube of data warehouse for a hospital.

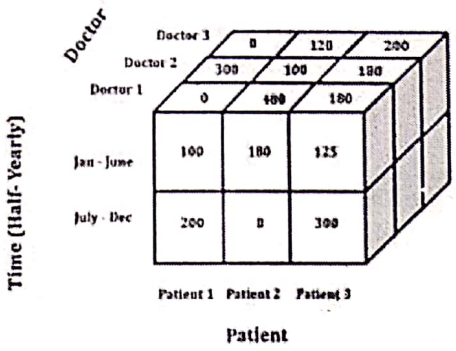


Figure 4.18: Cube of marks obtained by students.

OLAP OPERATIONS:

I) Roll-up:

1. Roll-up performs aggregation on a data cube in any of the following ways:
   a. By climbing up a concept hierarchy for a dimension.
   b. By dimension reduction.
2. The following figure 4.19 illustrates how roll-up works.
3. When roll-up is performed, one or more dimensions from the data cube are removed.
4. We will perform roll up on course dimension.

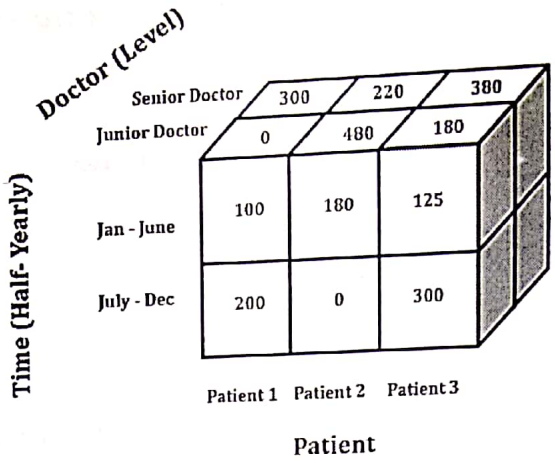


Figure 4.19: Roll-up Operation on Doctor Dimension.

II) Drill-down:

1. Drill-down is **the reverse operation of roll-up**.
2. It is performed by either of the following ways:
   a. By stepping down a concept hierarchy for a dimension.
   b. By introducing a new dimension.
3. The following figure 4.20 illustrates how drill-down works.
4. When drill-down is performed, one or more dimensions from the data cube are added.
5. It navigates the data from less detailed data to highly detailed data.

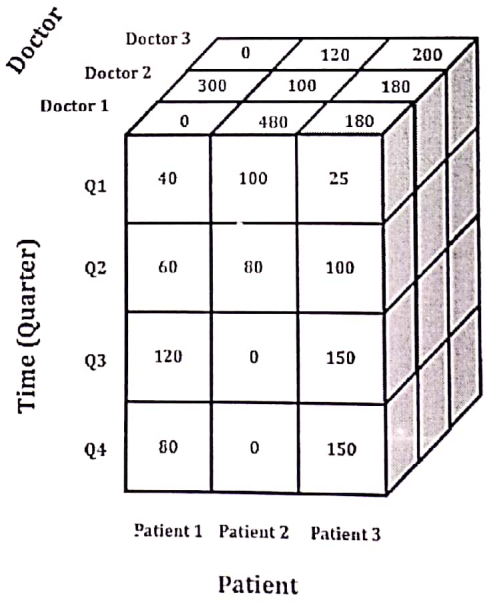


Figure 4.20: Drill Down Operation on Time Dimension.

III) Slice:

1. The slice operation selects one particular dimension from a given cube and provides a new sub-cube.
2. Consider the following figure 4.21 that shows how slice works.
3. Here Slice is performed for the dimension "time" using the criterion time = "Jan - June".
4. It will form a new sub-cube by selecting one or more dimensions.

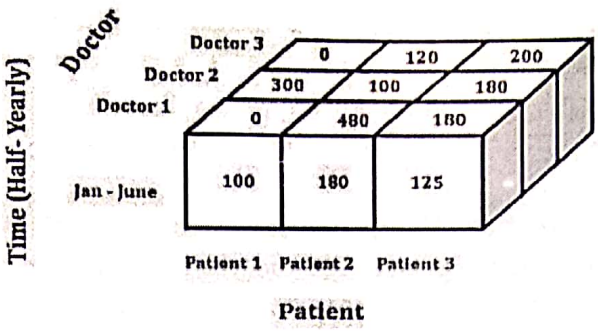


Figure 4.21: Slice Operation on Time Dimension.

IV) Dice:

1. Dice selects two or more dimensions from a given cube and provides a new sub-cube.
2. Consider the following figure 4.22 that shows the dice operation.
3. The dice operation on the cube based on the following selection criteria involves three dimensions.
   a. Doctor = "Doctor 1" and "Doctor 2"
   b. Time = "Jan-June" and "July-Dec"
   c. Patient ="Patient 1" and "Patient 2"

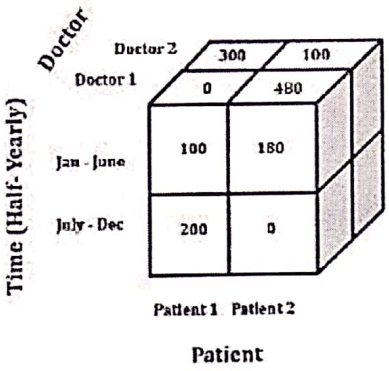


Figure 4.22: Dice Operation.

V) Pivot:

1. The pivot operation is also known as rotation.
2. It rotates the data axes in view in order to provide an alternative presentation of data.
3. Consider the following figure 4.23 that shows the pivot operation.
4. In this the item and location axes in 2-D slice are rotated.

Doctor 3	0	120	200
Doctor 2	300	100	180
Doctor 1	0	480	180
	Patient 1	**Patient 2**	**Patient 3**

Figure 4.23: Pivot Operation.

Q7. Discuss how computations can be performed efficiently on data cubes. [10M | Dec16]

Ans:

DATA CUBE COMPUTATION:

1. Data cube computation is an essential task in **data warehouse implementation**.
2. The pre-computation of all or part of a data cube can greatly reduce the response time and enhance the performance of OLAP.
3. However, such computation is challenging because it may require substantial computational time and storage space.

DATA CUBE COMPUTATION METHODS:

I) Multi-way Array Aggregation:

1. The Multi-way Array Aggregation method computes a full data cube by using a **multidimensional array**.
2. It is array based **bottom up algorithm**.
3. It is a typical MOLAP approach that uses direct array addressing.
4. It uses multi-dimensional chunks.
5. Figure 4.24 shows Multi-way Array Aggregation exploration for a 3-D data cube computation.

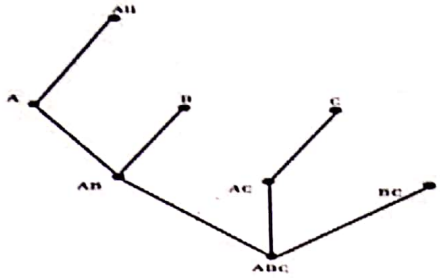


Figure 4.24: Multi-way Array Aggregation exploration for a 3-D data cube computation.

Limitations: It can compute well only for a small number of dimensions.

II) BUC:

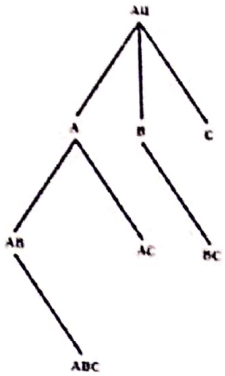


Figure 4.25: BUC exploration for a 3-D data cube computation.

1. BUC Stands for **Bottom Up Cube Computation**.
2. BUC is an algorithm for the **computation of sparse and iceberg cubes.**

3. BUC divides dimensions into partitions and facilitates iceberg pruning.
   a. If a partition does not satisfy min_sup, its descendants can be pruned.
   b. If min_sup = 1 → compute full CUBE.
4. In BUC, No simultaneous aggregation is allowed.
5. Figure 4.25 shows BUC exploration for a 3-D data cube computation.

III) Star Cubing:

1. Star-Cubing combines the **strengths of the Multi-way array aggregation and BUC**.
2. It integrates top-down and bottom-up cube computation.
3. It explores both multidimensional aggregation (similar to Multi-Way) and Apriori-like pruning (similar to BUC).
4. It operates from a data structure called a **star-tree**.

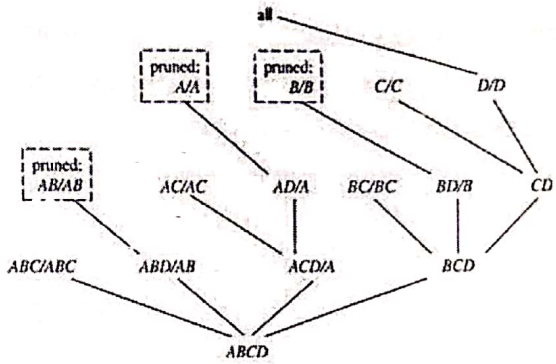


Figure 4.26: Star-Cubing bottom-up computation with top-down expansion of shared dimensions.

Advantage: Reduce the computation time and memory requirements.

Q8. Differentiate OLTP Vs OLAP

Ans: **[5M | Dec16]**

Table 4.1 shows the comparison between OLTP and OLAP.

Table 4.1

Parameters	OLTP	OLAP
Full Form	Online Transaction Processing.	Online Analytical Processing.
Oriented	Transaction Oriented.	Subject Oriented.
Characteristics	Operational Processing.	Informational Processing.
Data Redundancy	Data Redundancy is bad.	Data Redundancy is good.
Granularity	Few Levels of Granularity.	Multiple Level of Granularity.
Users	Many Users	Few Users.
Size	10 MB to GB	100 GB to TB.
Priority	High Performance and Availability.	High Flexibility.
Access	Read and write.	Mostly Read.
Function	It is used for Day to Day Operations.	It is used for long term informational requirements.

CHAP - 5: INTRODUCTION TO DATA MINING

Q1. Describe the various functionalities of Data Mining as a step in the process of knowledge discovery. **[10M | Dec16]**

Ans:

DATA MINING:

1. Data Mining is defined as the procedure of extracting information from huge sets of data.
2. It is a **non-trivial process**.
3. It is used to identify patterns and establish relationships.
4. Data mining is the analysis step of the "knowledge discovery in databases" process, or KDD.

KDD:

1. KDD Stands for **Knowledge Discovery in Database**.
2. KDD is the process of discovering knowledge in data.
3. The main goal is to **extract knowledge** from large database.
4. KDD includes wide variety of application domains which includes Artificial Intelligence, Pattern Recognition, Machine Learning Statistics and Data Visualization.
5. Figure 5.1 shows the KDD Process.

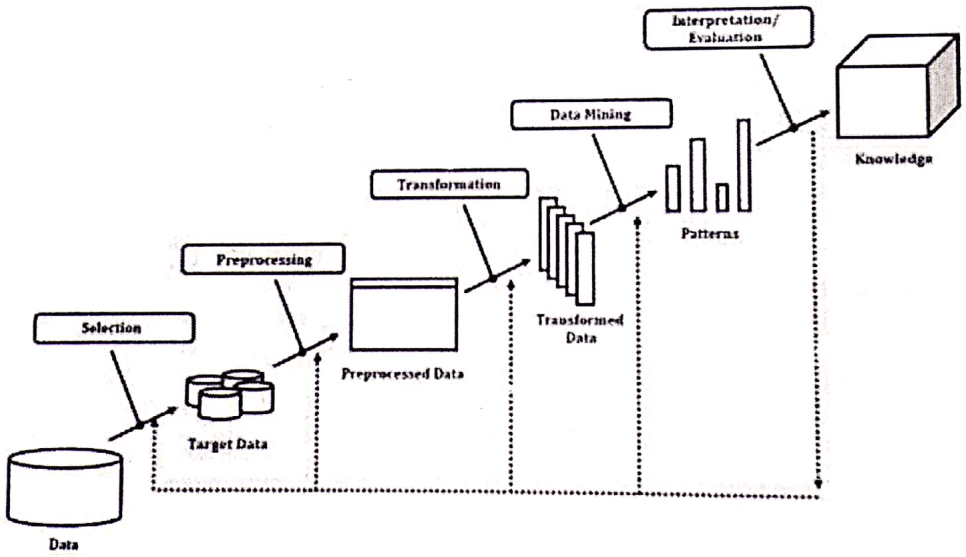


Figure 5.1: KDD Process.

