



**The Motto of Our University
(SEWA)**

SKILL ENHANCEMENT

EMPLOYABILITY

WISDOM

ACCESSIBILITY

SELF-INSTRUCTIONAL STUDY MATERIAL FOR JGND PSOU

ALL COPYRIGHTS WITH JGND PSOU, PATIALA

**JAGAT GURU NANAK DEV
PUNJAB STATE OPEN UNIVERSITY, PATIALA**

(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

B.Sc.(Data Science)

Semester III

Head Quarter: C/28, The Lower Mall, Patiala-147001

Website: www.psou.ac.in

The Study Material has been prepared exclusively under the guidance of Jagat Guru Nanak Dev Punjab State Open University, Patiala, as per the syllabi prepared by Committee of experts and approved by the Academic Council.

The University reserves all the copyrights of the study material. No part of this publication may be reproduced or transmitted in any form.

COURSE COORDINATOR AND EDITOR:

Dr. Amitoj Singh

Associate Professor

School of Sciences and Emerging Technologies

Jagat Guru Nanak Dev Punjab State Open University

Code	Subject	Nature of Course	Credits
BSDB32301T	Database Management Systems	CC-3	4
BSDB32302T	Data Mining and Visualization	DSC-9	4
BSDB32303T	Data Preparation	DSC-10	4
BSDB32304T	Mathematical Foundation for DS	SEC-1	4
BSDB32301P	Database Management Systems Lab	CC-3	2
BSDB32302P	Data Mining and Visualization Lab	DSC-11	2
BSDB32303P	Data Preparation Lab	DSC-12	2



**The Motto of Our University
(SEWA)**

SKILL ENHANCEMENT

EMPLOYABILITY

WISDOM

ACCESSIBILITY

SELF-INSTRUCTIONAL STUDY MATERIAL FOR JGND PSOU

ALL COPYRIGHTS WITH JGND PSOU, PATIALA

**JAGAT GURU NANAK DEV
PUNJAB STATE OPEN UNIVERSITY, PATIALA**

(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

B.Sc.(Data Science)

Semester III

BSDB32301T

Database Management System

Head Quarter: C/28, The Lower Mall, Patiala-147001

Website: www.psou.ac.in

The Study Material has been prepared exclusively under the guidance of Jagat Guru Nanak Dev Punjab State Open University, Patiala, as per the syllabi prepared by Committee of Experts and approved by the Academic Council.

The University reserves all the copyrights of the study material. No part of this publication may be reproduced or transmitted in any form.

COURSE COORDINATOR AND EDITOR:

Dr. Amitoj Singh

Associate Professor

School of Sciences and Emerging Technologies

Jagat Guru Nanak Dev Punjab State Open University

LIST OF CONSULTANTS/ CONTRIBUTORS

Sr. No.	Name
1	Dr. Sachin Ahuja
2	Dr. Rachpal Singh



JAGAT GURU NANAK DEV PUNJAB STATE OPEN UNIVERSITY, PATIALA
(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

PREFACE

Jagat Guru Nanak Dev Punjab State Open University, Patiala was established in December 2019 by Act 19 of the Legislature of State of Punjab. It is the first and only Open University of the State, entrusted with the responsibility of making higher education accessible to all, especially to those sections of society who do not have the means, time or opportunity to pursue regular education.

In keeping with the nature of an Open University, this University provides a flexible education system to suit every need. The time given to complete a programme is double the duration of a regular mode programme. Well-designed study material has been prepared in consultation with experts in their respective fields.

The University offers programmes which have been designed to provide relevant, skill-based and employability-enhancing education. The study material provided in this booklet is self-instructional, with self-assessment exercises, and recommendations for further readings. The syllabus has been divided in sections, and provided as units for simplification.

The University has a network of 10 Learner Support Centres/Study Centres, to enable students to make use of reading facilities, and for curriculum-based counselling and practicals. We, at the University, welcome you to be a part of this institution of knowledge.

Prof. Anita Gill
Dean Academic Affairs



B.Sc. (Data Science)
Core Course (CC)
Semester III
BSDB32301T: Data Base Management System

Total Marks: 100
External Marks: 70
Internal Marks: 30
Credits: 4
Pass Percentage: 35%

Objectives

This course explains fundamental elements of relational database management systems and made student familiar with the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL

Section A

Unit I: DBMS and its Architecture - Overview of DBMS, Basic DBMS terminology, Data independence. Architecture of a DBMS, Disadvantages of Traditional DBMS, Advantages and Characteristics of DBMS.

Unit II: Data Models –Relational Keys: Primary Key, Foreign Key, Candidate Key, Super Key etc., and Integrity Constraints, Relational model, Relational schema Hierarchical model, and Network model.

Unit III: Conceptual Data Modeling using E-R Data Model -Entities, attributes, relationships, generalization, specialization, specifying constraints, Conversion of ER Models to Tables, Practical problems based on E-R data model.

Unit IV: Normal Forms - Functional Dependency, Multi valued dependencies and Joined dependencies, 1NF, 2NF, 3NF, BCNF, 4NF, 5NF.

Section B

Unit V: Structured Query Language - Introduction to SQL, data types, DDL, DML, DCL, querying database tables, Data Definition Language (DDL), Creating Tables, Inserting and updating values into a Table.

Unit VI: Data Manipulation Language: Various form of SELECT- simple, using special operators, aggregate functions, group by clause, sub query, joins, co-related sub query, union clause, exist operator, Aggregate Functions.

Unit VII: Views- Introduction to views, data independence, Statements on Join Views, Dropping a VIEW. Database security, Security Techniques, Two Phase Locking Techniques.

Unit VIII: Data Control Operations - GRANT command, REVOKE command, COMMIT and ROLLBACK. Concurrency Control Techniques, Recovery Control techniques.

Suggested Readings

1. Silberschatz A., Korth F. H. and Sudarshan S., Database System Concepts, Tata McGraw Hill 6th ed., 2019
2. Elmasri R. and Navathe B. S., Fundamentals of Database Systems, Pearson 7th ed, 2016
3. Bayross I., SQL, PL/SQL the Programming Language of Oracle, BPB Publications 4th Ed., 2009
4. Vikram Vaswani., MySQL(TM): The Complete Reference, McGraw Hill Education, 2017
5. Robin Nixon, Learning Php, Mysql & Javascript Paperback (English) 4th Edition, O'Reilly Media, Inc., 2014



JAGAT GURU NANAK DEV PUNJAB STATE OPEN UNIVERSITY, PATIALA
(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

BSDB32301T: DATA BASE MANAGEMENT SYSTEM
COURSE COORDINATOR AND EDITOR: DR. AMITOJ SINGH

UNIT NO.	UNIT NAME
UNIT 1	DBMS AND ITS ARCHITECTURE
UNIT 2	DATA MODELS –RELATIONAL KEYS
UNIT 3	CONCEPTUAL DATA MODELING USING E-R DATA MODEL
UNIT 4	NORMAL FORMS
UNIT 5	STRUCTURED QUERY LANGUAGE
UNIT 6	DATA MANIPULATION LANGUAGE
UNIT 7	VIEWS
UNIT 8	DATA CONTROL OPERATIONS

B.Sc.(DATA SCIENCE)

SEMESTER-III

DATABASE MANAGEMENT SYSTEM

UNIT I: INTRODUCTION OF DBMS

STRUCTURE

1.0 Objectives

1.1 Introduction

1.2 Database Management System- Overview

1.2.1 What is a Database?

1.2.1.1 History of Database Systems

1.2.1.2 The Evolution of Database systems

1.2.1.3 Types of Databases

1.2.2 What is a Management System?

1.3 Basic terminology of DBMS

1.3.1 Goals of DBMS

1.3.2 Properties of DBMS

1.3.3 Need of DBMS

1.3.4 Features of DBMS

1.3.5 Users of DBMS

1.3.6 Levels of abstraction in a DBMS

1.4 Data Independence

1.4.1 Physical Data Independence

1.4.2 Logical Data Independence

1.5 Architecture of DBMS

1.5.1 Types of DBMS Architecture

1.5.2 Three Schema Architecture

1.6 Traditional File System VS DBMS

1.6.1 Traditional File system

1.6.2 Advantages of Traditional File System

1.6.3 Disadvantages of Traditional File System

1.6.4 Differences between Traditional File System & DBMS

1.7 Advantages of DBMS

1.8 Disadvantages of DBMS

1.9 Characteristics of DBMS

1.10 Applications of DBMS

1.11 Summary

1.12 References

1.13 Further Readings

1.14 Questions for Practice(Short & Long Answers Type)

1.0 OBJECTIVES

- To provide overview of Database Management System.
- To elaborate various types of databases.
- To explain basic DBMS terminology.
- To describe features of DBMS.
- To discuss various levels of abstraction in DBMS.
- To provide description of Data Independence.
- To explain architecture of DBMS.
- To discuss advantages and disadvantages of traditional file system.
- To differentiate traditional file system from modern DBMS.
- To describe advantages and disadvantages of DBMS.
- To describe characteristics as well as applications of DBMS.

1.1 INTRODUCTION

Database Management System is a collection of programs that allow you to create and maintain databases. It can also be defined as a software system that allows you to define, construct and manipulate databases used for various purposes such as to store customer and financial information of an organisation. In this unit, we will explain the basics of DBMS such as its Overview, Basic DBMS terminology, various terms of data independence and architecture of a DBMS. In addition, we will also discuss the disadvantages of traditional DBMS which will be followed by the advantages and characteristics of DBMS.

1.2 DATABASE MANAGEMENT SYSTEM-OVERVIEW

Database Management System or DBMS refers to the technology of storing and retrieving data of users with utmost efficiency along with appropriate security measures. As the name suggests, the database management system consists of two parts. They are:

- Database and
- Management System

1.2.1 What is a Database?

To find out what database is, we have to start from data along with a few other terms, which are the basic building block of any DBMS.

Data: It consists of facts, figures, statistics etc. having no particular meaning (e.g. 1, XYZ, 20 etc).

Record: It is a collection of related data items. For example, the three data items specified in the above data had no meaning. But if we organize them in the following way, then they collectively represent meaningful information.

Roll No.	Name	Age
1	XYZ	20

Thus, Here is the difference between data and record or information.

DATA	INFORMATION
1. It contains raw facts	1. It contains processed data
2. It is in unorganized form	2. It is in organized form
3. Data doesn't help in decision making process.	3. Information helps in decision making process.

Table/Relation: It is a collection of related records.

Roll No.	Name	Age
1	XYZ	20
2	ABC	26
3	EFG	30

The columns of this relation are called Fields, Attributes or Domains.

The rows are called Tuples or Records.

Database: It is a collection of related relations. Consider the following collection of tables:

T1

Roll No.	Name	Age
1	XYZ	20
2	ABC	26
3	EFG	30

T2

Roll No.	Address
1	IJK
2	LMN
3	QRS

T3

Roll No.	Year
1	I
2	II
3	I

T4

Year	Hostel
I	H1
II	H2

We now have a collection of 4 tables named as T1,T2,T3 and T4. They can be called a “related collection” because we can clearly find out that there are some common attributes existing in a selected pair of tables. Because of these common attributes we may combine the data of two or more tables together to find out the complete details of a student. Questions like “Which hostel does the youngest student live in?” can be answered now, although Age and Hostel attributes are in different tables.

Thus, database is a collection of inter-related data which is used to retrieve, insert and delete the data efficiently. It is also used to organize the data in the form of a table, schema, views, and reports, etc.

1.2.1.1 History of Database Systems

I. 1950’s and early 1960’s:

- Magnetic tapes were developed for data storage
- Data processing tasks such as payroll were automated, with data stored on tapes.
- Data could also be input from punched card decks, and output to printers.

II. Late 1960’s and 1970’s:

- The use of hard disks in the late 1960s changed the scenario for data processing greatly, since hard disks allowed direct access to data.
- With disks, network and hierarchical databases could be created that allowed data structures such as lists and trees to be stored on disk. Programmers could construct and manipulate these data structures.
- In the 1970’s , EF CODD defined the Relational Model.

III. In the 1980’s:

- Initial commercial relational database systems, such as IBM DB2, Oracle, Ingress, and DEC Rdb, played a major role in advancing techniques for efficient processing of declarative queries.
- Relational databases had become competitive with network and hierarchical database systems even in the area of performance.
- The 1980s also saw much research on parallel and distributed databases, as well as initial work on object-oriented databases.

IV. Early 1990’s:

- The SQL language was designed primarily in the 1990's and this is used for the transaction processing applications.
- Decision support and querying re-emerged as a major application area for databases.
- Database vendors also began to add object-relational support to their databases.

V. **Late 1990's:**

- The major event was the explosive growth of the World Wide Web.
- Databases were deployed much more extensively than ever before.
- Database systems now had to support very high transaction processing rates, as well as very high reliability and 24 * 7 availability (availability 24 hours a day, 7 days a week, meaning no downtime for scheduled maintenance activities).
- Database systems also had to support web interfaces to data.

1.2.1.2 **The Evolution of Database systems**

From pre-stage flat-file system, to relational and object-relational systems, database technology has gone through several generations and its history that is spread over more than 40 years now. The Evolution of Database systems is as follows:

- I. File Management System
- II. Hierarchical database System
- III. Network Database System
- IV. Relational Database System
- V. Object-Oriented Database System
- VI. Object-Relational Database System
- VII. Web-Enabled Database System

I. **File Management System(1960's-1980's)**

- Earlier, punched cards technology was used to store data – later, the File management system(FMS) is used to do so.
- It is also considered as a predecessor of database **in which** data is maintained in a flat file.
- FMS is a database that stores information in a single file or table.
- In a text file, every line contains one record where fields either have fixed length or they are separated by commas, whitespaces, tabs or any other character.

Advantages of File Management System

- It is best for small databases.
- It is easy to understand and implement. Fewer skills are required to handle a flat file database.
- Less hardware and software skills are required to maintain a flat file database.
- Various access methods , e.g., sequential, indexed, random.

Disadvantages of File Management System

- It may contain fields which duplicate the data as there is no automation in flat files.
- If one record is to be deleted from the system, then all the relevant information in different fields has to be deleted manually making the data manipulation inefficient.

- It wastes the computer space by requiring it to keep the information on items that are logically cannot be available.
- Information retrieving is very time consuming in a large database.
- It requires extensive programming in third-generation language such as COBOL, BASIC.

Implementation of a File Management System

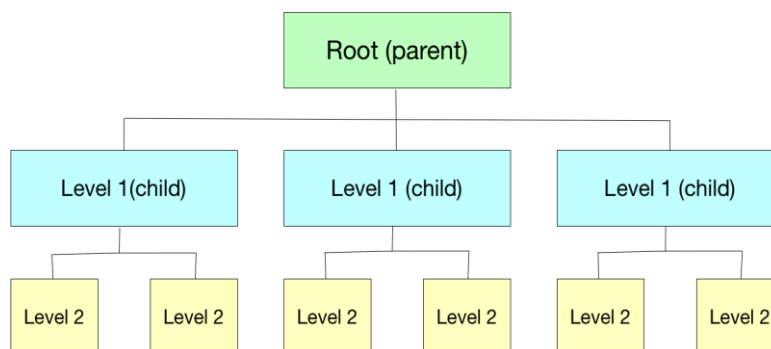
- It is implemented in Berkeley DB, SQLite, Mimesis, The Integration Engineer etc.

II. Hierarchical Database System (1968-1980)

Prominent hierarchical database model was IBM's first DBMS called **IMS (Information Management System)**

- Mid 1960s Rockwell collaborates with IBM to create the Information Management System (IMS) which lead the mainframe database market in 70's and early 80's.
- The drawbacks of previous system FMS in terms of accessing records and sorting records which took a long time was overcome in this database system via introduction of parent-child relationship between records in a database.
- The origin of the data is called the root from which several branches have data at different levels and the last level is called the leaf.

The Hierarchical Database Model



Advantages of Hierarchical Database System

- In a hierarchical database system, pace of accessing the information is speedy due to the predefined paths. This increases the performance of a database.
- The relationships among different entities are easy to understand.
- Less redundant data.
- Data independence.
- Database security and integrity.

Disadvantages of Hierarchical Database System

- Hierarchical database model lacks flexibility. If a new relationship is to be established between two entities then a new and possibly a redundant database structure has to be build.

- Maintenance of data is inefficient in a hierarchical model. Any change in the relationships may require manual reorganization of the data.
- This system is also inefficient for non-hierarchical accesses.
- Complex implementation
- Lacks structural independence.

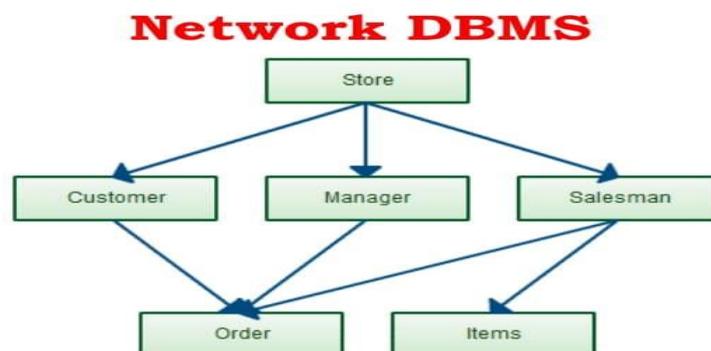
In order to avoid these drawbacks, the following system took its origin named as the Network Database System.

III. Network Database System (1960's – 1990's)

- Early 1960s, Charles Bachmann developed first DBMS at Honeywell, **Integrated Data Store (IDS)**
- It was standardized in 1971 by the **CODASYL** group (**Conference on Data Systems Languages**).
- Unlike hierarchical database model, Network database system allows multiple parent and child relationships i.e., it maintains many-to-many relationship.
- Network database system is basically a graph structure which was created to achieve three main objectives:
 - a) To represent complex data relationships more effectively.
 - b) To improve the performance of the database.
 - c) To implement a database standard.
- In a Network database system, a relationship is referred to as a set. Each set comprises of two types of records, an owner record which is same as parent type in hierarchical and a member record which is similar to the child type record in hierarchical database model.

Thus, this system identifies the following three database components:

1. Network schema—database organization[structure]
 2. Sub-schema—views of database per user
 3. Data management language — at low level , procedural
- It can also be represented as follows:



Advantages of Network Database System

- The network database system makes the data access quite easy and proficient as an application can access the owner record and all the member records within a set.
- This system is conceptually easy to design.
- This system ensures data integrity because no member can exist without an owner. So the user must make an owner entry and then the member records.
- It also ensures the data independence because the application works independently of the data.

Disadvantages of Network Database System

- This system lacks structural independence which means that to bring any change in the database structure; the application program must also be modified before accessing the data.
- A user friendly database management system cannot be established via network model.

Implementation of Network Database System

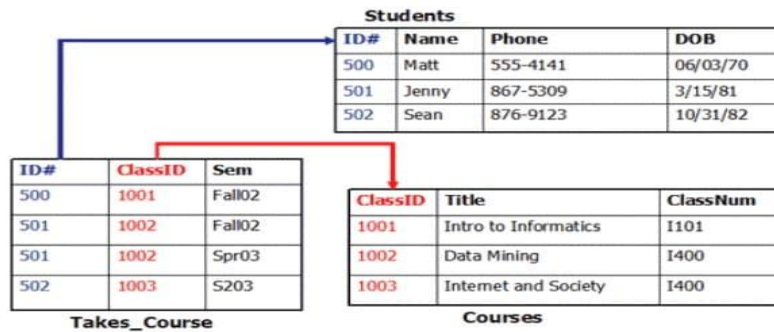
- Network database system is implemented in Digital Equipment Corporation DBMS-10, Digital Equipment Corporation DBMS-20, RDM Embedded, Turbo IMAGE, Univac DMS-1100 etc.

IV. Relational Database System (1970's-present)

- The relational database model was proposed by E. F. Codd in 1970.
- It can be defined using the following two terminologies:
 1. Instance – a table with rows or columns.
 2. Schema – specifies the structure (name of relation, name and type of each column)
- This system is based on branches of mathematics called set theory and predicate logic.
- After the hierarchical and network database systems, the birth of this system was huge step ahead.
- It allows the entities to be related through a common attribute. So in order to relate two tables (entities), they simply need to have a common attribute.
- Thus using relational database ample information can be stored using small tables.
- The accessing of data is also very efficient. The user only has to enter a query, and the application provides the user with the asked information.
- Relational databases are established using a computer language, Structured Query Language (SQL). This language forms the basis of all the database applications available today, from Access to Oracle.

- Relationships between tables are also formed in this system as following.

Relational DBMS



Advantages of Relational Database System

- Relational database system supports mathematical set of operations like union, intersection, difference and Cartesian product. It also supports select, project, relational join and division operations.
- It uses normalization structure which helps to achieve data independence more easily.
- Security control can also be implemented more effectively by imposing an authorization control on the sensitive attributes present in a table.
- It uses a language which is easy and human readable.

Disadvantages of Relational Database System

- The response to a query becomes time-consuming and inefficient if the number of tables between which the relationships are established increases.

Implementation of Relational Database System

- It is implemented in Oracle, Microsoft, IBM, MySQL, PostgreSQL, SQLite etc.

V. Object – Oriented Database System (1990's – present)

- Object oriented database system is one in which the data or information is presented in the form of objects, much like in object-oriented programming language as shown in Figure 1.
- Furthermore, object oriented DBMS also facilitate the user by offering transaction support, language for various queries, and indexing options.
- Also, these database systems have the ability to handle data efficiently over multiple servers.
- Unlike relational database, object-oriented database works in the framework of real programming languages like JAVA or C++.

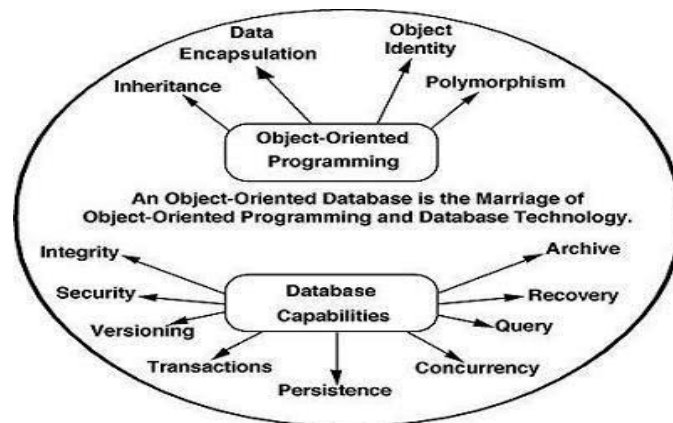


Figure 1. Makeup of an Object-Oriented Database

Advantages of Object-Oriented Database System

- If there are complex (many-to-many) relationships between the entities, the object-oriented database handles them much faster than any of the above discussed database systems.
- Navigation through the data is much easier.
- Objects do not require assembly or disassembly hence it saves the coding and execution time.

Disadvantages of Object-Oriented Database System

- Lower efficiency level when data or relationships are simple.
- Data can be accessible via specific language using a particular API which is not the case in relational databases.

VI. Object – Relational Database System (1990's – present)

- Object relational databases span the object and relational concepts.
- It displays a modified object-oriented user-display over the already implemented relational database management system. When various software interact with this modified-database management system, they will customarily operate in a manner such that the data is assumed to be saved as objects.
- The basic working of this database system is that it translates the useful data into organized tables, distributed in rows and columns, and from then onwards, it manages data the same way as done in a relational database system.
- Similarly, when the data is to be accessed by the user, it is again translated from processed to complex form.

Advantages of Object-Relational Database System

- Data remains encapsulated in object-relational database.
- Concept of inheritance and polymorphism can also be implemented in this database.

Disadvantages of Object-Relational Database System

- Object relational database is complex.

- Proponents of relational approach believe simplicity and purity of relational model are lost.
- It is costly as well.

VII. Web-Enabled Database (1990's – present)

- Web-Enabled database simply put a database with a web-based interface.
- This implies that there can be a separation of concerns; namely, the web designer does not need to know the details about the DB's underlying design.
- Similarly, the DB designer needs to concern himself with the DB's web interface.
- A web enabled database uses three layers to function: a presentation layer, a middle layer and the database layer.

Advantages of Web-Enabled Database

- A web-enabled database allows users to get the information they need from a central repository on demand.
- The database is easy and simple to use.
- The data accessibility is easy via web-enabled database.

Disadvantages of Web-Enabled Database

- Main disadvantage is that it can be hacked easily.
- Web enabled databases support the full range of DB operations, but in order to make them easy to use, they must be “dumped down”.

1.2.1.3 Types of Databases

There are various types of databases used for storing different varieties of data:

I. Centralized Database

- It is the type of database that stores data at a centralized database system.
- It comforts the users to access the stored data from different locations through several applications.
- These applications contain the authentication process to let users access data securely.
- An example of a Centralized database can be Central Library that carries a central database of each library in a college/university.

Advantages of Centralized Database

- It has decreased the risk of data management, i.e. manipulation of data will not affect the core data.
- Data consistency is maintained as it manages data in a central repository.
- It also provides better data quality, which enables organizations to establish data standards.
- It is less expensive because a few vendors are required to handle the data sets.

Disadvantages of Centralized Database

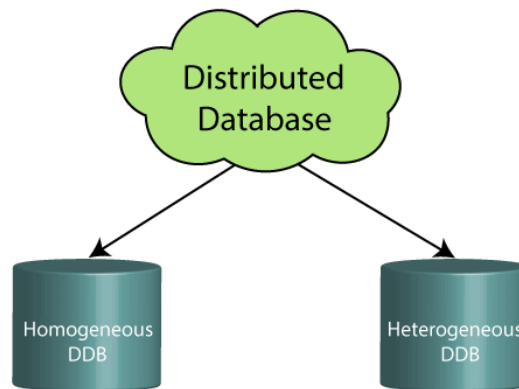
- The size of the centralized database is large, which increases the response time for fetching the data.
- It is not easy to update such an extensive database system.
- If any server failure occurs, entire data will be lost which could be a huge loss.

II. Distributed Database(DDB)

- Unlike a centralized database system, data is distributed among different database systems of an organization in distributed systems.
- These database systems are connected via communication links.
- Such links help the end-users to access the data easily.
- Examples of the Distributed database are Apache Cassandra, HBase, Ignite, etc.

Classification of Distributed Database System

- A distributed database system can further be classified into two parts as following:



Homogeneous DDB: The database systems which execute on the same operating system and use the same application process and carry the same hardware devices are known as homogeneous distributed database systems.

Heterogeneous DDB: The database systems which execute on different operating systems under different application procedures, and carry different hardware devices are known as heterogeneous distributed database systems.

Advantages of Distributed Database

- Modular development is possible in a distributed database. i.e. the system can be expanded by including new computers and connecting them to the distributed system.
- One server failure will not affect the entire data set.

Disadvantages of Distributed Database

Although, distributed DBMS is capable of effective communication and data sharing still it suffers from various disadvantages are as following below.

- **Complex nature** :The nature of Distributed DBMS is more complex than a centralized DBMS because of requirement of complex softwares. Also, It ensures no data replication, which adds even more complexity in its nature.
- **Overall Cost** :Various costs such as maintenance cost, procurement cost, hardware cost, network/communication costs, labor costs, etc, adds up to the overall cost and make it costlier than normal DBMS.
- **Security issues**:In a Distributed Database, along with maintaining no data redundancy, the security of data as well as network is a prime concern. A network can be easily attacked for data theft and misuse.
- **Integrity Control**:In a vast distributed database system, maintaining data consistency is important. All changes made to data at one site must be reflected to all the sites. The communication and processing cost is high in distributed DBMS in order to enforce the integrity of data.
- **Lacking Standards**:Although it provides effective communication and data sharing, still there are no standard rules and protocols to convert a centralized DBMS to a large distributed DBMS. Lack of standards decreases the potential of distributed DBMS.

III. Relational Database

- This database is based on the relational data model, which stores data in the form of rows(tuple) and columns(attributes), and together forms a table(relation).
- A relational database uses SQL for storing, manipulating, as well as maintaining the data.
- E.F. Codd invented the database in 1970. Each table in the database carries a key that makes the data unique from others.
- Examples of Relational databases are MySQL, Microsoft SQL Server, Oracle, etc.

Properties of Relational Database

There are following four commonly known properties of a relational model known as ACID properties, where:

- **A means Atomicity**: This ensures the data operation will complete either with success or with failure. It follows the 'all or nothing' strategy. For example, a transaction will either be committed or will abort.
- **C means Consistency**: If we perform any operation over the data, its value before and after the operation should be preserved. For example, the account balance before and after the transaction should be correct, i.e., it should remain conserved.
- **I means Isolation**: There can be concurrent users for accessing data at the same time from the database. Thus, isolation between the data should remain isolated. For example, when multiple transactions occur at the same time, one transaction effects should not be visible to the other transactions in the database.
- **D means Durability**: It ensures that once it completes the operation and commits the data, data changes should remain permanent.

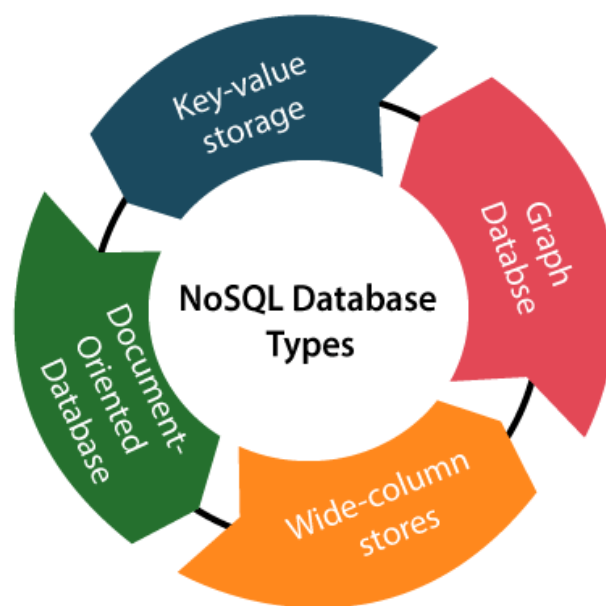
IV. NoSQL Database

- Non-SQL/Not Only SQL is a type of database that is used for storing a wide range of data sets.
- It is not a relational database as it stores data not only in tabular form but in several different ways.
- It came into existence when the demand for building modern applications increased.

Thus, NoSQL presented a wide variety of database technologies in response to the demands.

Classification of NoSQL Database

We can further divide a NoSQL database into the following four types:



Key-value storage: It is the simplest type of database storage where it stores every single item as a key (or attribute name) holding its value, together.

Document-oriented Database: A type of database used to store data as JSON-like document. It helps developers in storing data by using the same document-model format as used in the application code.

Graph Databases: It is used for storing vast amounts of data in a graph-like structure. Most commonly, social networking websites use the graph database.

Wide-column stores: It is similar to the data represented in relational databases. Here, data is stored in large columns together, instead of storing in rows.

Advantages of NoSQL Database

- It enables good productivity in the application development as it is not required to store data in a structured format.
- It is a better option for managing and handling large data sets.
- It provides high scalability.

- Users can quickly access data from the database through key-value.

Disadvantages of NoSQL Database

- NoSQL databases don't have the reliability functions which Relational Databases have (basically don't support ACID).
- This also means that NoSQL databases offer consistency in performance and scalability.
- In order to support ACID, developers will have to implement their own code, making their systems more complex.
- This may reduce the number of safe applications that commit transactions, for example bank systems.
- NoSQL is not compatible (at all) with SQL.
 - *Note:* Some NoSQL management systems do use a Structured Query Language.
 - This means that you will need a manual query language, making things slower and more complex.
- NoSQL are very new compared to Relational Databases, which means that are far less stable and may have a lot less functionalities.

V. Cloud Database

- A type of database where data is stored in a virtual environment and executes over the cloud computing platform.
- It provides users with various cloud computing services (SaaS, PaaS, IaaS, etc.) for accessing the database.
- There are numerous cloud platforms, but the best options are Amazon Web Services(AWS), Microsoft Azure, Kamatera, PhonixNAP, ScienceSoft, Google Cloud SQL, etc.

Advantages of Cloud Database

- Scalability
- Reduced Administrative Burden
- Improved Security

Disadvantages of Cloud Database

- Inflexibility
- Cost
- Downtime

VI. Personal Database

- Collecting and storing data on the user's system defines a Personal Database.
- This database is basically designed for a single user.

Advantages of Personal Database

- It is simple and easy to handle.

- It occupies less storage space as it is small in size.

Disadvantages of Personal Database

- Fewer amounts of data and information are stored in personal database management system.
- There is no connectivity with other computer to get more data.

VII. Operational Database

- The type of database which creates and updates the database in real-time is known as an operational database.
- It is basically designed for executing and handling the daily data operations in several businesses.
- For example, An organization uses operational databases for managing per day transactions.

Advantages of Operational Database

- Operational Databases systems are greatly versatile and accommodate distributed systems like NoSQL, SQL, New SQL Databases.
- These systems are highly available, fault-tolerant and highly scalable as discussed.
- They are highly secured as they offer built-in support for encryption, auditing, and protection from cyber
- They are extremely flexible and can work with multiple applications without losing the state of the database.
- Such systems are often more economical as systems in an operational database are usually based on distributed networks and systems thereby reducing costs drastically and ensuring consistency.

Disadvantages of Operational Database

- They usually have a learning curve where personnel are required to give relevant training to manage such databases and that increases the overheads expenses
- Even the installation process of such an operational database system requires time and effort where they need to be set up in an optimal way of meeting business goals and extracting most benefit from the system.
- Security could also be an issue where since the data is stored in a remote location having overall control could be difficult and we have seen recent examples of customer data being hacked from some of the most prominent tech companies.

VIII. Enterprise Database

- Large organizations or enterprises use this database for managing a massive amount of data.
- It helps organizations to increase and improve their efficiency.
- Such database also allows simultaneous access to users.

Advantages of Enterprise Database

- Multi processes are supportable over the Enterprise database.

- It allows executing parallel queries on the system.

Disadvantages of Enterprise Database

- High cost of implementation and maintenance..
- Inflexibility of the system.

Other than above mentioned database systems, following are the database systems which are already discussed in evolution part such as:

- Object-Oriented Databases
- Hierarchical Databases
- Network Databases

Database can also be classified according to the following factors. They are:

I. Number of Users

a)Single user database: It supports only one user at a time. For e.g. Desktop or personal computer database.

b)Multiuser database:It supports multiple users at the same time. For e.g. Workgroup database and enterprise database

II. Database Location

a)Centralized Database

b)Distributed Database

III. Expected type

IV. Extent of use

1.2.2 What is a Management System?

- The Management system is important because without the existence of some kind of rules and regulations, it is not possible to maintain the database.
- We have to select the particular attributes which should be included in a particular table; the common attributes to create relationship between two tables; if a new record has to be inserted or deleted then which tables should have to be handled etc.
- These issues must be resolved by having some kind of rules to follow in order to maintain the integrity of the database.
- Database systems are designed to manage large bodies of information.
- Management of data involves both defining structures for storage of information and providing mechanisms for the manipulation of information.
- In addition, the database system must ensure the safety of the information stored, despite system crashes or attempts at unauthorized access.
- If data is to be shared among several users, the system must avoid possible anomalous results.
- Because information is so important in most organizations, computer scientists have developed a large body of concepts and techniques for managing data. These concepts and techniques form the focus of this chapter.

Self-Check Exercise

1. What is a Database? Give example.

2. Explain various types of database systems.
3. Define Relational Database System along with its advantage and disadvantage.
4. Describe the various advantages of Operational Database?
5. Explain Centralized database.
6. Write the difference between Homogeneous and Heterogeneous Distributed DataBase.

1.3 BASIC TERMINOLOGY OF DBMS

Database management system is a software which is used to manage the database. For example: MySQL, Oracle, etc are very popular commercial databases which are used in different applications.

Thus, The interactions catered for by most existing DBMS fall into four main groups:

- a) **Data Definition:** It helps in creation, modification and removal of definitions that define the organization of data in database.
- b) **Data Updation:** It helps in insertion, modification and deletion of the actual data in the database.
- c) **Data Retrieval:** It helps in retrieval of data from the database which can be used by applications for various purposes.
- d) **User Administration:** It helps in registering and monitoring users, enforcing data security, monitoring performance, maintaining data integrity, dealing with concurrency control and recovering information corrupted by unexpected failure.

1.3.1 Goals of DBMS

- The primary goal of a DBMS is to provide a way to store and retrieve database information that is both convenient and efficient
- Manages large bodies of information
- Provides convenient and efficient ways to store and access information
- Secures information against system failure or tampering
- Permits data to be shared among multiple users

1.3.2 Properties of DBMS

- It represents some aspect of the real world. Changes to the real world reflected in the database.
- It is a logically coherent collection of data with some inherent meaning.
- It is designed and populated with data for a specific purpose.

1.3.3 Need of DBMS

- Before the advent of DBMS, organizations typically stored information using a “File Processing Systems”.
- Example of such systems is File Handling in High Level Languages like C, Basic and COBOL etc., these systems have major disadvantages to perform the data manipulation. So to overcome those drawbacks now we are using the DBMS.

- In addition to that, the database system must ensure the safety of the information stored, despite system crashes or attempts at unauthorized access. If data is to be shared among several users, the system must avoid possible anomalous results.
- Therefore, DBMS was a new concept then, and all the research was done to make it overcome the deficiencies in traditional style of data management.

1.3.4 Features of DBMS

Features of database management system are:

- Prevents data redundancy.
- Prevents unauthorised access.
- Provides persistent storage for program objects and data structures.
- Provides Multiple user interfaces.
- Provides Integrity Constraints.
- Provides Backup and Recovery.

1.3.5 Users of DBMS

A typical DBMS has users with different rights and permissions who use it for different purposes. Some users retrieve data and some back it up. The users of a DBMS can be broadly categorized as follows –

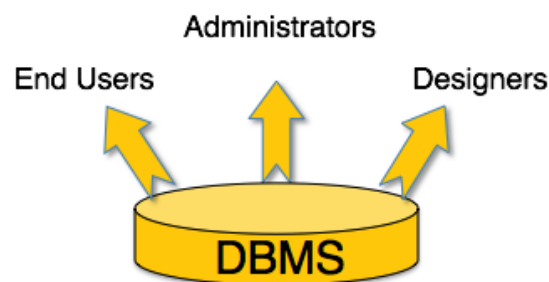


Figure: DBMS Users

I. Administrators

- Administrators maintain the DBMS and are responsible for administrating the database.
- They are responsible to look after its usage and by whom it should be used.
- They create access profiles for users and apply limitations to maintain isolation and force security.
- Administrators also look after DBMS resources like system license, required tools, and other software and hardware related maintenance.

II. Designers

- Designers are the group of people who actually work on the designing part of the database.

- They keep a close watch on what data should be kept and in what format.
- They identify and design the whole set of entities, relations, constraints, and views.

III. End Users

- End users are those who actually reap the benefits of having a DBMS.
- End users can range from simple viewers who pay attention to the logs or market rates to sophisticated users such as business analysts.

Thus, People who work with a database can be categorized as database users or database administrators.

➤ Database Users:

There are four different types of database-system users, differentiated by the way they expect to interact with the system.

a) Naive users

- Naive users are unsophisticated users who interact with the system by invoking one of the application programs that have been written previously.
- For example, a bank teller who needs to transfer \$50 from account A to account B invokes a program called transfer.
- This program asks the teller for the amount of money to be transferred, the account from which the money is to be transferred, and the account to which the money is to be transferred.

b) Application programmers

- Application programmers are computer professionals who write application programs.
- Application programmers can choose from many tools to develop user interfaces.
- Rapid application development (RAD) tools are tools that enable an application programmer to construct forms and reports without writing a program.

c) Sophisticated users

- Sophisticated users interact with the system without writing programs.
- Instead, they form their requests in a database query language.
- They submit each such query to a query processor, whose function is to break down DML statements into instructions that the storage manager understands.
- Analysts who submit queries to explore data in the database fall in this category.

d) Specialized users

- Specialized users are sophisticated users who write specialized database applications that do not fit into the traditional data-processing framework.

1.3.6 Levels of abstraction in a DBMS

- Hiding certain details of how the data are stored and maintained. A major purpose of database system is to provide users with an “Abstract View” of the data.
- The goal of the abstraction in the DBMS is to separate the users request and the physical storage of data in the database.

- In DBMS there are 3 levels of data abstraction.

i. Physical Level

The lowest Level of Abstraction describes “How” the data is actually stored. The physical level describes complex low level data structures in detail.

ii. Logical Level

This level of data Abstraction describes “What” data is to be stored in the database and what relationships exist among those data. Database Administrators use the logical level of abstraction.

iii. View Level

It is the highest level of data Abstracts that describes only part of entire database. Different users require different types of data elements from each database. The system may provide many views for the some database.

Self-Check Exercise

1. List the properties of DBMS.
2. How can you differentiate sophisticated users from specialized users?
3. What are the goals of database management system?
4. Why do you need database management system?
5. What is the role of administrators in dbms?
6. What is the goal of abstraction in the database management system?

1.4 DATA INDEPENDENCE

- A very important advantage of using DBMS is that it offers Data Independence which is an ability to modify a scheme definition in one level without affecting a scheme definition in a higher level.
- It is of two types:
 - Physical Data Independence
 - Logical Data Independence

1.4.1 Physical Data Independence

- It is an ability to modify the physical schema without causing application programs to berewritten.
- Modifications at this level are usually to improve performance.

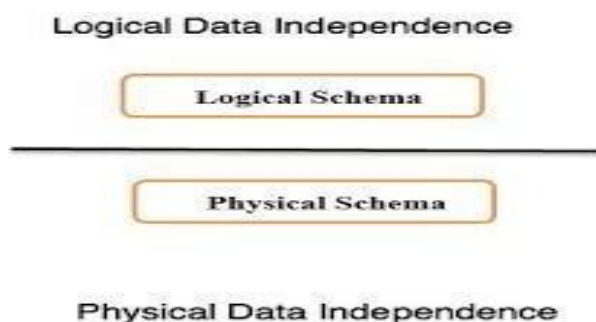


Fig: Data Independence

1.4.2 Logical Data Independence

- It is an ability to modify the conceptual schema without causing application programs to be rewritten.
- It is usually done when logical structure of database is altered.
- Logical data independence is harder to achieve as the application programs are usually heavily dependent on the logical structure of the data.

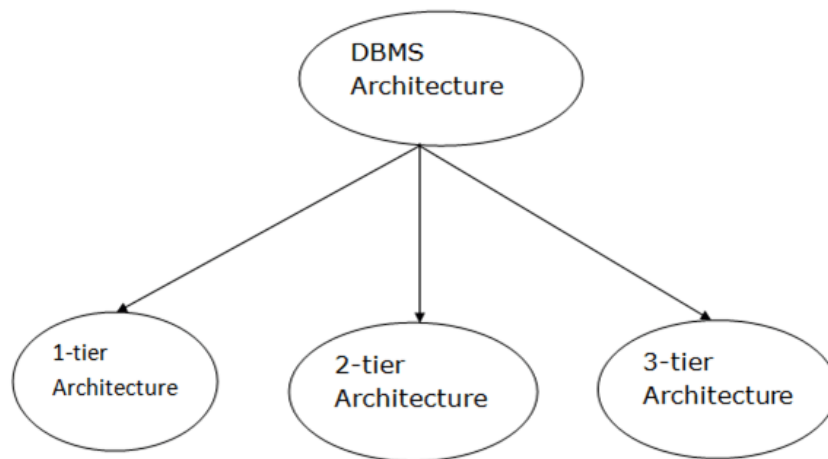
1.5 ARCHITECTURE OF DBMS

The DBMS design depends upon its architecture. The basic client/server architecture is used to deal with a large number of PCs, web servers, database servers and other components that are connected with networks.

The client/server architecture consists of many PCs and a workstation which are connected via the network.

DBMS architecture depends upon how users are connected to the database to get their request done.

1.5.1 Types of DBMS Architecture



- Database architecture can be seen as a single tier or multi-tier.
- But logically, database architecture is of two types like:

1.5.1.1 1-Tier Architecture

- In this architecture, the database is directly available to the user. It means the user can directly sit on the DBMS and uses it.
- Any changes done here will directly be done on the database itself. It doesn't provide a handy tool for end users.
- The 1-Tier architecture is used for development of the local application, where programmers can directly communicate with the database for the quick response.

1.5.1.2 2-Tier Architecture

- The 2-Tier architecture is same as basic client-server. In the two-tier architecture, applications on the client end can directly communicate with the database at the server side. For this interaction, API's like: **ODBC**, **JDBC** are used.
- The user interfaces and application programs are run on the client-side.
- The server side is responsible to provide the functionalities like: query processing and transaction management.
- To communicate with the DBMS, client-side application establishes a connection with the server side.

1.5.1.3 3-Tier Architecture

- The 3-Tier architecture contains another layer between the client and server. In this architecture, client can't directly communicate with the server.
- The application on the client-end interacts with an application server which further communicates with the database system.
- End user has no idea about the existence of the database beyond the application server. The database also has no idea about any other user beyond the application.
- The 3-Tier architecture is used in case of large web application.

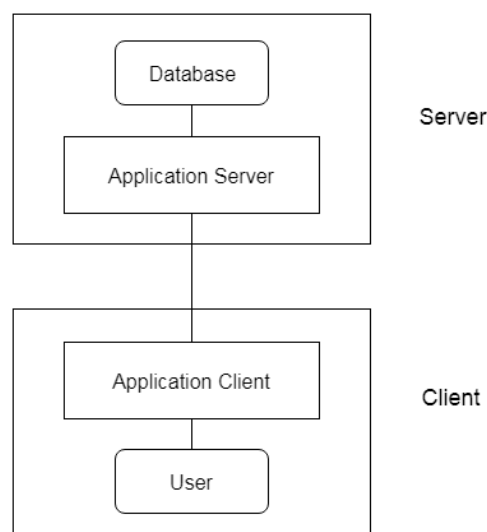
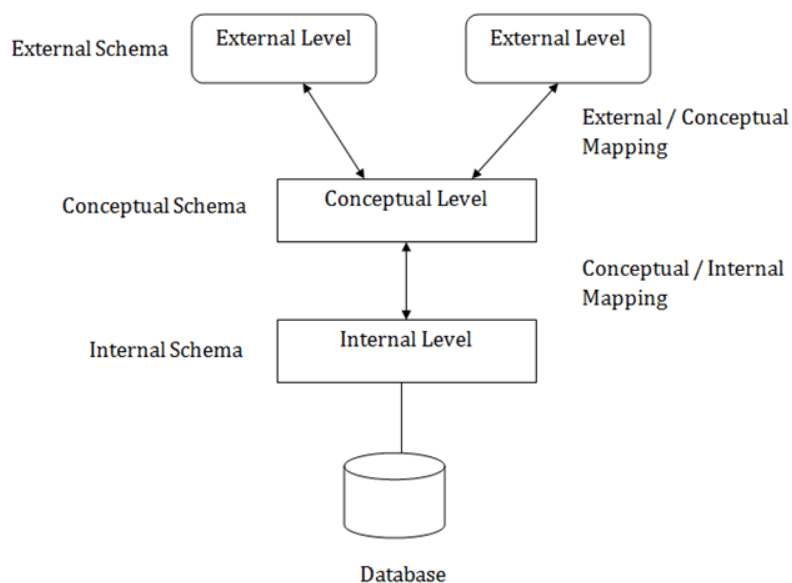


Fig: 3-tier Architecture

1.5.2 Three Schema Architecture

- The three schema architecture is also called ANSI/SPARC architecture or three-level architecture.
- This framework is used to describe the structure of a specific database system.
- The three schema architecture is also used to separate the user applications and physical database.
- The three schema architecture contains three-levels. It breaks the database down into three different categories.
- **The three-schema architecture is as follows:**



- Above shows the DBMS architecture.
- Mapping is used to transform the request and response between various database levels of architecture.
- Mapping is not good for small DBMS because it takes more time.
- In External / Conceptual mapping, it is necessary to transform the request from external level to conceptual schema.
- In Conceptual / Internal mapping, DBMS transform the request from the conceptual to internal level.

1. Internal Level

- The internal level has an internal schema which describes the physical storage structure of the database.
- The internal schema is also known as a physical schema.
- It uses the physical data model. It is used to define that how the data will be stored in a block.
- The physical level is used to describe complex low-level data structures in detail.

2. Conceptual Level

- The conceptual schema describes the design of a database at the conceptual level. Conceptual level is also known as logical level.
- The conceptual schema describes the structure of the whole database.
- The conceptual level describes what data are to be stored in the database and also describes what relationship exists among those data.
- In the conceptual level, internal details such as an implementation of the data structure are hidden.
- Programmers and database administrators work at this level.

3. External Level

- At the external level, a database contains several schemas that sometimes called as subschema. The subschema is used to describe the different view of the database.
- An external schema is also known as view schema.
- Each view schema describes the database part that a particular user group is interested and hides the remaining database from that user group.
- The view schema describes the end user interaction with database systems.

Self-check exercise:

1. What do you understand by the term data independence?
2. What is the difference between Internal level and Conceptual level in architecture of DBMS?
3. Why do you use mapping in three schema architecture of DBMS?
4. How can you differentiate 1-tier architecture from 2-tier architecture in DBMS?

1.6 TRADITIONAL FILE SYSTEM VS DBMS

1.6.1 Traditional File system

Traditional File systems were the collection of data with the help of written procedures for managing the database. It provides the details of data representation and storage of data. In traditional file system the data is stored in the group of files. The data stored in the files are dependent on each other. The traditional file system does not have a crash recovery mechanism but it has a fast file system recovery. Normally, low level languages like C, C++ and COBOL were used to design the files.

1.6.2 Advantages of Traditional File System

Following are the advantages of traditional file system:

- File processing costs less and can be more speedy than database.
- File processing design approach is well suited to mainframe hardware and batch input.
- Companies mainly use file processing to handle large volumes of structured data on a regular basis.
- It can be more efficient and costs less than DBMS in certain situations.

- Design is simple.
- Customization is easy and efficient.

1.6.3 Disadvantages of Traditional File System

Following are the disadvantages of traditional file system:

- Data redundancy and inconsistency.
- Difficulty in accessing data.
- Data isolation – multiple files and formats.
- Integrity problems
- Unauthorized access is not restricted.
- It co-ordinates only physical access.

To overcome disadvantages of File system, DBMS came in use which is a collection of inter-related data. It has a set of programs to access the data. Basically, it contains information about particular enterprise. It provides convenient and efficient environment for use.

1.6.4 Differences between Traditional File System and DBMS

File System	DBMS
A file system is software that manages and organizes the files in a storage medium. It controls how data is stored and retrieved.	DBMS or Database Management System is a software application. It is used for accessing, creating, and managing databases.
The file system provides the details of data representation and storage of data.	DBMS gives an abstract view of data that hides the details
Storing and retrieving of data can't be done efficiently in a file system.	DBMS is efficient to use as there are a wide variety of methods to store and retrieve data.
It does not offer data recovery processes.	There is a backup recovery for data in DBMS.
The file system doesn't have a crash recovery mechanism.	DBMS provides a crash recovery mechanism
Protecting a file system is very difficult.	DBMS offers good protection mechanism.
In a file management system, the redundancy of data is greater.	The redundancy of data is low in the DBMS system.
Data inconsistency is higher in the file system.	Data inconsistency is low in a database management system.
The file system offers lesser security.	Database Management System offers high

	security.
File System allows you to stores the data as isolated data files and entities.	Database Management System stores data as well as defined constraints and interrelation.
Not provide support for complicated transactions.	Easy to implement complicated transactions.
The centralization process is hard in File Management System.	Centralization is easy to achieve in the DBMS system.
It doesn't offer backup and recovery of data if it is lost.	DBMS system provides backup and recovery of data even if it is lost.
There is no efficient query processing in the file system.	You can easily query data in a database using the SQL language.
These system doesn't offer concurrency.	DBMS system provides a concurrency facility.

1.7 ADVANTAGES OF DBMS

Following are the advantages of DBMS:

- **Controls database redundancy:** It can control data redundancy because it stores all the data in one single database file and that recorded data is placed in the database.
- **Data sharing:** In DBMS, the authorized users of an organization can share the data among multiple users.
- **Easily Maintenance:** It can be easily maintainable due to the centralized nature of the database system.
- **Reduce time:** It reduces development time and maintenance need.
- **Backup:** It provides backup and recovery subsystems which create automatic backup of data from hardware and software failures and restores the data if required.
- **Multiple user interface:** It provides different types of user interfaces like graphical user interfaces, application program interfaces etc.

1.8 DISADVANTAGES OF DBMS

Database Management System is quite useful compared to the file based management system. However, it does have some disadvantages. Some of those are as follows:

- **Expensive:** Creating and managing a database is quite costly. High cost software as well as hardware are required for the database. Also highly trained staff is required to handle the database and it also needs continuous maintenance. All of these ends up making a database quite a costly venture.
- **High Complexity:** A Database Management System is quite complex as it involves creating, modifying and editing a database. Consequently, the people who handle a database or work with it need to be quite skilled or valuable data can be lost.
- **Database handling staff required:** As discussed in the previous point, database and DBMS are quite complex. Hence, skilled personnel are required to handle the database so that it works in optimum condition. This is a costly venture as these professionals need to be very well paid.
- **Database Failure:** All the relevant data for any company is stored in a database. So it is imperative that the database works in optimal condition and there are no failures. A database failure can be catastrophic and can lead to loss or corruption of very important data.
- **High Hardware Cost:** A database contains vast amount of data. So a large disk storage is required to store all this data. Sometimes extra storage may even be needed. All this increases hardware costs by a lot and makes a database quite expensive.
- **Huge Size:** A database contains a large amount of data, especially for bigger organisations. This data may even increase as more data is updated into the database. All of these leads to a large size of the database. The bigger the database is, it is more difficult to handle and maintain. It is also more complex to ensure data consistency and user authentication across big databases.
- **Upgradation Costs:** Often new functionalities are added to the database. This leads to database upgradations. All of these upgradations cost a lot of money. Moreover it is also quite expensive to train the database managers and users to handle these new upgradations.
- **Cost of Data Conversion:** If the database is changed or modified in some manner, all the data needs to be converted to the new form. This cost may even exceed the database creation and management costs sometimes. This is the reason most organisations prefer to work on their old databases rather than upgrade to new ones.

1.9 CHARACTERISTICS OF DBMS

A modern DBMS has the following characteristics –

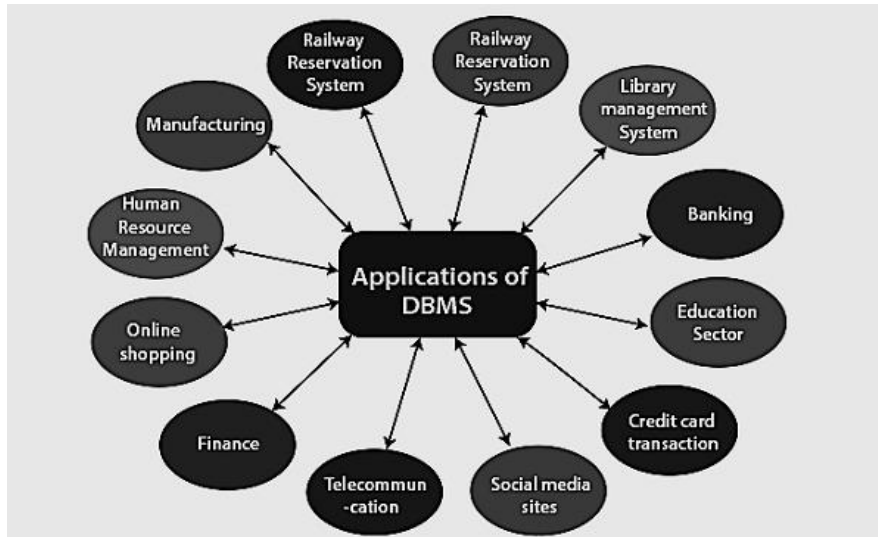
- **Real-world entity:** A modern DBMS is more realistic and uses real-world entities to design its architecture. It uses the behavior and attributes too. For example, a school database may use students as an entity and their age as an attribute.
- **Relation-based tables:** DBMS allows entities and relations among them to form tables. A user can understand the architecture of a database just by looking at the table names.
- **Isolation of data and application:** A database system is entirely different than its data. A database is an active entity, whereas data is said to be passive, on which the

database works and organizes. DBMS also stores metadata, which is data about data, to ease its own process.

- **Less redundancy:** DBMS follows the rules of normalization, which splits a relation when any of its attributes is having redundancy in values. Normalization is a mathematically rich and scientific process that reduces data redundancy.
- **Consistency:** Consistency is a state where every relation in a database remains consistent. There exist methods and techniques, which can detect attempt of leaving database in inconsistent state. A DBMS can provide greater consistency as compared to earlier forms of data storing applications like file-processing systems.
- **Query Language:** DBMS is equipped with query language, which makes it more efficient to retrieve and manipulate data. A user can apply as many and as different filtering options as required to retrieve a set of data. Traditionally it was not possible where file-processing system was used.
- **ACID Properties :** DBMS follows the concepts of **A**tomicity, **C**onsistency, **I**solation, and **D**urability (normally shortened as ACID). These concepts are applied on transactions, which manipulate data in a database. ACID properties help the database stay healthy in multi-transactional environments and in case of failure.
- **Multiuser and Concurrent Access:** DBMS supports multi-user environment and allows them to access and manipulate data in parallel. Though there are restrictions on transactions when users attempt to handle the same data item, but users are always unaware of them.
- **Multiple views:** DBMS offers multiple views for different users. A user who is in the Sales department will have a different view of database than a person working in the Production department. This feature enables the users to have a concentrate view of the database according to their requirements.
- **Security:** Features like multiple views offer security to some extent where users are unable to access data of other users and departments. DBMS offers methods to impose constraints while entering data into the database and retrieving the same at a later stage. DBMS offers many different levels of security features, which enables multiple users to have different views with different features. For example, a user in the Sales department cannot see the data that belongs to the Purchase department. Additionally, it can also be managed how much data of the Sales department should be displayed to the user. Since a DBMS is not saved on the disk as traditional file systems, it is very hard for miscreants to break the code.

1.10 APPLICATIONS OF DBMS

There are different fields where a database management system is utilized. Following are a few applications which utilize the information base administration framework as shown in below figure.



- **Railway Reservation System** – The railway reservation system database plays a very important role by keeping record of ticket booking, train’s departure time and arrival status and also gives information regarding train late to people through the database.
- **Library Management System** – Now-a-days it’s become easy in the Library to track each book and maintain it because of the database. This happens because there are thousands of books in the library. It is very difficult to keep a record of all books in a copy or register. Now DBMS is used to maintain all the information related to book issue dates, name of the book, author and availability of the book.
- **Banking** – Banking is one of the main applications of databases. We all know there will be a thousand transactions through banks daily and we are doing this without going to the bank. This is all possible just because of DBMS that manages all the bank transactions.
- **Universities and colleges** – Now-a-days examinations are done online. So, the universities and colleges are maintaining DBMS to store Student’s registrations details, results, courses and grades i.e. all the information in the database. For example, telecommunications. Without DBMS there is no telecommunication company. DBMS is most useful to these companies to store the call details and monthly postpaid bills.
- **Credit card transactions** – The purchase of items and transactions of credit cards are made possible only by DBMS. A credit card holder has to know the importance of the information which is secured through DBMS.
- **Social Media Sites** – By filling the required details we are able to access social media platforms. Many users sign up daily on social websites such as Facebook, Pinterest and Instagram. All the information related to the users are stored and maintained with the help of DBMS.

- **Finance** – Now-a-days there are lots of things to do with finance like storing sales, holding information and finance statement management etc. these all can be done with database systems.
- **Military** – In military areas the DBMS is playing a vital role. Military keeps records of soldiers and it has so many files that should be kept secure and safe. DBMS provides a high security to military information.
- **Online Shopping** – Now-a-days we all do Online shopping without wasting the time with the help of DBMS. The products are added and sold only with the help of DBMS that further includes Purchase information, invoice bills and payment.
- **Human Resource Management** – The management keeps records of each employee's salary, tax and work through DBMS.
- **Manufacturing** – Manufacturing companies make products and sell them on a daily basis. To keep records of all those details, DBMS is used.
- **Airline Reservation system** – Just like the railway reservation system, airlines also need DBMS to keep records of flights arrival, departure and delay status.

So finally, we can clearly conclude that the DBMS is playing a very important role in each and every field.

1.11 SUMMARY

- Database is a collection of inter-related data which is used to retrieve, insert and delete the data efficiently.
- Database Management System can also be defined as a software system that allows you to define, construct and manipulate databases used for various purposes such as to store customer and financial information of an organization.
- Data consists of facts, figures, statistics etc. having no particular meaning.
- Record is a collection of related data items.
- DBMS supports multi-user environment and allows them to access and manipulate data in parallel.
- A file system is a software that manages and organizes the files in a storage medium. It controls how data is stored and retrieved.
- DBMS architecture depends upon how users are connected to the database to get their request done.
- Relational database system supports mathematical set of operations like union, intersection, difference and Cartesian product.
- The primary goal of a DBMS is to provide a way to store and retrieve database information that is both convenient and efficient
- Mapping is used to transform the request and response between various database levels of architecture.

1.12 REFERENCES

“Fundamentals of Database Systems”, Elmasri Navrate, 6th edition, 2013, Pearson
“Introduction to Database Systems”, C.J.Date, Pearson Education
“Data base System Concepts”, Silberschatz, Korth, McGraw Hill, V edition
Starting with Oracle, by John Day and Craig Van Slyke

1.13 FURTHER READING

Database System Concepts - 6th edition - Avi Silberschatz
Database Systems - A Practical Approach to Design, Implementation & Management
By Thomas Connolly, Carolyn Begg
Fundamentals of Database Systems - Elmasari , Navathe
Database Management Systems By Raghu Ramkrishnan, Gehrke
Database System Concepts, by Henry F. Korth

1.14 QUESTIONS FOR PRACTICE (SHORT & LONG ANSWERS TYPE)

1. What is database management system?
2. What are the various types of database system?
3. What are the applications of database management system?
4. Differentiate between traditional file system and modern DBMS.
5. List the characteristics of database management system.
6. What are the various advantages and disadvantages of hierarchical database system?
7. List the features of database management system.
8. What do you mean by physical data independence?
9. What are the types of DBMS architecture?
10. Name the users of DBMS.

B.Sc.(DATA SCIENCE)
SEMESTER-III
DATABASE MANAGEMENT SYSTEM

UNIT II: DATA MODELS

STRUCTURE

- 2.0 Objectives
- 2.2 Introduction
- 2.2 Data Models
 - 2.2.1 Data model Schema and Instance
- 2.3 Keys in DBMA
 - 2.3.1 Why do we need a Key?
- 2.4 Types of Keys
- 2.5 Difference Between Primary Key & Foreign Key
- 2.6 Integrity Constraints
 - 2.6.1 Types of Integrity Constraints
- 2.7 Relational Model
 - 2.7.1 Codd's 12 Rules of RDBMS
 - 2.7.2 Relational Model Concepts
 - 2.7.3 Properties of Relations
 - 2.7.4 Operations in Relational Model
 - 2.7.5 Equipment with a Query Language
 - 2.7.6 ER Model to Relational Model
 - 2.7.7 Best Practices for Creating a Relational Model
 - 2.7.8 Advantages of using Relational Model
 - 2.7.9 Disadvantages of using Relational Model
- 2.8 Hierarchical Data Model
- 2.9 Network Data Model
- 2.10 Difference between Hierarchical, Network and Relational Data Model
- 2.11 Summary
- 2.12 References
- 2.13 Further Readings
- 2.14 Practice Questions

2.0 OBJECTIVES

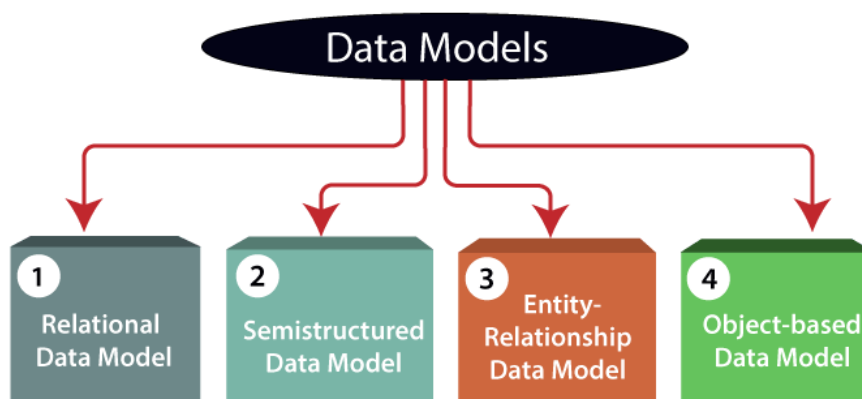
- To provide overview of data model.
- To discuss concept of various data model.
- To explain relational keys of DBMS.
- To discuss difference between primary and foreign keys.
- To explain integrity constraints.
- To discuss concept of relational model.
- To provide description of Codd's 12 Rules of RDBMS.
- To discuss Operations in Relational Model.
- To explain the terms of relational algebra and relational calculus.
- To describe ER Model to Relational Model.
- To describe advantages as well as disadvantages of Relational model.
- To explain hierarchical model and network model.

2.1 INTRODUCTION

A data model defines the logical design and structure of a database and defines how data will be stored, accessed and updated in a database management system. In this unit, we will discuss four types of data models such as Relational Data Model, Semi-Structured Data Model, Entity-Relationship Data Model, Object-based Data Model. This unit also explains the various relational keys such as primary key, foreign key, candidate key etc. We will discuss integrity constraints along with its types. Apart from above mentioned terms, concepts of various models such as relational model along with its rules, hierarchical model and network model will be discussed.

2.2 DATA MODELS

Data Model is the modeling of the data description, data semantics, and consistency constraints of the data. It provides the conceptual tools for describing the design of a database at each level of data abstraction. Therefore, there are following four data models used for understanding the structure of the database as shown in below figure:



1) Relational Data Model: This type of model designs the data in the form of rows and columns within a table. Thus, a relational model uses tables for representing data and in-between relationships. Tables are also called relations. This model was initially described by Edgar F. Codd, in 1969. The relational data model is the widely used model which is primarily used by commercial data processing applications.

2) Semi-Structured Data Model: This type of data model is different from the other three data models (explained above). The semistructured data model allows the data specifications at places where the individual data items of the same type may have different attributes sets. The Extensible Markup Language, also known as XML, is widely used for representing the semistructured data. Although XML was initially designed for including the markup information to the text document, it gains importance because of its application in the exchange of data.

3) Entity-Relationship Data Model: An ER model is the logical representation of data as objects and relationships among them. These objects are known as entities, and relationship is an association among these entities. This model was designed by Peter Chen and published in 1976 papers. It was widely used in database designing. A set of attributes describe the entities. For example, student_name, student_id describes the 'student' entity. A set of the same type of entities is known as an 'Entity set', and the set of the same type of relationships is known as 'relationship set'.

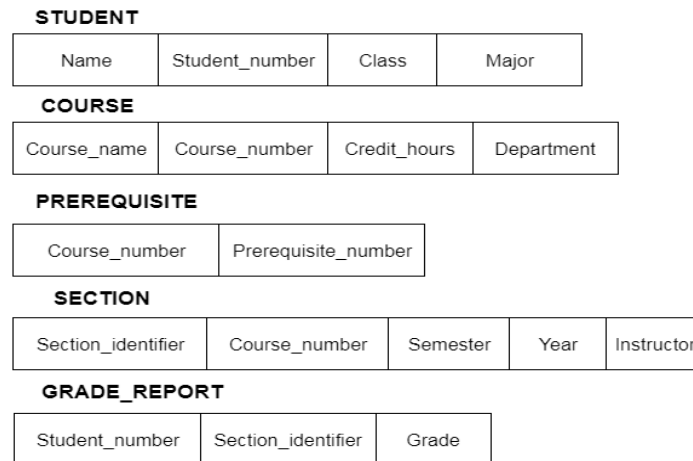
4) Object-based Data Model: An extension of the ER model with notions of functions, encapsulation, and object identity, as well. This model supports a rich type system that includes structured and collection types. Thus, in 1980s, various database systems following the object-oriented approach were developed. Here, the objects are nothing but the data carrying its properties.

Data model Schema and Instance

- The data which is stored in the database at a particular moment of time is called an instance of the database.
- The overall design of a database is called schema.
- A database schema is the skeleton structure of the database. It represents the logical view of the entire database.
- A schema contains schema objects like table, foreign key, primary key, views, columns, data types, stored procedure, etc.
- A database schema can be represented by using the visual diagram. That diagram shows the database objects and relationship with each other.
- A database schema is designed by the database designers to help programmers whose software will interact with the database. The process of database creation is called data modeling.
- A schema diagram can display only some aspects of a schema like the name of record type, data type, and constraints. Other aspects can't be specified through the schema

diagram. For example, the given figure neither show the data type of each data item nor the relationship among various files.

- In the database, actual data changes quite frequently. For example, in the given figure, the database changes whenever we add a new grade or add a student. The data at a particular moment of time is called the instance of the database.



Self-Check Exercise

1. What do you mean by Semi-structured data model ?
2. How many data models are there in DBMS?
3. Define Object-based data model.
4. Define database schema.

2.2 KEYS IN DBMS

It is an attribute or set of attributes which helps you to identify a row(tuple) in a relation(table). They allow you to find the relation between two tables. Keys help you uniquely identify a row in a table by a combination of one or more columns in that table. Key is also helpful for finding unique record or row from the table. Database key is also helpful for finding unique record or row from the table.

Example:

Employee ID	FirstName	LastName
11	Andrew	Johnson
22	Tom	Wood
33	Alex	Hale

In the above-given example, employee ID is a primary key because it uniquely identifies an employee record. In this table, no other employee can have the same employee ID.

2.3.1 Why do we need a Key?

Here are some reasons for using keys in the DBMS system.

- Keys help you to identify any row of data in a table. In a real-world application, a table could contain thousands of records. Moreover, the records could be duplicated. Keys in RDBMS ensure that you can uniquely identify a table record despite these challenges.
- It allows you to establish a relationship between and identify the relation between tables
- It helps you to enforce identity and integrity in the relationship.

2.4 TYPES OF KEYS IN DBMS

There are mainly Eight different types of Keys in DBMS and each key has it's different functionality:

1. Super Key
2. Primary Key
3. Candidate Key
4. Alternate Key
5. Foreign Key
6. Compound Key
7. Composite Key
8. Surrogate Key

Let's look at each of the keys in DBMS with example:

- **Super Key** - A super key is a group of single or multiple keys which identifies rows in a table.
- **Primary Key** - is a column or group of columns in a table that uniquely identify every row in that table.
- **Candidate Key** - is a set of attributes that uniquely identify tuples in a table. Candidate Key is a super key with no repeated attributes.
- **Alternate Key** - is a column or group of columns in a table that uniquely identify every row in that table.
- **Foreign Key** - is a column that creates a relationship between two tables. The purpose of Foreign keys is to maintain data integrity and allow navigation between two different instances of an entity.
- **Compound Key** - has two or more attributes that allow you to uniquely recognize a specific record. It is possible that each column may not be unique by itself within the database.

- **Composite Key** - is a combination of two or more columns that uniquely identify rows in a table. The combination of columns guarantees uniqueness, though individual uniqueness is not guaranteed.
- **Surrogate Key** - An artificial key which aims to uniquely identify each record is called a surrogate key. These kind of key are unique because they are created when you don't have any natural primary key.

2.4.1 Super Key

A superkey is a group of single or multiple keys which identifies rows in a table. A Super key may have additional attributes that are not needed for unique identification.

Example:

EmpSSN	EmpNum	Empname
9812345098	AB05	Shown
9876512345	AB06	Roslyn
199937890	AB07	James

In the above-given example, EmpSSN and EmpNum name are superkeys.

2.4.2 Primary Key

PRIMARY KEY in DBMS is a column or group of columns in a table that uniquely identify every row in that table. The Primary Key can't be a duplicate meaning the same value can't appear more than once in the table. A table cannot have more than one primary key.

Rules for defining Primary key:

- Two rows can't have the same primary key value
- It must for every row to have a primary key value.
- The primary key field cannot be null.
- The value in a primary key column can never be modified or updated if any foreign key refers to that primary key.

Example:

In the following example, `StudID` is a Primary Key.

StudID	Roll No	First Name	LastName	Email
1	11	Tom	Price	<u>abc@gmail.com</u>
2	12	Nick	Wright	<u>xyz@gmail.com</u>
3	13	Dana	Natan	<u>mno@yahoo.com</u>

2.4.3 Alternate Key

ALTERNATE KEY is a column or group of columns in a table that uniquely identify every row in that table. A table can have multiple choices for a primary key but only one can be set as the primary key. All the keys which are not primary key are called an Alternate Key.

Example:

In this table, StudID, Roll No, Email are qualified to become a primary key. But since StudID is the primary key, Roll No, Email becomes the alternative key.

StudID	Roll No	First Name	LastName	Email
1	11	Tom	Price	<u>abc@gmail.com</u>
2	12	Nick	Wright	<u>xyz@gmail.com</u>
3	13	Dana	Natan	<u>mno@yahoo.com</u>

2.4.4 Candidate Key

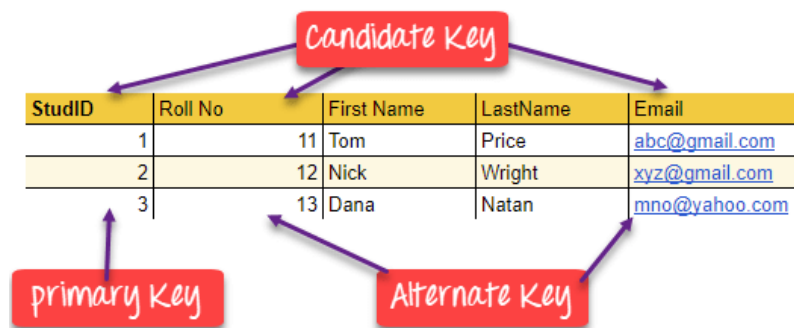
CANDIDATE KEY in SQL is a set of attributes that uniquely identify tuples in a table. Candidate Key is a super key with no repeated attributes. The Primary key should be selected from the candidate keys. Every table must have at least a single candidate key. A table can have multiple candidate keys but only a single primary key.

Properties of Candidate Key

- It must contain unique values
- Candidate key in SQL may have multiple attributes
- Must not contain null values
- It should contain minimum fields to ensure uniqueness
- Uniquely identify each record in a table

Candidate key Example: In the given table Stud ID, Roll No, and email are candidate keys which help us to uniquely identify the student record in the table.

StudID	Roll No	First Name	LastName	Email
1	11	Tom	Price	abc@gmail.com
2	12	Nick	Wright	xyz@gmail.com
3	13	Dana	Natan	mno@yahoo.com



Candidate Key in DBMS

2.4.5 Foreign Key

FOREIGN KEY is a column that creates a relationship between two tables. The purpose of Foreign keys is to maintain data integrity and allow navigation between two different instances of an entity. It acts as a cross-reference between two tables as it references the primary key of another table.

Example:

DeptCode	DeptName
001	Science
002	English
005	Computer

Teacher ID	Fname	Lname
B002	David	Warner
B017	Sara	Joseph
B009	Mike	Brunton

In this key in dbms example, we have two tables, teacher and department in a school. However, there is no way to see which search work in which department.

In this table, adding the foreign key in Deptcode to the Teacher name, we can create a relationship between the two tables.

Teacher ID	DeptCode	Fname	Lname
B002	002	David	Warner
B017	002	Sara	Joseph
B009	001	Mike	Brunton

This concept is also known as Referential Integrity.

2.4.6 Compound Key

COMPOUND KEY has two or more attributes that allow you to uniquely recognize a specific record. It is possible that each column may not be unique by itself within the database. However, when combined with the other column or columns the combination of composite keys become unique. The purpose of the compound key in database is to uniquely identify each record in the table.

Example:

OrderNo	ProductID	Product Name	Quantity
B005	JAP102459	Mouse	5
B005	DKT321573	USB	10
B005	OMG446789	LCD Monitor	20
B004	DKT321573	USB	15
B002	OMG446789	Laser Printer	3

In this example, OrderNo and ProductID can't be a primary key as it does not uniquely identify a record. However, a compound key of Order ID and Product ID could be used as it uniquely identified each record.

2.4.7 Composite Key

COMPOSITE KEY is a combination of two or more columns that uniquely identify rows in a table. The combination of columns guarantees uniqueness, though individually uniqueness is not guaranteed. Hence, they are combined to uniquely identify records in a table.

The difference between compound and the composite key is that any part of the compound key can be a foreign key, but the composite key may or maybe not a part of the foreign key.

2.4.8 Surrogate Key?

SURROGATE KEY is an artificial key which aims to uniquely identify each record is called a surrogate key. This kind of partial key in dbms is unique because it is created when you don't have any natural primary key. They do not lend any meaning to the data in the table. Surrogate key in DBMS is usually an integer. A surrogate key is a value generated right before the record is inserted into a table.

Fname	Lastname	Start Time	End Time
Anne	Smith	09:00	18:00

Jack	Francis	08:00	17:00
Anna	McLean	11:00	20:00
Shown	Willam	14:00	23:00

Above, given example, shown shift timings of the different employee. In this example, a surrogate key is needed to uniquely identify each employee.

Surrogate Keys in sql are Allowed When

- No property has the parameter of the primary key.
- In the table when the primary key is too big or complicated.

Self-Check Exercise

1. What do you mean by Key ?
2. How does key help you in relational database management system?
3. What is a Foreign key? Give example.
4. Define properties of Candidate key.
5. What is meant by Surrogate Key? Give example.

2.5 DIFFERENCE BETWEEN PRIMARY KEY & FOREIGN KEY

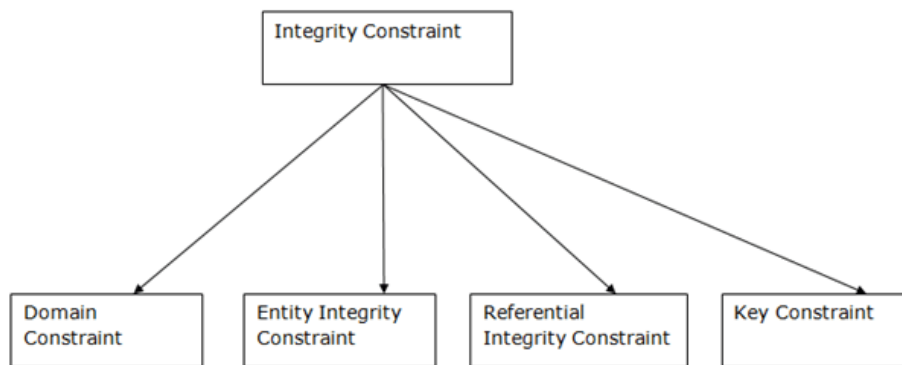
Following is the main difference between primary key and foreign key:

Primary Key	Foreign Key
Helps you to uniquely identify a record in the table.	It is a field in the table that is the primary key of another table.
Primary Key never accept null values.	A foreign key may accept multiple null values.
Primary key is a clustered index and data in the DBMS table are physically organized in the sequence of the clustered index.	A foreign key cannot automatically create an index, clustered or non-clustered. However, you can manually create an index on the foreign key.
You can have the single Primary key in a table.	You can have multiple foreign keys in a table.

2.6 INTEGRITY CONSTRAINTS

- Integrity constraints are a set of rules. It is used to maintain the quality of information.
- Integrity constraints ensure that the data insertion, updating, and other processes have to be performed in such a way that data integrity is not affected.
- Thus, integrity constraint is used to guard against accidental damage to the database.

2.6.1 Types of Integrity Constraints



Domain Constraints

- Domain constraints can be defined as the definition of a valid set of values for an attribute.
- The data type of domain includes string, character, integer, time, date, currency, etc. The value of the attribute must be available in the corresponding domain.

Example:

ID	NAME	SEMENSTER	AGE
1000	Tom	1 st	17
1001	Johnson	2 nd	24
1002	Leonardo	5 th	21
1003	Kate	3 rd	19
1004	Morgan	8 th	A

Not allowed. Because AGE is an integer attribute

2. Entity Integrity Constraints

- The entity integrity constraint states that primary key value can't be null.
- This is because the primary key value is used to identify individual rows in relation and if the primary key has a null value, then we can't identify those rows.
- A table can contain a null value other than the primary key field.

Example:

EMPLOYEE

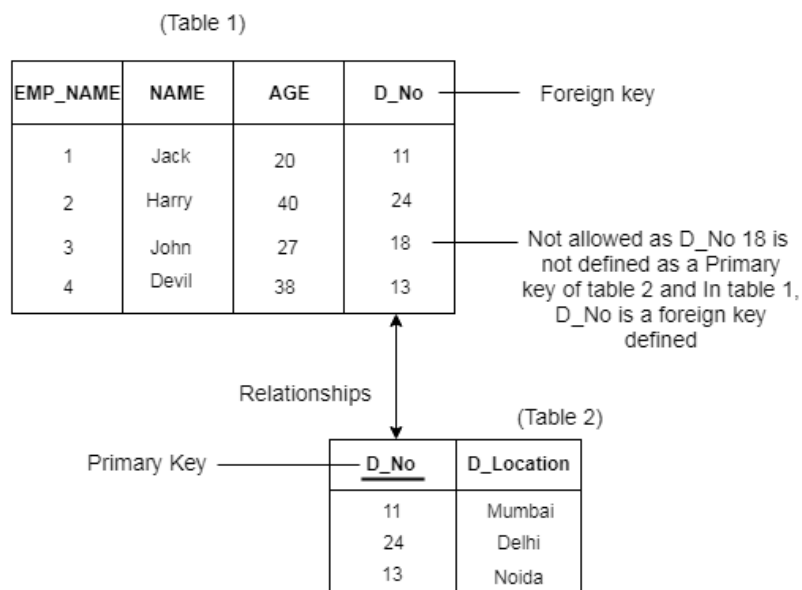
EMP_ID	EMP_NAME	SALARY
123	Jack	30000
142	Harry	60000
164	John	20000
	Jackson	27000

Not allowed as primary key can't contain a NULL value

3. Referential Integrity Constraints

- A referential integrity constraint is specified between two tables.
- In the Referential integrity constraints, if a foreign key in Table 1 refers to the Primary Key of Table 2, then every value of the Foreign Key in Table 1 must be null or be available in Table 2.

Example:



4. Key Constraints

- Keys are the entity set that is used to identify an entity within its entity set uniquely.
- An entity set can have multiple keys, but out of which one key will be the primary key. A primary key can contain a unique and null value in the relational table.

Example

ID	NAME	SEMENSTER	AGE
1000	Tom	1 st	17
1001	Johnson	2 nd	24
1002	Leonardo	5 th	21
1003	Kate	3 rd	19
1002	Morgan	8 th	22

Not allowed. Because all row must be unique

2.7 RELATIONAL MODEL

- **Relational Model (RM)** represents the database as a collection of relations. A relation is nothing but a table of values. Every row in the table represents a collection of related data values. These rows in the table denote a real-world entity or relationship.
- The table name and column names are helpful to interpret the meaning of values in each row.
- The data are represented as a set of relations. In the relational model, data are stored as tables.
- However, the physical storage of the data is independent of the way the data are logically organized.

Some popular Relational Database management systems are:

- DB2 and Informix Dynamic Server - IBM
- Oracle and RDB – Oracle
- SQL Server and Access – Microsoft

2.7.1 Codd's 12 Rules of RDBMS

- Dr Edgar F. Codd, after his extensive research on the Relational Model of database systems, came up with twelve rules of his own, which according to him, a database must obey in order to be regarded as a true relational database.
- These rules can be applied on any database system that manages stored data using only its relational capabilities. This is a foundation rule, which acts as a base for all the other rules.

Rule 1: Information Rule

- The data stored in a database, may it be user data or metadata, must be a value of some table cell. Everything in a database must be stored in a table format.

Rule 2: Guaranteed Access Rule

- Every single data element (value) is guaranteed to be accessible logically with a combination of table-name, primary-key (row value), and attribute-name (column value). No other means, such as pointers, can be used to access data.

Rule 3: Systematic Treatment of NULL Values

- The NULL values in a database must be given a systematic and uniform treatment. This is a very important rule because a NULL can be interpreted as one the following – data is missing, data is not known, or data is not applicable.

Rule 4: Active Online Catalog

- The structure description of the entire database must be stored in an online catalog, known as **data dictionary**, which can be accessed by authorized users. Users can use the same query language to access the catalog which they use to access the database itself.

Rule 5: Comprehensive Data Sub-Language Rule

- A database can only be accessed using a language having linear syntax that supports data definition, data manipulation, and transaction management operations. This language can be used directly or by means of some application. If the database allows access to data without any help of this language, then it is considered as a violation.

Rule 6: View Updating Rule

- All the views of a database, which can theoretically be updated, must also be updatable by the system.

Rule 7: High-Level Insert, Update, and Delete Rule

- A database must support high-level insertion, updation, and deletion. This must not be limited to a single row, that is, it must also support union, intersection and minus operations to yield sets of data records.

Rule 8: Physical Data Independence

- The data stored in a database must be independent of the applications that access the database. Any change in the physical structure of a database must not have any impact on how the data is being accessed by external applications.

Rule 9: Logical Data Independence

- The logical data in a database must be independent of its user's view (application). Any change in logical data must not affect the applications using it. For example, if two tables are merged or one is split into two different tables, there should be no

impact or change on the user application. This is one of the most difficult rule to apply.

Rule 10: Integrity Independence

- A database must be independent of the application that uses it. All its integrity constraints can be independently modified without the need of any change in the application. This rule makes a database independent of the front-end application and its interface.

Rule 11: Distribution Independence

- The end-user must not be able to see that the data is distributed over various locations. Users should always get the impression that the data is located at one site only. This rule has been regarded as the foundation of distributed database systems.

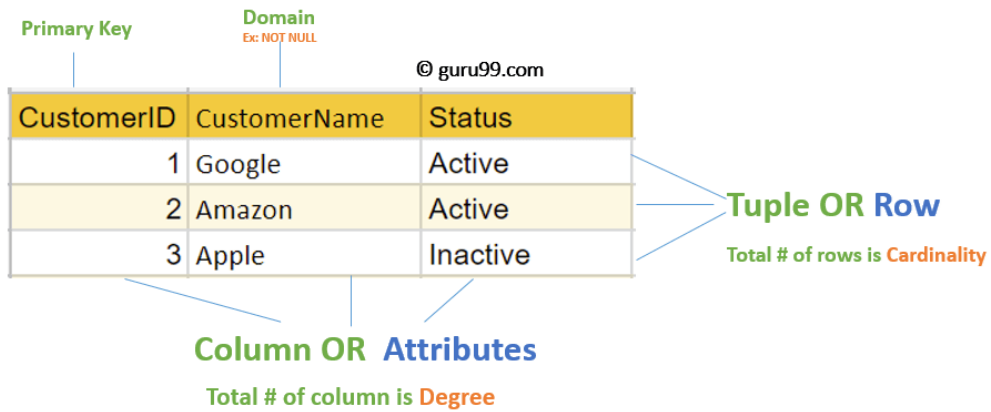
Rule 12: Non-Subversion Rule

- If a system has an interface that provides access to low-level records, then the interface must not be able to subvert the system and bypass security and integrity constraints.

2.7.2 Relational Model Concepts

1. **Attribute:** It contains the name of a column in a particular table. Attributes are the properties which define a relation. e.g., Student_Rollno, NAME,etc.
2. **Tables** – In the Relational model, the relations are saved in the table format. It is stored along with its entities. A table has two properties rows and columns. Rows represent records and columns represent attributes.
3. **Tuple** – It is nothing but a single row of a table, which contains a single record.
4. **Relation Schema:** A relation schema represents the name of the relation and name of all columns or attributes.
5. **Degree:** The total number of attributes which in the relation is called the degree of the relation.
6. **Cardinality:** Total number of rows present in the Table.
7. **Column:** The column represents the set of values for a specific attribute.
8. **Relation instance** – Relation instance is a finite set of tuples in the RDBMS system. Relation instances never have duplicate tuples.
9. **Relation key** - In the relational key, each row has one or more attributes. It can identify the row in the relation uniquely.
10. **Attribute domain** – It contains a set of atomic values that an attribute can take.

Table also called Relation



Example: STUDENT Relation

NAME	ROLL_NO	PHONE_NO	ADDRESS	AGE
Ram	14795	7305758992	Noida	24
Shyam	12839	9026288936	Delhi	35
Laxman	33289	8583287182	Gurugram	20
Mahesh	27857	7086819134	Ghaziabad	27
Ganesh	17282	9028 9i3988	Delhi	40

- In the given table, NAME, ROLL_NO, PHONE_NO, ADDRESS, and AGE are the attributes.
- The instance of schema STUDENT has 5 tuples.
- t3 = <Laxman, 33289, 8583287182, Gurugram, 20>

2.7.3 Properties of Relations

- Name of the relation is distinct from all other relations.
- Each relation cell contains exactly one atomic (single) value
- Each attribute contains a distinct name
- Attribute domain has no significance
- tuple has no duplicate value
- Order of tuple can have a different sequence

2.7.4 Operations in Relational Model

Four basic update operations performed on relational database model are as follows:

Insert, update, delete and select.


- Insert is used to insert data into the relation
- Delete is used to delete tuples from the table.
- Modify allows you to change the values of some attributes in existing tuples.
- Select allows you to choose a specific range of data.

Whenever one of these operations are applied, integrity constraints specified on the relational database schema must never be violated.

Insert Operation

The insert operation gives values of the attribute for a new tuple which should be inserted into a relation.

CustomerID	CustomerName	Status
1	Google	Active
2	Amazon	Active
3	Apple	Inactive



CustomerID	CustomerName	Status
1	Google	Active
2	Amazon	Active
3	Apple	Inactive
4	Alibaba	Active

Update Operation

You can see that in the below-given relation table CustomerName= 'Apple' is updated from Inactive to Active.

CustomerID	CustomerName	Status
1	Google	Active
2	Amazon	Active
3	Apple	Inactive
4	Alibaba	Active




CustomerID	CustomerName	Status
1	Google	Active
2	Amazon	Active
3	Apple	Active
4	Alibaba	Active

Delete Operation

To specify deletion, a condition on the attributes of the relation selects the tuple to be deleted.

CustomerID	CustomerName	Status
1	Google	Active
2	Amazon	Active
3	Apple	Active
4	Alibaba	Active



CustomerID	CustomerName	Status
1	Google	Active
2	Amazon	Active
4	Alibaba	Active

In the above-given example, CustomerName= "Apple" is deleted from the table.

The Delete operation could violate referential integrity if the tuple which is deleted is referenced by foreign keys from other tuples in the same database.

Select Operation

CustomerID	CustomerName	Status
1	Google	Active
2	Amazon	Active
4	Alibaba	Active



CustomerID	CustomerName	Status
2	Amazon	Active

In the above-given example, CustomerName="Amazon" is selected

2.7.5 Equipment with a query language

- Relational database systems are expected to be equipped with a query language that can assist its users to query the database instances.
- There are two kinds of query languages – relational algebra and relational calculus.

2.7.5.1 Relational Algebra

Relational algebra is a procedural query language, which takes instances of relations as input and yields instances of relations as output.