List of steps involved in the knowledge discovery process:

I) Data Cleaning:

1. In this step, the noise and inconsistent data is removed.

II) Data Integration:

1. In this step, multiple data sources are combined.

III) Data Selection:

1. In this step, data relevant to the analysis task are retrieved from the database.

IV) Data Transformation:

1. In this step, data is transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations.

V) Data Mining:

1. In this step, intelligent methods are applied in order to extract data patterns.

VI) Pattern Evaluation:

1. In this step, data patterns are evaluated.
2. It is used to identify the **truly interesting patterns** representing knowledge based on interesting measures.

VII) Knowledge Presentation:

1. In this step, knowledge is represented.
2. **Visualization** and **knowledge representation techniques** are used to present mined knowledge to users.

Q2. Discuss:

1. **The steps in KDD Process.**
2. **The architecture of a typical DM System.**

Q3. Explain Data mining as a step in KDD .Illustrate the architecture of typical data mining system

Ans: **[10M | May16, May18 & Dec18]**

DATA MINING:

Refer Q1 Data Mining Section.

KDD PROCESS:

Refer Q1 KDD Section.

ARCHITECTURE OF A TYPICAL DM SYSTEM:

Figure 5.2 shows the Architecture of a typical data mining system.

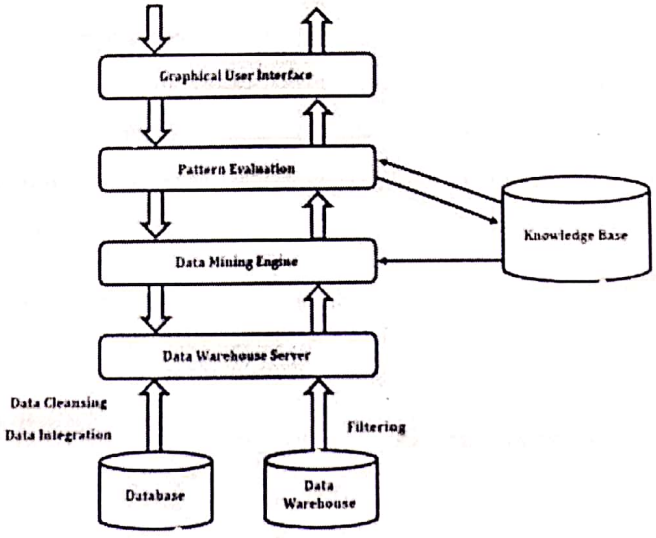


Figure 5.2: Architecture of Typical Data Mining System.

I) Database, data warehouse, or other information repository:

1. This is **information repository**.
2. Data cleaning and data integration techniques are performed on the data.

II) Databases or data warehouse server:

1. It fetches the data as per the users' requirement which is need for data mining task.

III) Knowledge base:

1. This is used to guide the search, and gives the interesting and hidden patterns from data.

IV) Data mining engine:

1. It performs the data mining task such as **characterization, association, classification, cluster analysis** etc.

V) Pattern evaluation module:

1. It is integrated with the mining module and it give the search of only the interesting patterns.

VI) Graphical user interface:

1. This module is used to communicate between user and the data mining system.
2. It allow users to browse databases or data warehouse schemas.

Q4. Major issues in Data Mining

Ans: **[5M | May17]**

ISSUES IN DATA MINING:

1. Data mining is not an easy task, as the algorithms used can get very complex and data is not always available at one place.
2. It needs to be integrated from various heterogeneous data sources.
3. These factors also create some issues.
4. So the major issues regarding Data Mining are as follows:

I) Mining Methodology and User Interaction:

1. Mining different kinds of knowledge in databases:

a. Different users may be interested in different kinds of knowledge.

b. Therefore it is necessary for data mining to cover a broad range of knowledge discovery task.

2. Interactive mining of knowledge at multiple levels of abstraction:

a. The data mining process needs to be interactive because it allows users to focus the search for patterns, providing and refining data mining requests based on the returned results.

3. Incorporation of background knowledge:

a. To guide discovery process and to express the discovered patterns, the background knowledge can be used.

4. Data mining query languages and ad hoc data mining:

a. Data Mining Query language that allows the user to describe ad hoc mining tasks, should be integrated with a data warehouse query language.

5. Presentation and visualization of data mining results:

a. Once the patterns are discovered it needs to be expressed in high level languages, and visual representations.

b. These representations should be easily understandable.

6. **Handling noisy or incomplete data:**
   a. The data cleaning methods are required to handle the noise and incomplete objects while mining the data regularities.
7. **Pattern evaluation:**
   a. The patterns discovered should be interesting because either they represent common knowledge or lack novelty.

II) Performance Issues:

1. **Efficiency and scalability of data mining algorithms:**
   a. In order to effectively extract the information from huge amount of data in databases, data mining algorithm must be efficient and scalable.
2. **Parallel, distributed, and incremental mining algorithms:**
   a. The factors such as huge size of databases, wide distribution of data, and complexity of data mining methods motivate the development of parallel and distributed data mining algorithms.

III) Diverse Data Types Issues:

1. **Handling of relational and complex types of data:**
   a. The database may contain complex data objects, multimedia data objects, spatial data, temporal data etc.
   b. It is not possible for one system to mine all these kind of data.
2. **Mining information from heterogeneous databases and global information systems:**
   a. The data is available at different data sources on LAN or WAN.
   b. These data source may be structured, semi structured or unstructured.

Q5. Discuss:
(i) Architecture of a typical data mining system.
(ii) Application and major issues in Data Mining.

Ans: **[10M | Dec17]**

ARCHITECTURE OF A TYPICAL DATA MINING SYSTEM:

Refer Q1 Data Mining Section.

APPLICATION OF DATA MINING:

1. Financial Data Analysis.
2. Retail Industry.
3. Telecommunication Industry.
4. Biological Data Analysis.
5. Other Scientific Applications.
6. Intrusion Detection.

MAJOR ISSUES IN DATA MINING:

Refer Q4 Issues in Data Mining Section.

Q6. Application of Data Mining to Financial Analysis.

Q7. Applications of Data Mining **[5M | Dec16, May18 & Dec18]**

Ans:

DATA MINING:

1. Data Mining is defined as the **procedure of extracting information from huge sets of data**.
2. It is a **non-trivial process**.
3. It is used to identify patterns and establish relationships.
4. Data mining is the analysis step of the "knowledge discovery in databases" process, or KDD.

APPLICATION OF DATA MINING TO FINANCIAL ANALYSIS:

1. The financial data in banking and financial industry is generally **reliable and of high quality**.
2. So it facilitates the systematic data analysis and data mining.
3. Some of the typical cases are as follows:
   a. Design and construction of data warehouses for multidimensional data analysis and data mining.
   b. Loan payment prediction.
   c. Customer credit policy analysis.
   d. Classification and clustering of customers for targeted marketing.
   e. Detection of money laundering and other financial crimes.

APPLICATION OF DATA MINING TO RETAIL INDUSTRY:

1. Data Mining has its great application in Retail Industry because it collects large amount of data from on sales, customer purchasing history, goods transportation, consumption and services.
2. Data mining in retail industry helps in identifying customer buying patterns and trends that lead to improved quality of customer service and good customer retention and satisfaction.

TELECOMMUNICATION INDUSTRY:

1. Data mining in telecommunication industry helps in identifying the telecommunication patterns, catch fraudulent activities, make better use of resource, and improve quality of service.
2. List of examples for which data mining improves telecommunication services:
   a. Multidimensional Analysis of Telecommunication data.
   b. Fraudulent pattern analysis.
   c. Use of visualization tools in telecommunication data analysis.

BIOLOGICAL DATA ANALYSIS:

1. Biological data mining is a very important part of **Bioinformatics**.
2. Following are the aspects in which data mining contributes for biological data analysis:
   a. Discovery of structural patterns and analysis of genetic networks and protein pathways.
   b. Association and path analysis.
   c. Visualization tools in genetic data analysis.

OTHER SCIENTIFIC APPLICATIONS:

Applications of data mining in the field of Scientific Applications:

a. Data Warehouses and data preprocessing.
b. Graph-based mining.
c. Visualization and domain specific knowledge

INTRUSION DETECTION:

1. Intrusion refers to any kind of action that threatens integrity, confidentiality, or the availability of network resources.
2. List of areas in which data mining technology may be applied for intrusion detection:
   a. Development of data mining algorithm for intrusion detection.
   b. Analysis of Stream data.
   c. Distributed data mining.
   d. Visualization and query tools.

CHAP – 6: DATA EXPLORATION

Q1. Data Visualization [5M | Dec17]

Ans:

DATA VISUALIZATION:

1. Data Visualization is the study of the **visual representation of data**.
2. The data can be in **graphical or pictorial format**.
3. Data visualization is both an art and a science.
4. Data visualization refers to the techniques used to communicate data or information by encoding it as visual objects (e.g., points, lines or bars) contained in graphics.
5. The goal is to communicate information clearly and efficiently to users.
6. It is one of the steps in **data analysis**.

DATA VISUALIZATION TECHNIQUES:

I) Pixel-Oriented Visualization Techniques:

1. In pixel based visualization techniques, the color of pixel represents the data object's value.
2. For each data set a separate pixel window is created.
3. The values are then mapped to corresponding pixel positions.
4. The pixel color are chosen like smaller values → Lighter color and bigger values → Darker the color.
5. The color mapping of the pixel is decided on the basis of data characteristics and visualization tasks.

II) Geometric Projection Visualization Techniques:

Techniques used to find geometric transformation are:

1. **Scatter-plot matrices:** It consists of scatter plots of all possible pairs of variables in a dataset.
2. **Hyper slice:**
   a. It is an extension to scatter-plot matrices.
   b. They represent multi-dimensional function as a matrix of orthogonal two dimensional slices.
3. **Parallel co-ordinates:**
   a. The parallel vertical lines which are separated defines the axes.
   b. A point in the Cartesian coordinates corresponds to a polyline in parallel coordinates.

III) Icon Based Visualization Techniques:

1. Icon-based visualization techniques are also known as **iconic display techniques**.
2. Each multidimensional data item is mapped to an icon.
3. This technique allows visualization of large amount of data.
4. The most commonly used technique is **Chernoff faces**.

IV) Hierarchical Visualization Techniques:

1. Hierarchical visualization techniques are used for partitioning of all dimensions in to subset.
2. These subsets are visualized in hierarchical manner.
3. Some of the visualization techniques are:
   a. Dimensional Stacking.
   b. Mosaic Plot.

d. Tree Maps.
e. Visualization of complex data and relations.

Q2. Explain types of attributes and data visualization for data exploration.

Ans:

[10M | May17 & Dec18]

ATTRIBUTES:

1. The attribute is the **property of the object**.
2. An attribute is a data field that is representing a feature of a data object.
3. Attribute values helps to analyze the nature of the data object.
4. **Example:** Roll No, Name, and Result are attributes of the object Student.