It uses operators to perform queries. An operator can be either **unary** or **binary**. They accept relations as their input and yield relations as their output. Relational algebra is performed recursively on a relation and intermediate results are also considered relations.

The fundamental operations of relational algebra are as follows –

- Select
- Project
- Union
- Set difference
- Cartesian product
- Rename

We will discuss all these operations in the following sections.

- **Select Operation (σ)**

It selects tuples that satisfy the given predicate from a relation.

Notation – $\sigma_p(r)$

Where σ stands for selection predicate and r stands for relation. p is propositional logic formula which may use connectors like **and**, **or**, and **not**. These terms may use relational operators like $=$, \neq , \geq , $<$, $>$, \leq .

For example –

$\sigma_{\text{subject} = \text{"database"}}(\text{Books})$

Output – Selects tuples from books where subject is 'database'.

$\sigma_{\text{subject} = \text{"database"} \text{ and price} = \text{"450"}}(\text{Books})$

Output – Selects tuples from books where subject is 'database' and 'price' is 450.

$\sigma_{\text{subject} = \text{"database"} \text{ and price} = \text{"450"} \text{ or year} > \text{"2010"}}(\text{Books})$

Output – Selects tuples from books where subject is 'database' and 'price' is 450 or those books published after 2010.

- **Project Operation (Π)**

It projects column(s) that satisfy a given predicate.

Notation – $\Pi_{A_1, A_2, A_n}(r)$

Where A_1, A_2, A_n are attribute names of relation r .

Duplicate rows are automatically eliminated, as relation is a set.

For example –

$\Pi_{\text{subject, author}}(\text{Books})$

Selects and projects columns named as subject and author from the relation Books.

- **Union Operation (\cup)**

It performs binary union between two given relations and is defined as –

$r \cup s = \{ t \mid t \in r \text{ or } t \in s \}$

Notation – $r \cup s$

Where r and s are either database relations or relation result set (temporary relation).

For a union operation to be valid, the following conditions must hold –

- r , and s must have the same number of attributes.
- Attribute domains must be compatible.
- Duplicate tuples are automatically eliminated.

$\Pi_{\text{author}}(\text{Books}) \cup \Pi_{\text{author}}(\text{Articles})$

Output – Projects the names of the authors who have either written a book or an article or both.

- **Set Difference ($-$)**

The result of set difference operation is tuples, which are present in one relation but are not in the second relation.

Notation – $r - s$

Finds all the tuples that are present in **r** but not in **s**.

$\Pi_{\text{author}}(\text{Books}) - \Pi_{\text{author}}(\text{Articles})$

Output – Provides the name of authors who have written books but not articles.

- **Cartesian Product (X)**

Combines information of two different relations into one.

Notation – $r \times s$

Where **r** and **s** are relations and their output will be defined as –

$r \times s = \{ q \ t \mid q \in r \text{ and } t \in s \}$

$\sigma_{\text{author} = 'abc'}(\text{Books} \times \text{Articles})$

Output – Yields a relation, which shows all the books and articles written by abc.

- **Rename Operation (ρ)**

The results of relational algebra are also relations but without any name. The rename operation allows us to rename the output relation. 'rename' operation is denoted with small Greek letter **rho** ρ .

Notation – $\rho_x(E)$

Where the result of expression **E** is saved with name of **x**.

Additional operations are –

- Set intersection
- Assignment
- Natural join

2.7.5.2 Relational Calculus

In contrast to Relational Algebra, Relational Calculus is a non-procedural query language, that is, it tells what to do but never explains how to do it.

Relational calculus exists in two forms –

- **Tuple Relational Calculus (TRC)**

Filtering variable ranges over tuples

Notation – $\{T \mid \text{Condition}\}$

Returns all tuples T that satisfies a condition.

For example –

$\{ T.\text{name} \mid \text{Author}(T) \text{ AND } T.\text{article} = 'database' \}$

Output – Returns tuples with 'name' from Author who has written article on 'database'.

TRC can be quantified. We can use Existential (\exists) and Universal Quantifiers (\forall).

For example –

$\{ R \mid \exists T \in \text{Authors}(T.\text{article}='database' \text{ AND } R.\text{name}=T.\text{name}) \}$

Output – The above query will yield the same result as the previous one.

- **Domain Relational Calculus (DRC)**

In DRC, the filtering variable uses the domain of attributes instead of entire tuple values (as done in TRC, mentioned above).

Notation –

$\{ a_1, a_2, a_3, \dots, a_n \mid P(a_1, a_2, a_3, \dots, a_n) \}$

Where a_1, a_2 are attributes and **P** stands for formulae built by inner attributes.

For example –

$\{ \langle \text{article}, \text{page}, \text{subject} \rangle \mid \in \text{abc} \wedge \text{subject} = 'database' \}$

Output – Yields Article, Page, and Subject from the relation abc, where subject is database.

Just like TRC, DRC can also be written using existential and universal quantifiers. DRC also involves relational operators.

The expression power of Tuple Relation Calculus and Domain Relation Calculus is equivalent to Relational Algebra.

2.7.6 ER Model to Relational Model

ER Model, when conceptualized into diagrams, gives a good overview of entity-relationship, which is easier to understand. ER diagrams can be mapped to relational schema, that is, it is possible to create relational schema using ER diagram. We cannot import all the ER constraints into relational model, but an approximate schema can be generated.

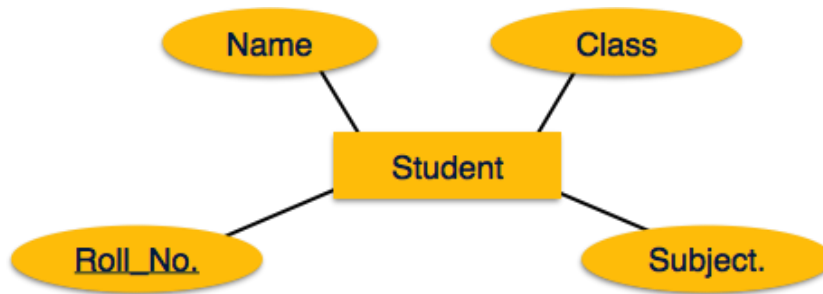
There are several processes and algorithms available to convert ER Diagrams into Relational Schema. Some of them are automated and some of them are manual. We may focus here on the mapping diagram contents to relational basics.

ER diagrams mainly comprise of –

- Entity and its attributes
- Relationship, which is association among entities.

2.7.6.1 Mapping Entity

An entity is a real-world object with some attributes.

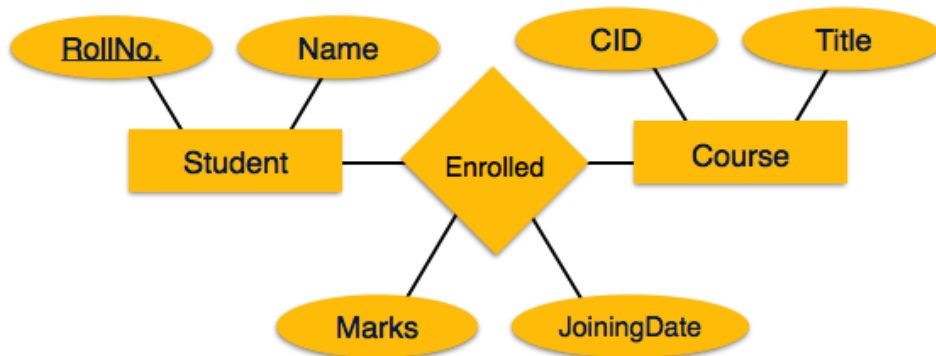


Mapping Process (Algorithm)

- Create table for each entity.
- Entity's attributes should become fields of tables with their respective data types.
- Declare primary key.

2.7.6.2 Mapping Relationship

A relationship is an association among entities.

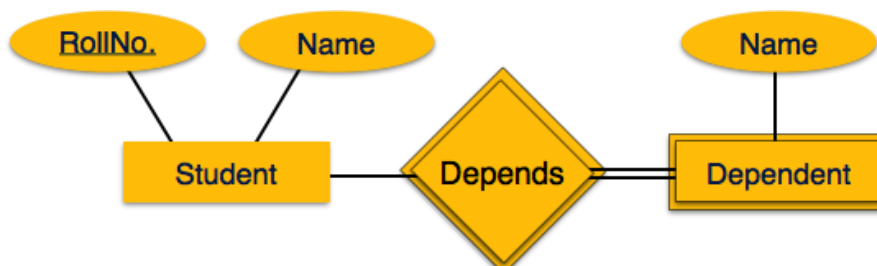


Mapping Process

- Create table for a relationship.
- Add the primary keys of all participating Entities as fields of table with their respective data types.
- If relationship has any attribute, add each attribute as field of table.
- Declare a primary key composing all the primary keys of participating entities.
- Declare all foreign key constraints.

2.7.6.3 Mapping Weak Entity Sets

A weak entity set is one which does not have any primary key associated with it.

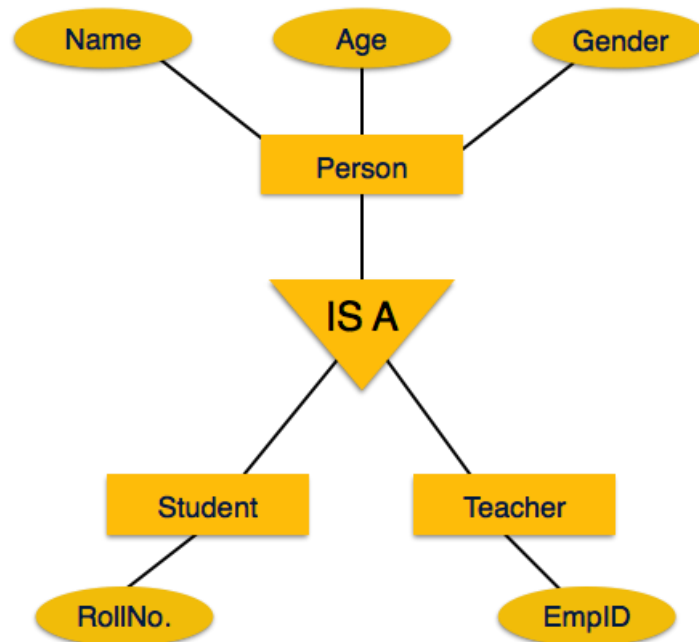


Mapping Process

- Create table for weak entity set.
- Add all its attributes to table as field.
- Add the primary key of identifying entity set.
- Declare all foreign key constraints.

2.7.6.4 Mapping Hierarchical Entities

ER specialization or generalization comes in the form of hierarchical entity sets.



Mapping Process

- Create tables for all higher-level entities.
- Create tables for lower-level entities.
- Add primary keys of higher-level entities in the table of lower-level entities.
- In lower-level tables, add all other attributes of lower-level entities.
- Declare primary key of higher-level table and the primary key for lower-level table.
- Declare foreign key constraints.

2.7.7 Best Practices for creating a Relational Model

- Data needs to be represented as a collection of relations
- Each relation should be depicted clearly in the table
- Rows should contain data about instances of an entity
- Columns must contain data about attributes of the entity
- Cells of the table should hold a single value
- Each column should be given a unique name

- No two rows can be identical
- The values of an attribute should be from the same domain

2.7.8 Advantages of using Relational Model

- **Simplicity:** A Relational data model in DBMS is simpler than the hierarchical and network model.
- **Structural Independence:** The relational database is only concerned with data and not with a structure. This can improve the performance of the model.
- **Easy to use:** The Relational model in DBMS is easy as tables consisting of rows and columns are quite natural and simple to understand
- **Query capability:** It makes possible for a high-level query language like SQL to avoid complex database navigation.
- **Data independence:** The Structure of Relational database can be changed without having to change any application.
- **Scalable:** Regarding a number of records, or rows, and the number of fields, a database should be enlarged to enhance its usability.

2.7.9 Disadvantages of using Relational Model

- Few relational databases have limits on field lengths which can't be exceeded.
- Relational databases can sometimes become complex as the amount of data grows, and the relations between pieces of data become more complicated.
- Complex relational database systems may lead to isolated databases where the information cannot be shared from one system to another.

Self-Check Exercise

1. Explain the concept of ER Model to Relational Model.
2. What is meant by the rule of distribution independence.
3. What is cardinality in relational model.
4. How can you describe relation schema?
5. Write the difference between relation instance and relation key.

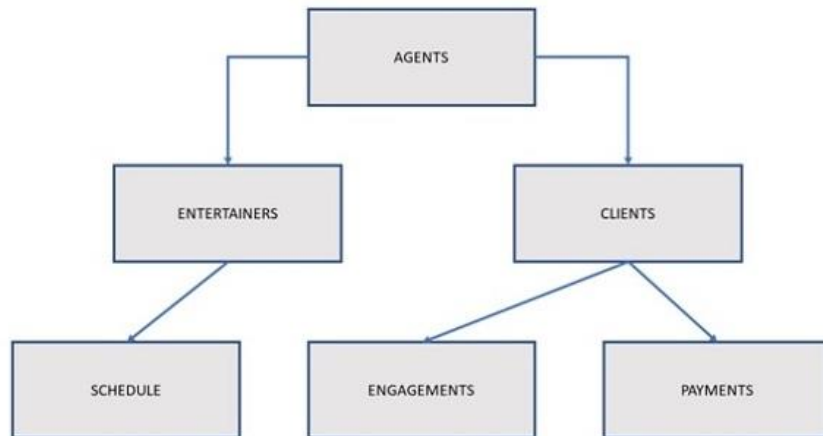
2.8 HIERARCHICAL DATA MODEL

A hierarchical model represents the data in a tree-like structure in which there is a single parent for each record. To maintain order there is a sort field which keeps sibling nodes into a recorded manner. These types of models are designed basically for the early mainframe database management systems, like the Information Management System (IMS) by IBM.

This model structure allows the one-to-one and a one-to-many relationship between two/ various types of data. This structure is very helpful in describing many relationships in the real world; table of contents, any nested and sorted information.

The hierarchical structure is used as the physical order of records in storage. One can access the records by navigating down through the data structure using pointers which are combined with sequential accessing. Therefore, the hierarchical structure is not suitable for certain database operations when a full path is not also included for each record.

Data in this type of database is structured hierarchically and is typically developed as an inverted tree. The "root" in the structure is a single table in the database and other tables act as the branches flowing from the root. The diagram below shows a typical hierarchical database structure.



Agents Database

In the above diagram, an agent books several entertainers, and each entertainer, in return has his/her own schedule. It is the duty of an agent to maintain several clients whose entertainment needs are to be met. A client books engagement through the agent and makes payments to the agent for his services.

A relationship in this database model is represented by the term parent/child. A parent table can be linked with one or more child tables in this type of relationship, but a single child table can be linked with only one parent table. The tables are explicitly linked via a pointer/index or by the physical arrangement of the records within the tables.

A user can access the data by starting at the root table and working down through the tree to the target data. the user must be familiar with the structure of the database to access the data without any complexity.

Advantages

- A user can retrieve data very quickly due to the presence of explicit links between the table structures.
- The referential integrity is built in and automatically enforced due to which a record in a child table must be linked to an existing record in a parent table, along with that if a record deleted in the parent table then that will cause all associated records in the child table to be deleted as well.

Disadvantages

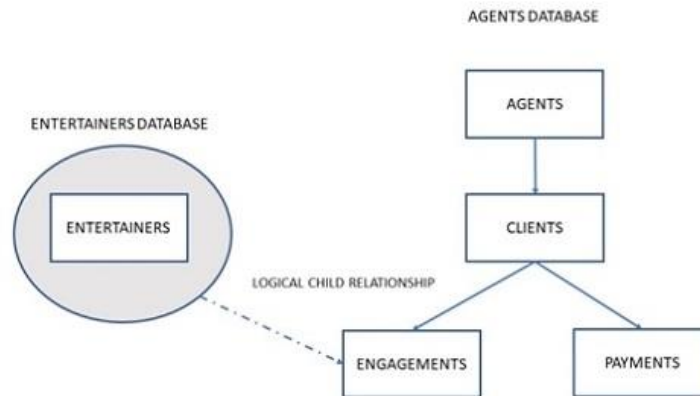
- When a user needs to store a record in a child table that is currently unrelated to any record in a parent table, it gets difficult in recording and user must record an additional entry in the parent table.
- This type of database cannot support complex relationships, and there is also a problem of redundancy, which can result in producing inaccurate information due to the inconsistent recording of data at various sites.

Consider an example using the database diagram shown in the previous diagram. A user cannot enter a new record for the entertainer in the Entertainers table until the entertainer is assigned to a specific agent in the Agents table since a record in a child table (Entertainers) must be related to a record in the parent table (Agents). Therefore, this type of database suffers from the problem of redundant data. For example, if there is a many-to-many relationship between clients and entertainers; an entertainer will perform for many clients, and a client will hire many entertainers. This type of relationship in a hierarchical database cannot easily model, so developers must introduce redundant data into both the Schedule and Engagements tables.

- The Schedule table will now have client data which contains information such as client name, address, and phone number to show for whom and where each entertainer is performing. This data is redundant because it is currently stored also in the Clients table.
- The Engagements table will now contain data on entertainers which contains information such as entertainer name, phone number, and type of entertainer to indicate which entertainers are performing for a given client. This data is also redundant because it is currently stored in the Entertainers table.

The problem with this redundancy is that it can result in producing inaccurate information because it opens the possibility of allowing a user to enter a single piece of data inconsistently.

This problem can be solved by creating one hierarchical database specifically for entertainers and another one specifically for agents. The Entertainers database will contain only the data recorded in the Entertainers table, and the revised Agents database will contain the data recorded in Agents, Clients, Payments, and Engagements tables. There is no need of as you can define a logical child relationship between the Engagements table in the Agents database and the Entertainers table in the Entertainers database. With this relationship in place, you can retrieve a variety of information, such as a list of booked entertainers for a given client or a performance schedule for a given entertainer. The below diagram describes the whole picture.



The hierarchical database suited well to the tape storage systems which is used by mainframes in the 1970s and was very popular in organizations whose database is based on those systems. But, even though the hierarchical database provided fast and direct access to data and was useful in several circumstances, it was clear that a new database model was needed to address the growing problems of data redundancy and complex relationships among data.

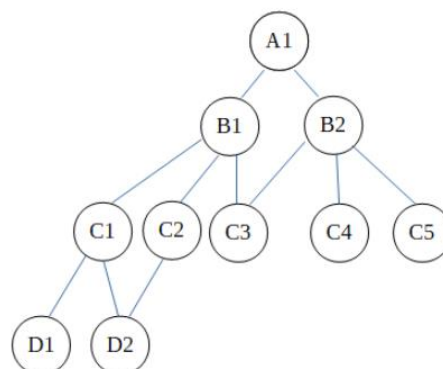
The idea behind this database model is useful for a certain type of data storage, but it is not extremely versatile and is confined to some specific uses.

For example, where each individual person in a company may report to a given department, the department can be used as a parent record and the individual employees will represent secondary records, each of which links back to that one parent record in a hierarchical structure.

2.9 NETWORK DATA MODEL

The network database model was created to solve the shortcomings of the hierarchical database model. In this type of model, a child can be linked to multiple parents, a feature that was not supported by the hierarchical data model. The parent nodes are known as owners and the child nodes are called members.

The network data model can be represented as –



Advantages of Network Model

The network model can support many to many relationships as seen in the diagram. D2 and C3 each have multiple masters. The masters for D2 are C1 and C2 while for C3 are B1 and B2. In this way, the network data model can handle many to many relationships where the hierarchical data model didn't.

Disadvantages of Network Model

There are some disadvantages in the network model even though it is an improvement over the hierarchical model. These are –

- The network model is much more complicated than the Hierarchical model. As such, it is difficult to handle and maintain.
- Although the Network model is more flexible than the Hierarchical model, it still has flexibility problems. Not all relations can be handled by assigning them in the form of owners and members.
- The structure of the Network Model is quite complicated and so the programmer has to understand it well in order to implement or modify it.

2.10 Difference between Hierarchical, Network and Relational Data Model :

Hierarchical Data Model	Network Data Model	Relational Data Model
In this model, to store data hierarchy method is used. It is the oldest method and not in use today.	It organizes records to one another through links or pointers.	It organizes records in the form of table and relationship between tables are set using common fields.
To organize records, it uses tree structure.	It organizes records in the form of directed graphs.	It organizes records in the form of tables.
It implements 1:1 and 1:n relations.	In addition to 1:1 and 1:n it also implements many to many relationships.	In addition to 1:1 and 1:n it also implements many to many relationships.
Insertion anomaly exists in this model i.e. child node cannot be inserted without the parent node.	There is no insertion anomaly.	There is no insertion anomaly.

Hierarchical Data Model	Network Data Model	Relational Data Model
Deletion anomaly exists in this model i.e. it is difficult to delete the parent node.	There is no deletion anomaly.	There is no deletion anomaly.
This model lacks data independence.	There is partial data independence in this model.	This model provides data independence.
It is used to access the data which is complex and asymmetric.	It is used to access the data which is complex and symmetric.	It is used to access the data which is complex and symmetric.
&XML and XAML use this model.	VAX-DBMS, DMS-1100 of UNIVAC and SUPRADBMS's use this model.	It is mostly used in real world applications. Oracle, SQL.

2.11 SUMMARY

- A key in DBMS is an attribute or set of attributes which helps you to identify a row(tuple) in a relation(table)
- Keys in RDBMS allow you to establish a relationship between and identify the relation between tables
- Eight types of key in DBMS are Super, Primary, Candidate, Alternate, Foreign, Compound, Composite, and Surrogate Key.
- A super key is a group of single or multiple keys which identifies rows in a table.
- A column or group of columns in a table which helps us to uniquely identifies every row in that table is called a primary key
- All the different keys in DBMS which are not primary key are called an alternate key
- A super key with no repeated attribute is called candidate key
- A compound key is a key which has many fields which allow you to uniquely recognize a specific record
- A key which has multiple attributes to uniquely identify rows in a table is called a composite key
- An artificial key which aims to uniquely identify each record is called a surrogate key
- Primary Key never accept null values while a foreign key may accept multiple null values.

- The Relational database modelling represents the database as a collection of relations (tables)
- Attribute, Tables, Tuple, Relation Schema, Degree, Cardinality, Column, Relation instance, are some important components of Relational Model
- Relational Integrity constraints are referred to conditions which must be present for a valid Relation approach in DBMS
- Domain constraints can be violated if an attribute value is not appearing in the corresponding domain or it is not of the appropriate data type
- Insert, Select, Modify and Delete are the operations performed in Relational Model constraints
- The relational database is only concerned with data and not with a structure which can improve the performance of the model
- Advantages of Relational model in DBMS are simplicity, structural independence, ease of use, query capability, data independence, scalability, etc.
- Few relational databases have limits on field lengths which can't be exceeded.
- A hierarchical model represents the data in a tree-like structure in which there is a single parent for each record.
- The network database model was created to solve the shortcomings of the hierarchical database model.

2.12 REFERENCES

“Fundamentals of Database Systems”, Elmasri Navrate, 6th edition, 2013, Pearson

“Introduction to Database Systems”, C.J.Date, Pearson Education

“Data base System Concepts”, Silberschatz, Korth, McGraw Hill, V edition

Starting with Oracle, by John Day and Craig Van Slyke

Fundamentals of Database Systems, by Navathe

2.13 FURTHER READING

Database System Concepts, by Henry F. Korth

“Database Systems, The Complete Book”, Hector Garcia-Molina, Jeffrey D. Ullman and Jennifer Widom, 6th impression, 2011, Pearson

“Data base Management Systems”, Raghu Rama Krishnan, Johannes Gehrke, 3rd Edition, 2003, McGraw Hill

2.14 QUESTIONS FOR PRACTICE (SHORT & LONG ANSWERS TYPE)

1. What do you mean by data model? Explain its types.
2. What is the difference between semi-structured data model and object-oriented data model?
3. What is a primary key? Give example.

4. What do you understand by integrity constraints?
5. How can you differentiate primary key from foreign key?
6. List the best practices for creating a Relational Model.
7. Explain the types of integrity constraints.
8. List the advantages of relational model.
9. Describe concept of hierarchical model.
10. What is the difference between relational algebra and relational calculus?
11. Explain the concept of ER Model to Relational Model.

B.Sc.(DATA SCIENCE)
SEMESTER-III
DATABASE MANAGEMENT SYSTEM

UNIT III: CONCEPTUAL DATA MODELING USING E-R DATA MODEL

STRUCTURE

- 3.0 Objectives
- 3.1 Introduction
- 3.2 ER Model
 - 3.2.1 Components of ER Diagram
 - 3.2.1.1 Entity
 - 3.2.1.2 Attribute
 - 3.2.1.3 Relationship
 - 3.2.2 Generalization
 - 3.2.3 Specialization
- 3.3 Specifying Constraints
- 3.4 Conversion of ER diagram into tables
- 3.5 Practice Problems Based on Converting ER diagram to tables
- 3.6 Summary
- 3.7 References
- 3.8 Further Readings
- 3.9 Questions for Practice (Short & Long Answers Type)

3.0 OBJECTIVES

- To provide overview of ER Model.
- To elaborate components of ER Diagram.
- To provide description of entity, attribute, relationship.
- To explain constraints.
- To discuss the concept of Generalization.
- To discuss the concept of Specialization.
- To elaborate conversion of ER diagram into tables.
- To practice problems based on converting ER diagrams into tables.

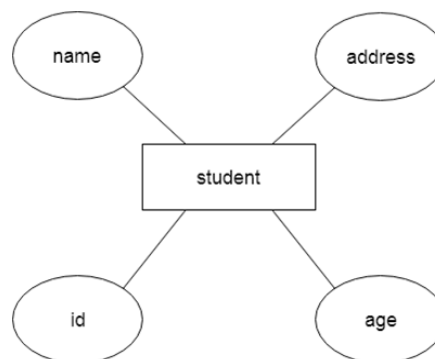
3.1 INTRODUCTION

Entity-Relationship Model describes an overall and conceptual view of the organization of data. This unit describes how the data in a database is represented using E-R model. It discusses components of ER diagram that includes entity, attribute as well as relationship. In this unit, we also discuss the concept of Generalization and Specialization. It also describes various constraints such as Not Null, Unique, domain, mapping etc. This unit also provides the Conversion of ER Models to Tables followed by practical problems based on E-R data model.

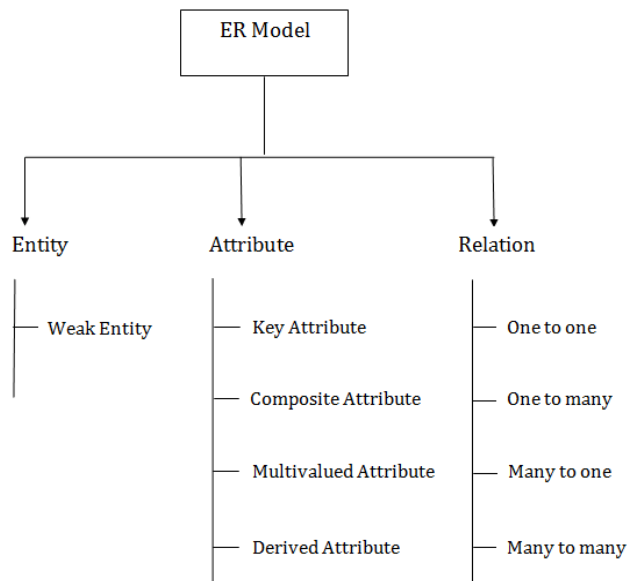
3.2 ER MODEL

- ER model stands for an Entity-Relationship model. It is a high-level data model. This model is used to define the data elements and relationship for a specified system.
- It develops a conceptual design for the database. It also develops a very simple and easy to design view of data.
- In ER modeling, the database structure is portrayed as a diagram called an entity-relationship diagram.

For example: Suppose we design a school database. In this database, the student will be an entity with attributes like address, name, id, age, etc. The address can be another entity with attributes like city, street name, pin code, etc and there will be a relationship between them.



3.2.1 Component of ER Diagram



3.2.1.1 Entity

An entity may be any object, class, person or place. In the ER diagram, an entity can be represented as rectangles.

Consider an organization as an example- manager, product, employee, department etc. can be taken as an entity.



3.2.1.1.1 Weak Entity

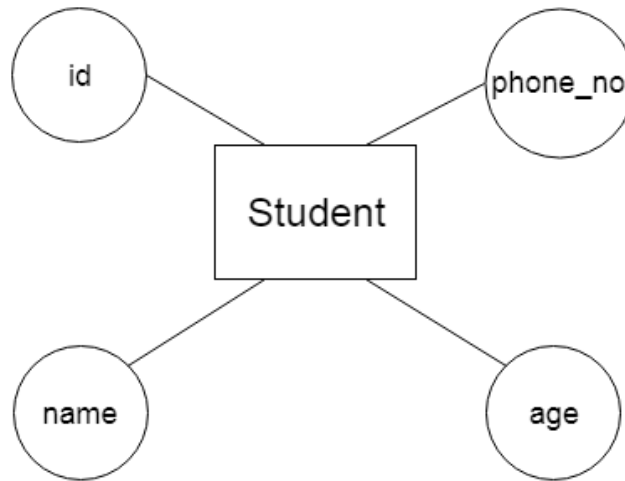
An entity that depends on another entity called a weak entity. The weak entity doesn't contain any key attribute of its own. The weak entity is represented by a double rectangle.



3.2.1.2 Attribute

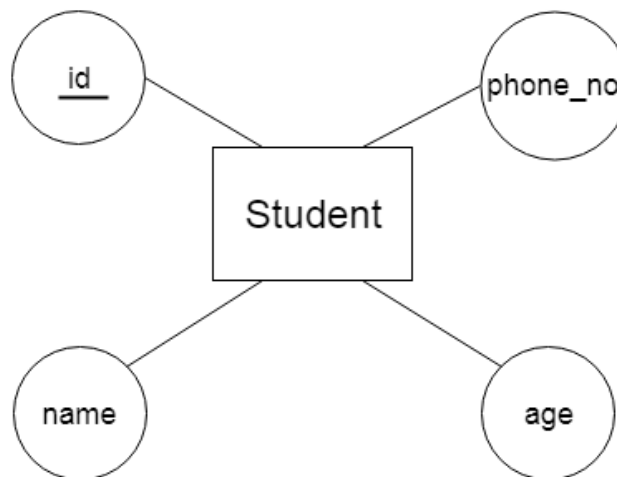
An attribute is used to describe the property of an entity. Eclipse is used to represent an attribute.

For example, id, age, contact number, name, etc. can be attributes of a student.



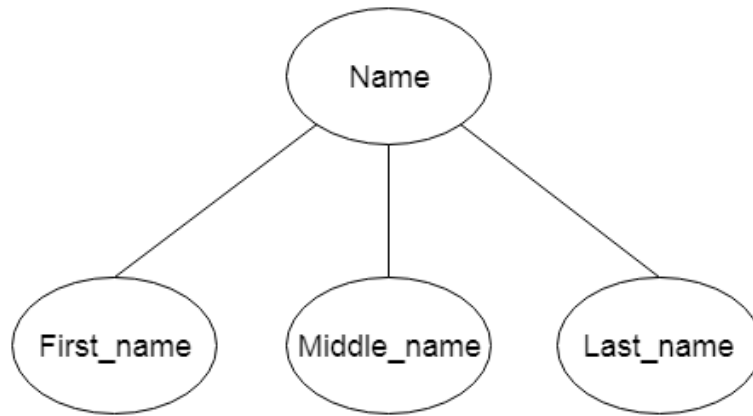
3.2.1.2.1 Key Attribute

The key attribute is used to represent the main characteristics of an entity. It represents a primary key. The key attribute is represented by an ellipse with the text underlined.



3.2.1.2.2 Composite Attribute

An attribute composed of many other attributes is known as a composite attribute. The composite attribute is represented by an ellipse, and those ellipses are connected with an ellipse.



3.2.1.2.3 Multivalued Attribute

An attribute can have more than one value. These attributes are known as a multivalued attribute. The double oval is used to represent a multivalued attribute.

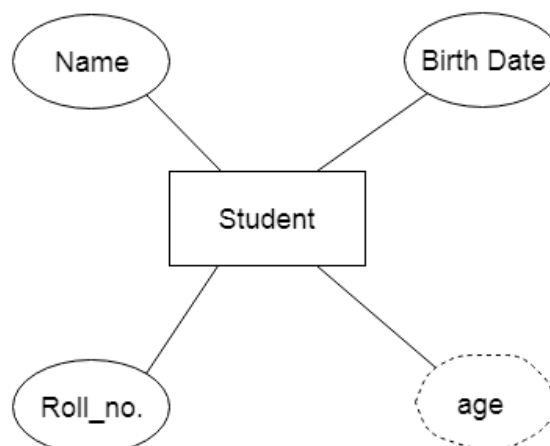
For example, a student can have more than one phone number.



3.2.1.2.4 Derived Attribute

An attribute that can be derived from another attribute is known as a derived attribute. It can be represented by a dashed ellipse.

For example, A person's age changes over time and can be derived from another attribute like Date of birth.



3.2.1.3 Relationship

A relationship is used to describe the relation between entities. Diamond or rhombus is used to represent the relationship.



Types of Relationship are as follows

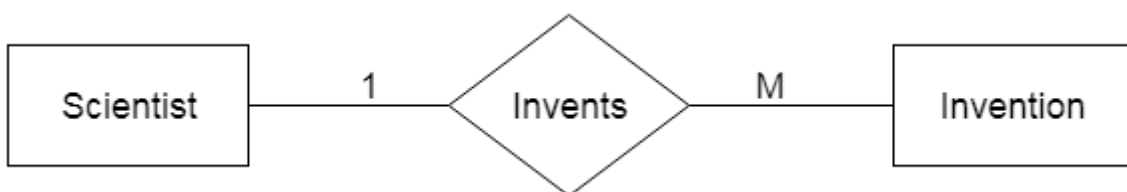
3.2.1.3.1 One-to-One Relationship: When only one instance of an entity is associated with the relationship, then it is known as one to one relationship.

For example, A female can marry to one male, and a male can marry to one female.



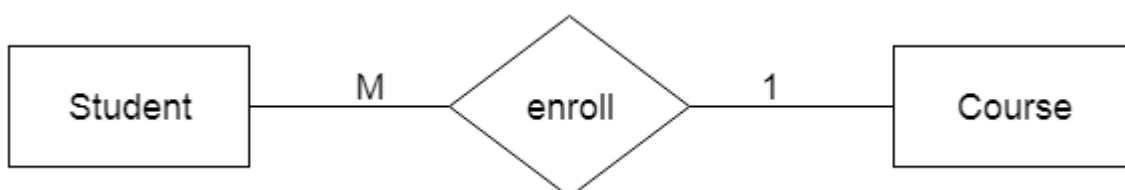
3.2.1.3.2 One-to-many relationship: When only one instance of the entity on the left, and more than one instance of an entity on the right associates with the relationship then this is known as a one-to-many relationship.

For example, scientists can invent many inventions, but the invention is done by the only specific scientist.



3.2.1.3.3 Many-to-one relationship: When more than one instance of the entity on the left, and only one instance of an entity on the right associates with the relationship then it is known as a many-to-one relationship.

For example, Student enrolls for only one course, but a course can have many students.



3.2.1.3.4 Many-to-many relationship: When more than one instance of the entity on the left, and more than one instance of an entity on the right associates with the relationship then it is known as a many-to-many relationship.

For example, employees can assign by many projects and project can have many employees.



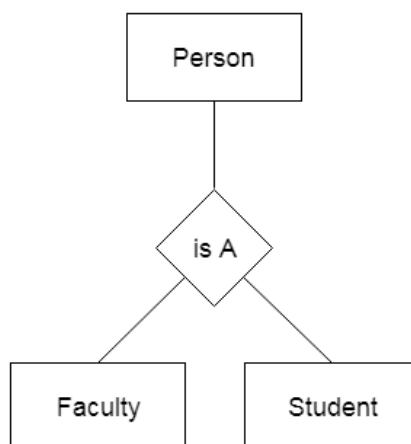
Self-Check Exercise

1. What do you mean by entity?
2. Why do you use key attribute? Give example.
3. Explain various terms of relationship.
4. Describe derived attribute. How can you represent it?

3.2.2 Generalization

- Generalization is like a bottom-up approach in which two or more entities of lower level combine to form a higher level entity if they have some attributes in common.
- In generalization, an entity of a higher level can also combine with the entities of the lower level to form a further higher level entity.
- Generalization is more like subclass and superclass system, but the only difference is the approach. Generalization uses the bottom-up approach.
- In generalization, entities are combined to form a more generalized entity, i.e., subclasses are combined to make a superclass.

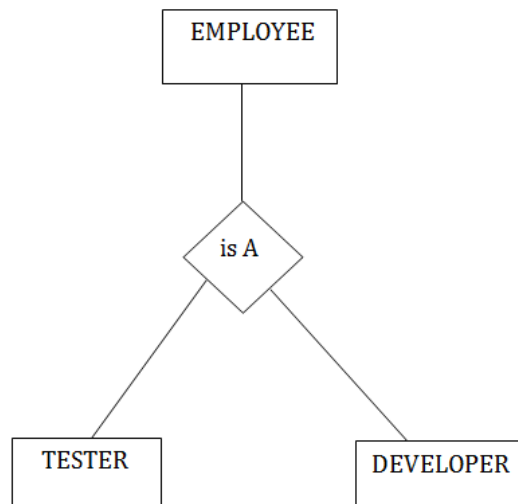
For example, Faculty and Student entities can be generalized and create a higher level entity Person.



3.2.3 Specialization

- Specialization is a top-down approach, and it is opposite to Generalization. In specialization, one higher level entity can be broken down into two lower level entities.
- Specialization is used to identify the subset of an entity set that shares some distinguishing characteristics.
- Normally, the superclass is defined first, the subclass and its related attributes are defined next, and relationship set are then added.

For example: In an Employee management system, EMPLOYEE entity can be specialized as TESTER or DEVELOPER based on what role they play in the company.



3.3 SPECIFYING CONSTRAINTS

Constraints enforce limits to the data or type of data that can be inserted/updated/deleted from a table. The whole purpose of constraints is to maintain the **data integrity** during an update/delete/insert into a table. Following are the types of various constraints:

- NOT NULL Constraints
- UNIQUE Constraints
- DEFAULT Constraints
- CHECK Constraints
- Key Constraints – PRIMARY KEY, FOREIGN KEY
- Domain Constraint
- Mapping Constraint

3.3.1 NOT NULL Constraint

NOT NULL constraint makes sure that a column does not hold NULL value. When we don't provide value for a particular column while inserting a record into a table, it takes NULL value by default. By specifying NULL constraint, we can be sure that a particular column(s) cannot have NULL values.

Example:

```
CREATE TABLE STUDENT(  
ROLL_NO INT NOT NULL,  
STU_NAME VARCHAR (35) NOT NULL,  
STU_AGE INT NOT NULL,  
STU_ADDRESS VARCHAR (235),  
PRIMARY KEY (ROLL_NO)  
);
```

3.3.2 UNIQUE Constraint

UNIQUE Constraint enforces a column or set of columns to have unique values. If a column has a unique constraint, it means that particular column cannot have duplicate values in a table.

Example:

```
CREATE TABLE STUDENT(  
ROLL_NO INT NOT NULL,  
STU_NAME VARCHAR (35) NOT NULL UNIQUE,  
STU_AGE INT NOT NULL,  
STU_ADDRESS VARCHAR (35) UNIQUE,  
PRIMARY KEY (ROLL_NO)  
);
```

3.3.3 DEFAULT Constraint

The DEFAULT constraint provides a default value to a column when there is no value provided while inserting a record into a table.

Example:

```
CREATE TABLE STUDENT(  
ROLL_NO INT NOT NULL,  
STU_NAME VARCHAR (35) NOT NULL,  
STU_AGE INT NOT NULL,  
EXAM_FEE INT DEFAULT 10000,  
STU_ADDRESS VARCHAR (35),  
PRIMARY KEY (ROLL_NO)  
);
```

3.3.4 CHECK Constraint

This constraint is used for specifying range of values for a particular column of a table. When this constraint is being set on a column, it ensures that the specified column must have the value falling in the specified range.

Example:

```
CREATE TABLE STUDENT(  
ROLL_NO INT NOT NULL CHECK(ROLL_NO >1000) ,  
STU_NAME VARCHAR (35) NOT NULL,  
STU_AGE INT NOT NULL,  
EXAM_FEE INT DEFAULT 10000,  
STU_ADDRESS VARCHAR (35) ,  
PRIMARY KEY (ROLL_NO)  
);
```

In the above example we have set the check constraint on ROLL_NO column of STUDENT table. Now, the ROLL_NO field must have the value greater than 1000.

3.3.5 Key Constraints

It contains two further Key constraints: Primary and Foreign Key.

3.3.5.1 PRIMARY KEY:

Primary key uniquely identifies each record in a table. It must have unique values and cannot contain nulls. In the below example the ROLL_NO field is marked as primary key, that means the ROLL_NO field cannot have duplicate and null values.

Example:

```
CREATE TABLE STUDENT(  
ROLL_NO INT NOT NULL,  
STU_NAME VARCHAR (35) NOT NULL UNIQUE,  
STU_AGE INT NOT NULL,  
STU_ADDRESS VARCHAR (35) UNIQUE,  
PRIMARY KEY (ROLL_NO)  
);
```

3.3.5.2 FOREIGN KEY:

Foreign keys are the columns of a table that points to the primary key of another table. They act as a cross-reference between tables.

For example:

In the below example, the Stu_Id column in Course_enrollment table is a foreign key as it points to the primary key of the Student table.

Course_enrollment table:

Course_Id	Stu_Id
C01	101
C02	102
C03	101
C05	102
C06	103
C07	102

Student table:

Stu_Id	Stu_Name	Stu_Age
101	Chaitanya	22
102	Arya	26
103	Bran	25
104	Jon	21

3.3.6 Domain Constraint

Each table has certain set of columns and each column allows a same type of data, based on its data type. The column does not accept values of any other data type.

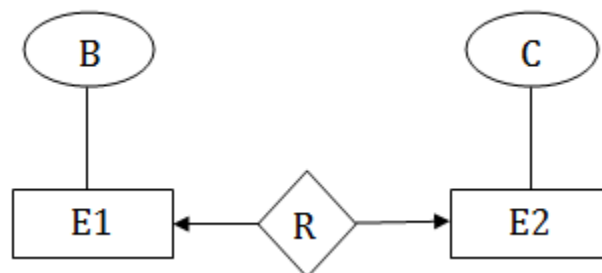
Domain Constraint = data type + Constraints (NOT NULL / UNIQUE / PRIMARY KEY / FOREIGN KEY / CHECK / DEFAULT)

3.3.7 Mapping Constraint

- A mapping constraint is a data constraint that expresses the number of entities to which another entity can be related via a relationship set.
- It is most useful in describing the relationship sets that involve more than two entity sets.
- For binary relationship set R on an entity set A and B, there are four possible mapping cardinalities. These are as follows:
 1. One to one (1:1)
 2. One to many (1:M)
 3. Many to one (M:1)
 4. Many to many (M:M)

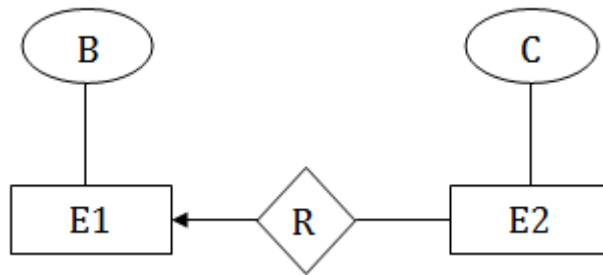
3.3.7.1 One-to-one

In one-to-one mapping, an entity in E1 is associated with at most one entity in E2, and an entity in E2 is associated with at most one entity in E1.



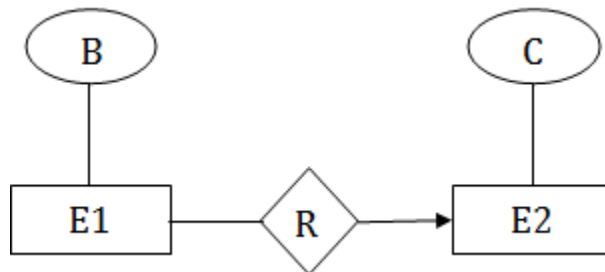
3.3.7.2 One-to-many

In one-to-many mapping, an entity in E1 is associated with any number of entities in E2, and an entity in E2 is associated with at most one entity in E1.



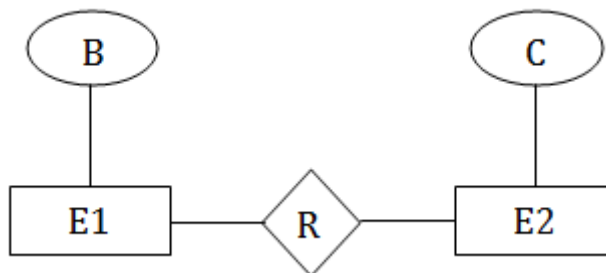
3.3.7.3 Many-to-one

In one-to-many mapping, an entity in E1 is associated with at most one entity in E2, and an entity in E2 is associated with any number of entities in E1.



3.3.7.4 Many-to-many

In many-to-many mapping, an entity in E1 is associated with any number of entities in E2, and an entity in E2 is associated with any number of entities in E1.



You can have these constraints in place while creating tables in database.

Example:

```
CREATE TABLE Customer (
customer_id int PRIMARY KEY NOT NULL,
```

```

first_name varchar(20),
last_name varchar(20)
);

CREATE TABLE Order (
order_id int PRIMARY KEY NOT NULL,
customer_id int,
order_details varchar(50),
constraint fk_Customers foreign key (customer_id)
references dbo.Customer
);

```

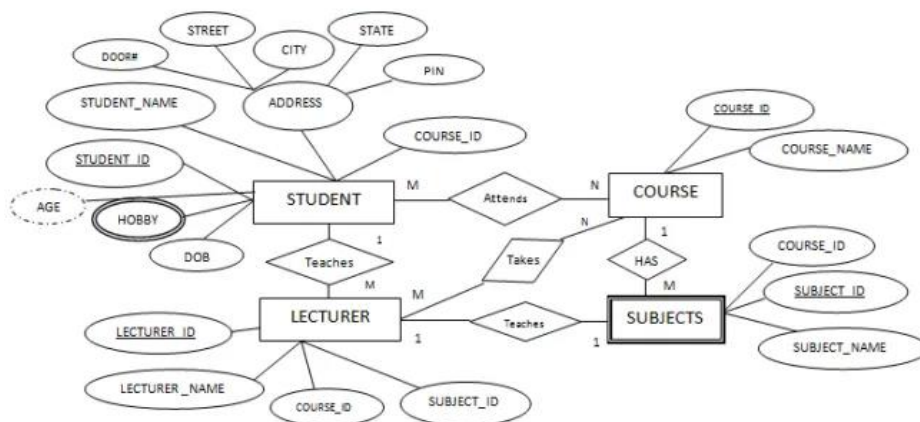
Assuming, that a customer orders more than once, the above relation represents **one to many** relation. Similarly you can achieve other mapping constraints based on the requirements.

Self-Check Exercise

1. What do you understand by generalization? Explain.
2. What is meant by specialization? Explain.
3. How can you differentiate domain constraint from Not Null constraint?
4. How can you describe unique constraint? Give example.
5. What do you mean by mapping constraint? Explain with example.

3.4 CONVERSION OF ER DIAGRAM INTO TABLES

There are various steps involved in converting it into tables and columns. Each type of entity, attribute and relationship in the diagram takes their own depiction. Consider the below ER diagram and it will be converted into tables, columns and mappings.



The basic rule for converting the ER diagrams into tables is as follows:

- Convert all the Entities in the diagram to tables.

- All the entities represented in the rectangular box in the ER diagram become independent tables in the database. In the below diagram, STUDENT, COURSE, LECTURER and SUBJECTS forms individual tables.
- All single valued attributes of an entity is converted to a column of the table

All the attributes, whose value at any instance of time is unique, are considered as columns of that table. In the STUDENT Entity, STUDENT_ID, STUDENT_NAME form the columns of STUDENT table. Similarly, LECTURER_ID, LECTURER_NAME form the columns of LECTURER table. And so on.

- Key attribute in the ER diagram becomes the Primary key of the table.

In diagram above, STUDENT_ID, LECTURER_ID, COURSE_ID and SUB_ID are the key attributes of the entities. Hence we consider them as the primary keys of respective table.

- Declare the foreign key column, if applicable.

In the diagram, attribute COURSE_ID in the STUDENT entity is from COURSE entity. Hence add COURSE_ID in the STUDENT table and assign it foreign key constraint. COURSE_ID and SUBJECT_ID in LECTURER table forms the foreign key column. Hence by declaring the foreign key constraints, mapping between the tables are established.

- Any multi-valued attributes are converted into new table.

A hobby in the Student table is a multivalued attribute. Any student can have any number of hobbies. So we cannot represent multiple values in a single column of STUDENT table. We need to store it separately, so that we can store any number of hobbies, adding/ removing / deleting hobbies should not create any redundancy or anomalies in the system. Hence we create a separate table STUD_HOBBY with STUDENT_ID and HOBBY as its columns. We create a composite key using both the columns.

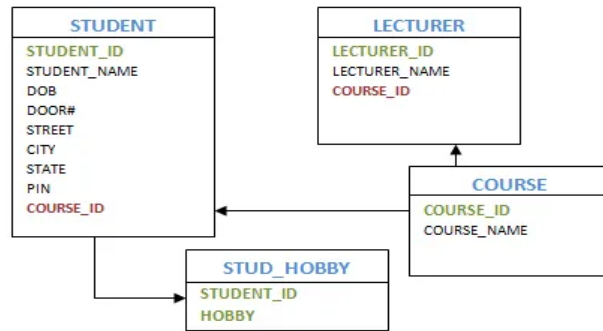
- Any composite attributes are merged into same table as different columns.

In the diagram above, Student Address is a composite attribute. It has Door#, Street, City, State and Pin. These attributes are merged into the STUDENT table as individual columns.

- One can ignore the derived attribute, since it can be calculated at any time.

In the STUDENT table, Age can be derived at any point of time by calculating the difference between DateOfBirth and current date. Hence we need not create a column for this attribute. It reduces the duplicity in the database.

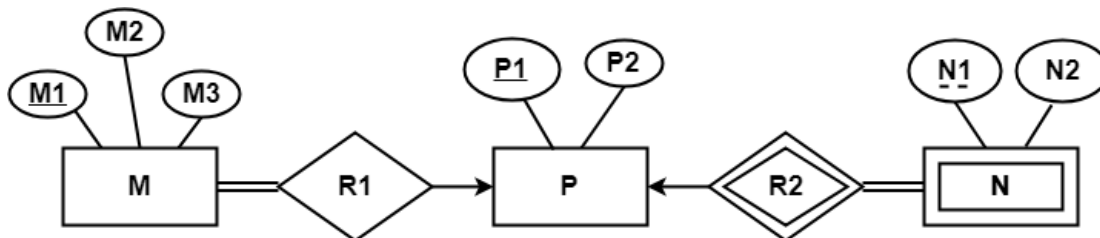
These are the very basic rules of converting ER diagram into tables and columns, and assigning the mapping between the tables. Table structure at this would be as below:



3.5 PRACTICE PROBLEMS

Problem-01:

Find the minimum number of tables required for the following ER diagram in relational model-



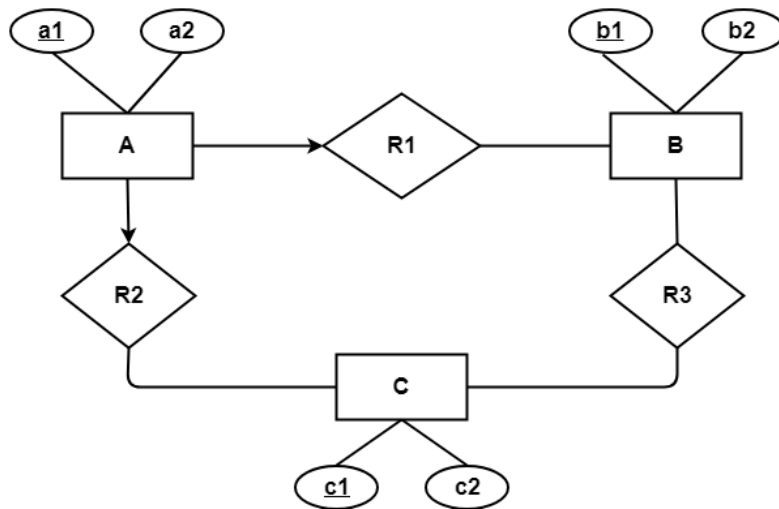
Solution-

Applying the rules, minimum 3 tables will be required-

- MR1 (M1 , M2 , M3 , P1)
- P (P1 , P2)
- NR2 (P1 , N1 , N2)

Problem-02:

Find the minimum number of tables required to represent the given ER diagram in relational model-



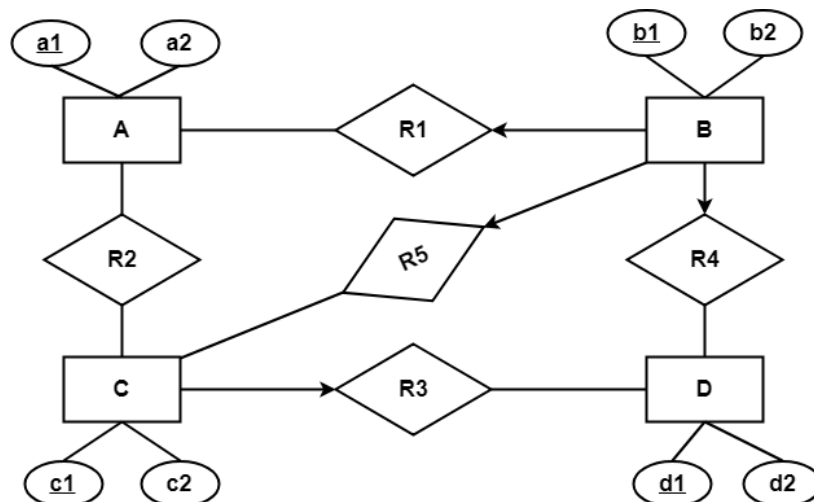
Solution-

Applying the rules, minimum 4 tables will be required-

- AR1R2 (a1 , a2 , b1 , c1)
- B (b1 , b2)
- C (c1 , c2)
- R3 (b1 , c1)

Problem-03:

Find the minimum number of tables required to represent the given ER diagram in relational model-



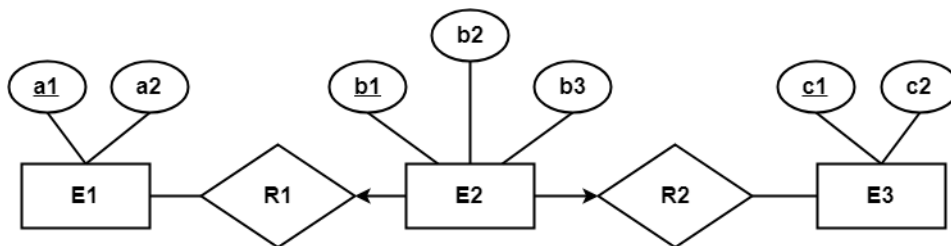
Solution-

Applying the rules, minimum 5 tables will be required-

- BR1R4R5 (b1 , b2 , a1 , c1 , d1)
- A (a1 , a2)
- R2 (a1 , c1)
- CR3 (c1 , c2 , d1)
- D (d1 , d2)

Problem-04:

Find the minimum number of tables required to represent the given ER diagram in relational model-



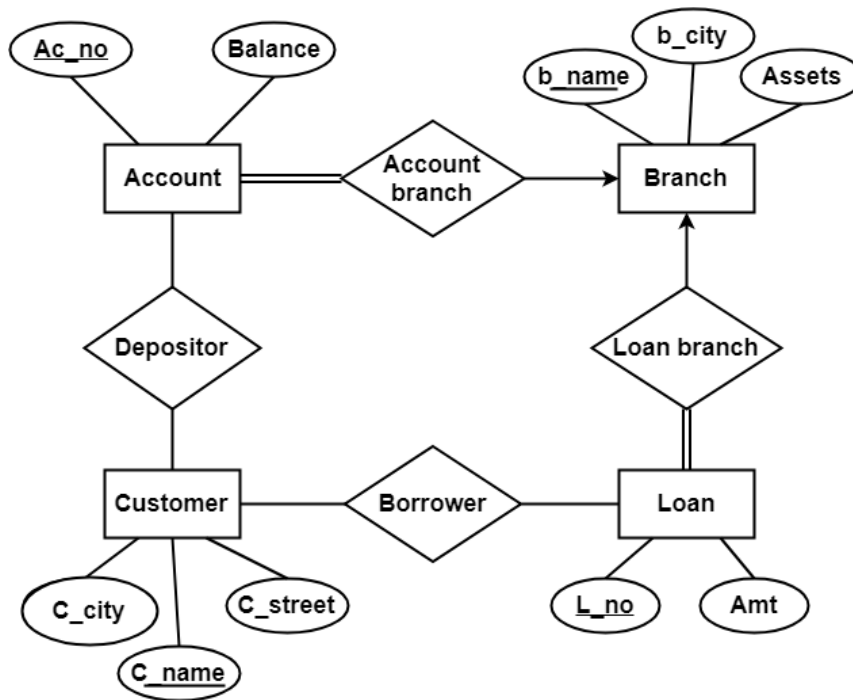
Solution-

Applying the rules, minimum 3 tables will be required-

- E1 (a1 , a2)
- E2R1R2 (b1 , b2 , a1 , c1 , b3)
- E3 (c1 , c2)

Problem-05:

Find the minimum number of tables required to represent the given ER diagram in relational model-



Solution-

Applying the rules that we have learnt, minimum 6 tables will be required-

- Account (Ac_no , Balance , b_name)
- Branch (b_name , b_city , Assets)
- Loan (L_no , Amt , b_name)
- Borrower (C_name , L_no)
- Customer (C_name , C_street , C_city)
- Depositor (C_name , Ac_no)

3.6 SUMMARY

- ER Model is used to define the data elements and relationship for a specified system.
- An entity may be any object, class, person or place. In the ER diagram, an entity can be represented as rectangles.
- An attribute is used to describe the property of an entity. Eclipse is used to represent an attribute.
- The key attribute is used to represent the main characteristics of an entity. It represents a primary key.
- An attribute composed of many other attributes is known as a composite attribute.
- An attribute can have more than one value. These attributes are known as a multivalued attribute. The double oval is used to represent a multivalued attribute.

- An attribute that can be derived from another attribute is known as a derived attribute. It can be represented by a dashed ellipse.
- Generalization is like a bottom-up approach in which two or more entities of lower level combine to form a higher level entity if they have some attributes in common.
- Specialization is a top-down approach, and it is opposite to Generalization. In specialization, one higher level entity can be broken down into two lower level entities.
- Constraints enforce limits to the data or type of data that can be inserted/updated/deleted from a table.
- A mapping constraint is a data constraint that expresses the number of entities to which another entity can be related via a relationship set.

3.7 REFERENCES

“Fundamentals of Database Systems”, Elmasri Navrate, 6th edition, 2013, Pearson
 “Introduction to Database Systems”, C.J.Date, Pearson Education
 “Data base System Concepts”, Silberschatz, Korth, McGraw Hill, V edition
 Fundamentals of Database Systems, by Navathe
<https://beginnersbook.com/2015/04/constraints-in-dbms/>
<https://www.gatevidyalay.com/er-diagrams-to-tables-practice-problems/>

3.8 FURTHER READING

Database System Concepts - 6th edition - Avi Silberschatz
 Database Systems - A Practical Approach to Design, Implementation & Management
 By Thomas Connolly, Carolyn Begg
 Database Management Systems By Raghu Ramkrishnan, Gehrke

3.9 QUESTIONS FOR PRACTICE (SHORT & LONG ANSWERS TYPE)

1. What is ER Model?
2. Name the various types of attributes?
3. What do you mean by relationship?
4. What is the difference between Generalization and Specialization?
5. What do you understand by domain constraint?
6. Explain possible mapping cardinalities along with diagram.
7. Write the basic rules for converting the ER diagrams into tables.
8. How can you differentiate Composite attribute from Multivalued attribute? Explain with example.
9. What is default constraint? Give example.
10. What do you understand by weak entity? How can you represent it?

B.Sc.(DATA SCIENCE)
SEMESTER-III
DATABASE MANAGEMENT SYSTEM

UNIT IV: NORMAL FORMS

STRUCTURE

- 4.0 Objectives
- 4.1 Introduction
- 4.2 Functional Dependency
 - 4.2.1 Types of Functional Dependency
- 4.3 Inference Rule
 - 4.3.1 Types of Inference Rule
- 4.4 Normal Forms/Normalization
 - 4.4.1 Need of Normalization
 - 4.4.2 Types of Normal Forms
- 4.5 Multivalued Dependency
- 4.6 Join Dependency
- 4.7 Summary
- 4.8 References
- 4.9 Further Readings
- 4.10 Practice Exercises

4.0 OBJECTIVES

- To describe the process of normalization.
- To provide description of terms used in normalization.
- To describe first normal form.
- To explain partial dependency.
- To describe second normal form.
- To define transitive dependency.
- To describe third normal form.
- To explain fourth & fifth normal form.
- To describe data anomalies in 1NF, 2NF & 3NF.
- To provide description of Multivalued dependency.
- To explain join dependency.

4.1 INTRODUCTION

Normalization or normal form refers to the structure of a database which was developed by IBM researcher E.F. Codd in the 1970s. Normalization in database management system is used to organize data in the form of a related tables. In this unit, we will discuss the concept of normalization and the terminology frequently used in normalization. We will discuss in detail the first, second, third, fourth and fifth normal forms used to normalise the relational tables so that redundant information is removed from a database. In addition, we will discuss the concepts such as partial dependency and transitive dependency on which second and third normal forms are based.

4.2 FUNCTIONAL DEPENDENCY

Functional dependency (FD) is a set of constraints between two attributes in a relation. It typically exists between the primary key and non-key attribute within a table.

$X \rightarrow Y$

The left side of FD is known as a determinant, the right side of the production is known as a dependent.

For example:

Assume, we have an employee table with following attributes:

Emp_Id, Emp_Name, Emp_Address.

Here,

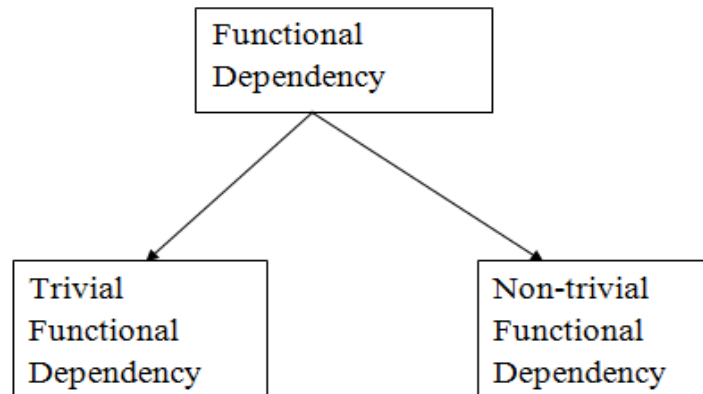
Emp_Id attribute can uniquely identify the Emp_Name attribute of employee table because if we know the Emp_Id, we can tell that employee name associated with it.

So, Functional dependency can be written as:

$Emp_Id \rightarrow Emp_Name$

We can say that Emp_Name is functionally dependent on Emp_Id. Thus, Functional dependency says that if two tuples have same values for attributes X1, X2,..., Xn, then those two tuples must have to have same values for attributes Y1, Y2, ..., Yn.

4.2.1 Types of Functional dependency



4.2.1.1 Trivial functional dependency

- If a functional dependency (FD) $X \rightarrow Y$ holds, where Y is a subset of X, then it is called a trivial FD. Thus, Trivial FDs always hold.
- The following dependencies are also trivial like:

$X \rightarrow X$ $Y \rightarrow Y$

For Example:

Consider a table with two columns Employee_Id and Employee_Name.

$\{Employee_id, Employee_Name\} \rightarrow Employee_Id$

It's a trivial functional dependency as Employee_Id is a subset of {Employee_Id, Employee_Name}.

Also, The following dependencies are trivial like

$Employee_Id \rightarrow Employee_Id$ $Employee_Name \rightarrow Employee_Name$

4.2.1.2 Non-trivial functional dependency

- $X \rightarrow Y$ has a non-trivial functional dependency if Y is not a subset of X.
- Moreover, When $X \cap Y$ is NULL, then $X \rightarrow Y$ is called as complete non-trivial functional dependency.

For Example:

ID \rightarrow Name
Name \rightarrow DOB

Self-check Exercise

1. Define Functional Dependency.
2. What is the difference between Trival and Non-Trival functional dependency?
3. Give example of functional dependency?

4.3 INFERENCE RULE (IR)

- The Armstrong's axioms are the basic inference rule to conclude functional dependencies on a relational database.
- The inference rule is a type of assertion which can be used to derive additional functional dependency from the initial set.

4.3.1 Types of Inference Rule

The Functional dependency has 6 types of inference rule:

4.3.1.1 Reflexive Rule (IR₁)

- In the reflexive rule, if Y is a subset of X, then X determines Y as shown below.

If $X \supseteq Y$ then $X \rightarrow Y$

For Example:

$X = \{a, b, c, d, e\}$
 $Y = \{a, b, c\}$

4.3.1.2 Augmentation Rule (IR₂)

- The augmentation is also called as a partial dependency. In augmentation, if X determines Y, then XZ determines YZ for any Z i.e. adding attributes in dependencies, does not change the basic dependencies.

i.e. If $X \rightarrow Y$ then $XZ \rightarrow YZ$

For Example:

For R(XYZA), if $X \rightarrow Y$ then $XZ \rightarrow YZ$

4.3.1.3 Transitive Rule (IR₃)

It is same as transitive rule in algebra. In the transitive rule, if X determines Y and Y determines Z, then X must also determine Z.

i.e. If $X \rightarrow Y$ and $Y \rightarrow Z$ then $X \rightarrow Z$

4.3.1.4 Union Rule (IR₄)

Union rule says, if X determines Y and X determines Z, then X must also determine Y and Z.

i.e. If $X \rightarrow Y$ and $X \rightarrow Z$ then $X \rightarrow YZ$

Proof:

$X \rightarrow Y$ (given).....eq.(1)
 $X \rightarrow Z$ (given).....eq.(2)
 $X \rightarrow XY$ (using IR₂ on eq.(1) by augmentation with X, where $XX = X$)
 $XY \rightarrow YZ$ (using IR₂ on eq.(2) by augmentation with Y)
 $X \rightarrow YZ$ (using IR₃ on eq.(3) and eq.(4))

4.3.1.5 Decomposition Rule (IR₅)

- Decomposition rule is also known as project rule. It is the reverse of union rule.
- According to this rule, if X determines Y and Z, then X determines Y and X determines Z separately.

i.e. If $X \rightarrow YZ$ then $X \rightarrow Y$ and $X \rightarrow Z$

Proof:

$X \rightarrow YZ$ (given).....eq.(1)
 $YZ \rightarrow Y$ (using IR₁ Rule)..... eq.(2)
 $X \rightarrow Y$ (using IR₃ on eq.(1) and eq.(2))

4.3.1.6 Pseudo transitive Rule (IR₆)

In Pseudo transitive Rule, if X determines Y and YZ determines W, then XZ determines W.

i.e. If $X \rightarrow Y$ and $YZ \rightarrow W$ then $XZ \rightarrow W$

Proof:

$X \rightarrow Y$ (given).....eq.(1)
 $WY \rightarrow Z$ (given).....eq.(2)
 $WX \rightarrow WY$ (using IR_2 on 1 by augmenting with W)...eq.(3)
 $WX \rightarrow Z$ (using IR_3 on eq.(3) and eq.(2))

Self-Check Exercise

1. What do you mean by inference rule? Explain its types.
2. What is reflexivity rule?
3. How can you define Union rule?
4. Define the term transitive dependency.
5. What do you mean by decomposition rule?

4.4 NORMAL FORMS (NORMALIZATION)

Database normalization is a database schema design technique, by which an existing schema is modified to minimize redundancy and dependency of data as it splits a large table into smaller tables and defines relationships between them to increase the clarity in organizing data.

4.4.1 Need of Normalization

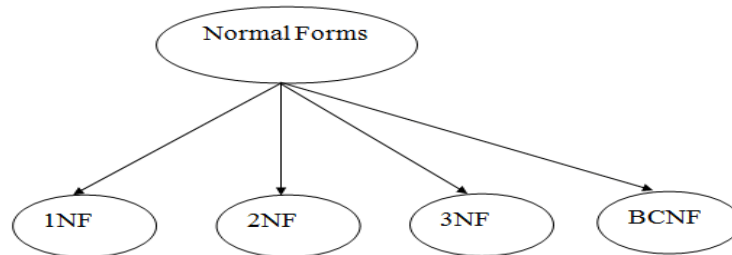
If a database design is not perfect, it may contain anomalies which are problems that can occur in poorly planned, un-normalised databases where all the data is stored in one table. It is also considered as a bad dream for any database administrator and managing a database with following anomalies is next to impossible. So, Such anomalies in database are of three types that can be defined as following:

- **Update anomalies** – If data items are scattered and are not linked to each other properly, then it could lead to strange situations. For example, when we try to update one data item having its copies scattered over several places, a few instances get updated properly while a few others are left with old values. Such instances leave the database in an inconsistent state and considered as update anomalies.
- **Deletion anomalies** – When we try to delete a record, but parts of it remains undeleted because of unawareness, the data is also saved somewhere else, so such state gives result in deletion anomalies.
- **Insert anomalies** – When we try to insert data in a record that does not exist at all then such instance gives result in insert anomalies.

Thus, Normalization minimizes the redundancy from a relation or set of relations and also eliminates the undesirable characteristics like Insertion, Update and Deletion Anomalies.

4.4.2 Types of Normal Forms

There are the four types of normal forms:



Normal Form	Description
<u>1NF</u>	A relation is in 1NF if it contains an atomic value.
<u>2NF</u>	A relation will be in 2NF if it is in 1NF and all non-key attributes are fully functional dependent on the primary key.
<u>3NF</u>	A relation will be in 3NF if it is in 2NF and no transition dependency exists.
<u>4NF</u>	A relation will be in 4NF if it is in Boyce Codd normal form and has no multi-valued dependency.
<u>5NF</u>	A relation is in 5NF if it is in 4NF and does not contain any join dependency and joining should be lossless.

4.4.2.1 First Normal Form

- A relation will be 1NF if it contains an atomic value.
- It states that an attribute of a table cannot hold multiple values. It must hold only single-valued attribute.
- It disallows the multi-valued attribute, composite attribute, and their combinations.

Example: Relation EMPLOYEE is not in 1NF because of multi-valued attribute EMP_PHONE.

EMPLOYEE table:

EMP_ID	EMP_NAME	EMP_PHONE	EMP_STATE
14	John	7272826385, 9064738238	UP
20	Harry	8574783832	Bihar
12	Sam	7390372389, 8589830302	Punjab

The decomposition of the EMPLOYEE table into 1NF has been shown below:

EMP_ID	EMP_NAME	EMP_PHONE	EMP_STATE
14	John	7272826385	UP
14	John	9064738238	UP
20	Harry	8574783832	Bihar
12	Sam	7390372389	Punjab
12	Sam	8589830302	Punjab

Thus, It is clear that each attribute must contain only single value from its pre-defined domain.

4.4.2.2 Second Normal Form

- Before we learn about the second normal form, we need to understand the following –
 - **Prime attribute** – An attribute, which is a part of the candidate-key, is known as a prime attribute.
 - **Non-prime attribute** – An attribute, which is not a part of the prime-key, is said to be a non-prime attribute.
- If we follow second normal form, then relational must be in 1NF.
- Every non-prime attribute should be fully functionally dependent on prime key attribute i.e. if $X \rightarrow A$ holds, then there should not be any proper subset Y of X , for which $Y \rightarrow A$ also holds true.

Example: Let's assume, a school can store the data of teachers and the subjects they teach. In a school, a teacher can teach more than one subject.

TEACHER Table:

TEACHER_ID	SUBJECT	TEACHER_AGE
25	Chemistry	30
25	Biology	30
47	English	35
83	Math	38
83	Computer	38

In the given table, non-prime attribute TEACHER_AGE is dependent on TEACHER_ID which is a proper subset of a candidate key. That's why it violates the rule for 2NF.

To convert the given table into 2NF, we decompose it into two tables:

TEACHER_DETAIL Table:

TEACHER_ID	TEACHER_AGE
25	30
47	35
83	38

TEACHER_SUBJECT Table:

TEACHER_ID	SUBJECT
25	Chemistry
25	Biology
47	English
83	Math
83	Computer

- Thus, there exists no partial dependency in 2NF.

4.4.2.3 Third Normal Form

- A relation will be in 3NF if it is in 2NF and does not contain any transitive partial dependency.
- 3NF is used to reduce the data duplication. It is also used to achieve the data integrity.
- If there is no transitive dependency for non-prime attributes, then the relation must be in third normal form.
- A relation is in third normal form if it holds atleast one of the following conditions for every non-trivial function dependency
 $X \rightarrow Y$.

Where,

X is a super key.

Y is a prime attribute, i.e., each element of Y is part of some candidate key.

Example:

EMPLOYEE_DETAIL table:

EMP_ID	EMP_NAME	EMP_ZIP	EMP_STATE	EMP_CITY
222	Harry	201010	UP	Noida
333	Stephan	02228	US	Boston
444	Lan	60007	US	Chicago
555	Katharine	06389	UK	Norwich
666	John	462007	MP	Bhopal

Super key in the above table:

{EMP_ID}, {EMP_ID, EMP_NAME}, {EMP_ID, EMP_NAME, EMP_ZIP}.....so on

Candidate key:

{EMP_ID}

Non-prime attributes: In the given table, all attributes except EMP_ID are non-prime.

Here,

EMP_STATE & EMP_CITY dependent on EMP_ZIP and EMP_ZIP dependent on EMP_ID. The non-prime attributes (EMP_STATE, EMP_CITY) transitively dependent on super key(EMP_ID). It violates the rule of third normal form.

That's why we need to move the EMP_CITY and EMP_STATE to the new <EMPLOYEE_ZIP> table, with EMP_ZIP as a Primary key.

EMPLOYEE table:

EMP_ID	EMP_NAME	EMP_ZIP
222	Harry	201010
333	Stephan	02228
444	Lan	60007
555	Katharine	06389
666	John	462007

EMPLOYEE_ZIP table:

EMP_ZIP	EMP_STATE	EMP_CITY
201010	UP	Noida
02228	US	Boston
60007	US	Chicago
06389	UK	Norwich
462007	MP	Bhopal

4.4.2.4 Boyce-Codd Normal Form

- BCNF is an extension of 3NF on strict terms.
- A table is in BCNF if every functional dependency $X \rightarrow Y$, X is the super key of the table.
- For BCNF, the table should be in 3NF, and for every FD, LHS is super key.

Example: Let's assume there is a company where employees work in more than one department.

EMPLOYEE table:

EMP_ID	EMP_COUNTRY	EMP_DEPT	DEPT_TYPE	EMP_DEPT_NO
264	India	Designing	D394	283
264	India	Testing	D394	300
364	UK	Stores	D283	232
364	UK	Developing	D283	549

In the above table Functional dependencies are as follows:

EMP_ID → EMP_COUNTRY
EMP_DEPT → {DEPT_TYPE, EMP_DEPT_NO}

Candidate key: {EMP-ID, EMP-DEPT}

- The table is not in BCNF because neither EMP_DEPT nor EMP_ID alone are keys.
- To convert the given table into BCNF, we decompose it into three tables:

EMP_COUNTRY table:

EMP_ID	EMP_COUNTRY
264	India
264	India

EMP_DEPT table:

EMP_DEPT	DEPT_TYPE	EMP_DEPT_NO
Designing	D394	283
Testing	D394	300
Stores	D283	232
Developing	D283	549

EMP_DEPT_MAPPING table:

EMP_ID	EMP_DEPT
D394	283
D394	300
D283	232
D283	549

Functional dependencies:

EMP_ID → EMP_COUNTRY
EMP_DEPT → {DEPT_TYPE, EMP_DEPT_NO}

Candidate keys:

For the first table: EMP_ID

For the second table: EMP_DEPT

For the third table: {EMP_ID, EMP_DEPT}

4.4.2.5 Fourth Normal Form (4NF)

- A relation will be in 4NF if it is in Boyce Codd normal form and has no multi-valued dependency.
- For a dependency $X \twoheadrightarrow Y$, if for a single value of X, multiple values of Y exist, then the relation will be a multi-valued dependency.

Example:

STUDENT

STU_ID	COURSE	HOBBY
21	Computer	Dancing
21	Math	Singing
34	Chemistry	Dancing
74	Biology	Cricket

59	Physics	Hockey
----	---------	--------

The given STUDENT table is in 3NF, but the COURSE and HOBBY are two independent entity. Hence, there is no relationship between COURSE and HOBBY.

In the STUDENT relation, a student with STU_ID, 21 contains two courses, Computer and Math and two hobbies, Dancing and Singing. So there is a Multi-valued dependency on STU_ID, which leads to unnecessary repetition of data.

So to make the above table into 4NF, we can decompose it into two tables:

STUDENT_COURSE

STU_ID	COURSE
21	Computer
21	Math
34	Chemistry
74	Biology
59	Physics

STUDENT_HOBBY

STU_ID	HOBBY
21	Dancing
21	Singing
34	Dancing
74	Cricket
59	Hockey

4.4.2.6 Fifth Normal Form (5NF)

- A relation is in 5NF if it is in 4NF and does not contain any join dependency and joining should be lossless.

- 5NF is satisfied when all the tables are broken into as many tables as possible in order to avoid redundancy.
- 5NF is also known as Project-join normal form (PJ/NF).

Example:

SUBJECT	LECTURER	SEMESTER
Computer	Anshika	Semester 1
Computer	John	Semester 1
Math	John	Semester 1
Math	Akash	Semester 2
Chemistry	Praveen	Semester 1

In the above table, John takes both Computer and Math classes for Semester 1 but he doesn't take Math class for Semester 2. In this case, combination of all these fields required to identify a valid data.

Suppose we add a new Semester as Semester 3 but do not know about the subject and who will be taking that subject so we leave Lecturer and Subject as NULL. But all three columns together act as a primary key, so we can't leave other two columns blank.

So to make the above table into 5NF, we can decompose it into three relations P1, P2 & P3:

P1

SEMESTER	SUBJECT
Semester 1	Computer
Semester 1	Math
Semester 1	Chemistry
Semester 2	Math

P2

SUBJECT	LECTURER
Computer	Anshika
Computer	John
Math	John
Math	Akash
Chemistry	Praveen

P3

SEMSTER	LECTURER
Semester 1	Anshika
Semester 1	John
Semester 1	John
Semester 2	Akash
Semester 1	Praveen

Self-Check Exercise

1. What do you understand by the term 'Normalization'?
2. List the various normal forms involved in the normalization process.
3. How do you apply 2NF to a relational table?
4. How is data redundancy prevented using 1NF?
5. How do you apply 3NF to a relational table?

4.5 MULTIVALUED DEPENDENCY

- Multivalued dependency occurs when two attributes in a table are independent of each other but, both depend on a third attribute.
- A multivalued dependency consists of at least two attributes that are dependent on a third attribute that's why it always requires at least three attributes.

Example: Suppose there is a bike manufacturer company which produces two colors(white and black) of each model every year.

Bike_model	Manuf_year	Color
M2011	2008	White
M2001	2008	Black
M3001	2013	White
M3001	2013	Black
M4006	2017	White
M4006	2017	Black

Here columns COLOR and MANUF_YEAR are dependent on BIKE_MODEL and independent of each other. In this case, these two columns can be called as multivalued dependent on BIKE_MODEL. The representation of these dependencies is shown below:

BIKE_MODEL → → MANUF_YEAR

BIKE_MODEL → → COLOR

This can be read as "BIKE_MODEL multidetermined MANUF_YEAR" and "BIKE_MODEL multidetermined COLOR".

4.6 JOIN DEPENDENCY

- If a table can be recreated by joining multiple tables and each of these tables will have a subset of the attributes of the table, then the table is in Join Dependency. It is a generalization of Multivalued Dependency
- Join Dependency can be related to 5NF, wherein a relation is in 5NF, only if it is already in 4NF and it cannot be decomposed further.

Example:

<Employee>

EmpName	EmpSkills	EmpJob (Assigned Work)
Tom	Networking	EJ001
Harry	Web Development	EJ002
Katie	Programming	EJ002

The above table can be decomposed into the following three tables;

Therefore, it's not in 5NF:

<Employee Skills>

EmpName	EmpSkills
Tom	Networking
Harry	Web Development
Katie	Programming

<Employee Job>

EmpName	EmpJob
Tom	EJ001
Harry	EJ002
Katie	EJ002

<Job Skills>

EmpSkills	EmpJob
Networking	EJ001
Web Development	EJ002
Programming	EJ002

Our Join Dependency

{(EmpName,EmpSkills),(EmpName,EmpJob),(EmpSkills,EmpJob)}

The above relations have join dependency, so they are not in 5NF. That would mean that a join relation of the above three relations is equal to our original relation <Employee>.

Self-Check Exercise

1. What do you mean by join dependency?
2. What is Multivalued Dependency?
3. How do you achieve multi-valued dependency for a table?

4.7 SUMMARY

- Functional dependency (FD) is a set of constraints between two attributes in a relation.
- A database is normalized to ensure that the relational tables in the database contain only related information and store data only at single location in the database.
- The inference rule is a type of assertion which can be used to derive additional functional dependency from the initial set.
- Database normalization is a database schema design technique, by which an existing schema is modified to minimize redundancy and dependency of data.
- Multivalued dependency occurs when two attributes in a table are independent of each other but, both depend on a third attribute.
- Join Dependency can be related to 5NF, wherein a relation is in 5NF, only if it is already in 4NF and it cannot be decomposed further.
- A relation is in 5NF if it is in 4NF and does not contain any join dependency and joining should be lossless.
- 5NF is satisfied when all the tables are broken into as many tables as possible in order to avoid redundancy

4.8 REFERENCES

“Fundamentals of Database Systems”, Elmasri Navrate, 6th edition, 2013, Pearson
“Data base Systems design”, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition
“Database Systems, The Complete Book”, Hector Garcia-Molina, Jeffrey D. Ullman and Jennifer Widom, 6th impression, 2011, Pearson
“Data base Management Systems”, Raghu Rama Krishnan, Johannes Gehrke, 3rd Edition, 2003, McGraw Hill

4.9 FURTHER READINGS

Starting with Oracle, by John Day and Craig Van Slyke
Database Management Concepts, by Henry F. Korth
Fundamentals of Database Systems, by Navathe

4.10 Practice Exercises

1. What are the steps to normalize a relational table to first normal form?
2. What do you mean by functional dependency in a relational table?
3. What are the various steps to normalize a relational table to second normal form?
4. What are the various steps to normalize a relational table to third normal form?
5. How can you eliminate transitive dependency in a relational table?
6. List the various advantages of normalization.
7. What is trivial functional dependency?
8. What is fully functional dependency?
9. What is partial dependency in a relational table?
10. What is a primary key in a relational table?
11. What are the various objectives of normalization?

B.Sc.(DATA SCIENCE)

SEMESTER-III

DATABASE MANAGEMENT SYSTEM

UNIT V: STRUCTURED QUERY LANGUAGE

STRUCTURE

5.0 Objectives

5.1 Introduction to SQL

5.2 Data Types

5.3 DDL, DML and DCL (Parts of SQL)

5.4 Querying Database Tables

5.5 Data Definition Language (DDL)

5.6 Creating Tables

5.7 Inserting and Updating values into Table

5.8 Summary

5.0 OBJECTIVE

Understanding basic sql commands and architecture and structure of the language.

5.1 INTRODUCTION TO SQL

Structured Query Language (SQL, called as “Sequel”) is the first query based domain-specific programming language. It is used to manipulate and store relational databases or data in the relational database management system called as RDBMS. Simply it is very closely tied with the relational model. The major job of SQL is to create tables or relational databases, insert data, edit or update data, read and delete data in tables or relational databases. SQL handle data having relations between variables and entities (structured data). SQL solves the queries from the relational databases by using English language. It deals with all the RDBMS standard database based languages like Oracle, MS-Access, Informix, MySQL, mariaDB, postgresSQL and SQL Server. SQL is 4GL declarative language (4GL) that is based on procedural elements or procedural statements. Note that all the keywords of Structure query language are in the uppercase and is not case sensitive. SQL statements are text lines dependent and so a single statement of SQL can be used in multiple text line. Number of actions on the database can be performed by SQL statements. Relational algebra and relational calculus statements are solved by SQL. SQL commands execute any statement for any RDBMS by using SQL engine i.e. procedure to interpret the statements in query format. In SQL processing number of components like Query Engine , optimization Engine, classic query engine and Query dispatcher are included. Classic query engine solves the non-SQL queries. In SQL multiple records can be easily accessed by single SQL command. SQL scope is data definition (creation and modification), data manipulation (insert, update and delete), data query and data access control. It is the first commercial languages. In 1970, SQL first it used Edgar F. Codd’s relational model. Initially in 1970, SQL was developed by Donald D. Chamberlin and Raymond F. Boyce at IBM by studying the relational model of Edgar F. Codd. First name of SQL was SEQUEL (Structured English Query Language). It also developed a System R at IBM San Jose Research Laboratory that was the first relational model. First Relational Software Inc. a software company approved and used as first RDBMS in 1978 (Now this software company is known as Oracle Corporation). In 1979, it developed first Oracle RDBMS. In 1986, SQL became a standard approved by American National Standards Institute (ANSI) and further in 1987 standardized by International Organization for Standardization (ISO). The SQL language has been standardized by the ANSI X3H2 Database Standards Committee. Two of the latest standards are SQL-92 and SQL-99. Over the years, each vendor of relational databases has introduced new commands to extend their particular implementation of SQL. Oracle8i's implementation of the SQL language conforms to the "Entry Level" SQL-99 standards and is partially compliant with the Transitional, Intermediate, and Full levels of SQL-99. The goal of SQL is used for performing the C.R.U.D (Create, Retrieve, Update & Delete) operations on relational databases. Also administrative tasks on database performed by SQL. Administration tasks are to provide security to database and backup of data.

In the relational model, data is stored in structures called relations or *tables*. Each table has one or more attributes or *columns* that describe the table. In relational databases, the table is the fundamental building block of a database application. Tables are used to store data on Employees, Equipment, Materials, Warehouses, Purchase Orders, Customer Orders, etc. Columns in the Employee table, for example, might be Last Name, First Name, Salary, Hire Date, Social Security Number, etc.

SQL statements are issued for the purpose of:

- Data definition - Defining tables and structures in the database (DB).
- Data manipulation - Inserting new data, Updating existing data, Deleting existing data, and Querying the Database (Retrieving existing data from the database).

Another way to say this is the SQL language is actually made up of

1) the Data Definition Language (DDL) used to create, alter and drop schema objects such as tables and indexes, and

2) The Data Manipulation Language (DML) used to manipulate the data within those schema objects.

Basically SQL has five different languages in combined form. These are:

1. DQL (Data Query Language):

Full form of DQL is Data Query Language. Already stored information in the database can easily be fetched by the DQL. Data can be accessed by DQL commands. DQL commands are used to select data, view data from the table and deals with table variables and to create temp variables etc. SELECT command is the only command in DQL. The syntax is:

SELECT * FROM Table name;

Here * may be any table variable(s) or field(s) or attribute(s) or column(s).

For example, if you want to display all the columns (whole data) from student table, then command is:

SELECT * FROM student

2. DDL (Data Definition Language):

Full form of DDL is Data Definition Language. Table schemas are defined by DDL commands. DDL commands are used to create and alter the database object and database variables in the RDBMS. The various commands are CREATE TABLE , CREATE DATABASE, ALTER TABLE etc. Commonly used commands in DDL are CREATE, ALTER, TRUNCATE and DROP.

i). CREATE: CREATE command is used to create a database and database object like a table, index, view, trigger, stored procedure, etc. The general syntax is:

CREATE TABLE Table-name (attribute1 datatype, attribute2, datatype.....);

Here attribute1, 2..... are the column name or field name and data type is attribute data type.

ii). ***ALTER:*** To set the database and restruct the database object etc. by ALTER command. The general syntax is:

ALTER TABLE Table-name ADD attribute datatype;

iii). ***TRUNCATE:*** *To remove all the variables, fields or data from the table, this command is used. It empties a table. The general syntax is:*

TRUNCATE TABLE Table-name;

iv). ***DROP:*** To remove all the database fields, variables, the database and database object DROP command is used. The general syntax is:

DROP TABLE Table-name;

3. DCL (Data Control Language):

Full form of DCL is Data Control Language. Job of DCL is to control for accessing of database and allow user for specific tasks. In otherwords it is used for user and permission management. It controls the access to the database. It is used for providing and taking back the access rights on the database and database objects. Mainly GRANT and REVOKE are the DCL commands.

i). ***GRANT Command:*** To provide access right to the user, GRANT Command is used. The general syntax is :

GRANT INSERT, DELETE ON Table-name TO user-name;

ii). ***REVOKE Command:*** To take back access right from the user, REVOKE Command is used. It also cancels all the access right of the user from the database and the database object. The general syntax is:

REVOKE ALL ON Table-name FROM user-name;

4. TCL (Transaction Control Language):

Full form of TCL is Transaction Control Language. It is used for handling transactions in the database. Data integrity during transactions in the multi-user environment is checked by this command. Job of TCL commands is to rollback and commit data updation in the database. Commonly used TCL commands are COMMIT, ROLLBACK, SAVEPOINT, and SET TRANSACTION.

i). ***COMMIT Command:*** It is used to save or apply the modification in the database.

ii). ***ROLLBACK Command:*** It is used to undo the modification.

iii). ***SAVEPOINT Command:*** It is used to temporarily save a transaction. Note that the transaction can roll back to this point according to requirement.

5. DML (Data Manipulation Language):

Full form of DML is Data Manipulation Language. It is used for inserting, updating and deleting data from the database. Manipulation of data in RDBMS easily managed by the DML Commands and the various operations (adding, deleting, updating) in this are done by using INSERT INTO TableName, UPDATE tableName set data and DELETE FROM TableName etc. Commonly used DML commands are INSERT INTO, DELETE FROM, UPDATE.

i). **INSERT INTO:** It is used to add data to the database table. The general syntax is:

INSERT INTO Table-name (Attribute1, Attribute2,.....) VALUES (data1, data2,.....);

ii). **UPDATE:** It is used for updation of data in the database or table. Also a condition added by using WHERE clause in the command. The general syntax is:

UPDATE Table-name SET attribute = value WHERE condition;

iii). **DELETE:** It is used to remove data from the database table. Also condition can be added using the WHERE clause to remove a specific data according to the condition. The general syntax is:

DELETE FROM Table-name WHERE condition;

The various language elements in the SQL language are subdivided as:

- **Clauses** used for constituent components of queries and statements. It is optional.
- **Expressions** are applied for producing of data in the tables consisting of columns and rows of data
- **Predicates** is used for specification of conditions in the statements on the bases of logics or Boolean values. It control the queries flow control.
- **Queries** that is used to retrieve the data based on specific conditions and is most useful elements in the SQL.
- **Statements** are program flow, control transactions, sessions, connections, diagnostics etc.

Note that every SQL statements ends with semicolon (";") as statement terminator.

Applications of SQL

Followings are the various applications of SQL:

- It defines and describes the data for users.
- It allows users to access data in RDBMS.
- It is used for manipulation of data in the database.
- It also allows for embedding of one SQL modules into another and also link with pre-compilers and libraries.

- It is used for creation of databases and allow to drop databases and tables.
- It allows users to stored procedure linked with databases, to create view, to deals with functions in a database.
- It allows users to set permissions on procedures, tables and views.

5.2 DATA TYPES

Data type is used for to specify the type of the data for particular attribute or for any object in the table or database. In SQL every variable, column and expression has a related data type. Data types are used during creation of the tables or databases. According to your requirement, you can select a data type for a table column. Followings are the basic data types used in SQL:

i). **VARCHAR2:** It is of Character data type that has letters, numbers and punctuation. The syntax for this data type is:

VARCHAR2(size)

where *size* is the maximum number of alphanumeric characters the column can hold.

For example VARCHAR2(25) can hold up to 25 alphanumeric characters. In SQL or latest version of Oracle, the maximum size of a VARCHAR2 column is 8,000 bytes. The VARCHAR data type is a synonym for VARCHAR2. It is recommended to use VARCHAR2 instead of VARCHAR.

ii). **NUMBER:** It is of Numeric data type that contain integer or floating point numbers only. The syntax for this data type is:

NUMBER(precision, scale)

where *precision* is the total size of the number including decimal point and *scale* is the number of places to the right of the decimal.

For example, NUMBER(6,2) can hold a number between -999.99 and 999.99.

iii). **DATE:** It is of Date and Time data type that has a date and time portion in the format: DD-MON-YY HH:MI:SS. No additional information is needed when specifying the DATE data type. If no time component is supplied when the date is inserted, the time of 00:00:00 is used as a default. The output format of the date and time can be modified to conform to local standards. In general, to manipulate the time portion of the DATE datatype, one must make use of the TO_DATE and TO_CHAR SQL functions.

iv). **RAW:** It id free form binary data that contain binary data up to 255 characters. Data type LONG RAW can contain up to 2 gigabytes of binary data. RAW and LONG RAW data cannot be indexed and can not be displayed or queried in SQL*Plus. Only one RAW column is allowed per table.

v). **LOB:** It is of Large Object data types. These include BLOB (Binary Large Object) and CLOB (Character Large Object). More than one LOB column can appear in a table. These data

types are the preferred method for storing large objects such as text documents (CLOB), images, or video (BLOB).

Note that there are six types of data types used in the SQL with range are in the below tabular form:

i). Exact Numeric Data Types:

DATA TYPE	FROM	TO
Bigint	-9,223,372,036,854,775,808	9,223,372,036,854,775,807
Int	-2,147,483,648	2,147,483,647
smallint	-32,768	32,767
tinyint	0	255
bit	0	1
decimal	$-10^{38} + 1$	$10^{38} - 1$
numeric	$-10^{38} + 1$	$10^{38} - 1$
money	-922,337,203,685,477.5808	+922,337,203,685,477.5807
smallmoney	-214,748.3648	+214,748.3647

ii). Approximate Numeric Data Types:

DATA TYPE	FROM	TO
Float	$-1.79E + 308$	$1.79E + 308$
Real	$-3.40E + 38$	$3.40E + 38$

iii). Date and Time Data Types:

DATA TYPE	FROM	TO
Datetime	Jan 1, 1753	Dec 31, 9999
Smalldatetime	Jan 1, 1900	Jun 6, 2079
Date	Stores a date like June 30, 1991	
Time	Stores a time of day like 12:30 P.M.	

Note – Here, datetime has 3.33 milliseconds accuracy where as smalldatetime has 1 minute accuracy.

iv). Character Strings Data Types:

Sr.No.	DATA TYPE & Description
1	Char Maximum length of 8,000 characters.(Fixed length non-Unicode characters)
2	Varchar Maximum of 8,000 characters.(Variable-length non-Unicode data).
3	varchar(max) Maximum length of 2E + 31 characters, Variable-length non-Unicode data (SQL Server 2005 only).
4	Text Variable-length non-Unicode data with a maximum length of 2,147,483,647 characters.

v). *Unicode Character Strings Data Types:*

Sr.No.	DATA TYPE & Description
1	Nchar Maximum length of 4,000 characters.(Fixed length Unicode)
2	Nvarchar Maximum length of 4,000 characters.(Variable length Unicode)
3	nvarchar(max) Maximum length of 2E + 31 characters (SQL Server 2005 only).(Variable length Unicode)
4	Ntext Maximum length of 1,073,741,823 characters. (Variable length Unicode)

vi). *Binary Data Types:*

Sr.No.	DATA TYPE & Description
1	Binary Maximum length of 8,000 bytes(Fixed-length binary data)
2	Varbinary Maximum length of 8,000 bytes.(Variable length binary data)
3	varbinary(max) Maximum length of 2E + 31 bytes (SQL Server 2005 only). (Variable length Binary data)
4	Image Maximum length of 2,147,483,647 bytes. (Variable length Binary Data)

vii). Misc Data Types:

Sr.No.	DATA TYPE & Description
1	sql_variant Stores values of various SQL Server-supported data types, except text, ntext, and timestamp.
2	Timestamp Stores a database-wide unique number that gets updated every time a row gets updated
3	Uniqueidentifier Stores a globally unique identifier (GUID)
4	Xml Stores XML data. You can store xml instances in a column or a variable (SQL Server 2005 only).
5	Cursor Reference to a cursor object
6	Table Stores a result set for later processing

A column may be specified with a NULL or NOT NULL constraint meaning the column may or may not be left blank, respectively. This check is made just before a new row is inserted into the table. By default, a column is created as NULL if no option is given. In addition to specifying NOT NULL constraints, tables can also be created with constraints that enforce referential integrity (relationships among data between tables). Constraints can be added to one or more columns, or to the entire table.

Each table may have one PRIMARY KEY that consists of a single column containing no NULL values and no repeated values. A PRIMARY KEY with multiple columns can be designated using the ALTER TABLE command. We create primary keys with NOT NULL constraints to uphold entity integrity.

Up to 255 columns may be specified per table. Column names and table names must start with a letter and may not contain spaces or other punctuation except for the underscore character. Column names and table names are case insensitive.

5.3 DDL, DML AND DCL (PARTS OF SQL)

There are four major parts of SQL, Which are given below:

- ✓ DDL
- ✓ DML
- ✓ DCL
- ✓ DQL

i). Data Definition Language or Data Description Language (DDL): DDL statements are used to define the database structure or schema. It is used for creation and modification of database objects such as tables, indices and users. DDL statements are same as to define data structure in to a computer programming. Basic statements used by DDL are `CREATE`, `ALTER`, and `DROP`. DDL was firstly used by Codasyl database model in establishing the relational model and afterward it was used by SQL. Some of the DDL commands are as:

- ✓ `CREATE` - to create objects in the database
- ✓ `ALTER` - alters the structure of the database
- ✓ `DROP` - delete objects from the database
- ✓ `TRUNCATE` - remove all records from a table, including all spaces allocated for the records are removed
- ✓ `COMMENT` - add comments to the data dictionary
- ✓ `RENAME` - rename an object

ii). Data Manipulation Language (DML): DML statements are used for managing data within schema objects. It is used as computer programming language for insertion, deletion and updation or editing or modification purposes in a database. Some operators are used by DML and it is the sublanguage of SQL. These operators are used both for reading or selecting and writing the data in and from database. Some of the DML commands are as:

- ✓ `SELECT` - retrieve data from the a database
- ✓ `INSERT` - insert data into a table
- ✓ `UPDATE` - updates existing data within a table
- ✓ `DELETE` - deletes all records from a table, the space for the records remain
- ✓ `MERGE` - UPSERT operation (insert or update)
- ✓ `CALL` - call a PL/SQL or Java subprogram
- ✓ `EXPLAIN PLAN` - explain access path to data
- ✓ `LOCK TABLE` - control concurrency

iii). Data Control Language (DCL): DCL is used for controlling the access to data stored in a database or provide an authorization. Syntax are very close to computer programming language. Some of the DCL commands are as:

- ✓ GRANT - gives user's access privileges to database
- ✓ REVOKE - withdraw access privileges given with the GRANT command
- ✓ COMMIT - save work done
- ✓ SAVEPOINT - identify a point in a transaction to which you can later roll back
- ✓ ROLLBACK - restore database to original since the last COMMIT
- ✓ SET TRANSACTION - Change transaction options like isolation level and what rollback segment to use

iv) Data Query Language (DQL): DQL statements are used for performing queries on the data within schema objects. The purpose of DQL commands is to get the schema relation based on the query passed to it. SELECT statement or SELECT Command is mostly useful command for DQL. Here data is taken from database on the bases of some condition by adding FROM or WHERE data manipulators.

5.4 QUERYING DATABASE TABLES

A Query is a set of instruction in the relational database management system (RDBMS). It is to get the data according to the statement or requirement from the database. For that purpose query tables are used. Query Table is used for **preparing the data** for easy reporting and analysis. Query can be done either from single table or multiple table by using **SQL SELECT** command. The various operations performed are used for report creation, data sharing and also to create another query from the Query Table over an existing Query Table. You can create Query Tables for filtering datasets, batching datasets together (union), transforming data, applying SQL query functions, joining datasets and more. Also some of the condition can taken for completing the exact query. The general syntax used are:

SELECT * FROM Table-name

WHERE condition

A complete SQL process is as shown in the figure 5.1 below:

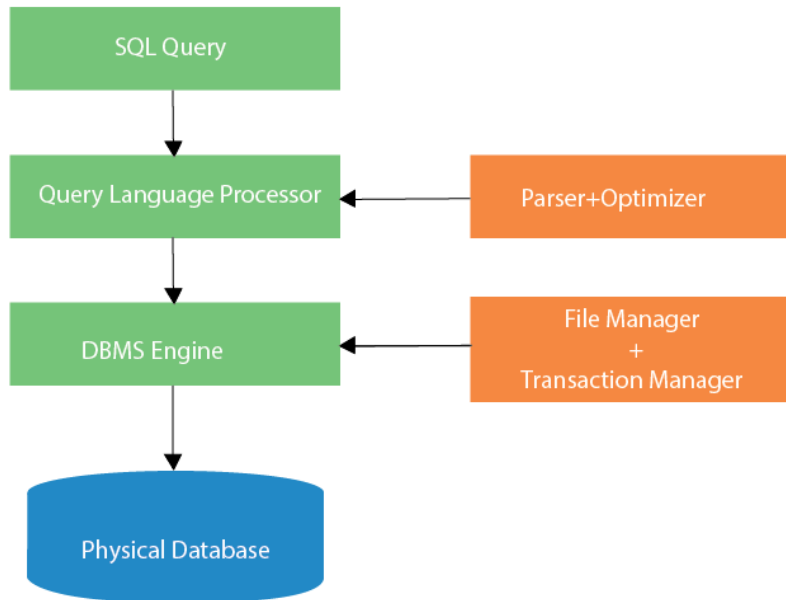


Figure 5.1

For example to fetch the student name from the STUDENT table or database, query can be written as:

```
SELECT student-name from STUDENT;
```

IF you take a condition to fetch the data for rollno 122 then the query can be written as:

```
SELECT student-name from STUDENT;
```

```
WHERE rollno=122
```

SELECT command or statement for query purpose: The basic syntax consists of four clauses as written in the below syntax of SELECT command or SELECT statement:

```
SELECT <attribute>
```

```
FROM <table-name>
```

```
WHERE <condition or boolean predicate to pick rows or columns>
```

```
[ ORDER BY <attribute> ];
```

First two clauses are compulsory from the four clauses. These are SELECT and FROM. Other two are optional. Optional clauses are WHERE and ORDER BY. The four basic clauses of a SELECT commands are:

- The SELECT clause is used for specification of attribute name separated by commas. You can select * (an asterisk character) for retrieval of all the columns.
- FROM clause is used to get the data from the table or database.

- Conditional statement takes place by using the WHERE clause. Here also boolean predicate (logical statements) can be used.
- The ORDER BY clause gives us a way to order the display of the rows in the result of the statement.

5.5 DATA DEFINITION LANGUAGE (DDL)

DDL is used for the creation of database or tables. Also view, indexes, cursor, triggers etc. can be created by using the popular DDL Create command.

Create Command: It is used to create various objects like

- ✓ Table
- ✓ View
- ✓ Index
- ✓ User
- ✓ Cursor
- ✓ Trigger
- ✓ Function
- ✓ Procedure

Create is a DDL i.e. data definition language statement and is used to create the above object. Like all other DDL statements create command cannot be roll backed

5.6 CREATING TABLES

A table or relation is defined as collection of interrelated records of tuples, where records (Tuples) are further collection of interrelated fields (attributes) and fields are defined as set of values defined over single domain. Create table command is DDL statement used to create a table. The general syntax of create table is

```
Create table table-name
(column1 datatype size(),
column2 datatype size(),
.....
.....
.....
columnN datatype size());
```

Some of the examples are:

1. Creating table from scratch:

```
create table student  
(rollno      number(4),  
name        varchar2(20),  
class       varchar2(12),  
address     varchar2(20));
```

2. Create a copy of an existing table:

Create table is also used to create a copy of an existing table rather than recreating it from scratch if required. The general syntax is:

```
Create table newtable-name  
As  
Select * from existing-table;
```

For example: create a copy of student table

```
Create table tempstu  
As  
Select * from student
```

3. Create a copy of structure of table:

Create table is also used to create the copy of the structure of an existing table in such case only structure will be copied and not contents. The general syntax is:

```
Create table newtable-name  
As  
Select * from existingtable  
Where (wrong condition)
```

For example: create a copy of structure of student table

```
create table temp-stu-stru  
as  
select * from student  
where 1=2;
```

4. Create a table with primary key:

```
Create table ptable
```

```
(rollno number (4) primary key,  
name varchar2 (20),  
class varchar2 (20)  
address varchar2 (20));
```

Here rollno is primary, hence note that

- ✓ It should not be null.
- ✓ It should be unique.

5. Create a table with other constraints:

```
create table pconstraint  
(Sid number (4) references ptable(rollno),  
regno varchar2 (4) unique key,  
name Varchar2 (20) Not Null,  
marks number (3) check (marks<=100));
```

Here

- ✓ Sid references to rollno of ptable
- ✓ regno should be unique and not duplicated.
- ✓ name should not be empty
- ✓ marks should be less than 100

6. Creating and Dropping Indexes:

An index is a performance-tuning method of allowing faster retrieval of records. An index creates an entry for each value that appears in the indexed columns. By default, Oracle creates B-tree indexes. An index is a data structure that affords rapid lookup of data in a table. An index is normally created on those columns of a table used to look up data. For example, in the employee table, the key *ssn* can be used to look up the rest of an employee's information. Creating a index on the *ssn* field would be accomplished by the following statement. The general syntax of creating index is

```
create index index_name  
on tablename(column_name);
```

For example,

```
CREATE INDEX idx_stu  
ON student(rollno) ;
```


Note, however, that by default, Oracle will automatically create an index on any column designated as a primary key. A good naming convention to use for indexes is to start with the letters `idx_` followed by the name of the table, and finally the name of the column(s) used to create the index.

➤ **Rename An Index:**

An index can be renamed by using the alter command which is a DDL statement, the syntax to rename an index is given below:

```
ALTER INDEX index_name RENAME TO new_index_name;
```

Indexes can be dropped using the DROP INDEX statement the general syntax to drop an index is

```
Drop index index_name
```

For example, to drop the `idx_stu` index, one could submit:

```
DROP INDEX idx_stu;
```

Dropping a table (using the DROP TABLE statement) automatically drops all indexes that have been built on columns in that table.

To see which indexes are defined in a schema, submit a query to the `USER_INDEXES` view of the database system catalog:

```
SELECT index_name, table_name FROM user_indexes ;
```

ID31	STUDENT
IDX_SEC	STUDENT
IDX_GAGAN	GAGAN
IDX_HONEY	HONEY
NDXCOM	TEMP

➤ **Creating and Dropping Views:**

In the SQL language, a view is a representation of one or more tables. A view can be used to hide the complexity of relationships between tables or to provide security for sensitive data in tables. In the following example, a limited view of the employee table is created. When a view is defined, a SQL statement is associated with the view name. Whenever the view is accessed, the SQL statement will be executed. In general View is a part of table it describes only certain

portion of a table. View are used to apply security to database by hiding certain portion of table. The general syntax of view is:

➤ **Creating View from Single Table:**

```
Create view viewname  
As  
Select * from tablename
```

For example a view having empno, ename and sal of EMP table can be created by following command

```
Create view vemp  
as  
select empno, ename, sal from emp
```

➤ **Creating View From Multiple Tables:**

A view can be created from more than one table. In such a case it hide the complexity associated with joins the general syntax to create view from multiple table is

```
Create view vname  
As  
select col1,col2,col3.....col N from table1, table2...tableM  
where condition;
```

For example

```
Create view vjoin  
As  
Select emp.ename,dept.dname from emp,dept  
Where emp.deptno=deptno.deptno;
```

The above view gives employee name with their department name, here employee name is taken from emp table and department name is taken from dept table

➤ **Truncate:**

Truncate is the DDL statement which is used to removes **all rows** from a table. The operation cannot be rolled back and no triggers will be fired. As such, TRUCATE is faster and doesn't use as much undo space as a DELETE. The general syntax of truncate statement is

```
Truncate table tablename;
```

For example to truncate an emp table give the following command at SQL prompt.

Truncate table emp;

Drop:

The DROP command is a DDL statement which is used to remove following objects

- ✓ **Table**
- ✓ **View**
- ✓ **Index**

from the database. All the tables' rows, indexes and privileges will also be removed. No DML triggers will be fired. The operation cannot be rolled back. The general syntax of DROP statement is:

Drop Object Objectname;

For example to remove a table the general syntax will be look like as given below

Drop table tablename;

For example:

Drop table emp;

DROP is DDL commands, therefore once table is dropped it cannot be roll backed.

Note: in Oracle 10g a table can be "undropped" by using Flashback as given below:

```
SQL> FLASHBACK TABLE emp TO BEFORE DROP;
```

Alter Table:

It is another DDL statement which is commonly used in database modification. it has two objective:

- ✓ it is used to add/delete a column from existing table
- ✓ it is used to change the size of column

The general syntax of alter statement is

Alter table tablename

Add/modify columnname datatype (size);

1. Adding New Column to Student Table Having Following Columns:

- ✓ RollNo
- ✓ Name
- ✓ Class

Suppose we want to add address column to the student table, for this give the following command at SQL prompt.

```
alter table student  
add address varchar2 (30);
```

2. Deleting Column from an existing table:

Now if you want to delete class column from above student table the give the following command at SQL prompt.

```
Alter table student  
Drop column Class;
```

1. Modifying the length of Name column from 10 characters to 20 characters.

Now suppose student table's column name have 10 character length and you want to increase it from 10 to 20, and then give the following command at SQL prompt.

```
Alter table student  
Modify name varchar2 (20);
```

5.7 INSERTING AND UPDATING VALUES INTO A TABLE

Inserting which is straightforward task that is used for inserting a single row or a single record in the database. So a new row can be created by using the insert command or operation. Updating is another simple task for modification database. At a time one record can be updated. But you can also update whole sets of records at once.

Inserting a New Record: If we want to insert a new record into the database or into a table, then INSERT command is used. For example, if you want to insert a new tuple or row or record into the STUDENT table. The value for Rollno should be 122, Student-name should be "Deepak" and city should be "BATALA" and this can be added by using the INSERT Command.

```
insert into student (rollno,sname,city)  
values (122,'Deepak','BATALA')
```

Multiple values (rows or tuples or records) can inserted by using the below command as:

```
/* multi row insert */  
insert into student (rollno,sname,city)  
values (122,'Deepak','BATALA')
```

```
(123,'Balram','Amritsar')
```

Inserting Default Values: A table or database can be created or defined to take default values for specific columns. Default values without having any specific information or values can be inserted into the rows as:

```
create table Student (id integer default 0)
```

If you want default values to all the attributes or to a single attribute then DEFAULT keyword is used for this purpose. For example, DEFAULT keyword is used as:

```
insert into Student values (default)
```

You may also explicitly specify the column name, which you'll need to do anytime you are not inserting into all columns of a table:

```
insert into Student (rollno) values (default)
```

Overriding a Default Value with NULL: If you want override the default value by NULL value into the table, then lets take the first example or consider the table as:

```
create table Student (rollno integer default 0, sname VARCHAR(10))
```

If you want a row with a NULL value for rollno, then this can be written as:

You can explicitly specify NULL in your values list:

```
insert into Student (rollno, sname) values (null, 'Mona')
```

Modifying Records in a Table (Updating): You want to modify values for some or all rows in a table. For example, you might want to increase the salaries of everyone in department 20 by 10%. The following result set shows the DEPTNO, ENAME, and SAL for employees in that department:

```
Select deptno,ename,sal  
From emp  
Where deptno = 20
```

Order by 1,3

DEPTNO	ENAME	SAL
20	SMRIT	800
20	AMRIK	1100
20	JOLLY	2975
20	SURAJ	3000
20	RAJA	3000

You want to bump all the SAL values by 10%. Use the UPDATE statement to modify existing rows in a database table. For example:

```
update emp
set sal = sal*1.10
where deptno = 20
```

Use the UPDATE statement along with a WHERE clause to specify which rows to update; if you exclude a WHERE clause, then all rows are updated. The expression SAL*1.10 in this solution returns the salary increased by 10%. When preparing for a mass update, you may wish to preview the results. You can do that by issuing a SELECT statement that includes the expressions you plan to put into your SET clauses. The following SELECT shows the result of a 10% salary increase:

```
Select deptno,
      ename,
      sal
As orig_sal,
      sal*.10 as amt_to_add,
      sal*1.10 as new_sal
From emp
Where deptno=20
Order by 1,5
```

DEPTNO	ENAME	ORIG_SAL	AMT_TO_ADD	NEW_SAL
20	SMRIT	800	80	880
20	AMRIK	1100	110	1210
20	JOLLY	2975	298	3273
20	SURAJ	3000	300	3300
20	RAJA	3000	300	3300

The salary increase is broken down into two columns: one to show the increase over the old salary, and the other to show the new salary.

5.8 SUMMARY

Structured Query Language (SQL, called as “Sequel”) is the first query based domain-specific RDBMS programming language. Basically SQL has five different languages in combined form. These are DDL, DML, DCL, DQL and TCL. Data type is used for to specify the type of the data for particular attribute or for any object in the table or database. There are number of Data types used in SQL statements like VARCHAR2, Number, Date etc. There are four major parts of SQL, DDL, DML, DCL and DQL. Basic statements used by DDL are CREATE, ALTER, and DROP. Some of the DML commands are SELECT, INSERT, UPDATE, DELETE, MERGE, CALL etc. Some of the DCL commands are GRANT, REVOKE, COMMIT, SAVEPOINT and ROLLBACK etc. A Query is a set of instruction in the relational database management system (RDBMS). It is to get the data according to the statement or requirement from the database. For that purpose query tables are used. Create Command is used to create various objects like Table, View, Index, User, Cursor, Trigger, Function and Procedure etc. Inserting which is straightforward task that is used for inserting a single row or a single record in the database. So a new row can be created by using the insert command or operation. Updating is another simple task for modification database. At a time one record can be updated.

Question for practice:

- Q1. What do you know about SQL?
- Q2. How SQL was evolved?
- Q3. What are the basic SQL? Explain all.
- Q4. What do you mean by Data Types? Explain all the Data types used in SQL.
- Q5. Which data type is mostly used?

- Q6. Explain all the DDL commands with syntax and example?
- Q7. Explain all the DML commands with syntax and example?
- Q8. Explain all the DCL commands with syntax and example?
- Q9. Explain all the parts of SQL?
- Q10. What do you mean by Query?
- Q11. What are the various query statements used?
- Q12. What are the various Create commands with all the options?
- Q13. How can you insert values into the table?
- Q14. How can you update values in a table?

B.Sc.(DATA SCIENCE)
SEMESTER-III
DATABASE MANAGEMENT SYSTEM

UNIT VI: DATA MANIPULATION LANGUAGE

STRUCTURE

6.0 Objectives

6.1 Select Statement

6.2 Joins in SQL

6.3 Aggregate Functions

6.4 String Functions

6.5 SQL Having Sub Query

6.6 SQL Having Co-related Sub Query

6.7 Summary

6.0 OBJECTIVE

To understand syntax, semantic and working of data manipulation language.

6.1 SELECT STATEMENT

Select statement is a DML statement which is used to retrieve the records from a table. It **returns a result set of records**, from one table (relational database) or multiple tables. It retrieves zero row or multiple rows from one table or more database tables or database views.

Various form of SELECT statements are – simple SELECT, using special operators, aggregate functions, group by clause, sub query, joins, co-related sub query, union clause, exist operator.

Select statement is use for the following purposes:

- ✓ **To select selective data**
- ✓ **To sort the data either in ascending of descending order**
- ✓ **To select the data according to groups**
- ✓ **To filter the data according to user requirements**

The general syntax of select statement is give below

Select col1,col2... coln from tablename

Where condition

Group by column

Having condition

Order by column;

We will explain complete select statement by considering the various parts of select statement.

i). Simple Select (Complete Selection)

In this type of select statement we have to give all the column name of table to select them. For example if you have to select all the information regarding EMP table then give the following command

Select empno,ename,job,mgr,hiredate,sal,comm,deptno from emp;

7369	SMITH	CLERK	7902	17-DEC-80	5770		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	2600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20

7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	5075		10
7788	SCOTT	ANALYST	7566	19-APR-87	10000		20
7839	KING	PRESIDENT		17-NOV-81	5200		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	23-MAY-87	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	1131		30
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	3300		10

ii). Using Special Operator (*) - Short Form of Selections

In this mode you simply have to put an asterisk (*) instead of writing all the column names e.g. for the above statement we can write

*Select * From Emp*

7369	SMITH	CLERK	7902	17-DEC-80	5770		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	2600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	5075		10
7788	SCOTT	ANALYST	7566	19-APR-87	10000		20
7839	KING	PRESIDENT		17-NOV-81	5200		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	23-MAY-87	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	1131		30

7902	FORD	ANALYST	7566	03-DEC-81	3000	20
7934	MILLER	CLERK	7782	23-JAN-82	3300	10

iii). Partial Selection

In such type of selection we will select some column from a list according to user requirements. The data is not completely selected as in the above case, but selective data will be shown according to user requirement. This gives you a feel of database abstraction. For example

Select empno,ename,sal from emp;

7369	SMITH	5770
7499	ALLEN	2600
7521	WARD	1250
7566	JONES	2975
7654	MARTIN	1250
7698	BLAKE	2850
7782	CLARK	5075
7788	SCOTT	10000
7839	KING	5200
7844	TURNER	1500
7876	ADAMS	1100
7900	JAMES	1131
7902	FORD	3000
7934	MILLER	3300

iv). Filtered Selection

In such type of selection we will use where condition, where condition is used to restrict the number of rows i.e. it filters the rows of an EMP table

*select * from emp*

where deptno=10;

7782	CLARK	MANAGER	7839	09-JUN-81	5075	10
7839	KING	PRESIDENT		17-NOV-81	5200	10
7934	MILLER	CLERK	7782	23-JAN-82	3300	10

v). Sorting Data Using Select Statement

Select statement with order by statement is used to sort the data either in ascending or descending order. For example to sort the emp data in ascending order according to salary is given by following statement

```
Select * from emp
order by sal;
```

7876	ADAMS	CLERK	7788	23-MAY-87	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	1131		30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7499	ALLEN	SALESMAN	7698	20-FEB-81	2600	300	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	3300		10
7782	CLARK	MANAGER	7839	09-JUN-81	5075		10
7839	KING	PRESIDENT		17-NOV-81	5200		10
7369	SMITH	CLERK	7902	17-DEC-80	5770		20
7788	SCOTT	ANALYST	7566	19-APR-87	10000		20

vi). Group by Clause (Group by Expression)

Select statement can be used with group by expression to give the information according to group or in groups. It should be noted that group by clause is used with aggregate function. The following three major points should be considered while executing group by statement.

1. There should be an aggregate function in group by statement.
2. Group by statement can not be used with select * statement.
3. The column by which grouping is to be done should be present in select statement.

For example:

Select deptno, max(sal) from emp

Group by deptno;

10	5200
20	10000
30	2850

The above statement will give maximum salary of each department. We can restrict or filter the rows given by group by expression using having clause. For example

Select deptno, max(sal) from emp

Group by deptno

Having count()>5;*

30	2850
-----------	-------------

The above statement will give the maximum salaries of those departments in which there are more than 5 employees.

Examples of Select Statement:

1. Find All the Information of Employees Who Works In 10 Number Department

*Select * from emp*

Where deptno=20;

7369	SMITH	CLERK	7902	17-DEC-80	5770	20
7566	JONES	MANAGER	7839	02-APR-81	2975	20
7788	SCOTT	ANALYST	7566	19-APR-87	10000	20
7876	ADAMS	CLERK	7788	23-MAY-87	1100	20
7902	FORD	ANALYST	7566	03-DEC-81	3000	20

2. Find all the information of employee whose commission is Null

*Select * from emp
where comm is null;*

7369	SMITH	CLERK	7902	17-DEC-80	5770		20
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	5075		10
7839	KING	PRESIDENT		17-NOV-81	5200		10
7900	JAMES	CLERK	7698	03-DEC-81	1131		30
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	3300		10

3. Find information of all the employees who have more than two year experience.

*Select * from emp
Where (sysdate-hiredate)>2*365;*

7369	SMITH	CLERK	7902	17-DEC-80	5770		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	2600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	5075		10
7788	SCOTT	ANALYST	7566	19-APR-87	10000		20
7839	KING	PRESIDENT		17-NOV-81	5200		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	23-MAY-87	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	1131		30
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	3300		10

6.2 JOINS IN SQL

SQL joins are used to join two or more tables that is with the help of SQL joins we can find or retrieve an information from two or more than two tables. There are four major types of joins.

- ✓ Equi joins
- ✓ Non equi joins
- ✓ Self joins
- ✓ Outer joins

a). Equi Join

The join which retrieve an information from two or more table by using the '=' operator is non as equi join. For example if you want to find the employee name and department of the employee then you will have to use equi join on EMP and DEPT table. One thing should be noted that there should be some common column in two table on which join has to be performed. as we know both emp and dept table have common column i.e. deptno.

SQL> Select emp.ename, dept.dname from EMP, DEPT

Where emp.deptno=dept.deptno;

SMITH	RESEARCH
ALLEN	SALES
WARD	SALES
JONES	RESEARCH
MARTIN	SALES
BLAKE	SALES
CLARK	ACCOUNTING
SCOTT	RESEARCH
KING	ACCOUNTING
TURNER	SALES
ADAMS	RESEARCH
JAMES	SALES
FORD	RESEARCH
MILLER	ACCOUNTING

b). Non equi join:

The joins that are not based on equality sign are known as non equi join. It can have <, > <= or >= sign it should not have “=” sign.

SQL>Select emp.empno, emp.ename,emp.sal,salgrade.grade from emp, salgrade

Where emp.sal>salgrade.losal and emp.sal<salgrade.hisal;

7876	ADAMS	1100	1
7900	JAMES	1131	1
7521	WARD	1250	2
7654	MARTIN	1250	2
7844	TURNER	1500	3
7499	ALLEN	2600	4
7566	JONES	2975	4
7698	BLAKE	2850	4
7369	SMITH	5770	5
7782	CLARK	5075	5
7839	KING	5200	5
7934	MILLER	3300	5

In the above example it shows empno, ename and sal from emp table and grade from salgrade table where sal should be greater than losal and less than hisal of salgrade table.

c). Self join

Self join is a join of a table to itself. That is a join of table with itself is known as self join. The name of the table to be joined using self join appear twice in the from clause with alias name. for example if you have to find the employee name and his manager name then following query with self join can be used.

SQL>Select a.ename,b.ename "manager" from emp a, emp b

where a.mgr=b.empno;

SMITH	FORD
ALLEN	BLAKE
WARD	BLAKE
JONES	KING
MARTIN	BLAKE
BLAKE	KING
CLARK	KING
SCOTT	JONES
TURNER	BLAKE
ADAMS	SCOTT
JAMES	BLAKE
FORD	JONES
MILLER	CLARK

Here two aliases of EMP as a and b are created to perform join operation.

d). Outer Join

The outer join return all the rows from one table and the rows from another table that satisfy the join condition. There are two subtypes of outer join

- ✓ Left outer join
- ✓ Right outer join
- **Left Outer Join:** only unmatched rows from left hand side table are retained.
- **Right Outer Join:** only unmatched rows from right hand side are retained

To understand the concept of outer join consider the following example

SQL>Select emp.ename,dept.dname from emp, dept

Where emp.deptno(+)=dept.deptno;

CLARK	ACCOUNTING
KING	ACCOUNTING
MILLER	ACCOUNTING
SMITH	RESEARCH
ADAMS	RESEARCH
FORD	RESEARCH
SCOTT	RESEARCH
JONES	RESEARCH
ALLEN	SALES
BLAKE	SALES
MARTIN	SALES
JAMES	SALES
TURNER	SALES
WARD	SALES
	OPERATIONS

SQL>Select emp.ename,dept.dname from emp,dept

Where emp.deptno=dept.deptno(+)

SMITH	RESEARCH
ALLEN	SALES
WARD	SALES
JONES	RESEARCH
MARTIN	SALES
BLAKE	SALES
CLARK	ACCOUNTING
SCOTT	RESEARCH
KING	ACCOUNTING
TURNER	SALES

ADAMS	RESEARCH
JAMES	SALES
FORD	RESEARCH
MILLER	ACCOUNTING

6.3 AGGREGATE FUNCTIONS (SQL FUNCTIONS)

SQL functions are built into Oracle and are available for use in various appropriate SQL statements. Do not confuse SQL functions with user functions written in PL/SQL. If you call a SQL function with an argument of a data type other than the data type expected by the SQL function, Oracle implicitly converts the argument to the expected data type before performing the SQL function. If you call a SQL function with a null argument, the SQL function automatically returns null. The only SQL functions that do not necessarily follow this behavior are CONCAT, NVL, and REPLACE.

In the syntax diagrams for SQL functions, arguments are indicated by their data types. When the parameter "function" appears in SQL syntax, replace it with one of the functions described in this section. Functions are grouped by the data types of their arguments and their return values. The various types of functions used in SQL are given below:

- ✓ **Aggregate functions**
- ✓ **Arithmetic functions**
- ✓ **String functions**
- ✓ **Date functions**

➤ **Aggregate Functions**

Aggregate functions return a single result row based on groups of rows, rather than on single rows. Aggregate functions can appear in select lists and in ORDER BY and HAVING clauses. They are commonly used with the GROUP BY clause in a SELECT statement, where Oracle divides the rows of a queried table or view into groups. In a query containing a GROUP BY clause, the elements of the select list can be aggregate functions, GROUP BY expressions, constants, or expressions involving one of these. Oracle applies the aggregate functions to each group of rows and returns a single result row for each group.

If you omit the GROUP BY clause, Oracle applies aggregate functions in the select list to all the rows in the queried table or view. You use aggregate functions in the HAVING clause to eliminate groups from the output based on the results of the aggregate functions, rather than on the values of the individual rows of the queried table or view. Many (but not all) aggregate functions that take a single argument accept these options:

- DISTINCT causes an aggregate function to consider only distinct values of the argument expression.
- ALL causes an aggregate function to consider all values, including all duplicates.

There are six major aggregate functions:

1. Min
2. Max
3. Sum
4. Avg
5. Count
6. variance

To explain the concept of aggregate function consider an EMP table having following records.

SQL> select * from emp;

EMPO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17-DEC-80	5770		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	2600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	5075		10
7788	SCOTT	ANALYST	7566	19-APR-87	10000		20
7839	KING	PRESIDENT		17-NOV-81	5200		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	23-MAY-87	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	1131		30
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	3300		10
9898	HONEY						
56	BABLU						
344	GAGANDEEP						

MIN()

Min function is used to find the minimum value from a set of values e.g. if you have to find the minimum salary from EMP table the give following command at SQL prompt.

```
SQL> select min(sal) from emp;
```

1100

MAX()

Max function is used to find the maximum value from a set of values e.g. if you have to find the maximum salary from EMP table the give following command at SQL prompt.

```
SQL> select max(sal) from emp;
```

10000

SUM()

Sum function is used to find the sum of set of values e.g. if you want to find the sum of salary of all employee then use the following command

```
SQL> select sum(sal) from emp;
```

47001

AVG()

Avg function is used to find the average of set of values For example if you want to find the average of salary of all the employee then use the following command

```
SQL> select avg(sal) from emp;
```

3357.2143

COUNT

Count() function is used to count the number of records e.g. If you want to count all the employee who got the salary then use the following command

```
SQL> select count(sal) from emp;
```

14

VARIANCE ()

VARIANCE returns variance of *expr*. You can use it as an aggregate or analytic function. Oracle calculates the variance of *expr* as follows:

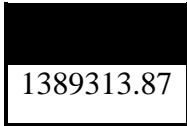
- ✓ 0 if the number of rows in *expr* = 1
- ✓ VAR_SAMP if the number of rows in *expr* > 1

If you specify DISTINCT, you can specify only the *query_partition_clause* of the *analytic_clause*. The *ORDER_BY_clause* and *windowing_clause* are not allowed.

Examples:

The following example calculates the variance of all salaries in the emp table:

```
SELECT VARIANCE(sal) "Variance" FROM emp;
```



1389313.87

Examples of Functions:

Problem: Write a query to find the second highest salary from EMP table

```
Select max(sal) from emp  
Where sal < (select max(sal) from emp);
```

Problem: Write a query to find the third highest salary from EMP table

```
Select max(sal) from emp  
Where sal <(select max(sal) from emp  
Where sal <(Select max(sal) from emp));
```

➤ Arithmetic Operator and Functions:

In SQL there are four main operator used in the queries of sub queries and are given below.

1. + (PLUS)
2. - (MINUS)
3. * (MULTIPLICATION)
4. / (DIVISION)

+ (PLUS):

This operator is used to find the sum of two or more variables. E.g. If you want to find the sum of 2 and 2 then use the following command

SQL> select (2+2) "Sum" from dual;

4

- **(MINUS):**

This operator is used to find the difference of two or more variables. For example If you want to find the subtract 3 from 10 then use the following command

SQL> select (10 – 3) "Minus" from dual;

7

* **(MULITPLY):**

This operator is used to find the product of two or more variables. E.g. If you want to find the product of 12 and 2 then use the following command

SQL> select (12*2) "Multiplication" from dual;

24

/ **(DIVISION):**

This operator is used to find the result of division of two numbers. E.g. if you want to find the result when 4 divided by 2 then use the following statement

SQL> select (4/2) from dual;

2

➤ **Arithmetic Functions or Number Functions:**

Arithmetic functions Number functions accept numeric input and return numeric values. Most of these functions return values that are accurate to 38 decimal digits. The transcendental functions COS, COSH, EXP, LN, LOG, SIN, SINH, SQRT, TAN, and TANH are accurate to 36 decimal digits. The transcendental functions ACOS, ASIN, ATAN, and ATAN2 are accurate to 30 decimal digits. The number functions are:

Mod()	Log()	Asin()	Cos()
Greatest()	Power()	Atan()	Tan()
Least()	Sin()	Atan2()	Sinh()
Sqrt()	Cos()	Sin()	Cosh()

MOD():

This function gives remainder after division of two given numbers. For example mod(7,3) gives 2 as quotient as 1 as remainder the following function gives 1 as output.

SQL> select mod(7,3) from dual;

[REDACTED]
1

GREATEST():

GREATEST returns the greatest of the list of *exprs*. All *exprs* after the first are implicitly converted to the data type of the first *expr* before the comparison. Oracle compares the *exprs* using nonpadded comparison semantics. Character comparison is based on the value of the character in the database character set. One character is greater than another if it has a higher character set value. If the value returned by this function is character data, its data type is always VARCHAR2. For example consider the following:

SQL>Select greatest ('HARRY', 'HARRIOT', 'HAROLD') "GREATEST" from dual;

[REDACTED]
HARRY

SQL>Select greatest(2,4,5,6,77,3) from dual;

[REDACTED]
77

LEAST ():

LEAST returns the least of the list of *exprs*. All *exprs* after the first are implicitly converted to the datatype of the first *expr* before the comparison. Oracle compares the *exprs* using nonpadded comparison semantics. If the value returned by this function is character data, its datatype is always VARCHAR2. For example

SQL>Select least('MANIK','ZARRIOT','ZAROLD') "LEAST" from dual;

[REDACTED]
MANIK

SQL>Select least(2,3,4,5,33,1,55) from dual;

1

SQRT():

SQRT is another important function that returns square root of n . The value n cannot be negative. SQRT returns a "real" result. For example

SQL>Select sqrt(26) "Square root" from dual;

5.09901951

SELECT SQRT(25) "Square root" FROM DUAL;

5

POWER():

POWER is a mathematical function that returns m raised to the n th power. The base m and the exponent n can be any numbers, but if m is negative, n must be an integer. For example

SQL>Select power(3,2) "Power" from dual;

9

LOG ():

LOG returns the logarithm, base m , of n . The base m can be any positive number other than 0 or 1 and n can be any positive number. For example

SQL>Select log(10,100) "Log base 10 of 100" from dual;

2

SIN():

sin() function give the value of sine. it should be noted that value given in brackets as argument are not in degree, but in radian hence sin(90) will not give 1 as output. For example

SQL> select sin(21) from dual;

.83665564

COS():

cos() function give the value of cosine. it should be noted that value given in brackets as argument are not in degree, but in radian. For example.

SQL> select cos(90) from dual;

-.4480736

TAN():

Tan() function give the value of tangent. it should be noted that value given in brackets as argument are not in degree, but in radian. For example

SQL> select tan(30) from dual;

-6.405331

CEIL():

CEIL is another important arithmetic function that returns smallest integer greater than or equal to *n*. For example

SQL>Select ceil(215.8) "Ceiling" from dual;

216

FLOOR():

FLOOR returns largest integer equal to or less than *n*. for example consider following

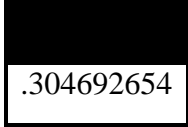
SELECT FLOOR(15.7) "Floor" FROM DUAL;

15

ASIN():

ASIN returns the arc sine of *n*. Inputs are in the range of -1 to 1, and outputs are in the range of $\pi/2$ to $\pi/2$ and are expressed in radians. Example

SELECT ASIN(.3) "Arc_Sine" FROM DUAL;



.304692654

ATAN ()

ATAN returns the arc tangent of n . Inputs are in an unbounded range, and outputs are in the range of $-\pi/2$ to $\pi/2$ and are expressed in radians. Example

```
SELECT ATAN(.3) "Arc_Tangent" FROM DUAL;
```




.291456794

ATAN2():

ATAN2 returns the arc tangent of n and m . Inputs are in an unbounded range, and outputs are in the range of $-\pi$ to π , depending on the signs of n and m , and are expressed in radians. ATAN2(n,m) is the same as ATAN2(n/m) . Example

```
SELECT ATAN2(.3, .2) "Arc_Tangent2" FROM DUAL;
```

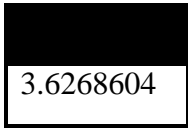


.982793723

SINH():

sinh() function computes the value of sine of hyperbolic. again the argument passed is in radian not in degree.

```
SQL> select sinh(2) from dual;
```



3.6268604

COSH():

cosh() function computes the value of cosine of hyperbolic. again the argument passed is in radian not in degree.

```
SQL> select cosh(40) from dual;
```



1.177E+17

```
SQL> select exp(2) from dual;
```

7.3890561

ABS():

ABS is a mathematical function that returns the absolute value of n that is it convert negative value to positive and does not effect on positive value

Select ABS(-15) "Absolute" from dual;

15

ACOS():

ACOS returns the arc cosine of *n*. Inputs are in the range of -1 to 1, and outputs are in the range of 0 to pi and are expressed in radians. For example

SELECT ACOS(.3)"Arc_Cosine" FROM DUAL;

1.26610367

RANK():

RANK is an analytic function. It computes the rank of each row returned from a query with respect to the other rows returned by the query, based on the values of the *value_exprs* in the *ORDER_BY_clause*. Rows with equal values for the ranking criteria receive the same rank. Oracle then adds the number of tied rows to the tied rank to calculate the next rank. Therefore, the ranks may not be consecutive numbers. Examples:

The following statement ranks the employees within each department based on their salary and commission. Identical salary values receive the same rank and cause nonconsecutive ranks.

SELECT deptno, ename, sal, comm,

RANK() OVER (PARTITION BY deptno ORDER BY sal DESC, comm) as rk

FROM emp;

DEPTNO	ENAME	SAL	COMM	RK
10	KING	5000		1
10	CLARK	2450		2
10	MILLER	1300		3
20	SCOTT	3000		1

20	FORD	3000		1
20	JONES	2975		3
20	ADAMS	1100		4
20	SMITH	800		5
30	BLAKE	2850		1
30	ALLEN	1600	300	2
30	TURNER	1500	0	3
30	WARD	1250	500	4
30	MARTIN	1250	1400	5
30	JAMES	950		6

ROUND (number function):

ROUND returns *n* rounded to *m* places right of the decimal point. If *m* is omitted, *n* is rounded to 0 places. *m* can be negative to round off digits left of the decimal point. *m* must be an integer. For example

SQL>Select round(15.193,1) "Round" from dual;

15.2

SQL>Select round(15.193,-1) "Round" from dual;

20

ASCII:

ASCII returns the decimal representation in the database character set of the first character of *char*. If your database character set is 7-bit ASCII, this function returns an ASCII value. If your database character set is EBCDIC Code, this function returns an EBCDIC value. There is no corresponding EBCDIC character function. For example

SQL>Select ASCII('Q') from dual;

81

TRUNC (number function):

TRUNC returns n truncated to m decimal places. If m is omitted, n is truncated to 0 places. m can be negative to truncate (make zero) m digits left of the decimal point. For example

SQL>Select TRUNC(15.79,1) "Truncate" from dual;

15.7

SQL>Select TRUNC(15.79,-1) "Truncate" from dual;

10

6.4 STRING FUNCTIONS

String functions are SQL built in functions that are operated upon strings. These are mainly used to process the strings that is with the help of string functions you can convert lower characters to upper character, upper characters to lower, find length of string , concatenate two or more string etc. There are various string functions available in SQL which are given below:

7369	SMITH	CLERK	7902	17-DEC-80	5770		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	2600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	5075		10
7788	SCOTT	ANALYST	7566	19-APR-87	10000		20
7839	KING	PRESIDENT		17-NOV-81	5200		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	23-MAY-87	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	1131		30
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	3300		10

9898	SIMI	0	500
56	HARMAN	0	500
344	SIMMI	0	700
111	JASKARAN	0	876

LOWER():

Lower function is used to convert all upper character into lower character e.g. if you want to convert all the ename of EMP table to lower character then use the following command.

SQL> select lower(ename) from emp;

Smith
Allen
Ward
Jones
Martin
Blake
Clark
Scott
King
Turner
Adams
James
Ford
Miller
Simi
Harman
Simmi
jaskaran

UPPER():

Upper function is used to convert all lower character into upper character e.g. if you want to convert all the ename of EMP table to upper character then use the following command.

SQL> select upper(ename) from emp;

```
SMITH
ALLEN
WARD
JONES
MARTIN
BLAKE
CLARK
SCOTT
KING
TURNER
ADAMS
JAMES
FORD
MILLER
SIMI
HARMAN
SIMMI
JASKARAN
```

INITCAP():

Initcap makes the first character or name capital all other character will be represented in small letters. For example

SQL> select initcap(ename) from emp;

```
Smith
Allen
Ward
Jones
```

Martin
Blake
Clark
Scott
King
Turner
Adams
James
Ford
Miller
Simi
Harman
Simmi
Jaskaran

LENGTH():

Length is another string function that is used to find the length of string i.e. it describes the length of string in number of character e.g. if you want to find the length of all the ename of EMP table then give the following command.

SQL> select length(ename) from emp;

5
5
4
5
6
5
5
5
4

```
6
5
5
4
6
4
6
5
8
```

SQL> select ltrim(ename,'S') from emp;

```
MITH
ALLEN
WARD
JONES
MARTIN
BLAKE
CLARK
COTT
KING
TURNER
ADAMS
JAMES
FORD
MILLER
IMI
HARMAN
IMMI
JASKARAN
```

SQL> select rtrim(ename,'S') from emp;

SMITH
ALLEN
WARD
JONE
MARTIN
BLAKE
CLARK
SCOTT
KING
TURNER
ADAM
JAME
FORD
MILLER
SIMI
HARMAN
SIMMI
JASKARAN

SQL> select chr(67) "Char" from dual;

C

SOUNDEX:

SOUNDEX returns a character string containing the phonetic representation of *char*. This function allows you to compare words that are spelled differently, but sound alike in English. For example

SQL>Select ename from emp

Where Soundex(ename) = Soundex('SMYTHE');

SMITH

VSIZE:

VSIZE returns the number of bytes in the internal representation of *expr*. If *expr* is null, this function returns null. For example

SQL>Select ename, VSize (ename) "BYTES" from emp

Where deptno = 10;

CLARK	5
KING	4
MILLER	6

SUBSTR:

SUBSTR returns a portion of *char*, beginning at character *m*, *n* characters long.

- If *m* is 0, it is treated as 1.
- If *m* is positive, Oracle counts from the beginning of *char* to find the first character.
- If *m* is negative, Oracle counts backwards from the end of *char*.
- If *n* is omitted, Oracle returns all characters to the end of *char*. If *n* is less than 1, a null is returned.

Floating-point numbers passed as arguments to SUBSTR are automatically converted to integers.

SQL>Select Substr('ABCDEFGF',3,4) "Substring" from dual;

CDEF

LPAD():

Lpad is a string function that is used to padded or fill the empty space on the left hand side of string by some special or specified character. For example

SQL> select lpad(ename,15,'\$') from emp;

\$\$\$\$\$\$\$\$SMITH
\$\$\$\$\$\$\$\$ALLEN

```

$$$$$$$$$WARD
$$$$$$$$$JONES
$$$$$$$$$MARTIN
$$$$$$$$$BLAKE
$$$$$$$$$CLARK
$$$$$$$$$SCOTT
$$$$$$$$$KING
$$$$$$$$$TURNER
$$$$$$$$$ADAMS
$$$$$$$$$JAMES
$$$$$$$$$FORD
$$$$$$$$$MILLER
$$$$$$$$$SIMI
$$$$$$$$$HARMAN
$$$$$$$$$SIMMI
$$$$$$$$$JASKARAN

```

RPAD():

Rpad is a string function that is used to padded or fill the empty space on the right hand side of string by some special or specified character. For example

SQL> select rpad(ename,15,'^') from emp;

```

SMITH^^^^^^^^^^^
ALLEN^^^^^^^^^^^
WARD^^^^^^^^^^^
JONES^^^^^^^^^^^
MARTIN^^^^^^^^^^
BLAKE^^^^^^^^^^^
CLARK^^^^^^^^^^^
SCOTT^^^^^^^^^^^

```

```

KING^^^^^^^^^^^^^
TURNER^^^^^^^^^^^
ADAMS^^^^^^^^^^^^
JAMES^^^^^^^^^^^^
FORD^^^^^^^^^^^^^^
MILLER^^^^^^^^^^^^
SIMI^^^^^^^^^^^^^^
HARMAN^^^^^^^^^^^^
SIMMI^^^^^^^^^^^^^
JASKARAN^^^^^^^^^^

```

Date Functions:

Date functions are inbuilt function of SQL that are associated with data manipulation. The various date function with their use are given below for consideration.

a). SYSDATE(): SYSDATE is a date function which gives the current date from the system. It should be noted that the date displayed will depend upon current system date.

SQL> select sysdate from dual;

```

07-APR-07

```

LAST_DAY(): LAST_DAY returns the date of the last day of the month that contains *d*. You might use this function to determine how many days are left in the current month.

SQL>Select SYSDATE, LAST_DAY(SYSDATE) "Last",
LAST_DAY(SYSDATE) - SYSDATE "Days Left" from dual;

```

02-OCT-05  31-OCT-05  29

```

The following example adds 5 months to the hired ate of each employee to give an evaluation date:

SQL>Select ename, hiredate, TO_CHAR(ADD_MONTHS(LAST_DAY(hiredate), 5))
"Eval Date" from emp;

SMITH	17-DEC-80	31-MAY-81
ALLEN	20-FEB-81	31-JUL-81
WARD	22-FEB-81	31-JUL-81
JONES	02-APR-81	30-SEP-81
MARTIN	28-SEP-81	28-FEB-82
BLAKE	01-MAY-81	31-OCT-81
CLARK	09-JUN-81	30-NOV-81
SCOTT	19-APR-87	30-SEP-87
KING	17-NOV-81	30-APR-82
TURNER	08-SEP-81	28-FEB-82
ADAMS	23-MAY-87	31-OCT-87
JAMES	03-DEC-81	31-MAY-82
FORD	03-DEC-81	31-MAY-82
MILLER	23-JAN-82	30-JUN-82

NEXT_DAY:

NEXT_DAY returns the date of the first weekday named by *char* that is later than the date *d*. The argument *char* must be a day of the week in the date language of your session, either the full name or the abbreviation. The minimum number of letters required is the number of letters in the abbreviated version. Any characters immediately following the valid abbreviation are ignored. The return value has the same hours, minutes, and seconds component as the argument *d*. For example Write a query to returns the date of the next Tuesday after March 15, 1998.

```
SQL>Select NEXT_DAY('15-MAR-98','TUESDAY') "NEXT DAY" from dual;
```

16-MAR-98

TO_CHAR (date conversion):

TO_CHAR converts *d* of DATE datatype to a value of VARCHAR2 datatype in the format specified by the date format *fmt*. If you omit *fmt*, *d* is converted to a VARCHAR2 value in the default date format. The '*nlsparams*' specifies the language in which month and day names and abbreviations are returned. This argument can have this form:

```
'NLS_DATE_LANGUAGE = language'
```


If you omit *nlsparams*, this function uses the default date language for your session.

```
SQL> Select TO_CHAR(HIREDATE, 'Month DD, YYYY') "New date format" from emp Where  
ename = 'BLAKE';
```

```
May 01,  
1981
```

```
SQL> select to_char(sysdate,'yyyy-mm-dd') from dual;
```

```
2007-04-07
```

```
SQL> select to_char(sysdate,'dd-mm-yyyy') from dual;
```

```
07-04-2007
```

```
SQL> select to_char(sysdate,'dd-mon-yyyy') from dual;
```

```
07-apr-2007
```

```
SQL> select to_char(sysdate,'dd-mm-yy') from dual;
```

```
07-04-07
```

```
SQL> select to_char(sysdate,'day') from dual;
```

```
saturday
```

```
SQL> select to_char(sysdate,'day-mon-yyyy') from dual;
```

```
saturday -apr-2007
```

```
SQL> select to_char(sysdate,'dd-day-mon-yyyy') from dual;
```

```
07-saturday -apr-2007
```

```
SQL> select to_char(sysdate,'dd-month-yyyy') from dual;
```

```
07-april -2007
```

```
SQL> select to_char(sysdate,'dd,day-month-yy') from dual;
```

```
07,saturday -april -07
```

6.5 SQL HAVING SUB QUERY

In SQL a Subquery is a query within another query. It is also called nested query. Subquery is actually defined as “**a query that is embedded in WHERE clause of another SQL query**”. SQL sub query actually depends upon the statement and the main query where it is applied. Sub query applied having embedded with HAVING, FROM or WHERE Clauses. The different general syntax are as:

Sub query with FROM clause:

```
SELECT column_name(s)
FROM (SELECT column_name(s) from table_name) as table_alias
WHERE condition;
```

Another syntax is (Sub query using WHERE Clause):

```
SELECT column_name(s)
FROM table_name_1
WHERE column_name expression_operator{=,NOT IN,IN, <,>, etc}(SELECT
column_name(s) from table_name_2);
```

Sub query with HAVING Clause is as:

```
SELECT column_name(s)
FROM table_name_1
WHERE condition
GROUP BY column_name(s)
HAVING Aggregate_function(column_name)expression_operator{=,
<,>}(SELECT column_name(s) from table_name_2);
```

Here **SELECT column_name(s)** is used for selection of attribute or field or column data from the table, **FROM** is used to for database name from where you want to get data, **WHERE condition** is used to for specification of condition on the basis of which it selects the data, **GROUP BY column_name(s)** is used to group rows, having similar values into summary rows and at end **HAVING condition** is used to filter groups based on the specified condition.

For example: To get the number of employees in each department, SQL statement is”

```
SELECT departmentid, count_employees
FROM (SELECT count(DISTINCT employeeid) AS "count_employees",departmentid
FROM employees GROUP BY departmentid) AS employee_summary
ORDER BY count_employees;
```

Subquery in WHERE Clause filter the rows after comparing a column in the main table with the results of the subquery. For example, SQL Query is as:

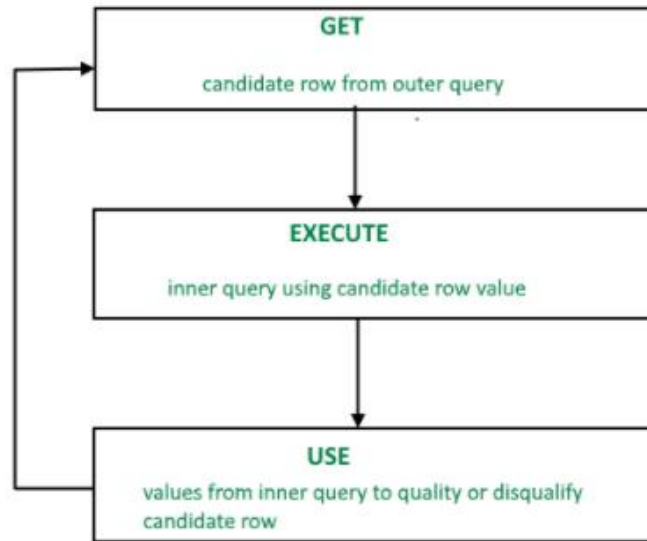
```
SELECT departmentname
FROM department
WHERE head IN (SELECT employeeid::varchar
FROM employees
WHERE city = 'Manhattan');
```

Another example of Subquery in HAVING Clause is as:

```
SELECT d.departmentname,count(e.employeeid)
FROM department as d INNER JOIN employees as e
ON d.departmentid::varchar = e.departmentid
GROUP BY d.departmentname
HAVING count(e.employeeid)>(SELECT count(employeeid) FROM employees WHERE city
= 'New Delhi');
```

6.6 SQL HAVING CO-RELATED SUB QUERY

A correlated subquery is also called as synchronized subquery. It is a subquery **that uses the data values from the outer query i.e. inner query within outer query. It is** row-by-row processing. Every subquery is processed once for every row of the outer query.



A correlated subquery is executed once for every row processed by its parent statement like **SELECT**, **UPDATE** etc. The general syntax is as:

```

SELECT column1, column2, ....
FROM table1 outer
WHERE column1 operator
      (SELECT column1, column2
      FROM table2
      WHERE expr1 =
          outer.expr2);
  
```

A correlated subquery is one way of reading every row in a table and comparing values in each row against related data. It is used whenever a subquery must return a different result or set of results for each candidate row considered by the main query. In other words, you can use a correlated subquery to answer a multipart question whose answer depends on the value in each row processed by the parent statement.

The following query gets employees values whose salary is greater than the average salary of all employees:

```

SELECT
      Empid, name, salary
FROM employee
WHERE salary > (SELECT
                  AVG(salary)
  
```

FROM employee

xi). SQL using union clause:

The UNION clause applied for combining the result-set of two or more SELECT statements. Note that each SELECT command within UNION must have the same number of columns with similar data types and are in the same order. The general syntax is:

```
SELECT column_name(s) FROM table1
UNION
SELECT column_name(s) FROM table2;
```

UNION ALL Syntax

Another syntax by using the UNION ALL clause is as:

```
SELECT column_name(s) FROM table1
UNION ALL
SELECT column_name(s) FROM table2;
```

For example, if you want to returns the cities (only distinct values) from both the "Customers" and the "Suppliers" table, then it will be:

```
SELECT City FROM Customers
UNION
SELECT City FROM Suppliers
ORDER BY City;
```

Another example is UNION With WHERE clause:

```
SELECT City, Country FROM Customers
WHERE Country='Germany'
UNION
SELECT City, Country FROM Suppliers
WHERE Country='Germany'
ORDER BY City;
```

xii) SQL with exist operator:

Use of EXISTS operator is to test for the existence of any record in the subquery and note that TRUE value returned if the subquery returns one or more records. The general syntax is:

```
SELECT column_name(s)
FROM table_name
WHERE EXISTS
(SELECT column_name FROM table_name WHERE condition);
```

For example EXISTS returns TRUE value for the suppliers having a product price less than 20:

```
SELECT SupplierName
FROM Suppliers
WHERE EXISTS (SELECT ProductName FROM Products WHERE Products.SupplierID =
Suppliers.supplierID AND Price < 20);
```

Another example of using NOT EXIST operator is to get all departments that do not have any employees, the solution is as:

```
SELECT department_id, department_name
FROM departments d
WHERE NOT EXISTS (SELECT 'X'
FROM employees
WHERE department_id
= d.department_id);
```

6.7 SUMMARY

In DML Select statement is mainly used. Select statement is a DML statement which is used to retrieve the records from a table. Various form of SELECT statements are – simple SELECT, using special operators, aggregate functions, group by clause, sub query, joins, co-related sub query, union clause, exist operator. SQL joins are used to join two or more tables that is with the help of SQL joins we can find or retrieve an information from two or more than two tables. There are four major types of joins as Equi joins, Non equi joins, Self joins, Outer joins etc. SQL functions are built into Oracle and are available for use in various appropriate SQL statements. Do not confuse SQL functions with user functions written in PL/SQL. The various types of functions used in SQL are **Aggregate functions, Arithmetic functions, String functions, Date functions etc.** There are six major aggregate functions as Min, Max, Sum, Avg, Count and variance. String functions are SQL built in functions that are operated upon strings. These are mainly used to process the strings that is with the help of string functions you can convert lower characters to upper character, upper characters to lower, find length of string , concatenate two or more string etc. In SQL a Subquery is a query within another query. It is also called nested query. Subquery is actually defined as “**a query that is embedded in WHERE clause of another SQL query**”. SQL sub query actually depends upon the statement and the main query where it is applied. Sub query applied having embedded with HAVING, FROM or WHERE Clauses.

6.8 PRACTICE EXERCISES

Q1. What are the various DML statements used?

Q2. What do you know about SQL statement?

- Q3. Explain the simple SELECT statement with syntax and examples?
- Q4. Explain the SELECT statement having special operators with syntax and examples?
- Q5. Explain the SELECT statement using the aggregate function with syntax and examples?
- Q7. Explain the SELECT statement using group clause with syntax and examples?
- Q8. What do you mean by Sub-Query?
- Q9. Explain the SELECT statement having sub-query statements with syntax and examples?
- Q10. What are the various Joins used in DML?
- Q11. Explain the SELECT statement using the Join options with syntax and examples?
- Q12. Explain the SELECT statement having co-related query statements with syntax and examples?
- Q13. Explain the SELECT statement using Union option with syntax and examples?
- Q14. Explain the SELECT statement using Exists operator with syntax and examples?

B.Sc.(DATA SCIENCE)
SEMESTER-III
DATABASE MANAGEMENT SYSTEM

UNIT VII: VIEWS

STRUCTURE

7.0 Objective

7.1 Introduction to Views

7.2 Data Independence

7.3 Statements on Join Views

7.4 Dropping a View

7.5 Database Security

7.6 Security Techniques

7.7 Two Phase Locking Techniques

7.8 Summary

7.0 OBJECTIVE

To understand different kind of views available in DBMS.

To understand statements on Join views and dropping a view

To understand database security and different security techniques

7.1 INTRODUCTION TO VIEWS

DBMS has different view from architectural point. A three-level architecture suggested by American National Standards Institute (ANSI) /Standards Planning and Requirements Committee (SPARC) in 1977. It is necessary to view data at different levels of abstraction. The three levels of the architecture are three different views of the data:

1. *External* - individual user view
2. *Conceptual* - community user view
3. *Internal* - physical or storage view

The three level database architecture allows a clear separation of the information meaning (conceptual view) from the external data representation and from the physical data structure layout. A database system that is able to separate the three different views of data is likely to be flexible and adaptable. We now briefly discuss the three different views as:

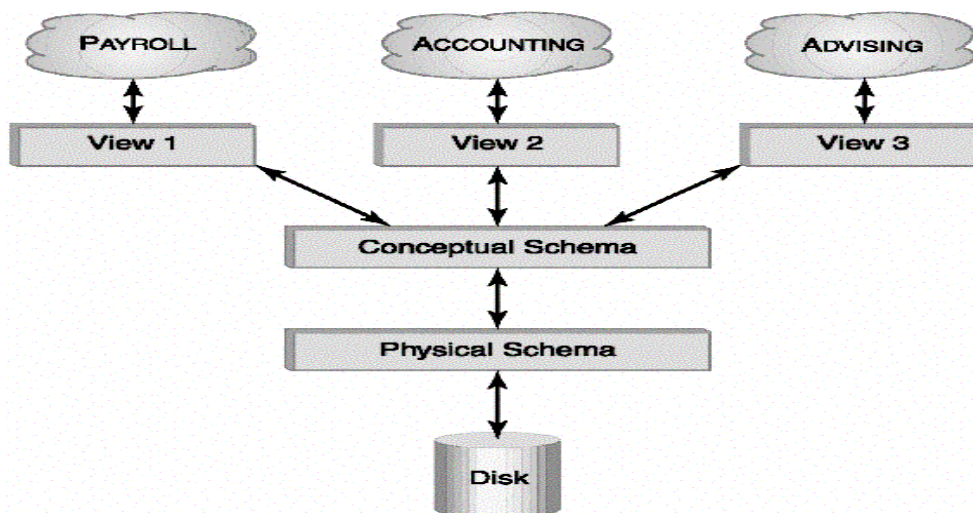


Figure 7.1

1. The external level is the view that the individual user of the database has that is why it also known as User view. This view is often a restricted view of the database and the same database may provide a number of different views for different classes of users. In general, the end users and even the applications programmers are only interested in a subset of the database. For example, a department head may only be interested in the departmental finances and student enrolments but not the library information. The

librarian would not be expected to have any interest in the information about academic staff. The payroll office would have no interest in student enrolments.

2. The conceptual view is the information model of the enterprise and contains the view of the whole enterprise without any concern for the physical implementation. This view is normally more stable than the other two views. In a database, it may be desirable to change the internal view to improve performance while there has been no change in the conceptual view of the database. The conceptual view is the overall community view of the database and it includes all the information that is going to be represented in the database. The conceptual view is defined by the conceptual schema which includes definitions of each of the various types of data.
3. The internal view is the view about the actual physical storage of data. It tells us what data is stored in the database and how. The **physical schema** describes details of how data is stored: tracks, cylinders, indices etc. on the random access disk system. It also typically describes the layout of files and type of files.

Mapping between Data Views: Assuming the three level view of the database, a number of mappings are needed to enable the users working with one of the external views. For example, the payroll office may have an external view of the database that consists of the following information only:

1. Staff number, name and address.
2. Staff tax information e.g. number of dependents.
3. Staff bank information where salary is deposited.
4. Staff employment status, salary level, leaves information etc.

The conceptual view of the database may contain academic staff, general staff, casual staff etc. A mapping will need to be created where all the staff in the different categories are combined into one category for the payroll office. The conceptual view would include information about each staff's position, the date employment started, full-time or part-time, etc.

This will need to be mapped to the salary level for the salary office. Also, if there is some change in the conceptual view, the external view can stay the same if the mapping is changed.

Data is structured in relational tables or databases by using different database objects either by using tables or using views etc. Data dependency and redundancy can be reduced in organization of tables in SQL databases. Big database can be well organized into small tables by splitting the big database by using the normalization. Normalization is used to set the data and databases in normal form for easy use in SQL programming. Note that a relationship can be created with the use of interlinked multiple tables. By using SQL commands, queries applied on tables and solve the problem easily. Complicated queries can be solved from multiple linked tables by using multiple joins.

A VIEW in SQL is similar to create virtual tables which have data from one table or one database or multiple tables or multiple databases. There is no physical existence of databases or tables and also not hold any data. Note that view name is unique in table or database as similar to a SQL table. Data can be easily fetched from the predefined SQL queries from the single database or table or from the multiple tables or databases. Below figure shows the VIEW having a query to join three relational database or tables and create a new virtual database or table from the data of multiple tables or databases.

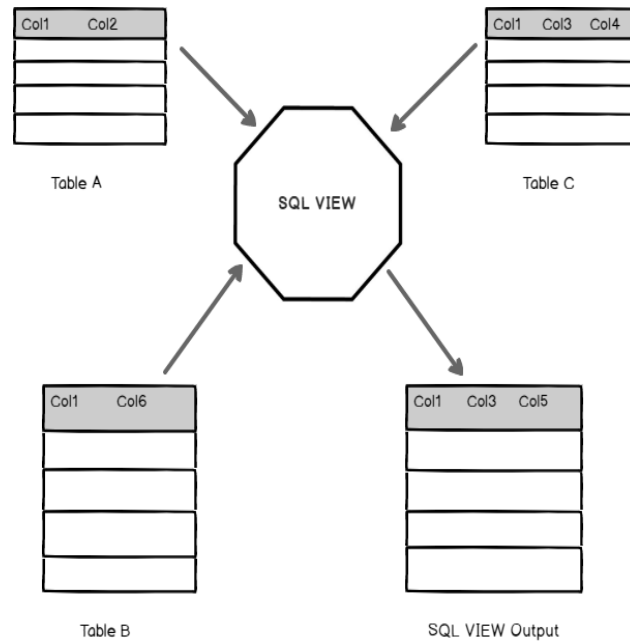


Figure 7.2

In the viewing the database security is also an important factor because there is no physical existence of a VIEW database or VIEW table and only data can be fetched by VIEW command. To create a view using the SQL command, the general syntax is as:

CREATE VIEW view-name AS

SELECT coll, col2,.....coln

FROM Table-name

WHERE condition;

For example to get all records from a table, a SQL VIEW command is:

CREATE VIEW Student-data AS

SELECT *

FROM Student

WHERE class-code=1

If we not wanted all the columns, then SQL VIEW command can be written as:

```
CREATE VIEW Student-data AS
```

```
SELECT rollno, s-name, fees
```

```
FROM Student
```

```
WHERE class-code=1
```

VIEW ENCRYPTION: VIEW can be easily encrypted by using WITH ENCRYPTION command. In past you have checked users to see the view definition. If you do not want users to view the definition, we can encrypt as:

```
CREATE VIEW Demo-View
```

```
WITH ENCRYPTION
```

```
AS
```

```
SELECT *
```

```
FROM Student
```

```
WHERE class-code=1
```

7.2 DATA INDEPENDENCE

To understand the concept of first of all understand the concept of Dependent data. It means such data which is fully dependent upon the data types, data storage location and accessing technique used. **Data independence** plays vital role in dealing with the database techniques and so is advantageous. Data Independence means data structure of the data is completely independent from the storage and accessing of the data. It is completely independent of any storage location, data types and accessing techniques. It means data at the centralized location or centralized DBMS is transparent in nature, i.e. data is completely transparent in centralized database management system. It states about the immunity of user related applications that not changes the structure and definition of the data. There are two levels of data independence:

➤ **Logical data dependency:**

It means logical structure is fully data independent and is also called as **schema definition**. Generally, in any case if any operation or change occurs on the sub set of the fields or attributes or columns in a relation or relational table or database, then in future it have no effects on the existing relation or table after inserting some of new attributes or after deletion of columns or attributes from the same relation. It means generally, logical independence means a program or statement not depends upon the structure of the data. In any case on the change of the structure of data in the program or in the relation will still have access to the database independently.

➤ **Physical data dependency:**

Meaning of physical data independence is hiding the complete storage structure from its relation or user applications. It means there is no effect on the application and storage location of the data in the relation or table or database or in multiple relations. For example, a program can access the data independently and have no effect whether data stored in hard disk or in pendrive or in floppy disc, CD ROM etc.

7.3 STATEMENTS ON JOIN VIEWS

First of let us understand the concept of View. In the SQL language, a view is a representation of one or more tables. A view can be used to hide the complexity of relationships between tables or to provide security for sensitive data in tables. In general View is a part of table it describes only certain portion of a table. View are used to apply security to database by hiding certain portion of table. In SQL Views are kind of virtual tables. A view is similar to the real table having rows and columns as in the real relation or database. View can be easily created by selecting attributes or fields or columns from one database or table or from multiple tables or databases or relations. A View may have a single row or single column or multiple rows or multiple columns depending upon the constraints given in the problem. Now further we discuss the Join View to create a view from the relation.

SQL joins are used to join two or more tables that is with the help of SQL joins we can find or retrieve an information from two or more than two tables. There are four major types of joins.

i). Equi Join:

The join which retrieve an information from two or more table by using the ‘=’ operator is non as equi join. For example if you want to find the employee name and department of the employee then you will have to use equi join on EMP and DEPT table. One thing should be noted that there should be some common column in two table on which join has to be performed. as we know both emp and dept table have common column i.e. deptno.

SQL> Select emp.ename, dept.dname from EMP, DEPT

Where emp.deptno=dept.deptno;

SMITH	RESEARCH
ALLEN	SALES
WARD	SALES
JONES	RESEARCH
MARTIN	SALES
BLAKE	SALES

CLARK	ACCOUNTING
SCOTT	RESEARCH
KING	ACCOUNTING
TURNER	SALES
ADAMS	RESEARCH
JAMES	SALES
FORD	RESEARCH
MILLER	ACCOUNTING

ii). Non equi join:

The joins that are not based on equality sign are known as non equi join. It can have <, > <= or >= sign it should not have “=” sign.

SQL>Select emp.empno, emp.ename,emp.sal,salgrade.grade from emp, salgrade

Where emp.sal>salgrade.losal and emp.sal<salgrade.hisal;

empno	ename	sal	grade
7876	ADAMS	1100	1
7900	JAMES	1131	1
7521	WARD	1250	2
7654	MARTIN	1250	2
7844	TURNER	1500	3
7499	ALLEN	2600	4
7566	JONES	2975	4
7698	BLAKE	2850	4
7369	SMITH	5770	5
7782	CLARK	5075	5
7839	KING	5200	5
7934	MILLER	3300	5

In the above example it shows empno, ename and sal from emp table and grade from salgrade table where sal should be greater than losal and less than hisal of salgrade table.

iii). Self join:

Self join is a join of a table to itself. That is a join of table with itself is known as self join. The name of the table to be joined using self join appear twice in the from clause with alias name. for example if you have to find the employee name and his manager name then following query with self join can be used.

```
SQL>Select a.ename,b.ename "manager" from emp a, emp b
      where a.mgr=b.empno;
```

SMITH	FORD
ALLEN	BLAKE
WARD	BLAKE
JONES	KING
MARTIN	BLAKE
BLAKE	KING
CLARK	KING
SCOTT	JONES
TURNER	BLAKE
ADAMS	SCOTT
JAMES	BLAKE
FORD	JONES
MILLER	CLARK

Here two aliases of EMP as a and b are created to perform join operation.

iv). Outer Join:

The outer join return all the rows from one table and the rows from another table that satisfy the join condition. There are two subtypes of outer join

Left Outer Join: only unmatched rows from left hand side table are retained.

Right Outer Join: only unmatched rows from right hand side are retained

To understand the concept of outer join consider the following example

```
SQL>Select emp.ename,dept.dname from emp, dept
```

```
      Where emp.deptno(+)=dept.deptno;
```

CLARK	ACCOUNTING
KING	ACCOUNTING
MILLER	ACCOUNTING
SMITH	RESEARCH
ADAMS	RESEARCH
FORD	RESEARCH
SCOTT	RESEARCH
JONES	RESEARCH
ALLEN	SALES
BLAKE	SALES
MARTIN	SALES
JAMES	SALES
TURNER	SALES
WARD	SALES
	OPERATIONS

SQL>Select emp.ename,dept.dname from emp,dept

Where emp.deptno=dept.deptno(+)

SMITH	RESEARCH
ALLEN	SALES
WARD	SALES
JONES	RESEARCH
MARTIN	SALES
BLAKE	SALES
CLARK	ACCOUNTING
SCOTT	RESEARCH
KING	ACCOUNTING
TURNER	SALES

ADAMS	RESEARCH
JAMES	SALES
FORD	RESEARCH
MILLER	ACCOUNTING

Join Operators:

JOIN is used to combine related tuples from two relations:

- ✓ In its simplest form the JOIN operator is just the cross product of the two relations.
- ✓ As the join becomes more complex, tuples are removed within the cross product to make the result of the join more meaningful.
- ✓ JOIN allows you to evaluate a join condition between the attributes of the relations on which the join is undertaken.

The notation used is

R JOIN join condition S

JOIN Example

R	ColA	ColB
	A	1
	B	2
	D	3
	F	4
	E	5

R JOIN	R.ColA = S.SColA	S		
	A	1	A	1
	D	3	D	3
	E	5	E	4

S	SColA	SColB
	A	1
	C	2
	D	3
	E	4

R JOIN	R.ColB = S.SColB	S		
	A	1	A	1
	B	2	C	2
	D	3	D	3
	F	4	E	4

➤ Natural Join:

Invariably the JOIN involves an equality test, and thus is often described as an equi-join. Such joins result in two attributes in the resulting relation having exactly the same value. A 'natural join' will remove the duplicate attribute(s).

- In most systems a natural join will require that the attributes have the same name to identify the attribute(s) to be used in the join. This may require a renaming mechanism.

- If you do use natural joins make sure that the relations do not have two attributes with the same name by accident.

➤ **Outer Join:**

Notice that much of the data is lost when applying a join to two relations. In some cases this lost data might hold useful information. An outer join retains the information that would have been lost from the tables, replacing missing data with nulls. There are three forms of the outer join, depending on which data is to be kept.

- LEFT OUTER JOIN - keep data from the left-hand table
- RIGHT OUTER JOIN - keep data from the right-hand table
- FULL OUTER JOIN - keep data from both tables

R LEFT OUTER JOIN R.ColA = S.SColA S

R	ColA	ColB
A	1	
B	2	
D	3	
F	4	
E	5	

S	SColA	SColB
A	1	
C	2	
D	3	
E	4	

Result	ColA	ColB	SColA	SColB
A	1		A	1
D	3		D	3
E	5		E	4
B	2		-	-
F	4		-	-

R RIGHT OUTER JOIN R.ColA = S.SColA S

R	ColA	ColB
A	1	
B	2	
D	3	
F	4	
E	5	

S	SColA	SColB
A	1	
C	2	
D	3	
E	4	

Result	ColA	ColB	SColA	SColB
A	1		A	1
D	3		D	3
E	5		E	4
-	-		C	2

R FULL OUTER JOIN R.ColA = S.SColA S

R	ColA	ColB
A	1	
B	2	
D	3	
F	4	
E	5	

S	SColA	SColB
A	1	
C	2	
D	3	
E	4	

Result	ColA	ColB	SColA	SColB
A	1		A	1
D	3		D	3
E	5		E	4
B	2		-	-
F	4		-	-
-	-		C	2

7.4 DROPPING A VIEW

To delete or to remove a view of a relation or an object view of a table is the dropping of the view. This can be done easily by using the DROP VIEW clause or command or statement. Note that definition of the view can be changed easily by dropping the view and further a new view or structure of the relation or table can be re-created.

DROP views from the database:

A view can always be disposed of with **DROP VIEW <view_name>** command.

DROP VIEW V1;

Note that when we execute the drop view command it removes the view definition. The underlying data stored in the base tables from which this view is derived remains unchanged. A view once dropped can be recreated with the same name. Also note that by dropping a view of a relation, there is no effect on the original data means it has unaffected database. So it is the great tool for end users for dealing the data from the relations or tables or databases, it means the table or database behind the scenes is safe. For this purpose DROP command is used. The purpose is to delete or drop an entire relation or database or table or column of data. It is not wise to remove the whole table. To make it safe we can use the View command from the SQL and further View can be removed any time when it is not required in future. That is why a view is a great option for end users, and dropping a view (instead of a table) is much less frightening.

VIEW Command in the SQL having the general syntax is as:

CREATE VIEW View-name AS

SELECT attribute1, attribute2.....

FROM Table

WHERE Condition

For example if you want to create a student view from the student table, then the SQL VIEW command is used as:

CREATE VIEW StudentView AS

SELECT rollno, sname, fees

FROM Student

WHERE Class-code=1

DROP VIEW Syntax for dropping a view is as written:

DROP VIEW View-name;

For example if you want to remove the view StudentView view table as created in the above example, the SQL command is used as:

DROP VIEW StudentView;

Note the semicolon should be at the end of the line which means database application in this command is complete. After deleting the view if you try to apply any query on the view then there will be an error occur.

7.5 DATABASE SECURITY

To preserve the database integrity, confidentiality and availability, some tools, measures and controls are designed. The finally database is secure by doing so and the above said process is called Data Security. Database security addresses and provides the protection in the following way:

- There must be data in the tables or in a database.
- There should be either a Database Management System (DBMS) or Relational Data Base Management System (RDBMS).
- Any applications or coding associated with the above defined and declared database.
- There should be physical and/or virtual database server with complete in hardware and networking.
- The computing technique/infrastructure should be for accessing the data from the database or table or from a relation

Database security is a challenging and complex endeavor for big data and the database more useable and accessible. It means more usable and accessible the database more is occurrences of threats to the data and needs more security, especially in case of big data.

Now question arises why it is important?

If there be any data breach occurs that create a failure in database maintenance and organization and also to establish the confidentiality of data in a database, then it may be a big loss to data. Following are the factors or consequences that inflict the business or your job that may be harmful after data breach:

- Your inventions, trade secrets and proprietary practices are intellectual property matter and if anybody access it then there is a big loss to you. So need Data Security.
- If your partners and customers are not feel secure during dealing with yours product or business, then this will damage your brand reputation and so need Data Security.

- Some business or jobs cannot continue to handle or operate till data breach problem is resolved.
- Any type of penalties or fine or loss dealing with the financial matters or digital money processing that create a high data threats and so need Data Security to control and avoid it.

Some of the common challenges and threats are:

- i). An insider threat is a major security threat from either a malicious insider or by negligent insider or from an outsider that is harmful to the data in a database.
- ii). Password sharing, weak passwords, accidents to data and other uninformed or unwise user existence create a problem to the important data and so is a big threat.
- iii). Hackers are also one of the major threat to the data and the database management softwares related with open source software industry.
- iv). A database-specific threat that is an insertion in non-SQL as attack or an arbitrary SQL statements that becomes harmful in HTTP headers or web applications.
- v). When attempt to write more data in a fixed length memory block after buffer overflow occurs, then there is loss of data and is a threat in data processing and accessing.
- vi). In case a denial of service (DoS) attack on the database server, server may be crashed and loss of data that creates a threat.
- vii). Failure in protection and backup data problem is another attack on important data.

These threats are impaired by using the following techniques:

- Data storage, capture and processing is a continues process and that grows the data exponentially in any institute or organizations. So after applying any data security practices or tools solve this.
- Networking infrastructure is increasing day by day and making the system more complex that creates a threat to the vital data, so by applying hybrid cloud or multicloud architecture solve this in easy way
- There is continuous grow in the worldwide regulatory compliance create a complexity in handling the data and databases related with these, so by following some rules and regulations to remove this complexity is a solution.

To evaluate database security around you, followings are the factors considered:

- Physical security should be on your database server or on cloud data center.
- Administrative and network access controls should **be** minimum to avoid any threat.
- Security should be to the end user account and devices related with server and databases to avoid any access and threat.

- Encryption should be **to** protect the data and database. All encryption keys should be handled in accordance with best-practice guidelines.
- To provide database software security, latest version of DBMS should be used.
- To provide security to the web servers and application server security that interacts with the database so that any type of threat can be avoided.
- Copies or backups or saving the images of the database is another important and best factor from any type of loss.
- Auditing should be regularly performed in a well manner by recording all the logins related with operating system and database server.

In your whole network infrastructure and environment there is need to implement layered security controls for securing yours databases, followings are the policies:

- Administrative controls to change, govern installation and to apply proper configuration management for securing the database.
- Preventative controls is another process to set encryption, apply some govern access techniques, masking and tokenization process.
- Detective controls for monitoring database activities and to prevent loss of data by using some prevention tools.

Database security policies should be integrated with and support your overall business goals, such as protection of critical intellectual property and your cyber security policies and cloud security policies. Ensure you have designated responsibility for maintaining and auditing security controls within your organization and that your policies complement those of your cloud provider in shared responsibility agreements. Security controls, security awareness training and education programs, and penetration testing and vulnerability assessment strategies should all be established in support of your formal security policies.

Data protection platforms and tools:

Followings are data protection tools and platforms:

- **Discovery:** Scanning of database tools should be at server level and local level.
- **Data activity monitoring:** Auditing and monitoring process should be to control the threats.
- **Tokenization and Encryption abilities:** Data shoul be transferred and used by using the Token technique and also coding-decoding process should be adopted.
- **Data security optimization and risk analysis:** Tools and techniques should be applied for measuring the data security optimization and to analysis the risk for data loss or threat to the database and database servers.

The Goal of the Database Administration:

It is collection of interrelated files. Data is of major concern in database and the main objective of DBA i.e. database administrator is to secure the database so that unauthorized or unauthenticated person cannot access and alter the database. The various mechanism used to secure the database are given below.

➤ *Security at file placement level:*

A database very often keeps sensible data about the users of the database. The data are placed in the files of the database and should be kept in a secure room. But there is no guarantee that the disk drives can't be stolen by an organized group of intruders. So the sensitive data like credit card numbers, addresses, health status, etc. will be misused. The databases are constructed to use many hard disk drives and table spaces can be placed (and even this is recommended) on different disks. The new modern communication environment often has a very high transmission rate and the link between the different nodes is very fast and reliable. One possibility of keeping data in different places is either to use separated SANs in different physical locations or to divide the data in smaller chunks (if logically possible) or to keep them in different instances (also on different places). The instances can be linked to work together using dblinks. Let us consider stealing of part of the disks. In this case the intruder will be able to use only a part of the data, but it will be impossible or very hard to get all data.

Another way to hide the sensitive data is to make a gateway between the root user data table and the tables with detail data. The idea is to use an intermediate table that hides the primary keys in the main user table introducing a set of different sub keys for different parts of the user data.

So normally a user will be represented with a set of random generated identification keys for different tables of the DB. The next step is to encrypt the intermediate table using cryptography tools.

1. Backup and Restore strategy:

A database backup strategy should be customized for the environment. It should take into consideration the system workload, usage schedule, importance of data, and hardware environment of the database. But there are some guidelines that apply to all databases. Incorporating these guidelines into the strategy will ensure more reliable and more cost effective backups. The main guidelines are:

- ✓ Always maintain a log of your backup schedule
- ✓ Occasionally store backups in a separate location
- ✓ Spread your database across several disks

- ✓ Always consider your existing hardware configuration and system workload
- ✓ Determine the volatility and value of your data
- ✓ Verify the integrity of your database regularly
- ✓ Verify the integrity of your backup media after each backup

a) Always maintain a backup schedule and log:

A well-designed database backup and recovery strategy is useless if the strategy is not publicized and practiced. Operational personnel need to be aware of the importance of the backup methodology, and they need to be able to quickly access backup media in the event database recovery is necessary. Accurate record-keeping and media storage are essential to a proper backup and recovery strategy. The backup schedule should include dates and times, type of backup and tape labels.

Be sure to always *review* the output log files that result from each database backup or recovery operation. Be especially watchful for errors or warnings that might indicate the correctness of the operation was compromised.

b) Occasionally store backups in an offsite location:

Backups are performed to facilitate database restoration, if your primary media fails, once the problem is corrected. There are many possible sources of failure of your primary media, and some of these sources must be considered in your backup strategy. Also, always remember that the data in your database is much more valuable than the physical media it is stored on.

If a site catastrophe occurs and your disks are destroyed, would your backup media still be safe? While it may be convenient, leaving your backup tapes on top of the tape drive is not a good idea. You should store your backup media in a separate location -- whether offsite or just in another area of the building. Although offsite storage is the safest method, access to an offsite archive will take longer than one on-site, and there may be security issues about removing confidential data from your site. A compromise might be to send backups offsite only once a month. Traditionally, you send the previous month's or week's backup offsite, keeping the most recent backup in a safe location on-site.

Magnetic tapes are an inexpensive, reliable, and reusable backup media. The environmental parameters for tapes are often broad and sometimes can lead to abuse. The audio tapes stored in your car can develop some hiss, and the video tapes in your home can drop out some colors, but if your backup tapes are damaged in this way, they are useless. You must follow the manufacturer's instructions on temperature, humidity, magnetic fields, and tape life span. A specially constructed tape library or archive is the safest storage strategy.

c) Always spread your database across several disks:

A backup strategy should provide guidelines on backups and restores of your database. In addition, your overall database strategy should include safeguards for protecting your data from

becoming corrupt and needing a restoration. While you cannot predict hardware failures, you can minimize their effects.

By spreading your database across several disks, you can eliminate having a single point of failure. Even if your database is not large enough to require multiple disks, it is strongly recommended putting the root file on a disk separate from the storage area disks. In addition, balancing busy storage areas between different disks is often a good tuning suggestion, but it also makes sense from a safety point of view.

Whether we backup our database to tapes or to disk, our backup schedule could overlap the routine **OS** backups of our system (especially if the database files share their disks with other users.) Because **OS** backups often run at a high priority, we should consider waiting until they are finished before starting a database backup. There will be less chance of mixing up tapes, less contention for the disks, and, if our system has multiple tape drives, we can allocate all of them to a multi-threaded database backup.

d) Always consider your existing hardware configuration and system workload:

The decision to backup your database either to tape or to disk will be determined primarily by your hardware environment and by personal preference. There are a few suggestions to consider:

- Rather than go to disk and then to tape, it is often faster and more reliable to go straight to tape.
- Multi-threaded backup to tape is usually faster than a single-threaded backup to disk. If you can allocate two tape drives, the backups complete in about half the normal time. Using three drives completes the task in about a third of the time.
- Historically, disk media is more reliable and faster than tapes. However, multi-threaded backup uses redundancy blocks which are more reliable than the single verify pass you get with a disk.
- Tapes are cheaper and easier to store than removable disk media. You can afford to keep more old backup versions around before you recycle the tapes. Try to maintain a minimum of 3 versions of full database backups, since a single past version is not sufficient for a reliable backup strategy.
- If you backup to non-removable disk media, you can probably only keep one past set of backups, which is also insufficient for a reliable backup strategy.

e) Determine the volatility and value of the data:

The volatility of your database refers to how often data changes. If you can determine what data is changing, and how often, you can create a strategy based on backing up the most-frequently changed storage areas more often. For example, the data in an inventory area could be updated several times an hour, an employee field could be updated at the end of every shift, and

a list of suppliers might only change every few months. Determining and mapping this workload will help determine the needs of your site.