TYPES OF ATTRIBUTES:

I) Ordinal Attribute:

1. An ordinal attribute is an attribute with the **meaningful order**.
2. The data values of such an attribute represents some ordering or ranking with predefined magnitude differences.
3. **Example:** 1st, 2nd, 3rd, etc.

II) Binary Attribute:

1. Binary attribute is an attribute having only 2 data values i.e. 0 or 1.
2. 0 represents false/absent and 1 represents true/present.
3. The binary attribute are also called **symmetrical attribute** because there are always 50-50 chances of data values i.e. either YES or NO; no other values.
4. **Example:** Binary Codes: 1 or 0.

III) Nominal Attribute:

1. It is also called as **Categorical Attribute**.
2. Nominal attribute is a naming attribute that means the data values are the names or symbols of the object.
3. **Example:** State names of Country object.

IV) Numeric Attribute:

1. This is a **quantitative attribute**.
2. It can be measured in terms of a quantity.
3. Numeric Attribute has the 2 types:
4. **Interval Scaled Attribute:**
   a. Interval Scaled Attributes are continuous measurement on a linear scale.
   b. Mathematical operations of means, median and mode can be applied.
   c. **Example:** Year 2000 → 2005 → 2010... has a equal size interval of 5 years.
5. **Ratio Scaled Attribute:**
   a. Ratio Scaled Attributes are continuous positive measurement on a non linear scale.

b. Operations like addition, subtraction can be performed.
c. Multiplication and division are not possible.
d. In this type of attribute, the values comprises zero reference point.
e. **Example:** Year of Experience (0 – 5 Years)

V) Continuous & Discrete Attribute

1. If an attribute can take any value between two specified values then it is called as continuous else it is discrete.
2. An attribute will be continuous on one scale and discrete on another.
3. **Continuous Attribute Example:** Direction of Travel.
4. **Discrete Attribute Example:** A Person's Age in Years.

DATA VISUALIZATION:

Refer Q1.

CHAP - 7: DATA PREPROCESSING

Q1. Discuss different steps involved in Data Preprocessing.

Q2. Data pre-processing

Ans:

[Q1 | 10M – May16, May-17, Dec17, May18 & Dec18]

DATA PREPROCESSING:

1. Data pre-processing is an important step in the **data mining process**.
2. Data preprocessing involves **transforming** raw data into an understandable format.
3. Real-world data is often incomplete, inconsistent, and/or lacking in certain behaviors or trends, and is likely to contain many errors.
4. Data preprocessing is a proven method of resolving such issues.
5. Data preprocessing prepares raw data for further processing.

STEPS INVOLVED IN DATA PREPROCESSING:

I) Data Cleaning:

1. Data Cleaning is also known as **scrubbing**.
2. It is a technique that is applied to **remove the noisy data** and **correct the inconsistencies** in data.
3. It involves filling missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies.
4. Steps in data cleansing:
   a. **Parsing:** Parsing is the process in which individual data elements are located and identified in the source systems and then these elements are isolated in the target files.
   b. **Correcting:** In this step, using data algorithm the individual data elements are corrected.
   c. **Standardizing:** In standardizing process, conversion routines are used to transform data into a consistent format using both standard and custom business rules.
   d. **Matching:** Matching process involves eliminating duplications by searching and matching records.
   e. **Consolidating:** Consolidating process involves merging the records into one representation by analyzing and identifying relationship between matched records.
5. Figure 7.1 shows example of data cleaning process.

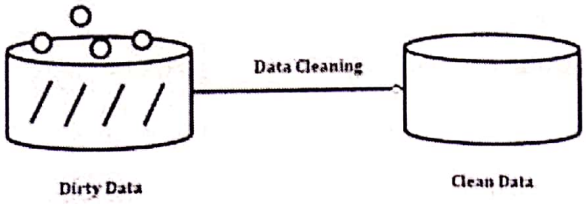


Figure 7.1: Data Cleaning Process.

II) Data Integration:

1. Data integration involves combining data residing in different sources and providing users with a unified view of these data.
2. Sources may include multiple databases, data cubes or data files.

3. Data Integration removes the duplicate and redundant data.
4. Figure 7.2 shows example of data integration process.

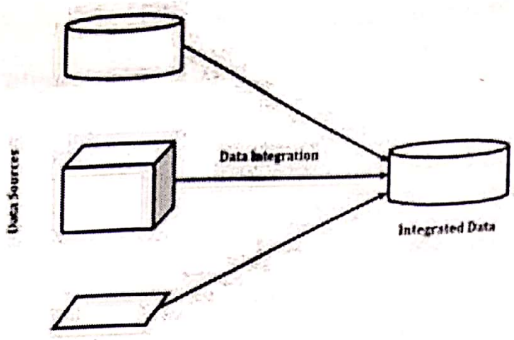


Figure 7.2: Data Integration Process.

III) Data Transformation:

1. In Data Transformation, data are transformed or consolidated into forms appropriate for mining.
2. Data transformation involves:
   - Smoothing.
   - Aggregation.
   - Generalization.
   - Normalization.
3. Figure 7.3 shows the example of data transformation process.

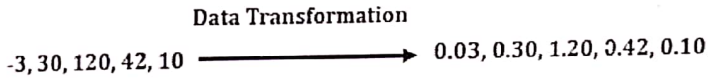


Figure 7.3: Data Transformation Process.

IV) Data Reduction:

1. Data Reduction is used to obtain a reduced representation of the data set that is much smaller in volume.
2. Strategies for data reduction includes:
   a. **Data Cube Aggregation:** In Data Cube Aggregation, aggregation operations are applied to the data in the construction of a data cube.
   b. **Attribute subset selection:** This process is used to detect and remove irrelevant, weakly relevant, or redundant attributes or dimensions.
   c. **Dimensionality Reduction:** In this process encoding mechanisms are used to reduce the data set size.
   d. **Numerosity Reduction:** In this process, the data are replaced by alternative, smaller data representations such as parametric models and non-parametric models like clustering.
3. Figure 7.4 shows data reduction process example.

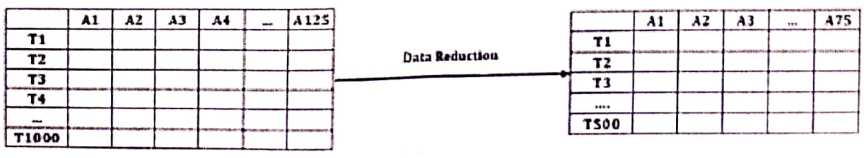


Figure 7.4: Data Reduction Process.

v) Data Discretization:

1. In Data Discretization, the range of a continuous attribute is divided into intervals.
2. By discretization the size of the data is reduced.
3. In this process, the data is prepared for further analysis.
4. Discretization process is applied recursively on an attribute.
5. Three types of attributes:
   a. **Nominal:** Values from an unordered set.
   b. **Ordinal:** Values from an ordered set.
   c. **Continuous:** Real numbers.

CHAP - 8: CLASSIFICATION

Q1. Decision Tree based Classification Approach. **[5M | Dec16]**

Ans:

DECISION TREE BASED CLASSIFICATION:

1. It is one of the most important **classification and prediction method** in data mining.
2. A decision tree represents **rules**.
3. Rules are easy to understand and can be directly used in SQL to retrieve the records from database.
4. A decision tree classifier has **tree type structure**.
5. It has **leaf nodes and decision nodes**.
6. A leaf node is the last node of each branch and indicates value of target attribute.
7. A decision node is the node of tree which has leaf node or sub-tree.
8. Figure 8.1 shows the representation of decision tree for tennis play.
9. As shown in figure 8.1, Humidity, Outlook and Wind is Attribute.
10. High, Normal, Strong, Weak, Sunny, Rain and Overcast is Value.
11. Yes and No is classification.

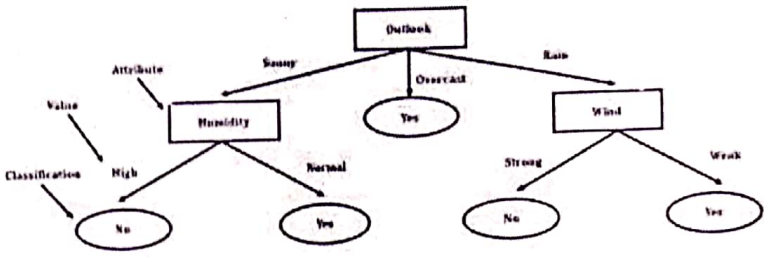


Figure 8.1: Decision tree for tennis play.

Q2. Metrics for Evaluating Classifier Performance. **[5M | May16]**

Ans:

Metrics for Evaluating Classifier Performance:

1. **Sensitivity:** Sensitivity is defined as **True Positive Recognition Rate** which is the proportion of positive tuples that are correctly identified.

   Sensitivity = TP/P

2. **Specificity:** Specificity is defined as **True Negative Recognition Rate** which is the proportion of negative tuples that are correctly identified.

   Specificity = TN/N

3. **Classifier Accuracy:** It is percentage of test set tuples that are correctly classified.

   Accuracy = (TP + TN) / (P + N)

4. **Error Rate:** It is percentage of error made over the whole set of instances used.

   Error Rate = 1 - Accuracy

5. **Precision:** It is percentage of tuples which are correctly classified as positive are actual positive. It is the measure of exactness.

   Precision = $\frac{|TP|}{|TP|+|FP|}$

6. **Recall:** It is percentage of positive tuples which the classifier labelled as positive. It is a measure of completeness.

   $$\text{Recall} = \frac{|TP|}{|TP|+|FN|}$$

7. **F Measures:** It is Harmonic mean of precision and recall

   $$F = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Note:

TP: Class Members which are classified as class members.

TN: Class Non-Members which are classified as class non-members.

FP: Class Non-Members which are classified as class members.

FN: Class Members which are classified as class non-members.

P: Number of positive tuples.

N: Number of negative tuples.

Q3. Why Naïve Bayesian Classification is called "naive"? Briefly outline the major ideas of naive Bayesian Classification.

Ans: **[10M | Dec16 & May18]**

NAÏVE BAYESIAN CLASSIFICATION:

1. Naïve Bayesian Classification is based on **Bayes Theorem**.
2. Bayesian classifiers are the **statistical classifiers**.
3. Naïve Bayesian Classification is referred as **Naïve** because it makes the assumption that each of its inputs are independent of each other, an assumption which rarely holds true.
4. For **example**, a fruit may be considered to be an apple if it is red, round, and about 4" in diameter.
5. A Naive Bayes Classifier considers all of these properties to independently contribute to the probability that this fruit is an apple.
6. This assumption is made to reduce computational costs, and hence is considered naïve.

BAYES THEOREM:

1. It is also known as **Bayes Rule**.
2. It is used to find **conditional probabilities**.
3. **Bayes Theorem:** P(H|X) = P(X|H) P(H) / P(X)
4. An initial probability is called as **Apriori Probability** which we get before any additional information is obtained.
5. A probability is called as **Posterior Probability** which we get after any additional information is obtained.
6. P (H|X) is Posterior Probability of H and P (X|H) is Posterior Probability of X.
7. P (H) is Apriori Probability of H and P (X) is Apriori Probability of X.

MAJOR IDEAS OF NAIVE BAYESIAN CLASSIFICATION:

1. The major idea behind naive Bayesian classification is to try and classify data by maximizing P (X | Ci) P (Ci) (where i is an index of the class) using the **Bayes' theorem** of posterior probability.