The time required to complete incremental backups, and especially storage area incremental backups, is a good indication of changes to a database. Consider that a percentage of data in a database changes daily. Calculate this percentage and perform a full backup when 10% of a database has changed. While it is difficult to calculate this figure, it is much simpler to observe that when incremental backups take longer to complete each day, a greater percentage of a database has been changed since the last full backup. In fact, because of the extra calculations involved, an incremental backup could take longer than a full backup. A large incremental restore can take much longer than a full restore. Through observation, you should be able to determine the break-even point for performing a full backup rather than an incremental backup.

In addition to the frequency of backups, a backup strategy should also consider the differing daily usage of a database. For example, a typical backup schedule might include incremental backups every night and a full backup on Friday night. But if an examination of your usage shows that most changes are made on Wednesdays, then you should consider moving the full backup to Wednesday night. Similarly, if your database is accessed by users in multiple time zones, you might consider starting backups an hour or two later when the activity level has quieted down. Scheduling backups around user activities causes less impact on the users and results in more efficient backups.

f) Verify the integrity of your database regularly:

A backup strategy is useless if the database being backed up is corrupt to begin with. The DBMS contain online utilities to verify the correctness of a database; these tools should be used prior to starting the database backup operation. No database backup operation should be allowed to proceed if the database has not been recently verified as consistent in all respects.

It is recognized that it may not always be feasible to verify the integrity of the database before *every* database backup. However, an attempt should be made to verify the database integrity as often as operationally possible. The verification of the database (DB integrity checking) could be done in the following ways:

- using db verify utility (DBV) provided with Oracle server
- making full export to /dev/null

In situations where it is not possible to always verify the integrity of the entire database before a backup operation, for instance because there is not enough time to perform both a full verify *and* a backup, make an attempt to verify individual data or index areas that are most vital to the database.

g) Verify the integrity of your backup media after each backup:

One of the most common mistakes made is not verifying the integrity of the database backup media; at least, not until the backup is needed to be restored and then it is too late. After a database is backed up, the backup media should be verified. Verification ensures:

- that the backup strategy works as intended
- that the backup media is verifiable
- that the backup media is restorable

There are several methods available to verify the integrity of the database backup media. However, only one method is guaranteed to accurately reflect the integrity of the backup media: the actual restoration or recovery of a database. Anything else is merely verifying the media, not the contents of the media.

2. Data guard (Standby) Databases:

The Data Guard ensures high availability, data protection, and disaster recovery for enterprise data. Data Guard provides a comprehensive set of services that create, maintain, manage, and monitor one or more standby databases to enable production databases to survive disasters and data corruptions. Data Guard maintains these standby databases as transactionally consistent copies of the production database. Then, if the production database becomes unavailable because of a planned or an unplanned outage, Data Guard can switch any standby database to the production role, thus minimizing the downtime associated with the outage. Data Guard can be used with traditional backup, restoration, and cluster techniques to provide a high level of data protection and data availability.

A **Data Guard configuration** consists of one production database and up to nine standby databases. The databases in a Data Guard configuration are connected by Net services and may be dispersed geographically. There are no restrictions on where the databases are located, provided that they can communicate with each other. For example, you can have a standby database on the same system as the production database, along with two standby databases on another system.

➤ **Primary Database:**

A Data Guard configuration contains one production database, also referred to as the primary database, that functions in the primary role. This is the database that is accessed by most of your applications. The primary database can be either a single-instance database or a Real Application Clusters database.

➤ **Standby Databases:**

A standby database is a transactionally consistent copy of the primary database. A standby database is initially created from a backup copy of the primary database. Once created,

Data Guard automatically maintains the standby database by transmitting primary database redo data to the standby system and then applying the redo logs to the standby database.

Similar to a primary database, a standby database can be either a single-instance database or an Real Application Clusters database. A standby database can be either a physical standby database or a logical standby database:

➤ **Physical standby database:**

It provides a physically identical copy of the primary database, with on-disk database structures that are identical to the primary database on a block-for-block basis. The database schema, including indexes, is the same. A physical standby database is kept synchronized with the primary database by recovering the redo data received from the primary database.

➤ **Logical standby database:**

It contains the same logical information as the production database, although the physical organization and structure of the data can be different. It is kept synchronized with the primary database by transforming the data in the redo logs received from the primary database into SQL statements and then executing the SQL statements on the standby database. A logical standby database can be used for other business purposes in addition to disaster recovery requirements. This allows users to access a logical standby database for queries and reporting purposes at any time. Thus, a logical standby database can be used concurrently for data protection and reporting.

➤ **Log Apply Services:**

The redo data transmitted from the primary database is archived on the standby system in the form of archived redo logs. **Log apply services** automatically apply archived redo logs on the standby database to maintain transactional synchronization with the primary database and to allow transactionally consistent read-only access to the data. The main difference between physical and logical standby databases is the manner in which log apply services apply the archived redo logs:

- *For physical standby databases*, Data Guard uses **redo apply** technology, which applies redo data on the standby database using standard recovery techniques of the Oracle database server.

- *For logical standby databases*, Data Guard uses **SQL apply** technology, which first transforms the received redo data into SQL statements and then executes the generated SQL statements on the logical standby database. Log applies services perform the following tasks:

- Automatic application of archived redo logs on the standby database
- Automatic detection of missing redo logs on a standby system and automatic retrieval of missing redo logs from the primary database or another standby database

3. Encryption and Decryption:

Among other security technologies, DBMS protect data in E-business systems through strong, standards-based encryption. For example, Oracle has supported encryption of network data through Oracle Advanced Security since Oracle7. Oracle9i also supports protection of selected data via encryption within the database. Although encryption is not a substitute for effective access control, one can obtain an additional measure of security by electively encrypting sensitive data before it is stored in the database. Examples of such data could include:

- ✓ credit card numbers
- ✓ national identity numbers
- ✓ passwords for applications whose users are not database users *)
- ✓ trade secrets
- ✓ quarter end profits
- ✓ passwords are usually hashed with a one way scheme, but it may be necessary to encrypt plain text passwords

➤ **Key Management:**

The secrecy of encrypted data is dependent on the existence of a secret key shared between the communicating parties. Providing and maintaining such secret keys is known as "key management". Key management, including both generation and secure storage of cryptographic keys, is one of the most important aspects of encryption. If keys are poorly chosen or stored improperly, it is far easier for a malefactor to break the encryption. Rather than using an exhaustive key search attack (that is, cycling through all the possible keys in hopes of finding the correct decryption key, also known as 'brute force attack'), cryptanalysts typically seek weaknesses in the choice of keys, or the way in which keys are stored.

Remember that encryption is something that is easily done wrong and very hard to get right. Once you have chosen a secure algorithm, consider that as the first step in achieving security requirements. The next main problem is key management.

Imagine, when a corporation wants to protect credit card information from rogue DBAs, it makes no sense to trust the DBAs with the encryption keys. In the real world, key management is the hardest part of cryptography.

Key storage is one of the most important, yet difficult aspects of encryption and one of the hardest to manage properly. To recover data encrypted with a secret key, the key must be accessible to the application or user seeking to decrypt data. The key needs to be easy enough to retrieve that users can access encrypted data when they need to without significant performance degradation. The key also needs to be secure enough so that it is not easily recoverable by an unauthorized user trying to access encrypted data he is not supposed to see. There are some possibilities to store the keys:

- ✓ Store the key in the database
- ✓ Store the key in the operating system
- ✓ Have the user manage the key

7.6 SECURITY TECHNIQUES

In the information technology and internet era, number of security threats occurs. So to protect useful information and crucial data, some of the data privacy mechanism and data security techniques should be implemented. So first thing is to identify the confidential data and protect it for any type of leak or threat. You can identify carefully and audit it completely. The phishers, hackers and the pharmers are so smart today to access the data, digital data, financial data and important documents in the databases and access database servers smartly and cleaver way. To secure your vital information and to avoid any malicious or accidental threats, some data security techniques are highlighted and some of the important examples are:

1. Install the Antivirus softwares.
2. Install latest updates to purge data security techniques.
3. Do not use and even avoid to use Spyware and Adware
4. Always select and use unusual and tricky password.
5. Create a strong and powerful password.
6. Try to avoid Endanges of Emails and messages.
7. Always install Firewall and time to time verify it.
8. Always lock your important documents and files.

7.7 TWO PHASE LOCKING TECHNIQUES

Two-Phase Locking (2PL) is a concurrency control method which divides the execution phase of a transaction into three parts. It ensures conflict serializable schedules. If read and write operations introduce the first unlock operation in the transaction, then it is said to be Two-Phase Locking Protocol. This protocol can be divided into two phases:

- 1. In Growing Phase**, a transaction obtains locks, but may not release any lock.
- 2. In Shrinking Phase**, a transaction may release locks, but may not obtain any lock.

Two-Phase Locking does not ensure freedom from deadlocks.

Types of Two – Phase Locking Protocol:

Following are the types of two – phase locking protocol:

1. Strict Two – Phase Locking Protocol
2. Rigorous Two – Phase Locking Protocol
3. Conservative Two – Phase Locking Protocol

1. Strict Two-Phase Locking Protocol: Strict Two-Phase Locking Protocol avoids cascaded rollbacks. This protocol not only requires two-phase locking but also all exclusive-locks should be held until the transaction commits or aborts. It is not deadlock free. It ensures that if data is

being modified by one transaction, then other transaction cannot read it until first transaction commits. Most of the database systems implement rigorous two – phase locking protocol.

2. Rigorous Two-Phase Locking: Rigorous Two – Phase Locking Protocol avoids cascading rollbacks. This protocol requires that all the share and exclusive locks to be held until the transaction commits.

3. Conservative Two-Phase Locking Protocol: Conservative Two – Phase Locking Protocol is also called as Static Two – Phase Locking Protocol. This protocol is almost free from deadlocks as all required items are listed in advanced. It requires locking of all data items to access before the transaction starts.

7.8 SUMMARY

DBMS has different view from architectural point. A three-level architecture suggested by American National Standards Institute (ANSI) /Standards Planning and Requirements Committee (SPARC) in 1977. It is necessary to view data at different levels of abstraction. The three levels of the architecture are three different views of the data as *External, Conceptual, and Internal* view. Data Independence means data structure of the data is completely independent from the storage and accessing of the data. It is completely independent of any storage location, data types and accessing techniques. There are two levels of data independence as Physical Data Independence and Logical Data Independence. In the SQL language, a view is a representation of one or more tables. A view can be used to hide the complexity of relationships between tables or to provide security for sensitive data in tables. In general View is a part of table it describes only certain portion of a table. View are used to apply security to database by hiding certain portion of table. In SQL Views are kind of virtual tables. SQL joins are used to join two or more tables that is with the help of SQL joins we can find or retrieve an information from two or more than two tables. To delete or to remove a view of a relation or an object view of a table is the dropping of the view. This can be done easily by using the DROP VIEW clause or command or statement. To preserve the database integrity, confidentiality and availability, some tools, measures and controls are designed. The finally database is secure by doing so and the above said process is called Data Security. In the information technology and internet era, number of security threats occurs. So to protect useful information and crucial data, some of the data privacy mechanism and data security techniques should be implemented. So first thing is to identify the confidential data and protect it for any type of leak or threat. Two-Phase Locking (2PL) is a concurrency control method which divides the execution phase of a transaction into three parts. It ensures conflict serializable schedules. If read and write operations introduce the first unlock operation in the transaction, then it is said to be Two-Phase Locking Protocol.

7.9 PRACTICE EXERCISE

Q1. What do you mean by Views in RDBMS?

Q2. Explain all the three level in RDBMS?

- Q3. What do you mean by Data Independence?
- Q4. What are the various types of Data Independence?
- Q5. Differentiate between Physical Data Independence and Logical Data Independence?
- Q6. What do you mean by Join Views? What are the various Join Views statements?
- Q7. How can you drop a view?
- Q8. How can you drop a view?
- Q9. What do you know about database security?
- Q10. What are the various Database Security techniques?
- Q11. What do you mean by Two-phase Locking?
- Q12. What are the various types of two-phase locking schemes?

B.Sc.(DATA SCIENCE)

SEMESTER-III

DATABASE MANAGEMENT SYSTEM

UNIT VIII: DATA CONTROL OPERATIONS

STRUCTURE

8.0 Objectives

8.1 Data Control

8.2 Grant Command

8.3 Revoke Command

8.4 Commit and Rollback

8.5 Concurrency Control Techniques

8.6 Recovery Control Techniques

8.7 Summary

8.0 OBJECTIVES

To understand data controlling and management of data. Different commands associates with Data Control Languages.

8.1 DATA CONTROL

Management of information policies for handling the organization's information is the data control. Major Job of data control is to observe and report about working of data processing. It also handles all the managing issues and it is similar to data quality. Various types of operations or functions in data controls are inspection of data, notification, data validation, tracking, documentation and finally reporting.

Followings are the various drivers used for Data Control:

- **Compliance of data**
- **Auditability**
- **Transparency in data processing**
- **Reportability about validated data**
- **Agility**
- **Risk awareness if any occurs**

Key objectives in Data Control are as:

- To identify all the risks as soon as possible in data processing.
- To manage and implement all the policies and rules and regulations.
- It provides auditing of framework.
- To use the critical and complex data elements and also ensure the proper protection of data.
- Tracking and management of data quality for any service level agreements and services.
- All the data issues are well managed for the identification, remediation and reporting.

Various benefits using Data Control are:

- Controlling and monitoring of data between storage devices and various applications becomes easy.
- Before occurrences of any cause and complexity in data processing, data flaws are identified timely.
- It identify root causes of any issues and remediated timely.
- Identification of any loss of sensitive data (accidental or malicious) quickly.
- Backup of data before any operation can be done timely.

There are number of query languages operated on database in a RDBMS (Relational Database Management System)are SQL (Structured Query Language), Data Definition Language (DDL), Data Manipulation Language (DML), Data Query Language (DQL) and Data Control Language (DCL). Here we are discussing Data Control Language that is used for accessing the data stored in the Database. It is very useful for any updating or change in case of handling multiple databases to multiple users.

Need of Data Control Language (Why Control Language required):

Followings are some needs of data control language:

- Users in the database have the privileges to control it by using DCL.
- It control any unauthorized access to the database during updation of data.
- Only some of the DML and DDL commands for handling the database are controlled by DCL.
- It set the responsibilities to the DBA for use and controlling the data.
- DCL provides security to the database.

Working of Data Control Language (How Data Control Language Works)?

Here we see that how the statements work using the DCL:

- First of all it a big privilege to the users for accessing of various objects related with the database. Object and System are the two privileges for basic operations.
- At next any type of accessing or permission takes place during query processing to create tables, views, sessions etc.
- The system honors and give permission for performing CREATE, ALTER and DROP commands that is useful for operating the SELECT, UPDATE, INSERT, DELETE and EXECUTE commands on the data in handling various databases and database objects.
- In case of multiple users using the particular database environment, there is a difficulty in handling the data that solve by using the grant or revoke operations.
- Grant of roles to the users for using the data in the database is a privilege.
- Similarly using the revoke command data get revoked automatically for the user.

Advantages or Pros of DCL (Data Control Language):

Followings are the advantages of DCL:

- It provides the security to the data in database.
- DCL grant or remove number of permissions or the privileges to the various users for the database.

- Owner of the database or DBA grant or revoke the privileges for handling and managing any issues related with the database.

8.2 GRANT COMMAND

DCL commands are used to enforce database security in a multiple user database environment. Two types of DCL commands are GRANT and REVOKE. Only Database Administrator's or owner's of the database object can provide/remove privileges on a database object.

SQL GRANT Command: SQL GRANT is a command used to provide access or privileges on the database objects to the users. The Syntax for the GRANT command is:

GRANT privilege_name

ON object_name

TO {user_name |PUBLIC |role_name}

[WITH GRANT OPTION];

Here *privilege_name* is the privilege or access right granted to the user. This can be done

by SELECT, ALL and EXECUTE command. *object_name* is the database name to deals with the database object. The various database objects are TABLE, VIEW, STORED PROC and SEQUENCE. Here *user_name* is the name of the user to whom an access right is being granted. **PUBLIC** word is used to grant access rights to all users. **ROLES** are a set of privileges grouped together. **WITH GRANT OPTION** - allows a user to grant access rights to other users.

For Example:

```
GRANT SELECT ON employee TO user1;
```

This command grants a SELECT permission on employee table to user1. You should use the WITH GRANT option carefully because for example if you GRANT SELECT privilege on employee table to user1 using the WITH GRANT option, then user1 can GRANT SELECT privilege on employee table to another user, such as user2 etc. Later, if you REVOKE the SELECT privilege on employee from user1, still user2 will have SELECT privilege on employee table.

8.3 REVOKE COMMAND

The REVOKE command removes user access rights or privileges to the database objects. The Syntax for the REVOKE command is:

REVOKE privilege_name

ON object_name

FROM {user_name |PUBLIC |role_name }

For Example:

```
REVOKE SELECT ON employee FROM user1;
```

This command will REVOKE a SELECT privilege on employee table from user1. When you REVOKE SELECT privilege on a table from a user, the user will not be able to SELECT data from that table anymore. However, if the user has received SELECT privileges on that table from more than one users, he/she can SELECT from that table until everyone who granted the permission revokes it. You cannot REVOKE privileges if they were not initially granted by you.

Privileges and Roles:

Privileges: Privileges defines the access rights provided to a user on a database object. There are two types of privileges.

- 1) System privileges** - This allows the user to CREATE, ALTER, or DROP database objects.
- 2) Object privileges** - This allows the user to EXECUTE, SELECT, INSERT, UPDATE, or DELETE data from database objects to which the privileges apply.

Few CREATE system privileges are listed below:

System Privileges	Description
CREATE object	allows users to create the specified object in their own schema.
CREATE ANY object	allows users to create the specified object in any schema.

The above rules also apply for ALTER and DROP system privileges.

Few of the object privileges are listed below:

Object Privileges	Description
INSERT	allows users to insert rows into a table.
SELECT	allows users to select data from a database object.
UPDATE	allows user to update data in a table.
EXECUTE	allows user to execute a stored procedure or a function.

Roles: Roles are a collection of privileges or access rights. When there are many users in a database it becomes difficult to grant or revoke privileges to users. Therefore, if you define roles, you can grant or revoke privileges to users, thereby automatically granting or revoking privileges. You can either create Roles or use the system roles pre-defined by oracle. Some of the privileges granted to the system roles are as given below:

System Role	Privileges Granted to the Role
CONNECT	CREATE TABLE, CREATE VIEW, CREATE SYNONYM, CREATE SEQUENCE, CREATE SESSION etc.
RESOURCE	CREATE PROCEDURE, CREATE SEQUENCE, CREATE TABLE, CREATE TRIGGER etc. The primary usage of the RESOURCE role is to restrict access to database objects.
DBA	ALL SYSTEM PRIVILEGES

Creating Roles:

The Syntax to create a role is:

CREATE ROLE role-name
[IDENTIFIED BY password];

For Example: To create a role called "developer" with password as "pwd",the code will be as follows:

```
CREATE ROLE student-role-name
[IDENTIFIED BY pwd];
```

It's easier to GRANT or REVOKE privileges to the users through a role rather than assigning a privilege directly to every user. If a role is identified by a password, then, when

you GRANT or REVOKE privileges to the role, you definitely have to identify it with the password. We can GRANT or REVOKE privilege to a role as below.

For example: To grant CREATE TABLE privilege to a user by creating a testing role:

First, create a student Role:

```
CREATE ROLE Student
```

Second, grant a CREATE TABLE privilege to the ROLE testing. You can add more privileges to the ROLE.

```
GRANT CREATE TABLE TO testing;
```

Third, grant the role to a user.

```
GRANT testing TO user1;
```

To revoke a CREATE TABLE privilege from testing ROLE, you can write:

```
REVOKE CREATE TABLE FROM testing;
```

The Syntax to drop a role from the database is as below:

```
DROP ROLE role_name;
```

For example: To drop a role called developer, you can write:

```
DROP ROLE testing
```

Difference between GRANT and REVOKE Commands:

GRANT	REVOKE
It grants permissions to the user on the data using the database objects	It removes permissions if any granted to the users on database objects.
It assigns access rights to users.	It revokes the user access rights of users.
For each user you need to specify the permissions.	If access for one user is removed; all the particular permissions provided by that users to others will be removed.
When the access is decentralized granting permissions will be easy.	If decentralized access removing the granted permissions is difficult.

8.4 COMMIT AND ROLLBACK

The process of placing into effect in the database the updates made by a transaction is called *commit*. The process of invalidating the updates made by a transaction is called *rollback*. When commit and rollback occur is determined as follows:

- At the times defined by an SQL
- At the times that are set automatically by HiRDB

Each of these types of commit and rollback timing is explained as follows.

(1) Commit timing

(2) Rollback timing

(3) Commitment control on a HiRDB/Parallel Server

(1) Commit timing:

This section explains commit timing.

Timing defined by an SQL: The COMMIT control SQL statement can be specified to commit a transaction whenever a COMMIT statement is executed.

Commit set automatically by HiRDB: Commit is performed automatically by HiRDB when a definition SQL or PURGE TABLE statement is executed. Commit is performed automatically by HiRDB when a UAP terminates.

(2) Rollback timing:

This section explains rollback timing:

Timing defined by an SQL: The ROLLBACK control SQL statement can be specified to roll back the process to the previous commit point whenever a ROLLBACK statement is executed.

Rollback set automatically by HiRDB: If a process cannot continue during SQL execution, HiRDB will roll back the process implicitly to the previous commit point. If a UAP terminates abnormally, HiRDB will roll back the process to the previous commit point.

(3) Commitment control on a HiRDB/Parallel Server:

A HiRDB/Parallel Server provides the following two methods of commitment control:

(a) One-phase commit:

With one-phase commit, only commit processing is performed, rather than both prepare processing and commit processing (which is the case with two-phase commitment control). This means that the number of communication transactions for synchronization point processing between the front-end server and the back-end server (dictionary server) is the number of branches $\times 2$ (whereas, with two-phase commit, it is the number of branches $\times 4$), which improves transaction processing performance. To use one-phase commit, specify ONEPHASE (default) in the `pd_trn_commit_optimize` operand. One-phase commit is

used only when one branch within a single transaction is being updated. Otherwise, two-phase commit must be used.

(b) Two-phase commit:

Two-phase commit performs synchronization point processing of the transaction by separating commitment control into two phases, prepare processing and commit processing. The number of communication transactions for synchronization point processing between the front-end server and the back-end server (dictionary server) is the number of branches \times 4. Figure 6-23 shows the processing for two-phase commit.

1. COMMIT:

COMMIT in SQL is a transaction control language which is used to permanently save the changes done in the transaction in tables/databases. The database cannot regain its previous state after the execution of it. Consider the following STAFF table with records:

STAFF

Id	Name	Age	Allowance	Salary
1	Rahul	26	400	4000
2	Khaitan	46	900	9000
3	Munjhal	36	400	4500
4	Ram	28	800	8000
5	Manav	24	400	6500
6	Kaira	21	700	7800

Example:

```
sql>  
SELECT *  
FROM Staff  
WHERE Allowance = 400;  
sql> COMMIT;
```

Output:

Id	Name	Age	Allowance	Salary
1	Rahul	26	4000	400
3	Munjhal	36	4500	400
5	Manav	24	6500	400

So, the SELECT statement produced the output consisting of three rows.

2. ROLLBACK:

ROLLBACK in SQL is a transactional control language which is used to undo the transactions that have not been saved in database. The command is only be used to undo changes since the last COMMIT. Consider the following STAFF table with records:

STAFF

Id	Name	Age	Allowance	Salary
1	Rahul	26	400	4000
2	Khaitan	46	900	9000
3	Munjhal	36	400	4500
4	Ram	28	800	8000
5	Manav	24	400	6500
6	Kaira	21	700	7800

Example:

sql>

SELECT *

FROM EMPLOYEES

WHERE ALLOWANCE = 400;

sql> ROLLBACK;

Output:

Id	Name	Age	Allowance	Salary
1	Rahul	26	400	4000
2	Khaitan	46	900	9000
3	Munjhal	36	400	4500
4	Ram	28	800	8000
5	Manav	24	400	6500
6	Kaira	21	700	7800

So, the SELECT statement produced the same output with ROLLBACK command.

Difference between COMMIT and ROLLBACK:

COMMIT	ROLLBACK
COMMIT permanently saves the changes made by current transaction.	ROLLBACK undo the changes made by current transaction.
Transaction can not undo changes after COMMIT execution.	Transaction reaches its previous state after ROLLBACK.

8.5 CONCURRENCY CONTROL TECHNIQUES

Concurrency Control in Database Management System is a procedure of managing simultaneous operations without conflicting with each other. It ensures that Database transactions are performed concurrently and accurately to produce correct results without violating data integrity of the respective Database.

Concurrent access is quite easy if all users are just reading data. There is no way they can interfere with one another. Though for any practical Database, it would have a mix of READ and WRITE operations and hence the concurrency is a challenge.

DBMS Concurrency Control is used to address such conflicts, which mostly occur with a multi-user system. Therefore, Concurrency Control is the most important element for proper functioning of a Database Management System where two or more database transactions are executed simultaneously, which require access to the same data.

Potential problems of Concurrency:

Here, are some issues which you will likely to face while using the DBMS Concurrency Control method:

- **Lost Updates** occur when multiple transactions select the same row and update the row based on the value selected
- Uncommitted dependency issues occur when the second transaction selects a row which is updated by another transaction (**dirty read**)
- **Non-Repeatable Read** occurs when a second transaction is trying to access the same row several times and reads different data each time.
- **Incorrect Summary issue** occurs when one transaction takes summary over the value of all the instances of a repeated data-item, and second transaction update few instances of that specific data-item. In that situation, the resulting summary does not reflect a correct result.

Why use Concurrency method?

Reasons for using Concurrency control method is DBMS:

- To apply Isolation through mutual exclusion between conflicting transactions
- To resolve read-write and write-write conflict issues
- To preserve database consistency through constantly preserving execution obstructions
- The system needs to control the interaction among the concurrent transactions. This control is achieved using concurrent-control schemes.
- Concurrency control helps to ensure serializability

Example: Assume that two people who go to electronic kiosks at the same time to buy a movie ticket for the same movie and the same show time.

However, there is only one seat left in for the movie show in that particular theatre. Without concurrency control in DBMS, it is possible that both moviegoers will end up purchasing a ticket. However, concurrency control method does not allow this to happen. Both moviegoers

can still access information written in the movie seating database. But concurrency control only provides a ticket to the buyer who has completed the transaction process first.

Concurrency Control Protocols:

Different concurrency control protocols offer different benefits between the amount of concurrency they allow and the amount of overhead that they impose. Following are the Concurrency Control techniques in DBMS:

- Lock-Based Protocols
- Two Phase Locking Protocol
- Timestamp-Based Protocols
- Validation-Based Protocols

i). Lock-based Protocols:

Lock Based Protocols in DBMS is a mechanism in which a transaction cannot Read or Write the data until it acquires an appropriate lock. Lock based protocols help to eliminate the concurrency problem in DBMS for simultaneous transactions by locking or isolating a particular transaction to a single user.

A lock is a data variable which is associated with a data item. This lock signifies that operations that can be performed on the data item. Locks in DBMS help synchronize access to the database items by concurrent transactions. All lock requests are made to the concurrency-control manager. Transactions proceed only once the lock request is granted.

Binary Locks: A Binary lock on a data item can either locked or unlocked states.

Shared/exclusive: This type of locking mechanism separates the locks in DBMS based on their uses. If a lock is acquired on a data item to perform a write operation, it is called an exclusive lock.

1. Shared Lock (S):

A shared lock is also called a Read-only lock. With the shared lock, the data item can be shared between transactions. This is because you will never have permission to update data on the data item. For example, consider a case where two transactions are reading the account balance of a person. The database will let them read by placing a shared lock. However, if another transaction wants to update that account's balance, shared lock prevent it until the reading process is over.

2. Exclusive Lock (X):

With the Exclusive Lock, a data item can be read as well as written. This is exclusive and can't be held concurrently on the same data item. X-lock is requested using lock-x instruction. Transactions may unlock the data item after finishing the 'write' operation. For example, when a transaction needs to update the account balance of a person. You can allows this transaction by

placing X lock on it. Therefore, when the second transaction wants to read or write, exclusive lock prevent this operation.

3. Simplistic Lock Protocol:

This type of lock-based protocols allows transactions to obtain a lock on every object before beginning operation. Transactions may unlock the data item after finishing the 'write' operation.

4. Pre-claiming Locking:

Pre-claiming lock protocol helps to evaluate operations and create a list of required data items which are needed to initiate an execution process. In the situation when all locks are granted, the transaction executes. After that, all locks release when all of its operations are over.

Starvation:

Starvation is the situation when a transaction needs to wait for an indefinite period to acquire a lock. Following are the reasons for Starvation:

- When waiting scheme for locked items is not properly managed
- In the case of resource leak
- The same transaction is selected as a victim repeatedly

Deadlock:

Deadlock refers to a specific situation where two or more processes are waiting for each other to release a resource or more than two processes are waiting for the resource in a circular chain.

ii). Two Phase Locking Protocol:

Two Phase Locking Protocol also known as 2PL protocol is a method of concurrency control in DBMS that ensures serializability by applying a lock to the transaction data which blocks other transactions to access the same data simultaneously. Two Phase Locking protocol helps to eliminate the concurrency problem in DBMS. This locking protocol divides the execution phase of a transaction into three different parts.

- In the first phase, when the transaction begins to execute, it requires permission for the locks it needs.
- The second part is where the transaction obtains all the locks. When a transaction releases its first lock, the third phase starts.
- In this third phase, the transaction cannot demand any new locks. Instead, it only releases the acquired locks.

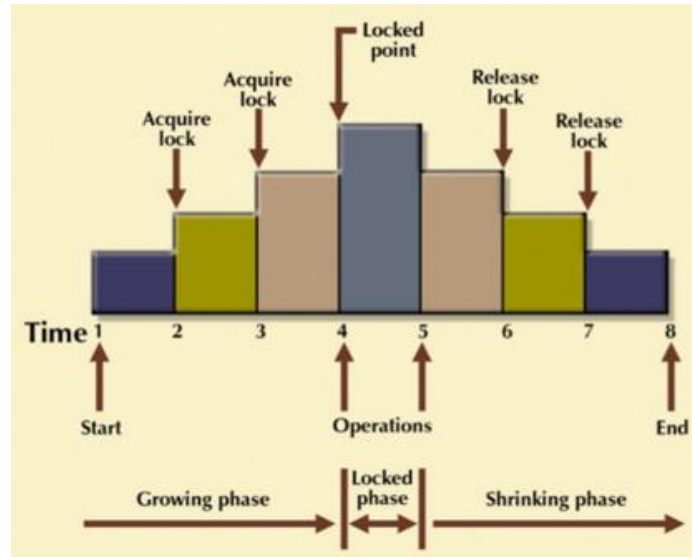


Figure 8.1

The Two-Phase Locking protocol allows each transaction to make a lock or unlock request in two steps:

- **Growing Phase:** In this phase transaction may obtain locks but may not release any locks.
- **Shrinking Phase:** In this phase, a transaction may release locks but not obtain any new lock

It is true that the 2PL protocol offers serializability. However, it does not ensure that deadlocks do not happen. In the above-given diagram, you can see that local and global deadlock detectors are searching for deadlocks and solve them with resuming transactions to their initial states.

Strict Two-Phase Locking Method:

Strict-Two phase locking system is almost similar to 2PL. The only difference is that Strict-2PL never releases a lock after using it. It holds all the locks until the commit point and releases all the locks at one go when the process is over.

Centralized 2PL:

In Centralized 2 PL, a single site is responsible for lock management process. It has only one lock manager for the entire DBMS.

Primary copy 2PL:

Primary copy 2PL mechanism, many lock managers are distributed to different sites. After that, a particular lock manager is responsible for managing the lock for a set of data items. When the primary copy has been updated, the change is propagated to the slaves.

Distributed 2PL:

In this kind of two-phase locking mechanism, Lock managers are distributed to all sites. They are responsible for managing locks for data at that site. If no data is replicated, it is equivalent to primary copy 2PL. Communication costs of Distributed 2PL are quite higher than primary copy 2PL

iii). Timestamp-based Protocols:

Timestamp based Protocol in DBMS is an algorithm which uses the System Time or Logical Counter as a timestamp to serialize the execution of concurrent transactions. The Timestamp-based protocol ensures that every conflicting read and write operations are executed in a timestamp order. The older transaction is always given priority in this method. It uses system time to determine the time stamp of the transaction. This is the most commonly used concurrency protocol.

Lock-based protocols help you to manage the order between the conflicting transactions when they will execute. Timestamp-based protocols manage conflicts as soon as an operation is created. Example: Suppose there are three transactions T1, T2, and T3.

T1 has entered the system at time 0010

T2 has entered the system at 0020

T3 has entered the system at 0030

Priority will be given to transaction T1, then transaction T2 and lastly Transaction T3.

Advantages:

- Schedules are serializable just like 2PL protocols
- No waiting for the transaction, which eliminates the possibility of deadlocks!

Disadvantages:

Starvation is possible if the same transaction is restarted and continually aborted

iv). Validation Based Protocol:

Validation based Protocol in DBMS also known as Optimistic Concurrency Control Technique is a method to avoid concurrency in transactions. In this protocol, the local copies of the transaction data are updated rather than the data itself, which results in less interference while execution of the transaction. The Validation based Protocol is performed in the following three phases:

1. Read Phase
2. Validation Phase
3. Write Phase

Read Phase:

In the Read Phase, the data values from the database can be read by a transaction but the write operation or updates are only applied to the local data copies, not the actual database.

Validation Phase:

In Validation Phase, the data is checked to ensure that there is no violation of serializability while applying the transaction updates to the database.

Write Phase:

In the Write Phase, the updates are applied to the database if the validation is successful, else; the updates are not applied, and the transaction is rolled back.

Characteristics of Good Concurrency Protocol:

An ideal concurrency control DBMS mechanism has the following objectives:

- Must be resilient to site and communication failures.
- It allows the parallel execution of transactions to achieve maximum concurrency.
- Its storage mechanisms and computational methods should be modest to minimize overhead.
- It must enforce some constraints on the structure of atomic actions of transactions.

8.6 RECOVERY CONTROL TECHNIQUES

The Databases are prone to failures due to inconsistency, network failure, errors or any kind of accidental damage. So, database recovery techniques are highly important to bring a database back into a working state after a failure. There can be any case in database system like any computer system when database failure happens. So data stored in database should be available all the time whenever it is needed. So database recovery means recovering the data when it get deleted, hacked or damaged accidentally. Atomicity is must whether is transaction is over or not it should reflect in the database permanently or it should not effect the database at all. So database recovery and database recovery techniques are must in DBMS.

Database systems, like any other computer system, are subject to failures but the data stored in it must be available as and when required. When a database fails it must possess the facilities for fast recovery. It must also have atomicity i.e. either transactions are completed successfully and committed (the effect is recorded permanently in the database) or the transaction should have no effect on the database.

There are both automatic and non-automatic ways for both, backing up of data and recovery from any failure situations. The techniques used to recover the lost data due to system crash, transaction errors, viruses, catastrophic failure, incorrect commands execution etc. are database recovery techniques. So to prevent data loss recovery techniques based on deferred update and immediate update or backing up data can be used.

There are four different **recovery techniques** are available in the Database.

1. *Mirroring*
2. *Recovery using Backups*
3. *Recovery using Transaction Logs*
4. *Shadow Paging*

1. **Mirroring:**

Two complete copies of the database maintains on-line on different stable storage devices. This method mostly uses in environments that require non-stop, fault-tolerant operations.

2. **Recovery using Backups:**

Backups are useful if there has been extensive damage to database. Backups are mainly two types :

i). Immediate Backup:

Immediate Backup are kept in a floppy disk, hard disk or magnetic tapes. These come in handy when a technical fault occurs in the primary database such as system failure, disk crashes, network failure. Damage due to virus attacks repair using the immediate backup.

ii). Archival Backup:

Archival Backups are kept in mass storage devices such as magnetic tape, CD-ROMs, Internet Servers etc. They are very useful for recovering data after a disaster such as fire, earthquake, flood etc. Archival Backup should be kept at a different site other than where the system is functioning. Archival Backup at a separate place remains safe from thefts and intentional destruction by user staff.

3. **Recovery using Transaction Logs:**

In Recovery using Transaction Logs, some following steps are :

Step1: The log searches for all the transaction that have recorded a

[**start transaction**, ‘ ‘] entry, but haven’t recorded a corresponding [**commit**, ‘ ‘] entry.

Step2: These transactions are rolling back.

Step3: Transactions which have recorded a [**commit**, ‘ ‘] entry in the log, it must have recorded the changes, they did to the database in the log. These change will follow to undo their effects on the database.

4. **Shadow Paging:**

These system can use for data recovery instead of using transaction logs. In the Shadow Paging, a database is divided into several fixed-sized disk pages, say n, thereafter a current directory creates. It having n entries with each entry pointing to a disk page in the database. the current directory transfer to the main memory. When a transaction begins executing, the current directory copies into a shadow directory. Then, the shadow directory saves on the disk. The

transaction will be using the current directory. During the transaction execution, all the modifications are made on the current directory and the shadow directory is never modified.

Checkpoint-Recovery is a common technique for imbuing a program or system with fault tolerant qualities, and grew from the ideas **used** in systems which employ transaction processing [lyu95]. It allows systems to **recover** after some fault interrupts the system, and causes the task to fail, or be aborted in some way.

Recovery and Atomicity:

When a system crashes, it may have several transactions being executed and various files opened for them to modify the data items. Transactions are made of various operations, which are atomic in nature. But according to ACID properties of DBMS, atomicity of transactions as a whole must be maintained, that is, either all the operations are executed or none. When a DBMS recovers from a crash, it should maintain the following:

- It should check the states of all the transactions, which were being executed.
- A transaction may be in the middle of some operation; the DBMS must ensure the atomicity of the transaction in this case.
- It should check whether the transaction can be completed now or it needs to be rolled back.
- No transactions would be allowed to leave the DBMS in an inconsistent state.

There are two types of techniques, which can help a DBMS in recovering as well as maintaining the atomicity of a transaction:

- Maintaining the logs of each transaction, and writing them onto some stable storage before actually modifying the database.
- Maintaining shadow paging, where the changes are done on a volatile memory, and later, the actual database is updated.

Log-based Recovery:

Log is a sequence of records, which maintains the records of actions performed by a transaction. It is important that the logs are written prior to the actual modification and stored on a stable storage media, which is failsafe. Log-based recovery works as follows –

- The log file is kept on a stable storage media.
- When a transaction enters the system and starts execution, it writes a log about it.

$\langle T_n, \text{Start} \rangle$

- When the transaction modifies an item X, it write logs as follows –

$\langle T_n, X, V_1, V_2 \rangle$

It reads T_n has changed the value of X, from V_1 to V_2 .

- When the transaction finishes, it logs –

$\langle T_n, \text{commit} \rangle$

The database can be modified using two approaches –

- **Deferred database modification** – All logs are written on to the stable storage and the database is updated when a transaction commits.
- **Immediate database modification** – Each log follows an actual database modification. That is, the database is modified immediately after every operation.

Recovery with Concurrent Transactions:

When more than one transaction are being executed in parallel, the logs are interleaved. At the time of recovery, it would become hard for the recovery system to backtrack all logs, and then start recovering. To ease this situation, most modern DBMS use the concept of 'checkpoints'.

Checkpoint:

Keeping and maintaining logs in real time and in real environment may fill out all the memory space available in the system. As time passes, the log file may grow too big to be handled at all. Checkpoint is a mechanism where all the previous logs are removed from the system and stored permanently in a storage disk. Checkpoint declares a point before which the DBMS was in consistent state, and all the transactions were committed.

Recovery:

When a system with concurrent transactions crashes and recovers, it behaves in the following manner:

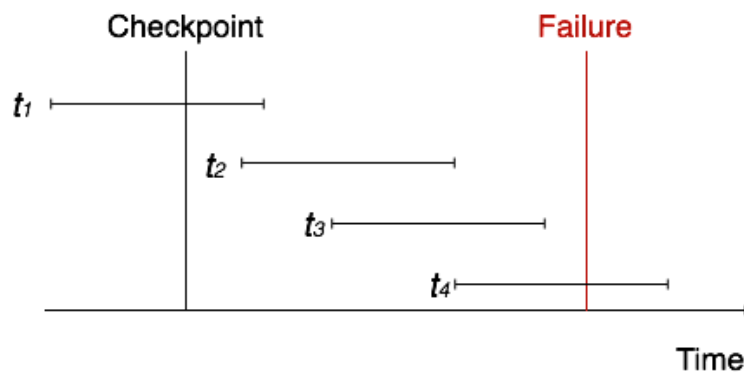


Figure 8.2

- The recovery system reads the logs backwards from the end to the last checkpoint.
- It maintains two lists, an undo-list and a redo-list.
- If the recovery system sees a log with $\langle T_n, \text{Start} \rangle$ and $\langle T_n, \text{Commit} \rangle$ or just $\langle T_n, \text{Commit} \rangle$, it puts the transaction in the redo-list.

- If the recovery system sees a log with $\langle T_n, \text{Start} \rangle$ but no commit or abort log found, it puts the transaction in undo-list.

All the transactions in the undo-list are then undone and their logs are removed. All the transactions in the redo-list and their previous logs are removed and then redone before saving their logs.

8.7 SUMMARY

Management of information policies for handling the organization's information is the data control. There are number of query languages operated on database in a RDBMS (Relational Database Management System) are SQL (Structured Query Language), Data Definition Language (DDL), Data Manipulation Language (DML), Data Query Language (DQL) and Data Control Language (DCL). Here we are discussing Data Control Language that is used for accessing the data stored in the Database. It is very useful for any updating or change in case of handling multiple databases to multiple users. DCL commands are used to enforce database security in a multiple user database environment. Two types of DCL commands are GRANT and REVOKE. SQL GRANT is a command used to provide access or privileges on the database objects to the users. The REVOKE command removes user access rights or privileges to the database objects. The process of placing into effect in the database the updates made by a transaction is called *commit*. The process of invalidating the updates made by a transaction is called *rollback*. Concurrency Control in Database Management System is a procedure of managing simultaneous operations without conflicting with each other. It ensures that Database transactions are performed concurrently and accurately to produce correct results without violating data integrity of the respective Database. Concurrency Control techniques in DBMS are Lock-Based Protocols, Two Phase Locking Protocol, Timestamp-Based Protocols and Validation-Based Protocols. The Databases are prone to failures due to inconsistency, network failure, errors or any kind of accidental damage. So, database recovery techniques are highly important to bring a database back into a working state after a failure. So database recovery and database recovery techniques are must in DBMS. There are four different recovery techniques are available in the Database are *Mirroring*, *Recovery using Backups*, *Recovery using Transaction Logs*, *Shadow Paging*.

Question for Practice

- Q1. What do you know about Data Control?
- Q2. What are the various Data Control Statements?
- Q3. What do you know about Grant command? How it works?
- Q4. What do you know about Revoke command? How it works?
- Q5. What do you know about Commit command? How it works?

Q6. What do you know about Rollback command? How it works?

Q7. Differentiate between Grant and Revoke Command?

Q8. Differentiate between Commit and Rollback command?

Q9. What do you mean by concurrency control?

Q10. What are the various Concurrency Control techniques?

Q11. What are the various recovery control techniques?



**The Motto of Our University
(SEWA)**

SKILL ENHANCEMENT

EMPLOYABILITY

WISDOM

ACCESSIBILITY

SELF-INSTRUCTIONAL STUDY MATERIAL FOR JGND PSOU

ALL COPYRIGHTS WITH JGND PSOU, PATIALA

**JAGAT GURU NANAK DEV
PUNJAB STATE OPEN UNIVERSITY, PATIALA**

(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

B.Sc.(Data Science)

Semester III

BSDB32302T

Data Mining and Visualization

Head Quarter: C/28, The Lower Mall, Patiala-147001

Website: www.psou.ac.in

The Study Material has been prepared exclusively under the guidance of Jagat Guru Nanak Dev Punjab State Open University, Patiala, as per the syllabi prepared by Committee of Experts and approved by the Academic Council.

The University reserves all the copyrights of the study material. No part of this publication may be reproduced or transmitted in any form.

COURSE COORDINATOR AND EDITOR:

Dr. Amitoj Singh

Associate Professor

School of Sciences and Emerging Technologies

Jagat Guru Nanak Dev Punjab State Open University

LIST OF CONSULTANTS/ CONTRIBUTORS

Sr. No.	Name
1	Dr. Jyotsna Sengupta
2	Dr. Chetan Dudhagara



JAGAT GURU NANAK DEV PUNJAB STATE OPEN UNIVERSITY, PATIALA
(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

PREFACE

Jagat Guru Nanak Dev Punjab State Open University, Patiala was established in December 2019 by Act 19 of the Legislature of State of Punjab. It is the first and only Open University of the State, entrusted with the responsibility of making higher education accessible to all, especially to those sections of society who do not have the means, time or opportunity to pursue regular education.

In keeping with the nature of an Open University, this University provides a flexible education system to suit every need. The time given to complete a programme is double the duration of a regular mode programme. Well-designed study material has been prepared in consultation with experts in their respective fields.

The University offers programmes which have been designed to provide relevant, skill-based and employability-enhancing education. The study material provided in this booklet is self-instructional, with self-assessment exercises, and recommendations for further readings. The syllabus has been divided in sections, and provided as units for simplification.

The University has a network of 10 Learner Support Centres/Study Centres, to enable students to make use of reading facilities, and for curriculum-based counselling and practicals. We, at the University, welcome you to be a part of this institution of knowledge.

Prof. Anita Gill
Dean Academic Affairs



B.Sc. (Data Science)
Discipline Specific Course (DSC)
Semester III
BSDB32302T: Data Mining and Visualization

Total Marks: 100

External Marks: 70

Internal Marks: 30

Credits: 4

Pass Percentage: 35%

Objective

This course helps student understand the need for Data Mining and advantages to the business world. They will get a clear idea of various classes of Data Mining techniques, their need, scenarios (situations), scope of their applicability and use of data visualization techniques

Section-A

Unit -1 Data Mining: Introduction, Scope, What is Data Mining; How does Data Mining Works, Predictive Modeling: Data Mining and Data Warehousing: Architecture for Data Mining: Profitable Applications: Data Mining Tools

Unit -II: Data Pre-processing: Overview, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

Unit- III: Data Mining Techniques- An Overview, Data Mining Versus Database Management System, Data Mining Techniques- Association rules, Classification, Regression, Clustering, Neural networks.

Unit -IV Clustering: Introduction, Cluster Analysis, Clustering Methods- K means, Hierarchical clustering, Agglomerative clustering, Divisive clustering, evaluating clusters.

Section-B

Unit -V Applications of Data Mining: Introduction, Business Applications Using Data Mining- Risk management and targeted marketing, Customer profiles and feature construction, Medical applications (diabetic screening), Scientific Applications using Data Mining, Other Applications.

Unit -VI Data Visualization: Introduction, Acquiring and Visualizing Data, Simultaneous acquisition and visualization, Applications of Data Visualization, Keys factors of Data Visualization (Control of Presentation, Faster and Better JavaScript processing, Rise of HTML5, Lowering the implementation Bar)

Unit -VII Exploring the Visual Data Spectrum: charting Primitives (Data Points, Line Charts, Bar Charts, Pie Charts, Area Charts), Exploring advanced Visualizations (Candlestick Charts, Bubble Charts, Surface Charts, Map Charts, Infographics).

Unit -VIII Visualizing data Programmatically: Starting with Google charts (Google Charts API Basics, A Basic bar chart, A basic Pie chart, Working with Chart Animations).

Suggested Readings

1. Jiawei Han, Micheline Kamber and Jian Pei, Data Mining Concepts and Techniques, Third Edition, 2000
2. Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, Introduction to Data Mining, Pearson 2005,
3. M. Kantardzic, "Data Mining: Concepts, Models, Methods, and Algorithms", 2nd edition, Wiley-IEEE Press, 2011
4. Jon Raasch, Graham Murray, Vadim Ogievetsky, Joseph Lowery, JavaScript and jQuery for Data Analysis and Visualization, 2014
5. Ben Fry, "Visualizing data: Exploring and explaining data with the processing environment", O'Reilly, 2007



JAGAT GURU NANAK DEV PUNJAB STATE OPEN UNIVERSITY, PATIALA
(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

BSDB32301T: DATAMINING AND VISULAIZATION
COURSE COORDINATOR AND EDITOR: DR. AMITOJ SINGH

UNIT NO.	UNIT NAME
UNIT 1	DATA MINING
UNIT 2	DATA PRE-PROCESSING
UNIT 3	DATA MINING TECHNIQUES
UNIT 4	CLUSTERING
UNIT 5	APPLICATIONS OF DATA MINING
UNIT 6	DATA VISUALIZATION
UNIT 7	EXPLORING THE VISUAL DATA SPECTRUM
UNIT 8	VISUALIZING DATA PROGRAMMATICALLY

B.Sc.(DATA SCIENCE)
SEMESTER-III
DATA MINING AND VISUALIZATION

UNIT-I DATA MINING

STRUCTURE

1.1 Introduction

1.2 Scope

1.3 What is Data Mining

1.4 How does Data Mining Works

1.5 Predictive Modeling

1.6 Data Mining and Data Warehousing

1.7 Architecture for Data Mining

1.8 Profitable Applications

1.9 Data Mining Tools

1.1 INTRODUCTION

Data mining is knowledge discovery from data. Data mining is also called knowledge discovery and data mining (KDD). Lots of data is being collected and stored. Examples are web data, e-commerce, purchases at names, which persons are the least likely to default on their credit cards? Which types of transactions are likely to be fraudulent, given the transactional history of a particular customer? Which of my customers are likely to be the most loyal, and which are most likely to leave for a competitor? Data Mining helps extract such information

Data Explosion Problem

Automated data collection tools and mature database technology leads to tremendous amount of data stored in databases and data warehouses. The fact is that we are drowning in data, but starving for knowledge!

Data mining involves extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) patterns or knowledge from huge amount of data. It is the application of descriptive and predictive analysis to support the marketing, sales and service functions. Although data mining can be performed on operational databases, it is more commonly applied to the more stable datasets held in data marts or warehouses

1.2 SCOPE

Data mining technology can generate new business opportunities by providing databases of sufficient size and quality, which leads to extraction of interesting knowledge of rules, regularities, patterns, and constraints in data in large data bases.

Automated prediction of trends and behaviors: Data mining automates the process of finding predictive information in large databases. Predictive problems include forecasting bankruptcy and other forms of default, and identifying segments of a population likely to respond similarly to given events.

Automated discovery of previously unknown patterns: Data mining tools inspect databases and identify previously hidden patterns in one step. An example of pattern discovery is the analysis of retail sales data to identify seemingly unrelated products that are often purchased together.

1.3 WHAT IS DATA MINING

Data mining is a process used by companies to turn raw data into useful information. By using software to look for patterns in large batches of data, businesses can learn more about their customers to develop more effective marketing strategies, increase sales and decrease costs. Data mining is the extraction of hidden predictive information from large databases and is a powerful new technology with great potential to help companies focus on the most important information in their data warehouses.

Data mining tools predict future trends and behaviors, allowing businesses to make proactive, knowledge-driven decisions. Data mining tools can answer business questions that traditionally were too time consuming to resolve. They examine databases for hidden patterns, finding predictive information that experts may miss because it lies outside their expectations.

What kind of data can be mined?

- Database-oriented data sets and applications
- Relational database, data warehouse, transactional database
- Advanced data sets and advanced applications
- Data streams and sensor data
- Time-series data, temporal data, sequence data (incl. bio-sequences)
- Structure data, graphs, social networks and multi-linked data
- Object-relational databases
- Heterogeneous databases and legacy databases
- Spatial data and spatiotemporal data
- Multimedia database
- Text databases
- The World-Wide Web

1.4 HOW DOES DATA MINING WORKS

Data mining is the process of understanding data through cleaning raw data, finding patterns, creating models, and testing those models. It includes statistics, machine learning, and database systems. After this, application software sorts the data based on the user's results, and finally, the end-user presents the data in an easy-to-understand format, such as a graph or table.

Data mining involves six phase workflow:

The first step in data mining is data collection. Today's organizations can collect records, logs, website visitors' data, application data, sales data, and more every day.

1. Business understanding

Comprehensive data mining projects start by first identifying project objectives and scope. The business investors will ask a question or state a problem that data mining can answer or solve.

2. Data understanding

Once the business problem is understood, data relevant to the problem is collected. This data often comes from multiple sources, including structured data and unstructured data. This stage may include some investigative analysis to expose some preliminary patterns. At the

end of this phase, the data mining team has selected the subset of data for analysis and modeling.

3. Data preparation

This phase begins with more thorough work. Data preparation involves preparing the final data set, which includes all the relevant data needed to answer the business question. Stakeholders will identify the dimensions and variables to explore and prepare the final data set for model creation.

4. Modeling

In this phase, the appropriate modeling techniques are selected for the given data. These techniques can include clustering, predictive models, classification, estimation, or a combination.

5. Evaluation

After creating the models, the test and measure of their success is evaluated. The model may answer things not accounted for, and the model needs to be edited. This phase is designed to allow to look at the progress so far and ensure it's on the right track for meeting the business goals.

6. Deployment

Finally, once the model is accurate and reliable, it is installed in the real world. The deployment can take place within the organization, be shared with customers, or be used to generate a report for participants to prove its reliability. The work doesn't end when the last line of code is complete; deployment requires careful thought, and a way to make sure the right people are appropriately informed. The data mining team is responsible for the user's understanding of the project.

Data mining involves six common classes of tasks:

1. Anomaly detection – The identification of unusual data records, that might be interesting or data errors that require further investigation.
2. Association rule learning (Dependency modelling) – Searches for relationships between variables. For example a supermarket might gather data on customer purchasing habits. Using association rule learning, the supermarket can determine which products are frequently bought together and use this information for marketing purposes.
3. Clustering – is the task of discovering groups and structures in the data that are in some way or another "similar", without using known structures in the data.
4. Classification – is the task of generalizing known structure to apply to new data. For example, an e-mail program might attempt to classify an e-mail as "legitimate" or as "spam".
5. Regression – attempts to find a function which models the data with the least error.

6. Summarization – providing a more compact representation of the data set, including visualization and report generation

1.5 PREDICTIVE MODELING

Predictive modeling are a form of data-mining technology that works by analyzing historical and current data and generating a model to help predict future outcomes. Predictive models analyze past performance to assess how likely a customer is to exhibit a specific behavior in the future.

In predictive modeling, data is collected, a statistical model is formulated, predictions are made, and the model is validated as additional data becomes available. For example, risk models can be created to combine member information in complex ways with lifestyle information from external sources to improve assuring accuracy. This category also includes models that seek out indirect data patterns to answer questions about customer performance, such as fraud detection models. Predictive models often perform calculations during live transactions. For example, to evaluate the risk of a given customer or transaction to guide a decision

1.6 DATA MINING AND DATA WAREHOUSING

Data warehousing is the process of pooling all relevant data together. Data mining is considered as a process of extracting data from large data sets.

Data mining comprises the following:

Data mining is the process of analyzing data patterns.

Data is analyzed regularly.

Data mining is the use of pattern recognition logic to identify patterns

Data mining is carried by business users with the help of engineers.

Data mining is considered as a process of extracting data from large data sets.

Data is stored periodically.

Data mining is carried by business users with the help of engineers.



Data Mining Process

Data warehousing comprises the following:

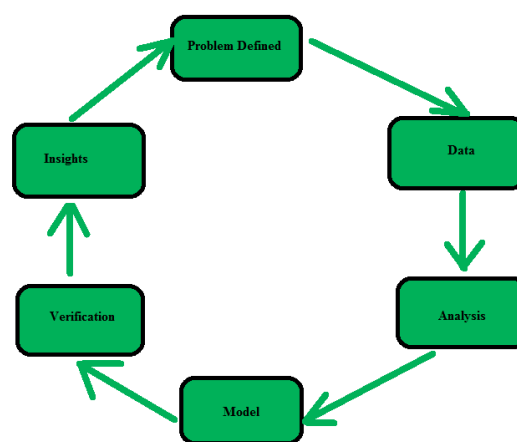
A data warehouse is database system which is designed for analytical analysis instead of transactional work.

Data is stored periodically.

Data warehousing is the process of extracting and storing data to allow easier reporting.

Data warehousing is solely carried out by engineers.

Data warehousing is the process of pooling all relevant data together.



Data warehouse

1.7 ARCHITECTURE FOR DATA MINING

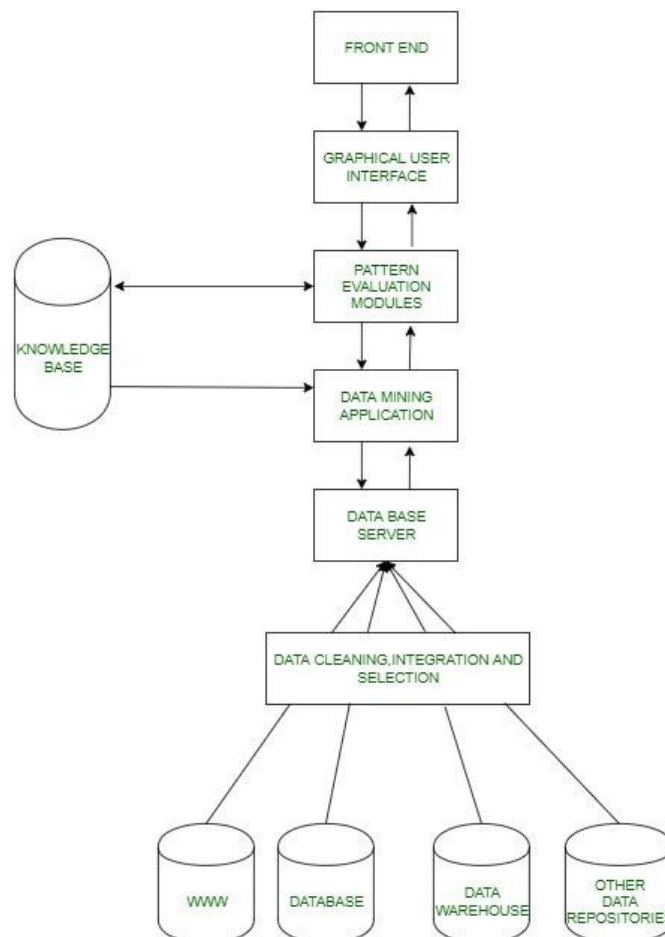
Data mining is an important method where previously unknown and potentially useful information is extracted from the vast amount of data. The data mining process involves several components, and these components constitute a data mining system architecture.

The significant components of data mining systems are a data source, data mining engine, data warehouse server, the pattern evaluation module, graphical user interface, and knowledge base.

Basic Working:

- It starts when the user puts up certain data mining requests, these requests are then sent to data mining engines for pattern evaluation.
- These applications try to find the solution of the query using the already present database.

- The metadata then extracted is sent for proper analysis to the data mining engine which sometimes interacts with pattern evaluation modules to determine the result.
- This result is then sent to the front end in an easily understandable manner using a suitable interface.



Architecture for Data Mining

The architecture of data mining has the following components:

1. Data Sources

Database, World Wide Web (WWW) and data warehouse are parts of data sources. The data in these sources may be in the form of plain text, spreadsheets or in other forms of media like photos or videos. WWW is one of the biggest sources of data.

2. Database Server

The database server contains the actual data ready to be processed. It performs the task of handling data retrieval as per the request of the user.

3. Data Mining Engine

It is one of the core components of the data mining architecture that performs all kinds of data mining techniques like association, classification, characterization, clustering, prediction, etc.

4. Pattern Evaluation Modules

They are responsible for finding interesting patterns in the data and sometimes they also interact with the database servers for producing the result of the user requests.

5. Graphic User Interface

Since the user cannot fully understand the complexity of the data mining process so graphical user interface helps the user to communicate effectively with the data mining system.

6. Knowledge Base

Knowledge Base is an important part of the data mining engine that is quite beneficial in guiding the search for the result patterns. Data mining engine may also sometimes get inputs from the knowledge base. This knowledge base may contain data from user experiences. The objective of the knowledge base is to make the result more accurate and reliable.

1.8 PROFITABLE APPLICATIONS

Data Mining is mainly used today by companies with a strong consumer focus. Some examples are retail, financial, communication, and marketing organizations, customer preferences and product positioning, impact on sales, customer satisfaction and corporate profits.

Some areas where data mining is widely used

- **Future Healthcare**

Data mining holds great possibilities to improve health systems. It uses data and analytics to identify best practices that improve care and reduce costs. Researchers use data mining approaches like multi-dimensional databases, machine learning, soft computing, data visualization and statistics. Mining can be used to predict the volume of patients in every category. Processes are developed that make sure that the patients receive appropriate care at the right place and at the right time. Data mining can also help healthcare insurers to detect fraud and abuse.

- **Market Basket Analysis**

Market basket analysis is a modelling technique based upon a theory that if you buy a certain group of items, you are more likely to buy another group of items. This technique may allow the retailer to understand the purchase behaviour of a buyer. This information may help the retailer to know the buyer's needs and change the store's layout accordingly. Using differential analysis comparison of results between different stores, between customers in different groups can be done.

- **Education**

Educational Data Mining (EDM) concerns with developing methods that discover knowledge from data originating from educational Environments. The goals of EDM are identified as predicting students' future learning behaviour, studying the effects of educational support, and advancing scientific knowledge about learning. Data mining can be used by an institution

to take accurate decisions and also to predict the results of the student. With the results the institution can focus on what to teach and how to teach. Learning pattern of the students can be captured and used to develop techniques to teach them.

- **Manufacturing Engineering**

Manufacturing Engineers focus on the design and operation of integrated systems for the production of high-quality, economically competitive products. These systems may include computer networks, robots, machine tools, and materials-handling equipment. Data mining tools can be very useful to discover patterns in complex manufacturing process. Data mining can be used in system-level designing to extract the relationships between product architecture, product portfolio, and customer needs data. It can also be used to predict the product development span time, cost, and dependencies among other tasks.

- **Customer Relationship Management**

Customer Relationship Management (CRM) is about acquiring and retaining customers, also improving customers' loyalty and implementing customer focused strategies. To maintain a proper relationship with a customer a business needs to collect data and analyse the information. This is where data mining plays its part. With data mining technologies the collected data can be used for analysis. Instead of being confused where to focus to retain customer, the seekers for the solution get filtered results.

- **Fraud Detection**

Traditional methods of fraud detection are time consuming and complex. Data mining aids in providing meaningful patterns and turning data into information. Any information that is valid and useful is knowledge. A perfect fraud detection system should protect information of all the users. A supervised method includes collection of sample records. These records are classified fraudulent or non-fraudulent. A model is built using this data and the algorithm is made to identify whether the record is fraudulent or not.

- **Lie Detection**

Arresting a criminal is easy whereas bringing out the truth from him is difficult. Law enforcement can use mining techniques to investigate crimes, monitor communication of suspected terrorists. This field includes text mining also. This process seeks to find meaningful patterns in data which is usually unstructured text. The data sample collected from previous investigations are compared and a model for lie detection is created. With this model processes can be created according to the necessity.

- **Financial Banking**

With computerised banking everywhere, huge amount of data is supposed to be generated with new transactions. Data mining can contribute to solving business problems in banking and finance by finding patterns, causalities, and correlations in business information and market prices that are not immediately apparent to managers because the volume data is too large or is generated too quickly to screen by experts. The managers may find these

information for better segmenting, targeting, acquiring, retaining and maintaining a profitable customer.

- **Research Analysis**

Data mining is helpful in data cleaning, data pre-processing and integration of databases. The researchers can find any similar data from the database that might bring any change in the research. Identification of any co-occurring sequences and the correlation between any activities can be known. Data visualisation and visual data mining provide us with a clear view of the data.

1.9 DATA MINING TOOLS

Data Mining is the set of techniques that utilize specific algorithms, statical analysis, artificial intelligence, and database systems to analyze data from different dimensions and perspectives

Data Mining tools have the objective of discovering patterns/trends/groupings among large sets of data and transforming data into more advanced information. It is a framework, such as Rstudio or Tableau that allows you to perform different types of data mining analysis. Such a framework is called a data mining tool.

Users can customise data visualisation to suit their business requirement. It features smart drag and drop templates and also includes visualisation maps. Data mining has come a long way, evolving at every step. With various tools already in the market and various other tools which are constantly being added up in the list. Top 10 Free Data Mining Tools for 2021

1. Rapid Miner
2. Orange
3. Weka
4. Sisense
5. Revolution
6. Qlik
7. SAS Data Mining
8. Teradata
9. InetSoft
10. Dundas

1. **Rapid Miner:** So far, this is one of the best tools which uses data to forecast various information. This tool takes up the JAVA language for receiving instructions and is a very insightful tool for predictive analysis. This tool can be used for a lot of functions like training, business applications, etc. This tool makes use of flow-based programming which makes data visualization much easier. It has tools for statistical analysis which are easy to use. Also, one does not need extensive knowledge of coding before using this product.

2. Orange: It is a machine learning software that is component-based and makes data visualization a lot easier. It provides various widgets which analyzes the data and then make it ready for visualization. It has a user engagement platform which is both fun and easy to use. Orange is an open-source data mining platform that can work with both scripts and with ETL workflow. This is one of the simplest tools to operate as it is programmed in Python language which is easier to learn compared to the other programming languages.

3. Weka: This machine learning software is one of the best tools for analyzing data. This tool also aids in predictive modeling and data visualization. This too is written in the JAVA programming language. It can also provide access to various SQL databases which can be analyzed further. It is an open-source tool that is free for use. This tool is mostly used for developing new machine learning algorithms and can support data files from multiple sources. parameters poses a huge challenge.

4. Sisense: It is one of the best artificial intelligence and data aggregation platform. It caters to the needs of different organizations based on the size of the company, the sector in which the company operates, etc. It further combines data from multiple sources and saves it for later use. It also generates visual reports which make the understanding even easier.

5. Revolution: Commonly known as R, this tool provides an interactive platform for statistical operations and data visualization. It is designed in a way that makes it quite user-friendly. It mines the data quite easily. Also, one could perform quite intricate statistical calculations. It makes use of several statistical functions to analyze the data. Also, it makes the heavy programming quite concise and less cumbersome. It has really good graphics features and elements.

6. Qlik: One of the most widely used Business Intelligence tools, Qlik has a easy to use data mining and visualisation platform. This tool allows users to fetch, integrate, process and analyse data from multiple sources.

7. SAS Data Mining: SAS data mining can be used for descriptive and predictive modelling. This tool is especially helpful for developing models quickly, understanding key relations and identifying patterns for streamlining the data mining process. This tool is best suited for text mining and optimisation. It also comes with distributed memory processing architecture that can be scaled up to fit business goals.

8. Teradata: Teradata offers a blend of tools, technologies and expertise that can optimise data mining. Users can integrate this tool into their existing systems and use data from varied sources. This tool is especially beneficial for organisations which have migrated or are migrating to the cloud. Additionally, it supports SQL and provides extensions for data tables. Users can even distribute the data to the discs without much manual intervention.

9. InetSoft: Inetsoft's data mining tool helps users to transform data into uniform data points to aid the analysing process even if it has been sourced variously. This tool can quickly transform data from both structured and unstructured sources . Users can optimise their own apps for updating and data consumption through Inetsoft's on-premise applications. It also allows users to share paginated reports with parameterisation and the corresponding business logic.

10. Dundas: If you are looking for an enterprise-level data mining tool, Dundas could be the perfect fit for you. Dundas can be used for building interactive dashboards, reports and more which can be used at scale. Companies often use it as the central data portal which all the employees can access.

PRACTICE QUESTIONS

1. What you mean by data mining?
2. Explain an example of data mining.
3. What is data mining and how it works?
4. What is data mining and its types?

B.Sc.(DATA SCIENCE)
SEMESTER-IV
DATA MINING AND VISUALIZATION

UNIT-II DATA PREPROCESSING

STRUCTURE

2.0 Objective

2.1 Introduction

2.2 Data Cleaning

2.3 Data Integration and Transformation

2.4 Data Reduction

2.5 Discretization and Concept Hierarchy Generation

2.6 Concept Hierarchies

2.7 Summary

2.7 Question

2.0 OBJECTIVE

To understand data preprocessing techniques and their effect on data

2.1 INTRODUCTION

Data preprocessing is crucial in any data mining process as they directly impact success rate of the project. Data is said to be unclean if it is missing attribute, attribute values, contain noise or outliers and duplicate or wrong data. Presence of any of these will degrade quality of the results. Preprocessing in Data Mining is a data mining technique which is used to transform the raw data in a useful and efficient format. Data preprocessing is the process of transforming raw data into an understandable format. It is also an important step in data mining as we cannot work with raw data. The quality of the data should be checked before applying machine learning or data mining algorithms.

Data preprocessing is needed as real-world data is often incomplete, inconsistent, and/or lacking in certain behaviors or trends, and is likely to contain many errors. Data preprocessing is a proven method of resolving such issues. Preprocessing of data is mainly to check the data quality. The quality can be checked by the following

- **Accuracy:** To check whether the data entered is correct or not.
- **Completeness:** To check whether the data is available or not recorded.
- **Consistency:** To check whether the same data is kept in all the places that do or do not match.
- **Timeliness:** The data should be updated correctly.
- **Believability:** The data should be trustable.

Major Tasks in Data Preprocessing:

- 1. Data cleaning**
- 2. Data integration**
- 3. Data reduction**
- 4. Data transformation**

Example

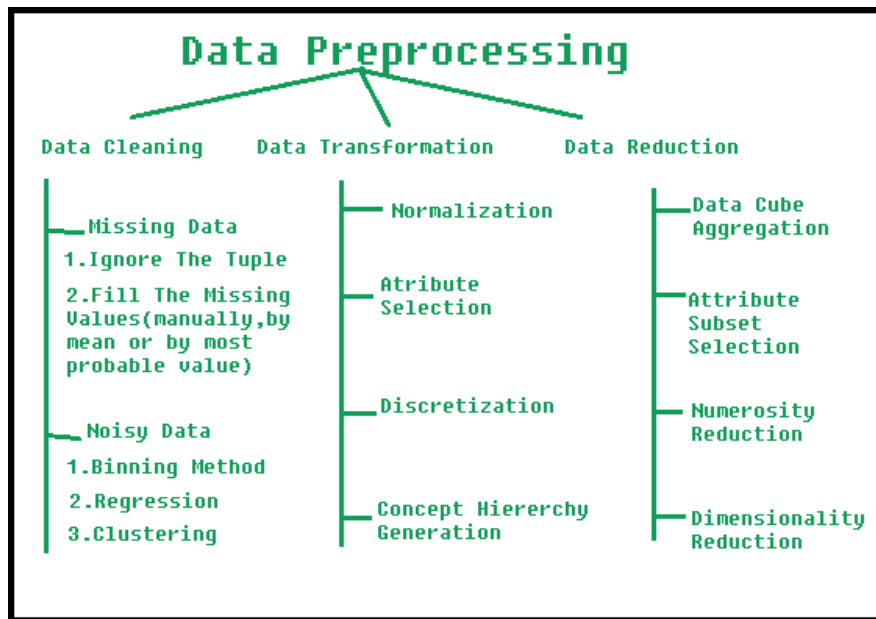
In this example we have 5 Adults in our dataset who have the Sex of Male or Female and whether they are pregnant or not. We can detect that Adult 3 and 5 are impossible data combinations.

		Sex	Pregnant
Adult	1	Male	No
	2	Female	Yes
	3	Male	Yes
	4	Female	No
	5	Male	Yes

We can perform a Data cleansing and choose to delete such data from our table. We remove such data because we can determine that such data existing in the dataset is caused by user entry errors or data corruption. A reason that one might have to delete such data is because the impossible data will affect the calculation or data manipulation process in the later steps of the

		Sex	Pregnant
Adult	1	Male	No
	2	Female	Yes
	4	Female	No

Data preprocessing is a data mining technique which is used to transform the raw data in a useful and efficient format.



Steps Involved in Data Preprocessing:

2.1 DATA CLEANING

The data can have many irrelevant and missing parts. To handle this, data cleaning is done, which involves handling of missing data, noisy data etc.

(a). Missing Data:

This situation arises when some data is missing in the data. It can be handled in various ways.

Some of them are:

1. Ignore the tuples:

This approach is suitable only when the dataset we have is quite large and multiple values are missing within a tuple.

2. Fill the Missing values:

There are various ways to do this task. You can choose to fill the missing values manually, by attribute mean or the most probable value.

(b). Noisy Data:

Noisy data is a meaningless data that can't be interpreted by machines. It can be generated due to faulty data collection, data entry errors etc. It can be handled in following ways:

1. **Binning Method:** This method works on sorted data in order to smooth it. The whole data is divided into segments of equal size and then various methods are performed to

complete the task. Each segmented is handled separately. One can replace all data in a segment by its mean or boundary values can be used to complete the task.

2. **Regression:** Here data can be made smooth by fitting it to a regression function. The regression used may be linear, which means having one independent variable, or having multiple independent variables.
3. **Clustering:** This approach groups the similar data in a cluster. The outliers may be undetected or it will fall outside the clusters.

2.2. DATA TRANSFORMATION

This is needed to transform the data in appropriate forms suitable for mining process, This involves following ways:

1. **Normalization:** It is done in order to scale the data values in a specified range (-1.0 to 1.0 or 0.0 to 1.0)
2. **Attribute Selection:** In this strategy, new attributes are constructed from the given set of attributes to help the mining process.
3. **Discretization:** This is done to replace the raw values of numeric attribute by interval levels or conceptual levels.
4. **Concept Hierarchy Generation:** Here attributes are converted from lower level to higher level in hierarchy. For Example-The attribute “city” can be converted to “country”.

2.3 DATA REDUCTION

Data mining is a technique that is used to handle huge amount of data. While working with huge volume of data, analysis became harder. In order to get free of this, data reduction technique is used. It aims to increase the storage efficiency and reduce data storage and analysis costs.

The various steps to data reduction are:

1. Data Cube Aggregation

Aggregation operation is applied to data for the construction of the data cube.

This technique is used to aggregate data in a simpler form. For example, imagine that information that is gathered for the analysis for the years 2012 to 2014, that data includes the revenue of your company every three months. They involve you in the annual sales, rather than the quarterly average. Therefore, it can be summarized that that the resulting data summarizes the total sales per year instead of per quarter. It summarizes the data.

1. Suppose there are the following attributes in the data set in which few attributes are redundant.
2. Initial attribute Set: {X1, X2, X3, X4, X5, X6}
3. Initial reduced attribute set: { }

- 4. Step-1: {X1}
- 5. Step-2: {X1, X2}
- 6. Step-3: {X1, X2, X5}
- 7. Final reduced attribute set: {X1, X2, X5}

Step-wise Backward Selection

This selection starts with a set of complete attributes in the original data and at each point, it eliminates the worst remaining attribute in the set.

Suppose there are the following attributes in the data set in which few attributes are redundant. Initial attribute Set: {X1, X2, X3, X4, X5, X6}

- Initial reduced attribute set: {X1, X2, X3, X4, X5, X6 }
- Step-1: {X1, X2, X3, X4, X5}
- Step-2: {X1, X2, X3, X5}
- Step-3: {X1, X2, X5}
- Final reduced attribute set: {X1, X2, X5}

Combination of Forward and Backward Selection

It allows us to remove the worst and select best attributes, saving time and making the process faster.

2 Data Compression

The data compression technique reduces the size of the files using different encoding mechanisms. It can be divided into two types based on their compression techniques.

- **Lossless Compression**
Encoding techniques (Run Length Encoding) allows a simple and minimal data size reduction. Lossless data compression uses algorithms to restore the precise original data from the compressed data.
- **Lossy Compression**
Methods such as Discrete Wavelet transform technique, PCA (principal component analysis) are examples of this compression. For e.g., JPEG image original the image. In lossy-data compression, the decompressed data may differ to the original data but are useful enough to retrieve information from them.

3. Numerosity Reduction

In this reduction technique the actual data is replaced with mathematical models or smaller representation of the data instead of actual data, it is important to only store the model parameter. Or non-parametric method such as clustering, histogram, sampling. For More Information on Numerosity Reduction Visit the link below:

4. Discretization & Concept Hierarchy Operation

Techniques of data discretization are used to divide the attributes of the continuous nature into data with intervals. We replace many constant values of the attributes by labels of small intervals. This means that mining results are shown in a concise, and easily understandable way.

Top-down discretization

If you first consider one or a couple of points (so-called breakpoints or split points) to divide the whole set of attributes and repeat of this method up to the end, then the process is known as top-down discretization also known as splitting.

Bottom-up discretization

If you first consider all the constant values as split-points, some are discarded through a combination of the neighbourhood values in the interval, that process is called bottom-up discretization.

5. Concept Hierarchies

It reduces the data size by collecting and then replacing the low-level concepts (such as 43 for age) to high-level concepts (categorical variables such as middle age or Senior). For numeric data following techniques can be followed:

6. Binning

Binning is the process of changing numerical variables into categorical counterparts. The number of categorical counterparts depends on the number of bins specified by the user.

7. Histogram analysis

Like the process of binning, the histogram is used to partition the value for the attribute X, into disjoint ranges called brackets. There are several partitioning rules:

1. **Equal Frequency partitioning:** Partitioning the values based on their number of occurrences in the data set.
2. **Equal Width Partioning:** Partioning the values in a fixed gap based on the number of bins i.e. a set of values ranging from 0-20.
3. **Clustering:** Grouping the similar data together.

8. Attribute Subset Selection

Only the highly relevant attributes should be used, rest all can be discarded. For performing attribute selection, one can use level of significance and p- value of the attribute. The attribute having p-value greater than significance level can be discarded.

9. Numerosity Reduction

This enables to store the model of data instead of whole data, for example: Regression Models.

10. Dimensionality Reduction

This reduces the size of data by encoding mechanisms. It can be lossy or lossless. If after reconstruction from compressed data, original data can be retrieved, such reduction are called lossless reduction else it is called lossy reduction. The two effective methods of dimensionality reduction are: Wavelet transforms and PCA (Principal Component Analysis)

Whenever we come across any data which is weakly important, then we use the attribute required for our analysis. It reduces data size as it eliminates outdated or redundant features.

Step-wise Forward Selection

The selection begins with an empty set of attributes later on we decide best of the original attributes on the set based on their relevance to other attributes. We know it as a p-value in statistics

2.5 DATA DISCRETIZATION AND CONCEPT HIERARCHY GENERATION

A concept hierarchy for a given numeric attribute attribute defines a discretization of the attribute. Concept hierarchies can be used to reduce the data y collecting and replacing low-level concepts (such as numeric value for the attribute age) by higher level concepts (such as young, middle-aged, or senior). Although detail is lost by such generalization, it becomes meaningful and it is easier to interpret.

Manual definition of concept hierarchies can be tedious and time-consuming task for the user or domain expert. Fortunately, many hierarchies are implicit within the database schema and can be defined at schema definition level. Concept hierarchies often can be generated automatically or dynamically refined based on statistical analysis of the data distribution. Discretization and Concept Hierarchy Generation for Numeric Data: It is difficult and laborious for to specify concept hierarchies for numeric attributes due to the wide diversity of possible data ranges and the frequent updates if data values. Manual specification also could be arbitrary.

Concept hierarchies for numeric attributes can be constructed automatically based on data distribution analysis. Five methods for concept hierarchy generation are defined below- binning histogram analysis entropy-based discretization and data segmentation by “natural partitioning

Binning: Attribute values can be discretized by distributing the values into bin and replacing each bin by the mean bin value or bin median value. These technique can be applied recursively to the resulting partitions in order to generate concept hierarchies.

Cluster Analysis: A clustering algorithm can be applied to partition data into clusters or groups. Each cluster forms a node of a concept hierarchy, where all noses are at the same conceptual level. Each cluster may be further decomposed into sub-clusters, forming a lower

level in the hierarchy. Clusters may also be grouped together to form a higher-level concept hierarchy.

Cluster Analysis: A clustering algorithm can be applied to partition data into clusters or groups. Each cluster forms a node of a concept hierarchy, where all nodes are at the same conceptual level. Each cluster may be further decomposed into sub-clusters, forming a lower level in the hierarchy. Clusters may also be grouped together to form a higher-level concept hierarchy.

Data Discretization techniques can be used to divide the range of continuous attribute into intervals. Numerous continuous attribute values are replaced by small interval labels. This leads to a concise, easy-to-use, knowledge-level representation of mining results.

Top-down discretization

If the process starts by first finding one or a few points (called split points or cut points) to split the entire attribute range, and then repeats this recursively on the resulting intervals, then it is called top-down discretization or splitting.

Bottom-up discretization

If the process starts by considering all of the continuous values as potential split-points, removes some by merging neighbourhood values to form intervals, then it is called bottom-up discretization or merging.

To summarize, Data discretization refers to a method of converting a huge number of data values into smaller ones so that the evaluation and management of data become easy. Another example is analytics, where we gather the static data of website visitors. ...

2.6 CONCEPT HIERARCHIES

A concept hierarchy defines a sequence of mappings from a set of low-level concepts to higher-level, more general concepts. Concept hierarchies can be used to reduce the data by collecting and replacing low-level concepts with higher-level concepts. In the multidimensional model, data are organized into multiple dimensions, and each dimension contains multiple levels of abstraction defined by concept hierarchies. This organization provides users with the flexibility to view data from different perspectives.

Data mining on a reduced data set means fewer input/output operations and is more efficient than mining on a larger data set. Because of these benefits, discretization techniques and concept hierarchies are typically applied before data mining, rather than during mining.

Transforming nominal data with the use of concept hierarchies allows higher-level knowledge patterns to be found. It allows mining at multiple levels of abstraction, which is a common requirement for data mining applications.

Many concept hierarchies are implicit within the database schema. For example, suppose that the dimension *location* is described by the attributes *number*, *street*, *city*, *province_or_state*, *zip_code*, and *country*. These attributes are related by a total order, forming a concept hierarchy such as “*street* < *city* < *province_or_state* < *country*” This hierarchy is shown in Figure 2.5 (a). Alternatively, the attributes of a dimension may be organized in a partial order, forming a lattice. An example of a partial order for the *time* dimension based on the attributes *day*, *week*, *month*, *quarter*, and *year* is “*day* < {*month* < *quarter*; *week*} < *year*.” This lattice structure is shown in Figure 2.5(b). A concept hierarchy that is a total or partial order among attributes in a database schema is called a schema hierarchy. Concept hierarchies that are common to many applications may be predefined in the data mining system. should provide users with the flexibility to tailor predefined hierarchies according to their particular needs. For example, users may want to define a fiscal year starting on April 1 or an academic year starting on September 1.

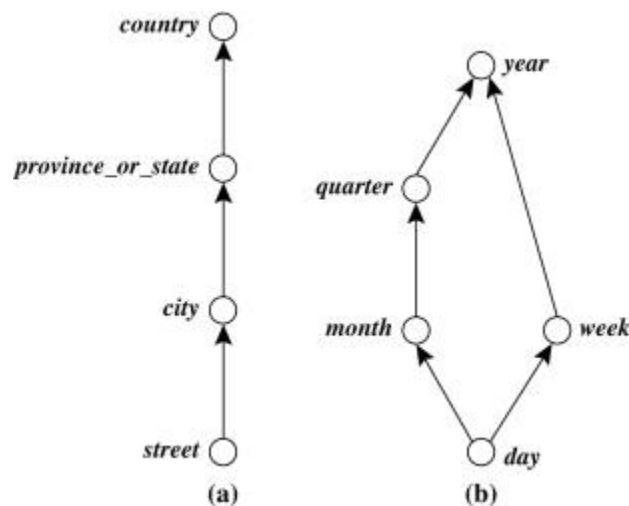


Figure 2.5(a).

Figure 2.5(b).

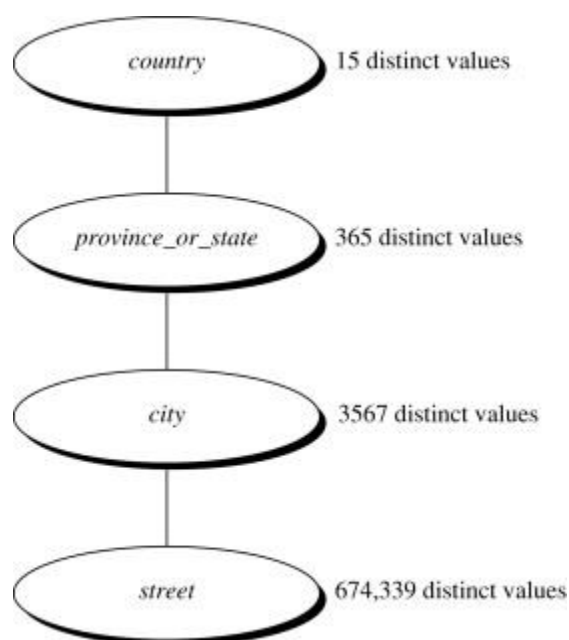
Concept hierarchy generation based on the number of distinct values per attribute

Exmple

Concept hierarchy generation using prespecified semantic connections

Suppose that a data mining expert has held together the five attributes *number*, *street*, *city*, *province_or_state*, and *country*, because they are closely linked semantically regarding the notion of *location*. If a user were to specify only the attribute *city* for a hierarchy defining *location*, the system can automatically drag in all five semantically related attributes to form a hierarchy. The user may choose to drop any of these attributes (e.g. *number* and *street*) from the hierarchy, keeping *city* as the lowest conceptual level.

A concept hierarchy for location can be generated automatically, as illustrated in Figure 2.6. First, sort the attributes in ascending order based on the number of distinct values in each attribute. Second, generate the hierarchy from the top down according to the sorted order, with the first attribute at the top level and the last attribute at the bottom level. Finally, the user can examine the generated hierarchy, and when necessary, modify it to reflect desired attributes. In this example, it is obvious that there is no need to modify the generated hierarchy.



2.7 SUMMARY

In summary, information at the schema level and on attribute–value counts can be used to generate concept hierarchies for nominal data. Transforming nominal data with the use of concept hierarchies allows higher-level knowledge patterns to be found. It allows mining at multiple levels of abstraction, which is a common requirement for data mining applications

The difference between data cleaning and data transformation is that Data cleaning is the process that removes data that does not belong in your dataset. Data transformation is the process of converting data from one format or structure into another. Transformation processes can also be referred to as data wrangling, or data munging, transforming and mapping data from one "raw" data form into another format for warehousing and analyzing.

While the techniques used for data cleaning may vary according to the types of data your company stores, you can follow these basic steps to map out a framework for your organization.

Step 1: Remove duplicate or irrelevant observations

Step 2: Fix structural errors

Step 3: Filter unwanted outliers

Step 4: Handle missing data

Step 5: Validate and QA

At the end of the data cleaning process, you should be able to answer these questions as a part of basic validation:

- Does the data make sense?
- Does the data follow the appropriate rules for its field?
- Does it prove or disprove your working theory, or bring any insight to light?
- Can you find trends in the data to help you form your next theory?
- If not, is that because of a data quality issue?

2.7 PRACTICE QUESTIONS

1. What is the process of data cleaning?
2. Explain an example of data cleaning.
3. What are the benefits of data cleaning and when data is clean?
4. What is data cleaning and data processing explain with proper example?
5. What is the purpose of concept hierarchy?
6. Why concept hierarchies are useful in data mining?

B.Sc.(DATA SCIENCE)

SEMESTER-IV

DATA MINING AND VISUALIZATION

UNIT-III DATA MINING TECHNIQUES

STRUCTURE

3.0 Objective

3.1 Introduction

3.2 Data Mining Versus Database Management System

3.3 Data Mining Techniques

3.3.1 Association rules

3.3.2 Classification

3.3.3 Regression

3.3.4 Clustering

3.3.5 Neural networks.

3.4 Summary

3.5 Applications of Cluster Analysis In Data Mining

3.6 Clustering is used in data mining

3.7 Question

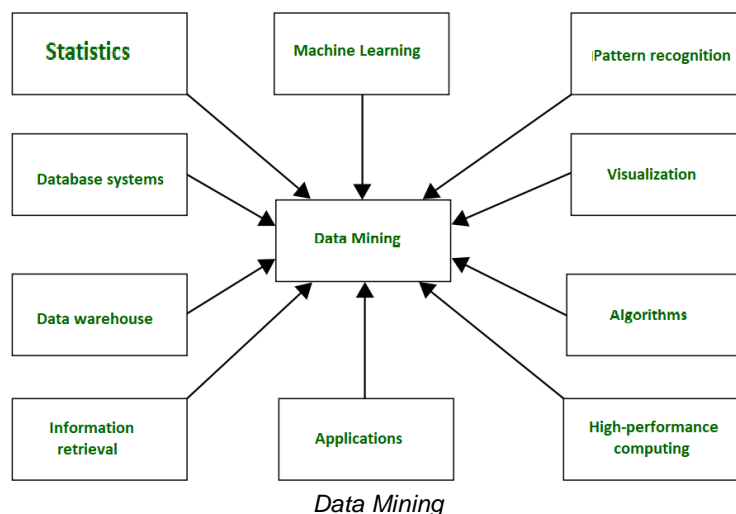
3.0 OBJECTIVE

To understand and implement some data mining

3.1 INTRODUCTION

“Mining” is the process of extraction of some valuable material from the earth e.g. coal mining, diamond mining, etc. In the context of computer science, “Data Mining” can be referred to as knowledge mining from data, knowledge extraction, data/pattern analysis, data archaeology, and data dredging. It is basically the process carried out for the extraction of useful information from a bulk of data. In the case of coal or diamond mining, the result of the extraction process is coal or diamond. But in the case of Data Mining, the result of the extraction process is not data. Instead, data mining results are the patterns and knowledge that we gain at the end of the extraction process. In that sense, we can think of Data Mining as a step in the process of Knowledge Discovery or Knowledge Extraction. However, the term ‘data mining’ became more popular in the business and press communities. Currently, Data Mining and Knowledge Discovery are used interchangeably. Nowadays, data mining is used in almost all places where a large amount of data is stored and processed. For example, banks typically use ‘data mining’ to find out their prospective customers who could be interested in credit cards, personal loans, or insurance as well. Since banks have the transaction details and detailed profiles of their customers, they analyze all this data and try to find out patterns that help them predict that certain customers could be interested in personal loans, etc.

Main Purpose of Data Mining



Basically, Data mining has been integrated with many other techniques from other domains such as statistics, machine learning, pattern recognition, database and data warehouse systems, information retrieval, visualization, etc. to gather more information about the data and to help predict hidden patterns, future trends, and behaviors and allows businesses to make decisions.

Data mining includes the utilization of refined data analysis tools to find previously unknown, valid patterns and relationships in huge data sets. These tools can incorporate statistical models, machine learning techniques, and mathematical algorithms, such as neural networks or decision trees. Thus, data mining incorporates analysis and prediction.