2. In general, we are given a set of unknown data tuples, where each tuple is represented by an 'n' dimensional vector.
3. X = (x1, x2. . . xn) depicting 'n' measurements made on the tuple from 'n' attributes, respectively A1,A2, .., An.
4. We are also given a set of 'm' classes, C1,C2, . . .Cm.
5. Using Bayes theorem, the naive Bayesian classifier calculates the **posterior probability** of each class conditioned on X.
6. X is assigned the class label of the class with the maximum posterior probability conditioned on X.
7. Therefore, we try to maximize P (Ci | X) = P (X | Ci) P (Ci) / P (X). However, since P(X) is constant for all classes, only P (X | Ci) P (Ci) need be maximized.
8. If the class prior probabilities are not known, then it is commonly assumed that the classes are equally likely, i.e. P (C1) = P (C2) = · · · = P (Cm), and we would therefore maximize P (X | Ci).
9. Otherwise, we maximize P (X | Ci) P (Ci).
10. The class prior probabilities may be estimated by P (Ci) = s_i s, where s_i is the number of training tuples of class Ci, and s is the total number of training tuples.
11. In order to reduce computation in evaluating P (X | Ci), the naive assumption of class conditional independence is made.
12. This presumes that the values of the attributes are conditionally independent of one another.
13. If A_k is a categorical attribute then P (x_k | Ci) is equal to the number of training tuples in Ci that have x_k as the value for that attribute, divided by the total number of training tuples in Ci.
14. If A_k is a continuous attribute then P (x_k |Ci) can be calculated using a **Gaussian density function**.

Q4. Define linear, non-linear and multiple regressions.

Ans: **[5M | Dec16]**

REGRESSION:

1. Regression is the method for **prediction** in data mining.
2. Regression shows a relationship between the average values of two variables.
3. Thus regression is very useful in estimating and predicting the average value of one variable for a given value of other variable.
4. The estimate or prediction may be made with the help of a regression line.
5. Regression may be used to determine for e.g. price of commodity, interest rates etc.

TYPES OF REGRESSION:

I) Linear Regression:

1. If the regression curve is a **straight line** then there is a linear regression between two variables.
2. The relationship between dependent and independent variable is described by straight line and it has only one independent variable.
   a. $Y = \alpha + \beta X$
   b. Where Y is dependent variable and X is independent variable and α, β are parameters.

III) Non-Linear Regression:

1. If the curve of regression is not a straight line then it is called as non-linear regression.
2. Regressions tries to find the mathematical relationship between variables, if it gives a curved line then it is a non-linear regression.
3. It is also known as **Curvilinear Regression**.

IIII) Multiple Regression:

1. Multiple Regression is given by following formula
   $Y = \alpha + \beta_1 X_1 + \beta_2 X_2$
2. Multiple regression includes more than one predictor variable.

Q5. A simple example from the stock market involving only discrete ranges has profit as categorical attribute, with values {Up, Down} and the training data set is given below.

Age	Competition	Type	Profit
Old	Yes	Software	Down
Old	No	Software	Down
Old	No	Hardware	Down
Mid	Yes	Software	Down
Mid	Yes	Hardware	Down
Mid	No	Hardware	Up
Mid	No	Software	Up
New	Yes	Software	Up
New	No	Hardware	Up
New	No	Software	Up

Apply decision tree algorithm and show the generated rules.

[10M | May16 & May18]

Ans:

DECISION TREE BASED CLASSIFICATION:

Refer Q1.

DECISION TREE FOR ABOVE EXAMPLE:

Figure 8.2 shows the decision tree for stock market case.

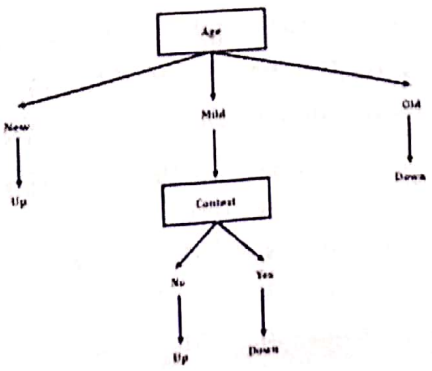


Figure 8.2: Decision tree for stock market case.

RULES:

1. IF Age = New THEN Profit = Up.
2. IF Age = Mild and Contest = No THEN Profit = Up.
3. IF Age = Mild and Contest = Yes THEN Profit = Down.
4. IF Age = Down THEN Profit = Down.

Q6. Define Classification. Discuss the issues in Classification. A simple example from the stock market involving only discrete ranges has profit as categorical attribute, with values { Up, Down} and the training data is:

Age	Competition	Type	Profit
Old	Yes	Software	Down
Old	No	Software	Down
Old	No	Hardware	Down
Mid	Yes	Software	Down
Mid	Yes	Hardware	Down
Mid	No	Hardware	Up
Mid	No	Software	Up
New	Yes	Software	Up
New	No	Hardware	Up
New	No	Software	Up

Apply decision tree algorithm and show the generated rules.

Ans: **[10M | Dec17]**

CLASSIFICATION:

1. Classification is the form of **data analysis**.
2. Classification constructs classification model based on **training data set**.
3. Using this model it classifies the new data.
4. Classification models **predict categorical class labels**.
5. For example, we can build a classification model to categorize bank loan applications as either safe or risky.

ISSUES IN CLASSIFICATION:

The major issue is preparing the data for classification. Preparing the data involves the following activities:

1. **Data Cleaning**: Data cleaning involves removing the noise and treatment of missing values. The noise is removed by applying smoothing techniques and the problem of missing values is solved by replacing a missing value with most commonly occurring value for that attribute.
2. **Relevance Analysis**: Database may also have the irrelevant attributes. Correlation analysis is used to know whether any two given attributes are related.
3. **Data Transformation and reduction**: The data can be transformed by any of the following methods.
    a. **Normalization**: Normalization involves scaling all values for given attribute in order to make them fall within a small specified range.
    b. **Generalization**: The data can also be transformed by generalizing it to the higher concept.

DECISION TREE BASED CLASSIFICATION:

Refer Q1.

DECISION TREE FOR ABOVE EXAMPLE:

Refer Q5.

Q7. Apply the Naive Bayes classifier algorithm for buys computer classification and classify the tuple X = (age = "young", income = "medium", student = "yes" and credit-rating = "fair")

Id	Age	Income	Student	Credit-Rating	Buys Computers
1	Young	High	No	Fair	No
2	Young	High	No	Good	No
3	Middle	High	No	Fair	Yes
4	Old	Medium	No	Fair	Yes
5	Old	Low	Yes	Fair	Yes
6	Old	Low	Yes	Good	No
7	Middle	Low	Yes	Good	Yes
8	Young	Medium	No	Fair	No
9	Young	Low	Yes	Fair	Yes
10	Old	Medium	Yes	Fair	Yes
11	Young	Medium	Yes	Good	Yes
12	Middle	Medium	No	Good	Yes
13	Middle	High	Yes	Fair	Yes
14	Old	Medium	No	Good	No

Ans:

[10M | May17]

BAYES THEOREM:

1. It is also known as **Bayes Rule**.
2. It is used to find **conditional probabilities**.
3. **Bayes Theorem:** P(H|X) = P(X|H) P(H) / P(X)
4. An initial probability is called as **Apriori Probability** which we get before any additional information is obtained.
5. A probability is called as **Posterior Probability** which we get after any additional information is obtained.
6. P (H|X) is Posterior Probability of H and P (X|H) is Posterior Probability of X.
7. P (H) is Apriori Probability of H and P (X) is Apricri Probability of X.

NAÏVE BAYES CLASSIFICATION FOR ABOVE EXAMPLE:

The unknown sample is X = (age = "young", income = "medium", student = "yes" and credit-rating = "fair")

Age	
P (Young | Yes) = 2/9	P (Young | No) = 3/5

P (Middle | Yes) = 4/9	P (Middle | No) = 0
P (Old | Yes) = 3/9	P (Old | No) = 2/5
Income	
P (High | Yes) = 2/9	P (High | No) = 2/5
P (Medium | Yes) = 4/9	P (Medium | No) = 2/5
P (Low | Yes) = 3/9	P (Low | No) = 1/5
Student	
P (No | Yes) = 3/9	P (No | No) = 4/5
P (Yes | Yes) = 6/9	P (Yes | No) = 1/5
Credit Rating	
P (Fair | Yes) = 6/9	P (Fair | No) = 2/5
P (Good | Yes) = 3/9	P (Good | No) = 3/5

Therefore,

P (Yes) = 9/14 = 0.643 and

P (No) = 5/14 = 0.357

So,

P(Ci | X) ≥ P(Cj | X)

By Bayes theorem,

$$\frac{P(X|Ci) * P(Ci)}{P(X)} > \frac{P(X|Cj) * P(Cj)}{P(X)}$$

Now by Multiplying P(X) on both sides we get;

P (X | C_i) * P (C_i) > P (X | C_j) * P (C_j) ..... Equation (1)

Naive Bayesian Classifier depicts that effect of an attribute value on a given class is Independent of the values of the other attributes.

$$P(X|Ci) = \prod_{K=1}^{n} P(Xk|Ci)$$

Sample X = (age="young", income="medium", student="yes" and credit-rating="fair")

For class C1 (buys_computer = "Yes"):

P (X_1|C_1) = P (age = "Young" | buys_computer = "Yes") = 2/9

P (X_2|C_1) = P (income = "medium" | buys_computer = "yes") = 4/9

P (X_3|C_1) = P (student = "yes" | buys_computer = "yes") = 6/9

P (X_4|C_1) = P (credit_rating = "fair" | buys_computer = "yes") = 6/9

P (X|C_1) = 2/9 * 4/9 * 6/9 * 6/9 = 0.0438

P (X|C_1) * P (C_1) = 0.0438 * 9/14 = 0.0281 ..... Equation (2)

For class C2 (buys_computer = "No"):

P (X_1|C_2) = P (age = "Young" | buys_computer = "No") = 3/5

P (X_2|C_2) = P (income = "medium" | buys_computer = "No") = 2/5

P (X_3|C_2) = P (student = "yes" | buys_computer = "No") = 1/5

$P(X_4|C_2)$ = P (credit_rating = "fair" | buys_computer = "No") = 2/5

$P(X|C_2)$ = 3/5 * 2/5 * 1/5 * 2/5 = 0.0192

$P(X|C_2) * P(C_2)$ = 0.0192 * 5/14 = 0.0068 ..... Equation (3)

From equation 1, 2 and 3 we get,

$P(X|C_1) * P(C_1) > P(X|C_2) * P(C_2)$

Therefore, Sample X belongs to class ("buys_computer = yes")

Q8. Apply the Naive Bayes classifier algorithm to classify an unknown sample X (outlook = sunny, temperature = cool, humidity = high, windy = false) The sample data set is as follows:

Outlook	Temperature	Humidity	Windy	Class
Sunny	Hot	High	False	N
Sunny	Hot	High	True	N
Overcast	Hot	High	False	P
Rain	Mild	High	False	P
Rain	Cool	Normal	False	P
Rain	Cool	Normal	True	N
Overcast	Cool	Normal	True	P
Sunny	Mild	High	False	N
Sunny	Cool	Normal	False	P
Rain	Mild	Normal	False	P
Sunny	Mild	Normal	True	P
Overcast	Mild	High	True	P
Overcast	Hot	Normal	False	P
Sunny	Mild	High	True	N

Ans: **[10M | Dec18]**

NAÏVE BAYES CLASSIFICATION FOR ABOVE EXAMPLE:

Out of 14 attributes, we have 9 positive (P) attributes and 5 negative (N) attributes.