Following is a list of Data Mining Techniques: The Complete List

- Data cleaning and preparation.
- Tracking patterns.
- Classification.
- Association.
- Outlier detection.
- Clustering.
- Regression.
- Prediction.

3.2 DATA MINING VERSUS DATABASE MANAGEMENT SYSTEM

Database Management System (DBMS) is a full-fledged system for covering and managing a set of digital databases. It is a collection of computer programs that is dedicated for the management (i.e. organization, storage and retrieval) of all databases that are installed in a system (i.e. hard drive or network).

Data Mining

Data mining is also known as Knowledge Discovery in Data (KDD). It deals with the extraction of previously unknown and interesting information from raw data.

Data Mining

Data mining is the process of analyzing data from a different perspective and summarizing it into useful information – information that can be used to increase revenue cuts cost or both.

Data mining the analysis step of the knowledge discovery in database process. For example, data mining software can help retail companies find customers with a common interest.

The phrase data mining is commonly misused to describe software that presents data in new ways. True data mining software doesn't just change the presentation, but actually discovers the previously unknown relationship among the data.

Database

The database is a collection of interrelated data and a set of programs to access those data. It is a software system that manages data stored in the database. It provides an effective method of defining, storing and retrieving the information contained in the database. (the primary goal of a DBMS is to provide an environment that is both convenient and efficient to use in retrieving and storing database information. It provides users with information that they required. Some examples of DBMS packages are dBASE, FoxPro, FoxBase, Oracle, Ms-Access etc

Difference

DBMS is a full-fledged system for housing and managing a set of digital databases. However, Data Mining is a technique or a concept in computer science, which deals with extracting useful and previously unknown information from raw data. Most of the times, these raw data are stored in very large databases. Therefore, Data miners use the existing functionalities of DBMS to handle, manage and even preprocess raw data before and during the Data mining process. However, a DBMS system alone cannot be used to analyze data. But, some DBMS at present have inbuilt data analyzing tools or capabilities.

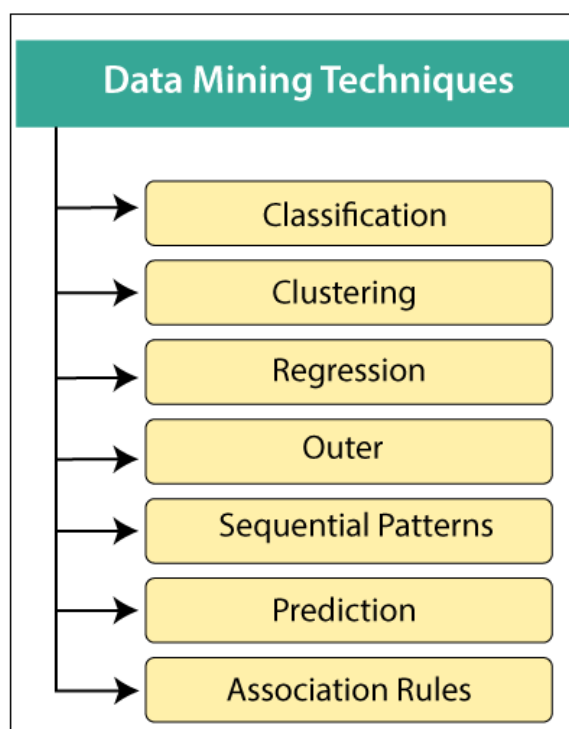
Database	Data mining
<p>The database is the organized collection of data. Most of the times, these raw data are stored in very large databases.</p> <p>A Database may contain different levels of abstraction in its architecture.</p> <p>Typically, the three levels: external, conceptual and internal make up the database architecture.</p>	<p>Data mining is analyzing data from different information to discover useful knowledge.</p> <p>Data mining deals with extracting useful and previously unknown information from raw data.</p> <p>The data mining process relies on the data compiled in the data warehousing phase in order to detect meaningful patterns.</p>

Is data mining a database?

Databases are growing in size to a stage where traditional techniques for analysis and visualization of the data are breaking down. Data mining and KDD are concerned with extracting models and patterns of interest from large databases.

3.3 DATA MINING TECHNIQUES

Data mining is a **process of extraction of useful information and patterns from huge data**. It is also called as knowledge discovery process, knowledge mining from data, knowledge extraction or data /pattern analysis. Data mining includes the **utilization of refined data analysis tools to find previously unknown**, valid patterns and relationships in huge data sets. These tools can incorporate statistical models, machine learning techniques, and mathematical algorithms, such as neural networks or decision trees.



3.3.1 Neural Networks

Neural networks are used for effective data mining in order to turn raw data into useful information. Neural networks look for patterns in large batches of data, allowing businesses to learn more about their customers which directs their marketing strategies, increase sales and lowers costs

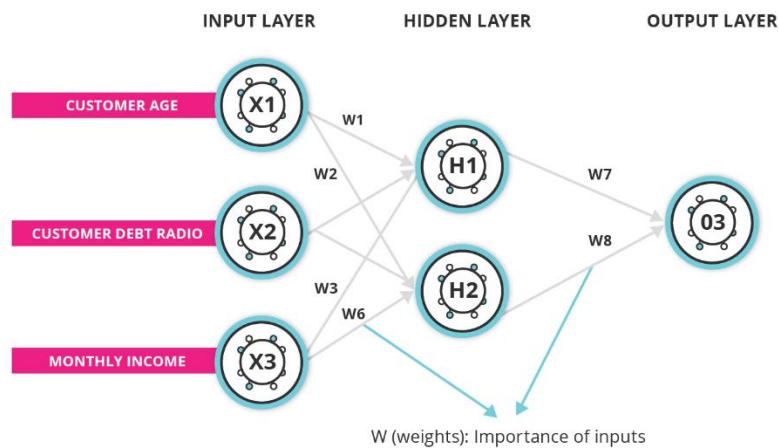


Image source: Mahanta, J. (Jul, 2017). 'Introduction to neural networks, advantages and applications'. Retrieved from Towards Data Science.

It involves a network of simple processing elements that exhibit complex global behavior determined by the connections between the processing elements and element parameters. Artificial neural networks are used with algorithms designed to alter the strength of the connections in the network to produce a desired signal flow.

A neural network is a series of algorithms that attempts to recognize underlying relationships in a set of data through a process that copies the way the human brain operates. Neural networks can adapt to changing input; so that the network generates the best possible result without needing to redesign the output criteria.

Neural Networks in Business

Most retail companies understand that their data provides decision-making information and is a valuable resource in business operations. As technology grows, businesses are leveraging neural networks for predictive analytics to fully harness the benefits of data streams.

Artificial Neural Networks (ANNs) can learn and model non-linear and complex relationships, and have the capacity to manage the reality between the relationship of inputs and outputs, as this is seldom linear or simple. There is a tendency to generalise and attribute unseen relationships on unseen data. ANNs also don't impose any restrictions on the input variables, unlike other prediction techniques

3.3.2 Association rules

In data science, association rules are used to find correlations and co-occurrences between data sets. They are ideally used to explain patterns in data from seemingly independent information sources, such as relational databases and transactional databases

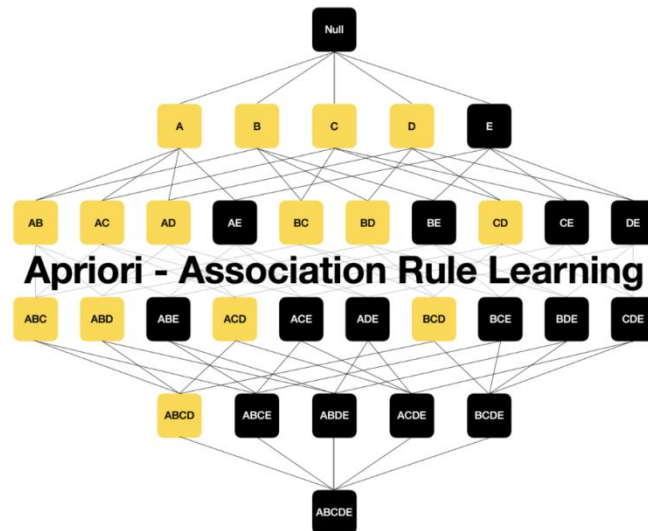
Association rule mining finds interesting associations and relationships among large sets of data items. This rule shows how frequently a itemset occurs in a transaction. A typical example is Market Based Analysis.

Market Based Analysis is one of the key techniques used by large relations to show associations between items. It allows retailers to identify relationships between the items that people buy together frequently.

A classic example of association rule mining refers to **a relationship between diapers and beers**. The example, which seems to be fictional, claims that men who go to a store to buy diapers are also likely to buy beer. Data that would point to that might look like this: A supermarket has 200,000 customer transactions. Given a set of transactions, we can find rules that will predict the occurrence of an item based on the occurrences of other items in the transaction. It is a data analysis task, i.e. the process of finding a model that describes and distinguishes data classes and concepts. Classification is the problem of identifying to which of a set of categories a new observation belongs to, on the basis of a training set of data containing observations and whose categories membership is known.

Many algorithms for generating association rules have been proposed. Some well-known algorithms are **Apriori, Eclat and FP-Growth Classification**

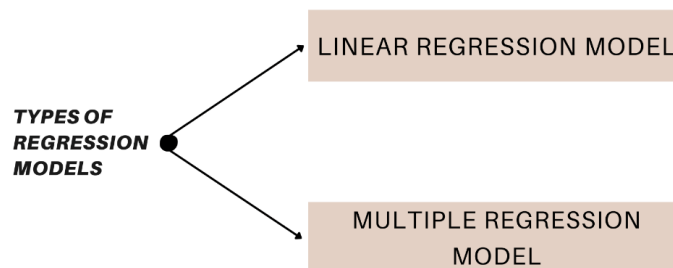
The Apriori algorithm finds association rules in a transaction data of items. A classification dataset, however, is normally in the form of a relational table, which is described by a set of distinct attributes (discrete and continuous)



3.3.3 Regression

Regression is a data mining technique used to predict a range of numeric values (also called continuous values), given a particular dataset. For example, regression might be used to predict the cost of a product or service, given other variables.

Regression techniques consist of finding a mathematical relationship between measurements of two variables, y and x , such that the value of variable y can be predicted from a measurement of the other variable, x



1

Types of Regression in Data Mining

Two types of Regression can be observed in data mining. Those two types are given below:

1. **Linear Regression Model**
2. **Multiple Regression Model**

Linear Regression is used mainly for the purpose of modeling the relationship between the two given variables. This is usually done by fitting a linear equation to perceive the data.

In addition to that, it can also be used for finding the mathematical relationship between the variables. It is the simplest form of Regression.

Multiple Regression Model

Multiple Regression Model is generally used to explain the relationship between multiple independent or multiple predictor variables.

Applications of Regression

Regression is widely used in many businesses and industries. It is very popular also. Listed below are some of the applications of Regression:

The process of Regression often involves the predictor variable (the values that are identified by the users) and the response variable (the values that are to be predicted).

To summarize, Regression can be defined as a data mining technique that is generally used for the purpose of predicting a range of continuous values (which can also be called “numeric values”) in a specific dataset.

3.3.3 Clustering

Clustering is an unsupervised Machine Learning-based Algorithm that comprises a group of data points into clusters so that the objects belong to the same group. Clustering helps to splits data into several subsets. Each of these subsets contains data similar to each other, and these subsets are called clusters. Now that the data from our customer base is divided into clusters, we can make an informed decision about who we think is best suited for this product.

In clustering, a group of different data objects is classified as similar objects. One group means a cluster of data. Data sets are divided into different groups in the cluster analysis, which is based on the similarity of the data. The general objective of clustering is to minimize the differences between members of a cluster while also maximizing the differences between clusters.

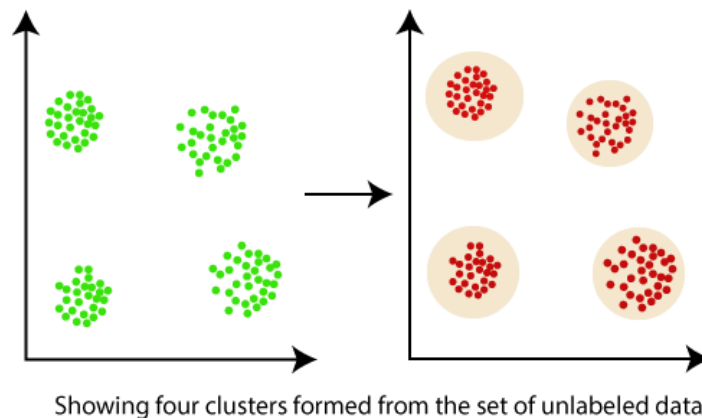
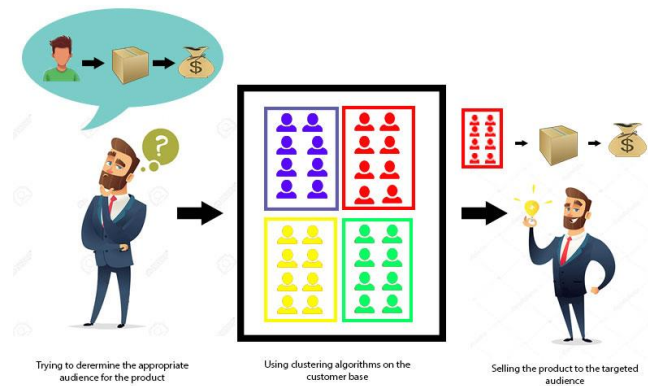
Clustering can help businesses to manage their data better –image segmentation, grouping web pages, market segmentation and information retrieval are four examples. For retail

businesses, data clustering helps with customer shopping behavior, sales campaigns and customer retention

Clustering is an unsupervised machine learning method of identifying and grouping similar data points in larger datasets without concern for the specific outcome. Clustering (sometimes called cluster analysis) is usually used to classify data into structures that are more easily understood and manipulated.

Example

Suppose we are a market manager, and we have a new tempting product to sell. We are sure that the product would bring enormous profit, as long as it is sold to the right people. So, how can we tell who is best suited for the product from our company's huge customer base?



7 Examples of Clustering Algorithms in Action.

- Identifying Fake News. Fake news is not a new phenomenon, but it is one that is becoming prolific. ...

- Spam filter. ...
- Marketing and Sales. ...
- Classifying network traffic. ...
- Identifying fraudulent or criminal activity. ...
- Document analysis. ...
- Fantasy Football and Sports.

3.4 SUMMARY

- A cluster is a subset of similar objects
- A subset of objects such that the distance between any of the two objects in the cluster is less than the distance between any object in the cluster and any object that is not located inside it.
- A connected region of a multidimensional space with a comparatively high density of objects.
- Clustering is the method of converting a group of abstract objects into classes of similar objects.
- Clustering is a method of partitioning a set of data or objects into a set of significant subclasses called clusters.
- It helps users to understand the structure or natural grouping in a data set and used either as a stand-alone instrument to get a better insight into data distribution or as a pre-processing step for other algorithms

Important points

- ✓ Data objects of a cluster can be considered as one group.
- ✓ We first partition the information set into groups while doing cluster analysis. It is based on data similarities and then assigns the levels to the groups.
- ✓ The over-classification main advantage is that it is adaptable to modifications, and it helps single out important characteristics that differentiate between distinct groups.

3.5 APPLICATIONS OF CLUSTER ANALYSIS IN DATA MINING:

- data analysis, market research, pattern recognition, and image processing.
- It assists marketers to find different groups in their client base and based on the purchasing patterns. They can characterize their customer groups.

- It helps in allocating documents on the internet for data discovery.
- Clustering is also used in tracking applications such as detection of credit card fraud.
- It helps in the identification of areas of similar land that are used in an earth observation database and the identification of house groups in a city according to house type, value, and geographical location.

3.6 CLUSTERING IS USED IN DATA MINING FOR THE FOLLOWING REASONS:

Clustering analysis has been an evolving problem in data mining due to its variety of applications. The advent of various data clustering tools in the last few years and their comprehensive use in a broad range of applications, including image processing, computational biology, mobile communication, medicine, and economics, must contribute to the popularity of these algorithms. The main issue with the data clustering algorithms is that it cant be standardized. The advanced algorithm may give the best results with one type of data set, but it may fail or perform poorly with other kinds of data set. Although many efforts have been made to standardize the algorithms that can perform well in all situations, no significant achievement has been achieved so far. Many clustering tools have been proposed so far. However, each algorithm has its advantages or disadvantages and cant work on all real situations.

3.7 QUESTIONS

1. What is clustering in data mining?
2. What are the examples of clustering?
3. What is Cluster Analysis example?
4. What is clustering and its applications?

B.SC. (DATA SCIENCE)
SEMESTER III
DATA MINING AND VISUALIZATION

UNIT- IV CLUSTERING

STRUCTURE

4.0 Objective

4.1 Introduction

4.2 Cluster Analysis

4.3 Clustering Methods

4.3.1 K means

4.3.2 Hierarchical clustering

4.3.3 Agglomerative clustering

4.3.4 Divisive clustering

4.4 Evaluating Clusters

4.5 Summary

4.6 Application

4.7 Question

4.0 OBJECTIVE

To understand the following clustering techniques

K means, Hierarchical clustering, Agglomerative clustering, Divisive clustering,

4.1 INTRODUCTION

Cluster is a group of objects that belongs to the same class. In other words, similar objects are grouped in one cluster and dissimilar objects are grouped in another cluster. Clustering is the process of making a group of abstract objects into classes of similar objects. A cluster of data objects can be treated as one group.

Cluster analysis or **clustering** is the task of grouping a set of objects in such a way that objects in the same group (called a **cluster**) are more similar (in some sense) to each other than to those in other groups (clusters).

It is a main task of exploratory data analysis, and a common technique for statistical data analysis, used in many fields, including pattern recognition, image analysis, information retrieval, bioinformatics, data compression, computer graphics and machine learning.



4.2 CLUSTER ANALYSIS

Clustering is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group than those in other groups. In simple words, the aim is to segregate groups with similar traits and assign them into clusters.

Cluster analysis is the general task to be solved by various algorithms that differ significantly in their understanding of what constitutes a cluster and how to efficiently find them. The appropriate clustering algorithm and parameter settings depend on the individual data set and intended use of the results. Cluster analysis as such is not an automatic task, but an iterative

process of knowledge discovery or interactive multi-objective optimization that involves trial and failure. It is often necessary to modify data preprocessing and model parameters until the result achieves the desired properties

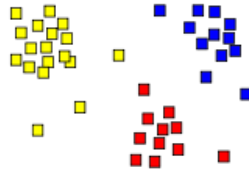


Figure shows the result of a cluster analysis shown as the coloring of the squares into three clusters

Cluster Analysis in Data Mining means that to find out the group of objects which are similar to each other in the group but are different from the object in other groups. While doing cluster analysis, first partition the set of data into groups based on data similarity and then assign the labels to the groups. The main advantage of clustering over classification is that, it is adaptable to changes and helps single out useful features that distinguish different groups. Clustering analysis is broadly used in many applications such as market research, pattern recognition, data analysis, and image processing.

Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups

Points to Remember

- A cluster of data objects can be treated as one group.
- While doing cluster analysis, we first partition the set of data into groups based on data similarity and then assign the labels to the groups.
- The main advantage of clustering over classification is that, it is adaptable to changes and helps single out useful features that distinguish different groups.

4.3 CLUSTERING METHODS

The method of identifying similar groups of data in a dataset is called clustering. It is one of the most popular techniques in data science. Entities in each group are comparatively more similar to entities of that group than those of the other groups.

Types of clustering algorithms

Since the task of clustering is subjective, this means that can be used for achieving this goal are plenty. Every methodology follows a different set of rules for defining the 'similarity; among data points. In fact, there are more than 100 clustering algorithms known. But few of the algorithms are used popularly:

- **Connectivity models:** As the name suggests, these models are based on the notion that the data points closer in data space exhibit more similarity to each other than the data points lying farther away. These models can follow two approaches. In the first approach, they start with classifying all data points into separate clusters & then aggregating them as the distance decreases. In the second approach, all data points are classified as a single cluster and then partitioned as the distance increases. Also, the choice of distance function is subjective. These models are very easy to interpret but lacks scalability for handling big datasets. Examples of these models are hierarchical clustering algorithm and its variants.
- **Centroid models:** These are iterative clustering algorithms in which the notion of similarity is derived by the closeness of a data point to the centroid of the clusters. K-Means clustering algorithm is a popular algorithm that falls into this category. In these models, the no. of clusters required at the end have to be mentioned beforehand, which makes it important to have prior knowledge of the dataset. These models run iteratively to find the local optima.
- **Distribution models:** These clustering models are based on the notion of how probable is it that all data points in the cluster belong to the same distribution (For example: Normal, Gaussian). These models often suffer from overfitting. A popular example of these models is Expectation-maximization algorithm which uses multivariate normal distributions.
- **Density Models:** These models search the data space for areas of varied density of data points in the data space. It isolates various different density regions and assign the data points within these regions in the same cluster.

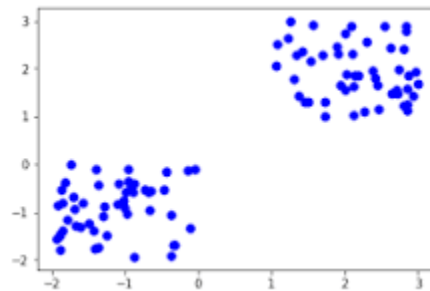
Broadly speaking, clustering can be divided into two subgroups :

- **Hard Clustering:** In hard clustering, each data point either belongs to a cluster completely or not.

- **Soft Clustering:** In soft clustering, instead of putting each data point into a separate cluster, a probability or likelihood of that data point to be in those clusters is assigned.

4.3.1 K Means Clustering

K-means clustering is one of the simplest and popular unsupervised machine learning algorithms. ... In other words, the K-means algorithm **identifies k number of centroids**, and then allocates every data point to the nearest cluster, while keeping the centroids as small as possible.



Kmeans algorithm is an iterative algorithm that tries to partition the dataset into Kpre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group. The less variation we have within clusters, the more homogeneous (similar) the data points are within the same cluster. K-means clustering algorithm computes the centroids and iterates until it finds optimal centroid. It assumes that the number of clusters are already known. It is also called flat clustering algorithm. The number of clusters identified from data by algorithm is represented by 'K' in K-means

K-means clustering is a method of vector quantization, that is popular for cluster analysis in data mining.

K-means Clustering Method

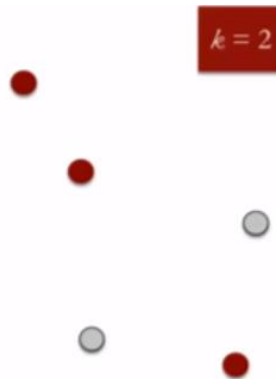
If k is given, the K-means algorithm can be executed in the following steps:

1. Partition of objects into k non-empty subsets
2. Identifying the cluster centroids (mean point) of the current partition.
3. Assigning each point to a specific cluster
4. Compute the distances from each point and allot points to the cluster where the distance from the centroid is minimum.
5. After re-allotting the points, find the centroid of the new cluster formed.

K means is an iterative clustering algorithm that aims to find local maxima in each iteration.

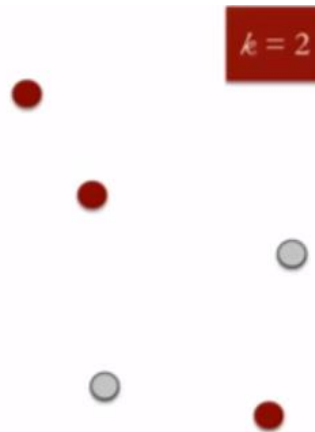
This algorithm works in these 5 steps :

1. Specify the desired number of clusters K : Let us choose $k=2$ for these 5 data points in 2-D space.



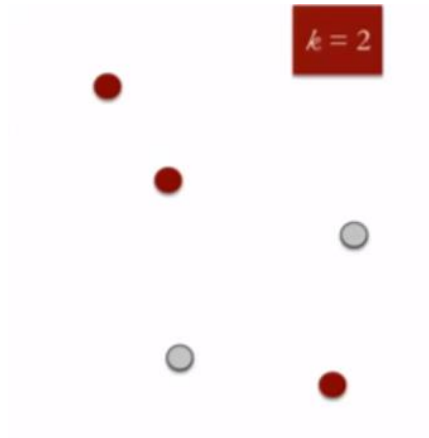
Step 1

2. Randomly assign each data point to a cluster : Let's assign three points in cluster 1 shown using red color and two points in cluster 2 shown using grey color.

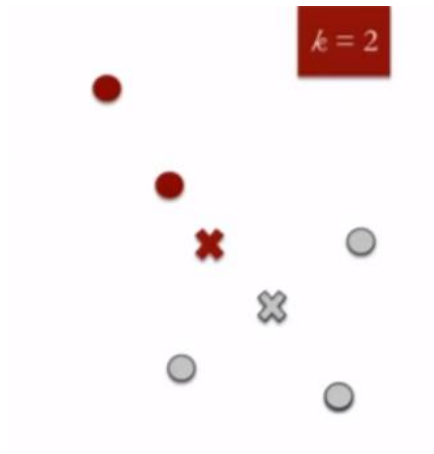


Step 2

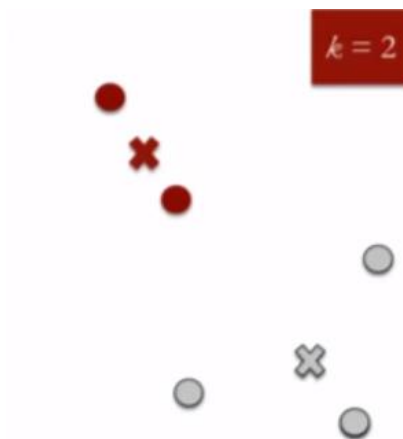
3. Compute cluster centroids : The centroid of data points in the red cluster is shown using red cross and those in grey cluster using grey cross.



4. Re-assign each point to the closest cluster centroid : Note that only the data point at the bottom is assigned to the red cluster even though its closer to the centroid of grey cluster. Thus, we assign that data point into grey cluster.



5. Re-compute cluster centroids : Now, re-computing the centroids for both the clusters.



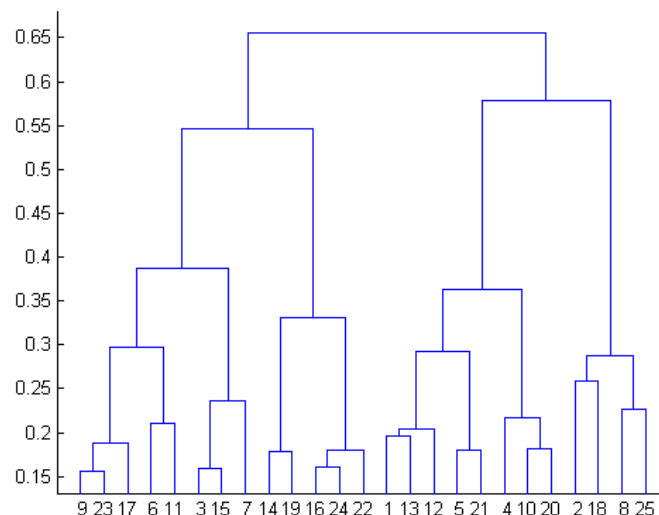
6. Repeat steps 4 and 5 until no improvements are possible : Similarly, repeat the 4th and 5th steps until you reach global optima. When there will be no further switching of data points between two clusters for two successive repeats. It will mark the termination of the algorithm if not explicitly mentioned.

4.3.2 Hierarchical Clustering

Hierarchical clustering, as the name suggests is an algorithm that builds hierarchy of clusters. Hierarchical clustering is an algorithm that groups similar objects into groups called clusters. The endpoint is a set of clusters, where each cluster is distinct from each other cluster, and the objects within each cluster are broadly similar to each other.

In this algorithm, the hierarchy of clusters is developed in the form of a tree, and this tree-shaped structure is known as the dendrogram. To group the datasets into clusters, it follows the **bottom-up approach**. It means, this algorithm considers each dataset as a single cluster at the beginning, and then start combining the closest pair of clusters together. It does this until all the clusters are merged into a single cluster that contains all the datasets.

This algorithm starts with all the data points assigned to a cluster of their own. Then two nearest clusters are merged into the same cluster. In the end, this algorithm terminates when there is only a single cluster left. The results of hierarchical clustering can be shown using dendrogram. The dendrogram can be interpreted as:

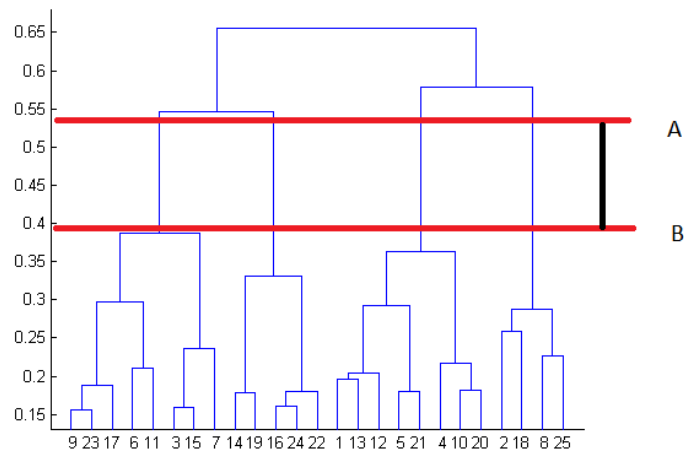


At the bottom, start with 25 data points, each assigned to separate clusters. Two closest clusters are then merged till we have just one cluster at the top. The height in the dendrogram

at which two clusters are merged represents the distance between two clusters in the data space.

The decision of the no. of clusters that can best depict different groups can be chosen by observing the dendrogram. The best choice of the no. of clusters is the no. of vertical lines in the dendrogram cut by a horizontal line that can transverse the maximum distance vertically without intersecting a cluster.

In the above example, the best choice of no. of clusters will be 4 as the red horizontal line in the dendrogram below covers maximum vertical distance AB.



In data mining and statistics, hierarchical clustering analysis is a method of cluster analysis which seeks to build a hierarchy of clusters i.e. tree type structure based on the hierarchy.

Basically, there are two types of hierarchical cluster analysis strategies

The hierarchical clustering technique has two approaches:

1. **Agglomerative:** Agglomerative is a **bottom-up** approach, in which the algorithm starts with taking all data points as single clusters and merging them until one cluster is left.
2. **Divisive:** Divisive algorithm is the reverse of the agglomerative algorithm as it is a **top-down approach**.

To group the datasets into clusters, it follows the **bottom-up approach**. It means, this algorithm considers each dataset as a single cluster at the beginning, and then start combining the closest pair of clusters together. It does this until all the clusters are merged into a single cluster that contains all the datasets.

4.3.3 Agglomerative Clustering

The agglomerative clustering is the most common type of hierarchical clustering used to group objects in clusters based on their similarity. Next, pairs of clusters are successively merged until all clusters have been merged into one big cluster containing all objects.

The step that Agglomerative Clustering take are:

1. Each data point is assigned as a single cluster.
2. Determine the distance measurement and calculate the distance matrix.
3. Determine the linkage criteria to merge the clusters.
4. Update the distance matrix.
5. Repeat the process until every data point become one cluster.

Algorithm for Agglomerative Hierarchical Clustering is: Calculate the **similarity** of one cluster with all the other clusters (calculate proximity matrix) Consider every data point as a individual cluster. ... Repeat Step 3 and 4 until only a single cluster remains

1. Agglomerative Clustering: Also known as bottom-up approach or hierarchical agglomerative clustering (HAC). A structure that is more informative than the unstructured set of clusters returned by flat clustering. This clustering algorithm does not require us to prespecify the number of clusters.

In data mining and statistics, hierarchical clustering analysis is a method of cluster analysis which seeks to build a hierarchy of clusters i.e. tree type structure based on the hierarchy.

for i=1 to N:

as the distance matrix is symmetric about

the primary diagonal so we compute only lower

part of the primary diagonal

for j=1 to i:

dis_mat[i][j] = distance[d_i, d_j]

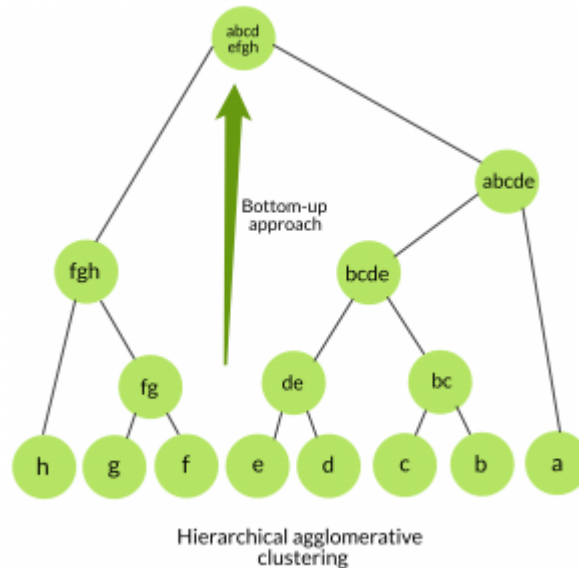
each data point is a singleton cluster

repeat

merge the two cluster having minimum distance

update the distance matrix

until only a single cluster remains



4.3.4 Divisive Clustering

The divisive clustering algorithm is a top-down clustering approach. Initially, all the points in the dataset belong to one cluster and split is performed recursively as one moves down the hierarchy

Steps of Divisive Clustering:

1. Initially, all points in the dataset belong to one single cluster.
2. Partition the cluster into two least similar cluster.
3. Proceed recursively to form new clusters until the desired number of clusters is obtained.

2. Divisive clustering: Also known as top-down approach. This algorithm also does not require to prespecify the number of clusters. Top-down clustering requires a method for splitting a cluster that contains the whole data and proceeds by splitting clusters recursively until individual data have been splitted into singleton cluster.

Algorithm :

given a dataset ($d_1, d_2, d_3, \dots, d_N$) of size N

at the top we have all data in one cluster

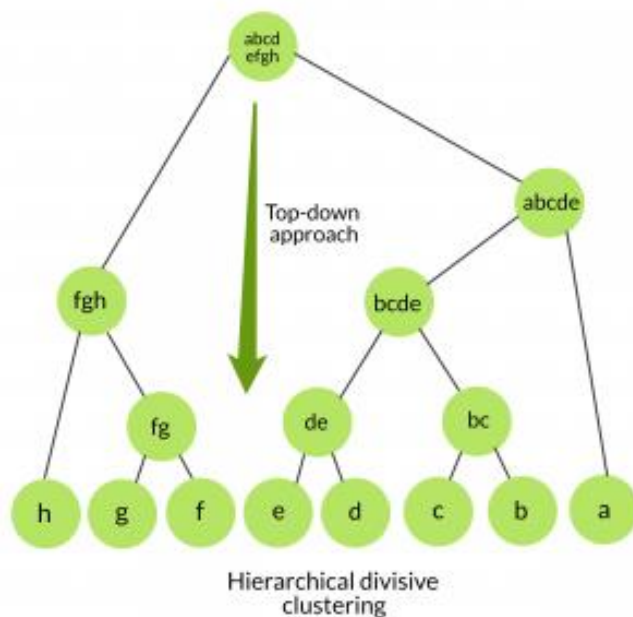
the cluster is split using a flat clustering method eg. K-Means etc

repeat

choose the best cluster among all the clusters to split

split that cluster by the flat clustering algorithm

until each data is in its own singleton cluster



Hierarchical Agglomerative vs Divisive clustering –

- Divisive clustering is more *complex* as compared to agglomerative clustering, as in case of divisive clustering we need a flat clustering method as “subroutine” to split each cluster until we have each data having its own singleton cluster.
- Divisive clustering is more *efficient* if we do not generate a complete hierarchy all the way down to individual data leaves. Time complexity of a naive agglomerative clustering is $O(n^3)$ because we exhaustively scan the $N \times N$ matrix $dist_mat$ for the lowest distance in each of $N-1$ iterations. Using priority queue data structure we can reduce this complexity to $O(n^2 \log n)$. By using some more optimizations it can be brought down to $O(n^2)$. Whereas for divisive clustering given a fixed number of top levels, using an efficient flat algorithm like K-Means, divisive algorithms are linear in the number of patterns and clusters.
- Divisive algorithm is also more *accurate*. Agglomerative clustering makes decisions by considering the local patterns or neighbor points without initially taking into

account the global distribution of data. These early decisions cannot be undone. whereas divisive clustering takes into consideration the global distribution of data when making top-level partitioning decisions.

4.4 EVALUATING CLUSTERS

Cluster Analysis in Data Mining means that to find out the group of objects which are similar to each other in the group but are different from the object in other groups

Here clusters are evaluated based on some similarity or dissimilarity measure such as the distance between cluster points. If the clustering algorithm separates dissimilar observations apart and similar observations together, then it has performed well

Typical objective functions in clustering formalize the goal of attaining high intra-cluster similarity (documents within a cluster are similar) and low inter-cluster similarity (documents from different clusters are dissimilar). This is an internal criterion for the quality of a clustering.

4.5 SUMMARY

A cluster is a subset of similar objects. A subset of objects such that the distance between any of the two objects in the cluster is less than the distance between any object in the cluster and any object that is not located inside it. A connected region of a multidimensional space with a comparatively high density of objects.

Clustering is an unsupervised Machine Learning-based Algorithm that comprises a group of data points into clusters so that the objects belong to the same group.

Clustering helps to splits data into several subsets. Each of these subsets contains data similar to each other, and these subsets are called clusters. Now that the data from our customer base is divided into clusters, we can make an informed decision about who we think is best suited for this product.

4.6 APPLICATIONS

In many applications, clustering analysis is widely used, such as data analysis, market research, pattern recognition, and image processing. It assists marketers to find different groups in their client base and based on the purchasing patterns. They can characterize their customer groups.

It helps in allocating documents on the internet for data discovery. Clustering is also used in tracking applications such as detection of credit card fraud. As a data mining function, cluster analysis serves as a tool to gain insight into the distribution of data to analyze the characteristics of each cluster.

It helps in the identification of areas of similar land that are used in an earth observation database and the identification of house groups in a city according to house type, value, and geographical location.

4.7 QUESTIONS

1. What is clustering data mining?
2. What are the three types of clusters?
3. How is clustering evaluated?
4. What are the different types of clustering algorithms?
5. What are the major tasks in clustering evaluation?
6. What is clustering assessment?

B.SC. (DATA SCIENCE)

SEMESTER III

DATA MINING AND VISUALIZATION

UNIT-V APPLICATIONS OF DATA MINING

STRUCTURE

5.0 Objective

5.1 Introduction

5.2 Business Applications using Data Mining

5.2.1 Risk Management and Targeted Marketing

5.2.2 Customer Profiles and Feature Construction

5.2.3 Medical Applications

5.2.4 Scientific Applications using Data Mining

5.2.5 Other Applications

5.3 Summary

5.4 Advantages of Data Mining in Healthcare

5.5 Challenges in Healthcare Data Mining

5.6 Practice Exercise

5.0 OBJECTIVE

To understand some application of data mining techniques

5.1 INTRODUCTION

Data mining is a process of extracting and discovering patterns in large data sets involving methods at the involving machine learning, statistics, and database systems. Data mining is an interdisciplinary subfield of computer science and statistics with an overall goal to extract information (with intelligent methods) from a data set and transform the information into a comprehensible structure for further use. Data mining is the analysis step of the "knowledge discovery in databases" process, or KDD. Aside from the raw analysis step, it also involves database and data management aspects, data preprocessing, model and inference considerations, interestingness metrics, complexity considerations, post-processing of discovered structures and visualization.

The goal of data mining is the extraction of patterns and knowledge from large amounts of data, not the extraction of data itself.

The actual data mining task is the semi-automatic or automatic analysis of large quantities of data to extract previously unknown, interesting patterns such as groups of data records (cluster analysis), unusual records (anomaly detection), and dependencies (association rule mining, sequential pattern mining).

Data Mining is mainly used today by companies with a strong consumer focus involving retail, financial, communication, and marketing to handle their transactional data and determine pricing, customer preferences and product positioning, impact on sales, customer satisfaction and corporate profits. With data mining, a retailer can use point-of-sale records of customer purchases to develop products and promotions to appeal to specific customer segments.

5.2 BUSINESS APPLICATIONS USING DATA MINING

Commercial databases often contain critical business information concerning past performance which could be used to predict the future.

5.2.1 Risk Management and Targeted Marketing,

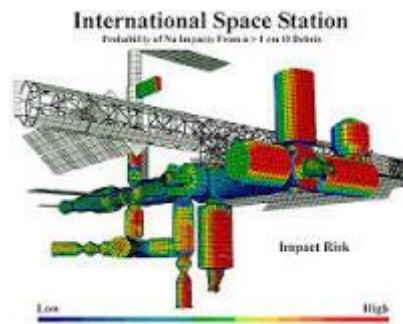
Risk Management is a logical and systematic method of identifying, analyzing, treating and monitoring the risks involved in any activity or process. The key to successful risk

management lies in the ability to modify a formal risk management process that addresses the needs of the systematically addressing risk throughout the product/project life-cycle. Risks can be introduced at the very earliest stages of the project life-cycle. The ability to identify risks earlier translates into earlier risk removal, at less cost, which promotes higher project success probability.

Once risks have been identified and assessed, all techniques to manage the risk fall into one or more of these four major categories:

- Avoidance (eliminate, withdraw from or not become involved)
- Reduction (optimize – mitigate)
- Sharing (transfer – outsource or insure)
- Retention (accept and budget)

Strategies to manage threats (uncertainties with negative consequences) typically include avoiding the threat, reducing the negative effect or probability of the threat, transferring all or part of the threat to another party, and even retaining some or all of the potential or actual consequences of a particular threat. The opposite of these strategies can be used to respond to opportunities (uncertain future states with benefits).



Example of risk assessment: A NASA model showing areas at high risk from impact for the International Space Station

5.2.2 Customer Profiles and Feature Construction

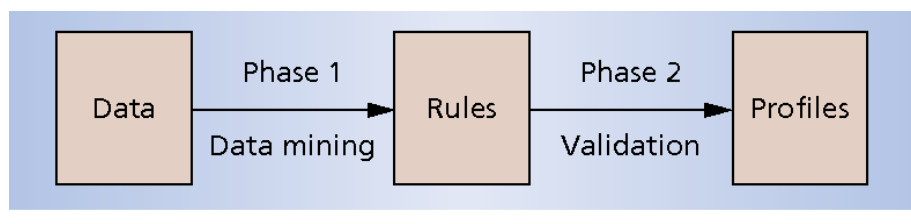
Customer profiling is a method to create a portrait of customers, which includes their personal and transactional details. It helps companies make various customer-centered decisions regarding their business. Customer Profiles or Personas are created with the help of customer research.

In a customer profile, purchasing behaviour of customer is identified, with the intent of targeting similar customers in the sales and marketing campaigns.

However, the huge amounts of data can make the extraction of this business information almost impossible by manual methods or standard software techniques. Data mining techniques can analyze, understand and visualize the huge amounts of stored data gathered from business applications and help stay competitive in today's marketplace.

How Customer Profiling Is Done

Customer profiling is done by breaking customers down into groups that share similar characteristics and goals. For example, if the goal is to purchase a smartphone online, there might not be too much of a difference in the habits of a single professional and a married business executive.



Building Customer profile

Systems construct personal profiles based on customer's transactional histories. System uses data mining techniques to discover a set of rules describing customer's behaviour.

5.2.3 Medical Applications

Data mining is the process of pattern discovery and extraction where huge amount of data is involved. Both the data mining and healthcare industry have emerged some of reliable early detection systems and other various healthcare related systems from the clinical and diagnosis data.

The best procedure for taking data mining beyond the rule of academic research is the three system approach. Implementing all three systems is the way to drive a real-world improvement with any analytics initiative in healthcare. Unfortunately, very few healthcare organizations execute all three of these systems.

These are the following three systems:

The Analytics System:

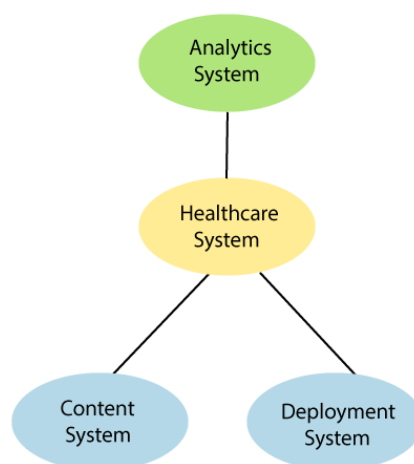
The analytics system incorporates the technology and expertise to accumulate information, comprehend it, and standardize measurements. Aggregating clinical, patient satisfaction, financial, and other data into an enterprise data warehouse (EDW) is the foundation of the system.

The content system:

The content system includes standardizing knowledge work. It applies evidence-based best practices to care delivery. Scientists make significant discoveries each year about clinical best practice, but it mentioned previously, it takes a long time for these discoveries to be incorporated into clinical practice. A strong content system enables organizations to put the latest medical conformation into practice quickly.

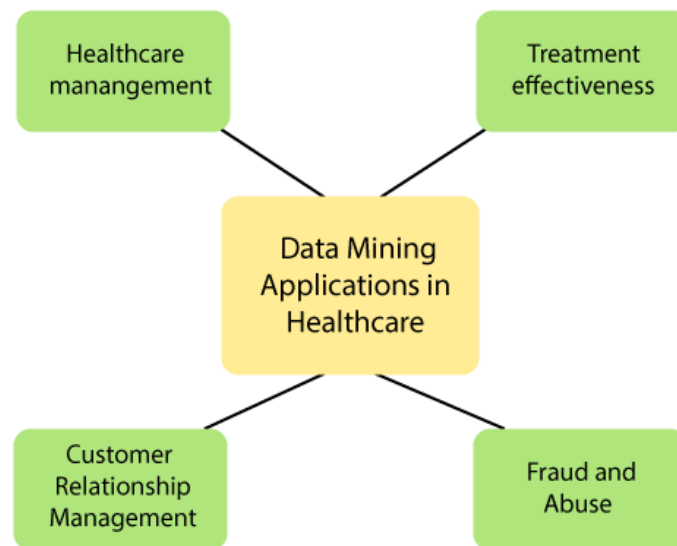
The deployment system:

The deployment system involves driving change management over new hierarchical structures. Particularly, it includes implementing group structures that empower consistently, enterprise-wide deployment of best practices. It requires a real hierarchical change to drive the adoption of best practices throughout an organization.



Application of Data Mining in Healthcare:

Data mining has been used intensively and widely by numerous industries. In healthcare, data mining is becoming more popular nowadays. Data mining applications can incredibly benefit all parties who are involved in the healthcare industry. For example, data mining can help the healthcare industry in fraud detection and abuse, customer relationship management, effective patient care, and best practices, affordable healthcare services. The large amounts of data generated by healthcare transactions are too complex and huge to be processed and analyzed by conventional methods. Data mining provides the framework and techniques to transform these data into useful information for data-driven decision purposes.



Treatment Effectiveness:

Data Mining applications can be used to assess the effectiveness of medical treatments. Data mining can convey analysis of which course of action demonstrates effective by comparing and differentiating causes, symptoms, and courses of treatments.

Healthcare Management:

Data mining applications can be used to identify and track chronic illness states and incentive care unit patients, decrease the number of hospital admissions, and supports healthcare management. Data mining used to analyze massive data sets and statistics to search for patterns that may demonstrate an assault by bio-terrorists.

Customer relationship management:

Customer and management interactions are very crucial for any organization to achieve business goals. Customer relationship management is the primary approach to managing interactions between commercial organizations normally retail sectors and banks, with their customers. Similarly, it is important in the healthcare context. Customer interactions may happen through call centers, billing departments, and ambulatory care settings.

Fraud and abuse:

Data mining fraud and abuse applications can focus on inappropriate or wrong prescriptions and fraud insurance and medical claims.

Results of comparative analysis of various disease in Healthcare:

A comparative analysis of data mining applications in the healthcare sector by various specialists has given in detail. Primarily data mining tools are used to predict the results from the information recorded on healthcare problems. Various data mining tools are utilized to predict the precision level in different healthcare problems. In the given list of medical problems have been examined and evaluated.

Diabetic screening

Diabetes is a major health problem nowadays. Medical applications (diabetic screening), shows that decision tree analyses can be applied to screen individuals for early diabetes risk without the need for offensive tests. This procedure will be particularly useful in developing regions with high epidemiological risk and poor socioeconomic status, and enable clinical practitioners to rapidly screen patients for increased risk of diabetes. The key features in the tree structure could further facilitate diabetes prevention through targeted community interventions, which can improve early diabetes diagnosis and reduce burdens on the healthcare system.

In the present investigation, regression based data mining techniques are applied to diabetes data. The goal of this investigation is to use data mining techniques to discover patterns that identify the best mode of treatment for diabetes across different age groups.

This research concentrates upon predictive analysis of diabetic treatment using a regression-based data mining technique. The Oracle Data Miner (ODM) was employed as a software mining tool for predicting modes of treating diabetes. The support vector machine algorithm

was used for experimental analysis. The dataset was studied and analyzed to identify effectiveness of different treatment types for different age groups.. Preferential orders of treatment were investigated. It was concluded that drug treatment for patients in the young age group can be delayed to avoid side effects. In contrast, patients in the old age group should be prescribed drug treatment immediately, along with other treatments, because there are no other alternatives available.

5.2.4 Scientific Applications using Data Mining

An application that simulates real-world activities using mathematics, scientific applications turn real-world objects into mathematical models, and their actions are simulated by executing the required formulas. For example, an airplane's flight characteristics can be simulated in the computer.

5.2.5 Other Applications

Data Mining is primarily used today by companies with a strong consumer focus — retail, financial, communication, and marketing organizations, to bring into their transactional data and determine pricing, customer preferences and product positioning, impact on sales, customer satisfaction and corporate profits. With data mining, a retailer can use point-of-sale records of customer purchases to develop products and promotions to appeal to specific customer segments.

Here is the list of 14 other important areas where data mining is widely used:

1. Future Healthcare

Data mining holds great potential to improve health systems. It uses data and analytics to identify best practices that improve care and reduce costs. Researchers use data mining approaches like multi-dimensional databases, machine learning, soft computing, data visualization and statistics. Mining can be used to predict the volume of patients in every category. Processes are developed that make sure that the patients receive appropriate care at the right place and at the right time. Data mining can also help healthcare insurers to detect fraud and abuse.

2. Market Basket Analysis

Market basket analysis is a modelling technique based upon a theory that if you buy a certain group of items you are more likely to buy another group of items. This technique may allow

the retailer to understand the purchase behaviour of a buyer. This information may help the retailer to know the buyer's needs and change the store's layout accordingly. Using differential analysis comparison of results between different stores, between customers in different demographic groups can be done.

3. Education

There is a new emerging field, called Educational Data Mining, concerns with developing methods that discover knowledge from data originating from educational Environments. The goals of EDM are identified as predicting students' future learning behaviour, studying the effects of educational support, and advancing scientific knowledge about learning. Data mining can be used by an institution to take accurate decisions and also to predict the results of the student. With the results the institution can focus on what to teach and how to teach. Learning pattern of the students can be captured and used to develop techniques to teach them.

4. Manufacturing Engineering

Knowledge is the best asset a manufacturing enterprise would possess. Data mining tools can be very useful to discover patterns in complex manufacturing process. Data mining can be used in system-level designing to extract the relationships between product architecture, product portfolio, and customer needs data. It can also be used to predict the product development span time, cost, and dependencies among other tasks.

5. CRM

Customer Relationship Management is all about acquiring and retaining customers, also improving customers' loyalty and implementing customer focused strategies. To maintain a proper relationship with a customer a business need to collect data and analyse the information. This is where data mining plays its part. With data mining technologies the collected data can be used for analysis. Instead of being confused where to focus to retain customer, the seekers for the solution get filtered results.

6. Fraud Detection

Billions of dollars have been lost to the action of frauds. Traditional methods of fraud detection are time consuming and complex. Data mining aids in providing meaningful patterns and turning data into information. Any information that is valid and useful is knowledge. A perfect fraud detection system should protect information of all the users. A

supervised method includes collection of sample records. These records are classified fraudulent or non-fraudulent. A model is built using this data and the algorithm is made to identify whether the record is fraudulent or not.

7. Intrusion Detection

Any action that will compromise the integrity and confidentiality of a resource is an intrusion. The defensive measures to avoid an intrusion includes user authentication, avoid programming errors, and information protection. Data mining can help improve intrusion detection by adding a level of focus to anomaly detection. It helps an analyst to distinguish an activity from common everyday network activity. Data mining also helps extract data which is more relevant to the problem.

8. Lie Detection

Apprehending a criminal is easy whereas bringing out the truth from him is difficult. Law enforcement can use mining techniques to investigate crimes, monitor communication of suspected terrorists. This field includes text mining also. This process seeks to find meaningful patterns in data which is usually unstructured text. The data sample collected from previous investigations are compared and a model for lie detection is created. With this model processes can be created according to the necessity.

9. Customer Segmentation

Traditional market research may help us to segment customers but data mining goes in deep and increases market effectiveness. Data mining aids in aligning the customers into a distinct segment and can tailor the needs according to the customers. Market is always about retaining the customers. Data mining allows to find a segment of customers based on vulnerability and the business could offer them with special offers and enhance satisfaction.

10. Financial Banking

With computerised banking everywhere huge amount of data is supposed to be generated with new transactions. Data mining can contribute to solving business problems in banking and finance by finding patterns, causalities, and correlations in business information and market prices that are not immediately apparent to managers because the volume data is too large or is generated too quickly to screen by experts. The managers may find these information for better segmenting, targeting, acquiring, retaining and maintaining a profitable customer.

11. Corporate Surveillance

Corporate surveillance is the monitoring of a person or group's behaviour by a corporation. The data collected is most often used for marketing purposes or sold to other corporations, but is also regularly shared with government agencies. It can be used by the business to tailor their products desirable by their customers. The data can be used for direct marketing purposes, such as the targeted advertisements on Google and Yahoo, where ads are targeted to the user of the search engine by analyzing their search history and emails.

12. Research Analysis

History shows that we have witnessed revolutionary changes in research. Data mining is helpful in data cleaning, data pre-processing and integration of databases. The researchers can find any similar data from the database that might bring any change in the research. Identification of any co-occurring sequences and the correlation between any activities can be known. Data visualisation and visual data mining provide us with a clear view of the data.

13. Criminal Investigation

Criminology is a process that aims to identify crime characteristics. Actually crime analysis includes exploring and detecting crimes and their relationships with criminals. The high volume of crime datasets and also the complexity of relationships between these kinds of data have made criminology an appropriate field for applying data mining techniques. Text based crime reports can be converted into word processing files. These information can be used to perform crime matching process.

14. Bio Informatics

Data Mining approaches seem ideally suited for Bioinformatics, since it is data-rich. Mining biological data helps to extract useful knowledge from massive datasets gathered in biology, and in other related life sciences areas such as medicine and neuroscience. Applications of data mining to bioinformatics include gene finding, protein function inference, disease diagnosis, disease prognosis, disease treatment optimization, protein and gene interaction network reconstruction, data cleansing, and protein sub-cellular location prediction.

5.3 SUMMARY

Data mining is the process of pattern discovery and extraction where huge amount of data is involved. Both the data mining and healthcare industry have emerged some of reliable early

detection systems and other various healthcare related systems from the clinical and diagnosis data.

Various sectors effectively use data mining. It enables the retail sectors to display customer response and helps the banking sector to predict customer profitability. It serves many similar sectors such as manufacturing, telecom, healthcare, automotive industry, education, and many more. Data mining holds incredible potential for healthcare services due to the exponential growth in the number of electronic health records. Previously Doctors and physicians hold patient information in the paper where the data was quite difficult to hold. Digitalization and innovation of new techniques reduce human efforts and make data easily assessable. For example, the computer keeps a massive amount of patient data with accuracy, and it improves the quality of the whole data management system. Still, the major challenge is what should healthcare services providers do to filter all the data efficiently? This is the place where data mining has proven to be extremely useful.

The best procedure for taking data mining beyond the rule of academic research is the three system approach. Implementing all three systems is the way to drive a real-world improvement with any analytics initiative in healthcare. Unfortunately, very few healthcare organizations execute all three of these systems.

5.4 ADVANTAGES OF DATA MINING IN HEALTHCARE:

The data framework simplifies and automates the workflow of health care institutions. Integration of data mining in data frameworks, healthcare institutions reduce decision-making effort and provide new valuable medical knowledge. Predictive models give the best information support and knowledge to healthcare workers. The objective of predictive data mining in medicine is to build up a predictive model that is clear, provides reliable predictions, supports doctors to improve their diagnosis and treatment planning processes. An essential application of data mining is for biomedical signal processing communicated by internal guidelines and reactions to boost the condition, whenever there is a lack of knowledge about the connection between various subsystems, and when the standard analysis methods are ineffective, as it is often in the case of nonlinear associations

5.5 CHALLENGES IN HEALTHCARE DATA MINING:

One of the biggest issues in data mining in healthcare is that the raw medical data is huge and heterogeneous. These data can be accumulated from different sources. For example, from conversations with patients, doctors review, and laboratory results. All these components can have a significant effect on the diagnosis, and treatment of a patient. Missing, incorrect, inconsistent data such as pieces of information saved in various formats from different data sources create a significant obstacle to successful data mining.

Another challenge is that almost all diagnoses and treatments in healthcare are inaccurate and subject to error rates. Here the analysis of specificity and sensitivity are being considered for the measurements of these errors. Within the issue of knowledge integrity evaluation, two major challenges are:

How to create effective algorithms for differentiating the content of two versions (after and before)?

How to create algorithms for evaluating the impact of specific data modifications on the statistical significance of individual patterns, which is collected with the assistance of basic classes of data mining algorithm?

Algorithms that measure the impact that modifications of data values have on the discovered statistical significance of patterns are being created, despite the fact that it is difficult to build up universal measures for all data mining algorithms.

The challenges of Data Mining are as given below:

- Security and Social Challenges.
- Noisy and Incomplete Data.
- Distributed Data.
- Complex Data.
- Performance.
- Scalability and Efficiency of the Algorithms.
- Improvement of Mining Algorithms.
- Incorporation of Background Knowledge.

5.6 PRACTICE QUESTIONS

What is risk taking in marketing management?

What is medical data mining?

What is medical data mining?

What are the data mining applications?

What is the most common application of data mining?

Is data mining used in healthcare?

B.SC. (DATA SCIENCE)
SEMESTER III
DATA MINING AND VISUALIZATION

UNIT-VI DATA VISUALIZATION

STRUCTURE

6.0 Objective

6.1 Introduction

6.2 Acquiring and Visualizing Data

6.3 Simultaneous Acquisition and Visualization

6.4 Applications of Data Visualization

6.5 Keys factors of Data Visualization

6.5.1 Control of Presentation

6.5.2 Faster and Better JavaScript processing

6.5.3 Rise of HTML5

6.5.4 Lowering the Implementation Bar

6.6 Summary

6.7 Practice Question

6.0 OBJECTIVE

To understand Data visualization and the rise of HTML 5

6.1 INTRODUCTION

Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends and patterns in data.

Data visualization is an interdisciplinary field and is a particularly efficient way of communicating when the data is numerous as for example a time series. From an academic point of view, this representation can be considered as a mapping between the original data (usually numerical) and graphic elements (for example, lines or points in a chart). The mapping determines how the attributes of these elements vary according to the data. In this light, a bar chart is a mapping of the length of a bar to a magnitude of a variable. Since the graphic design of the mapping can adversely affect the readability of a chart, mapping is a core competency of Data visualization. Data visualization has its roots in the field of Statistics and is therefore generally considered a branch of Descriptive Statistics. However, because both design skills and statistical and computing skills are required to visualize effectively, it is argued by some authors that it is both an Art and a Science.

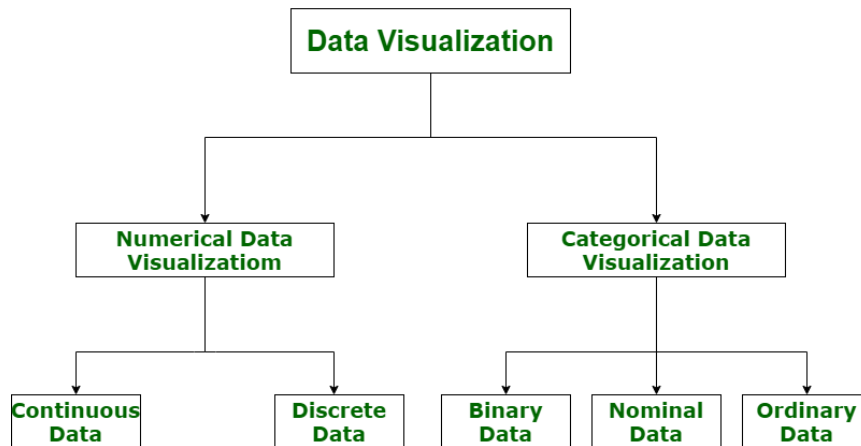
Research into how people read and misread various types of visualizations is helping to determine what types and features of visualizations are most understandable and effective in conveying information. Our eyes are attracted to colours and patterns. We can quickly identify red from blue, square from circle. Our culture is visual, including everything from art and advertisements to TV and movies. Data visualization is another form of visual art that takes our interest and keeps our eyes on the message. When we see a chart, we quickly see trends and outliers. If we can see something, we internalize it quickly. It's storytelling with a purpose. If you've ever stared at a massive spreadsheet of data and couldn't see a trend, you know how much more effective a visualization can be.

In the "age of Big Data", visualization is an increasingly key tool to make sense of the trillions of rows of data generated every day. Data visualization helps to tell stories by curating data into a form easier to understand, highlighting the trends. A good visualization tells a story, removing the noise from data and highlighting the useful information. Effective data visualization is a delicate balancing act between form and function. The plainest graph could be too boring to catch any notice or it could make a powerful point; the most stunning visualization could utterly fail at conveying the right message or it could speak volumes. The data and the visuals need to work together, and there's an art to combining great analysis with

It's hard to think of a professional industry that doesn't benefit from making data more understandable. Every field benefits from understanding data, and so do fields in government, finance, marketing, history, consumer goods, service industries, education, sports, and so on. While traditional education typically draws a distinct line between creative storytelling and

technical analysis, the modern professional world also values those who can cross between the two: data visualization sits right in the middle of analysis and visual storytelling.

Due to an increase in statistical data, visual representation of that data is preferred rather than going through spreadsheets. It is easy to understand as well as it saves time, which is of extreme importance. The trends can be easily identified and studied in a graphical representation. Data visualization is mostly used for business analytics. For almost every business, data visualization is used, because they have large data. For the analysis of the data, visuals are the most efficient.



Categories of Data Visualization

6.2 ACQUIRING AND VISUALIZING DATA.

Data viz is the communication of data in a visual manner, or turning raw data into insights that can be easily interpreted by your readers. Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data. In the world of Big Data, data visualization tools and technologies are essential to analyze massive amounts of information and make data-driven decisions.



The first step in visualizing data is to load it into your application. Typical data sources might be a file on a disk, a stream from a network, or a digitized signal (e.g., audio or sensor readings). The need of immense data is because of various reasons like processing of weeks of data related to surveillance video, one quantitatively acquire data from an hour-long meeting that involved a verbal discussion, drawings on a whiteboard, and note taking done by individual participants?

Thus, the acquisition stage covers several tasks:

The seven stages of visualizing data

Step 1: Define a clear purpose.

Step 2: Know your audience.

Step 3: Keep visualizations simple.

Step 4: Choose the right visual.

Step 5: Make sure your visualizations are inclusive.

Step 6: Provide context.

Step 7: Make it actionable.

There are two kinds Visualization:

1. Interactive Visualization

These days when technological inventions are affecting every market segment, regardless of big or small, interactive visualization needs to be controlled for different portions of charts and graphs to obtain a more detailed analysis of the information being presented.

2. Intuitive Visualization

Data visualization is interactive. Another big picture is available on a click to get a particular information segment. They are also tailored according to the target audience and could be easily updated if the information modifies.

6.3 SIMULTANEOUS ACQUISITION AND VISUALIZATION

Simultaneous acquisition and visualization is needed as millions of visitors take different paths through a website, of a few hundred thousand files on your computer's hard disk, which ones are taking up the most space, and how often do you use them? By applying methods from the fields of computer science, statistics, data mining, graphic design, and visualization, we can begin to answer these questions in a meaningful way that also makes the answers accessible to others.

The problem is increased by the continually changing nature of data, which can result from new information being added or older information continuously being refined. This flood of data necessitates new software-based tools, and its complexity requires extra consideration.

Each set of data has particular display needs, and the purpose for which it is being used, the data set has effect on those needs as the data itself. There are dozens of quick tools for developing graphics in office programs, on the Web, and elsewhere, but complex data sets used for specialized applications require unique treatment. The characteristics of a data set help determine what kind of visualization is used.

Data acquisition is the **process that serializes the data**, thereby producing data values in a sequence. Visualization is a powerful tool for the qualitative analysis of GCxGC data (e.g., to troubleshoot the chromatography). Various types of visualizations are useful.

Visualization is a powerful tool for qualitative analysis of GC"GC data (e.g., to troubleshoot the chromatography). Various types of visualizations are useful: two-dimensional images provide a comprehensive overview, three-dimensional visualizations effectively illustrate quantitative relationships over a large dynamic range, one-dimensional graphs are useful for overlaying multivariate data, tabular views reveal the numeric values in the data, and graphical and text annotations communicate additional information. This section explores some of the methods and considerations in the various types of visualizations.

Acquire: Obtain the data, whether from a file on a disk or a source over a network.

Parse: Provide some structure for the data's meaning, and order it into categories. The amount of Data one can collect and analyze is immense. It is necessary to put the data you collect it into a structure. This structure will make it easier to know convey to others what data you have by format, tags, names, and indices.

Filter: Remove all but the data of interest. After putting the data into a structure. You will have to filter out the data that is not necessary for your Data Visualization.

Mine: Apply methods from statistics or data mining as a way to discern patterns or place the data in mathematical context. The focus will be on basic statistics in the beginning. More emphasis in discovering patterns will occur in later sections. This step helps get basic understanding of the data before doing the representational step.

Represent: Choose a basic visual model, such as a bar graph, list, or tree. Focus on how to choose the basic visual model as well as how to represent it.

Refine: Improve the basic representation to make it clearer and more visually engaging.

Interact: Add methods for manipulating the data or controlling what features are visible.

6.4 APPLICATIONS OF DATA VISUALIZATION,

Data mining is widely used in diverse areas. There are a number of commercial data mining system available today and yet there are many challenges in this field. Data Visualization Application enables users to visualize data, draw insights and understand it better. It allows people to organize and present information intuitively. Visualizations can be combined into a dashboard. Dashboards are useful because they allow you to relate different views of information visually.



Data Mining Applications

Here is the list of areas where data mining is widely used –

- Financial Data Analysis
- Retail Industry
- Telecommunication Industry
- Biological Data Analysis
- Other Scientific Applications
- Intrusion Detection

Financial Data Analysis

The financial data in banking and financial industry is generally reliable and of high quality which facilitates systematic data analysis and data mining. Some of the typical cases are as follows –

- Design and construction of data warehouses for multidimensional data analysis and data mining.
- Loan payment prediction and customer credit policy analysis.
- Classification and clustering of customers for targeted marketing.
- Detection of money laundering and other financial crimes.

Retail Industry

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. It is natural that the quantity of data collected will continue to expand rapidly because of the increasing ease, availability and popularity of the web.

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. Here is the list of examples of data mining in the retail industry –

- Design and Construction of data warehouses based on the benefits of data mining.
- Multidimensional analysis of sales, customers, products, time and region.
- Analysis of effectiveness of sales campaigns.
- Customer Retention.
- Product recommendation and cross-referencing of items.

Telecommunication Industry

Today the telecommunication industry is one of the most emerging industries providing various services such as fax, pager, cellular phone, internet messenger, images, e-mail, web data transmission, etc. Due to the development of new computer and communication technologies, the telecommunication industry is rapidly expanding. This is the reason why data mining is become very important to help and understand the business.

Data mining in telecommunication industry helps in identifying the telecommunication patterns, catch fraudulent activities, make better use of resource, and improve quality of service. Here is the list of examples for which data mining improves telecommunication services –

- Multidimensional Analysis of Telecommunication data.
- Fraudulent pattern analysis.
- Identification of unusual patterns.
- Multidimensional association and sequential patterns analysis.
- Mobile Telecommunication services.
- Use of visualization tools in telecommunication data analysis.

Biological Data Analysis

In recent times, we have seen a tremendous growth in the field of biology such as genomics, proteomics, functional Genomics and biomedical research. Biological data mining is a very important part of Bioinformatics. Following are the aspects in which data mining contributes for biological data analysis –

- Semantic integration of heterogeneous, distributed genomic and proteomic databases.
- Alignment, indexing, similarity search and comparative analysis multiple nucleotide sequences.

- Discovery of structural patterns and analysis of genetic networks and protein pathways.
- Association and path analysis.
- Visualization tools in genetic data analysis.

Other Scientific Applications

The applications discussed above tend to handle relatively small and homogeneous data sets for which the statistical techniques are appropriate. Huge amount of data have been collected from scientific domains such as geosciences, astronomy, etc. A large amount of data sets is being generated because of the fast numerical simulations in various fields such as climate and ecosystem modeling, chemical engineering, fluid dynamics, etc. Following are the applications of data mining in the field of Scientific Applications –

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

Intrusion Detection

Intrusion refers to any kind of action that threatens integrity, confidentiality, or the availability of network resources. In this world of connectivity, security has become the major issue. With increased usage of internet and availability of the tools and tricks for intruding and attacking network prompted intrusion detection to become a critical component of network administration. Here is the list of areas in which data mining technology may be applied for intrusion detection –

- Development of data mining algorithm for intrusion detection.
- Association and correlation analysis, aggregation to help select and build discriminating attributes.
- Analysis of Stream data.
- Distributed data mining.
- Visualization and query tools.

Data Mining System Products

There are many data mining system products and domain specific data mining applications. The new data mining systems and applications are being added to the previous systems. Also, efforts are being made to standardize data mining languages.

Choosing a Data Mining System

The selection of a data mining system depends on the following features –

- **Data Types** – The data mining system may handle formatted text, record-based data, and relational data. The data could also be in ASCII text, relational database data or data warehouse data. Therefore, we should check what exact format the data mining system can handle.
- **System Issues** – We must consider the compatibility of a data mining system with different operating systems. One data mining system may run on only one operating system or on several. There are also data mining systems that provide web-based user interfaces and allow XML data as input.
- **Data Sources** – Data sources refer to the data formats in which data mining system will operate. Some data mining system may work only on ASCII text files while others on multiple relational sources. Data mining system should also support ODBC connections or OLE DB for ODBC connections.
- **Data Mining functions and methodologies** – There are some data mining systems that provide only one data mining function such as classification while some provides multiple data mining functions such as concept description, discovery-driven OLAP analysis, association mining, linkage analysis, statistical analysis, classification, prediction, clustering, outlier analysis, similarity search, etc.
- **Coupling data mining with databases or data warehouse systems** – Data mining systems need to be coupled with a database or a data warehouse system. The coupled components are integrated into a uniform information processing environment. Here are the types of coupling listed below
 - No coupling
 - Loose Coupling
 - Semi tight Coupling
 - Tight Coupling
- **Scalability** – There are two scalability issues in data mining –
 - **Row (Database size) Scalability** – A data mining system is considered as row scalable when the number or rows are enlarged 10 times. It takes no more than 10 times to execute a query.

6.5 KEY FACTORS OF DATA VISUALIZATION

6.5.1 Control of Presentation

Visualization is the first step to make sense of data. To translate and present data and data correlations in a simple way, data analysts use a wide range of techniques. Some of them are charts, diagrams, maps, etc. Choosing the right technique and its setup is often the only way to make data understandable.

Five factors that influence data visualization choices:

1. Audience

Adjust data representation to the specific target audience. For example, fitness mobile app users who browse through their progress can easily work with uncomplicated visualizations. On the other hand, if data visualizations are intended for researchers and experienced decision-makers who regularly work with data,

2. Content

The type of data to be presented will determine the strategies. For example, if it's time-series metrics, you will use line charts to show the dynamics in many cases. To show the relationship between two elements, scatter plots are often used. In turn, bar charts work well for comparative analysis.

3. Context

You can use different data visualization approaches and read data depending on the context. To emphasize a certain figure, for example, significant profit growth, you can use the shades of one color on the chart and highlight the highest value with the brightest one.

4. Dynamics

There are various types of data, and each type has a different rate of change. For example, financial results can be measured monthly or yearly, while time series and tracking data are changing constantly. Depending on the rate of change, you may consider dynamic representation (streaming) or static visualization techniques in data mining.

5. Purpose

The goal of data visualization affects the way it is implemented. In order to make a complex analysis, visualizations are compiled into dynamic and controllable dashboards that work as visual data analysis techniques and tools. However, dashboards are not necessary to show a single or occasional data insight.

Depending on these factors, different data visualization techniques can be used. Here are the common types of visualization techniques:

- **Charts**

The easiest way to show the development of one or several data sets is a chart. Charts vary from bar and line charts that show the relationship between elements over time.

- **Plots**

Plots allow to distribute two or more data sets over a 2D or even 3D space to show the relationship between these sets and the parameters on the plot.

- **Maps**

They allow to locate elements on relevant objects and areas — geographical maps, building plans, website layouts, etc.

- **Diagrams and matrices**

Diagrams are usually used to demonstrate complex data relationships and links and include various types of data on one visualization.

Faster and Better JavaScript processing,

For a Java Script (JS) developer, the ability to visualize data is just as valuable as making interactive Web pages. Especially that the two often go in pairs. As JavaScript continues to gain popularity in data visualization domain, the market is flooded with even new libraries with which to create beautiful charts for the Web

Data is visualized in JavaScript by using the following method:

Write the Code

1. Build the HTML.
2. Understand the Data.
3. JavaScript to Load the Data.
4. Understand the Algorithm.
5. Build the Data Table with JavaScript.
6. Add the Data to the Table with JavaScript.
7. Add the Color Legend.
8. Style the Visualization with CSS.

The form of the visualization depends on the relationship between the chosen graphical elements or types and the data points involved. The end goal is to communicate information derived from data sources clearly and effectively via visual or graphical means.

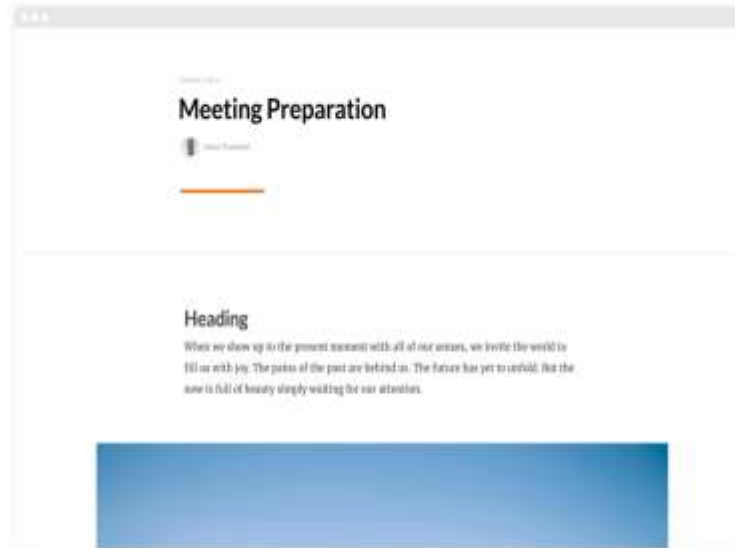
Following are some JavaScript's data visualization libraries

- Highcharts
- Toast UI Chart
- D3.js
- Recharts
- Chart.js

6.5.2 Rise of HTML5

Rise HTML is a **Learning Journal** that allows the learner to enter text responses to journal prompts throughout a Rise course and at the end print their learning journal of all their responses. If you are familiar with Javascript, you can build on it. Learning Journal in Rise. Rise 360 is a modern, dynamic eLearning authoring tool allowing designers to create responsive courses for any device. Using a web-based course builder, Rise allows instructional designers to create beautiful online courses with a few clicks of a button.

Rise training is the all-in-one training system that makes training easy to create, enjoyable to take, and simple to manage. Taking a closer look at why Rise is all that is needed to create, distribute, track, and manage online training people actually love.



6.6 SUMMARY

The types of data Visualisation are Column Chart. This is one of the most common types of data visualization tools. Bar Graph, Stacked Bar Graph, Line Graph, Dual-Axis Chart, Pie Chart, Scatter Plot.

Data Visualization process is a series of steps:

Acquire, Parse, Filter, Mine, Represent, Refine, Interact

Data mining concepts are still evolving and here are the latest trends that we get to see in this field –Application Exploration, Scalable and interactive data mining methods, Integration of data mining with database systems, data warehouse systems and web database systems, Visual data mining, New methods for mining complex types of data, Web mining, Distributed data mining, Real time data mining.

6.7 Practice Exercise

1. What is data visualization?
2. What are data visualization techniques?
3. Where is data visualization used?
4. What is acquiring data and visualize data?
5. What are the seven stages of visualizing data?

B.SC. (DATA SCIENCE)
SEMESTER III
DATA MINING AND VISUALIZATION

UNIT-VII EXPLORING THE VISUAL DATA SPECTRUM

STRUCTURE

7.0 Objective

7.1 Introduction

7.2 Charting Primitives

7.2.1 Data Points

7.2.2 Line Charts

7.2.3 Bar Charts

7.2.4 Pie Charts

7.2.5 Area Charts

7.3 Exploring advanced Visualizations

7.3.1 Candlestick Charts

7.3.2 Bubble Charts

7.3.3 Surface Charts

7.3.4 Map Charts

7.3.5 Infographics

7.4 Summary

7.5 Question

7.0 OBJECTIVE

To Explore advanced visual data analysis with the help of charts and other advanced techniques

7.1 INTRODUCTION

Visual communication is the use of visual elements to convey ideas and information' which include but are not limited to, signs, typography, drawing, graphic design, illustration, industrial design, advertising, animation, and electronic resources. Humans have used visual communication since prehistoric times. Within modern culture, there are several types of characteristics when it comes to visual elements, they consist of objects, models, graphs, diagrams, maps and photographs. Outside the different types of characteristics and elements, there are seven components of visual communication: Color, Shape, Tones, Texture, Figure-Ground, Balance, and Hierarchy.

Each of these characteristics, elements, and components play an important role in daily lives. In considering these different aspects, visual elements present various uses and how they convey information. Whether it is advertisements, teaching and learning, or speeches and presentations, they all involve visual aids that communicate a message.

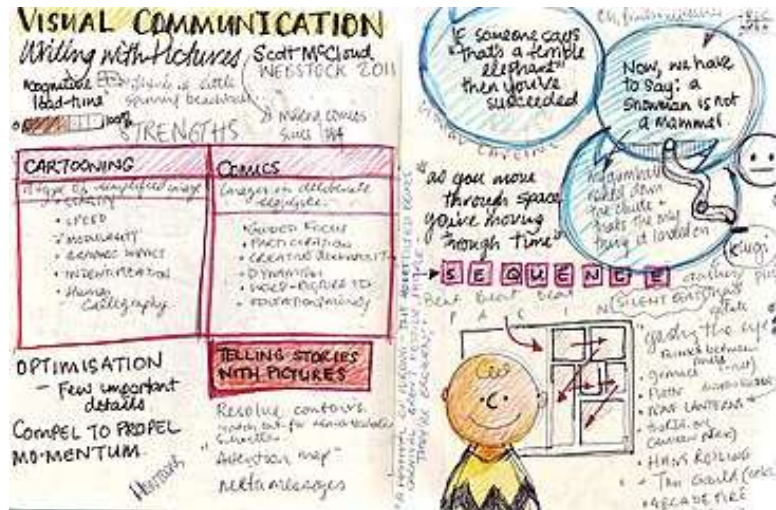


Image showing the visual communication process

Visualizations are tools that can make complex concepts easier for humans to understand. It allows for new, previously-impossible ways of thinking, of living, of being.” The utility of data visualization can be divided into three main goals: to **explore**, to monitor, and to explain. While some visualizations can span more than one of these, most focus on a single goal. When users are looking for an open-ended tool that helps them to find patterns and insights in

data, a data visualization focused on exploration and fast iteration can help. Exploration tools should have strong connections to other tools that collect (extract), clean (transform), and curate (load) data. When users need to check on the performance of something, a data visualization focused on monitoring is best. Research shows that immense data is created every day. In fact, 90% of the world's data was generated in the past two years alone! With so much information accessible all the time, it needs to be understood how to organize it into analyzable, actionable insights. Therefore, the best data visualization types to use should be known. Data visualization is the process of turning your data into graphical representations that communicate logical relationships and lead to more informed decision-making. In short, data visualization is the representation of data in a graphical or pictorial form

7.2 CHARTING PRIMITIVES

Data visualization is defined as a graphical representation that contains the information and the data. By using visual elements like charts, graphs, and maps, data visualization techniques provide an accessible way to see and understand trends, outliers, and patterns in data.

Visualization is any technique for creating images, diagrams, or animations to communicate a message. Visualization through visual imagery has been an effective way to communicate both abstract and concrete ideas since the dawn of humanity.

A chart is a graphical representation for data visualization, in which the data is represented by symbols, such as bars in a bar chart, lines in a line chart, or slices in a pie chart". A chart can represent tabular numeric data, functions or some kinds of quality structure and provides different info.

7.2 1 Data Points

Data Point is a single value located in a worksheet cell plotted in a chart or graph A column, dot, pie slice, or another symbol in the chart representing a data value. For example, in a line graph, each point on the line is a data marker representing a single data value located in a worksheet cell.20-

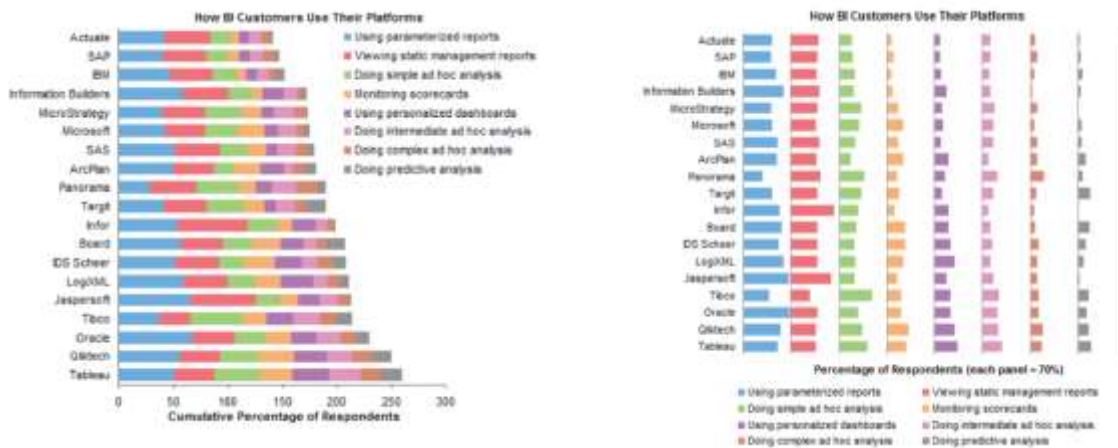
A data point is one piece of data appearing on the chart. For most chart types, each data point shows the value of the contents of one cell in the data range linked to the chart. A chart's details show what kind of data is represented in the chart.

How many data points should be on a graph?

The average of the repeated measures is then reported and used for the development of a graph if appropriate. How many data points are enough?: As an answer to the initial question, a simple and fast rule for introductory labs would be to collect 6 data points minimum.

Essential Elements of Good Graphs:

- A title which describes the experiment. ...
- The graph should fill the space allotted for the graph. ...
- Each axis should be labeled with the quantity being measured and the units of measurement. ...
- Each data point should be plotted in the proper position. ...
- A line of best fit.



Choose the best types of chart for your data

7.2.2 Line Charts

Line charts are the most effective chart for displaying time series data. They can handle a ton of data points and multiple data series, and everyone knows how to read them. A line chart is a graphical representation of an asset's historical price action that connects a series of data points with a continuous line. This is the most basic type of chart used in finance, and it typically depicts a security's closing prices over time. A line chart is used to show the change of data over a continuous time interval or time span. It is characterized by a tendency to reflect things as they change over time or ordered categories.



Line Chart

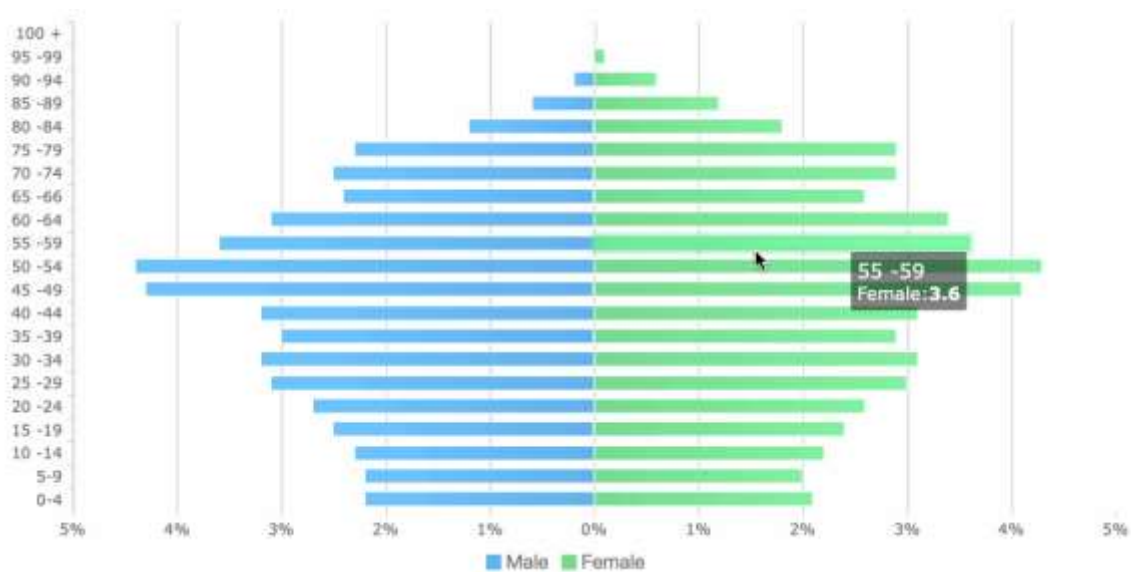
7.2.3 Bar Charts

A bar chart or bar graph is a chart or graph that presents categorical data with rectangular bars with heights or lengths proportional to the values that they represent. The bars can be plotted vertically or horizontally. A vertical bar chart is sometimes called a **column chart**.

A bar graph shows comparisons among discrete categories. One axis of the chart shows the specific categories being compared, and the other axis represents a measured value. Some bar graphs present bars clustered in groups of more than one, showing the values of more than one measured variable.

Bar charts use vertical columns to show numerical comparisons between categories, and the number of bars can be relatively large., the positions of its two axes are changed.

Application Scenario: comparison of data (the category name can be longer because there is more space on the Y axis)

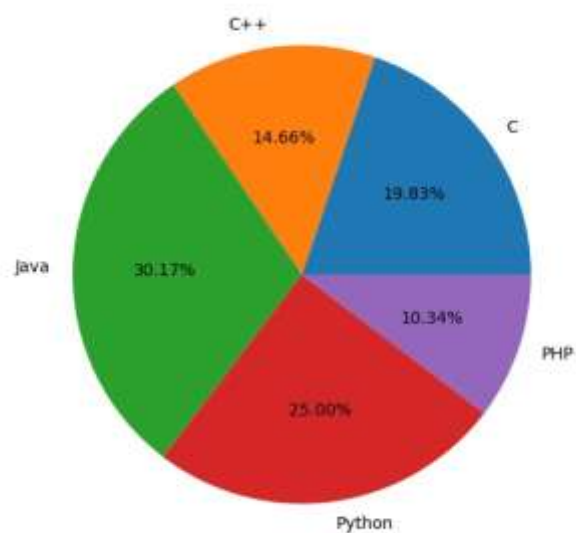


Bar Chart

7.2.4 Pie Charts

Pie charts represent a graph in the shape of a circle. The whole chart is divided into subparts, which look like a sliced pie. They are widely used in various fields to represent the proportion of different classifications, and to compare various classifications by the arc. A pie chart can also be made into a multi-layer pie chart, showing the proportion of different categorical data, while also reflecting the hierarchical relationship. The pie chart is not suitable for multiple series of data, because as the series increase, each slice becomes smaller, and finally the size distinction is not obvious.

Application: series ratio, series size comparison



Pie Chart

7.2.5 Area Charts

An area chart or area graph displays graphically quantitative data. It is based on the line chart. The area between axis and line are commonly emphasized with colors, textures and hatchings. Two or more quantities are compared with an area chart

The area chart is formed on the basis of the line chart. It fills the area between the polyline and the axis in the line chart with color. The filling of the color can better highlight the trend information. The fill color of the area chart should have a certain transparency. The transparency can help the user to observe the overlapping relationship between different series. The area without transparency will cause the different series to cover each other.

Area charts are used to represent cumulated totals using numbers or percentages (stacked area charts in this case) over time. Use the area chart for showing trends over time among related attributes. The area chart is like the plot chart except that the area below the plotted line is filled in with color to indicate volume.

When multiple attributes are included, the first attribute is plotted as a line with color fill followed by the second attribute, and so on.

An area chart represents **the change in a one or more quantities over time**. It's made by plotting a series of data points over time, connecting those data points with line segments, and then filling in the area between the line and the x-axis with color or shading



Area Chart

7.3 EXPLORING ADVANCED VISUALIZATIONS

Advanced Data Visualization refers to a high-level method, typically beyond that of traditional Business Intelligence, that uses data or content to discover deeper insights, make predictions, or generate recommendations

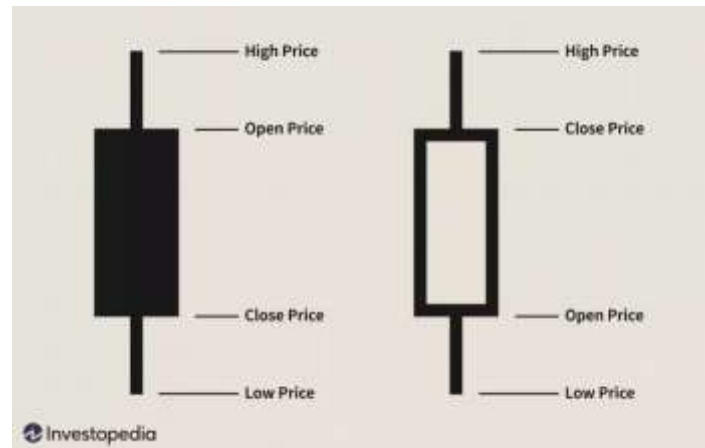
7.3.1 Candlestick Charts

A candlestick is a type of price chart used in technical analysis that displays the high, low, open, and closing prices of a product for a specific period. It originated from Japanese rice merchants and traders to track market prices and daily momentum hundreds of years before becoming popularized in the United States.

Candlesticks show that emotion by visually representing the size of price moves with different colors. Traders use the candlesticks to make trading decisions based on regularly occurring patterns that help forecast the direction of the price.