P (Yes) = 9/14

P (No) = 5/14

Outlook	
P (Sunny | P) = 2/9	P (Sunny | N) = 3/5
P (Overcast | P) = 4/9	P (Overcast | N) = 0
P (Rain | P) = 3/9	P (Rain | N) = 2/5
Temperature	
P (Hot | P) = 2/9	P (Hot | N) = 2/5
P (Mild | P) = 4/9	P (Mild | N) = 2/5
P (Cool | P) = 3/9	P (Cool | N) = 1/5

Humidity	
P (High | P) = 3/9	P (High | N) = 4/5
P (Normal | P) = 6/9	P (Normal | N) = 1/5
Wind	
P (False | P) = 6/9	P (False | N) = 2/5
P (True | P) = 3/9	P (True | N) = 3/5

Therefore,

P (P) = 9/14 = 0.643 and

P (N) = 5/14 = 0.357

Sample (X) = (outlook = sunny, temperature = cool, humidity = high, windy = false)

For class C1 (Class = P):

$P(X_1|C_1)$ = P (Outlook = "Sunny" | Class = "P") = 2/9

$P(X_2|C_1)$ = P (Temperature = "Cool" | Class = "P") = 3/9

$P(X_3|C_1)$ = P (Humidity = "High" | Class = "P") = 3/9

$P(X_4|C_1)$ = P (Windy = "False" | Class = "P") = 6/9

$P(X|C_1)$ = 2/9 * 3/9 * 3/9 * 6/9 = 0.0164

$P(X|C_1) * P(C_1)$ = 0.0164 * 9/14 = 0.0105 ..... Equation (2)

For class C2 (Class = N):

$P(X_1|C_2)$ = P (Outlook = "Sunny" | Class = "No") = 3/5

$P(X_2|C_2)$ = P (Temperature = "Cool" | Class = "No") = 1/5

$P(X_3|C_2)$ = P (Humidity = "High" | Class = "No") = 4/5

$P(X_4|C_2)$ = P (Windy = "False" | Class = "No") = 2/5

$P(X|C_2)$ = 3/5 * 1/5 * 4/5 * 2/5 = 0.0384

$P(X|C_2) * P(C_2)$ = 00384 * 5/14 = 0.0137 ..... Equation (3)

From equation 1, 2 and 3 we get,

$P(X|C_1) * P(C_1) < P(X|C_2) * P(C_2)$

Therefore, Sample X belongs to class (Class = "N")

CHAP- 9: CLUSTERING

Q1. Explain K-Means clustering algorithm? Apply K-Means Algorithms for the following data set with two clusters. Data Set = {1, 2, 6, 7, 8, 10, 15, 17, 20}

Ans:

CLUSTERING:

[10M | May16]

1. Clustering is **unsupervised learning problem**.
2. It is data mining technique used to place data elements into related groups without advance knowledge of the group definitions.
3. It is a process of portioning data objects into sub classes which are called as clusters.
4. Clustering Algorithms are used in Marketing, Biology, and Insurance etc.

K-MEANS CLUSTERING ALGORITHM:

1. K-Means Clustering is one of the **partitioning method**.
2. It is simplest **unsupervised learning algorithm**.
3. K-Means Clustering aims to partition 'n' observations into 'k' clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster.
4. This results in a partitioning of the data space.
5. K is positive integer number.

K-MEANS CLUSTERING PROCESS:

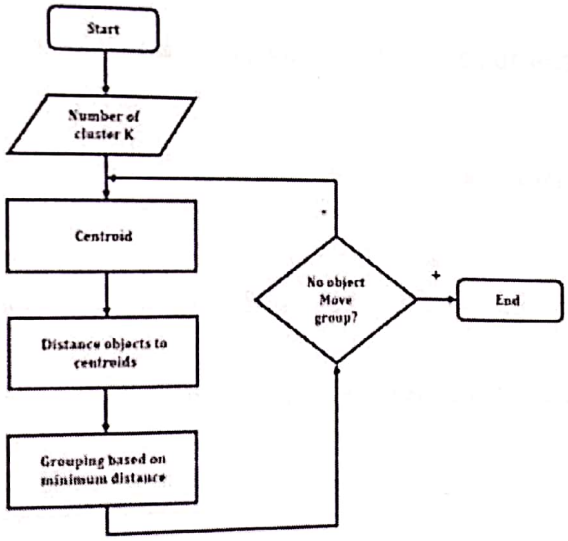


Figure 9.1: Flowchart for K-Means Clustering.

1. Figure 9.1 shows the flowchart for K-Means Clustering.
2. Define K centroids for K clusters which are generally far away from each other.
3. Then group the elements into clusters which are nearer to the centroid of that cluster.
4. After this first step, again calculate the new centroid for each cluster based on the elements of that cluster.
5. Follow the same method, and group the elements based on new centroid.
6. In every step, the centroid changes and elements move from one cluster to another
7. Do the same process till no element is moving from one cluster to another.

EXAMPLE:

Given:

Data Set = {1, 2, 6, 7, 8, 10, 15, 17, 20}

No. of clusters = 2

Solution:

1. **(Define K Centroid)**

   Consider initial two centroids for two clusters $C_1 = 6$ and $C_2 = 15$

2. **(Randomly assign data to two clusters)**

   K_1 = {1, 2, 6, 7, 8, 10}

   K_2 = {15, 17, 20}

3. **(Calculate Mean)**

   No. of clusters = 2

   Therefore K_1 = {1, 2, 6, 7, 8, 10} C_1 = Mean = 34/6 = 5.67

   K_2 = {15, 17, 20} C_2 = Mean = 52/3 = 17.33

4. **(Reassign)**

   K_1 = {1, 2, 6, 7, 8, 10}

   K_2 = {15, 17, 20}

   As no elements is moving from cluster, so the final answer is **K_1 = {1, 2, 6, 7, 8, 10} and K_2 = {15, 17, 20}**

Q2. Explain K-Means clustering algorithm? Apply K-Means algorithms for the following Data set with two clusters.

Data Set = {15,15,16,19,19,20,20,21,22,28,35,40,41,42,43,44,60,61,65}

Ans: **[10M | May17]**

K-MEANS CLUSTERING ALGORITHM:

Refer Q1.

EXAMPLE:

Given:

Data Set = {15,15,16,19,19,20,20,21,22,28,35,40,41,42,43,44,60,61,65}

No. of clusters = 2

Solution:

Initial clusters:

Centroid (C1) = 16 [16]

Centroid (C2) = 22 [22]

Iteration 1:

C1 = 15.33 [15, 15, 16]

C2 = 36.25 [19, 19, 20, 20, 21, 22, 28, 35, 40, 41, 42, 43, 44, 60, 61, 65]

Iteration 2:

C1 = 18.56 [15, 15, 16, 19, 19, 20, 20, 21, 22]

C2 = 45.90 [28, 35, 40, 41, 42, 43, 44, 60, 61, 65]

Iteration 3:

C1 = 19.50 [15, 15, 16, 19, 19, 20, 20, 21, 22, 28]

C2 = 47.89 [35, 40, 41, 42, 43, 44, 60, 61, 65]

Iteration 4:

C1 = 19.50 [15, 15, 16, 19, 19, 20, 20, 21, 22, 28]

C2 = 47.89 [35, 40, 41, 42, 43, 44, 60, 61, 65]

1. No change between iterations 3 and 4 has been noted.
2. By using clustering, 2 groups have been identified 15-28 and 35-65.
3. The initial choice of centroids can affect the output clusters, so the algorithm is often run multiple times with different starting conditions in order to get a fair view of what the clusters should be.

Q3. DBSCAN

Ans:

DBSCAN: **[5M | May17 & Dec17]**

1. DBSCAN Stands for **Density-based spatial clustering of applications with noise**.
2. It is a **data clustering algorithm**.
3. DBSCAN is designed to discover clusters of arbitrary shape.

ALGORITHMIC STEPS FOR DBSCAN CLUSTERING:

1. Let X = {x1, x2, x3, ..., xn} be the set of data points.
2. DBSCAN requires two parameters: ε (eps) and the minimum number of points required to form a cluster (minPts).
   a. Start with an arbitrary starting point that has not been visited.
   b. Extract the neighborhood of this point using 'ε'.
   c. If there are sufficient neighborhood around this point then clustering process starts and point is marked as visited else this point is labeled as noise.
   d. If a point is found to be a part of the cluster then its 'ε' neighborhood is also the part of the cluster and the above procedure from step 2 is repeated for all 'ε' neighborhood points. This is repeated until all points in the cluster is determined.
   e. A new unvisited point is retrieved and processed, leading to the discovery of a further cluster or noise.
   f. This process continues until all points are marked as visited.

ADVANTAGES:

1. Does not require a-priori specification of number of clusters.
2. Able to identify noise data while clustering.
3. DBSCAN algorithm is able to find arbitrarily size and arbitrarily shaped clusters.

DISADVANTAGES:

1. DBSCAN algorithm fails in case of varying density clusters.
2. Fails in case of neck type of dataset.
3. Does not work well in case of high dimensional data.

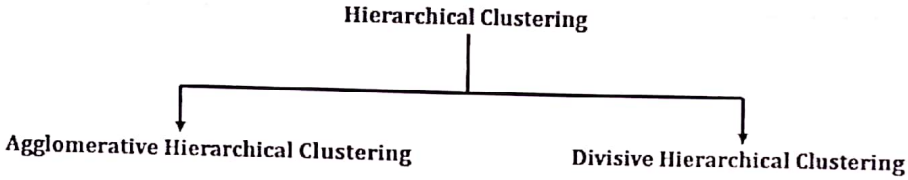
Q4. Hierarchical Clustering

Ans:

[5M | May17 & May18]

HIERARCHICAL CLUSTERING:

1. Hierarchical clustering is also called as **hierarchical cluster analysis**.
2. It is a method of cluster analysis which seeks to build a hierarchy of clusters.
3. Hierarchical clustering involves creating clusters that have a predetermined ordering from top to bottom.
4. For example, all files and folders on the hard disk are organized in a hierarchy.
5. There are two types of **hierarchical clustering, Divisive and Agglomerative**.



AGGLOMERATIVE HIERARCHICAL CLUSTERING:

1. Agglomerative Algorithm is used in **Hierarchical based clustering**.
2. It is also known as **AGNES (agglomerative nesting)**.
3. This approach is also known as the **bottom-up approach**.
4. In this, we start with each object forming a separate group.
5. It keeps on merging the objects or groups that are close to one another.
6. It keep on doing so until all of the groups are merged into one or until the termination condition holds.

DIVISIVE HIERARCHICAL CLUSTERING:

1. It is just the **reverse of Agglomerative Hierarchical approach**.
2. This approach is also known as the **top-down approach**.
3. In this, we start with all of the objects in the same cluster.
4. In the continuous iteration, a cluster is split up into smaller clusters.
5. Process is repeated until each object in one cluster is split up or the termination condition holds.
6. This method is rigid, i.e., once a merging or splitting is done, it can never be undone.

Q5. What is Clustering Techniques? Discuss the Agglomerative algorithm with the following data and plot a Dendrogram using single link approach. The table below comprises sample data items indicting the distance between the elements.

Item	E	A	C	B	D
E	0	1	2	2	3
A	1	0	2	5	3
C	2	2	0	1	6
B	2	5	1	0	3
D	3	3	6	3	0

Ans:

[10M | Dec16]

CLUSTERING:

1. Clustering is **unsupervised learning problem**.
2. It is data mining technique used to place data elements into related groups without advance knowledge of the group definitions.
3. It is a process of portioning data objects into sub classes which are called as clusters.
4. Clustering Algorithms are used in **Marketing, Biology, and Insurance** etc.

CLUSTERING TECHNIQUES:

Clustering Techniques can be classified into the following categories:

I) Partitioning Method:

1. In Partitioning based approach, various partition is created.
2. Each partition represents a **cluster**.

II) Hierarchical Method:

1. This method creates a **hierarchical decomposition** of the given dataset of objects.
2. There are two approaches – Agglomerative approach and Divisive approach.

III) Density - Based Method:

1. This method is based on the **notion of density**.
2. The basic idea is to continue growing the given cluster as long as the density in the neighborhood exceeds some threshold.

IV) Grid - Based Method:

1. In this, the various objects together **form a grid**.
2. The object space is quantized into finite number of cells that form a grid structure.

V) Model-Based Method:

1. In this method, a model is hypothesized for each cluster to find the best fit of data for a given model.
2. This method uses **density function** to locate clusters.

VI) Constraint-based Method:

1. In this method, the clustering is performed by the incorporation of **constraints**.
2. Constraints can be user-oriented or application-oriented.

AGGLOMERATIVE ALGORITHM:

1. Agglomerative Algorithm is used in **Hierarchical based clustering**.
2. It is also known as **AGNES (agglomerative nesting)**.
3. This approach is also known as the **bottom-up approach**.
4. In this, we start with each object forming a separate group.
5. It keeps on merging the objects or groups that are close to one another.
6. It keep on doing so until all of the groups are merged into one or until the termination condition holds.
7. A Hierarchical Agglomerative Clustering is typically visualized as a Dendrogram as shown in figure 9.2.
8. Dendrogram is tree like structure used to illustrate hierarchical clustering technique.

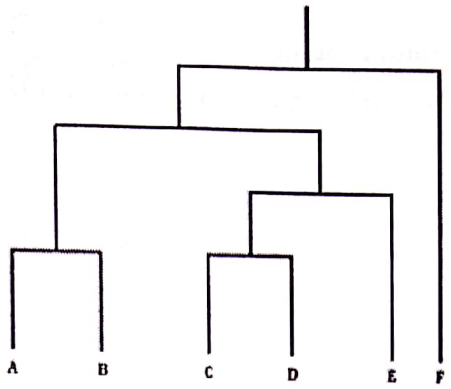


Figure 9.2: Dendrogram.

EXAMPLE:

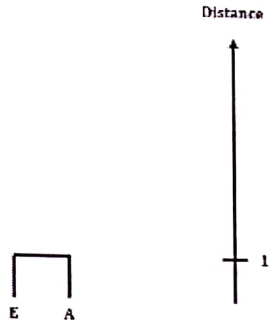
Given:

Distance Matrix:

Item	E	A	C	B	D
E	0				
A	1	0			
C	2	2	0		
B	2	5	1	0	
D	3	3	6	3	0

Step - 1:

From above given distance matrix, E and A clusters has minimum distance i.e. 1, so merge them together to form cluster (E, A)



Distance Matrix:

Dist ((E A), C) = MIN (Dist (E, C), Dist (A, C))
= MIN (2, 2) = 2

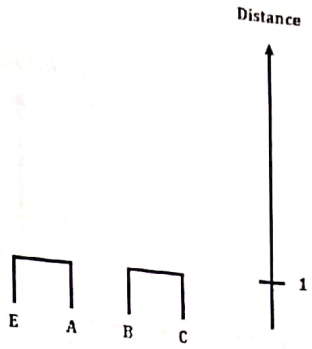
Dist ((E, A), B) = MIN (Dist (E, B), Dist (A, B))
= MIN (2, 5) = 2

Dist ((E, A), D) = MIN (Dist (E, D), Dist (A, D))
= MIN (3, 3) = 3

Item	E, A	C	B	D
E, A	0			
C	2	0		
B	2	1	0	
D	3	6	3	0

Step - 2:

Consider the distance matrix obtained in step 1. Since B, C distance is minimum, we combine B and C.



Distance Matrix:

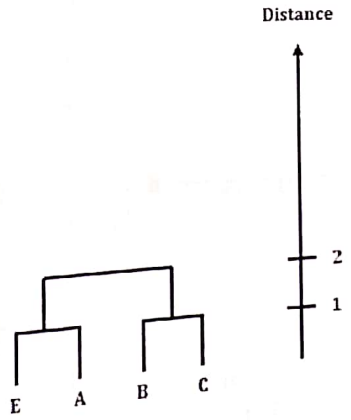
Dist ((B, C), (E, A)) = MIN (Dist (B, E), Dist (B, A), Dist (C, E), Dist (C, A))
= MIN (2, 5, 2, 2) = 2

Dist ((B, C), D) = MIN (Dist (B, D), Dist (C, D))
= MIN (3, 6) = 3

Item	E, A	B, C	D
E, A	0		
B, C	2	0	
D	3	3	0

Step - 3:

Consider the distance matrix obtained in step 2. Since (E, A) and (B, C) distance is minimum, we combine them.



Distance Matrix:

Dist ((E, A), (B, C)) = MIN (Dist (E, B), Dist (E, C), Dist (A, B), Dist (A, C)
= MIN (2, 2, 5, 2) = 2

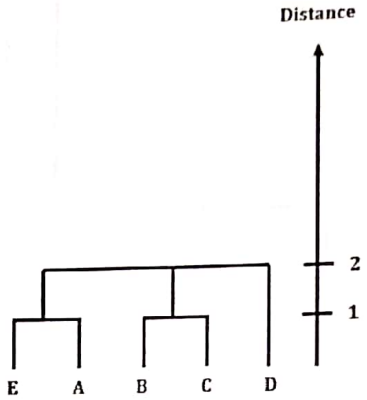
Dist ((B, C), D) = MIN (Dist (B, D), Dist (C, D))
= MIN (3, 6) = 3

Item	E, A, B, C	D
E, A, B, C	0	
D	2	0

Step – 4:

Finally we combine D with (E, A, B, C)

Final Dendrogram:

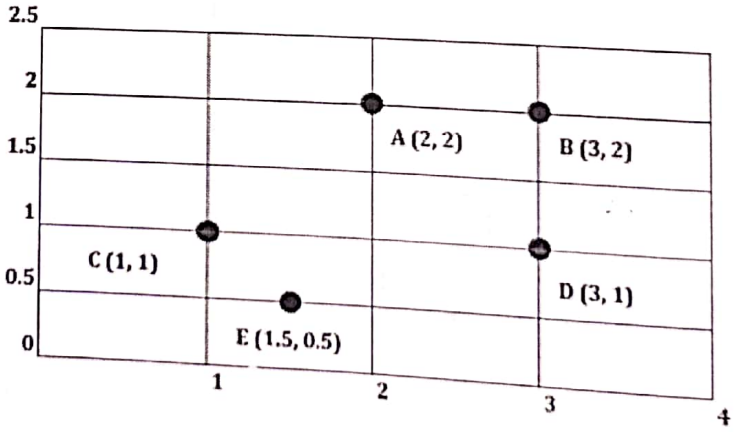


Q6. Consider the data given below. Create adjacency matrix. Apply single link algorithm to cluster the given data set and draw the Dendrogram.

Object	Attribute 1(X)	Attribute 2(Y)
A	2	2
B	3	2
C	1	1
D	3	1
E	1.5	0.5

Ans: [10M | Dec17]

CLUSTER GRAPH:



ADJACENT MATRIX:

For simplicity we can find the adjacent matrix which gives distances of all object from each other. Using Euclidean Distance we have,

$D(i, j) = \sqrt{|X2 - X1|^2 + |Y2 - Y1|^2}$

$D(A, B) = \sqrt{(2-3)^2 + (2-2)^2} = 1$

Similarly we can compute for the rest.

	A	B	C	D	E
A	0				
B	1	0			
C	1.41	2.24	0		
D	1.41	1	2	0	
E	1.58	2.12	0.71	1.58	0

Single Link:

Step-1: Since C, E is minimum we can combine clusters C, E.

Dist ((C, E), A) = MIN (Dist (C, A), Dist (C, E))
= MIN (1.41, 0) = **1.41**

Dist ((C, E), B) = MIN (Dist (C, B), Dist (E, B))
= MIN (2.24, 2.12) = **2.12**

Dist ((C, E), D) = MIN (Dist (C, D), Dist (E, D))
= MIN (0, 1.58) = **1.58**

	A	B	(C, E)	D
A	0			
B	1	0		
(C, E)	1.41	2.12	0	
D	1.41	1	1.58	0

Step-2: Now A and B is having minimum value therefore we merge these two clusters.

Dist ((C, E), (A, B)) = MIN (Dist (C, A), Dist (C, B), Dist (E, A), Dist (E, B))
= MIN (1.41, 2.24, 1.58, 2.12) = **1.41**

Dist ((A, B), D) = MIN (Dist (A, D), Dist (B, D))
= MIN (1.41, 1) = **1**

Dist ((C, E), D) = MIN (Dist (C, D), Dist (E, D))
= MIN (2, 1.58) = **1.58**

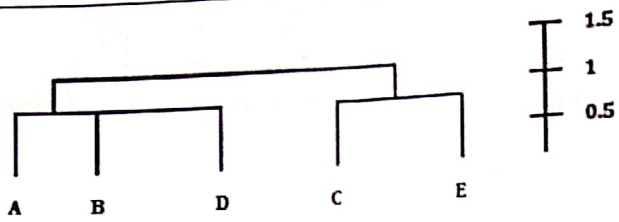
	(A, B)	(C, E)	D
(A, B)	0		
(C, E)	1.41	0	
D	1	1.58	0

Step-3: Cluster (A, B) and D can be merged together as they are having minimum distance value.

	(A, B, D)	(C, E)
(A, 3, D)	0	
(C, E)	1.41	0

Step-4: In the last step there are only two clusters to be combined they are (A, B, D) and (C, E).

Now the final Dendrogram is as shown below.



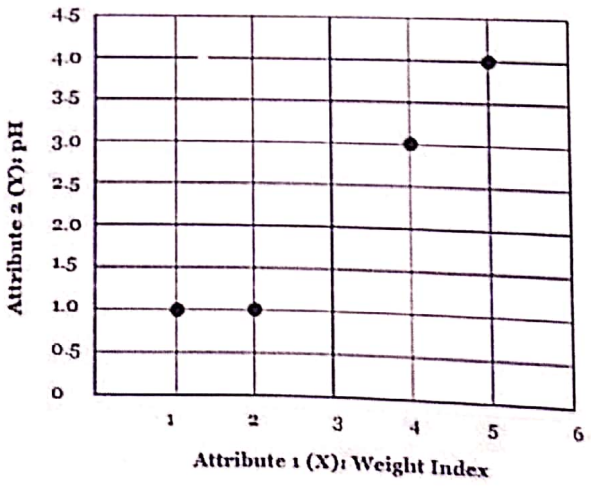
Q7. **Find clusters using k-means clustering algorithm if we have several objects (4 types of medicines) and each object have two attributes or features as shown in the table below. The goal is to group these objects into k = 2 group of medicine based on the two features (pH and weight index)**

Object	Attribute 1 (X) Weight Index	Attribute 2 (Y) pH
Medicine A	1	1
Medicine B	2	1
Medicine C	4	3
Medicine D	5	4

Ans: **[10M | May18 & Dec18]**

ITERATION-0, DETERMINE CENTROIDS:

1. Given objects belong to two groups of medicine (cluster 1 and cluster 2).
2. The problem is to determine which medicines belong to cluster 1 and which medicines belong to the other cluster.
3. Each medicine represents one point with two features (X, Y) that we can represent it as coordinate in a feature space as shown in the figure below.
4. Each medicine represents one point with two components coordinate.