Candlestick Components

Just like a bar chart, a daily candlestick shows the market's open, high, low, and close price for the day. When the real body is filled in black, it means the close was lower than the open. If the real body is empty, it means the close was higher than the open.



Candlestick Chart

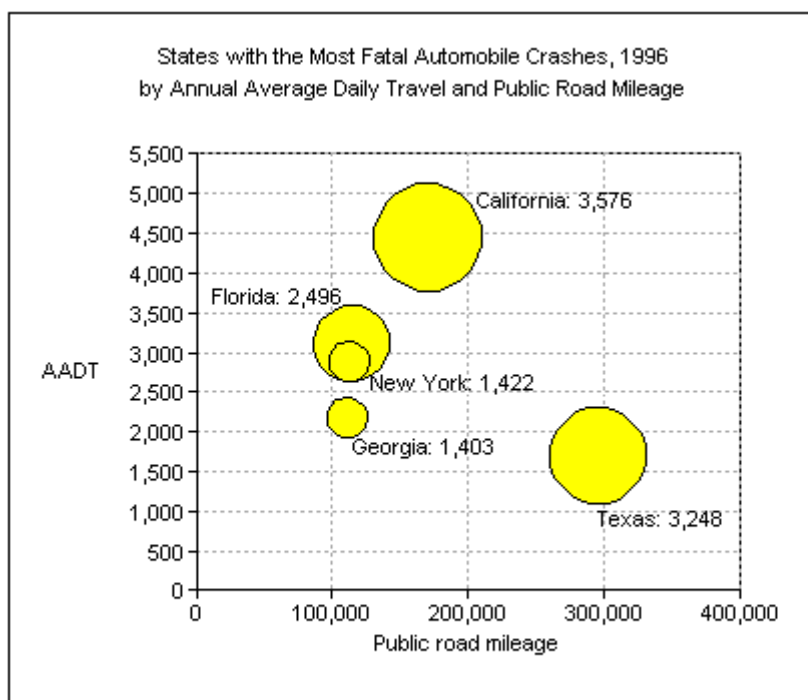
Basic Candlestick Patterns

Candlesticks are created by up and down movements in the price. While these price movements sometimes appear random, at other times they form patterns that traders use for analysis or trading purposes. Patterns are separated into 'bullish' or 'bearish'. Bullish patterns indicate that the price is likely to rise, while bearish patterns indicate that the price is likely to fall. No pattern works all the time, as candlestick patterns represent tendencies in price movement, not guarantees.

7.3.2 Bubble Charts

Bubble charts show **two groups of numbers as a series of XY coordinates**. A third set of numbers indicates the size of each datapoint, or bubble. Bubble charts show the relatedness of three different sets of value

Except for the values of the variables represented by the X and Y axes, the area of each bubble represents the third value. Note that the size of the bubble is limited, and too many bubbles will make the chart difficult to read.



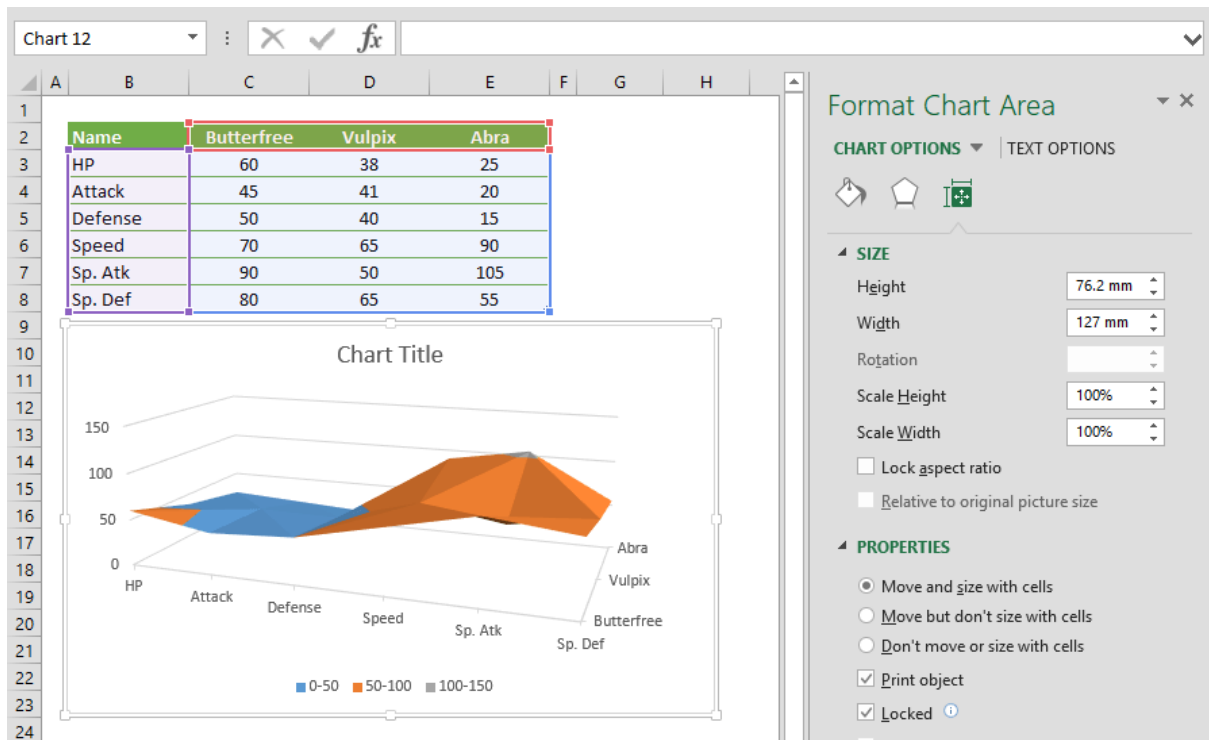
This chart shows the five states in which the highest number of fatal automobile accidents occurred in 1996. Grid lines extend from both value axes in this chart.

Bubble Chart

7.3.3 Surface Charts

Surface Chart (3D Surface Plot) displays a set of three-dimensional data as a mesh surface. It is useful when you need to find the optimum combinations between two sets of data. The colors and patterns in Surface Charts indicate the areas that are in the same range of values by analogy with a topographic map.

A typical 3D Surface Plot is constructed from three variables: X, Y, and Z. Two of them are independent, located on the horizontal axes. The other is dependent, along the vertical axis. Hence, Surface Charts represent a functional relationship between a designated dependent variable and two independent ones.



Surface chart in Excel

7.3.4 Map Charts

Map charts allows to position the data in a context, often geographical, using different layers. The map is divided into three types: regional map, point map, and flow map.

1) Regional Map

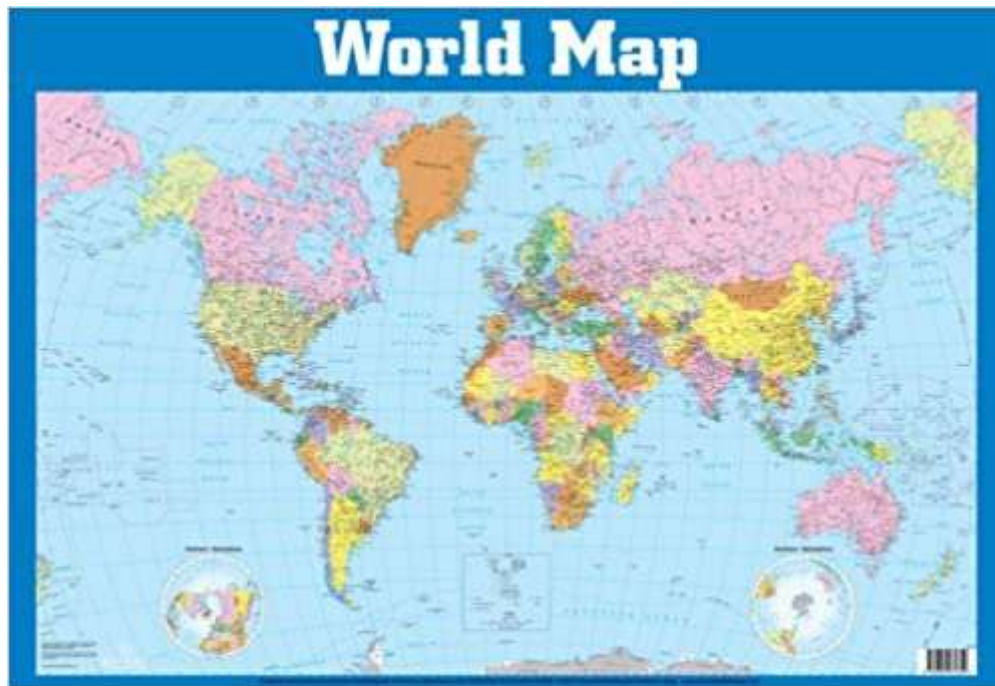
A regional map is a map that uses color to represent the distribution of a certain range of values on a map partition.

2) Point Map

A point map is a method of representing the geographical distribution of data by plotting points of the same size on a geographical background. The distribution of points show the overall distribution of data, but it is not suitable when you need to observe a single specific data.

3) Flow Map

The flow map displays the interaction data between the outflow area and the inflow area. It is usually expressed by the line connecting the geometric centers of gravity of the spatial elements. The width or color of the line indicates the flow value.



7.3.5 Infographics

Infographics are graphic visual representations of information, data, or knowledge intended to present information quickly and clearly. They can improve understanding by utilizing graphics to enhance the human visual system's ability to see patterns and trends.

The three parts of all infographics are the visual, the content, and the knowledge. The visual consists of colors and graphics. There are two different types of graphics: theme, and reference. Statistics and facts usually serve as the content for infographics and can be obtained from any number of sources, including census data and news reports. One of the most important aspects of infographics is that they contain some sort of insight into the data that they are presenting.

Infographics are effective because of their visual element. Humans receive input from all five of their senses (sight, touch, hearing, smell, taste), but they receive significantly more

information from vision than any of the other four. Fifty percent of the human brain is dedicated to visual functions, and images are processed faster than text. The brain processes pictures all at once, but processes text in a linear fashion, meaning it takes much longer to obtain information from text. Online trends, such as the increasingly short attention span of Internet users, has also contributed to the increasing popularity and effectiveness of infographics.

When designing the visual aspect of an infographic, a number of considerations must be made to optimize the effectiveness of the visualization. The six components of visual encoding are spatial, marks, connection, enclosure, retinal properties, and temporal encoding. Therefore, the designers often spatially represent the most important relationship being depicted in an infographic.



Infographics

Thus, Infographics is a clipped compound of "information" and "graphics" which present graphic visual representations of information, data, or knowledge intended to present information quickly and clearly. They can improve cognition by utilizing graphics to enhance the human visual system's ability to see patterns and trends.

When should you use infographics? Infographics can be useful whenever you need to communicate information quickly, or any time you want to make an impact with your data or your message.

7.4 SUMMARY

A graph or chart or diagram is a diagrammatical illustration of a set of data. If the graph is uploaded as an image file, it can be placed within articles just like any other image.

Graphs must be accurate and convey information efficiently. They should be viewable at different computer screen resolutions. Ideally, graphs will also be visually pleasing.

A chart can take a large variety of forms. However, there are common features that provide the chart with its ability to extract meaning from data.

The data in a chart is represented graphically since humans can infer meaning from pictures more quickly than from text. Thus, the text is generally used only to interpret the data.

One of the most important uses of text in a graph is the **title**. A graph's title usually appears above the main graphic and provides a description of what the data in the graph refers to.

Dimensions in the data are often displayed on axes. If a horizontal and a vertical axis are used, they are usually referred to as the x-axis and y-axis. Each axis will have a scale, denoted by periodic graduations and usually accompanied by numerical or categorical indications. Each axis will typically also have a label displayed outside or beside it, briefly describing the dimension represented. If the scale is numerical, the label will often be suffixed with the unit of that scale in parentheses. For example, "Distance traveled (m)" is a typical x-axis label and would mean that the distance travelled, in units of meters, is related to the horizontal position of the data within the chart.

Within the graph, a grid of lines may appear to aid in the visual alignment of data. The grid can be enhanced by visually emphasizing the lines at regular or significant graduations. The emphasized lines are then called major gridlines, and the remainder is minor grid lines.

A chart's data can appear in all manner of formats and may include individual textual labels describing the data associated with the indicated position in the chart. The data may appear as dots or shapes, connected or unconnected, and in any combination of colors and patterns. In addition, inferences or points of interest can be overlaid directly on the graph to further aid information extraction.

When the data appearing in a chart contains multiple variables, the chart may include a legend or key. A legend contains a list of the variables appearing in the chart and an example of their appearance. This information allows the data from each variable to be identified in the chart.

Types of Graphs and Charts include Bar Chart/Graph, Pie Chart., Line Graph or Chart, Histogram Chart, Area Chart, Dot Graph or Plot., Scatter Plot., Bubble Chart.

Every type of graph is a visual representation of data on diagram plots (ex. bar, pie, line chart) that show different types of graph trends and relationships between variables. Although it is hard to tell what are all the types of graphs, this page consists all of the common types of statistical graphs and charts (and their meanings) widely used in any science.

7.5 QUESTIONS

1. What is the importance of graphs and charts?
2. Why are charts and graphs useful?
3. How do you create an infographic?
4. What are the 7 types of infographics?

B.SC. (DATA SCIENCE)
SEMESTER III
DATA MINING AND VISUALIZATION

UNIT -8 VISUALIZING DATA PROGRAMMATICALLY

STRUCTURE

- 8.0 Objective**
- 8.1 Introduction**
- 8.2 Starting with Google charts**
- 8.3 Google Charts API Basics**
- 8.4 A Basic bar chart**
- 8.5 A basic Pie chart**
- 8.6 Working with Chart Animations.**
- 8.7 Summary**
- 8.8 Practice Exercise**

8.0 OBJECTIVE

Understanding how to handle online data with the help of APIs e.g. google API.

8.1 INTRODUCTION

Data visualization is the process of translating large data sets and metrics into charts, graphs and other visuals. The resulting visual representation of data makes it easier to identify and share real-time trends, outliers, and new insights about the information represented in the data.

Visual data exploration is a mandatory initial step whether or not more formal analysis follows. When combined with descriptive statistics, Visualization tasks can range from generating fundamental distribution plots to understanding the interplay of complex influential variables in machine learning algorithms.

We need data visualization because a visual summary of information makes it easier to identify patterns and trends than looking through thousands of rows on a spreadsheet. It's the way the human brain works. Since the purpose of data analysis is to gain insights, data is much more valuable when it is visualized.

Data exploration definition: Data exploration refers to the initial step in data analysis in which data analysts use data visualization and statistical techniques to describe dataset characterizations, such as size, quantity, and accuracy, in order to better understand the nature of the data. Data Exploration is Important because exploration allows for deeper understanding of a dataset, making it easier to navigate and use the data later. The better an analyst knows the data they're working with, the better their analysis will be.



Following programming language is used for data visualization:

JavaScript is a language used the most in websites and web applications. Although there are many JavaScript libraries that enable data visualization, D3 is one of the most powerful and widely used.

Coding is required for data visualization. You don't need to write any code to easily create interactive data visualization. When it comes to presenting data, spreadsheets and reports full of text are not enough to explain what we found. This is when we need data visualization to present the data in a way that helps everyone understand difficult concepts.

Visualizing data Programmatically is the stage in the data science cycle where you get to present your findings after you have worked on understanding and cleaning a dataset.

8.2 GOOGLE CHARTS

Google Charts is an interactive Web service that creates graphical charts from user-supplied information. The user supplies data and a formatting specification expressed in JavaScript embedded in a Web page; in response the service sends an image of the chart.

Google chart tools are powerful, simple to use, **and free**. Try out our rich gallery of interactive charts and data tools.

Google Charts is **an open source chart library** which is powerful and very simple to use. It has many interactive charts to display and render live data. It has a rich chart gallery that include options like pie charts, bar charts, Scatter Charts, donut charts, etc.

Google chart tools are powerful, simple to use, and free. Try out our rich gallery of interactive charts and data tools.

Google Charts specify a way to visualize data on the website. The charts provide a large number of ready-to-use chart types. Google Charts is a pure JavaScript based charting library meant to enhance web applications by adding interactive charting capability. Google Charts provide wide variety of charts. For example, line charts, spline charts, area charts, bar charts, pie

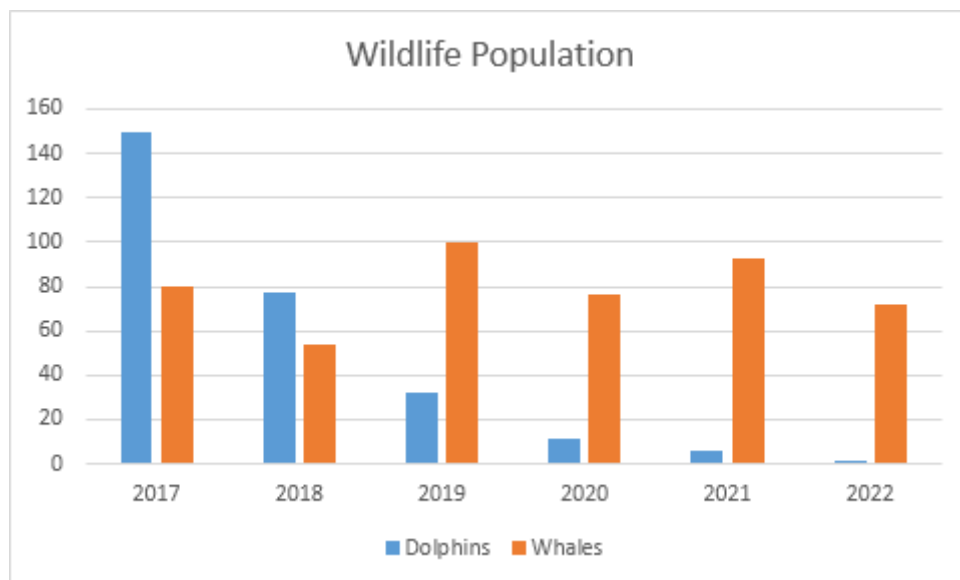
Following are the some features of Google Charts library.

- **Compatability**– Works seamlessly on all major browsers and mobile platforms like android and iOS.
- **Multitouch Support** – Supports multitouch on touch screen based platforms like android and iOS. Ideal for iPhone/iPad and android based smart phones/ tablets.
- **Free to Use** – Open source and is free to use for non-commercial purpose.
- **Lightweight** – loader.js core library, is extremely lightweight library.
- **Simple Configurations** – Uses json to define various configuration of the charts and very easy to learn and use.
- **Dynamic** – Allows to modify chart even after chart generation.
- **Multiple axes** – Not restricted to x, y axis. Supports multiple axis on the charts.
- **Configurable tooltips** – Tooltip comes when a user hover over any point on a charts. googlecharts provides tooltip inbuilt formatter or callback formatter to control the tooltip programmatically.

- **DateTime support** – Handle date time specially. Provides numerous inbuilt controls over date wise categories.
- **Print** – Print chart using web page.
- **External data** – Supports loading data dynamically from server. Provides control over data using callback functions.
- **Text Rotation** – Supports rotation of labels in any direction.

Choose from a variety of charts. From simple scatter plots to hierarchical treemaps, find the best fit for your data. Make the charts your own. Configure an extensive set of options to perfectly match the look and feel of your website.

Full HTML Page Example



An example for creating a web page with visualization charts embedded in it. It also demonstrates a chart connected to Google Spreadsheets.

```
<html>
<head>
  <script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
  <script type="text/javascript">
    google.charts.load('current', {
      'packages': ['table', 'map', 'corechart'],
      // Note: you will need to get a mapsApiKey for your project.
      // See: https://developers.google.com/chart/interactive/docs/basic_load_libs#load-
settings
      'mapsApiKey': 'AIzaSyD-9tSrke72PouQMnMX-a7eZSW0jkFMBWY'
    });
    google.charts.setOnLoadCallback(initialize);
```

```

function initialize() {
  // The URL of the spreadsheet to source data from.
  var query = new google.visualization.Query(
    'https://spreadsheets.google.com/pub?key=pCQbetd-CptF0r8qmCOIZGg');
  query.send(draw);
}

function draw(response) {
  if (response.isError()) {
    alert('Error in query');
  }

  var ticketsData = response.getDataTable();
  var chart = new google.visualization.ColumnChart(
    document.getElementById('chart_div'));
  chart.draw(ticketsData, {'isStacked': true, 'legend': 'bottom',
    'vAxis': {'title': 'Number of tickets'}});

  var geoData = google.visualization.arrayToDataTable([
    ['Lat', 'Lon', 'Name', 'Food?'],
    [51.5072, -0.1275, 'Cinematics London', true],
    [48.8567, 2.3508, 'Cinematics Paris', true],
    [55.7500, 37.6167, 'Cinematics Moscow', false]]);

  var geoView = new google.visualization.DataView(geoData);
  geoView.setColumns([0, 1]);

  var table =
    new google.visualization.Table(document.getElementById('table_div'));
  table.draw(geoData, {showRowNumber: false, width: '100%', height: '100%'});

  var map =
    new google.visualization.Map(document.getElementById('map_div'));
  map.draw(geoView, {showTip: true});

  // Set a 'select' event listener for the table.
  // When the table is selected, we set the selection on the map.
  google.visualization.events.addListener(table, 'select',
    function() {
      map.setSelection(table.getSelection());
    });

  // Set a 'select' event listener for the map.

```

```

// When the map is selected, we set the selection on the table.
google.visualization.events.addListener(map, 'select',
  function() {
    table.setSelection(map.getSelection());
  });
}
</script>
</head>

<body>
<table align="center">
<tr valign="top">
<td style="width: 50%;">
<div id="map_div" style="width: 400px; height: 300;"></div>
</td>
<td style="width: 50%;">
<div id="table_div"></div>
</td>
</tr>
<tr>
<td colspan=2>
<div id="chart_div" style="align: center; width: 700px; height: 300px;"></div>
</td>
</tr>
</table>

</body>
</html>

```

8.3 API BASICS

The **Google Chart API** is an interactive Web Service that creates graphical charts from user-supplied data. Google servers create a Portable Network Graphics (PNG) image of a chart from data and formatting parameters specified by a user's HTTP request. The service supports a wide variety of chart information and formatting.

Google Chart API uses JavaScript and data knowledge, to make them interactive and integrate into you. Google Charts does not support the HTTP request method, and is thus not backward-compatible with user code for the Google Charts API. Upgrading legacy user code from Google Chart API to Google Charts thus requires a complete rewrite of the HTTP semantics in JavaScript sites.

In this section we will discuss about how to set up Google Charts library to be used in web application development.

Install Google Charts

Google Charts can be installed by download.

- **Download** – Download it locally from <https://developers.google.com/chart> and use it.

Using Downloaded Google Chart

Include the googlecharts JavaScript file in the HTML page using following script –

```
<head>
  <script src = "/googlecharts/loader.js"></script>
</head>
```

Google Charts - Configuration Syntax

configuration required to draw a chart using Google Chart API.

Step 1: Create HTML Page

Create an HTML page with the Google Chart libraries.

googlecharts_configuration.htm

```
<html>
  <head>
    <title>Google Charts Example</title>
    <script type = "text/javascript" src = "https://www.gstatic.com/charts/loader.js">
    </script>
    <script type = "text/javascript">
      google.charts.load('current', {packages: ['corechart']});
    </script>
  </head>

  <body>
    <div id = "container" style = "width: 550px; height: 400px; margin: 0 auto">
    </div>
  </body>
</html>
```

Step 2: Create configurations

Google Chart library uses very simple configurations using json syntax.

```
// Instantiate and draw the chart.  
var chart = new google.visualization.PieChart(document.getElementById('container'));  
chart.draw(data, options);
```

Here data represents the json data and options represents the configuration which Google Chart library uses to draw a chart withing container div using draw() method.

Now the various parameter to create the required json string will be configured.

title

Configure the options of the chart.

```
// Set chart options  
var options = {'title':'Browser market shares at a specific website, 2014',  
              'width':550,
```

DataTable

Configure the data to be displayed on the chart. DataTable is a special table structured collection which contains the data of the chart. Columns of data table represents the legends and rows represents the corresponding data. addColumn() method is used to add a column where first parameter represents the data type and second parameter represents the legend. addRows() method is used to add rows accordingly.

```
// Define the chart to be drawn.  
var data = new google.visualization.DataTable();  
data.addColumn('string', 'Browser');  
data.addColumn('number', 'Percentage');  
data.addRows([  
  ['Firefox', 45.0],  
  ['IE', 26.8],  
  ['Chrome', 12.8],  
  ['Safari', 8.5],  
  ['Opera', 6.2],  
  ['Others', 0.7]  
]);
```

Step 3: Draw the chart

```
// Instantiate and draw the chart.
var chart = new google.visualization.PieChart(document.getElementById('container'));
chart.draw(data, options);
```

8.4 A BASIC BAR CHART

googlecharts_bar_basic.htm

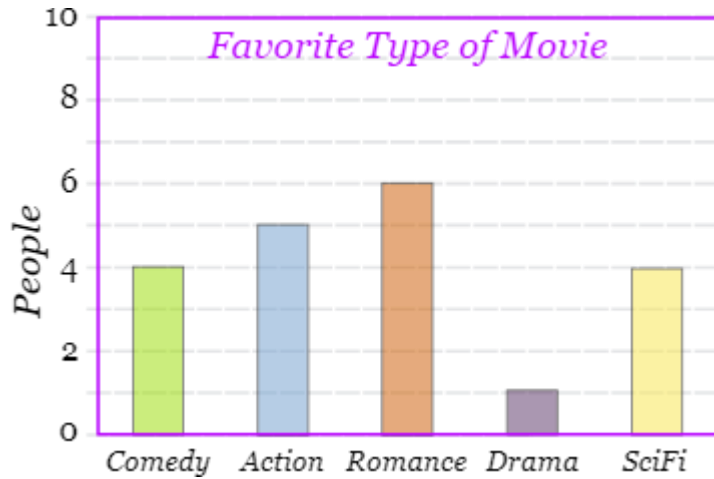
```
<html>
  <head>
    <title>Google Charts Example</title>
    <script type = "text/javascript" src = "https://www.gstatic.com/charts/loader.js">
    </script>
    <script type = "text/javascript">
      google.charts.load('current', {packages: ['corechart']});
    </script>
  </head>

  <body>
    <div id = "container" style = "width: 550px; height: 400px; margin: 0 auto">
    </div>
    <script language = "JavaScript">
      function drawChart() {
        // Define the chart to be drawn.
        var data = google.visualization.arrayToDataTable([
          ['Year', 'Asia'],
          ['2012', 900],
          ['2013', 1000],
          ['2014', 1170],
          ['2015', 1250],
          ['2016', 1530]
        ]);

        var options = {title: 'Population (in millions)'};

        // Instantiate and draw the chart.
        var chart = new
google.visualization.BarChart(document.getElementById('container'));
        chart.draw(data, options);
      }
      google.charts.setOnLoadCallback(drawChart);
    </script>
```

```
</body>
</html>
```



8.5 A BASIC PIE CHART

googlecharts_pie_basic.htm

```
<html>
  <head>
    <title>Google Charts </title>
    <script type = "text/javascript" src = "https://www.gstatic.com/charts/loader.js">
    </script>
    <script type = "text/javascript">
      google.charts.load('current', {packages: ['corechart']});
    </script>
  </head>

  <body>
    <div id = "container" style = "width: 550px; height: 400px; margin: 0 auto">
    </div>
    <script language = "JavaScript">
      function drawChart() {
```



```

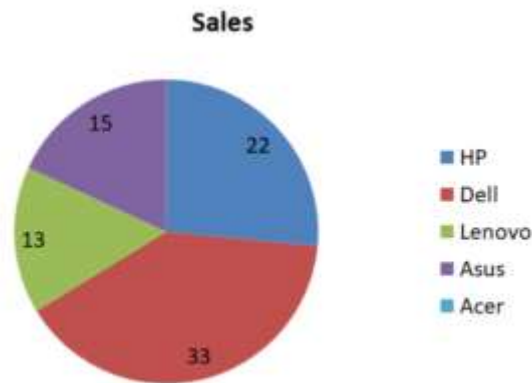
// Define the chart to be drawn.

var data = new google.visualization.DataTable();
data.addColumn('string', 'Browser');
data.addColumn('number', 'Percentage');
data.addRows([
  ['Firefox', 45.0],
  ['IE', 26.8],
  ['Chrome', 12.8],
  ['Safari', 8.5],
  ['Opera', 6.2],
  ['Others', 0.7]
]);

// Set chart options
var options = {
  'title':'Browser market shares at a specific website, 2014',
  'width':550,
  'height':400
};

// Instantiate and draw the chart.
var chart = new
google.visualization.PieChart(document.getElementById('container'));
chart.draw(data, options);
}
google.charts.setOnLoadCallback(drawChart);
</script>
</body>
</html>
Result

```



8.6 WORKING WITH CHART ANIMATIONS

8.6.1 Introduction

Animation is a method in which figures are manipulated to appear as moving images. In traditional animation, images are drawn or painted by hand on transparent celluloid sheets to be photographed and exhibited on film. Today, most animations are made with computer-generated imagery (CGI).

There are 5 Forms of Animation

- Traditional Animation.
- 2D Animation.
- 3D Animation.
- Motion Graphics.
- Stop Motion.

We can add animation to the charts that we create. Chart animation can be used to show a trend in a series of graphs, allowing real-time analysis and comparison of graphic results. With the growth of animated charts came the development of guidelines for creating animated chartsmaps. Visual variables such as spacing, lightness and shape used for static maps apply. However, visual variables unique to animated maps were developed in 1991. Duration is the unit of time a frame or scene is displayed, affecting the smoothness of the animation. The shorter a frame is displayed, the smoother the animation will appear. Smoothness of animation is also a function of the rate of change. These variables were extended in 1995 to include display date (time at which change is initiated), frequency (number of times identifiable forms are displayed).

Thus, an animated chart is basically a fascinating animated trend chart, with bars that race to the top based on ranks. There are usually 3 variables involved in making an animated bar chart, 1. time variable, 2. variable representing categories such as Country names, Company names and so on, 3. variable representing the corresponding value of the each category.

Google charts can animate smoothly in one of two ways, either on startup when you first draw the chart, or when you redraw a chart after making a change in data or options.

To animate on startup:

1. Set the chart data and options.
2. Set animation: {"startup": true} — setting this option will cause the chart to start with series values drawn at the baseline, and animate out to their final state.
3. Call chart.draw(), passing in the data and options.

Following is an example which changes the single value presented in a bar chart upon each click on a button:

```
function init() {
  var options = {
    width: 400,
    height: 240,
    animation: {
      duration: 1000,
      easing: 'out',
    },
    vAxis: {minValue:0, maxValue:1000}
  };
  var data = new google.visualization.DataTable();
  data.addColumn('string', 'N');
  data.addColumn('number', 'Value');
  data.addRow(['V', 200]);

  var chart = new google.visualization.ColumnChart(
    document.getElementById('visualization'));
  var button = document.getElementById('b1');

  function drawChart() {
    // Disabling the button while the chart is drawing.
    button.disabled = true;
    google.visualization.events.addListener(chart, 'ready',
      function() {
        button.disabled = false;
      });
    chart.draw(data, options);
  }

  button.onclick = function() {
    var newValue = 1000 - data.getValue(0, 1);
    data.setValue(0, 1, newValue);
  }
}
```

```

    drawChart();
  }
  drawChart();
}

function init() {
  var options = {
    width: 400,
    height: 240,
    animation: {
      duration: 1000,
      easing: 'out',
    },
    vAxis: {minValue:0, maxValue:1000}
  };
  var data = new google.visualization.DataTable();
  data.addColumn('string', 'N');
  data.addColumn('number', 'Value');
  data.addRow(['V', 200]);

  var chart = new google.visualization.ColumnChart(
    document.getElementById('visualization'));
  var button = document.getElementById('b1');

  function drawChart() {
    // Disabling the button while the chart is drawing.
    button.disabled = true;
    google.visualization.events.addListener(chart, 'ready',
      function() {
        button.disabled = false;
      });
    chart.draw(data, options);
  }

  button.onclick = function() {
    var newValue = 1000 - data.getValue(0, 1);
    data.setValue(0, 1, newValue);
    drawChart();
  }
  drawChart();
}

```

Suggested Readings

1. Jiawei Han, Micheline Kamber and Jian Pei, Data Mining Concepts and Techniques, Third Edition, 2000
2. Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, Introduction to Data Mining, Pearson 2005,
3. M. Kantardzic, "Data Mining: Concepts, Models, Methods, and Algorithms", 2nd edition, Wiley-IEEE Press, 2011
4. Jon Raasch, Graham Murray, Vadim Ogievetsky, Joseph Lowery, JavaScript and jQuery for Data Analysis and Visualization, 2014
5. Ben Fry, "Visualizing data: Exploring and explaining data with the processing environment", O'Reilly, 2007



The Motto of Our University
(SEWA)

SKILL ENHANCEMENT

EMPLOYABILITY

WISDOM

ACCESSIBILITY

SELF-INSTRUCTIONAL STUDY MATERIAL FOR JGND PSOU

ALL COPYRIGHTS WITH JGND PSOU, PATIALA

JAGAT GURU NANAK DEV
PUNJAB STATE OPEN UNIVERSITY, PATIALA

(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

B.Sc.(Data Science)

Semester III

BSDB32303T

Data Preparation

Head Quarter: C/28, The Lower Mall, Patiala-147001

Website: www.psou.ac.in

The Study Material has been prepared exclusively under the guidance of Jagat Guru Nanak Dev Punjab State Open University, Patiala, as per the syllabi prepared by Committee of Experts and approved by the Academic Council.

The University reserves all the copyrights of the study material. No part of this publication may be reproduced or transmitted in any form.

COURSE COORDINATOR AND EDITOR:

Dr. Amitoj Singh

Associate Professor

School of Sciences and Emerging Technologies

Jagat Guru Nanak Dev Punjab State Open University

LIST OF CONSULTANTS/ CONTRIBUTORS

Sr. No.	Name
1	Dr. S.N. Panda
2	Dr. Deepika Koundal



JAGAT GURU NANAK DEV PUNJAB STATE OPEN UNIVERSITY, PATIALA
(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

PREFACE

Jagat Guru Nanak Dev Punjab State Open University, Patiala was established in December 2019 by Act 19 of the Legislature of State of Punjab. It is the first and only Open University of the State, entrusted with the responsibility of making higher education accessible to all, especially to those sections of society who do not have the means, time or opportunity to pursue regular education.

In keeping with the nature of an Open University, this University provides a flexible education system to suit every need. The time given to complete a programme is double the duration of a regular mode programme. Well-designed study material has been prepared in consultation with experts in their respective fields.

The University offers programmes which have been designed to provide relevant, skill-based and employability-enhancing education. The study material provided in this booklet is self-instructional, with self-assessment exercises, and recommendations for further readings. The syllabus has been divided in sections, and provided as units for simplification.

The University has a network of 10 Learner Support Centres/Study Centres, to enable students to make use of reading facilities, and for curriculum-based counselling and practicals. We, at the University, welcome you to be a part of this institution of knowledge.

Prof. Anita Gill
Dean Academic Affairs



B.Sc. (Data Science)
Discipline Specific Course (DSC)
Semester III
BSDB32303T: Data Preparation

Total Marks: 100
External Marks: 70
Internal Marks: 30
Credits: 4
Pass Percentage: 35%

Objective

This course will make student familiar with the methods of data gathering and preparing the data for processing. Student will explore the sampling and data processing techniques.

Section A

Unit I: Data Storage and management: Introduction, Sources of data, Data collection and APIs, Exploring and fixing data, Data storage and management, using multiple data sources, Data Quality, Addressing Data Quality Issues

Unit II: Data Collection: Experiments and surveys, Collection of Primary and Secondary Data, selection of appropriate method for data collection, case study method. Introduction to Data Preparation process, Some problems in preparation process, Missing values and Outliers, types of Analysis.

Unit III Data Preparation: Data formats, parsing and transformation, Scalability and real-time issues. Data Cleaning: Consistency checking, Heterogeneous and missing data, Data Transformation and segmentation

Unit IV: Research Design: Meaning, Need for Research Design, Features of a Good Design, Important Concepts relating to Research Design, Different Research Designs. cluster analysis: Introduction, distance measures Clustering algorithms, agglomerative clustering.

Section B

Unit V: Sampling distributions: Non - central chi - square, t and F distributions and their properties - Distributions of quadratic forms under normality -independence of quadratic form and a linear form - Cochran's theorem.

Unit VI: Simple random sampling (WR and WOR) - Use of Random number Table, Stratified Random Sampling: Properties, Systematic Sampling: Estimation of the Mean and Variance – Comparison of Simple, Stratified and Systematic Sampling

Unit VII Testing of Hypothesis: Hypothesis, Basic Concepts Concerning Testing the Hypotheses, Test Statistic and Critical region, critical value and Decision Rule, Procedure for Hypothesis Testing, Hypothesis Testing for – Means, Proportions, variance, difference of two mean, difference of two proportions, difference of two variances; P-Value approach, power of test,

Unit VIII: Concepts of Time Series – Components of time series – Additive and multiplicative models for the analysis of time series - Measurement of trend by (i) Graphic method, (ii) Semi Average method, (iii) Method of Curve Fitting by principle of least squares, (iv) Method of Moving Averages

Suggested Readings

1. C. R. Kothari – Research Methodology Methods and Techniques - New Age International Publishers, 3rd Edition, 2014.
2. Garg, B.L., Karadia, R., Agarwal, F. and Agarwal, An introduction to Research Methodology, RBSA Publishers. 2002
3. Gupta, S.C. and Kapoor, V.K.: Fundamentals of Applied Statistics, Sultan Chand & Co., 11th ed., 2001
4. William G. Cochran.: Sampling Techniques, John Wiley Sons, 3rd edition, 1977



JAGAT GURU NANAK DEV PUNJAB STATE OPEN UNIVERSITY, PATIALA
(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

BSDB32303T: DATA PREPARATION

COURSE COORDINATOR AND EDITOR: DR. AMITOJ SINGH

UNIT NO.	UNIT NAME
UNIT 1	DATA STORAGE AND MANAGEMENT:
UNIT 2	DATA COLLECTION
UNIT 3	DATA PREPARATION
UNIT 4	RESEARCH DESIGN
UNIT 5	SAMPLING DISTRIBUTIONS
UNIT 6	SIMPLE RANDOM SAMPLING (WR AND WOR)
UNIT 7	TESTING OF HYPOTHESIS
UNIT 8	CONCEPTS OF TIME SERIES

B.Sc.(DATA SCIENCE)

SEMESTER-III

DATA PREPARATION

UNIT I: DATA COLLECTION AND MANAGEMENT

STRUCTURE

1.0 Objective

1.1 Introduction

1.2 Sources of Data

1.2.1. Google's Datasets Search Engine

1.2.2. Gov Dataset

1.2.3. Kaggle Datasets

1.2.4. Amazon Datasets

1.2.5. UCI Machine Learning Repository

1.2.6. Yahoo WebScope

1.2.6. Yahoo WebScope

1.3 Data Collection And API

1.4 Evaluate Data Quality

1.5 Summary

1.0 OBJECTIVE

Understanding the concept of data collection.

Different techniques available for data collection and management

1.1 INTRODUCTION

Data science is a wide field that alludes to the collective processes, theories, ideas, devices and innovations that empower the survey, investigation and extraction of significant information and data from raw information.

The line separating the individuals who can play with ML and the people who can't is drawn by long period of gathering data. A few associations have been accumulating records for quite a long time with such incredible achievement that now they need trucks to move it to the cloud as conventional broadband is simply not broad enough.

For those who've quite recently come on the scene, absence of information is normal, yet luckily, there are approaches to transform that minus into a plus.

In the first place, depend on open source datasets to start ML execution. There are heaps of information for AI around and a few organizations (like Google) are prepared to part with it. While those chances exist, normally the genuine value comes from inside gathered brilliant information pieces mined from the business decisions and exercises of your own organization.

Second – and as anyone might expect – presently you get an opportunity to gather information the correct way. The organizations that began information collection with paper records and finished with .xlsx and .csv documents will probably make some harder time with information readiness than the individuals who have a little yet pleased ML-accommodating dataset. If you know the assignments that ML ought to tackle, you can tailor an information gathering technique in advance.

Shouldn't something be said about enormous information? It's so hummed, it seems like what everybody ought to do. Focusing on huge information from the beginning is a decent attitude, yet enormous information isn't about petabytes. Everything's with regards to the capacity to handle them the correct way. The bigger your dataset, the harder it will utilize it and yield bits of knowledge. Having huge loads of wood doesn't really mean you can change it over to a stockroom brimming with seats and tables. Along these lines, the overall suggestion for novices is to begin little and lessen the intricacy of their information.

1. Articulate the issue early

Knowing what you need to foresee will assist you with choosing which information might be more significant to gather. While forming the issue, direct information investigation and attempt to think in the categories of classification, clustering, regression, and ranking. In layman's terms, these errands are separated in the accompanying way:

Classification. You need a calculation to answer binary yes-or-no inquiries (cats or dogs, good or bad, sheep or goats, you get the thought) or you need to make a multiclass classification (grass, trees, or bushes; cats, dogs, or birds) You likewise need the right answers labeled, so an algorithm can learn from them.

Clustering. You need a calculation to discover the standards of classification and the quantity of classes. The fundamental difference from classification tasks is that you don't really have a clue what the groups and the principles of their division are. For example, this generally happens when you need to fragment your clients and tailor a particular way to deal with each section depending upon its characteristics.

Regression. You need a calculation to yield some numeric value. For instance, in the event that you invest an excessive amount of time thinking of the right cost for your item since it relies upon many factors, regression algorithms can support assessing this value.

Ranking. Some ML algorithms simply rank items by number of features. Ranking is effectively used to suggest movies in video streaming service or show the items that a client may buy with a high likelihood dependent on their past search and purchase activities.

All things considered, your business issue can be tackled inside this straightforward division and you might begin adjusting a dataset likewise. The general guideline on this stage is to keep away from over-complicated issues.

2. Establish data collection mechanisms

Making an information driven culture in an association is maybe the hardest part of the entire initiative. In the event that you intend to utilize ML for predictive analytics, the primary thing to do is to content with data fragmentation.

For example, if you look at travel tech – one of AltexSoft's critical subject matters – data fragmentation is one of the top analytics problems here. In hotel organizations, the divisions that are accountable for actual property dive into pretty personal insights concerning their visitors. Hotels realize visitors' Mastercard numbers, kinds of conveniences they pick, in some cases places of residence, room administration use, and even beverages and dinners requested during a stay. The site where individuals book these rooms, in any case, may treat them as complete outsiders.

This information gets siloed in various departments and surprisingly tracking points within a department. Advertisers might approach a CRM yet the clients there aren't related with web analytics. It's not generally conceivable to unite all information streams into a centralized storage in the event that you have many channels of engagement, acquisition, and retention, however as a rule it's manageable.

There are two significant sorts of data collection mechanisms.

Data Warehouses and ETL

The first one is storing information in warehouses. These storages are typically made for organized (or SQL) records, which means they fit into standard table arrangements. Most would agree that every one of your business records, payrolls, and CRM information fall into this class. One more conventional trait of managing warehouses is changing information prior to stacking it there. In any case, it implies that you know which information you need and how it should look, so you do all the processing prior to storing. This methodology is called Extract, Transform, and Load (ETL).

The issue with this methodology is that you don't generally know ahead of time which information will be helpful and which will not. In this way, warehouses are ordinarily used to get to information through business insight interfaces to visualize the metrics we realize we need to follow. Furthermore, there's another way.

Data Lakes and ELT

Data lakes are storages equipped for keeping both organized and unstructured information, including pictures, recordings, sounds records, PDF documents... you get the thought. Yet, regardless of whether information is organized, it's not changed prior to storing. You would stack information there with no guarantees and conclude how to utilize and handle it later, on request. This methodology is called Extract, Load, and — then, at that point when you need — Transform.

Handling human factor

One more point here is the human factor. Data collection might be a tedious task that loads your workers and overwhelms them with directions. If individuals should continually and physically make records, the changes are they will consider these tasks at this point another administrative impulse and let the work slide. For example, Salesforce gives a good toolset to follow and break down sales reps exercises yet manual information section and action logging alienates salesmen.

This can be tackled utilizing robotic process automation systems. RPA calculations are straightforward, rule based bots that can do tedious and repetitive tasks.

3. Check your data quality

The main inquiry you should pose — do you trust your information? Indeed, even the most complex ML algorithms can't work with poor information.

How tangible is human mistake? If your information is gathered or labeled by people, check a subset of information and estimate how regularly mistakes occur.

Were there any specialized problems while transferring data? For example, similar records can be copied due to server error, or you had a capacity crash, or possibly you encountered a cyberattack. Assess what these event affected your information.

What number of omitted values does your information have? While there are approaches to deal with omitted records.

Is your information sufficient to your task? In case you've been selling home machines in the US and presently anticipate stretching into Europe, would you be able to utilize similar information to foresee stock and demand?

Is your information imbalanced? Envision that you're attempting to mitigate supply chain risks and filter out those suppliers that you consider untrustworthy and you utilize the quantity of traits (e.g., area, size, rating, and so forth) In the event that your labeled dataset has 1,500 passages labeled as reliable and just 30 that you consider unpredicable, the model will not have enough examples to find out the untrustworthy ones.

4. Format data to make it consistent

Data formatting is sometimes referred to as the file format you're utilizing. Furthermore, this isn't a very remarkable issue to change over a dataset into a record design that fits your ML framework best.

We're discussing about format consistency of records themselves. In case you're gathering information from various sources or your dataset has been actually restored by various individuals, it merits ensuring that all factors inside a given attribute are constantly composed. These might be date formats, amounts of cash (4.03 or \$4.03, or even 4 dollars 3 cents), addresses, and so forth The information format ought to be similar across the whole dataset.

Also, there are different parts of information consistency. For example, in the event that you have a set numeric reach in a trait from 0.0 to 5.0, ensure that there are no 5.5s in your set.

5. Reduce data

It's enticing to incorporate however much information as could be expected, as a result of... well, big data! Indeed, you certainly need to gather all information feasible. However, in case you're setting up a dataset considering specific assignments, it's smarter to reduce data.

Since you know what the objective characteristic (what esteem you need to anticipate) is, sound judgment will direct you further. You can accept which values are basic and which will add more dimensions and complexity to your dataset with no anticipating contribution.

This methodology is called **attribute sampling**.

For instance, you need to anticipate which clients are inclined to make enormous purchases in your online store. The age of your clients, their area, and gender can be preferable indicators over their Mastercard numbers. Be that as it may, this additionally works another way. Consider which different qualities you might have to gather to uncover more conditions. For example, adding bounce rates might build precision in anticipating conversion.

That is where domain expertise assumes a major part. Getting back to our starting story, not all information researchers realize that asthma can cause pneumonia confusions. Similar works with decreasing enormous datasets. On the off chance that you haven't utilized a unicorn who has one foot in medical services fundamentals and the other in data science, almost certainly, an data scientist may struggle understanding which values are of genuine importance to a dataset.

Another methodology is called record **sampling**. This suggests that you essentially eliminate records (objects) with missing, erroneous, or less representative values to make prediction more exact. The procedure can likewise be utilized in the later stages when you need a prototype model to comprehend whether a selected ML strategy yields expected outcomes and estimate ROI of your ML initiative.

You can likewise reduce data by gathering it into more extensive records by partitioning the whole attribute data into multiple groups and drawing the number for each gathering. Rather than investigating the most purchased products of a given day through five years of online store presence, combined them to week by week or month to month scores. This will assist with decreasing data size and computing time without noticeable prediction losses.

6. Complete data cleaning

Since missing qualities can substantially lessen prediction accuracy, focus on this issue. As far as ML, expected or approximated values are " more right" for an algorithm than simply missing ones. Regardless of whether you don't have a clue about the specific worth, methods exist to better "assume" which value is missing. How to clean data? Picking the right methodology likewise vigorously relies upon data and the domain you have:

- Substitute missing qualities with faker values, e.g., n/a for categorical or 0 for numerical values
- Substitute the missing numerical values with mean figures
- For categorical values, you can likewise utilize the most incessant items to fill in.

If you utilize some ML as a service platform, data cleaning can be mechanized. For example, Azure Machine Learning permits you to pick among accessible strategies, while Amazon ML will do it without your inclusion by any stretch of the imagination.

7. Create new features out of existing ones

A few qualities in your dataset can be mind boggling and decomposing them into different parts will help in catching more explicit relationships. This process is inverse to reducing data as you need to add new attributes dependent on the current ones.

For instance, if your sales performance differs relying upon the day of week, isolating the day as a different categorical value from the date (Mon; 06.19.2017) may deliver the algorithm with more relevant data.

8. Join transactional and attribute data

Transactional data comprises of events that captures explicit moments, for example what was the cost of the boots and when a client with this IP tapped on the Buy now button?

Attribute data is more static, similar user demographics or age and doesn't straightforwardly identify with explicit events.

You might have a few information sources or logs where these kinds of information dwell. The two sorts can upgrade each other to accomplish greater predictive power. For example, in case you're following sensor readings to enable predictive maintenance, in all likelihood you're producing logs of transactional data, yet you can add such characteristics as the equipment model, the batch, or its area to search for dependencies between equipment behavior and its attributes.

Likewise you can aggregate transactional data into attributes. Say, you assemble website session logs to appoint various attributes to various users, e.g., researcher (visits 30 pages overall, infrequently purchases something), *reviews reader* (investigates the surveys page start to finish), *instant buyer*, and so forth, then, at that point you can utilize this data to, for instance, upgrade your retargeting efforts or foresee client lifetime value.

9. Rescale data

Data rescaling has a place with a collection of data normalization methods that target working on the quality of a dataset by reducing dimensions and keeping away from the circumstance when some values overweight others. What's the significance here?

Envision that you run a chain of vehicle sales centers and most of the properties in your dataset are either categorical to portray models and body styles (car, hatchback, van, and so on) or have 1-2 digit numbers, for example, for quite a long time of utilization. Yet the costs are 4-5 digit numbers (\$10000 or \$8000) and you need to foresee the average time for the vehicle to be sold dependent on its attributes (model, previous year use, body style, value, condition, and so on) While the cost is a significant rule, you don't need it to overweight different ones with a bigger number.

For this situation, min-max normalization can be utilized. It involves changing numerical values to ranges, e.g., from 0.0 to 1.0 where 0.0 addresses the negligible and 1.0 the greatest qualities to even out the weight of the value attribute with different properties in a dataset.

A bit more straightforward methodology is decimal scaling. It involves scaling information by moving a decimal point one or the other way for similar purposes.

10. Discretize data

Now and then you can be more effective in your forecasts in the event that you transform numerical values into categorical values. This can be accomplished, for instance, by separating the whole range of values into various groups.

If you track client age figures, there is certifiably not a major difference between the age of 13 and 14 or 26 and 27. So these can be changed over into applicable age groups. Making the values categorical, you improve on the work for an algorithm and basically make prediction more relevant.

1.2 SOURCES OF DATA

Let us start by discovering ML Datasets that are issue explicit, and ideally cleaned and pre-handled .

It certainly is an exhausting task to discover explicit datasets like MS-COCO for all assortments of issues. In this manner, we should be smart with regards to how we use datasets. For instance, utilizing Wikipedia for NLP errands is most likely the best NLP dataset there perhaps is. Now we examine a portion of the different sources for Machine Learning Datasets, and how we can continue further with something very similar. An expression of alert, be cautious while perusing the agreements that each of these datasets impose, and follow likewise. This is to the greatest advantage of everybody for sure.

1.2.1. Google's Datasets Search Engine

Google has been the giant search engine, and they helped all the ML professionals out there by doing what they are masters at, assisting us with finding datasets. The search engine makes a fantastic job with getting datasets identified with the keywords from different sources, including government sites, Kaggle, and other open-source stores.

1.2.2. .Gov Dataset

With the United States, China and a lot more nations focussing on ML, information is being democratized. The standards and guidelines identified with these datasets are typically severe as they are real information gathered from different areas of a country. In this way, cautious use is suggested.

The list of open access datasets are as follows:

Indian Government Dataset

Australian Government Dataset

EU Open Data Portal

New Zealand's Government Dataset

Singapore Government Dataset

1.2.3. Kaggle Datasets

Kaggle is known for facilitating ML and deep learning challenges. The pertinence of Kaggle in this regard is that they give datasets, and simultaneously give to ML professionals, whose work will assist us with our advancement. Each challenge has a particular dataset, and it is typically cleaned so we don't need to accomplish preprocessing essentially and can rather focus on refining the algorithm. The datasets are effectively downloadable. Under the resources segment, there are essentials and links to learning material, which helps us either in algorithm or in implementation. Kaggle is a phenomenal site for beginners to wander into uses of ML and deep learning and is a definite asset pool for intermediate specialists of ML.

1.2.4. Amazon Datasets (Registry of Open Data on AWS)

Amazon has recorded a portion of the datasets accessible on their servers as freely open. Consequently, when utilizing AWS assets for adjusting and tweaking models, utilizing these locally accessible datasets will secure the information stacking measure by many occasions. The library contains a few datasets characterized by the field of utilizations like satellite pictures, environmental assets, and so on

1.2.5. UCI Machine Learning Repository

UCI Machine Learning Repository gives simple to utilize and cleaned datasets. These have been the go-to datasets for quite a while in scholarly world.

1.2.6. Yahoo WebScope

An interesting component that this site gives is it records the paper which utilized the dataset. Subsequently, all exploration researchers and individuals from the scholarly world will discover this asset convenient. The datasets accessible can't be utilized for business purposes. For additional subtleties, genuinely look at the sites of the datasets gave.

1.2.7. Datasets Subreddit

The subreddit can be utilized as an auxiliary aide when any remaining alternatives turn into dead end. Individuals ordinarily talk about the different accessible datasets and how to utilize existing datasets for new undertakings. A great deal of bits of knowledge in regards to the essential tweaking needed for datasets to work in various conditions can be acquired also. By and large, this ought to be the last asset point for datasets.

Let us concentrate on data sets specific to the key domains which during last 2 decades 've observed accelerated progress. The availability of domain-specific data sets improves the strength of the model, which enables more realistic and accurate results. Computer vision, NLP and data analysis are among these areas.

1.3 DATA COLLECTION AND API

Each data scientist should know the data collection and the APIs, five APIs

- Facebook API

A large amount of data is generated on a daily basis by Facebook API. The countless post, comments and shares producing huge data in multiple clusters and pages. And that massive data offers a wide range of ways to analyze the audience.

- **Google Map API**

One of the most often used APIs is Google Map API. Its applications range from integration into the popular Pokemon Go to the transportation services application.

All information such as location coordinates, location distances, routes, etc. is available. It's also enjoyable to utilize this API to create own data sets with the distance feature.

- **Twitter API**

Twitter data can also be accessible through the Twitter API, just as the Facebook Graph API. You can get all data such as user tweets, tweets with a specific term, or even a mix of terms, tweets in a specific date range on the subject, etc.

Twitter data is an excellent resource for carrying out activities such as opinion analysis.

- **IBM Watson API**

IBM Watson offers an collection of APIs to complete a variety of sophisticated activities, such as sound analyzers, conversions of documents, personality insights, visual identification, speech to text, text to text and so on... This group of APIs differs in that they serve to manipulate and obtain information on the data from other APIs.

- **Quandl API**

Quandl allows you to access information about the time series of a wide number of stocks for the detailed date series. The Quandl API is fairly simple and offers an excellent resource for tasks like as Stock pricing and stock profiling etc.

. These API Linked to specified data science projects .

I'm sure you're attracted by the aforementioned APIs, but you can ask if you can build an API project which adds tremendous value to your CV? Well, this is the list of ideas with which you can begin. These APIs can either be used to collect and change data to gain insights from them or to give information to the APIs and to accomplish sophisticated tasks.

Here is the list of projects you are dealing with. I'm going to leave you with the implementation of these ideas.

Data Science Projects	API's Used in Projects
Social Media Sentiment Analysis	Twitter, Facebook API
Opinion Mining	Twitter, Facebook API
Stock Prediction	Yahoo Stock API, Quandl API
Most Popular languages on Github	Github API
Microsoft Face Sentiment Recognition	Microsoft face API

It is extremely important to ensure that the data used to train models are of very good quality in order to develop models of machine learning with high performance. 43 percent respondents reported that data quality is the greatest barrier to machine learning, according to one statistics from a poll.

What is a data set of low quality?

The following scenarios can be described as low in the quality of data utilized for machine learning models:

Data set incorrectly labeled: data with improper labels may be wrong.

Data set of low quality: Data with missing values or incorrect values may be available. This could be unclear in the event of pictures. An example of the bad quality data is shown in Fig 1:



Fig 1. Low quality data

• **Data samples for training and test data glitches:** Here is an instance if the training and test data set do not correspond:

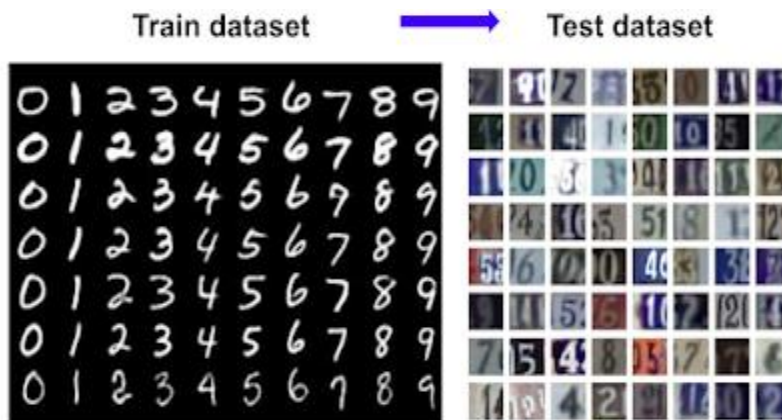


Fig 2. Training and test data set mismatch

1.4 EVALUATE DATA QUALITY

There are several reasons why data quality has to be assessed during the training of one or more models:

High performance models to be achieved: Models trained with inaccurate, poor quality or data-labeling samples many times influence model performance in the sense that the model does not generalize for the population. The deletion of low or improperly categorized data frequently leads to increased model performance in such cases. In addition, if the training and test data samples do not match, more model performance can be achieved with reducing the data in the training data that matches the test data set.

Assess if machine learning models can be used in SaaS settings: in solutions that are compatible with SaaS, where models depend on the data set of customers, only on the basis of data quality may be established whether SaaS clients can be supplied with a machine learning solution.

Use Cases: Evaluation of data quality with a kind of quantification that refers to high, medium, low or very little data might lead to some interesting uses.

Determine strategies for data collection: Quantifying data quality often leads to a data gathering strategy for various data sources.

Data Quality Assessment Frameworks for Machine Learning

Here are some of the noteworthy data quality evaluation frameworks that may be used to evaluate data quality to achieve the advantages of having a high quality ML model training dataset as indicated in the last section.

Data Valuation using Reinforcement Learning (DVRL)

Google has tried a framework called an estimator framework for data value that employs reinforcement learning to estimate data values (instead of gradient descent) and chooses the most important samples for training the model of predicting. It is a new meta-learning framework for data evaluation that defines the likelihood of each sample of training to utilize in predictor model training.

The model training procedure involves both the estimator of the data value and the model training method. More valuable training samples are used to train the model compared to lower valuable data with the data value estimator. The estimator, sample and predictor model of data value together with loss calculation are utilized to detect high-value data points.

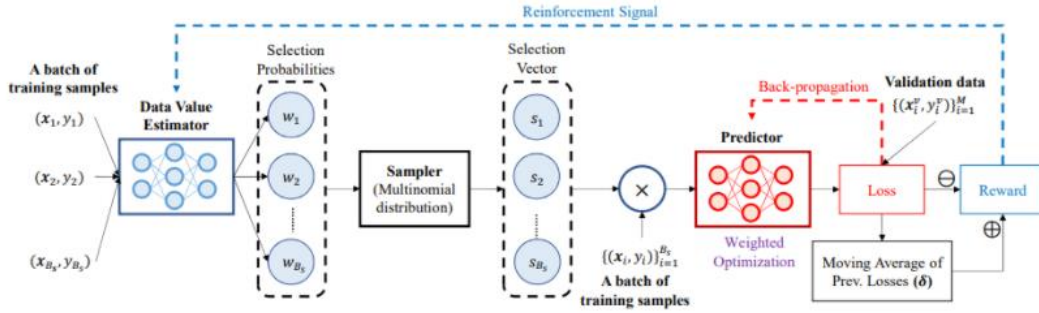


Fig 3. Google Data Value Estimator Framework (Reinforcement Learning)

It is shown that the Google DVRL framework works well with respect to the worth of each data from the three data quality assessment framing described in this article (such as Data Shapley, LOO & DVRL).

Shapley Data

A fair paradigm for quantifying the value of specific training sources is Data Shapley. Three inherent qualities of equitable data assessment relating to the following can only be met by the data sheet framework:

- If you do not impact model performance by adding data points to the train data, the data points' value is zero.
- If the combination of new varying values does not really influence performance of the model, the two data points will have equal value. •

Data Shapley gives a statistic for assessing performance in the machine learning algorithms for each learning data point.

Leave One Out (LOO) Technique

One way to assess model performance training dataset set is among the most prevalent methods. The most common way for determining the value of every one of the data points is this procedure. The value of each data point is determined throughout this procedure when each data point is deleted from the training data set and before and after the model performance measured. The value of the data point is computed by removing the performance from the model before and after the data point has been removed. With LOO the objective is to assess model performance after the data item has been removed. LOO technique has restrictions on the inability to recycle the model and to reassess it.

B.Sc.(DATA SCIENCE)

SEMESTER-III

DATA PREPARATION

UNIT II: DATA COLLECTION

STRUCTURE

2.0 Objectives

2.1 Data Collection

2.2 Data Collection Methods

2.2.1. Interviews

2.2.2 Observations

2.2.3 Surveys and Questionnaires

2.2.4. Focus Groups

2.2.5. Oral Histories

2.3 Secondary Data Collection Methods

2.4 Data Preparation

2.4.1 Dealing with Missing Data

2.4.2 Transformation

2.5 Outlier

2.6 Summary

2.0 OBJECTIVES

Understanding Experiments and surveys for Collection of Primary data. Understanding Secondary Data selection. Identifying appropriate method for data collection. Introduction to Data Preparation process.

2.1 DATA COLLECTION

Data collection is the method involved with gathering, estimating and dissecting various sorts of data utilizing a set of standard validated procedures. The major target of Data collection is to accumulate data rich and reliable data, and analyze them to make business assessments. When the information is gathered, it goes through process of data cleaning and information handling to make this data really helpful for organizations.

Accurate data collection is fundamental to keeping up with the uprightness of research, settling on educated business decisions and guaranteeing quality assurance. For instance, in retail deals, data may be gathered from mobile applications, website visits, faithfulness programs and online studies to study clients. In a server consolidation project, data collection would incorporate an physical inventory, an accurate description of all the installed software in servers such as operating system, middleware and the application or data set that the server supports.

2.2 DATA COLLECTION METHODS

Surveys, interviews and focus groups are essential instruments for gathering data. Today, with assistance from Web and analytics tools, organizations are additionally ready to gather information from mobile devices, site traffic, server activity and other applicable sources, contingent upon the project. There are two primary techniques for data collection in research dependent on the data that is required, in particular:

- Primary Data Collection
- Secondary Data Collection

Primary Data Collection Methods

Primary data alludes to information gathered from direct experience straightforwardly from the primary source. It alludes to information that has never been utilized previously. The information accumulated by essential primary data collection strategies are for the generally regarded as the most ideal sort of data in research.

The strategies for gathering primary data can be additionally partitioned into quantitative data collection techniques (manages factors that can be counted) and data collection methods techniques (manages factors that are not really mathematical in nature).

There are various primary data collection techniques. These are described as follows:

2.2.1 Interviews

Interviews are a uninterrupted technique for collection of data. It is basically a procedure in which the questioner poses inquiries and the interviewee reacts to them . It delivers a high level of adaptability since questions can be changed and changed whenever as indicated by the circumstance.

2.2.2 Observations

In this strategy, analysts observe a circumstance around them and record the outcomes. The evaluation of behaviour of different people can be done through observations . This technique is profoundly viable in light of the fact that it is clear and not straightforwardly subject to different members. For instance, an individual glances indiscriminately individuals that walk their pets on a bustling road, and afterward utilizes this information to choose whether or not to open a pet food store around there.

2.2.3 Surveys and Questionnaires

Questionnaires and surveys give a wide point of view from huge gatherings of individuals. They can be directed up face to face, formal conversation , or even presented on the Internet to get responses of the people around the world.. The appropriate responses can be yes or no, multiple choice, true or false and even open-ended queries. Notwithstanding, a disadvantage of reviews and polls is deferred reaction and the chance of uncertain answers.

2.2.4. Focus Groups

Focus group is directed with a gathering of individuals who all share something for all intents and purpose. The data collected is like face to face meets, however they offer a superior comprehension of why a specific gathering of individuals thinks with a certain goal in mind. In any case, a few downsides of this technique are absence of security and control of the meeting by a couple of members. Focus group can likewise be tedious and testing, however they assist with uncovering probably the best data for complex circumstances.

2.2.5. Oral Histories

In Oral histories the collection of data is related to single phenomenon. It also includes enquiring queries such as conferences and focus groups. The people involved in particular event are focused for collection of views and particular practices of people. As an example the new product in an organization can be effect of the individuals perception.

2.3 SECONDARY DATA COLLECTION METHODS

The data collected by somebody else is known as secondary data. Secondary data refers to data that has already been collected by someone else. It is reasonable and easier to gather than primary data. Although primary data is more authentic, secondary data also provides unlimited

importance to organizations. Here are probably the most widely secondary data collection approaches:

1. Internet

The utilization of the Internet has become one of the most famous secondary data collection techniques. There is a huge pool of free and paid research assets that can be effortlessly accessed on the Internet. While this strategy is a quick and simple method of data collection, you should just refer from genuine sites while gathering data.

2. Government Archives

There is a lot of data accessible from government documents that you can utilize. The main benefit is that the information in government files are legitimate and unquestionable. The test, notwithstanding, is that information isn't in every case promptly accessible because of various elements. For instance, criminal records can go under arranged data and are hard for anybody to approach them.

3. Libraries

Most analysts give a few copies of their scholarly research to libraries. You can gather significant and authentic information dependent on various exploration contexts. Libraries likewise serve as a storage facility for professional references, yearly reports and other similar documents that help organizations in their research.

Use Case: Conducting Customer Surveys to Multiply Sales

An exploration study was directed by Rice University Professor Dr. Paul Dholakia and Dr. Vicki Morwitz to realize whether an organization could impact clients' reliability or purchasing habits. The exploration study was led throughout the span of a year. One gathering of clients were studied and the other set was not overviewed about consumer contentment. In the following year, the gathering that took the study were threefold as prone to renew their loyalty towards the association than the other group.

Information can be attained from diverse information sources, for example,

1. APIs
2. File size
3. Database
4. Videos/pictures/sounds

Python Libraries utilized in Data Analysis are explained as follows:

CSV Format:

Comma Separated Values are present in text files. Representation of data is in tabular form. Here delimiter comma is used for each record in a separate line.

Code:

```
import pandas as pd
dataset = pd.read_csv("abc.csv")
dataset.head(5)
```

Image 1



File Edit Format View Help
age:"job":"marital":"education":"default":"balance":"housing":"loan":"contact":"day":"month":"duration":"campaign":"pdays":"previous":"poutcome":"y"
30:"unemployed":"married":"primary":"no":1787:"no":"no":"cellular":19:"oct":79:1:-1:0:"unknown":"no"
33:"services":"married":"secondary":"no":4789:"yes":"yes":"cellular":11:"may":220:1:339:4:"failure":"no"
35:"management":"single":"tertiary":"no":1350:"yes":"no":"cellular":16:"apr":185:1:330:1:"failure":"no"
30:"management":"married":"tertiary":"no":1476:"yes":"yes":"unknown":3:"jun":199:4:-1:0:"unknown":"no"
59:"blue-collar":"married":"secondary":"no":0:"yes":"no":"unknown":5:"may":226:1:-1:0:"unknown":"no"
35:"management":"single":"tertiary":"no":747:"no":"no":"cellular":23:"feb":141:2:176:3:"failure":"no"
36:"self-employed":"married":"tertiary":"no":307:"yes":"no":"cellular":14:"may":341:1:330:2:"other":"no"
39:"technician":"married":"secondary":"no":147:"yes":"no":"cellular":6:"may":151:2:-1:0:"unknown":"no"
41:"entrepreneur":"married":"tertiary":"no":221:"yes":"no":"unknown":14:"may":57:2:-1:0:"unknown":"no"
43:"services":"married":"primary":"no":88:"yes":"yes":"cellular":17:"apr":313:1:147:2:"failure":"no"
39:"services":"married":"secondary":"no":9374:"yes":"no":"unknown":20:"may":273:1:-1:0:"unknown":"no"
43:"admin":"married":"secondary":"no":264:"yes":"no":"cellular":17:"apr":113:2:-1:0:"unknown":"no"
36:"technician":"married":"tertiary":"no":1109:"no":"no":"cellular":13:"aug":328:2:-1:0:"unknown":"no"
20:"student":"single":"secondary":"no":502:"no":"no":"cellular":30:"apr":261:1:-1:0:"unknown":"yes"
31:"blue-collar":"married":"secondary":"no":360:"yes":"yes":"cellular":29:"jan":89:1:241:1:"failure":"no"



age	job	marital	education	default	balance	housing	loan	contact	day	month	duration	campaign	pdays	previous	poutcome	y	
0	30	unemployed	married	primary	no	1787	no	no	cellular	19	oct	79	1	-1	0	unknown	no
1	33	services	married	secondary	no	4789	yes	yes	cellular	11	may	220	1	339	4	failure	no
2	35	management	single	tertiary	no	1350	yes	no	cellular	16	apr	185	1	330	1	failure	no
3	30	management	married	tertiary	no	1476	yes	yes	unknown	3	jun	199	4	-1	0	unknown	no
4	59	blue-collar	married	secondary	no	0	yes	no	unknown	5	may	226	1	-1	0	unknown	no

SciPy

- SciPy is the **group of open-source libraries**, which assists with getting sorted out our information for investigation.
- There are different libraries utilized specifically,

Numpy

- o Used for logical figuring like Numerical Analysis, Linear polynomial math, and metric calculation.
- o It is fundamental for Machine Learning (ML) execution

Code:

#importing numpy package

```
import numpy as numpy
```

#creating array

```
ar = numpy.array([0,1,2,3,"hi"])
```

```
print(ar)
```

#type of ar

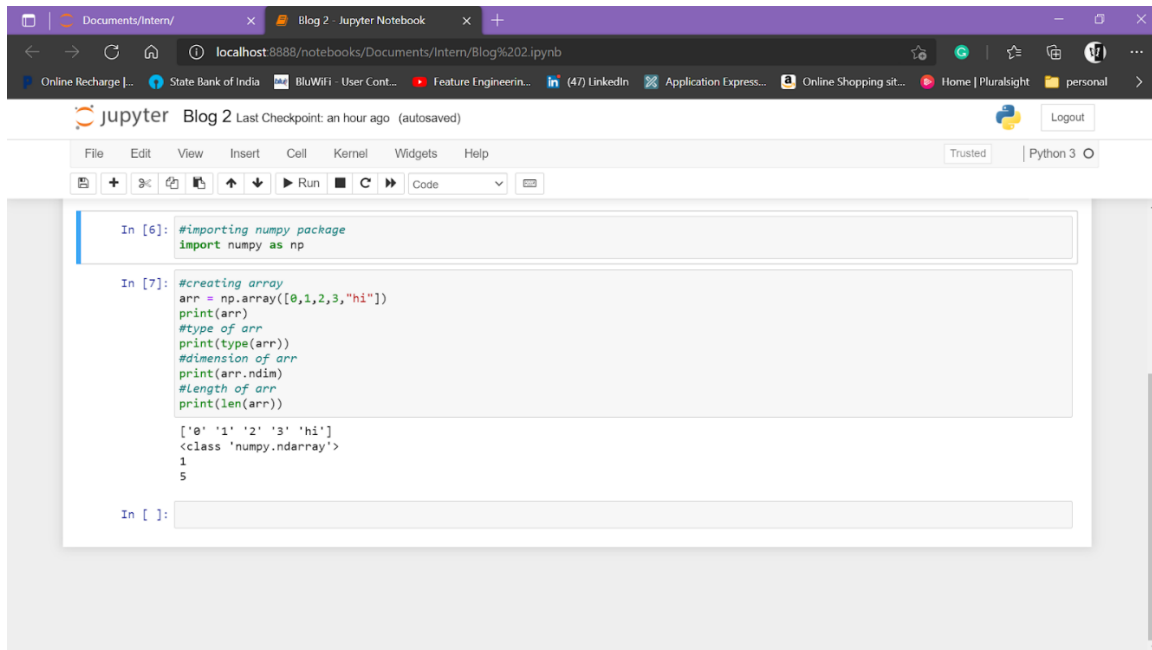
```
print(type(ar))
```

#dimension of ar

```
print(ar.ndim)
```

#length of arr

```
print(len(ar))
```



The screenshot shows a Jupyter Notebook window with the following code and output:

```
In [6]: #importing numpy package
import numpy as np

In [7]: #creating array
arr = np.array([0,1,2,3,"hi"])
print(arr)
#type of arr
print(type(arr))
#dimension of arr
print(arr.ndim)
#length of arr
print(len(arr))

[0 '1' '2' '3' 'hi']
<class 'numpy.ndarray'>
1
5

In [ ]:
```

Matplotlib

- o Matplotlib is the plotting library to create quality data like histogram, dissipation plot and so on...

- o It can be utilised in data visualisation as well.

Code:

#importing matplotlib library

```
import matplotlib.pyplot as pl
```

#plotting values

```
pl.plot([4,5,6],[15,20,25])
```

#title

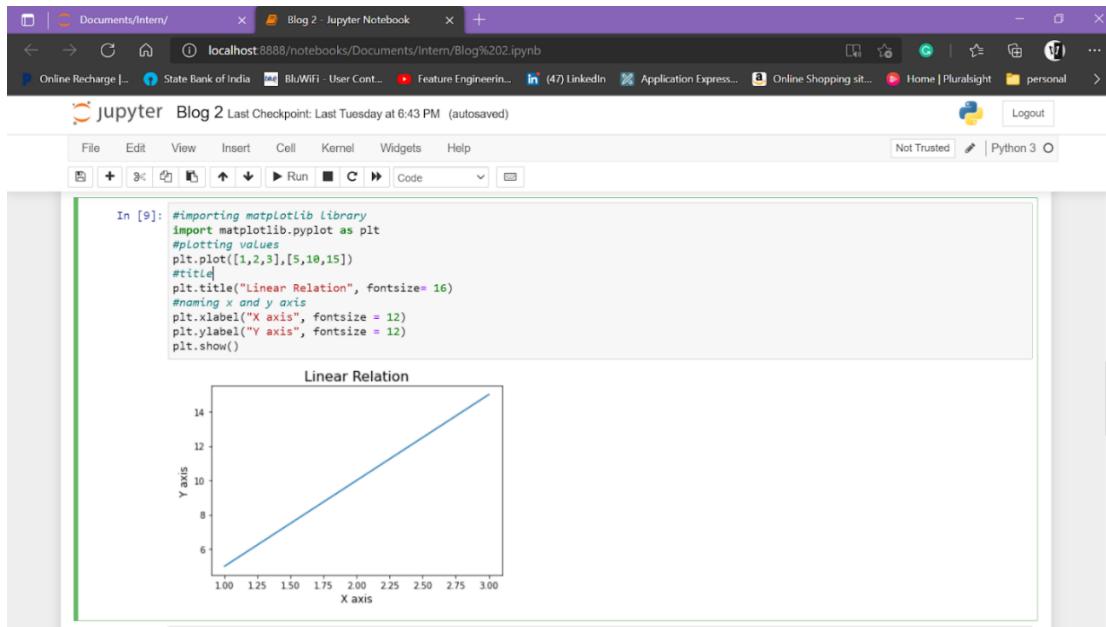

```
pl.title("Linear Relation", fontsize= 12)
```

```
#naming x and y axis
```

```
pl.xlabel("X axis", fontsize = 10)
```

```
pl.ylabel("Y axis", fontsize = 10)
```

```
pl.show()
```



Pandas

- o Pandas is an open-source library with High execution, simple to-utilize Data Structures, and Analysis tools for Python.
- o Data Science works like Calculating insights, cleaning information, and so forth...
- o It is profoundly utilized in Data Mining and Preparation however less in Data Modeling and Analysis.

Code:

```
#importing pandas library
```

```
import pandas as pd
```

```
dataset = pd.read_csv("xyz.csv")
```

```
#defining dataframe from 2 series
```

```
data = { 'cars' : [5,2,3], 'bus':[3,4,0]}
```

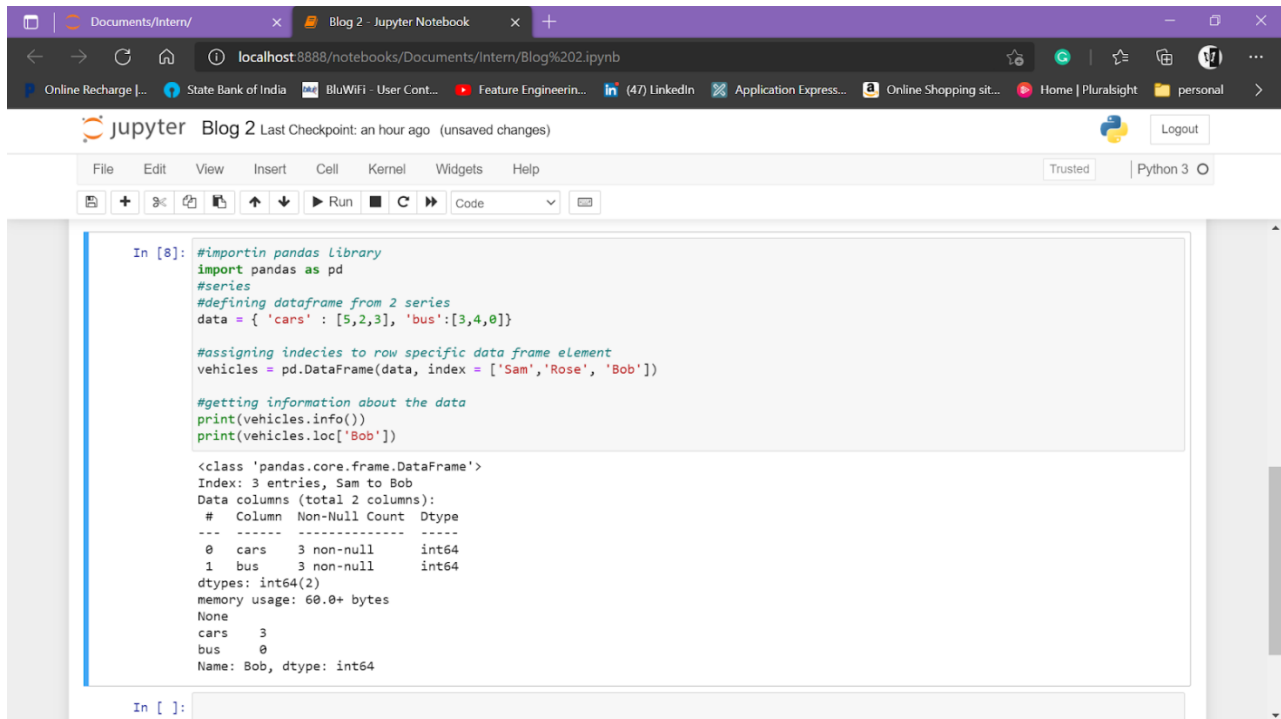
#assigning indices to row specific dataframe element

```
vehicles = pd.DataFrame(data, index = ['Sam','Rose', 'Bob'])
```

#getting information about the data

```
print(vehicles.info())
```

```
print(vehicles.loc['Bob'])
```



The screenshot shows a Jupyter Notebook interface with a code cell and its output. The code cell contains the following Python code:

```
In [8]: #importin pandas library
import pandas as pd
#series
#defining dataframe from 2 series
data = { 'cars' : [5,2,3], 'bus':[3,4,0]}

#assigning indecies to row specific data frame element
vehicles = pd.DataFrame(data, index = ['Sam','Rose', 'Bob'])

#getting information about the data
print(vehicles.info())
print(vehicles.loc['Bob'])
```

The output of the code cell is as follows:

```
<class 'pandas.core.frame.DataFrame'>
Index: 3 entries, Sam to Bob
Data columns (total 2 columns):
# Column Non-Null Count Dtype
-----
0 cars 3 non-null int64
1 bus 3 non-null int64
dtypes: int64(2)
memory usage: 60.0+ bytes
None
cars 3
bus 0
Name: Bob, dtype: int64
```

2.4 DATA PREPARATION

Whenever you've stacked your information into Pandas, there are a couple of basic things you can do quickly to tidy it up. For instance, you could:

- Remove any segments with over half missing qualities (if your dataset is adequately enormous - inclining further toward that in the following segment)
- Remove lines of redundant text that keeps the Pandas library from parsing information appropriately
- Remove any segments of URLs that you can't get to or that aren't valuable

After looking into it further of what every segment means and regardless of whether it's pertinent to your purposes, you could then take out any that:

- Are severely organized

- Contain unimportant or repetitive data
- Would need considerably more pre-processing work or extra information to deliver helpful (despite the fact that you might need to consider simple approaches to fill in the gaps utilizing secondary data)
- Leak future data which could interrupt the analytical components of your model

2.4.1 Dealing with Missing Data

In case you are managing an extremely enormous dataset, eliminating sections with a high extent of missing qualities will speed things up without harming or changing the general importance. This is pretty much as simple as utilizing Pandas' `.dropna()` function. For example, the accompanying content could get the job done:

```
df['column_1'] = df['column_1'].dropna(axis=0)
```

Nonetheless, it's additionally important the issue so you can recognize potential data sources to join with this dataset, to fill any gaps and advance your model later on.

In case you are utilizing a more modest dataset, or are generally stressed that dropping the example/quality with the missing qualities could debilitate or misshape your model, there are a few different techniques you can utilize. These include:

- Replacing the mean/middle/mode value for every null value (you can utilize `df['column'].fillna()` and pick `.mean()`, `.middle()`, or `.mode()` capacities to rapidly tackle the issue)
- Using direct relapse to credit the trait's missing values
 - If there is sufficient information that invalid or zero qualities will not affect your information, you can basically utilize `df.fillna(0)` to displace NaN esteems with 0 to take into consideration calculation.
 - Clustering your dataset into known classes and computing missing qualities utilizing inter cluster regression
 - Merging any of the above with dropping occurrences or properties dependent upon the situation

Consider cautiously concerning which of these methodologies will work best with the ML model you are setting up the information for. Decision trees don't take excessively generous to missing qualities, for instance. Note that, when utilizing Python, Pandas marks missing mathematical information with the coasting esteem point NaN (not a number). You can track down this exceptional worth characterized under the NumPy library, which you will likewise have to

import. The way that you have this default marker makes it significantly simpler to rapidly spot missing values and do an underlying visual evaluation of how broad the issue is.

Would it be a good idea for you to eliminate anomalies?

Before you can settle on this choice, you need to have a genuinely clear thought of why you have outliers. Is this the result of mistakes made during collection of data? Or then again is it a genuine anomaly, a valuable piece of information that can add something to your preparation?

One speedy approach to check is parting your dataset into quantiles with a straightforward content that will return Boolean upsides of True for anomalies and False for normal values:

```
import pandas as pd

df = pd.read_csv("data.csv")

Q1 = df.quantile(0.25)

Q3 = df.quantile(0.75)

IQR = Q3 - Q1

print(IQR)

print(df < (Q1 - 1.5*IQR)) | (df > (Q3 + 1.5*IQR))
```

You can likewise place your information into a box plot to effectively visualize the values of outliers:

```
df = pd.read_csv('dataset.csv')

plt.boxplot(df["column"])

plt.show()
```

You may likewise think that it is helpful to apply a log or square root change. This will limit the effect on the model if the outlier is a independent variable while assisting your assumptions with working better in case it's a reliant variable.

All things considered, the main thing is to think including or eliminating the outlier. Rather than attempting a one-size-fits-all methodology and afterward disregarding it, this will assist you with staying conscious of likely difficulties and issues in the model to examine with your associates and refine your methodology.

2.4.2 Transformation

Having fixed the issues above, you can start to divide your dataset into information and yield factors for ML and to apply a preprocessing change to your input variables.

Definitively what sort of changes you make will, obviously, rely upon what you plan to with the information in your ML model. A couple of choices are:

- Standardize the information

Best for: *logistic regression, linear regression, linear discriminate analysis*

On the off chance Gaussian distribution is present in any attribute of input variable in which either mean or standard deviation varies, you can utilize these procedures to normalize the mean to 0 and the standard deviation to 1. You can import the sklearn.preprocessing library to utilize its StandardScaler standardization tool:

```
from sklearn import preprocessing

names = df.columns

scaler = preprocessing.StandardScaler()

scaled_df = scaler.fit_transform(df)

scaled_df = pd.DataFrame(scaled_df, columns = names)
```

- Rescale the data

Best for *gradient descent (and other optimization algorithms), regression, neural networks, procedures that use distance measures, e.g. K-Nearest Neighbors*

This additionally includes normalizing data attributes with various scales so that they're all on a similar scale, ordinarily going from 0-1. (You can perceive how the scaling capacity functions in the model beneath.)

- Normalize the data

Best for: procedures that weight input values, for example neural networks, calculations that utilization distance measures, for example K-Nearest Neighbors

On the off chance that your dataset is exceptionally sparse and contains a stack of 0s, however the properties you do have utilize differing scales, you might have to rescale each column/perception so it has a unit standard/length of 1. It's significant, in any case, that to run standardization scripts, you'll likewise require the scikit-learn library (sklearn):

```
from sklearn import preprocessing

df = pd.read_csv('dataset.csv')

min_max_scaler = preprocessing.MinMaxScaler()

df_scaled = min_max_scaler.fit_transform(df)

df = pd.DataFrame(df_scaled)
```

The result is a table that has values normalized so that you can run them without getting extreme results.

- Make the Data Binary

Best for: feature engineering, transforming probabilities into clear values

This implies applying a binary threshold to information so that all values underneath the threshold become 0 and every one of those above it become 1. Indeed, we can utilize a scikit-learn device (Binarizer) to assist us with rapidly tackling the issue:

```
from sklearn.preprocessing import Binarizer

df = pd.read_csv('testset.csv')

#we're selecting the columns to binarize

age = df.iloc[:, 1].values

gpa = df.iloc[:, 4].values

#now we turn them into values we can work with

x = age

x = x.reshape (1, -1)

y = gpa

y =y.reshape (1, -1)

#we need to set a threshold to define as 1 or 0

binarizer_1 = Binarizer(35)
```

```
binarizer_2 = Binarizer(3)
```

```
#finally we run the Binarizer function
```

```
binarizer_1.fit_transform(x)
```

```
binarizer_2.fit_transform(y)
```

Your output will go from something like this:

Original age data values :

```
[25 21 45 ... 29 30 57]
```

Original gpa data values :

```
[1.9 2.68 3.49 ... 2.91 3.01 2.15]
```

To this:

Binarized age :

```
[[0 0 1 ... 0 0 1]]
```

Binarized gpa :

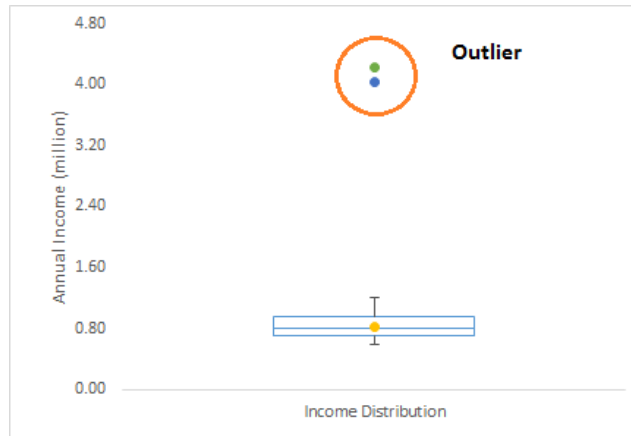
```
[[0 0 1 ... 0 1 0]]
```

... Don't neglect to sum up your information to feature the progressions before you continue on.

2.5 OUTLIER

Anomaly is a generally utilized phrasing by analysts and information researchers as it needs close consideration else it can result in wrong assessments. Essentially speaking, Outlier is a perception that shows up far away and deviates from general outline in a sample.

We should take a model, we do client profiling and discover that the normal yearly pay of clients is \$0.8 million. However, there are two clients having yearly pay of \$4 and \$4.2 million. These two clients yearly pay is a lot higher than rest of the population. These two perceptions will be viewed as Outliers.



Types of analysis

Exploratory Data Analysis (EDA) is the methodology for investigating the dataset to sum up its main attributes. The dataset outlines can be of 2 kinds,

1. Numerical Summary :Numerical outlines are outlines in Numbers. Ex: Mean(Average), Median, and so on... It can be all things considered

- Univariate – Measure depends just on one variable or
- Bivariate – measure depends on two factors.

2. Graphical Summary: Graphical outlines will be represented in forms of diagrams. Ex: Histogram, Box-plot, and so on...

We need to break down the information for the accompanying reasons:

1. Identifying dataset appropriation
2. Choosing the right Machine Learning procedure
3. Extracting Right attributes
4. Evaluate our ML procedure and introducing our outcomes

Univariate Numerical Analysis

Mean

Mean is characterized as the proportion of the values, all things considered, to the aggregate number of values. Mean is likewise called as Average of the dataset.

Mean = SUM OF ALL VALUES / TOTAL NUMBER OF VALUES

PROS	CONS
Consider all values	Mean is elusive for extreme values

Code:

#import library

import pandas as pd

#reading dataset

dataset = pd.read_csv("abc.csv")

#calculating mean

def mean(df):

return sum(dataset.age)/len(dataset)

print(mean(dataset))

Note:

Try Not to TRUST Mean!!

Median

Median is the rate isolating the lower half from the upper portion of the information.

Steps:

Steps:

1. Organize the information in Ascending order

2. On the off chance that the complete number of values is :

- ODD: Take the center number as Median
- EVEN: Take the normal of the center two numbers. (ie) Median = (Num 1 + Num 2)/2

PROS	CONS
Insensitive to Extreme Values	Does not consider dataset distribution

Code:

```
def median(dataset):
median = sorted(dataset) [len(dataset)// 2]
return median
```

Percentile

Percentile is the amount showing a specific level of the dataset is underneath the value!

25%, 50% (median), 75%

PROS	CONS
More expensive	Multiple measures

Code:

```
data = [13,14,15,16,20,95,66,88]
```

#25th percentile

```
sort_data = sorted(data)
index1 = len(sort_data)*.25
print(index1)
```

#50th percentile

```
sort_data = sorted(data)
index2 = len(sort_data)*.50
print(index2)
```

#75th percentile

```
sort_data = sorted(data)
index3 = len(sort_data)*.75
print(index3)
```

Since all the above strategies do have a few advantages and disadvantages. These techniques don't give us the specific outcome we are searching for. SO WHAT TO DO THEN??? LET'S SEE.