ITERATION-0, INITIAL VALUE OF CENTROIDS:

1. Suppose we use medicine A and medicine B as the first centroids.
2. Let C1 and C2 denote the coordinate of the centroids, then C1 = (1, 1) and C2 = (2, 1)

ITERATION-0, OBJECTS-CENTROIDS DISTANCE:

1. We calculate the distance between cluster centroid to each object.
2. Let us use Euclidean distance, then we have distance matrix at iteration 0 is

$$D^0 = \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 1 & 0 & 2.83 & 4.24 \\ A & B & C & D \end{bmatrix} \begin{matrix} c_1 = (1,1) & group-1 \\ c_2 = (2,1) & group-2 \end{matrix}$$

$$\begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \begin{matrix} X \\ Y \end{matrix}$$

3. Each column in the distance matrix symbolizes the object.
4. The first row of the distance matrix corresponds to the distance of each object to the first centroid and the second row is the distance of each object to the second centroid.
5. Distance from medicine C = (4, 3) to the first centroid is C1 = (1, 1) = $\sqrt{(4-1)^2 + (3-1)^2}$ = 3.61
6. Distance to the second centroid is C2 = (2, 1) = $\sqrt{(4-2)^2 + (3-1)^2}$ = 2.83

ITERATION-0, OBJECTS CLUSTERING:

1. We assign each object based on the minimum distance.
2. Thus, medicine A is assigned to group 1, medicine B to group 2, medicine C to group 2 and medicine D to group 2.
3. The element of Group matrix below is 1 if and only if the object is assigned to that group.

$$G^0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ A & B & C & D \end{bmatrix} \begin{matrix} group-1 \\ group-2 \end{matrix}$$

ITERATION-1, DETERMINE CENTROIDS:

1. Knowing the members of each group, now we compute the new centroid of each group based on these new memberships.
2. Group 1 only has one member thus the centroid remains in C_1 = (1, 1).
3. Group 2 now has three members, thus the centroid is the average coordinate among the three members:

   C_2 = ([2 + 4 + 5]/3, [1 + 3 + 4]/3) = (11/3, 8/3)

ITERATION-1, OBJECTS-CENTROIDS DISTANCES:

1. The next step is to compute the distance of all objects to the new centroids.
2. We have distance matrix at iteration 1 is:

$$D^1 = \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 3.14 & 2.36 & 0.47 & 1.89 \\ A & B & C & D \end{bmatrix} \begin{matrix} c_1 = (1,1) & group-1 \\ c_2 = (\frac{11}{3}, \frac{8}{3}) & group-2 \end{matrix}$$

$$\begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \begin{matrix} X \\ Y \end{matrix}$$

Iteration-1, Objects clustering:

1. Now we assign each object based on the minimum distance.
2. Based on the new distance matrix, we move the medicine B to Group 1 while all the other objects remain
3. The Group matrix is shown below:

$$G^1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ A & B & C & D \end{bmatrix} \begin{matrix} group-1 \\ group-2 \end{matrix}$$

ITERATION 2, DETERMINE CENTROIDS:

1. Now we calculate the new centroids coordinate based on the clustering of previous iteration.
2. Group1 and group 2 both has two members, thus the new centroids are:

$$C_1 = ([1 + 2]/2, [1 + 1]/2) = (1.5, 1)$$

$$C_2 = ([4 + 5]/2, [3 + 4]/2) = (4.5, 3.5)$$

ITERATION-2, OBJECTS-CENTROIDS DISTANCES:

Repeat object centroid step again, we have new distance matrix at iteration 2 as:

$$D^2 = \begin{bmatrix} 0.5 & 0.5 & 3.20 & 4.61 \\ 4.30 & 3.54 & 0.71 & 0.71 \end{bmatrix} \begin{matrix} c_1 = (1\frac{1}{2}, 1) & group-1 \\ c_2 = (4\frac{1}{2}, 3\frac{1}{2}) & group-2 \end{matrix}$$

$$\begin{matrix} A & B & C & D \end{matrix}$$

$$\begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \begin{matrix} X \\ Y \end{matrix}$$

ITERATION-2, OBJECTS CLUSTERING:

Again, we assign each object based on the minimum distance.

$$G^2 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{matrix} group-1 \\ group-2 \end{matrix}$$

$$\begin{matrix} A & B & C & D \end{matrix}$$

1. We obtain result that $G^2 = G^1$. Comparing the grouping of last iteration and this iteration reveals that the objects does not move group anymore.
2. Thus, the computation of the k-mean clustering has reached its stability and no more iteration is needed.
3. We get the final grouping as the results:

Object	Attribute 1 (X) Weight Index	Attribute 2 (Y) pH	Group
Medicine A	1	1	1
Medicine B	2	1	1
Medicine C	4	3	2
Medicine D	5	4	2

CHAP - 10: MINING FREQUENT PATTERN AND ASSOCIATION RULE

Q1. FP Tree.

Ans:

[5M | May16, May17, Dec17 & Dec18]

FP TREE:

1. FP Tree Stands for **Frequent Pattern Tree**.
2. An FP Tree is a tree structure which consists of one root labelled as "null" and Set of item-prefix sub trees.
3. Each node in the item-prefix sub tree consists of three fields:
   a. Item-name.
   b. Count.
   c. Node-link.
4. FP Tree is a compact structure that stores **quantitative information** about frequent patterns in a database.
5. The size of FP Tree is bounded by size of database.
6. But due to frequent pattern sharing, the size of the tree is usually much smaller than its original database.
7. Figure 10.1 shows the example of an FP Tree.

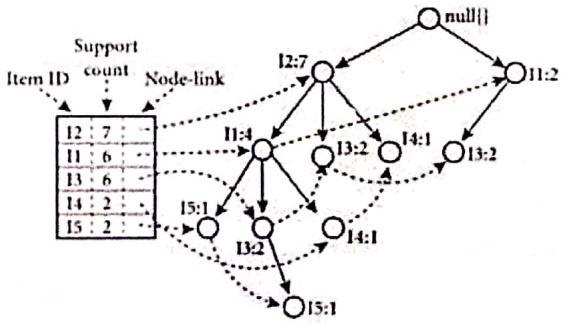


Figure 10.1: Example of an FP Tree.

ADVANTAGES:

1. FP Tree is much faster than Apriori.
2. It provides compresses data set.

DISADVANTAGES:

1. FP Tree may not fit in memory.
2. It is expensive to build.

Q2. Multilevel & Multidimensional Association Rule.

Ans:

[5M | May16]

MULTILEVEL ASSOCIATION RULE:

1. Rules which combine association with hierarchy of concepts are called as **Multilevel Association Rules**.
2. In multilevel association rule, items are always in the **form of hierarchy**.

3. Items which are placed at leaf nodes has lower support.
4. An item can be generalized or specialized as per the described hierarchy of that item.
5. Figure 10.2 shows the example of multilevel association rule.

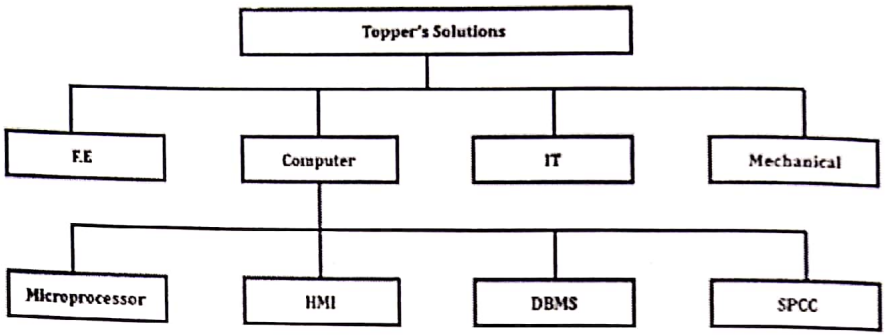


Figure 10.2: Example of multilevel association rule.

MULTIDIMENSIONAL ASSOCIATION RULE:

1. Rules which combine association with multiple dimensions are called as **Multidimensional Association Rules.**
2. In this, Rule contains two or more dimensions or predicates.
3. There are two types; Inter dimension association rules and hybrid dimension association rules.
   a. **Inter dimension association rules:** This rule does not have any repeated predicate. For Example: **Gender (X, "Male") ∧ Salary (X, "High") → Buys (X, "Computer")**
   b. **Hybrid dimension association rules:** This rule have many occurrences of same predicate i.e. buys. **Gender (X, "Male") ∧ Buys (X, "TV") → Buys (X, "DVD")**

Q3. Discuss Association Rule Mining and Apriori Algorithm

Q4. Discuss Association Rule Mining and Apriori Algorithm. Apply AR Mining to find all frequent item sets and association rules for the following dataset:

Minimum Support Count = 2

Minimum Confidence = 70%

Transaction_ID	Items
100	1, 2, 5
200	2, 4
300	2, 3
400	1, 2, 4
500	1, 3
600	1, 3
700	1, 3, 2, 5
800	1, 3
900	1, 2, 3

Ans:

ASSOCIATION RULE MINING: **[5 - 10M | May16 & Dec17]**

1. Association rule mining is the **data mining process**.
2. It is used to find the **rules that govern the associations**.

3. Association rule mining is a procedure which is meant to find frequent patterns, correlations and associations in various kinds of databases.
4. Databases can be relational databases, transactional databases, and other forms of data repositories.
5. Association Rule Mining are of two types; multilevel association rule and Multidimensional association rule.

I) **Multilevel Association Rule:**

Refer Q2.

II) **Multidimensional Association Rule:**

Refer Q2.

APRIORI ALGORITHM:

1. Apriori Algorithm is one of **frequent Itemset mining method**.
2. It is used to solve the frequent item set problem.
3. Apriori Algorithm uses a **"Bottom Up" Approach**.
4. Apriori Algorithm analyzes a data set to determine which combinations of items can occur together frequently.

Advantages:

1. Easy to implement.
2. Apriori Algorithm can be easily parallelized.

Disadvantages:

1. Performance is low.
2. It requires many database scans.

Example:

Given:

Minimum Support Count = 2

Minimum Confidence = 70%

Transaction_ID	Items
100	1, 2, 5
200	2, 4
300	2, 3
400	1, 2, 4
500	1, 3
600	1, 3
700	1, 3, 2, 5
800	1, 3
900	1, 2, 3

Solution:

Step-1:

Scan the transaction database and find the count of Items.

Candidate List = {1, 2, 3, 4, 5}

C_1 =

Itemset	Support Count
1	7
2	6
3	6
4	2
5	2

Step-2:

Check whether each candidate item is present in at least two transactions because the given support count is 2.

L_1 =

Itemset	Support Count
1	7
2	6
3	6
4	2
5	2

Step-3:

Now generate Candidate C_2 from L_1 and find the support count for items.

C_2 =

Itemset	Support Count
1, 2	4
1, 3	5
1, 4	1
1, 5	2
2, 3	3
2, 4	2
2, 5	2
3, 4	0
3, 5	1
4, 5	0

Step-4:

Now we compare Candidate C_2 generated in step 3 with the minimum support count and prune those Itemsets which do not satisfy the minimum support count.

L_2 =

Itemset	Support Count
1, 2	4
1, 3	5
1, 5	2
2, 3	3
2, 4	2
2, 5	2

Step-5

Now generate Candidate C_3 from L_2 and find the support count for items.

C_3 =

Itemset	Support Count
1, 2, 3	2
1, 2, 5	2
1, 3, 5	1
2, 3, 4	0
2, 3, 5	1
2, 4, 5	0

Step-6:

Now we compare Candidate C_3 generated in step 5 with the minimum support count and prune those Itemsets which do not satisfy the minimum support count.

L_3 =

Itemset	Support Count
1, 2, 3	2
1, 2, 5	2

Step-7:

Frequent Itemset are {1, 2, 3} and {1, 2, 5}

Let consider the frequent Itemset {1, 2, 5}

Following are the association rules that can be generated shown below with the support and confidence.