Standard Deviation

SD discloses to us the normal distinction between genuine values and mean.

CASE(i) – High standard deviation shows high scattering

CASE(ii) – Low standard deviation shows Low scattering

PROS	CONS
Consider all elements in the dataset.	Hard to compute
Consider all the distribution	–

Code:

```
Import numpy as np
```

```
array = [1,2,3,4,5,6]
```

```
print(numpy.std(array))
```

Other measures

1. Maximum and Minimum: Max and Min information in the dataset
2. Count: Counts the complete number of information focuses
3. Mode: Indicates esteems with high recurrence
4. Range: Range is characterized as the contrast among Maximum and Minimum qualities in the information
5. Outliers: Outlier is characterized as the point that lies at an unusual separation from different information focuses.

Bivariate Numerical Analysis

Bivariate Numerical Analysis is characterized as the best approach to recognize the connection between 2 variables.

Correlation

Correlation is the action characterizing that "how much at least 2 factors are connected".

- It discloses to us the level of the straight connection among x and y factors.
- It can be positive (solid) or negative or no connection.
- If the worth of relationship ranges,
 - o CASE(i)
 - Between 0 and 1 : Positive relationship
 - o CASE(ii)
 - 0
 - : No relationship
 - o CASE(iii)
 - Between - 1 and 0 : Negative relationship

Pearson Correlation

Among different strategies for relationship, PEARSONS CORRELATION is generally utilized for analysis.

Code:

```
import pandas as pd
#import dataset
df = pd.read_csv("filename.csv")
print(df.corr(method = "pearson"))
```

Note:

1. The correlation of a variable is always 1.
2. Machine Learning models work only with numbers

B.Sc.(DATA SCIENCE)

SEMESTER-III

DATA PREPARATION

UNIT III DATA PREPARATION AND CLEANSING

STRUCTURE

3.0 Objectives

3.1 Data Formats

3.1.1 File formats and Reading Files in Python

3.1.1.1 Comma Separated Values (CSV)

3.1.1.2 XLSX files

3.1.1.3 ZIP files

3.1.1.4 Plain Text (txt)

3.1.1.5 JSON

3.1.1.6 XML

3.1.1.7 Image files

3.1.1.8 Hierarchical Data Format (HDF)

3.1.1.9 PDF

3.1.1.10 DOCX

3.1.1.11 MP3

3.1.1.12 MP4

3.2 Data Transformation

3.2.1 Extraction and Parsing

3.2.2 Filtering, Aggregation, and Summarization

3.2.3 Enrichment and Imputation

3.2.4 Indexing and Ordering

3.2.5 Anonymization and Encryption

3.2.6 Modeling, Typecasting, Formatting, and Renaming

3.2.7 Refining the data Transformation Process

3.3 Data Cleaning

3.3.1 Removing Null/Duplicate Records

3.3.2 Dropping unnecessary Columns

3.3.3 Renaming Columns

3.3.4 Treating Missing Values

3.3.5 Outlier Detection

3.3.6 Reshape/restructuring the data

3.4 The K-Means Algorithm

3.5 Summary

3.0 OBJECTIVES

Understanding data formats. Analyzing scalability and real-time issues in processing of data. Conducting Data Cleaning segregating heterogeneous and identifying missing values in the datasets.

3.1 DATA FORMATS

The format of a file is a standard process in which data is stored in a file. Firstly, the format of a file defines whether the file is an ASCII or binary file. Secondly, it illustrates how the data is organized. For example, tabular data in plain text is store in comma-separated values (CSV) file format.

To recognize the file format, you can observe the file extension. For example, a file with name “Information” in “CSV” format will seem as “Information.csv”. By seeing “.csv” extension we can recognize that it is a “CSV” file and the data is arranged in a tabular format.

CSV				
	A	B	C	D
1	ID	Gender	City	Monthly
2	ID000002C	Female	Delhi	20000
3	ID000004E	Male	Mumbai	35000
4	ID000007H	Male	Panchkula	22500
5	ID000008I	Male	Saharsa	35000
6	ID000009J	Male	Bengaluru	100000
7	ID000010K	Male	Bengaluru	45000
8	ID000011L	Female	Sindhudui	70000
9	ID000012N	Male	Bengaluru	20000
10	ID000013M	Male	Kochi	75000
11	ID000014C	Female	Mumbai	30000
12	ID000016C	Male	Mumbai	25000
13	ID000018S	Female	Surat	25000
14	ID000019T	Female	Pune	24000
15	ID000021V	Male	Bhubanes	27000
16	ID000022V	Female	Howrah	28000

JSON	
<pre>"Employee": [{ "id": "1", "Name": "Ankit", "Sal": "1000", }, { "id": "2", "Name": "Faizv",</pre>	<pre><?xml version="1.0"?> <contact-info> <name>Ankit</name> <company>Analytics Vidhya</company> <phone>+9187654321</phone> </contact-info></pre>

Usually, the files you may stumble upon will rely on the application you are developing. For instance, in an image processing system, you will require image files as an input and output. So you will generally see files in gif, jpeg, or png format.

As a data scientist, you require to identify the fundamental structure of various file formats, their benefits and drawbacks. If you are not familiar to the fundamental structure of the data, you will not be capable to explore it. Likewise, at times you need to make conclusions about how to save data.

Choosing the optimal file format for saving data can improve the performance of your models in data processing.

Now, look at the underlying file formats and so that you understand their use in Python:

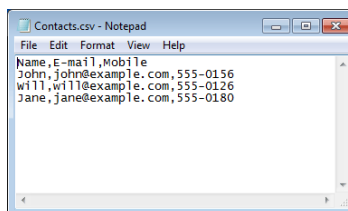
- Comma-separated values
- Plain Text (txt)
- Images
- DOCX
- MP3
- XLSX
- Hierarchical Data Format
- ZIP
- PDF
- XML
- HTML
- JSON
- MP4

3.1.1 File formats and reading files in Python

3.1.1.1 Comma Separated Values (CSV)

Comma Separated Values file format are part of spreadsheet file format. In spreadsheet file format, information is stored in cells. Every cell is organized in rows and columns. In the spreadsheet file, a column can have numerous types. For instance, a column can be of an integer type, a date type or a string type, a date type. Some of the most prevalent spreadsheet file formats are Microsoft Excel Spreadsheet (xls), Microsoft Excel Open XML Spreadsheet (xlsx) and Comma Separated Values (CSV). Every line in CSV file signifies an observation or usually called a record. Every record may have one or more fields which are parted by a comma. Sometime you may encounter some files where fields are separated by tab instead of comma. This type of file format is well-known as Tab Separated Values (TSV) file format.

The underneath image shows a CSV file which is opened in Notepad.



Reading the data from CSV in Python

The aforementioned will show how to read a CSV file in Python. Loading data can be done through “Pandas” library in python.

```
import pandas as pd
```

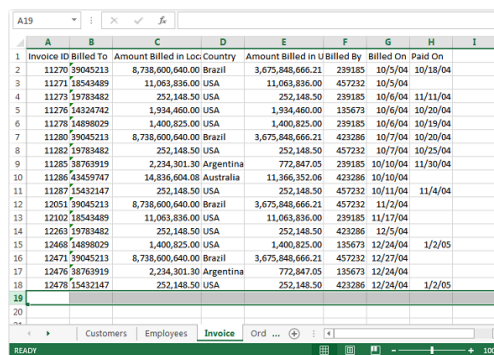


```
df = pd.read_csv("/home/Loan_Prediction/bus.csv")
```

Above code will load the train.csv file in DataFrame df.

3.1.1.2 XLSX files

Microsoft Excel Open XML file format short form is XLSX. XLSX is under the Spreadsheet file format. XML based file format was created by Microsoft Excel. XLSX file format was introduced by Microsoft Office 2007. XLSX data is organized under the heading columns and cells in a sheet. One or more sheets are there in every XLSX file. That's why a workbook can have multiple sheets. The underneath image shows a "xlsx" file which is opened in Microsoft Excel.



Invoice ID	Billed To	Amount Billed in Loc	Country	Amount Billed in U	Billed By	Billed On	Paid On
11270	39045213	8,738,600.640.00	Brazil	3,675,848,666.21	239185	10/5/04	10/18/04
11271	18543489	11,063,836.00	USA	11,063,836.00	457232	10/5/04	
11273	19783482	252,148.50	USA	252,148.50	239185	10/6/04	11/11/04
11276	14124742	1,934,460.00	USA	1,934,460.00	135673	10/6/04	10/29/04
11278	14898029	1,400,825.00	USA	1,400,825.00	239185	10/6/04	10/19/04
11280	39045213	8,738,600.640.00	Brazil	3,675,848,666.21	423286	10/7/04	10/29/04
11282	19783482	252,148.50	USA	252,148.50	457232	10/7/04	10/29/04
11285	38763919	2,234,301.30	Argentina	772,847.05	239185	10/10/04	11/30/04
11286	43459747	14,836,604.08	Australia	11,366,352.06	423286	10/10/04	
11287	15432147	252,148.50	USA	252,148.50	457232	10/11/04	11/4/04
12051	39045213	8,738,600.640.00	Brazil	3,675,848,666.21	457232	11/2/04	
12102	18543489	11,063,836.00	USA	11,063,836.00	239185	11/17/04	
12263	19783482	252,148.50	USA	252,148.50	423286	12/5/04	
12468	14898029	1,400,825.00	USA	1,400,825.00	135673	12/24/04	1/2/05
12471	39045213	8,738,600.640.00	Brazil	3,675,848,666.21	457232	12/27/04	
12476	38763919	2,234,301.30	Argentina	772,847.05	135673	12/24/04	
12478	15432147	252,148.50	USA	252,148.50	423286	12/24/04	1/2/05

Reading the data from XLSX file

Open the XLSX file for fetching the data, and renaming the sheet name. The data can be loaded using Pandas Python package.

```
import pandas as pd
```

```
df = pd.read_excel("/home/Loan_Prediction/bus.xlsx", sheetname = "Receipt")
```

The code will upload the sheet "Receipt" from "bus.xlsx" file in DataFrame df.

3.1.1.3 ZIP files

An archive file format is ZIP format that is why a ZIP file is a lossless compression format. It means that after compressing several files with ZIP, you can fully recover the data later by decompressing the ZIP file. The ZIP file format compresses the file content using a variety of compression algorithms. A ZIP file can be easily recognized by .zip extension.

Reading a .zip file in Python

Import the "zipfile" package. It allows you to read a zip file. The python code for reading the "bus.csv" file contained within the "B.zip" is provided below.

```
import zipfile
archive = zipfile.ZipFile('B.zip', 'r')
df = archive.read('bus.csv')
```

3.1.1.4 Plain Text (txt) file format

In Plain Text file format, data is written in plain text, in the Plain Text file format. Typically, this text is in an unstructured format with no meta-data attached with it. Any application may read a text file in the txt format. However interpreting this is very problematic by a computer program. Let's take a simple example of a text File.

The following example shows text file data that contain text:

“In my previous article, I introduced you to the basics of R-Programming, different data representations (DataFrame / Dataset) and basics of operations . We had also solved a machine learning problem from one of our past competition. In this article, I will continue from the place I left in my previous article.”

If the above content is stored in a file called “text.txt” and you wish to read it, use the code below.

```
text_file = open("text.txt", "r")
lines = text_file.read()
```

3.1.1.5 JSON file format

JavaScript Object Notation(JSON) is a text-based open standard designed for exchanging the data over web. The JSON format is used to send structured data over the internet. Because it is a language-independent data format, the JSON file format can be read by any programming language. Let's take an example of a JSON file

The following example shows how a typical JSON file stores information of employees.

```
{ "Employee": [
  { "id": "1",
    "Name": "Ankit",
    "Sal": "1000",
  },
  { "id": "2",
    "Name": "Faizy",
```

```
        "Sal": "2000",
    }
]
}
```

Reading a JSON file

Let's have a look at how to load data from a JSON file. The pandas library in Python can be used to load the data.

```
import pandas as pd

df = pd.read_json("/home/kunal/Downloads/Loan_Prediction/train.json")
```

3.1.1.6 XML file format

Extensible Markup Language (XML) is another name for XML. It's a markup language, as the name implies. It follows a set of rules when it comes to data encoding. XML is a file format that is both human and machine readable file format. XML is a self-descriptive language that is used to transport data across the internet. XML is comparable to HTML but differs in a few ways. For instance, XML does not use predefined tags as HTML.

Let's take the simple example of XML File format.

The following example shows an xml document that contains the information of an employee.

```
<?xml version="1.0"?>
<contact-info>
<name>Ankit</name>
<company>University</company>
<phone>+91987654321</phone>
</contact-info>
```

The “<?xml version=“1.0“?>” is a XML declaration at the start of the file (it is optional). In this iteration, *version* specifies the XML version and *encoding* specifies the character encoding used in the document. <contact-info> is a tag in this document. Each XML-tag needs to be closed.

Reading XML in python

You can use xml.etree to read data from an XML file. ElementTree library.

Let's load an xml file called train and print its root tag.

```
import xml.etree.ElementTree as ET
tree = ET.parse('/home/sunilray/Desktop/2 sigma/train.xml')
root = tree.getroot()
print root.tag
```

3.1.1.7 Image files

The most exciting file type utilised in data science is undoubtedly image files. Image processing is the foundation of any computer vision application. As a result, familiarity with several image file format is required.

The most common image files are 3-dimensional and have RGB values. They can, however, be 2-Dimensional (grayscale) or 4-Dimensional (having intensity) — an image made up of pixels and meta-data related to it.

Each image is made up of one or more pixels frames. Each frame consists of a two-dimensional array of pixel values. The intensity of pixel values can be any. An image's meta-data can include things like the image type (.png) or pixel dimensions.

Let's take the example of an image by loading it.

```
from scipy import misc
f = misc.face()
misc.imsave('face.png', f) # uses the Image module (PIL)
import matplotlib.pyplot as plt
plt.imshow(f)
plt.show()
```



Now, let's check the type of this image and its shape.

```
type(f) , f.shape
```

```
numpy.ndarray,(768, 1024, 3)
```

3.1.1.8 Hierarchical Data Format (HDF)

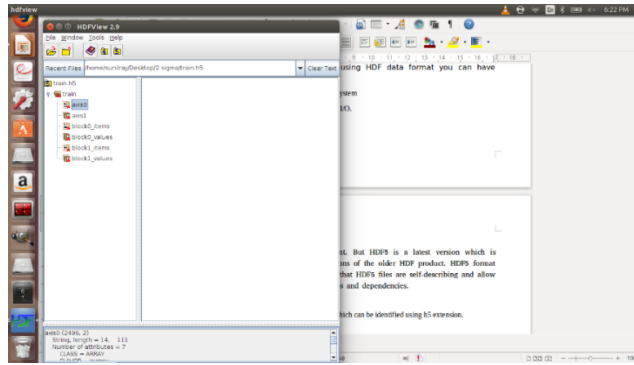
You may easily store a big amount of data in Hierarchical Data Format (HDF). It's not just for storing large amounts of data or complex data; it's also for storing tiny amounts of data or simple data.

The merits of using HDF are as mentioned below:

- It can be utilized in any system of any size or type.
- It has a versatile, efficient storage system as well as fast I/O.
- Numerous formats support HDF.

There are various HDF formats present. However, HDF5 is the most recent version, which is intended to address some of the shortcomings of previous HDF file formats. The HDF5 format is comparable to XML in certain ways. HDF5 files are self-descriptive, similar to XML, and allow users to specify complicated data relationships and dependencies.

Let's take the example of an HDF5 file format which can be recognized using .h5 extension.



Read the HDF5 file

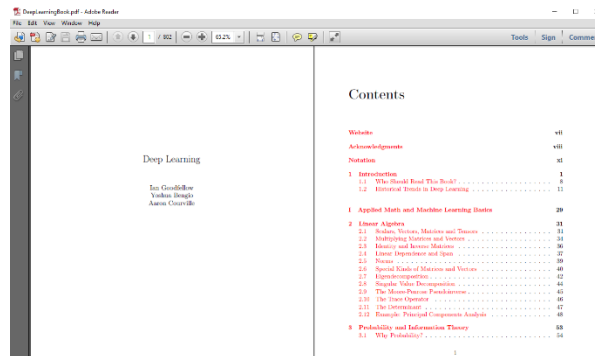
The HDF file can be read using pandas. Below is the python code can load the train.h5 data into the “t”.

```
t = pd.read_hdf('train.h5')
```

3.1.1.9 PDF file format

The Portable Document Format (PDF) is a very handy format for interpreting and displaying text documents with embedded graphics. A PDF file's unique characteristic is that it can be password-protected.

Here’s an example of a pdf file.



Reading a PDF file

Reading a PDF file through a programme, on the other hand, is a difficult task. Although there are libraries that succeed in parsing PDF files, PDFMiner is one of them. To read a PDF file using PDFMiner, you must first:

- Download PDFMiner and install it through the [website](#)
- Extract PDF file by the following code

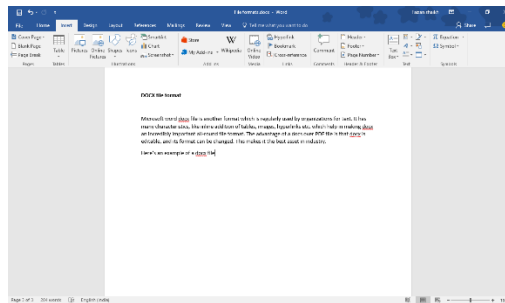
pdf2txt.py <pdf_file>.pdf

3.1.1.10 DOCX file format

Another file type commonly used by organizations for text-based data is the Microsoft Word docx file. It contains a number of features, such as the ability to incorporate tables, graphics, and hyperlinks inline, that serve to make docx an extremely essential file format.

A docx file has an advantage over a PDF file that it can be edited. A docx file can also be converted to any other format.

Here's an example of a docx file:



Reading a docx file

Python offers a community contributed library to parse a docx file, similar to PDF format. It's known as python-docx2txt.

Installing this library is easy through pip by:

```
pip install docx2txt
```

To read a docx file in Python use the following code:

```
import docx2txt  
  
text = docx2txt.process("file.docx")
```

3.1.1.11 MP3 file format

The MP3 file format is classified as a multimedia file format. Multimedia file formats are similar to image file formats, however they are one of the most complex to understand. Multimedia file formats may store a wide range of data, including text, image, graphical, video, and audio data. A multimedia format, for example, can allow text to be stored as RTF data rather than ASCII data, which is a plain-text format.

MP3 is one of the most widely used digital audio coding formats. The MPEG-1 (Moving Picture Experts Group – 1) encoding format, which is a standard for lossy video and audio compression, is used in mp3 files. Once you've compressed the original file with lossy compression, you won't be able to restore the original contents.

The mp3 file format reduces the quality of audio by filtering out audio that humans cannot hear. MP3 compression often achieves a size reduction of 75 to 95 percent, which saves a lot of space.

mp3 File Format Structure

A mp3 file is made up of numerous frames. A frame can be further divided into a header and data block. These frame sequences are referred to as an elementary stream.

A header in mp3 normally, defines the starting of a valid frame and a data blocks contain the (compressed) audio information in terms of amplitudes and frequencies. If you want to know more about mp3 file structure you can refer this [link](#).

Reading the multimedia files in python

For reading or manipulating the multimedia files in Python you can use a library called [PyMedia](#).

3.1.1.12 MP4 file format

Videos and movies are stored in the MP4 file format. It consists of a series of images (called frames) that play in the form of a video over a set of time. There are two methods for interpreting a mp4 file. One is a closed entity, which considers the entire video as a single entity. Another method is mosaic of images, in which each image in the video is treated as a separate entity, with these images being sampled from the video.

Here's is an example of mp4 video

Reading an mp4 file

MoviePy, a community-built library for reading and modifying MP4 files, is another feature of MP4.

You can install the library from this [link](#). To read a mp4 video clip, in Python use the following code.

```
from moviepy.editor import VideoFileClip
clip = VideoFileClip('<video_file>.mp4')
```


You can then display this in jupyter notebook as below

```
ipython_display(clip)
```

Information analysis demands structured and accessible data for the best results. Data transformation allows organisations to change the structure and format of raw data as needed.

3.2 DATA TRANSFORMATION

Data transformation is the process of changing the format, structure, or values of data. For data analytics initiatives, data can be updated at two stages of the data pipeline. ETL (extract, transform, load) is a common method used in on-premises data warehouses, with data transformation functioning as the middle stage. Cloud-based data warehouses, which can boost processing and storage resources in seconds or minutes, are now used by the majority of organization. Because of the cloud platform's scalability, organization can forego preload transformations and instead load raw data into the data warehouse, which they can subsequently modify at query time — a model called ELT (**extract, load, transform**).

Data transformation may be constructive (adding, copying, and replicating data), destructive (deleting fields and records), aesthetic (standardizing salutations or street names), or structural (renaming, moving, and combining columns in a database).

Benefits and challenges of data transformation

Transforming data yields several benefits:

- To make data more organized, data is transformed. Humans and computers may find it easier to use transformed data.
- Null values, unexpected duplicates, erroneous indexing, and incompatible formats are all possible landmines in applications that aren't properly prepared and validated.
- Data transformation makes it easier for applications, systems, and different types of data to work together. Data that is utilized for several purposes may require different transformations.

Data transformation can help analytic and business processes run more efficiently and allow for improved data-driven decision-making. Data type conversion and flattening of hierarchical data should be included in the first phase of data transformations. These processes alter data in order to make it more compatible with analytics software. Additional transformations can be applied as

needed by data analysts and data scientists as distinct layers of processing. Each processing layer should be built to accomplish a specified set of operations in order to satisfy a known business or technical requirement.

Within the data analytics stack, data transformation serves a variety of purposes.

3.2.1 Extraction and parsing

Data ingestion in the current ELT process starts with extracting information from a data source, then copying the data to its destination. The first transformations focus on shaping the data's format and structure to ensure compliance with both the destination system and the data currently on hand. This sort of data transformation includes extracting fields from comma-delimited log data for loading into a relational database.

3.2.2 Filtering, aggregation, and summarization

The goal of data transformation is to reduce the size of data and make it more manageable. Filtering out unneeded fields, columns, and records can help to consolidate data. Numerical indices in data meant for graphs and dashboards, or records from business locations that aren't relevant to a given study, are examples of omitted data.

Although BI systems can perform this filtering and aggregation, it may be more effective to perform the transformations before accessing the data with a reporting tool.

3.2.3 Enrichment and imputation

Data from several sources can be combined to produce denormalized, enriched data. The transactions of a customer can be rolled up into a grand total and stored in a customer information table for easy reference or use by customer analytics tools. As a result of these modifications, long or freeform fields can be split into many columns, and missing values or corrupted data can be imputed or replaced.

3.2.4 Indexing and ordering

Creating indexes in relational database management systems can boost performance or improve the management of relationships between tables.

3.2.5 Anonymization and encryption

Before being distributed, data containing personally identifiable information or other information that can compromise privacy or security should be anonymized. Many industries demand encryption of confidential data, and systems can encrypt data at multiple levels, from individual database cells to entire records or fields.

3.2.6 Modeling, typecasting, formatting, and renaming

Finally, a number of transformations can be used to reshape data without changing its content. This involves casting and converting data types for compatibility, renaming schemas, tables, and columns for clarity, and modifying dates and timings with offsets and format localization.

3.2.7 Refining the data transformation process

You must duplicate the data to a data warehouse architected for analytics before your company can perform analytics, and even before you transform the data. Most companies today opt for a cloud data warehouse, which allows them to fully utilise ELT. Stitch can load all of your data in its raw state to your selected data warehouse, ready for transformation.

One of the most difficult tasks for a data scientist is cleaning the data before diving into it for valuable insights. One of the most critical steps that should never be overlooked is data cleaning. The accuracy of your model will be compromised if the data is not fully cleansed.

Due to poor data quality, outcomes are skewed, with low accuracy and significant error percentages. As a result, fully cleaning the data before fitting a model to it is essential. As a data scientist, it's critical to recognise that not all of the data we're given is helpful, and we need to know how to deal with it.

3.3 DATA CLEANING

Data cleaning entails removing null records, eliminating redundant columns, handling missing values, rectifying trash values (also known as outliers), reorganising the data to make it more legible, and so on.



The use of data cleaning in data warehouses is one of the most common examples. Before any model fitting can be done, a data warehouse stores a range of data from many sources and optimizes it for analysis.

Data cleaning isn't only about removing old data to make room for new data; it's also about figuring out how to increase the correctness of a data set without losing the old data. Different sorts of data will necessitate different forms of cleansing, but the correct method will always be the deciding factor.

The data will become consistent with other similar data sets in the system after it has been cleansed.

3.3.1 Removing Null/Duplicate Records

If a large quantity of data is missing from a single row, that row should be dropped because it adds no value to our model. You can substitute an appropriate value for the missing data by imputing the value. Always remember to remove duplicate/redundant values from your dataset, as they may cause your model to be biased.

For example, let us consider the student dataset with the following records.

name	score	address	height	weight
A	56	Goa	165	56
B	45	Mumbai	3	65
C	87	Delhi	170	58
D				
E	99	Mysore	167	60

As we see that corresponding to student name “D”, most of the data is missing hence we drop that particular row.

```
student_df.dropna() # drops rows with 1 or more Nan value
```

```
#output
```

name	score	address	height	weight
A	56	Goa	165	56
B	45	Mumbai	3	65
C	87	Delhi	170	58
E	99	Mysore	167	60

3.3.2 Dropping unnecessary Columns

We receive a lot of data from stakeholders, and it's usually a lot. There could be a trove of data that adds no value to our model. Such information should be eliminated because it wastes memory and processing time.

For example, while looking at students' performance over a test, students' weight or their height does not have anything to contribute to the model.

```
student_df.drop(['height','weight'], axis = 1,inplace=True) #Drops Height column form the dataframe
```

#Output

name	score	Address
A	56	Goa
B	45	Mumbai
C	87	Delhi
E	99	Mysore

3.3.3 Renaming columns

It's always ideal to rename columns and format them in the most understandable format that both the data scientist and the business can understand. For instance in the student data set, renaming the column “name” as “Sudent_Name” makes it meaningful.

```
student_df.rename(columns={'name': 'Student_Name'}, inplace=True) #renames name column to Student_Name
```

#Output

Student_Name	score	address
A	56	Goa
B	45	Mumbai
C	87	Delhi
E	99	Mysore

3.3. 4 Treating missing values

There are numerous approaches to dealing with missing values in a dataset. It is up to the data scientist and the dataset at hand to determine which strategy is most appropriate. The most widely used methods are imputing the dataset with mean, median, or mode. Missing values are treated by removing records with one or more missing values and, in some situations, by using machine learning algorithms such as linear regression and K nearest neighbour.

Student_Name	score	address
A	56	Goa

B	45	Mumbai
C		Delhi
E	99	Mysore

`Student_df['col_name'].fillna((Student_df['col_name'].mean()), inplace=True) # Na values in col_name is replaced with mean`

#Output

Student_Name	score	address
A	96	Goa
B	45	Mumbai
C	66	Delhi
E	99	Mysore

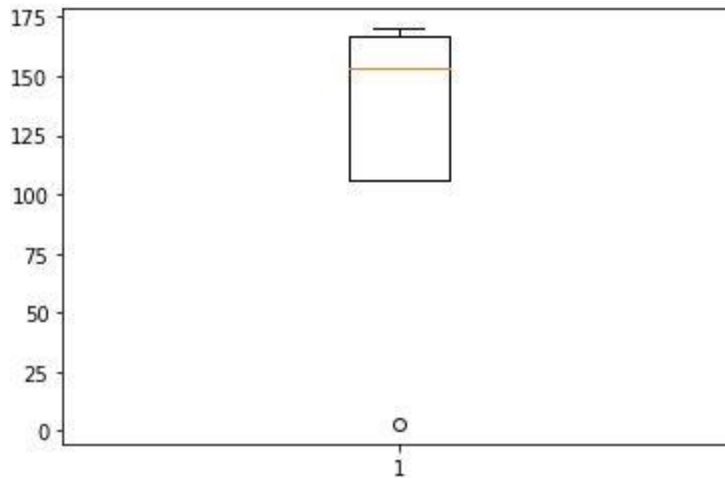
3.3.5 Outlier Detection

Outliers can be measured as noise in the dataset. There can be various reasons for outliers like manual error, data entry error, experimental error, etc.

In the below example, the score of student "B" is entered 130 which is clearly not correct.

Student_Name	score	address	height	weight
A	56	Goa	165	56
B	45	Mumbai	3	65
C	66	Delhi	170	58
E	99	Mysore	167	60

Plotting height on a box plot gives the below result



Not all extreme values are outliers; some can lead to surprising results. Outliers can be found using tests such as the Z score test, box plot, or simply putting the data on a graph.

3.3.6 Reshape/restructuring the data

The majority of the business data sent to the data scientist is in an unreadable format. It is our responsibility to restructure the data and convert it to a format that can be analysed. For example, combining two or more variables or constructing a new variable from existing variables.

One of the most difficult challenges that marketing teams have is allocating resources to reduce CPA – cost per acquisition – to a minimal while increasing return. This is made feasible through segmentation, a strategy for dividing clients into distinct groups based on their characteristics or behaviour. Machine learning customer segmentation can help you save money on marketing activities by minimising waste. If you know which consumers are similar to each other, you'll be able to better target your advertising to the right people. Customer segmentation can also help with other marketing tasks including up-selling, product recommendations, and pricing. Customer segmentation used to be a time-consuming and challenging process that needed hours of human work searching through multiple tables and querying the data in the hopes of finding ways to group consumers together.

In recent years, machine learning, or artificial intelligence algorithms that recognise statistical regularities in data, has made it simpler. Machine learning models can be used to process customer data, which can then be utilised to detect repeated trends across a variety of variables.

Marketing analysts can use machine learning algorithms to find client subgroups that are difficult to recognise using intuition and manual data analysis. Customer segmentation is an excellent example of how artificial intelligence and human intuition may combine to produce something better.

3.4 THE K-MEANS ALGORITHM

Machine learning algorithms are available in a variety of flavours, each suited to a certain task. K-means clustering is one of the strategies that may be used to segment customers.

Unsupervised algorithms don't have any labelled data to compare their performance to.

The basic concept underlying k-means is to group data into clusters that are more similar

In this situation, the similarity across clusters is calculated using the difference in the consumers' age, income, and expenditure scores. When training a k-means model, you provide the number of clusters into which your data should be divided. The model starts with centroids, which are variables that are randomly placed and identify the centre of each cluster.

The training data is examined by the model, which assigns them to the cluster with the closest centroid. The centroids' parameters are re-adjusted to put them in the middle of their clusters after all of the training and examples have been categorised. The elbow approach, in addition to k-means clustering, is a useful data segmentation technique for obtaining the optimal number of machine learning segmentation clusters.

Importance of segmentation

Once trained, your machine learning model can determine which segment incoming clients belong to by analysing their distance from each of the cluster centroids. This can be used in a variety of ways.

Your machine learning model will help you identify your client's segment and the most common products associated with that segment.

Customer segmentation using machine learning algorithms will help you fine-tune your product marketing strategy. For example, you may start an ad campaign using a random sample of customers from different segments. After some time has passed, you may examine which groups are the most active and adapt your strategy to only show advertisements to individuals who are members of those groups. You may also run many versions of your campaign and utilise machine learning to segment your customers based on their responses to different initiatives. In general, you'll have a lot more tools for testing and fine-tuning your ad campaigns.

- **K-means clustering is a machine learning segmentation method that is quick and efficient**

It is not, however, a magic wand that will instantly transform your data into logical client categories. You must first choose your marketing target demographic and the factors that will be significant to them. Geographic location will be useless if your adverts are focused on specific places, for example, and you'll be better off filtering your data for that region.

If you're selling a men's health product, you should filter your client data to only include men and avoid including gender as an attribute in your machine learning model. In some cases, you may wish to include extra details, such as items they've already purchased. In this case, you'll need to create a customer-product matrix, which is a table with customers as rows and things as columns, as well as the number of items purchased at the intersection of each customer and item.

Consider establishing an embedding, in which the products are represented as values in a multidimensional vector space, if there are too many products. Machine learning is a great tool for customer segmentation and marketing in general. It will almost definitely never be able to completely replace human intuition and decision-making, but it can assist humans in reaching previously unachievable goals.

B.Sc.(DATA SCIENCE)

SEMESTER-III

DATA PREPARATION

UNIT IV RESEARCH DESIGN

STRUCTURE

4.0 Objectives

4.1 Introduction

4.2 Need For Research Design

4.3 Features Of A Good Research Design

4.4 Types Of Research Design

4.5 Concepts Relating To Research Design

4.6 Common Applications Of Cluster Analysis

4.7 Types Of Cluster Analysis

4.8 K-Means Clustering

4.9 Agglomerative Clustering

4.10 Summary

4.0 OBJECTIVES

- Need for Research Design.
- Understanding Features of a Good Design
- Important Concepts relating to Research Design
- Understanding Different Research Designs.
- Understanding cluster analysis, Clustering algorithms and agglomerative clustering

4.1 INTRODUCTION

Research design is the system of exploration strategies and procedures preferred by a specialist. The design allows specialists to focus on research techniques that are appropriate for the topic and laid out the groundwork for investigations.

The design of a research subject clarifies the sort of research (experimental, survey, correlational, semi-experimental, review) and furthermore its sub-type (experimental design, research problem, descriptive case-study). There are three principle sorts of research design: Data collection, measurement, and analysis.

The kind of research problem an association is confronting will decide the research design. The design phase figures out which tools to utilize and how they are utilized.

A research typically makes a base inclination in information and extends trust in the precision of gathered information. The desired outcome is considered when a design creates a minimal margin of error in experimental research is. The fundamental components are:

1. Accurate reason explanation
2. Techniques to be executed for gathering and examining research
3. The technique applied for examining gathered details.
4. Type of research procedure.
5. Probable doubts for research
6. Settings for the examination study
7. Timeline
8. Measurement of investigation

Fruitful examination studies give knowledge that are precise and fair. You'll have to make a study that meets the entirety of the primary qualities of a design.

A researcher should have a clear perception of the different kinds of research design to choose which model to execute for study. The design study can be extensively characterized into quantitative and subjective.

Qualitative: Qualitative examination decides connections between gathered information and perceptions dependent on mathematical calculation. Hypotheses related with a normally existing occurrence can be demonstrated or discredited utilizing statistical techniques. Analysts depend on

qualitative research techniques that determine "why" a specific hypothesis exists alongside "what" respondents need to say about it.

Quantitative: Quantitative examination is for cases where factual ends to gather significant experiences are important. Numbers give a superior point of view to settle on basic business choices. Quantitative exploration techniques are essential for the development of any association. Experiences drawn from hard mathematical information and investigation demonstrated being exceptionally convincing when decisions are to be made regarding future business.

4.2 NEED FOR RESEARCH DESIGN

The Significance of good Research design is required in light of the fact that it works with the going great of the different research processes, subsequently making research as proficient as conceivable yielding maximal data with insignificant use of money, time and effort. Similarly, affordable and alluring development of a house, we need a prototype (for sure is ordinarily called the house map) thoroughly examined and ready by expert, likewise we require a research plan ahead of data collection and exploration for our project. Research design represents early arrangement of the strategies to be embraced for gathering the applicable information and the procedures to be utilized in their research, considering the goal of the research. Arrangement of the design of research ought to be finished with extraordinary consideration as any mistake in it might disturb the whole task.

Research design conveys a significant effect on the dependability of the outcomes accomplished. It hence gives a strong base to the entire examination. It is required because of the way that it takes into account the smooth working of the many research operations. This makes the research as successful as conceivable by giving most extreme data with least amount of time, effort and money spending. For working of a vehicle, we should have a suitable design made by a knowledgeable designer. Likewise, we require a reasonable plan before collection of data and scrutiny of the research project. Arranging of outline should be done carefully as even a little error may wreck the reason for the whole venture. The design assists the agent with getting sorted out his thoughts, which assists with perceiving and fix his shortcomings, assuming any.

In a decent research design, every one of the component go along with one another intelligently. Similarly, the techniques of gathering data should fit with the study purposes, applied and hypothetical system and technique for analysis of data.

The significance of research design in research technique is because of the accompanying factors:

- It might bring about the favored sort of study with supportive conclusion.
- It eliminates mistake.
- Allows you get ideal effectiveness and dependability.
- Decrease consumption of time.

- Reduce vulnerability and confusion identified with any research problem.
- Of extraordinary assistance for assortment of research material and hypothesis testing.
- It is an advisor for giving the correct way to research.
- Gets unrestricted of partiality and minor errors.
- Provides a thought concerning the sort of resources required such as time, money, manpower and effort.
- Smooth and productive sailing.
- Maximizes dependability of results.
- Provides firm establishment to the undertaking.
- Averts misdirecting inferences and negligent futile exercise.
- Provides freedom to expect defects and insufficiencies (expects issues).
- Incorporates by gaining from others' basic remarks and assessments.

Efficient research is reliant first upon the accommodating mission statements and targets, and related research questions. These fundamental steps thus drive significant decisions in regards to concentrate on type, plans for evaluation and capable execution within limited time periods.

4.3 FEATURES OF A GOOD RESEARCH DESIGN

Though each plan has its own qualities and shortcomings and at the same time the perfect research design possibility is difficult. However a good research design is economical, flexible, efficient and appropriate etc. A design with minimum bias and maximum reliability is interpreted as a good design.

Likewise the design giving the minor error is considered to the best design and the design yielding maximal data covering different parts of an issue is understood as the most effective design since it is suitable to the research problem. Henceforth, thought of a good design relies on the target of the research problem and furthermore the idea of the issue being scrutinized. A decent research design ought to consistently satisfy the accompanying four conditions; objectivity, reliability, validity and generalizability of the outcomes.

(a) Objectivity

The findings are supposed to be target when they relate to the technique for collection of data and the scoring of the reactions. The objectivity of the system might be decided by the level of arrangement between the concluding scores appointed to different people by more than one independent spectator. The more the arrangement among the spectators the more goal are the perception, recording and assessment of the reactions. Thusly, a decent research design should allow target measuring instruments in which each spectator envisioning a presentation reaches a similar deduction.

(b) Reliability:

Reliability occurs when the presence of an issue stimulates in the knower an interest, for more than simple guess, however for something for which it will be helpful in a given circumstance

and maybe in other comparable circumstances. Dependable information implies any case that is validated as trustworthy for a given reason.

(c) Validity:

Validity suggests self-consistency or lack of self-contradiction. It is related to formal truth or self-consistency. A valid reasoning adjusts to the principles of right thinking. It is that kind of thinking where inferences naturally follow from the premises authentically.

(d) Generalisability:

Replicability and reproductibility of the outcome produces the degree of generalisability notwithstanding various measures and settings separately.

4.4 TYPES OF RESEARCH DESIGN

(i) Exploratory or Formulative Design:

The primary reason for exploratory design is to assemble data which will help in future for invention of an exact research problem. Based on the gathered facts the expert might have the option to plan sound hypothesis for additional research. It might likewise permit the specialist to get himself familiar with the wonders which he assumes to explore at a later stage. The point of an exploratory or formulative study might be explanation of ideas, building up needs for future consideration and assortment of information about the real conditions which influence a intended research.

Requirement of Exploratory Design:

The fundamentals for exploratory or formulative design are:

- (a) Review of appropriate texts
- (b) Experience Survey
- (c) Analysis of Insight Stimulating cases.

4.4.1 Review of pertinent literature:

While proceeding in the path of research the researcher has to take help from the work already done by his predecessors. By doing so, he will not only save himself from the problem of trial and error but also minimize the expenditure of his energy. Apart from reviewing available literature pertaining to the problem under investigation, the researcher may also take into account the literature pertinent to analogous problems.

4.4.2 Experience survey:

Because of the complicated nature of social problems, the researcher is not in a position to collect all the required materials about a particular problem from one place. At times the researcher has to contact the persons who have earned enough of experience to understand and analyze the social reactions. The researcher should take advantage of their experience in a very intelligent manner.

Taking good advantage of the experience of the persons involves the following steps:

4.4.2.1 Selection of respondents:

Formulation of a correct exploratory design requires that the investigator should make proper selection of the respondents. For this purpose he should select only those respondents who are dependable and who have actual knowledge regarding the problem under investigation.

The selection of the respondents may be made either directly or indirectly. In direct selection the investigator chooses those persons who are well known for their knowledge in the problem area. In case of indirect selection the investigator chooses those persons who are indirectly concerned with the problem. Hence, the selection of the respondents should not be confined to a particular group; rather it should be many sided.

4.4.2.2. Questioning of the respondents:

Proper questioning of the respondents ensures relevant information. Therefore while framing the questions, due attention should be given on clarity of concepts. For this purpose, the investigator should consult the books and the relevant portions of the bibliographical schemes adequately.

4.4.3 Analysis of insight stimulating cases:

Analysis of insight stimulating cases includes all those events, incidents and phenomena that stimulate the researcher. Such cases invoke in the investigator the thinking regarding the formulation of the hypotheses. In this regard, the attitude of the investigator, intensity of the case study and integrative power of investigators appear to be very important.

As regards the attitude of the investigator, receptivity and sensitivity are needed. These qualities enable the investigator to take stock of various developments occurring in his field of study and make steady progress.

Intensive case study involves studying the subject matter in all its dimensions and verifications, in the background of history.

In this regard, the groups, the community and groups of individuals may be treated as the units of study.

Integrative power of the investigator is considered important because on that basis he is able to collect even the minutest possible information regarding the subject matter. What appears significant, in this regard, is his attention on new observations rather than on experimentation.

(ii) Descriptive Research Design:

The purpose of descriptive type of design is to describe some event, situation, people, group or community or some phenomena. Fundamentally, it is a fact finding exercise which focuses on

relatively far dimensions of a well defined entity, aiming at precise and systematic measurement of some dimensions of a phenomenon.

Usually a descriptive design involves detailed numerical descriptions, such as distribution of the population of a community by age, sex, caste or education. The researcher may also take recourse to descriptive design for estimating the proportion of people in a particular geographical locality in respect of their specific views or attitudes.

However, the procedure followed in descriptive design is broadly analogous, notwithstanding the differences evinced in their field, formulation of hypotheses, objectives, for treatment of the problem and in matters of field expansion.

(iii) Diagnostic Research Design:

Being concerned with the express characteristics and existing social problems, the diagnostic research design endeavors to find out relationship between express causes and also suggests ways and means for the solution. Thus, the diagnostic studies are concerned with discovering and testing whether certain variables are associated. Such studies may also aim at determining the frequency with which something occurs or the ways in which a phenomenon is associated with some other factors.

Diagnostic studies are mostly motivated by hypotheses. A primary description of a problem serves the basis so as to relate the hypotheses with the source of the problem and only those data which form and corroborate the hypotheses are collected. As regards the objectives of diagnostic research design, it is based on such knowledge which can also be motivated or put into practice in the solution of the problem. Therefore, it is obvious that the diagnostic design is concerned with both the case as well as the treatment.

Diagnostic studies seek immediate to timely solution of the causal elements. The researcher, before going through other references, endeavors to remove and solve the factors and the causes responsible for giving rise to the problem.

The research design of diagnostic studies demands strict adherence to objectivity for elimination of any chances of personal bias or prejudice. Utmost care is taken while taking decisions regarding the variables, nature of observation to be made in the field, the type of evidence to be collected and tools of data collection. Simultaneously the research economy should not be lost sight of. Any faulty decision in these regard will result in wastage of time, energy and money.

Usually the first step in such designing is accurate formulation of research problem wherein research objectives are precisely stated and principal areas of investigation are properly linked. Otherwise the investigator will find it difficult to ensure the collection of required data in a

systematic manner. Simultaneously, the clarification of concepts and the operational definition of the terms should also be ensured so as to make them emendable to measurement.

At the next stage certain decisions regarding collection of data are taken. In this regard, the researcher should always bear in mind the advantages and disadvantages of the method to be employed and at the same time the nature of research problem, type of data needed, degree of desired accuracy etc. should be considered. That apart, while collecting data, effort must be made to maintain objectivity to the maximum possible extent.

In order to surmount the financial constraints, paucity of time, a representative sample of the research universe should be drawn so as to gather relevant information. A wide range of sampling techniques is prevalent which must be made use of, appropriately by the researchers.

At the stage of analysis of data, the researcher must take proper care in placing each item in the appropriate category, tabulating of data, applying statistical computations and so on.

Sufficient care must be taken to avoid potential errors due to faulty procedures of analysis of data. Advance decisions regarding the mode of tabulation, whether manual or by machine, accuracy of tabulating procedures, statistical application etc. will be of immense help in this regard.

(iv) Experimental Design:

The concept of experimental design in sociological research refers to systematic study of human relations by making the observations under conditions of control. In the words of Jahoda and Cook, ‘an experiment maybe considered as a way of organizing the collection of evidence so as to permit one to make inference about the tenability of a hypothesis. According to Chapin, “experiment is simply observation under controlled conditions. When observation alone fails to disclose the factors that operate in a given problem, it is necessary for the scientist to resort to experiment.”

In real terms, experimentation is resorted to when it is not possible to solve the problem through observation and general knowledge. The core of the experimental method lies in drawing inferences by observation of human relations under controlled conditions. Since a number of factors are in operation in every complex social situation, the social scientist, while seeking to describe the single causal relation of factor A to factor B, must attempt to create an artificial situation wherein all other factors, such as C, D, E etc., are controlled.

Such a state is achieved by selecting two groups which are equal in all significant receipts and choosing either of the groups as experimental group, and the other as the ‘control group’, and thereafter exposing the ‘experimental group’ to the assumed causal variable, while keeping the ‘control’ group under control. After a specific time period, the two groups are compared in terms

of the 'assumed effect'. The assumed causal variable and the assumed effect are otherwise called the independent variable and dependent variable respectively. Required evidence for testing causal relations among variables, already stated in the form of a hypothesis, is generated by the above method of experiment. Demonstration of causal relationship among variables in experimental design involves three clear-cut operations; such as demonstrating co-variation, eliminating spurious relationships and establishing the time order of occurrence. Here we will discuss the third operation which is concerned with establishing the time order of occurrence. This necessitates that the researcher should demonstrate that one phenomenon occurs first or gets transformed before the other phenomenon with the premise that the phenomenon which is yet to occur cannot be the determinant of the present or past phenomena.

Experimental design enables the researcher to draw causal inferences. It also smoothens, the observation of independent variable causing assumed effect. The three components of experimental design are: comparison, manipulation, and control. Through comparison, the correlation between variables is known. It also enables us to demonstrate the association between two or variables. Through manipulation the researcher establishes the time order of events. The major evidence which become essential to determine the sequence of events is that a change occurs only after the activation of the independent variable. In other words the independent variable precedes the dependent variable.

4.4.3 Types of Experimental Design

There are numerous ways in which experiments can be done in the field of social sciences. In their work "Experimental and Quasi-Experimental Designs of Research on Teaching", Donald T. Cambell and Julian C. Stanley have mentioned more than a hundred ways of conducting experiments which may be designated as experimental design.

But from the analytical point of view seven broad categories may be mentioned:

(i) After only Design:

Among all categories of experimental designs, after only design appears to be the simplest. This consists in measuring the dependent variable only after the experimental subjects have been exposed to the experimental variable. This design is considered more appropriate as an exploratory study than a real experiment.

(ii) Before-After Design:

As the name suggests, in this design measurement of the dependent variable is taken before as well as after exposure of the subject to the experimental variable, and the difference between the two measurements is taken to be the effect of the experimental variable. For example if the measured value of the dependent variable before exposure of the subject to the experimental variable is noted as 'A' and its measured value after exposure of the subject to experimental variable is noted as 'B' then the effect of the experimental variable is taken to be (B—A).

(iii) Before-After with Control Group Design:

In this design the research has a control group against which the results of the experimental groups are compared. The control group and experimental groups are selected in such a way that both the groups are similar and interchangeable. The control group is measured before as well as after without being exposed to the experimental variable.

Hence, there may hardly be any difference between before and after measurements. But if there is any difference between before and after measurement, it represents the result of uncontrolled variables.

On the other hand, the experimental variable is introduced in the experimental group. The difference between before and after measurements in respect of the experimental group is construed as the result of experimental variable as well as the uncontrolled variables. To know the exact effect of the experimental variable, the researcher deducts the difference between the two measurements of the controlled group from the difference of the two measurements of the experimental group.

The following notation explains this:

	Control Group	Experiment Group
Whether before-measurement is taken ?	Yes (A)	Yes (C)
Whether experimental variable is introduced ?	No	Yes
Whether after-measurement is taken ?	Yes (B)	Yes (D)
Effect of experimental variable	(D—C)—(B—A)	

(iv) Four Group-Six Study Design:

In this type of design two experimental groups and two control groups are taken. Measurements are made in six cases, i.e. before- measurement, and after-measurement in respect of experimental group-I, after-measurement in experimental group-II, before and after measurements in respect of control group-I; and only after measurement in control group-II.

Before measurements in all the four identical groups will be almost the same. If the before-measurements have no effect on the variable being studied, the two experimental groups should provide the same after-measurements and, similarly, the two control groups should also give the same after measurements. However, the results of in the two experimental groups are most likely to be different from the results of the two control groups, if the experimental variable exerts any influence.

(v) After Only with Control Group Design:

This is also known as two group-two study design, which is a modification of the four group-six study design. Here, the researcher does not study the experimental variable under different conditions. Hence, the effect of experimental variable is determined simply by finding out the differences between the after-measurements in respect of experimental and control groups. It so happens because if before-measurements of the experimental group-II and control group-II are taken, those are likely to be the same due to the identical characteristics of the groups. On this presumption, the researcher may very well ignore them.

We can express the design by the following notation :

	Experimental Group	Control Group
Whether before measurement is taken ?	No	No
Whether experimental variable is introduced ?	Yes	Yes
Whether after-measurement is taken ?	Yes (A)	Yes (B)
Effect of experimental variable	(A—B)	

(vi) Ex-Post Facto Design:

In Ex-post facto design the experimental and control groups are selected after the introduction of the experimental variable. Thus, it can be called as a variation of the after-only design. The main advantage of this design is that the test subjects are not influenced towards the subject by their knowledge of being tested. It also enables the researcher to introduce the experimental variable according to his own will and to control his observations.

(vii) Factorial Design:

All categories of experimental designs discussed above are designed to test experimental variable at one level only. But, on the other hand, the factorial designs enable the experimenter the testing of two or more variables simultaneously.

4.5 CONCEPTS RELATING TO RESEARCH DESIGN

Some of the important concepts relating to Research Design are discussed below:

4.5.1. Dependent And Independent Variables

A magnitude that varies is known as a variable. The concept may assume different quantitative values like height, weight, income etc. Qualitative variables are not quantifiable in the strictest sense of the term. However, the qualitative phenomena may also be quantified in terms of the presence or absence of the attribute(s) considered. The phenomena that assume different values quantitatively even in decimal points are known as ‘continuous variables’. But all variables need not be continuous. Values that can be expressed only in integer values are called ‘non-continuous variables’. In statistical terms, they are also known as ‘discrete variables’. For example, age is a

continuous variable, whereas the number of children is a non-continuous variable. When changes in one variable depend upon the changes in other variable or variables, it is known as a dependent or endogenous variable, and the variables that cause the changes in the dependent variable are known as the independent or explanatory or exogenous variables. For example, if demand depends upon price, then demand is a dependent variable, while price is the independent variable. And, if more variables determine demand, like income and price of the substitute commodity, then demand also depends upon them in addition to the price of original commodity. In other words, demand is a dependent variable which is determined by the independent variables like price of the original commodity, income and price of substitutes.

4.5.2 Extraneous Variables

The independent variables which are not directly related to the purpose of the study but affect the dependent variables, are known as extraneous variables. For instance, assume that a researcher wants to test the hypothesis that there is a relationship between children's school performance and their self-confidence, in which case the latter is an independent variable and the former, a dependent variable. In this context, intelligence may also influence the school performance. However, since it is not directly related to the purpose of the study undertaken by the researcher, it would be known as an extraneous variable. The influence caused by the extraneous variable(s) on the dependent variable is technically called the 'experimental error'. Therefore, a research study should always be framed in such a manner that the influence of extraneous variables on the dependent variable/s is completely controlled, and the influence of independent variable/s is clearly evident.

4.5.3 Control

One of the most important features of a good research design is to minimize the effect of extraneous variable(s). Technically, the term 'control' is used when a researcher designs the study in such a manner that it minimizes the effects of extraneous variables. The term 'control' is used in experimental research to reflect the restraint in experimental conditions.

4.5.4 Confounded Relationship

The relationship between the dependent and independent variables is said to be confounded by an extraneous variable, when the dependent variable is not free from its effects.

4.5.5. Research Hypothesis

When a prediction or a hypothesized relationship is tested by adopting scientific methods, it is known as research hypothesis. The research hypothesis is a predictive statement which relates to a dependent variable and an independent variable. Generally, a research hypothesis must consist of at least one dependent variable and one independent variable. Whereas, the relationships that are assumed but not to be tested are predictive statements that are not to be objectively verified, thus are not classified as research hypotheses.

4.5.6. Experimental and Non-experimental Hypothesis Testing Research

When the objective of a research is to test a research hypothesis, it is known as hypothesis-testing research. Such research may be in the nature of experimental design or non-experimental design. The research in which the independent variable is manipulated is known as ‘experimental hypothesis-testing research’, whereas the research in which the independent variable is not manipulated is termed as ‘non-experimental hypothesis-testing research’. For example, assume that a researcher wants to examine whether family income influences the school attendance of a group of students, by calculating the coefficient of correlation between the two variables. Such an example is known as a non-experimental hypothesis-testing research, because the independent variable - family income is not manipulated here. Again assume that the researcher randomly selects 150 students from a group of students who pay their school fees regularly and then classifies them into two sub-groups by randomly including 75 in Group A, whose parents have regular earning, and 75 in Group B, whose parents do not have regular earning. Assume that at the end of the study, the researcher conducts a test on each group in order to examine the effects of regular earnings of the parents on the school attendance of the student. Such a study is an example of experimental hypothesis-testing research, because in this particular study the independent variable regular earnings of the parents have been manipulated.

4.5.7. Experimental And Control Groups

When a group is exposed to usual conditions in an experimental hypothesis-testing research, it is known as ‘control group’. On the other hand, when the group is exposed to certain new or special condition, it is known as an ‘experimental group’. In the afore-mentioned example, Group A can be called as control group and Group B as experimental group. If both the groups, A and B are exposed to some special feature, then both the groups may be called as ‘experimental groups’. A research design may include only the experimental group or both the experimental and control groups together.

4.5.8 Treatments

Treatments refer to the different conditions to which the experimental and control groups are subject to. In the example considered, the two treatments are the parents with regular earnings and those with no regular earnings. Likewise, if a research study attempts to examine through an experiment the comparative effect of three different types of fertilizers on the yield of rice crop, then the three types of fertilizers would be treated as the three treatments.

4.5.9 Experiment

Experiment refers to the process of verifying the truth of a statistical hypothesis relating to a given research problem. For instance, an experiment may be conducted to examine the yield of a certain new variety of rice crop developed. Further, Experiments may be categorized into two types, namely, ‘absolute experiment’ and ‘comparative experiment’. If a researcher wishes to

determine the impact of a chemical fertilizer on the yield of a particular variety of rice crop, then it is known as absolute experiment. Meanwhile, if the researcher wishes to determine the impact of chemical fertilizer as compared to the impact of bio-fertilizer, then the experiment is known as a comparative experiment.

4.5.10 Experimental Unit(s):

Experimental units refer to the pre-determined plots, characteristics or the blocks, to which different treatments are applied. It is worth mentioning here that such experimental units must be selected with great caution.

4.6 Cluster analysis

Cluster analysis is a statistical method used to group similar objects into respective categories. It can also be referred to as segmentation analysis, taxonomy analysis, or clustering.

The goal of performing a cluster analysis is to sort different objects or data points into groups in a manner that the degree of association between two objects is high if they belong to the same group, and low if they belong to different groups.

Cluster analysis differs from many other statistical methods due to the fact that it's mostly used when researchers do not have an assumed principle or fact that they are using as the foundation of their research. This analysis technique is typically performed during the exploratory phase of research, since unlike techniques such as factor analysis, it doesn't make any distinction between dependent and independent variables. Instead, cluster analysis is leveraged mostly to discover structures in data without providing an explanation or interpretation.

Put simply, cluster analysis discovers structures in data without explaining why those structures exist. For example, when cluster analysis is performed as part of market research, specific groups can be identified within a population. The analysis of these groups can then determine how likely a population cluster is to purchase products or services. If these groups are defined clearly, a marketing team can then target varying cluster with tailored, targeted communication.

4.6 COMMON APPLICATIONS OF CLUSTER ANALYSIS

4.6.1 Marketing

Marketers commonly use cluster analysis to develop market segments, which allow for better positioning of products and messaging. company to better position itself, explore new markets, and development products that specific clusters find relevant and valuable.

4.6.2 Insurance

Insurance companies often leverage cluster analysis if there are a high number of claims in a given region. This enables them to learn exactly what is driving this increase in claims.

4.6.3 Geology

For cities on fault lines, geologists use cluster analysis to evaluate seismic risk and the potential weaknesses of earthquake-prone regions. By considering the results of this research, residents can do their best to prepare mitigate potential damage.

4.6.4 Putting Clustering into Context

It's easy to overthink cluster analysis, but our brains naturally cluster data on a regular basis in order to simplify the world around us. Whether we realize it or not, we deal with clustering in practically every aspect of our day-to-day lives.

For example, a group of friends sitting at the same table in a restaurant can be considered a cluster. In grocery stores, goods of a similar nature are grouped together in order to make shopping more convenient and efficient. This list of events during which we use clustering in our everyday lives could go on forever, but perhaps it makes more sense to consider a more classic, archetypal example. In biology, humans belong to the following clusters: primates, mammals, amniotes, vertebrates, and animals. In this example, note that as we move down the chain of clusters, humans show less and less similarities to the other members of the group. Humans have more in common with primates than they do with other mammals, and more in common with mammals than they do with all animals in general.

The Benefits of Cluster Analysis

Clustering allows researchers to identify and define patterns between data elements. Revealing these patterns between data points helps to distinguish and outline structures which might not have been apparent before, but which give significant meaning to the data once they are discovered. Once a clearly defined structure emerges from the dataset at hand, informed decision-making becomes much easier.

4.7 TYPES OF CLUSTER ANALYSIS

There are three primary methods used to perform cluster analysis:

4.7.1 Hierarchical Cluster

This is the most common method of clustering. It creates a series of models with cluster solutions from 1 (all cases in one cluster) to n (each case is an individual cluster). This approach also works with variables instead of cases. Hierarchical clustering can group variables together in a manner similar to factor analysis.

Finally, hierarchical cluster analysis can handle nominal, ordinal, and scale data. But, remember not to mix different levels of measurement into your study.

4.7.2 K-Means Cluster

This method is used to quickly cluster large datasets. Here, researchers define the number of clusters prior to performing the actual study. This approach is useful when testing different models with a different assumed number of clusters.

4.7.3 Two-Step Cluster

This method uses a cluster algorithm to identify groupings by performing pre-clustering first, and then performing hierarchical methods. Two-step clustering is best for handling larger datasets that would otherwise take too long a time to calculate with strictly hierarchical methods.

Essentially, two-step cluster analysis is a combination of hierarchical and k-means cluster analysis. It can handle both scale and ordinal data, and it automatically selects the number of clusters.

What Does The Clustering Process Look Like?

Step #1: Build and Distribute a Survey

Your survey should be designed to include multiple measures of propensity to purchase and the preferences for the product at hand. It should be distributed to your population of interest, and your sample size should be large enough to inform statistically-based decisions.

Step #2: Analyze Response Data

It's considered best practice to perform a factor analysis on your survey to minimize the factors being clustered. If after your factor analysis it's concluded that a handful of questions are measuring the same thing, you should combine these questions prior to performing your cluster analysis.

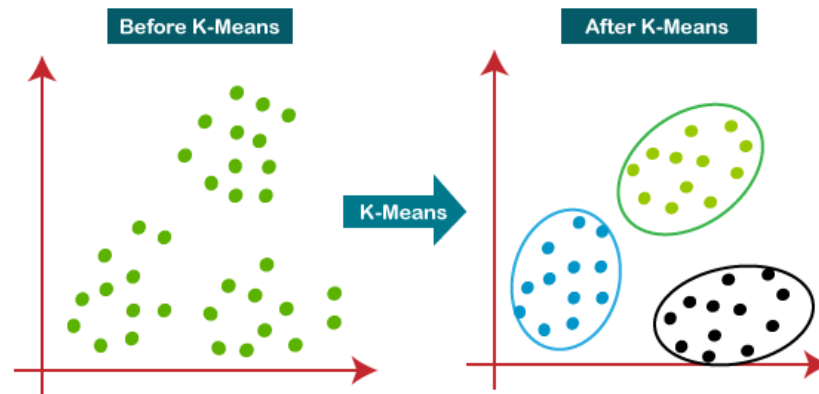
After reducing your data by factoring, perform the cluster analysis and decide how many clusters seem appropriate, and record those cluster assignments. You'll now be able to view the means of all of your factors across clusters.

Step #3: Take Informed Action!

Comb through your data to identify differences in the means of factors, and name your clusters based on these differences. These differences between clusters are then able to inform your marketing, allowing you to target precise groups of customers with the right message, at the right time, in the right manner.

4.8 K-MEANS CLUSTERING

K means is one of the most popular Unsupervised Machine Learning Algorithms Used for Solving Classification Problems. *K Means segregates the unlabeled data into various groups, called clusters, based on having similar features, common patterns.*



Kmeans Algorithm is an Iterative algorithm that divides a group of n datasets into k subgroups /clusters based on the similarity and their mean distance from the centroid of that particular subgroup/ formed.

K, here is the pre-defined number of clusters to be formed by the Algorithm. If $K=3$, It means the number of clusters to be formed from the dataset is 3

Algorithm steps Of K Means

The working of the K-Means algorithm is explained in the below steps:

Step-1: Select the value of K, to decide the number of clusters to be formed.

Step-2: Select random K points which will act as centroids.

Step-3: Assign each data point, based on their distance from the randomly selected points (Centroid), to the nearest/closest centroid which will form the predefined clusters.

Step-4: place a new centroid of each cluster.

Step-5: Repeat step no.3, which reassign each datapoint to the new closest centroid of each cluster.

Step-6: If any reassignment occurs, then go to step-4 else go to Step 7.

Step-7: FINISH

Diagrammatic Implementation of K Means Clustering

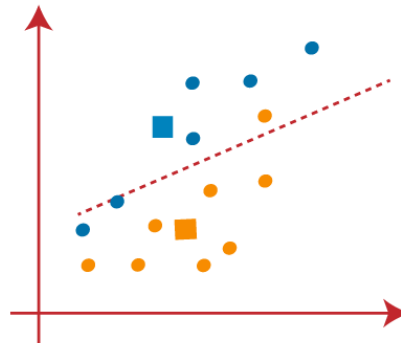
STEP 1: Let's choose number k of clusters, i.e., $K=2$, to segregate the dataset and to put them into different respective clusters. We will choose some random 2 points which will act as centroid to form the cluster.

STEP 2: Now we will assign each data point to a scatter plot based on its distance from the closest K -point or centroid. It will be done by drawing a median between both the centroids. Consider the below image:

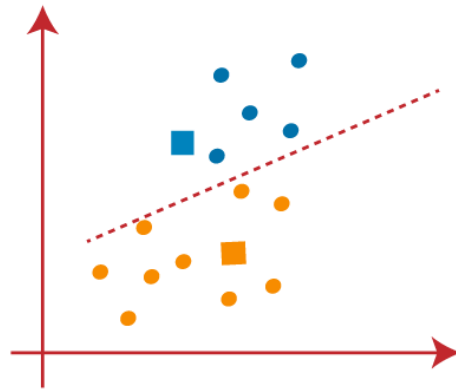
STEP 3: points left side of the line is near to blue centroid, and points to the right of the line are close to the yellow centroid. The left one Form cluster with blue centroid and the right one with the yellow centroid.

STEP 4: repeat the process by choosing a **new centroid**. To choose the new centroids, we will find the new center of gravity of these centroids, which is depicted below :

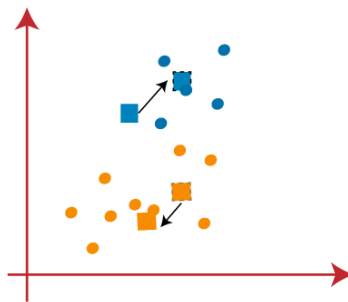
STEP 5: Next, we will reassign each datapoint to the new centroid. We will repeat the same process as above (using a median line). The yellow data point on the blue side of the median line will be included in the blue cluster



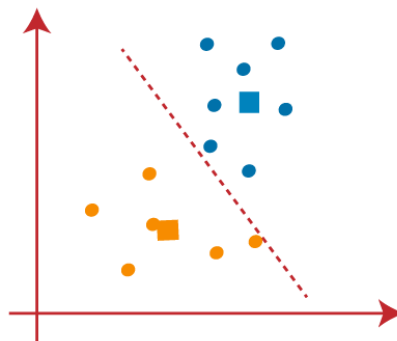
STEP 6: As reassignment has taken place, so we will repeat the above step of finding new centroids.



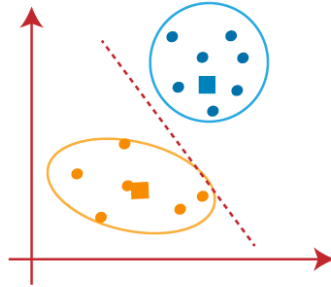
STEP 7: We will repeat the above process of finding the center of gravity of centroids, as being depicted below



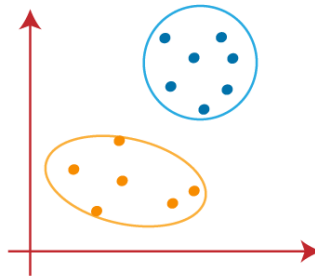
STEP 8: After Finding the new centroids we will again draw the median line and reassign the data points, like the above steps.



STEP 9: We will finally segregate points based on the median line, such that two groups are being formed and no dissimilar point to be included in a single group



The final Cluster being formed are as Follows



4. Choosing The Right Number Of Clusters

The number of clusters that we choose for the algorithm shouldn't be random. Each and Every cluster is formed by calculating and comparing the mean distances of each data points within a cluster from its centroid.

We Can Choose the right number of clusters with the help of the Within-Cluster-Sum-of-Squares (WCSS) method.

WCSS Stands for the sum of the squares of distances of the data points in each and every cluster from its centroid.

The main idea is to minimize the distance between the data points and the centroid of the clusters. The process is iterated until we reach a minimum value for the sum of distances.

To find the optimal value of clusters, the elbow method follows the below steps:

- 1 Execute the K-means clustering on a given dataset for different K values (ranging from 1-10).
- 2 For each value of K, calculates the WCSS value.
- 3 Plots a graph/curve between WCSS values and the respective number of clusters K.
- 4 The sharp point of bend or a point(looking like an elbow joint) of the plot like an arm, will be considered as the best/optimal value of K

5. Python Implementation

Importing relevant libraries

```
import numpy as np

import pandas as pd

import statsmodels.api as sm

import matplotlib.pyplot as plt

import seaborn as sns

sns.set()

from sklearn.cluster import KMeans
```

Loading the Data

```
data = pd.read_csv('Countryclusters.csv')

data
```

	Country	Latitude	Longitude	Language
0	USA	44.97	-103.77	English
1	Canada	62.40	-96.80	English
2	France	46.75	2.40	French
3	UK	54.01	-2.53	English
4	Germany	51.15	10.40	German
5	Australia	-25.45	133.11	English

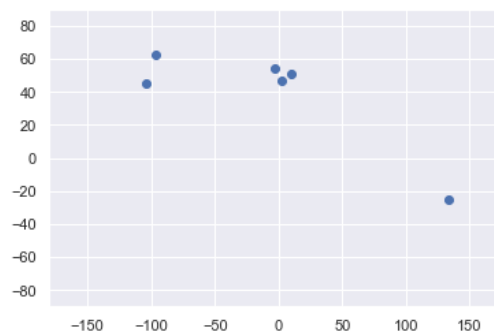
Plotting the data

```
plt.scatter(data['Longitude'],data['Latitude'])

plt.xlim(-180,180)

plt.ylim(-90,90)

plt.show()
```



Selecting the feature

```
x = data.iloc[:,1:3] # 1t for rows and second for columns
```

x

	Latitude	Longitude
0	44.97	-103.77
1	62.40	-96.80
2	46.75	2.40
3	54.01	-2.53
4	51.15	10.40
5	-25.45	133.11

Clustering

```
kmeans = KMeans(3)
```

```
means.fit(x)
```

Clustering Results

```
identified_clusters = kmeans.fit_predict(x)
```

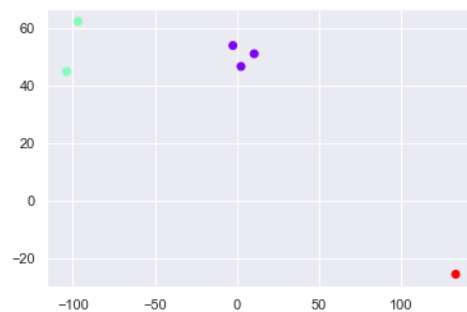
```
identified_clusters
```

```
array([1, 1, 0, 0, 0, 2])
```

```
data_with_clusters = data.copy()
```

```
data_with_clusters['Clusters'] = identified_clusters
```

```
plt.scatter(data_with_clusters['Longitude'],data_with_clusters['Latitude'],c=data_with_clusters['Clusters'],cmap='rainbow')
```



Trying different method (to find no .of clusters to be selected)

WCSS and Elbow Method

```
wcss=[]
```

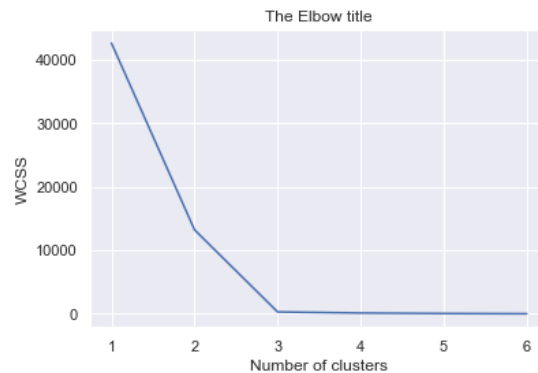
```
for i in range(1,7):
```

```

kmeans = KMeans(i)
kmeans.fit(x)
wcss_iter = kmeans.inertia_
wcss.append(wcss_iter)

number_clusters = range(1,7)
plt.plot(number_clusters,wcss)
plt.title('The Elbow title')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')

```



we can choose 3 as no. of clusters, this method shows what is the good number of clusters.

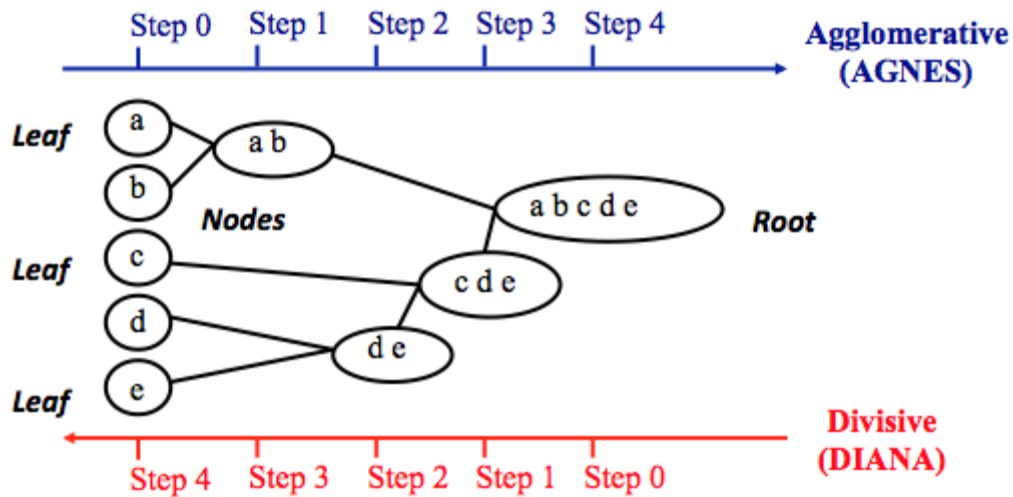
4.9 AGGLOMERATIVE CLUSTERING

The **agglomerative clustering** is the most common type of hierarchical clustering used to group objects in clusters based on their similarity. It's also known as *AGNES (Agglomerative Nesting)*. The algorithm starts by treating each object as a singleton cluster. Next, pairs of clusters are successively merged until all clusters have been merged into one big cluster containing all objects. The result is a tree-based representation of the objects, named *dendrogram*.

Agglomerative clustering works in a “bottom-up” manner. That is, each object is initially considered as a single-element cluster (leaf). At each step of the algorithm, the two clusters that are the most similar are combined into a new bigger cluster (nodes). This procedure is iterated until all points are member of just one single big cluster (root) (see figure below).

The inverse of agglomerative clustering is *divisive clustering*, which is also known as DIANA (*Divise Analysis*) and it works in a “top-down” manner. It begins with the root, in which all objects are included in a single cluster. At each step of iteration, the most heterogeneous cluster

is divided into two. The process is iterated until all objects are in their own cluster (see figure below).



Steps to agglomerative hierarchical clustering

We'll follow the steps below to perform agglomerative hierarchical clustering using R software:

1. Preparing the data
2. Computing (dis)similarity information between every pair of objects in the data set.
3. Using linkage function to group objects into hierarchical cluster tree, based on the distance information generated at step 1. Objects/clusters that are in close proximity are linked together using the linkage function.
4. Determining where to cut the hierarchical tree into clusters. This creates a partition of the data.

B.Sc.(DATA SCIENCE)

SEMESTER-III

DATA PREPARATION

UNIT V: SAMPLING DISTRIBUTIONS

STRUCTURE

5.0 Objectives

5.1 Introduction

5.2 Non-Central Chi – Square

5.3 T And F Distributions and Their Properties

5.4 Distributions of Quadratic Forms Under Normality

5.5 Independence of Quadratic Form And A Linear Form

5.6 Cochran’s Theorem

5.7 Summary

5.8 References

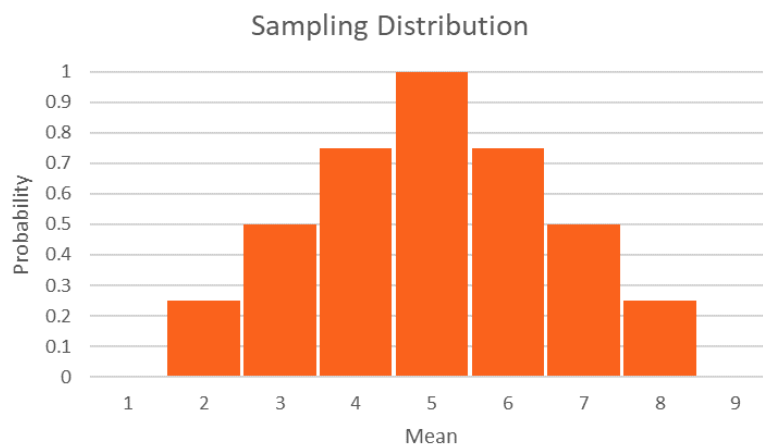
5.9 Practice Exercise

5.0 OBJECTIVES

This course will make student familiar with the methods of data gathering and preparing the data for processing. Student will explore the data sampling techniques.

5.1 INTRODUCTION

The population sampling distribution refers to the probability distribution of a statistics that is calculated by sampling the various possible outcomes and then counting the number of times each occurred. A random sample selection produces a probability distribution for a specific statistic drawn from a given population. Finite-sample distribution (also known as distribution of frequencies on how far the possible outcomes are spread apart) represents the frequencies of all possible outcomes that could appear for a given population. A population is the set of all individuals in a given group from which a random sample is drawn. It is thus possible to speak of populations as an aggregate analysis of groups of individuals with a common characteristic or parameter. A random sampling distribution is a common statistical tool, frequently used in hypothesis testing and making decisions when we do not know the full outcome. The probability of an outcome, such as the mean or mode of a variable, occurring for a statistic such as the mean or mode of a variable can be predicted by taking into consideration all possible outcomes. This data is gathered from samples rather than the entirety of the population.



The sampling distribution depends on various factors –sampling process, sample size, the statistic and the overall population. It helps in calculating various statistics features such as means, variances, ranges and standard deviations for a given sample from a population. It works as follow:

1. Calculate a statistical feature for a given population sample, such as the mean, median, or standard deviation.
2. For each sample statistic develop a frequency distribution that you calculated from the step 1 above.
3. For each sample statistic that you developed from the step 2 above, plot the frequency distribution. The resulting graph will be the sampling distribution.

Consider this example. A large tank of fish from a hatchery is being delivered to the lake. We want to know the average length of the fish in the tank. Instead of measuring all of

the fish, we randomly sample twenty fish and use the sample mean to estimate the population mean.

Denote the sample mean of the twenty fish as \bar{l}_1 . Suppose we take a separate sample of size twenty from the same hatchery. Denote that sample mean as \bar{l}_2 . Would \bar{l}_1 equal as \bar{l}_2 ? Not necessarily. What if we took another sample and found the mean? Consider now taking 2000 random samples of size twenty and recording all of the sample means. We could take the 2000 sample means and create a histogram. This would give us a picture of what the distribution of the sample means looks like. The distribution of all of these sample means is the sampling distribution of the sample mean.

We can find the sampling distribution of any sample statistic that would estimate a certain population parameter of interest. In this Lesson, we will focus on the sampling distributions for the sample mean, \bar{x} , and the sample proportion, p^\wedge .

5.2 NON-CENTRAL CHI – SQUARE

If squares of k independent standard normal random variables are added, it gives rise to central Chi-squared distribution with ‘ k ’ degrees of freedom. Instead, if squares of k independent normal random variables with non-zero means are added, it gives rise to non-central Chi-squared distribution. Non-central Chi-square distribution is related to Ricean distribution, whereas the central Chi-squared distribution is related to Rayleigh distribution. The non-central Chi-squared distribution is a generalization of Chi-square distribution. A non-central Chi-squared distribution is defined by two parameters: 1) degrees of freedom (k) and 2) non-centrality parameter λ .

As we know from previous article, the degrees of freedom specify the number of independent random variables we want to square and sum-up to make the Chi-squared distribution. Non-centrality parameter is the sum of squares of means of each independent underlying normal random variable.

The non-centrality parameter is given by

$$\lambda = \sum_{i=1}^k \mu_i^2 \quad (1)$$

The probability density function $f_{x_k^2}(x, \lambda)$ for the non-central Chi-squared distribution having non-centrality parameter λ and k degrees of freedom is represented as:

$$f_{x_k^2}(x, \lambda) = \sum_{n=0}^{\infty} \frac{e^{-\lambda/2} (\frac{\lambda}{2})^n}{n!} f_{Y_{k+2n}}(x) \quad (2)$$

Where, the random variable $f_{Y_{k+2n}}$ is central Chi-squared distributed with $k + 2n$ degrees of freedom. The factor $\frac{e^{-\lambda/2} (\frac{\lambda}{2})^n}{n!}$ gives the probabilities of Poisson distribution. A weighted sum of Chi-squared probabilities (probabilities of the Poisson distribution) represents the probability density function of the non-central Chi-squared distribution.

Code using Python:-----

Non-central Chi-square distributed sequences can be generated using the numpy package's `noncentral_chisquare()` generator.

Non-central Chi-square distribution 3

```
import numpy as npy
import matplotlib.pyplot as plt
#%%matplotlib inline
plt.style.use('ggplot')

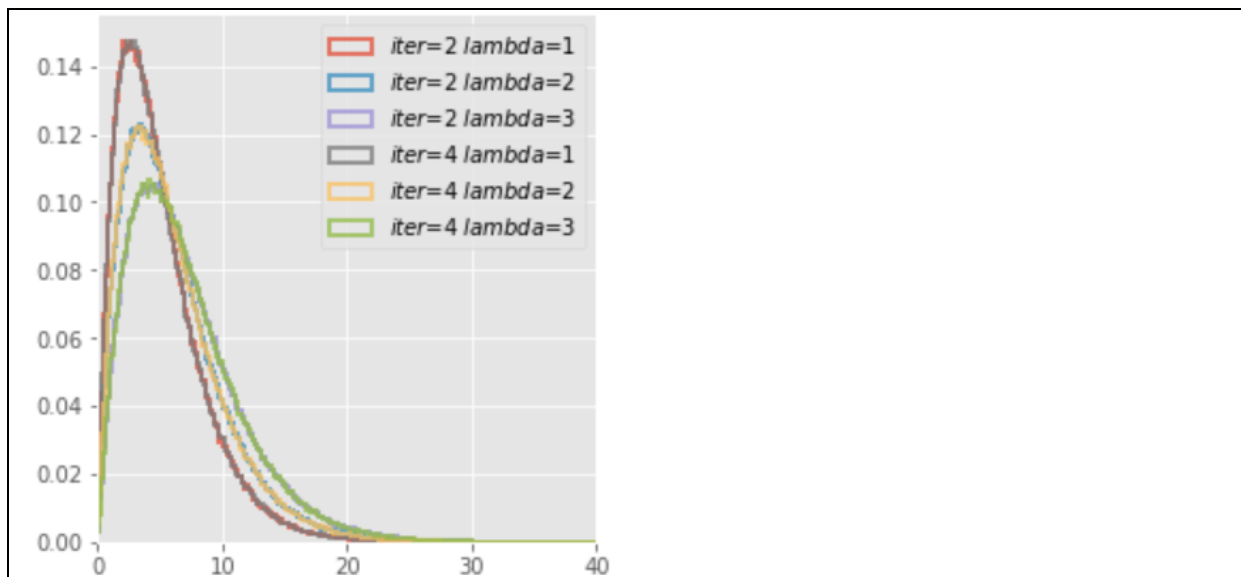
ks = npy.asarray([2,4]) #degrees of freedom to simulate
ldas = npy.asarray([1,2,3]) #non-centrality parameters to simulate
nSamp=1000000 #number of samples to generate

fig, ax = plt.subplots(ncols=1, nrows=1, constrained_layout=True)

for i,iter in enumerate(ks):
    for j,lda in enumerate(ldas):
        #Generate non-central Chi-squared distributed random numbers
        X = npy.random.noncentral_chisquare(df=k, nonc = lda, size = nSamp)
        ax.hist(X,bins=500,density=True,label=r'$iter$={ } $lambda$={ }'.format(iter,lda),\
            histtype='step',alpha=0.75, linewidth=2)

ax.set_xlim(left=0,right=40);ax.legend()
ax.set_title('Probabilty Density functions of non-central Chi square distribution');
plt.show()
```

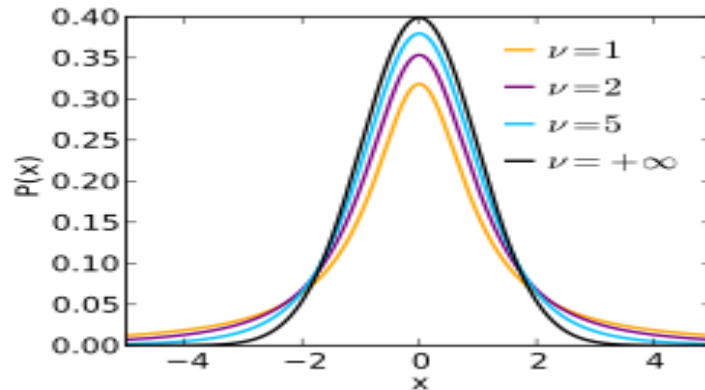
OUTPUT:



5.3 T AND F DISTRIBUTIONS AND THEIR PROPERTIES

T-distribution also known as **Student's T-Distribution** is a group of distributions heavily used in statistics which look alike to the normal distribution curve, but a bit fatter and shorter. The wide range of probable outcomes (more heavily weighted with many zigzag tails and bell-shaped curves) makes this distribution ideal for situations where

either the standard deviation of the population is unknown or the sample size is small. T-distributions have more chances for extreme values in comparison to normal distributions, hence they have fatter tails. It is used in place of the normal distribution when the size of the sample is small (for more on this, see: t-score vs. z-score). The more the size of the sample, the more the T-distribution curves looks similar to the normal distribution curves. In fact, for a sample sizes of greater than 20 (e.g., more degrees of freedom), the T-distribution curves are almost exactly in shape to that of normal distribution curves.



Graph of T-distribution

T-distribution is considered to be an approximation to the normal distribution. It is used to derive meaningful statistical values out from the population sample and can be used in calculating statistical inference. The degree of freedom of the T-distribution is one of many features that can be used to tell if the distribution is long and/or fat. A larger degree of freedom (DDF) indicates that the T-distribution looks like a standard normal distribution, with a mean of 0 and a standard deviation of 1. Meanwhile, a smaller degree of freedom (DF) denotes a T-distribution that is fat, with a longer mean and wider standard deviation. Because the population's mean, M , and standard deviation, D , are both known, there will be a variation in the sample mean, m , and standard deviation, d , that originates from the randomness of the n sample. The z-score, Z , can be found by taking the standard deviation of the population and subtracting it from the mean, with the result being $Z = (x - M) / D$, which has a normal distribution with mean of zero and standard deviation of one. But if you use the computed population standard deviation a t-score is derived by using $T = (m - M) / \{d / \sqrt{n}\}$, which results in that the distribution is T-distributed with $(n - 1)$ degrees of liberty rather than a normal distribution with mean = 0 and standard deviation = 1.

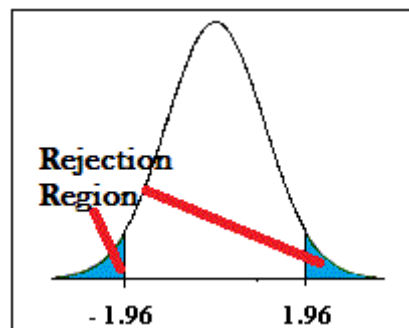
Example:

To illustrate how T-distributions are applied in statistical analysis, consider the following example. The first thing to know is that when you are drawing a confidence interval, you are simply selecting a range of values from the data with the intention of obtaining a “population” mean. When 't' is a crucial number from the T distribution, the interval of $m \pm t \cdot d / \sqrt{n}$ exists.

The preceding example shows how to calculate a 95% confidence interval for the mean return of the Dow Jones Industrial Average for a 27-day trading period prior to 9/11/2001. Using the formula $(-0.33\% \pm 2.055 * 1.07 / \sqrt{27})$, we arrive at a figure between -0.75% and +0.09%. The standard error is 2.055, and this is the T-distribution adjusted number.

Financial returns that show excess kurtosis can be modelled by the T-distribution, which has larger curve tails in comparison to a normal distribution. The T-distribution can be compared to the normal distribution, which is why it distorts precision. One of the disadvantages of normalcy is that it becomes difficult to achieve.

In case of hypothesis testing, the T-distribution and the related t-scores are used when we want to figure out whether a null hypothesis should be accepted or rejected depending upon the region in which the absolute value lies.



This plot has an acceptance area (in the middle) and a rejection area (on the left). In the example above, the blue region is the region in which a null hypothesis is rejected. The area under the tail can be described in terms of t-scores or z-scores. For example, the graph above shows an area under the tail on both sides of 5% (each side of 2.5%). The z-score for the above example would be 1.96 (obtained from z-table), representing a standard deviation of 1.96 from the mean. All the null hypothesis with value of z in the range $(-1.96 > z > 1.96)$ will be rejected.

When the sample size is small (usually under 30), or when the standard deviation of the population is unknown, the T-distribution is used. For practical problems (i.e., in the realistic situations), T-distribution is nearly always the option. So, in contrast to your elementary class on statistics, you will apparently use T-distributions in realistic problems more than that of normal distribution. If the sample size under consideration is large enough, then the two distributions appear to be same practically.

Properties of the Student t distribution

1. It is directly proportional to sample sizes (i.e., different for different sizes), or different degrees of freedom.
2. It has the same general symmetric bell-shaped curves as is the case with standard normal distribution, but it shows greater variability (with wider distributions) that is expected with samples of small size.
3. The mean 'm' of the distribution = 0.

4. The standard deviation of the distribution changes with respect to sample size, but it is > 1 .
5. As the size of the sample 'n' gets larger, the distribution gets more closer to the normal distribution.

Python code:

```
# t-test for dependent samples
from math import sqrt
from numpy.random import seed
from numpy.random import randn
from numpy import mean
from scipy.stats import t

# function for calculating the t-test for two dependent samples
def dependent_ttest(dat1, dat2, alphaa):
    # calculate means
    mean1, mean2 = mean(dat1), mean(dat2)
    # number of paired samples
    n = len(dat1)
    # sum squared difference between observations
    d1 = sum([(data1[i]-data2[i])**2 for i in range(n)])
    # sum difference between observations
    d2 = sum([data1[i]-data2[i] for i in range(n)])
    # standard deviation of the difference between means
    sd = sqrt((d1 - (d2**2 / n)) / (n - 1))
    # standard error of the difference between the means
    sed = sd / sqrt(n)
    # calculate the t statistic
    t_stat = (mean1 - mean2) / sed
    # degrees of freedom
    df = n - 1
    # calculate the critical value
    cv = t.ppf(1.0 - alpha, df)
    # calculate the p-value
    p = (1.0 - t.cdf(abs(t_stat), df)) * 2.0
    # return everything
    return t_stat, df, cv, p

# seed the random number generator
seed(1)
# generate two independent samples (pretend they are dependent)
dat1 = 5 * randn(100) + 50
dat2 = 5 * randn(100) + 51
# calculate the t test
alphaa = 0.05
t_stat, df, cv, p = dependent_ttest(dat1, dat2, alphaa)
print('t=%.3f, df=%d, cv=%.3f, p=%.3f' % (t_stat, df, cv, p))
# interpret via critical value
```



```

if abs(t_stat) <= cv:
    print('Accept null hypothesis that the means are equal.')
else:
    print('Reject the null hypothesis that the means are equal.')
# interpret via p-value
if p > alphaa:
    print('Accept null hypothesis when means are equal.')
else:
    print('Reject the null hypothesis when means are not equal.')

```

Output:

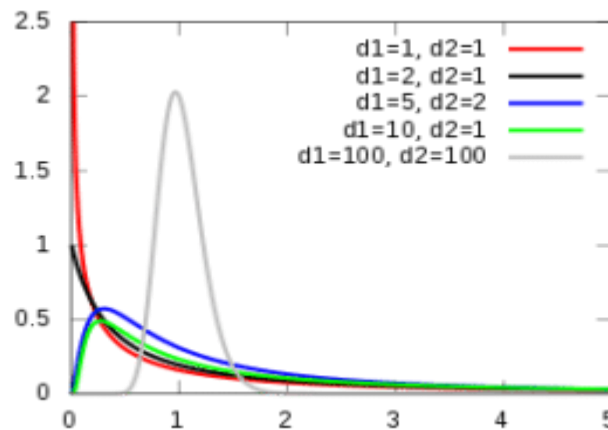
```

t=-2.372, df=99, cv=1.660, p=0.020
Reject the null hypothesis that the means are equal.
Reject the null hypothesis that the means are equal.