Association Rule	Support	Confidence	Confidence %
(1, 2) → 5	2	2/4	50
(1, 5) → 2	2	2/2	100
(2, 5) → 1	2	2/2	100
1 → (2, 5)	2	2/7	29
2 → (1, 5)	2	2/6	33
5 → (1, 2)	2	2/2	100

Minimum Confidence threshold is 70 %. So the following rules are considered as output, as they are strong rules.

Rules	Confidence
1 ∧ 5 → 2	100 %
2 ∧ 5 → 1	100 %
5 → 1 ∧ 2	100 %

Q5. **A database has five transactions. Let min-support = 60% and min-confidence = 80%. Find all find frequent item sets by using Apriori Algorithm. T_ID is the transaction ID**

T_ID	Items Bought
T-1000	M, O, N, K, E, Y
T-1001	D, O, N, K, E, Y
T-1002	M, A, K, E
T-1003	M, U, C, K, Y
T-1004	C, O, O, K, E

Ans: [10M | Dec16]

Given:

Minimum Support = 60 %

Minimum Confidence = 80%

T_ID	Items Bought
T-1000	M, O, N, K, E, Y
T-1001	D, O, N, K, E, Y
T-1002	M, A, K, E
T-1003	M, U, C, K, Y
T-1004	C, O, O, K, E

Solution:

Step-1:

Scan the transaction database and find the count of Items.

Candidate List = {A, C, D, E, K, M, N, O, U, Y}

C_1 =

Itemset	Support Count
A	1
C	2
D	1
E	4
K	5
M	3
N	2
O	4
U	1
Y	3

Step-2:

Now compare candidate support count with minimum support count (i.e. 60%)

L_1 =

Itemset	Support Count
E	4
K	5
M	3
O	4
Y	3

Step-3:

Now generate Candidate C_2 from L_1 and find the support count for items.

C_2 =

Itemset	Support Count
E, K	4
E, M	2
E, O	3
E, Y	2
K, M	3
K, O	3
K, Y	3
M, O	1
M, Y	2
O, Y	2

Step-4:

Now we compare Candidate C_2 generated in step 3 with the minimum support count and prune those Itemsets which do not satisfy the minimum support count (i.e. 60 %).

L_2 =

Itemset	Support Count
E, K	4
E, O	3
K, M	3
K, O	3
K, Y	3

Step-5

Now generate Candidate C_3 from L_2 and find the support count for items.

C_3 =

Itemset	Support Count
E, K, M	2
E, K, O	3
E, K, Y	2
E, O, Y	2
K, M, O	1
K, M, Y	1

Step-6:

Now we compare Candidate C_3 generated in step 5 with the minimum support count and prune those Itemsets which do not satisfy the minimum support count (i.e. 60 %).

L_3 =

Itemset	Support Count
E, K, M	2
E, K, O	3
E, K, Y	2
E, O, Y	2

Step-7:

Frequent Itemset are {E, K, M}, {E, K, O}, {E, K, Y} and {E, O, Y}

Let consider the frequent Itemset {E, K, O}

Following are the association rules that can be generated shown below with the support and confidence.

Association Rule	Support	Confidence	Confidence %
(E ∧ K) → O	3	3/4	75
(E ∧ O) →K	3	3/3	100
(K ∧ O) → E	3	3/3	100
E → (K ∧ O)	3	3/4	75
K → (E ∧ O)	3	3/5	60
O → (E ∧ K)	3	3/4	75

Minimum Confidence threshold is 80 %. So the following rules are considered as output, as they are strong rules.

Rules	Confidence
(E ∧ O) → K	100 %
(K ∧ O) → E	100 %

Q6. A database has ten transactions. Let minimum support = 30% and minimum Confidence = 70%

(i) Find all frequent patterns using Apriori Algorithm.

(ii) List strong association rules.

Transaction ID	Items
01	A, B, C, D
02	A, B, C, D, E, G
03	A, C, G, H, K
04	B, C, D, E, K
05	D, E, F, H, L
06	A, B, C, D, L
07	B, I, E, K, L
08	A, B, D, E, K
09	A, E, F, H, L
10	B, C, D, F

Ans: **[10M | May17]**

Given:

Minimum Support = 30 %

Minimum Confidence = 70%

Solution:

Step-1 (C1):

Scan the transaction database and find the count of Items.

Candidate List = {A, B, C, D, E, F, G, H, I, K, L}

Item set	Support Count
{A}	6
{B}	7
{C}	6
{D}	7
{E}	6
{F}	3

{G}	2
{H}	3
{I}	1
{K}	4
{L}	4

Step-2 (L1):

Now compare candidate support count with minimum support count (i.e. 60%)

Minimum Support Count = $\frac{30}{100} * 10 = 3$

Item set above 30 % Support	
{A}	6
{B}	7
{C}	6
{D}	7
{E}	6
{F}	3
{H}	3
{K}	4
{L}	4

Step-3 (C2):

Now generate Candidate C_2 from L_1 and find the support count for items

Items	Support
AB	4
AC	4
AD	4
AE	3
AF	1
AH	2
AK	2
AL	2
BC	5
BD	6
BE	4
BF	1
BH	0
BK	3
BL	2
CD	5
CE	2
CF	1

Items	Support
CH	1
CK	2
CL	1
DE	4
DF	2
DH	1
DK	2
DL	2
EF	2
EH	2
EK	3
EL	3
FH	2
FK	0
FL	2
HK	1
HL	2
KL	1

Step-4 (L2):

Now we compare Candidate C_2 generated in step 3 with the minimum support count and prune those Itemsets which do not satisfy the minimum support count (i.e. 30 %).

Items	Support
AB	4
AC	4
AD	4
AE	3
BC	5
BD	6
BE	4
BK	3
CD	5
DE	4
EK	3
EL	3

Step-5 (C3):

Now generate Candidate C_3 from L_2 and find the support count for items.

Items	Support
ABC	3
ABD	4
ABE	2
ABK	1
ACL	3
ACE	1
ADE	2
AEK	1
AEL	1
BCD	5
BCE	2
BCK	1
BDE	3
BDK	2
BEK	2
BEL	1
CDE	2
DEK	2
DEL	1

Step-6 (L3):

Now we compare Candidate C_3 generated in step 5 with the minimum support count and prune those Itemsets which do not satisfy the minimum support count (i.e. 30 %).

Items	Support
ABC	3
ABD	4
ACD	3
BCD	5
BDE	3

Step-7 (C4):

Now generate Candidate C_4 from L_3 and find the support count for items.

Items	Support
ABCD	3
ABDE	2
BCDE	2

Therefore ABCD is the large Itemset with minimum support 30%.

ASSOCIATION RULE:

Rule	Confidence	Confidence %
A → B ∧ C ∧D	3/6 = 0.5	50 %
B → A∧ C ∧ D	3/7 = 0.43	43 %
C → A ∧ B ∧ D	3/6 = 0.5	50 %
D → A ∧ B ∧ C	3/7 = 0.43	43 %
A ∧ B → C ∧ D	3/4 = 0.75	75 %
B ∧ C → A ∧ D	3/5 = 0.60	60 %
C ∧ D → A ∧ B	3/5 = 0.60	60 %
A ∧ C → B ∧ D	3/4 = 0.75	75 %
A ∧ D → B ∧ C	3/4 = 0.75	75 %
B ∧ C ∧ D → A	3/5 = 0.60	60 %
A ∧ C ∧ D → B	3/3 = 1	100 %
A ∧ B ∧ D → C	3/4 = 0.75	75 %
A ∧ B ∧ C → D	3/3 = 1	100 %

From the above rules generated, only the rules having greater than 70 % are considered as final rules. So final rules are,

$A \wedge B \rightarrow C \wedge D$

$A \wedge C \rightarrow B \wedge D$

$A \wedge D \rightarrow B \wedge C$

$A \wedge C \wedge D \rightarrow B$

$A \wedge B \wedge D \rightarrow C$

$A \wedge B \wedge C \rightarrow D$

Q7. A database has four transactions. Let minimum support = 50% and minimum confidence = 50%

TID	Items-Brought
T100	A, B, C
T200	A, C
T300	A, D
T400	B, E, F

Find all frequent item sets using Apriori algorithm. List strong association rules. **[5M | Dec17]**

Ans:

Given:

Minimum Support = 50 %

Minimum Confidence = 50%

Solution:

Step-1 (C1):

Scan the transaction database and find the count of Items.

Candidate List = {A, B, C, D, E, F}

Items	Support
A	3
B	2
C	2
D	1
E	1
F	1

Step-2 (L1):

Now compare candidate support count with minimum support count (i.e. 50%)

Items	Support
A	3
B	2
C	2

Step-3 (C2):

Now generate Candidate C_2 from L_1 and find the support count for items

Items	Support
AB	1
AC	2
BC	1

Step-4 (L2):

Now we compare Candidate C_2 generated in step 3 with the minimum support count and prune those Itemsets which do not satisfy the minimum support count (i.e. 50 %).

Items	Support
AC	2

So data contain the frequent item (A, C), therefore the association rule that can be generated from L2 are as shown below with the support and confidence.

Rule	Support	Confidence	Confidence %
A → C	2	2/3 = 0.66	66 %
C → A	2	2/2 = 1	100 %

Minimum confidence threshold is 50%, then both the rules are output as the confidence is above 50 %.

So Final Rules are:

A → C

C → A

Q8. A database has five transactions. Let minimum support = 30% and minimum Confidence = 70%

i. Find all frequent patterns using Apriori Algorithm.

ii. List strong association rules.

Transaction_Id	Items
A	1, 3, 4, 6
B	2, 3, 5, 7
C	1, 2, 3, 5, 8
D	2, 5, 9, 10
E	1, 4

Ans: **[10M | May18 & Dec18]**

Given:

Minimum Support = 30 %

Minimum Confidence = 70%

Solution:

Step-1 (C1):

Scan the transaction database and find the count of Items.

Candidate List = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

Items	Support
1	3
2	3
3	3
4	2
5	3
6	1
7	1
8	1
9	1
10	1

Step-2 (L1):

Now compare candidate support count with minimum support count (i.e. [30/100] x 5)

Items	Support
1	3
2	3
3	3
4	2
5	3

Step-3 (C2):

Now generate Candidate C_2 from L_1 and find the support count for items

Items	Support
(1, 2)	1
(1, 3)	2
(1, 4)	2
(1, 5)	1
(2, 3)	2
(2, 4)	0
(2, 5)	3
(3, 4)	1
(3, 5)	2
(4, 5)	0

Step-4 (L2):

Now we compare Candidate C_2 generated in step 3 with the minimum support count and prune those Item sets which do not satisfy the minimum support count (i.e. 30 %).

Items	Support
(1, 3)	2
(1, 4)	2
(2, 3)	2
(2, 5)	3
(3, 5)	2

Step-5 (C3):

Now generate Candidate C_3 from L_2 and find the support count for items

Items	Support
(1, 2, 3)	1
(1, 2, 5)	1
(2, 3, 5)	2

Step-6 (L3):

Now we compare Candidate C_3 generated in step 5 with the minimum support count and prune those Item sets which do not satisfy the minimum support count (i.e. 30 %).

Items	Support
(2, 3, 5)	2

So data contain the frequent item (2, 3, 5), therefore the association rule that can be generated from L3 are as shown below with the support and confidence.

Rule	Support	Confidence	Confidence %
2 → (3 ∧ 5)	2	2/3 = 0.66	66 %
3 → (2 ∧5)	2	2/3 = 0.66	66 %
5 → (2 ∧ 3)	2	2/3 = 0.66	66 %
(2 ∧ 3) → 5	2	2/2 = 1	100 %
(3 ∧ 5) → 2	2	2/2 = 1	100 %
(2 ∧ 5) → 3	2	2/3 = 0.66	66 %

So Final Rules are:

(2 ∧ 3) → 5

(3 ∧ 5) → 2