```

F-distribution

Probability distributions can be used to predict the likelihood of a specific event. Also, the F-distribution follows the same pattern. A skewed and continuous distribution of probabilities with similarities to a chi-squared distribution is known as the F-distribution, the Snedecor's F-distribution, or the F-ratio. F-distribution is related to the chi-squared distribution in that it is based on the sum of two chi-squared distributions. However, whereas the chi-squared distribution is concerned with the degrees of freedom with a single set of variables, the F-distribution is concerned with multiple levels of events with varying degrees of freedom. Since, therefore, we can say that there are several versions of the F-distribution with different degrees of freedom, it can be said that there are several F-distributions altogether.



Graph of F-distribution

Each curve in a F-distribution represents a different degree of freedom which implies that the area required for the hypothesis test to be significant is different. Mathematically the equation to calculate probability density function $f(x)$ of the F-distribution curve is:

$$f(x) = \frac{r\left(\frac{df_1+df_2}{2}\right)\left(\frac{df_1}{df_2}\right)^{\frac{df_1}{2} \times \frac{df_1}{2} - 1}}{r\left(\frac{df_1}{2}\right)r\left(\frac{df_2}{2}\right)\left(1+\frac{df_1}{df_2}x\right)^{\left(\frac{df_1+df_2}{2}\right)}} \quad (3)$$

Where f_1, f_2 are the shape parameters (degrees of freedom) and 'r' is the gamma function.

From the above equation it is clear that when the value for the degrees of freedom change (f_1 and f_2), the curve changes as well.

The F-test which uses the F-distribution, compare more than one level of independent variables with multiple groups. This type is commonly found in case of ANOVA and factorial ANOVA.

Let us consider that you want to test a new medicine for heart disease called M. In this case, what you want to know is the significant effects of various amounts of dosages. So, being a brilliant statistician, you set up three trials of 100 mg, 50 mg, and 0 mg of M in three randomly chosen groups of 30 people. Such a strategy is for ANOVA, which uses the F-distribution.

Whenever you are handed over a problem comparing two or more groups, you will require the F-distribution to carry out F-test.

The F-distribution is used for the F-test. The F-test is based on calculating an F-score, which is used to determine whether the sample size is appropriate for multiple levels. The formula to calculate F-score is as:

$$F = \frac{\frac{df_1 \cdot s_1^2}{\sigma_1^2} / df_1}{\frac{df_2 \cdot s_2^2}{\sigma_2^2} / df_2} \quad (4)$$

This test compares the variance obtained *within* a group (say e.g., all the 100 mg participants) to the variance obtained *between* the groups (three trials groups in our case). When you compute this formula, you obtain an F-score.

You will basically find out the value at which your degrees of freedom intersect with each other. If the estimated value exceeds the value indicated in the table, then samples under consideration are significantly different from each other. If the obtained value is lower than in the table, then the groups are significant with respect to each other.

Properties of F-distribution

Given P and Q, independent probabilistically from each other, each follows $\chi^2(r)$ and $\chi^2(s)$ respectively, the F-distribution of $\frac{P/r}{Q/s}$ denoted by $F(r, s)$ with degrees of freedom (r, s) . The density function is expressed

$$f(x) = \frac{\frac{r}{r-2} \frac{s}{s-2} \frac{x^{r-1}}{x^2-1}}{B\left(\frac{r}{2}, \frac{s}{2}\right) (rx+s)^{\frac{r+s}{2}}} \quad (5)$$

$$= \frac{\frac{r}{r-2} \frac{x^{r-1}}{x^2-1}}{B\left(\frac{r}{2}, \frac{s}{2}\right) \left(1+\frac{r}{s}x\right)^{\frac{r+s}{2}}} \quad (6)$$

where $x > 0$.

This distribution is utilized to determine the difference among a value that actually was obtained and a theoretical value. This may be applied successfully in applications like weather forecasting to calculate the precipitation probability. In general, we can utilize F-distribution in analysis of variance and regression analysis.

Now, let us see what shape of graph, F-distribution shall construct

Average can be obtained using:

$$E[P] = \frac{s}{s-2} \quad (s \geq 3) \quad (7)$$

And Variance can be calculated using:

$$V[P] = \frac{2s^2(r+s-2)}{r(s-2)^2(s-4)} \quad (s \geq 5) \quad (8)$$

Basic Properties

The formula of probability density is quite complicated for the F-distribution. In practical problems, we don't need to be much concerned with probability density formula of F-distribution. However, it can be helpful to have some knowledge of the characteristics concerned to the F-distribution. Some of the important properties of F-distribution are as follows:

- There are infinity of F-distributions and hence these distributions constitute a family of various distributions. For a specific application, the F-distribution we apply depends, as with chi-square-distribution and t-distribution, on the number of degree of freedom in the sample under consideration.
- The F-distribution is always greater than or equal to zero i.e., positive or zero which means F is never negative in similar to that of chi-square distribution.
- The F-distribution is always rightly skewed. Hence, F-distribution is nonsymmetrical in nature as is the case with the chi-square distribution.

There are few more easily identified and important features of this distribution. We will focus more keenly at the degrees of freedom.

Degrees of Freedom

F-distributions, chi-square distributions, and t-distributions are all infinite families of distributions that all have one trait in common: In order to determine which distribution a certain data set belongs to, we must calculate the number of degrees of freedom. The degrees of freedom of a t-distribution is equal to one less than the number of observations we have. While for a chi-square distribution, degrees of freedom are defined using a "F" instead of "X," a "F" is also used to identify degrees of freedom for an F-distribution.

In the following paragraphs we will see how exactly an F-distribution arises. Right now, we will consider how to obtain the number of degrees of freedom. The F-distribution is determined using a ratio based on two populations. From both the populations a sample is obtained and from both of these sample's degrees of freedom is determined. In fact, the two numbers of degrees of freedom are obtained by subtracting both of the sample sizes by one to determine degrees of freedom.

Statistics from both populations add up in a fraction for the F-distributions. Both the denominator and numerator have degrees of freedom. Instead of adding up these two numbers to obtain another, both of them are retained. Hence, any use of an F-statistic table needs us to see two different degrees of freedom.

Uses of the F-Distribution

The F-distribution is obtained from inferential statistics using population variances. More particularly, we employ an F-distribution only when we are determining the ratio of the variances of two different populations which are normally distributed.

The F-distribution is not utilized solely to test hypotheses and determine confidence intervals about population variances. This kind of distribution is also applied in case of one-factor analysis of variance. Analysis of variance is usually concerned with comparing inter-group and intra-group variations. To achieve this, we use a ratio of variances. This ratio of variances has the F-distribution. A somewhat complex formula helps us to determine an F-distribution as a test statistic.

5.4 DISTRIBUTIONS OF QUADRATIC FORMS UNDER NORMALITY

The distribution of quadratic type $y' Sy$ under normality, where $y \sim W_j(\vartheta, \Sigma)$ where Σ can be singular or not. As illustrated in example given below:

Suppose $S = I_2$ and $y = (z, 1)'$, where $z \sim W(0,1)$. Then $y \sim W_2(\vartheta, \Sigma)$, where

$$\vartheta = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \Sigma = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

It is clear that $(S\Sigma)^2 = S\Sigma$ and its rank $r(S\Sigma) = 1$, but

$$y' Sy = z^2 + 1, \quad (9)$$

the distribution of that is not $\chi_1^2(\lambda)$ with $\lambda = \frac{1}{2} \vartheta' S \vartheta = \frac{1}{2}$.

Example: Suppose that y_1, \dots, y_n is a sample randomly obtained from a $N(\mu, \sigma^2)$ distribution. Show that $\frac{\sum_{i=1}^n (y_i - \bar{y})^2}{\sigma^2} \sim \chi^2(n-1)$

Answer: $\frac{\sum_{i=1}^n (y_i - \bar{y})^2}{\sigma^2} = y' \frac{(I - \frac{1}{n}J)}{\sigma^2} y$

By theorem, $I - \frac{1}{n}J$ is idempotent, therefore all of its eigen values are either 1 or 0 with its rank equal to the number of eigen values that are 1. The sum of eigen values of $I - \frac{1}{n}J$ is

$$\text{tr}(I - \frac{1}{n}J) = \text{tr}(I) - \frac{1}{n} \text{tr}(J) = n - \frac{1}{n}n = n - 1,$$

So rank $(I - \frac{1}{n}J) = n - 1$. The non centrality parameter is

$$\begin{aligned} \lambda &= \frac{1}{2}(\mu j)' \left(\frac{1}{\sigma^2} \left(I - \frac{1}{n}J \right) \right) (\mu j) \\ &= \frac{\mu^2}{2} j^\top \left(I - \frac{1}{n}J \right) j \\ &= \frac{\mu^2}{2} \left(j^\top j - \frac{1}{n} j^\top j j^\top j \right) \\ &= \frac{\mu^2}{2} \left(n - \frac{1}{n} n^2 \right) \\ &= 0. \end{aligned}$$

Therefore, by theorem, $\frac{\sum_{i=1}^n (y_i - \bar{y})^2}{\sigma^2}$ is a chi square random variable having rank $(I - \frac{1}{n}J) = (n-1)$ degrees of freedom.

5.5 INDEPENDENCE OF QUADRATIC FORM AND A LINEAR FORM

- **Theorem 1:** If $y \sim Np(\mu, \Sigma)$, A is a matrix of $p \times p$ of constants and B is a matrix of $k \times p$ of constants, then By and $y^\top Ay$ are independent iff $B\Sigma A = 0$.
- **Theorem 2:** If $y \sim Np(\mu, \Sigma)$ and A and B are symmetric matrices of $p \times p$ of constants, then $y^\top Ay$ and $y^\top By$ are independent iff $A\Sigma B = 0$.
- **Example 1.1:** Lets suppose that $y \sim Np(\mu, \sigma^2 I)$ and H is a symmetric idempotent matrix of size $p \times p$ of constants having rank $r < p$ with $\mu^\top (I - H)\mu = 0$. Then the distribution of

$$\frac{y^\top Hy/r}{y^\top (I-H)y/(p-r)}?$$

- *Answer:* Note the $1/\sigma y \sim Np(1/\sigma \mu, I)$. Since H is idempotent matrix having rank r and $I-H$ is idempotent with rank $p-r$, Theorem 1.1 implies that

$$\left(\frac{1}{\sigma} y \right)' H \left(\frac{1}{\sigma} y \right) = \frac{1}{\sigma^2} y^\top (H) y \sim \chi^2 \left(r, \frac{1}{2\sigma^2} \mu^\top H \mu \right)$$

And

$$\left(\frac{1}{\sigma} y \right)' (I - H) \left(\frac{1}{\sigma} y \right) = \frac{1}{\sigma^2} y^\top (I - H) y \sim \chi^2 (p - r) \quad \text{since } \mu^\top (I - H)\mu = 0.$$

By Theorem 1.2, $y^\top Hy$ and $y^\top (I - H)y$ are independent since $H(I - H) = 0$. So, by Definition 1.3, we see that

$$\frac{\frac{y^T H y}{r}}{\frac{y^T (I - H) y}{(p - r)}} = \frac{\left(\frac{1}{\sigma^2} y^T H y\right) / r}{\frac{1}{\sigma^2} y^T (I - H) y / (p - r)} \sim F(r, p - r, \frac{1}{2\sigma^2} \mu^T H \mu)$$

- **Theorem 1.3:** Let $y \sim N_n(\mu, \sigma^2 I)$, A_i is an $n \times n$ symmetric matrix of rank r_i for $i = 1, \dots, k$, and

$$y^T y = \sum_{i=1}^k y^T A_i y$$

Then $\frac{1}{\sigma^2} y^T A_i y \sim \chi^2(r_i, \frac{1}{2\sigma^2} \mu^T A_i \mu)$ for $i = 1, \dots, k$ and

and $y^T A_1 y, \dots, y^T A_k y$ are mutually independent iff at least one of the following statements are true:

- A_1, \dots, A_k are idempotent matrices
- $A_i A_j = O$ for all $i \neq j$
- $n = \sum_{i=1}^k r_i$

5.6 COCHRAN'S THEOREM

Statistically, Cochran's theorem has been utilized for the analysis of variance. It specifies about the distributions of divided sums of squares of normally distributed random variables. A Cochran theorem is given in 1934 that states about the distribution of partitioning of sums of chi-squared variables. Suppose \mathbf{Z} represents a $n \times 1$ vector of independent standard normal random variables and suppose $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3, \dots, \mathbf{X}_k$ be non-zero symmetric matrices such that where \mathbf{Z}' is the transpose of \mathbf{Z} . Let $Q_j = \mathbf{Z}' \mathbf{X}_j \mathbf{Z}$. Cochran's theorem states that, if any one of the below mentioned three conditions hold, then so are the other two:

- The ranks of $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3, \dots, \mathbf{X}_k$ sum to n .
- Each of $Q_1, Q_2, Q_3, \dots, Q_k$ has a chi-squared distribution.
- Each of $Q_1, Q_2, Q_3, \dots, Q_k$ is independent of all the others.

Suppose S_1, S_2, \dots, S_k are independent standard random variables which are normally distributed, and an identity of the type

$$\sum_{i=1}^k U_i^2 = Q_1 + \dots + Q_k \quad (10)$$

can be written where each Q_i is a sum of squares of linear combinations of the U 's. Further suppose that

$$r_1 + \dots + r_k = n \quad (11)$$

where r_i is the rank of Q_i . Cochran's theorem states that the Q_i are independent, and Q_i has a chi-square distribution with r_i degrees of freedom.

Cochran's theorem is the converse of Fisher's theorem.

Example

If X_1, \dots, X_n are independent normally distributed random variables with mean μ and standard deviation σ then

$$U_i = (X_i - \mu) / \sigma \quad (12)$$

is standard normal for each i . It is possible to write

$$\sum U_i^2 = \sum \left(\frac{X_i - \bar{X}}{\sigma} \right)^2 + n \left(\frac{\bar{X} - \mu}{\sigma} \right)^2 \quad (13)$$

(here, summation is from 1 to n , that is over the observations). To see this identity, multiply throughout by σ and note that

$$\sum (X_i - \mu)^2 = \sum (X_i - \bar{X} + \bar{X} - \mu)^2 \quad (14)$$

and expand to give

$$\sum (X_i - \bar{X})^2 + \sum (\bar{X} - \mu)^2 + 2 \sum (X_i - \bar{X})(\bar{X} - \mu) \quad (15)$$

The third term is zero because it is equal to a constant time

$$\sum (\bar{X} - X_i) \quad (16)$$

and the second term is just n identical terms added together.

Combining the above results (and dividing by σ^2), we have:

$$\sum \left(\frac{\bar{X} - \mu}{\sigma} \right)^2 = \sum \left(\frac{X_i - \bar{X}}{\sigma} \right)^2 + n \left(\frac{\bar{X} - \mu}{\sigma} \right)^2 = Q_1 + Q_2 \quad (17)$$

Now the rank of Q_2 is just 1 (it is the square of just one linear combination of the standard normal variables). The rank of Q_1 can be shown to be $(n - 1)$, and thus the conditions for Cochran's theorem are met.

Cochran's theorem then states that Q_1 and Q_2 are independent, with chi-squared distribution having $(n - 1)$ and 1 degree of freedom respectively.

This shows that the sample mean and sample variance are independent; also

$$(\bar{X} - \mu)^2 \sim \frac{\sigma^2}{n} \chi_1^2 \quad (18)$$

To estimate the variance σ^2 , one estimator that is often used is

$$\hat{\sigma}^2 = \frac{1}{n} \sum (X_i - \bar{X})^2 \quad (19)$$

Cochran's theorem shows that

$$\hat{\sigma}^2 \sim \frac{\sigma^2}{n} \chi_{n-1}^2 \quad (20)$$

which shows that the expected value of $\hat{\sigma}^2$ is $\sigma^2 (n - 1)/n$.

Both these distributions are proportional to the true but unknown variance σ^2 ; thus, their ratio is independent of σ^2 and because they are independent, we have

$$\frac{(\bar{X}-\mu)^2}{\frac{1}{n}\sum(X_i-\bar{X})^2} \sim F_{1,n} \quad (21)$$

where $F_{1,n}$ is the F-distribution with 1 and n degrees of freedom (see also Student's t-distribution).

5.7 SUMMARY

The students will learn many things related to sampling distribution in this module and they will be able to perform the various sampling distribution related operation using Python.

- Ability to do the statistical distribution including non-central chi square test.
- Able to perform T and F distributions with their properties
- Also have knowledge about Distributions of quadratic forms under normality and Independence of quadratic form and a linear form and Cochran's theorem

5.8 REFERENCES

Books

1. The Sampling Distribution and Central Limit Theorem by Douglas Brooks
2. Schaum's Outline of Statistics, Sixth Edition, Dr. Murray R. Spiegel, Larry J. Stephens, ISBN: 9781260011463, 2018011, McGraw-Hill Education
3. Sukhatme P.V., Sukhatme B.V. and C. Asok Sampling theory of survey and applications (Indian society for Agricultural statistics) (2) Des Raj & Chandhok P.(1998), Sample survey theory (Narosa) (3) W. G. Cochran ,(1977) Sampling techniques (John Wiley and sons) (4) Murthy M.N.(1977) Sampling theory and methods (Statistical Publishing Society).

Web References

1. <https://www.geeksforgeeks.org>
2. <https://www.tutorialspoint.com>
3. <https://www.w3schools.com>
4. <https://pandas.pydata.org>
5. <https://pbpython.com>

5.9 PRACTICE EXERCISE

F-distribution

- If X and Y are independent random variables with χ^2 -distribution with m and n degrees of freedom respectively, then the distribution of the statistic $F = \frac{X/m}{Y/n}$ is called the F -distribution with m and n degrees of freedom. The probability density function of F is

$$f_F(f) = \frac{\Gamma\left(\frac{m+n}{2}\right)\left(\frac{m}{n}\right)^{m/2}}{\Gamma\left(\frac{m}{2}\right)\Gamma\left(\frac{n}{2}\right)} f^{\left(\frac{m-2}{2}\right)} \left(1 + \left(\frac{m}{n}\right)f\right)^{-\left(\frac{m+n}{2}\right)}; \quad 0 < f < \infty$$

- If X has a noncentral Chi-square distribution with m degrees of freedom and noncentrality parameter λ ; Y has a χ^2 distribution with n degrees of freedom, and X and Y are independent random variables, then the distribution of $F = \frac{X/m}{Y/n}$ is the noncentral F distribution with m and n degrees of freedom and noncentrality parameter λ .
1. Find Variance for an F-Distribution with $v_1=5$ and $v_2=9$
 2. How to Calculate Sampling Distribution
 3. Conduct an F-Test on the following samples:
 - Sample-1 having variance = 109.63, sample size = 41.
 - Sample-2 having Variance = 65.99, sample size = 21.

B.Sc.(DATA SCIENCE)

SEMESTER-III

DATA PREPARATION

UNIT VI: SIMPLE RANDOM SAMPLING (WR AND WOR)

STRUCTURE

6.0 Objectives

6.1 Introduction

6.2 Use of Random Number Table

6.3 Stratified Random Sampling

6.4 Advantages of Stratification

6.5 Properties of Stratified Random Sampling

6.6 Systematic Sampling: Estimation of The Mean and Variance

6.7 Comparison of Simple, Stratified and Systematic Sampling

6.8 Key Takeaways

6.9 Summary

6.10 References

6.11 Exercise

6.0 OBJECTIVES

After reading this unit, you should be able to select samples using simple random sampling, stratified sampling and systematic sampling procedures. The aim of the simple random sample is to reduce the potential for human bias in the selection of cases to be included in the sample. As a result, the simple random sample provides us with a sample that is highly representative of the population being studied, assuming that there is limited missing data.

6.1 INTRODUCTION

In our day to day routine we often have to make certain judgements about a large bulk or population after studying a small portion, or a sample of it. For example, a house wife tastes a spoonful of soup to see whether a little more salt is required before it is served to guests; a quality control inspector inspects a sample from a tin of oil before passing the whole tin as acceptable quality; or a doctor takes a few drops of blood from a patient to decide if the patient has malarial infection. These are typical examples where it is not practical to examine the entire lot or population and decision making is done on the basis of sample information. Essentially this is sampling. It saves time and if you have based your judgement on judicious observations, it serves your purpose. Think of situations, when decisions are to be taken at macro levels, say, national level. Sampling has become an effective tool for generating information for policy formulation at different administrative levels. The above examples have the special phenomenon that a spoon of soup, tinful of oil and the blood in the patient are known to be perfectly homogeneous material so that every part of the material represents the material exactly. Often, however, we are not in this simple situation. For example, suppose we are interested in knowing the average weight of an adult Indian male. Obviously, it will not be satisfactory to measure just a few adult males, as not all adult males are of the same weight. They show considerable heterogeneity. So how do we take a part of a large heterogeneous mass to draw valid conclusions from it? However, careful considerations are needed for selection of samples and making valid inferences from these samples. In the subject of 'sampling', these considerations and criteria are developed on a scientific basis. In this unit, we shall start by introducing the basic concepts of sampling. We shall then discuss the simplest procedure for sample collection i.e., simple random sampling (SRS). In the process you will get introduced to the concept of random numbers, simple random sampling as a method of selection, estimation as well as considerations for determination of sample size. A simple random sample is a subset of a statistical population in which each member of the subset has an equal probability of being chosen. A simple random sample is meant to be an unbiased representation of a group.

An example of a simple random sample would be the names of 25 employees being chosen out of a hat from a company of 250 employees. In this case, the population is all 250 employees, and the sample is random because each employee has an equal chance of being chosen. Random sampling is used in science to conduct randomized control tests or for blinded experiments.

Simple random sampling (SRS) is a method of selection of a sample comprising of n a number of sampling units out of the population having N number of sampling units such that

every sampling unit has an equal chance of being chosen. The samples can be drawn in two possible ways.

- The sampling units are chosen without replacement because the units once are chosen are not placed back in the population.
- The sampling units are chosen with replacement because the selected units are placed back in the population

Simple random sampling without replacement (SRSWOR): SRSWOR is a method of selection of n units out of the N units one by one such that at any stage of selection, any one of the remaining units have the same chance of being selected, i.e. $1/N$.

Simple random sampling with replacement (SRSWR): SRSWR is a method of selection of n units out of the N units one by one such that at each stage of selection, each unit has an equal chance of being selected, i.e., $1/N$. Procedure of selection of a random sample: The procedure of selection of a random sample follows the following steps:

1. Identify the N units in the population with the numbers 1 to N .
2. Choose any random number arbitrarily in the random number table and start reading numbers.
3. Choose the sampling unit whose serial number corresponds to the random number drawn from the table of random numbers.
4. In the case of SRSWR, all the random numbers are accepted even if repeated more than once. In the case of SRSWOR, if any random number is repeated, then it is ignored, and more numbers are drawn

6.2 USE OF RANDOM NUMBER TABLE

If squares of k independent standard normal random variables are added, it gives rise to If each sample among the all possible samples has the same chance of being selected, then the associated method is called simple random sampling. In simple random sampling procedure with unit by unit selection, every unit has got equal chance (probability) of selection at every draw. However, the converse is not true i.e. there are sampling schemes in which every unit gets the same chance of selection but they are not simple random sampling methods e.g. the systematic sampling. You shall learn about such sampling methods later in this unit. We now try to answer the question which may be occurring in your mind. How to select a simple random sample (SRS)? We consider the selection of a simple random sample through unit by unit selection method. At every draw equal probabilities are assigned to the available sampling units of the population. Thus, a pre-requisite for the selection is a random device by which selections are to be made. The most commonly used procedures for selecting a SRS are (1) lottery method, (2) through the use of random number tables.

Through random number tables before discussing the method based on the use of random number tables you may like to know what random numbers are? Random numbers are numbers generated by a random procedure involving repeated independent trials. Such numbers are generated with the help of random digits 0 through 9. When we say random digits 0 through 9 it is assumed that a trial of the procedure yields each of the ten digits with probability 0.1. One simple way of generating these random digits is to take ten cards of the same size and write the digits 0 through 9 on the cards so that each card has a different digit.

Then take a large hat, say, toss in the cards and mix them well. Now choose a card at random from the hat. Write down on a piece of paper the digit appearing on the card you have chosen. Put the card back into the hat and mix the cards again. Repeat the procedure by choosing a card at random, writing down the digit appearing on the card, replacing, mixing, choosing again, and so on. The string of digits we write down constitutes a string of random digits because it has been produced by a random device supposed to yield each digit with probability 0.1 in independent trials. Random digits can also be produced by using a modified roulette wheel in which the wheel is divided into ten equal parts, each one corresponding to one of the ten digits. Given random digits, we can get more complicated random numbers. Suppose we have generated the sequence 3217900597 of ten digits. Then each digit is random, and also the two digit numbers 32, 17,90,05,97 obtained by taking the numbers two at a time are random numbers because they have been produced by a random procedure so that each of one hundred two-digit numbers 00 through 99 has the probability 0.01 of appearing, and moreover selection of these two digit numbers are independent. Taking the original ten digit sequence and choosing the two digits at a time going backward to get 79,50,09,71,23 also gives random two-digit numbers. In the same way you might think of some other ways to get two-digit numbers using the generated string, as long as the method does not use the same selection more than once.

E5) Give two more sequences of 5 two digit random numbers obtained by using the string 3217900597 of ten digits.

In a similar manner, random numbers with three, four or even more digits can be obtained by using the given string of random digits. There are several standard random number tables available which give the arrangement of these numbers in a rectangular manner. Some of these which are commonly used are prepared by Tippett (1927), Fisher and Yates (1938), Kendall and Smith (1939), Rand Corporation (1955), and Rao et al. (1974). One such random number tables is reproduced in Appendix A. You may note that this random number table can be used as single digit numbers or two digit numbers or, three or four digit numbers depending on the size of the population you are sampling from. We shall now illustrate the use of random number tables for selecting samples. We shall discuss here three commonly used methods of using random number tables for selection of simple random samples. Direct Approach. The first step in the method is to assign serial numbers 1 to N to the N population units. If the population size N is made up of K digit., then consider K digit random numbers, either row wise or column wise, in the random number table. The sample of required size is then selected by drawing, one by one, random numbers from 1 to N, and including the units bearing these serial numbers in the sample.

Problem 1: Consider a population of 56 households. Select a simple random sample Simple Random Sampling of 10 households by with replacement as well as without replacement methods.

Solution: Here the sampling unit is a household. The first step is to serially arrange the households if they are not already arranged so. Since, the population size is a number consisting of two digits we have to use two digit random number table. Alternatively, in

the table of random numbers the first two digits of any column can be used randomly. While using these tables, it is advisable to take a blind start on the table by placing your finger on the table with closed eyes - let it be column 7, row 6 of the first page of Appendix A. Then the first number is 20 and going down the page subsequent random numbers between 1 to 56 are 12,03 etc. By selecting first 10 random numbers from 1 to 56, without discarding repetitions for with replacement procedure (WR), we obtain the serial numbers of the households in the sample. These are given below: For without replacement procedure (WOR), repetitions have to be avoided and the number 15, when appears again at the tenth draw, should be dropped. The next two digits are then chosen. Thus, a WOR sample will consist of: -- 20 12 03 16 30 15 24 37 01 07

Problem 1: Consider a population of 56 households. Select a simple random sample of 10 households by with replacement as well as without replacement methods.

Solution: Here the sampling unit is a household. The first step is to serially arrange the households if they are not already arranged so. Since, the population size is a number consisting of two digits we have to use two digit random number table. Alternatively, in the table of random numbers (Appendix A) the first two digits of any column can be used randomly. While using these tables, it is advisable to take a blind start on the table by placing your finger on the table with closed eyes – let it be column 7, row 6 of the first page of Appendix A. Then the first number is 20 and going down the page subsequent random numbers between 1 to 56 are 12, 03 etc.

By selecting first 10 random numbers from 1 to 56, without discarding repetitions for **with replacement procedure (WR)**, we obtain the serial numbers of the households in the sample. These are given below:

20 12 03 16 30 15 24 37 01 15

For **without replacement procedure (WOR)**, repetitions have to be avoided and the number 15, when appears again at the tenth draw, should be dropped. The next two digits are then chosen. Thus, a WOR sample will consist of:

20 12 03 16 30 15 24 37 01 07

This procedure may involve number of rejections of random numbers, since zero and all the numbers greater than 56 appearing in the table are not considered for selection. The use of random numbers has, therefore, to be modified. We now discuss two of the commonly used modified procedures,

6.3 STRATIFIED RANDOM SAMPLING

The precision of a simple random sample estimate depends upon

- (i) the size of the sample and
- (ii) the variability (or heterogeneity) of the population.

The size of the sample cannot be unduly increased; hence the only way to increase the precision of the estimate is to devise procedure which will effectively reduce the variability. One such procedure is known as the procedure of stratified random sampling. It consists in dividing the entire population into several non-overlapping classes or strata and drawing simple random samples from each of these strata and, then, combining all sample units together. It ensures any desired representation of units in all the strata and is intended to give a better cross-section of the population than that of unstratified random sampling. It follows that it will enhance the precision of the estimate since each strata within itself will be more homogeneous than the entire population.

6.4 ADVANTAGES OF STRATIFICATION

- (a) If data known precision are wanted for certain subdivisions of the population, it is advisable to treat each 'subdivision' as a 'population' in its own right.
- (b) Administrative convenience may tackle the use of stratification: for illustration, the agency considering the survey may have field offices, each of which can supervise the survey for a part of the population.
- (c) Sampling problems may differ markedly in different portions of the population: for example, these may be different types of sampling problems in plains, hilly areas and deserts which may need different approaches.
- (d) Stratification is particularly more effective when there are extreme values in the population which can be segregate to from different strata: for example, the adult population may be divided into higher income, lower and unemployed sections.
- (e) Stratification will lead to gain in precision since it may be possible to divide a heterogeneous population into sub-populations which are internally homogenous (as compared to whole population).

Let the population consisting of N units be divided into K strata. Let

$$N_h = \text{number of units in the } h\text{th Stratum } \left(\sum_1^k N_h = N \right)$$

$$y_{hi} = \text{value of } y \text{ in the } i\text{th units of the } h\text{th Stratum} \\ (i=1, \dots, N_h ; h = 1, \dots, k)$$

$$\bar{Y}_h = \sum_{i=1}^{N_h} y_{hi} / N_h \text{ is the } h\text{th stratum mean}$$

$$\bar{Y} = \sum_{h=1}^k \sum_{i=1}^{N_h} y_{hi} / N \text{ is the population mean}$$

$$S_h^2 = \sum_{i=1}^{N_h} (y_{hi} - \bar{Y}_h)^2 / (N_h - 1)$$

$$n_h = \text{number of units in the sample from the } h\text{th Stratum } \left(\sum_1^k n_h = n \right)$$

$$\bar{y}_h = \sum_i^{n_h} y_{hi} / n_h \text{ is the sample mean from the } h\text{th stratum}$$

$$\bar{y} = \sum_1^k n_h \bar{y}_h / n \text{ is the overall sample mean}$$

$$s_h^2 = \sum_1^{n_h} (y_{hi} - \bar{y}_h)^2 / (n_h - 1)$$

$$s_h^2 = \frac{1}{n_h} \sum_{i=1}^{n_h} (y_{hi} - \bar{y}_h)^2 / (n_h - 1)$$

$$W_n = Nn/N, \quad f_h = n_h/N_h, \text{ sampling fraction}$$

Definition: The stratified random sampling estimate \bar{y}_{st} of population mean \bar{Y} is defined by

$$\bar{y}_{st} = \frac{\sum_{h=1}^k N_h \bar{y}_h}{N}$$

which is, generally, different for \bar{y} unless $\frac{N_h}{N} = \frac{n_h}{n}$ in which case we have “proportional allocation”.

When the units of the population are scattered and not completely homogeneous in nature, then simple random sample does not give proper representation of the population. So if the population is heterogeneous the simple random sampling is not found suitable. In simple random sampling the variance of the sample mean is proportional to the variability of the sampling units in the population. So, in spite of increasing the sample size n or sampling fraction n/N , the only other way of increasing the precision is to devise a sampling which will effectively reduce the variability of the sample units, the population heterogeneity. One such method is stratified sampling method. In stratified sampling the whole population is to be divided in some homogeneous groups or classes with respect to the characteristic under study which are known as strata. That means, we have to do the stratification of the population. Stratification means division into layers. The auxiliary information related to the character under study may be used to divide the population into various groups or strata in such a way that units within each stratum are as homogeneous as possible and the strata are as widely different as possible. Thus, all strata would comprise the population. Then from each stratum sample would be drawn and lastly all samples would be combined to get the ultimate sample. For example, let us consider that population consists of N units and these are distributed in a heterogeneous structure. Now first of all 46 Statistical Techniques we divide the population into ‘ k ’ non overlapping strata of sizes $N_1, N_2, N_3, \dots, N_k$ such that each stratum becomes homogeneous. Evidently $N = N_1 + N_2 + N_3 + \dots + N_k$. Then from first stratum a sample of size n_1 would be drawn by simple random sampling method. Similarly, from the second stratum a sample of n_2 units would be drawn and so on, up to k th stratum. Now all these k samples would be combined to get the ultimate sample. So, the ultimate size of sample would be $n_1 + n_2 + n_3 + \dots + n_k = 1 + 2 + 3 + \dots + k$. This method of sampling is known as Stratified random sampling because here stratification is done first to make population homogeneous and then samples are drawn randomly by simple random sampling from each stratum.

The principles to be kept in mind while stratifying a population are given below:

1. The strata should not be overlapping and should together comprise the whole population.
2. The strata should be homogeneous within themselves and heterogeneous between themselves with respect to characteristic under study.
3. If an investigator is facing difficulties in stratifying a population with respect to the characteristic under study, then he/she has to consider the administrative convenience as the basis for stratification.
4. If the limit of precision is given for certain sub-population then it should be treated as stratum.

Steps to select a stratified random sample:

1. Define the target audience.
2. Recognize the stratification variable or variables and figure out the number of strata to be used. These stratification variables should be in line with the objective of the research. Every additional information decides the stratification variables. For instance, if the objective of research is to understand all the subgroups, the variables will be related to the subgroups and all the information regarding these subgroups will impact the variables. Ideally, no more than 4-6 stratification variables and no more than 6 strata should be used in a sample because an increase in stratification variables will increase the chances of some variables canceling out the impact of other variables.
3. Use an already existent sampling frame or create a frame that's inclusive of all the information of the stratification variable for all the elements in the target audience.
4. Make changes after evaluating the sampling frame on the basis of lack of coverage, over-coverage, or grouping.
5. Considering the entire population, each stratum should be unique and should cover each and every member of the population. Within the stratum, the differences should be minimum whereas each stratum should be extremely different from one another. Each element of the population should belong to just one stratum.
6. Assign a random, unique number to each element.
7. Figure out the size of each stratum according to your requirement. The numerical distribution amongst all the elements in all the strata will determine the type of sampling to be implemented. It can either be proportional or disproportional stratified sampling.
8. The researcher can then select random elements from each stratum to form the sample. Minimum one element must be chosen from each stratum so that there's representation from every stratum but if two elements from each stratum are selected, to easily calculate the error margins of the calculation of collected data.

6.5 PROPERTIES OF STRATIFIED RANDOM SAMPLING

Theorem 1: $E(\bar{x}_i) = \bar{X}$

Proof: We have

$$\bar{x}_{st} = \frac{1}{N} \sum_{i=1}^k N_i \bar{x}_i$$

Therefore,

$$E(\bar{x}_{st}) = E\left[\frac{1}{N} \sum_{i=1}^k N_i \bar{x}_i\right]$$

$$\frac{1}{N} \sum_{i=1}^k N_i E(\bar{x}_i)$$

Since the sample units selected from each of stratum are simple random sample, then we have

$$E(\bar{x}_i) = \bar{X}_i$$

Therefore,

$$E(\bar{x}_{st}) = \frac{1}{N} \sum_{i=1}^k N_i \bar{X}_i$$

$$= \frac{1}{N} \sum_{i=1}^k N_i \frac{1}{N_i} \sum_{j=1}^{N_i} X_{ij}$$

$$= \frac{1}{N} \sum_{i=1}^k \sum_{j=1}^{N_i} X_{ij} = \bar{X}$$

Hence, proved.

Theorem 2: Prove that

$$Var(\bar{x}_{st}) = \frac{1}{N^2} \sum_{i=1}^k N_i(N_i - n_i) \frac{S_i^2}{n_i} = \sum_{i=1}^k \left(\frac{1}{n_i} - \frac{1}{N}\right) W_i^2 S_i^2$$

Proof: We have

$$Var(\bar{x}_{st}) = Var\left(\sum_{i=1}^k W_i \bar{x}_i\right)$$

$$= \sum_{i=1}^k W_i^2 Var(\bar{x}_i)$$

The covariance term vanish since the samples from different strata are independent and the sample units in each stratum are the simple random sample without replacement, we have

$$Var(\bar{x}) = \left(\frac{1}{n} - \frac{1}{N}\right) S^2$$

Therefore,

$$Var(\bar{x}_{st}) = \sum_{i=1}^k W_i^2 \left(\frac{1}{n} - \frac{1}{N}\right) S^2$$

$$= \frac{1}{N^2} \sum_{i=1}^k N_i(N_i - n_i) \frac{S_i^2}{n_i}$$

From the above result the variance depends on S_i^2 the heterogeneity within the strata. Thus, if S_i^2 are small i.e. strata are homogeneous then stratified sampling schemes provides estimates with greater precision.

6.6 SYSTEMATIC SAMPLING: ESTIMATION OF THE MEAN AND VARIANCE

The systematic sampling technique is operationally more convenient than simple random sampling. It also ensures, at the same time that each unit has an equal probability of inclusion in the sample. In this method of sampling, the first unit is selected with the help of random numbers, and the remaining units are selected automatically according to a

predetermined pattern. This method is known as systematic sampling. Suppose the N units in the population are numbered 1 to N in some order. Suppose further that N is expressible as a product of two integers n and k , so that $N = nk$.

To draw a sample of size n ,

- select a random number between 1 and k .
- Suppose it is i .
- Select the first unit, whose serial number is i .
- Select every k^{th} unit after i^{th} unit.
- The sample will contain $i, i + k, 1 + 2k, \dots, i + (n - 1)k$ serial number units.

So the first unit is selected at random and other units are selected systematically. This systematic sample is called k^{th} systematic sample and k is termed as a sampling interval. This is also known as linear systematic sampling.

The observations in the systematic sampling are arranged as in the following table:

Systematic sample number		1	2	3	...	i	...	k
Sample composition	1	y_1	y_2	y_3	...	y_i	...	y_k
	2 :	y_{k+1}	y_{k+2}	y_{k+3}	...	y_{k+i}	...	y_{2k}
	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
	n	$y_{(n-1)k+1}$	$y_{(n-1)k+2}$	$y_{(n-1)k+3}$...	$y_{(n-1)k+i}$...	y_{nk}
Probability		$\frac{1}{k}$	$\frac{1}{k}$	$\frac{1}{k}$...	$\frac{1}{k}$...	$\frac{1}{k}$
Sample mean		\bar{y}_1	\bar{y}_2	\bar{y}_3	...	\bar{y}_i	...	\bar{y}_k

Example: Let $N = 50$ and $n = 5$. So $k = 10$. Suppose first selected number between 1 and 10 is 3 . Then systematic sample consists of units with following serial number $3, 13, 23, 33, 43$.

Advantages of systematic sampling

It is easier to draw a sample and often easier to execute it without mistakes. This is more advantageous when the drawing is done in fields and offices as there may be substantial saving in time. 2. The cost is low, and the selection of units is simple. Much less training is needed for surveyors to collect units through systematic sampling. 3. The systematic sample is spread more evenly over the population. So no large part will fail to be represented in the sample. The sample is evenly spread and cross-section is better. Systematic sampling fails in case of too many blanks.

Example: Suppose our population is $9,000$ students and we want to sample $1,200$ students. How do we sample these students systematically?

Since, $9000/1200 = 7.5$, we can perform a 1-in-7 systematic sample. Or, we should sample every 7th student. We can pick a starting point randomly from 1 to 600 and sample every 7th student from that on until we have reached 1200 samples.

How do we estimate the variance of this single systematic sample?

If the population is randomly ordered, then there is no problem. We can estimate the variance σ^2 by:

$$s^2 = \frac{\sum_{j=1}^M (y_j - \bar{y})^2}{M-1}$$

However, when the population is ordered, the systematic sampling is usually better than simple random sampling and the above formula will overestimate the variance.

When the population is periodic, the systematic sampling may be worse than the simple random sampling and the above formula will underestimate the variance since if the period k is chosen poorly, then the elements sampled may be too similar to each other.

6.7 COMPARISON OF SIMPLE, STRATIFIED AND SYSTEMATIC SAMPLING

In statistical analysis, the "population" is the total set of observations or data that exists. However, it is often unfeasible to measure every individual or data point in a population. Instead, researchers rely on samples. A sample is a set of observations from the population. The sampling method is the process used to pull samples from the population.

Simple random samples and stratified random samples are both common methods for obtaining a sample. A simple random sample is used to represent the entire data population and randomly selects individuals from the population without any other consideration.

A stratified random sample, on the other hand, first divides the population into smaller groups, or strata, based on shared characteristics. Therefore, a stratified sampling strategy will ensure that members from each subgroup is included in the data analysis.

6.8 KEY TAKEAWAYS

- Simple random and stratified random samples are statistical measurement tools.
- A simple random sample takes a small, basic portion of the entire population to represent the entire data set.
- The population is divided into different groups that share similar characteristics, from which a stratified random sample is taken.

Simple Random Sample

Simple random sampling is a statistical tool used to describe a very basic sample taken from a data population. This sample represents the equivalent of the entire population.

The simple random sample is often used when there is very little information available about the data population, when the data population has far too many differences to divide into various subsets, or when there is only one distinct characteristic among the data population.

For instance, a candy company may want to study the buying habits of its customers in order to determine the future of its product line. If there are 10,000 customers, it may use choose

100 of those customers as a random sample. It can then apply what it finds from those 100 customers to the rest of its base.

Statisticians will devise an exhaustive list of a data population and then select a random sample within that large group. In this sample, every member of the population has an equal chance of being selected to be part of the sample. They can be chosen in two ways:

- Through a manual lottery, in which each member of the population is given a number. Numbers are then drawn at random by someone to include in the sample. This is best used when looking at a small group.
- Computer-generated sampling. This method works best with larger data sets, by using a computer to select the samples rather than a human.

Using simple random sampling allows researchers to make generalizations about a specific population and leave out any bias. This can help determine how to make future decisions. So that candy company from the example above can use this tool to develop a new candy flavor to manufacture based on the current tastes of the 100 customers. But keep in mind, these are generalizations, so there is room for error. After all, it is a simple sample. Those 100 customers may not have an accurate representation of the tastes of the entire population.

Unlike simple random samples, stratified random samples are used with populations that can be easily broken into different subgroups or subsets. These groups are based on certain criteria, then randomly choose elements from each in proportion to the group's size versus the population. This method of sampling means there will be selections from each different group—the size of which is based on its proportion to the entire population. But the researchers must ensure the strata do not overlap. Each point in the population must only belong to one stratum so each point is mutually exclusive. Overlapping strata would increase the likelihood that some data are included, thus skewing the sample.

Stratified random sampling offers some advantages and disadvantages compared to simple random sampling. Because it uses specific characteristics, it can provide a more accurate representation of the population based on what's used to divide it into different subsets. This often requires a smaller sample size, which can save resources and time. In addition, by including sufficient sample points from each stratum, the researchers can conduct a separate analysis on each individual stratum. But more work is required to pull a stratified sample than a random sample. Researchers must individually track and verify the data for each stratum for inclusion, which can take a lot more time compared with random sampling.

When to use simple random sampling

Simple random sampling is used to make statistical inferences about a population. It helps ensure high internal validity: randomization is the best method to reduce the impact of potential confounding variables.

In addition, with a large enough sample size, a simple random sample has high external validity: it represents the characteristics of the larger population.

However, simple random sampling can be challenging to implement in practice. To use this method, there are some prerequisites:

- You have a complete list of every member of the population.
- You can contact or access each member of the population if they are selected.
- You have the time and resources to collect data from the necessary sample size.

Simple random sampling works best if you have a lot of time and resources to conduct your study, or if you are studying a limited population that can easily be sampled.

In some cases, it might be more appropriate to use a different type of probability sampling:

- **Systematic sampling** involves choosing your sample based on a regular interval, rather than a fully random selection. It can also be used when you don't have a complete list of the population.
- **Stratified sampling** is appropriate when you want to ensure that specific characteristics are proportionally represented in the sample. You split your population into strata (for example, divided by gender or race), and then randomly select from each of these subgroups.
- **Cluster sampling** is appropriate when you are unable to sample from the entire population. You divide the sample into clusters that approximately reflect the whole population, and then choose your sample from a random selection of these clusters.

When to use stratified sampling

To use stratified sampling, you need to be able to divide your population into mutually exclusive and exhaustive subgroups. That means every member of the population can be clearly classified into exactly one subgroup.

Stratified sampling is the best choice among the probability sampling methods when you believe that subgroups will have different mean values for the variable(s) you're studying. It has several potential advantages:

- Ensuring the diversity of your sample

A stratified sample includes subjects from every subgroup, ensuring that it reflects the diversity of your population. It is theoretically possible (albeit unlikely) that this would not happen when using other sampling methods such as simple random sampling.

- Ensuring similar variance

If you want the data collected from each subgroup to have a similar level of variance, you need a similar sample size for each subgroup.

With other methods of sampling, you might end up with a low sample size for certain subgroups because they're less common in the overall population.

- Lowering the overall variance in the population

Although your overall population can be quite heterogeneous, it may be more homogenous within certain subgroups.

For example, if you are studying how a new schooling program affects the test scores of children, both their original scores and any change in scores will most likely be highly correlated with family income. The scores are likely to be grouped by family income category.

In this case, stratified sampling allows for more precise measures of the variables you wish to study, with lower variance within each subgroup and therefore for the population as a whole.

- Allowing for a variety of data collection methods

Sometimes you may need to use different methods to collect data from different subgroups. For example, in order to lower the cost and difficulty of your study, you may want to sample urban subjects by going door-to-door, but rural subjects using mail. Research example You are interested in how having a doctoral degree affects the wage gap between men and women among graduates of a certain university. Because only a small proportion of this university's graduates have obtained a doctoral degree, using a simple random sample would likely give you a sample size too small to properly compare the differences between men and women with a doctoral degree versus those without one. Therefore, you decide to use a stratified sample, relying on a list provided by the university of all its graduates within the last ten years.

When to use systematic sampling

Systematic sampling is a method that imitates many of the randomization benefits of simple random sampling, but is slightly easier to conduct. You can use systematic sampling with a list of the entire population, as in simple random sampling. However, unlike with simple random sampling, you can also use this method when you're unable to access a list of your population in advance.

Order of the population

When using systematic sampling with a population list, it's essential to consider the order in which your population is listed to ensure that your sample is valid. If your population is in ascending or descending order, using systematic sampling should still give you a fairly representative sample, as it will include participants from both the bottom and top ends of the population.

For example, if you are sampling from a list of individuals ordered by age, systematic sampling will result in a population drawn from the entire age spectrum. If you instead used

simple random sampling, it is possible (although unlikely) that you would end up with only younger or older individuals. You should not use systematic sampling if your population is ordered cyclically or periodically, as your resulting sample cannot be guaranteed to be representative.

Example: Alternating list Your population list alternates between men (on the even numbers) and women (on the odd numbers). You choose to sample every tenth individual, which will therefore result in only men being included in your sample. This would obviously be unrepresentative of the population. Example: Cyclically ordered list. You are sampling from a population list of approximately 1000 hospital patients. The list is divided into 50 departments of around 20 patients each. Within each department, the list is ordered by age, from youngest to oldest. This results in a list of 20 repeated age cycles.

If you sample every 20th individual, because each department is ordered by age, your population will consist of the oldest person in each one. This will most likely not provide a representative sample of the entire hospital population.

Systematic sampling without a population list

You can use systematic sampling to imitate the randomization of simple random sampling when you don't have access to a full list of the population in advance.

You run a department store and are interested in how you can improve the store experience for your customers. To investigate this question, you ask an employee to stand by the store entrance and survey every 20th visitor who leaves, every day for a week. Although you do not necessarily have a list of all your customers ahead of time, this method should still provide you with a representative sample of your customers since their order of exit is essentially random.

6.9 SUMMARY

The students will learn many things related to sampling distribution in this module and they will be able to perform the various sampling techniques.

- Able to do Simple Random Sampling.
- Able to do Stratified Random Sampling.
- Able to do Systematic Sampling.

6.10 REFERENCES

Books

1. Sampling Techniques : W.G. Cochran, Wiley (Low price edition available)
2. Theory and Methods of Survey Sampling : Parimal Mukhopadhyay, Prentice Hall of India
3. Theory of Sample surveys with applications : P.V. Sukhatme, B.V Sukhatme, S. Sukhatme and C. Asok, IASRI, Delhi
4. Sampling Methodologies and Applications : P.S.R.S. Rao, Chapman and Hall/ CRC

5. Sampling Theory and Methods : M.N. Murthy, Statistical Publishing Society, Calcutta (Out of print)
6. Elements of sampling theory and methods : Z. Govindrajalu, Prentice Hall

Web References

1. <https://www.geeksforgeeks.org>
2. <https://www.tutorialspoint.com>
3. <https://www.w3schools.com>
4. <https://pandas.pydata.org>
5. <https://pbpython.com>

6.11 EXERCISE

1. Why is a simple random sample 'simple'?
2. What are some drawbacks of a simple random sample?
3. What is a stratified random sample?
4. How are random samples used?
5. How the units in simple random sampling WR and WOR has the equal probability of being selected (1/N)?

Long Questions

1. Define simple random sampling without replacement from a finite population. Derive the unbiased estimator of the population mean and find its sampling variance.
2. Explain the concept of post stratification. Show that, for large samples, the post stratification is as precise as stratified sampling with proportional allocation.
3. What is the relative efficiency of systematic sampling over simple random sampling?
4. Carry out a comparison of simple random sampling, stratified random sampling and systematic sampling in the presence of a linear trend in the population.
5. Consider a population consisting of 5 villages, the areas (in hectares) of which are given below

Villages	V	W	X	Y	Z
Areas	760	343	657	550	480

Enumerate all possible WOR samples of size 3. Also write the values of the study variable (area) for the sampled units. List all the WR samples of size 3 along with their area values.

B.Sc.(DATA SCIENCE)

SEMESTER-IV

DATA PREPARATION

UNIT VII: TESTING OF HYPOTHESIS

STRUCTURE

- 7.0 Objectives**
- 7.1 Introduction**
- 7.2 Testing the Hypotheses**
- 7.3 Procedure of Hypothesis testing**
- 7.4 Tools for Data Wrangling:**
- 7.5 Test statistics and Critical regions**
- 7.6 Critical value and Decision Rule**
- 7.7 Merging Dataset**
- 7.8 Reshaping Dataset**
- 7.9 Data Transformation**
- 7.10 String Manipulation**
- 7.11 Regular Expression**
- 7.12 Summary**
- 7.13 References**

7.0 OBJECTIVES

The main goal of this module is to help students learn, understand the Hypothesis, which includes the types of testing hypotheses with specific parameters. This module is different from other modules. It will address the second area of statistical inference, which is hypothesis testing. A specific statement or hypothesis is generated about a population parameter, and sample statistics are used to assess the likelihood that the hypothesis is true. The hypothesis is based on available information and the investigator's belief about the population parameters. This module will focus on hypothesis testing for means and proportions.

7.1 INTRODUCTION

Hypothesis testing is a key concept in statistics, analytics, and data science. A hypothesis is a claim or statement about a characteristic of a population of interest to us. A hypothesis test is a way for us to use our sample statistics to test a specific claim. When conducting scientific research, typically there is some known information, perhaps from some past work or from a long accepted idea. We want to test whether this claim is believable. This is the basic idea behind a hypothesis test:

- State what we think is true.
- Quantify how confident we are about our claim.
- Use sample statistics to make inferences about population parameters.

For example, past research tells us that the average life span for a hummingbird is about four years. You have been studying the hummingbirds in the southeastern United States and find a sample mean lifespan of 4.8 years. Should you reject the known or accepted information in favor of your results? How confident are you in your estimate? At what point would you say that there is enough evidence to reject the known information and support your alternative claim? How far from the known mean of four years can the sample mean be before we reject the idea that the average lifespan of a hummingbird is four years?

7.2 TESTING THE HYPOTHESES

Hypothesis testing is a statistical method that is used in making statistical decisions using experimental data. Hypothesis Testing is basically an assumption that we make about the population parameter.

Ex : you say avg student in class is 40 or a boy is taller than girls.

all those example we assume need some statistic way to prove those. we need some mathematical conclusion what ever we are assuming is true.

2. why do we use it ?

Hypothesis testing is an essential procedure in statistics. A hypothesis test evaluates two mutually exclusive statements about a population to determine which statement is best supported by the sample data. When we say that a finding is statistically significant, it's thanks to a hypothesis test.

3. what are basic of hypothesis ?

Normal Curve images with different mean and variance

The basic of hypothesis is normalisation and standard normalisation. all our hypothesis is revolve around basic of these 2 terms. let's see these.

Standardised Normal curve image and separation on data in percentage in each section.

You must be wondering what's difference between these two image, one might say i don't find, while other will see some flatter graph compare to steep. well buddy this is not what i want to represent , in 1st first you can see there are different normal curve all those normal curve can have different mean's and variances where as in 2nd image if you notice the graph is properly distributed and mean =0 and variance =1 always. concept of z-score comes in picture when we use standardised normal data.

Let's take an example to understand the concept of Hypothesis Testing. A person is on trial for a criminal offense and the judge needs to provide a verdict on his case. Now, there are four possible combinations in such a case:

First Case: The person is innocent and the judge identifies the person as innocent

Second Case: The person is innocent and the judge identifies the person as guilty

Third Case: The person is guilty and the judge identifies the person as innocent

Fourth Case: The person is guilty and the judge identifies the person as guilty

As you can clearly see, there can be two types of error in the judgment – Type 1 error, when the verdict is against the person while he was innocent and Type 2 error, when the verdict is in favor of Person while he was guilty

According to the Presumption of Innocence, the person is considered innocent until proven guilty. That means the judge must find the evidence which convinces him “beyond a reasonable doubt”. This phenomenon of “Beyond a reasonable doubt” can be understood as Probability (Judge Decided Guilty | Person is Innocent) should be small.

The basic concepts of Hypothesis Testing are actually quite analogous to this situation.

		In inferential statistics	
		Null Hypothesis (H_0)	Alternative Hypothesis
Truth about Population	Decision based on sample		
	Null Hypothesis (H_0)	No error ($1 - \alpha$)	Type 2 error
Alternative Hypothesis (H_1)	Type 1 error (α)	No error	

We consider the Null Hypothesis to be true until we find strong evidence against it. Then, we accept the Alternate Hypothesis. We also determine the Significance Level (α) which can be understood as the Probability of (Judge Decided Guilty | Person is Innocent) in the previous example. Thus, if α is smaller, it will require more evidence to reject the Null Hypothesis. Don't worry, we'll cover all of this using a case study later.

7.2.1 Components of a formal hypothesis test

The null hypothesis is a statement about the value of a population parameter, such as the population mean (μ) or the population proportion (p). It contains the condition of equality and is denoted as H_0 (H-naught).

$$H_0 : \mu = 157 \text{ or } H_0 : p = 0.37$$

The alternative hypothesis is the claim to be tested, the opposite of the null hypothesis. It contains the value of the parameter that we consider plausible and is denoted as H_1 .

$$H_1 : \mu > 157 \text{ or } H_1 : p \neq 0.37$$

The test statistic is a value computed from the sample data that is used in making a decision about the rejection of the null hypothesis. The test statistic converts the sample mean (\bar{x}) or sample proportion (\hat{p}) to a Z- or t-score under the assumption that the null hypothesis is true. It is used to decide whether the difference between the sample statistic and the hypothesized claim is significant.

The p-value is the area under the curve to the left or right of the test statistic. It is compared to the level of significance (α).

The critical value is the value that defines the rejection zone (the test statistic values that would lead to rejection of the null hypothesis). It is defined by the level of significance.

The level of significance (α) is the probability that the test statistic will fall into the critical region when the null hypothesis is true. This level is set by the researcher.

The conclusion is the final decision of the hypothesis test. The conclusion must always be clearly stated, communicating the decision based on the components of the test. It is important to realize that we never prove or accept the null hypothesis. We are merely saying that the sample evidence is not strong enough to warrant the rejection of the null hypothesis. The conclusion is made up of two parts:

1) Reject or fail to reject the null hypothesis, and 2) there is or is not enough evidence to support the alternative claim.

Option 1) Reject the null hypothesis (H_0). This means that you have enough statistical evidence to support the alternative claim (H_1).

Option 2) Fail to reject the null hypothesis (H_0). This means that you do NOT have enough evidence to support the alternative claim (H_1).

Another way to think about hypothesis testing is to compare it to the US justice system. A defendant is innocent until proven guilty (Null hypothesis—innocent). The prosecuting attorney tries to prove that the defendant is guilty (Alternative hypothesis—

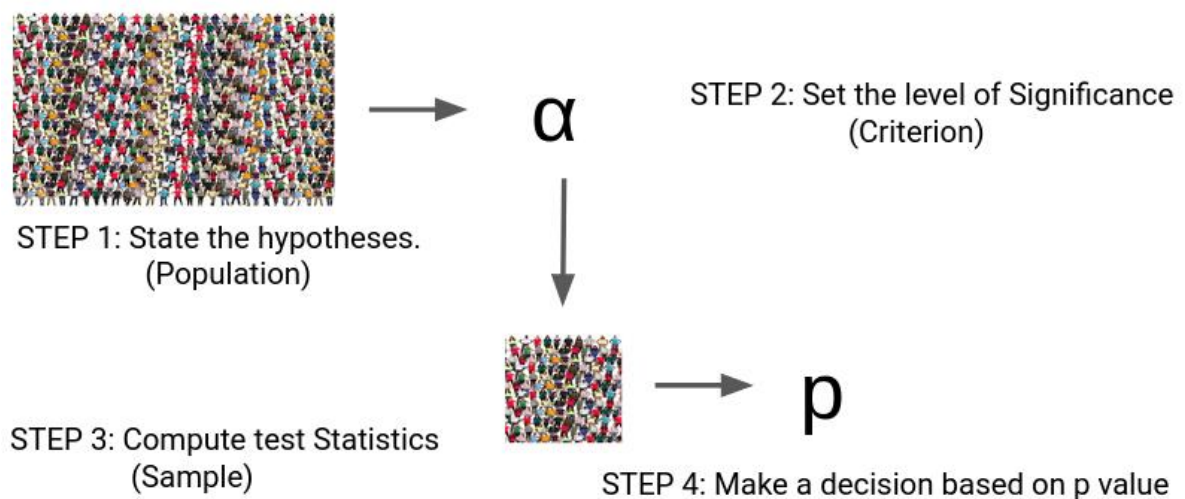
guilty). There are two possible conclusions that the jury can reach. First, the defendant is guilty (Reject the null hypothesis). Second, the defendant is not guilty (Fail to reject the null hypothesis). This is NOT the same thing as saying the defendant is innocent! In the first case, the prosecutor had enough evidence to reject the null hypothesis (innocent) and support the alternative claim (guilty). In the second case, the prosecutor did NOT have enough evidence to reject the null hypothesis (innocent) and support the alternative claim of guilty.

7.3 PROCEDURE OF HYPOTHESIS TESTING

Steps to Perform Hypothesis testing

There are four steps to perform Hypothesis Testing:

- Set the Hypothesis
- Set the Significance Level, Criteria for a decision
- Compute the test statistics
- Make a decision



Steps 1 to 3 are quite self-explanatory but on what basis can we make a decision in step 4? What does this p-value indicate?

We can understand this p-value as the measurement of the Defense Attorney's argument. If the p-value is less than α , we reject the Null Hypothesis or if the p-value is greater than α , we fail to reject the Null Hypothesis..

7.4 TOOLS FOR DATA WRANGLING:

There are some tools available for the data wrangling. Some of the popular tools are as follow:

- **Python**
Python is a most popular general-purpose high-level programming language. There are many popular Python libraries available for data science. The pandas is a most popular and open source library and it becomes a game changer for data science. It is a very fast, flexible, powerful and easy to used library which includes

Data Frame to perform more complex operation such as data joining, data merging, data transformation etc. for in data science.

- **R**
R is also popular, open source and more power tool for data science and management. It also supports many libraries such as dplyr, tidyr, ggplot2 etc. for data manipulation and data visualization.
- **Tabula**
Tabula is a tool for liberating data tables trapped inside the PDF files. It allows the user to upload the files in PDF format and extract the selected rows and columns from any tables available in the PDF file. It supports to extract this data from PDF to CSV or Microsoft Excel file format.
- **DataWrangler**
It is an interactive tool for data cleaning. It takes the read word data and transform it into data tables which can be used for further processing or analysis. It also supports to export the data tables in Microsoft Excel, R, Tabula etc.
- **OpenRefine**
OpenRefine, previously known as GoogleRefine. It is a Java based open source powerful tool for manipulates the huge data. It is used for data loading, understanding, cleaning and transforming from one format to another format. It also supports to extending the data with web services.
- **CSVKit**
CSVKit is a suite for command line tools for converting and working with CSV file. It supports the covert the data from Excel to CSV, JSON to CSV, query with SQL etc.

7.5 TEST STATISTICS AND CRITICAL REGIONS

A test statistic is a random variable that is calculated from sample data and used in a hypothesis test. You can use test statistics to determine whether to reject the null hypothesis. The test statistic compares your data with what is expected under the null hypothesis. The test statistic is used to calculate the p-value.

A test statistic measures the degree of agreement between a sample of data and the null hypothesis. Its observed value changes randomly from one random sample to a different sample. A test statistic contains information about the data that is relevant for deciding whether to reject the null hypothesis. The sampling distribution of the test statistic under the null hypothesis is called the null distribution. When the data show strong evidence against the assumptions in the null hypothesis, the magnitude of the test statistic becomes too large or too small depending on the alternative hypothesis. This causes the test's p-value to become small enough to reject the null hypothesis.

For example, the test statistic for a Z-test is the Z-statistic, which has the standard normal distribution under the null hypothesis. Suppose you perform a two-tailed Z-test with an α of 0.05, and obtain a Z-statistic (also called a Z-value) based on your data of 2.5. This Z-value corresponds to a p-value of 0.0124. Because this p-value is less than α , you declare statistical significance and reject the null hypothesis.

Different hypothesis tests use different test statistics based on the probability model assumed in the null hypothesis. Common tests and their test statistics include:

Hypothesis test	Test statistic
Z-test	Z-statistic
t-tests	t-statistic
ANOVA	F-statistic
Chi-square tests	Chi-square statistic

7.6 CRITICAL VALUE AND DECISION RULE

Critical Value, p-value

Let's understand the logic of Hypothesis Testing with the graphical representation for Normal Distribution.

Typically, we set the Significance level at 10%, 5%, or 1%. If our test score lies in the Acceptance Zone we fail to reject the Null Hypothesis. If our test score lies in the critical zone, we reject the Null Hypothesis and accept the Alternate Hypothesis. Critical Value is the cut off value between Acceptance Zone and Rejection Zone. We compare our test score to the critical value and if the test score is greater than the critical value, that means our test score lies in the Rejection Zone and we reject the Null Hypothesis. On the opposite side, if the test score is less than the Critical Value, that means the test score lies in the Acceptance Zone and we fail to reject the null Hypothesis.

But why do we need p-value when we can reject/accept hypotheses based on test scores and critical value?

p-value has the benefit that we only need one value to make a decision about the hypothesis. We don't need to compute two different values like critical value and test scores. Another benefit of using p-value is that we can test at any desired level of significance by comparing this directly with the significance level.

This way we don't need to compute test scores and critical value for each significance level. We can get the p-value and directly compare it with the significance level.

The Decision Rule: If the p-value is less than alpha, we reject the null hypothesis

Computing P-values

If it is a two-sided test (the alternative claim is \neq), the p-value is equal to two times the probability of the absolute value of the test statistic. If the test is a left-sided test (the

alternative claim is “<”), then the p-value is equal to the area to the left of the test statistic. If the test is a right-sided test (the alternative claim is “>”), then the p-value is equal to the area to the right of the test statistic.

Let’s look at Example

A forester studying diameter growth of red pine believes that the mean diameter growth will be different from the known mean growth of 1.35 in./year if a fertilization treatment is applied to the stand. He conducts his experiment, collects data from a sample of 32 plots, and gets a sample mean diameter growth of 1.6 in./year. The population standard deviation for this stand is known to be 0.46 in./year. Does he have enough evidence to support his claim?

Step 1) State the null and alternative hypotheses.

Ho: $\mu = 1.35$ in./year

H1: $\mu \neq 1.35$ in./year

Step 2) State the level of significance.

We will choose a level of significance of 5% ($\alpha = 0.05$).

Step 3) Compute the test statistic.

The p-value is two times the area of the absolute value of the test statistic (because the alternative claim is “not equal”).

Look up the area for the Z-score 3.07 in the standard normal table. The area (probability) is equal to $1 - 0.9989 = 0.0011$.

Multiply this by 2 to get the p-value = $2 * 0.0011 = 0.0022$.

Step 4) Compare the p-value to alpha and state a conclusion.

Use the Decision Rule (if the p-value is less than α , reject H0).

In this problem, the p-value (0.0022) is less than alpha (0.05).

We reject the H0. We have enough evidence to support the claim that the mean diameter growth is different from 1.35 inches/year.

Syntax:

```
pd.concat(objs, axis, join, join_axes, ignore_index, keys)
```

Here,

- **objs:** This is a sequence or mapping of Series, DataFrame, objects.
- **axis:** This is an axis to concatenate. This value is {0, 1, ...}, default is 0.
- **join:** This is used how to handle indexes on another axis(es). This value is {‘inner’, ‘outer’}, default is ‘outer’. The outer is used for union operation and inner is used for intersection operation.
- **join_axes:** This is the list of index objects. Its specific indexes to use for the other (n-1) axes instead of performing inner/outer set logic.

- **ignore_index:** This value is Boolean type, default is False. If this value is True, do not use the index values on the concatenation axis. The resulting axis will be labeled 0, ..., n-1.
- **keys:** This is a sequence to add an identifier to the result indexes, default is None.

Example: Here we perform the concatenation operation using two different data frames i.e. df1 and df2.

The below code creates the two different data frame df1 and df2.

```
# Importing pandas library
import pandas as pd

# First dataframe creation
df1 = pd.DataFrame({
    "Name":["Rahul", "Shreya", "Pankaj", "Monika", "Kalpesh"],
    "Age":[20, 19, 24, 25, 25],
    "Gender":["Male", "Female", "Male", "Male", "Female"],
    "Course":["B.E.", "B.Tech.", "MCA", "M.Tech.", "M.E."]},
    index = [1, 2, 3, 4, 5])

# Second dataframe creation
df2 = pd.DataFrame({
    "Name":["Mayank", "Jalpa", "Sanjana", "Vimal", "Raj"],
    "Age":[22, 21, 24, 26, 23],
    "Gender":["Male", "Female", "Female", "Male", "Male"],
    "Course":["MBA", "MCA", "B.E.", "B.Tech.", "M.Sc."]},
    index = [1, 2, 3, 4, 5])
```

Now we display both data frames df1 and df2.

The below code will display data frame df1.

```
# Display first dataframe
df1
```

The above code will give the following output.

	Name	Age	Gender	Course
1	Rahul	20	Male	B.E.
2	Shreya	19	Female	B.Tech.
3	Pankaj	24	Male	MCA
4	Monika	25	Male	M.Tech.
5	Kalpesh	25	Female	M.E.

The below code will display data frame df2.

```
# Display second dataframe
```

```
df2
```

The above code will give the following output.

	Name	Age	Gender	Course
1	Mayank	22	Male	MBA
2	Jalpa	21	Female	MCA
3	Sanjana	24	Female	B.E.
4	Vimal	26	Male	B.Tech.
5	Raj	23	Male	M.Sc.

Now we perform the concatenation operation on both data frames.

The below code will perform the concatenation operation on both the data frame df1 and df2.

```
# Concatenation of both dataframe
df3 = pd.concat([df1, df2])
df3
```

The above code will give the following output, which concatenate the five records of data frame df1 and five records of data frame df2 into single data frame df3 with ten records.

	Name	Age	Gender	Course
1	Rahul	20	Male	B.E.
2	Shreya	19	Female	B.Tech.
3	Pankaj	24	Male	MCA
4	Monika	25	Male	M.Tech.
5	Kalpesh	25	Female	M.E.
1	Mayank	22	Male	MBA
2	Jalpa	21	Female	MCA
3	Sanjana	24	Female	B.E.
4	Vimal	26	Male	B.Tech.
5	Raj	23	Male	M.Sc.

Now we perform the concatenation with axis as an argument.

The below code will perform the concatenation operation on both data frame df1 and df2 with using axis as an argument.

```
# Concatenation of both dataframe with axis as an argument
df3 = pd.concat([df1, df2],axis=1)
df3
```

The above code will give the following output, which concatenate the data frame df1 and data frame df2 into single data frame df3 horizontally with **axis=1** as an argument.

	Name	Age	Gender	Course	Name	Age	Gender	Course
1	Rahul	20	Male	B.E.	Mayank	22	Male	MBA
2	Shreya	19	Female	B.Tech.	Jalpa	21	Female	MCA
3	Pankaj	24	Male	MCA	Sanjana	24	Female	B.E.
4	Monika	25	Male	M.Tech.	Vimal	26	Male	B.Tech.
5	Kalpesh	25	Female	M.E.	Raj	23	Male	M.Sc.

Now we perform the concatenation with keys as an argument which is associated with specific keys.

The below code will perform the concatenation operation on both data frame df1 and df2 with keys as an argument.

```
# Concatenation of both dataframe with keys as an argument
df3 = pd.concat([df1, df2], keys=['x','y'])
df3
```

The above code will give the following output, which concatenate the data frame df1 and data frame df2 into single data frame df3 with *x* and *y* as *keys* argument.

		Name	Age	Gender	Course
x	1	Rahul	20	Male	B.E.
	2	Shreya	19	Female	B.Tech.
	3	Pankaj	24	Male	MCA
	4	Monika	25	Male	M.Tech.
	5	Kalpesh	25	Female	M.E.
y	1	Mayank	22	Male	MBA
	2	Jalpa	21	Female	MCA
	3	Sanjana	24	Female	B.E.
	4	Vimal	26	Male	B.Tech.
	5	Raj	23	Male	M.Sc.

Now we perform the concatenation with *keys* and *ignore_index* as an argument. It follows its own indexing if we set *ignore_index* is True.

The below code will perform the concatenation operation on both data frame df1 and df2 with *keys* and *ignore_index* as an argument.

```
# Concatenation of both dataframe using keys argument
df3 = pd.concat([df1, df2], keys=['x','y'], ignore_index=True)
df3
```

The above code will give the following output, which concatenate data frame df1 and data frame df2 into single data frame df3 using *x* and *y* as *keys* arguments and *ignore_index=True* argument.

	Name	Age	Gender	Course
0	Rahul	20	Male	B.E.
1	Shreya	19	Female	B.Tech.
2	Pankaj	24	Male	MCA
3	Monika	25	Male	M.Tech.
4	Kalpesh	25	Female	M.E.
5	Mayank	22	Male	MBA
6	Jalpa	21	Female	MCA
7	Sanjana	24	Female	B.E.
8	Vimal	26	Male	B.Tech.
9	Raj	23	Male	M.Sc.

Now we perform the concatenation using `append()` function.

The below code will perform the concatenation operation on both data frame df1 and df2. The data frame df2 is appended with the data frame df1.

```
# Concatenation of both dataframe using append
df1.append(df2)
```

The above code will give the following output, which concatenate the data frame df2 with the data frame df1.

	Name	Age	Gender	Course
1	Rahul	20	Male	B.E.
2	Shreya	19	Female	B.Tech.
3	Pankaj	24	Male	MCA
4	Monika	25	Male	M.Tech.
5	Kalpesh	25	Female	M.E.
1	Mayank	22	Male	MBA
2	Jalpa	21	Female	MCA
3	Sanjana	24	Female	B.E.
4	Vimal	26	Male	B.Tech.
5	Raj	23	Male	M.Sc.

7.7 MERGING DATASET

The word “*merge*” and “*join*” both are used relatively interchangeable in SQL, R and Pandas. Both merge and join doing the similar things, but there are separate “merge” and “join” functions in Pandas.

The merging/joining is the process of bringing two or more datasets together into single dataset and aligning the rows from each dataset based on the common attributes or columns.

The ‘*merge*’ function is used to perform the merging operation with the data frame. Here we merge the datasets using pandas.

Syntax:

```
pd.merge(left, right, how, on, left_on, right_on, left_index, right_index, sort)
```

Here,

- **left:** This is the first data frame.
- **right:** This is the second data frame.
- **how:** This is the method how to perform the merge operation. The values of this field are one of ‘*left*’, ‘*right*’, ‘*inner*’, ‘*outer*’, default is ‘*inner*’.
- **On:** This is the name of column in which action to be perform. This column must be available in both left and right data frame object.
- **left_on:** This is the name of column from left data frame to use as keys.
- **right_on:** This is the name of column from right data frame to use as keys.
- **left_index:** This is using the index (row label) from left data frame as its join keys, if it is True.
- **right_index:** This is used the index (row label) from right data frame as its join keys, if it is True.
- **sort:** This is use to sort the result data frame by join keys in specific order. The value of this field is Boolean, default is True.

Example: Here we perform the merge operation using two different data frames i.e. left and right.

The below code creates two different data frames left and right.

```
# Importing pandas library
import pandas as pd

# Left dataframe creation
left = pd.DataFrame({
    "Rno":[1, 2, 3, 4, 5],
    "Name":["Rahul", "Shreya", "Pankaj", "Monika", "Kalpesh"],
    "Course":["B.E.", "B.Tech.", "MCA", "M.Tech.", "M.E."])

# Right dataframe creation
right = pd.DataFrame({
    "Rno":[1, 2, 3, 4, 5],
```

```
"Name":["Mayank","Jalpa","Sanjana","Vimal","Raj"],  
"Course":["B.E.","MBA","MCA","B.Tech.","M.Sc."])
```

Now we display both data frame left and right.

The below code will display data frame left.

```
# Display left dataframe  
left
```

The above code will give the following output.

	Rno	Name	Course
0	1	Rahul	B.E.
1	2	Shreya	B.Tech.
2	3	Pankaj	MCA
3	4	Monika	M.Tech.
4	5	Kalpesh	M.E.

The below code will display the data frame right.

```
# Display right dataframe  
right
```

The above code will give the following output.

	Rno	Name	Course
0	1	Mayank	B.E.
1	2	Jalpa	MBA
2	3	Sanjana	MCA
3	4	Vimal	B.Tech.
4	5	Raj	M.Sc.

Now we perform the merge operation on both data frame using *on* as an argument.

The below code will perform the merge operation on both data frame left and right using single *on* key as an argument.

```
# Merge both left and right dataframe using single on key  
pd.merge(left, right, on='Rno')
```

The above code will give the following output, which merge both data frame left and right using single *on* key as an argument.

	Rno	Name_x	Course_x	Name_y	Course_y
0	1	Rahul	B.E.	Mayank	B.E.
1	2	Shreya	B.Tech.	Jalpa	MBA

2	3	Pankaj	MCA	Sanjana	MCA
3	4	Monika	M.Tech.	Vimal	B.Tech.
4	5	Kalpesh	M.E.	Raj	M.Sc.

The below code will perform the merge operation on both data frame left and right using multiple *on* key as an argument.

```
# Merge left and right dataframe using multiple on keys
pd.merge(left, right, on=['Rno','Course'])
```

The above code will give the following output, which merge both data frame left and right using multiple *on* key as an argument.

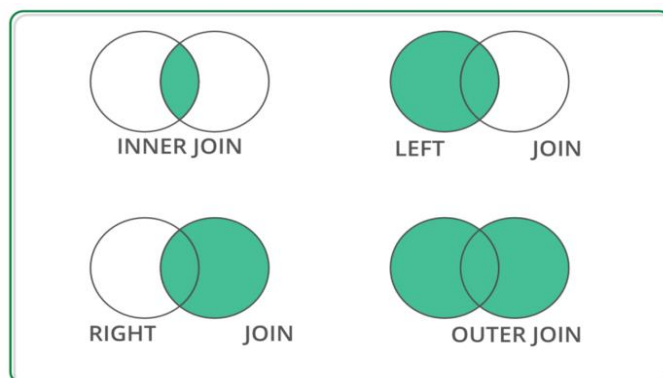
	Rno	Name_x	Course	Name_y
0	1	Rahul	B.E.	Mayank
1	3	Pankaj	MCA	Sanjana

Now we perform the merge operation using *how* as an argument. This argument specifies how to determine which keys are to be included in the resulting data frame or table. If the combination does not appear in any of the data frame or table, NA will be display in joined table.

The merge methods are same as SQL join equivalent as below:

Merge Method	SQL Join Equivalent	Description
left	Left Outer Join	Use keys from left object
right	Right Outer Join	Use keys from right object
inner	Inner Join	Use intersection of keys
outer	Full Outer Join	Use unions of keys

It will represent graphically as follows:



The below code will perform the merge operation on both data frame left and right *on* course using *how*='left' method.

```
# Merge both left and right dataframe using on and how
```



```
pd.merge(left, right, on='Course', how='left')
```

The above code will give the following output, which merge both data frame left and right using left join. It will display left data frame records plus common records as follows:

	Rno_x	Name_x	Course	Rno_y	Name_y
0	1	Rahul	B.E.	1.0	Mayank
1	2	Shreya	B.Tech.	4.0	Vimal
2	3	Pankaj	MCA	3.0	Sanjana
3	4	Monika	M.Tech.	NaN	NaN
4	5	Kalpesh	M.E.	NaN	NaN

The below code will perform the merge operation on both data frame left and right *on* course using *how='right'* method.

```
# Merge both left and right dataframe using on and how  
pd.merge(left, right, on='Course', how='right')
```

The above code will give the following output, which merge both data frame left and right using right join. It will display right data frame records plus common records as follows:

	Rno_x	Name_x	Course	Rno_y	Name_y
0	1.0	Rahul	B.E.	1	Mayank
1	NaN	NaN	MBA	2	Jalpa
2	3.0	Pankaj	MCA	3	Sanjana
3	2.0	Shreya	B.Tech.	4	Vimal
4	NaN	NaN	M.Sc.	5	Raj

The below code will perform the merge operation on both data frame left and right *on* course using *how='inner'* method.

```
# Merge both left and right dataframe using on and how  
pd.merge(left, right, on='Course', how='inner')
```

The above code will give the following output, which merge both data frame left and right using inner join. It performs the intersection operation on both data frame, which will display only common records as follows:

	Rno_x	Name_x	Course	Rno_y	Name_y
0	1	Rahul	B.E.	1	Mayank
1	2	Shreya	B.Tech.	4	Vimal
2	3	Pankaj	MCA	3	Sanjana

The below code will perform the merge operation on both data frame left and right *on* course using *how*=‘*outer*’ method.

```
# Merge both left and right dataframe using on and how
pd.merge(left, right, on='Course', how='outer')
```

The above code will give the following output, which merge both data frame left and right using outer join. It performs the union operation on both data frame, which will display left data frame records, right data frame records and common records as follows:

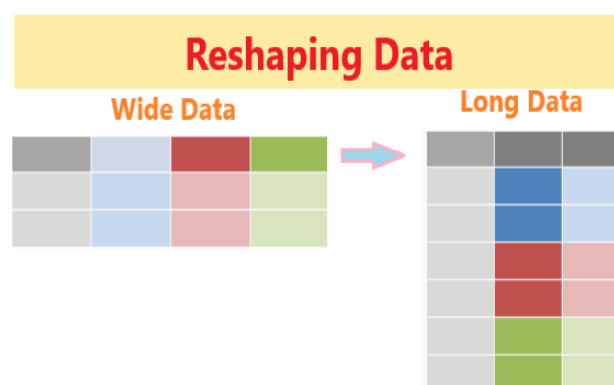
	Rno_x	Name_x	Course	Rno_y	Name_y
0	1.0	Rahul	B.E.	1.0	Mayank
1	2.0	Shreya	B.Tech.	4.0	Vimal
2	3.0	Pankaj	MCA	3.0	Sanjana
3	4.0	Monika	M.Tech.	NaN	NaN
4	5.0	Kalpesh	M.E.	NaN	NaN
5	NaN	NaN	MBA	2.0	Jalpa
6	NaN	NaN	M.Sc.	5.0	Raj

7.8 RESHAPING DATASET

The way in which dataset is arranged into rows and columns is referred as the shape of data. In a dataset, each row represents one observation in a vertical or long data and each column is considered a variable with multiple distinct values.

It is needed to convert or transform the dataset from one format to another format, which is called reshaping of dataset.

Reshaping is the process to change the shape or structure of datasets, such as convert “*wide*” data tables into “*long*”. The below figure shows the reshaping process graphically.



The data frame will be reshaped by using `melt()`, `stack()`, `unstack()` and `pivot()` functions as follows:

The below code creates and display data frame `df1`.

```
# Importing pandas library
```

```

import pandas as pd

# Dataframe creation
df = pd.DataFrame({
    "Rno":[1, 2, 3, 4, 5],
    "Name":["Rajan","Shital","Mayur","Mittal","Mahesh"],
    "Age":[25, 27, 24, 25, 21]})

# Display dataframe
df

```

The data frame output as follows:

	Rno	Name	Age
0	1	Rajan	25
1	2	Shital	27
2	3	Mayur	24
3	4	Mittal	25
4	5	Mahesh	21

7.8.1 Using melt() function:

We can reshape the data frame using this function. This function is used to wide data frame columns into rows.

The below code will perform the reshape operation using melt function.

```

# Performing melt function on dataframe
df.melt()

```

The above code will give the following output.

	variable	value
0	Rno	1
1	Rno	2
2	Rno	3
3	Rno	4
4	Rno	5
5	Name	Rajan
6	Name	Shital
7	Name	Mayur
8	Name	Mittal
9	Name	Mahesh
10	Age	25
11	Age	27

12	Age	24
13	Age	25
14	Age	21

7.8.2 Using stack() and unstack() function:

We can reshape the data frame using these functions. The stack() function is used to increase the level of index in a data frame.

The below code will perform the reshape operation using stack function.

```
# Performing stack function on dataframe
df.stack()
```

The above code will give the following output.

0	Rno	1
	Name	Rajan
	Age	25
1	Rno	2
	Name	Shital
	Age	27
2	Rno	3
	Name	Mayur
	Age	24
3	Rno	4
	Name	Mittal
	Age	25
4	Rno	5
	Name	Mahesh
	Age	21
dtype: object		

The unstack() function is used to do the revert back changes in a data frame was perform by the stack() function.

The below code will perform the reshape operation using unstack function.

```
# Performing unstack function on dataframe
df.unstack()
```

The above code will give the following output.

Rno	0	1
	1	2
	2	3

	3	4
	4	5
Name	0	Rajan
	1	Shital
	2	Mayur
	3	Mittal
	4	Mahesh
Age	0	25
	1	27
	2	24
	3	25
	4	21
dtype: object		

7.8.3 Using pivot() function:

We can reshape the data frame using this function. This function is used to reshape the data frame based on the specified column in a data frame.

The below code will perform the reshape operation on “**Rno**” column using pivot function.

```
# Performing pivot function on dataframe
df.pivot(columns='Rno')
```

The above code will give the following output.

	Name					Age				
Rno	1	2	3	4	5	1	2	3	4	5
0	Rajan	NaN	NaN	NaN	NaN	25.0	NaN	NaN	NaN	NaN
1	NaN	Shital	NaN	NaN	NaN	NaN	27.0	NaN	NaN	NaN
2	NaN	NaN	Mayur	NaN	NaN	NaN	NaN	24.0	NaN	NaN
3	NaN	NaN	NaN	Mittal	NaN	NaN	NaN	NaN	25.0	NaN
4	NaN	NaN	NaN	NaN	Mahesh	NaN	NaN	NaN	NaN	21.0

7.9 DATA TRANSFORMATION

Data is collected from the various sources and combine it into a unified data frame. This data frame has large number of columns with different data types.

Data transformation is the process to transform or convert the data as per required format for further processing as and when needed.

The data transformation includes add new columns, find NaN values, drop NaN, replace with mean value, field encoding and decoding, column splitting etc.

7.9.1 Data frame creation:

To perform the various transformation operation on data, first we have to create the data frame.

The below code will create and display a data frame df which contains the three columns such as “Name”, “Gender” and “City”.

```
#importing pandas library
import pandas as pd

# Dataframe creation
df = pd.DataFrame({
    "Name":["Jayesh Patel","Priya Shah","Vijay Sharma"],
    "Gender": ["Male","Female","Male"],
    "City": ["Rajkot","Delhi","Mumbai"]})

# Display dataframe
df
```

The above code will create and display data frame as follows:

	Name	Gender	City
0	Jayesh Patel	Male	Rajkot
1	Priya Shah	Female	Delhi
2	Vijay Sharma	Male	Mumbai

7.9.2 Missing Value:

The dataset contains the many rows and columns. There are some cells in a dataset that have NA or empty cell. This is called missing data in a dataset.

It is needed first to check that missing value before further processing. The common and very simple method to handle this missing value is to delete the rows which contain missing values.

The below code will check the data to containing the missing value or not.

```
df.isna().sum()
```

Here there are no any missing values in each column so it returns zero value in each column.

If there are any missing values in each column, it returns the number of missing values.

```
Name    0
Gender  0
City    0
dtype: int64
```

The below code will drop all the rows which containing missing value.

```
df = df.dropna()
```

The below code will drop the columns where all elements are missing values.

```
df.dropna(axis=1, how='all')
```

The below code will drop the columns where any of the elements containing missing values.

```
df.dropna(axis=1, how='any')
```

The below code will keep only the rows which contains maximum two missing values.

```
df.dropna(thresh=2)
```

The below code will fill all missing values with mean value of the particular column.

```
df.fillna(df.mean())
```

The below code will fill any missing values in specified column with median value of the particular column. Here we taken “Age” column for example.

```
df['Age'].fillna(df['Age'].median())
```

The below code will fill any missing values in specified column with mode value of the particular column. Here we taken “Age” column for example.

```
df['Age'].fillna(df['Age'].mode())
```

7.9.3 Encoding:

The dataset contains both numerical and categorical value. The categorical data is not much useful for data processing or analytics. It is needed to encoding the categorical value into numeric value.

Data encoding is the process to convert a categorical variable into a numerical form.

Here we discuss the label encoding which is simply converting each value in a column to a number. Our dataset has “**Gender**” column which has only two values “**male**” and “**female**” It encodes like this:

- Male → 0
- Female → 1

The below code will replace the “Male” to 0 and “Female” to 1.

```
df=df.Gender.replace({"Male":0,"Female":1})
df
```

The above code will display data frame as follows:

```
0  0
1  1
2  0
Name: Gender, dtype: int64
```

7.9.4 Inset New Column:

It is needed to add one or more columns in an existing dataset.

The below code will insert a new column in a data frame and display it.

```
df.insert(1, "Age", [21, 23, 24], True)
df
```

The above code will insert a new column “**Age**” with the values **21**, **23** and **24** respectively at second column in a data frame. The newly added column data frame will display as follow:

	Name	Age	Gender	City
0	Jayesh Patel	21	Male	Rajkot
1	Priya Shah	23	Female	Delhi
2	Vijay Sharma	24	Male	Mumbai

7.9.5 Split Column:

It is needed to split one column into two or more different columns. The process to create two or more different columns from single column in a dataset is called column splitting. Sometimes the “**Full Name**” column of dataset may be need to split into “**First Name**” and “**Last Name**” as a separate column.

The below code will split the column into two different columns and display new data frame.

```
df[['First Name','Last Name']] = df.Name.str.split(expand=True)
df
```

The above code will create two different columns (i.e. First Name, Last Name) from the single column “Name” of dataset. The newly split data frame will be display as follow:

	Name	Age	Gender	City	First Name	Last Name
0	Jayesh Patel	21	Male	Rajkot	Jayesh	Patel
1	Priya Shah	23	Female	Delhi	Priya	Shah
2	Vijay Sharma	24	Male	Mumbai	Vijay	Sharma

7.10 STRING MANIPULATION

String manipulation is the process of handling and analyzing the strings. The various operation can be performed on string such as string modification, parsing of string, string conversion etc. The various in-built functions available for string manipulation in different language. He we perform some common string manipulation operations in Python.

We take the following strings as an example.

```
str1 = "Data Science"
str2 = "using"
str3 = "Python"
str4 = "2021"
str5 = " Data Science "
```

Function	Description	Example	Output
capitalize()	It converts the first character of string into upper case	str2.capitalize()	'Using'
casefold()	It converts the string into lower case	str1.casefold ()	'data science'
center()	It returns the string in center of the specified size	str1.center (15)	' Data Science '
count()	It returns the number of times a specified value occurs in a string	str1.count("a")	2
endswith()	It returns true if the string ends with the specified value	str1.endswith ("nce")	True

find()	It searches the string for a specified value and returns the position of where it is found	str1.find("i")	7
format()	It formats the specified values in a string	str4.format()	'2021'
index()	It searches the string for a specified value and returns the position of where it was found	str1.index("c")	6
isalnum()	It returns True if all the characters in a string are alphanumeric	str4.isalnum()	True
isalpha()	It returns True if all characters in a string are alphabet	str2.isalpha()	True
isdigit()	It returns True if all characters in a string are digit	str4.isdigit()	True
islower()	It returns True if all characters in a string are lower case	str2.islower()	True
isnumeric()	It returns True if all characters in a string are numeric	str4.isnumeric() ()	True
isprintable()	It returns True if all characters in a string are printable	str1.isprintabl ()	True
isspace()	It returns True if all characters in a string are whitespaces	str1.isspace()	False
istitle()	It returns True if the string follows the title case rules	str1.istitle()	True
isupper()	It returns True if all characters in a string are upper case letter	str1.isupper()	False
len(string)	It returns the length of a string	len(str1)	12
lower()	It converts the string into lower case	str1.lower()	'data science'
lstrip()	It returns the string with left trim version	str5.lstrip()	'Data Science '
replace(old, new)	It replaces the old string with new string	str1.replace("Science", "Analytics")	'Data Analytics'
rfind()	It searches the string for a specified value and returns the last position of where it was found	str1.rfind("e")	11
rindex()	It searches the string for a specified value and returns the last index position of where it is	str1.rindex("a")	3

	found		
rstrip()	It returns the string with right trim version	str5.rstrip()	' Data Science'
split()	It splits the string with specified separator and returns a list	str1.split(" ")	['Data', 'Science']
startswith()	It returns true if the string starts with the specified value	str3.startswith("P")	True
strip()	It returns the both left and right trim version	str5.strip()	'Data Science'
swapcase()	It returns the swaps cases, lower case becomes upper case and vice versa	str1.swapcase()	'dATA sCIENCE'
title()	It converts the first character of each word to upper case	str2.title()	'Using'
upper()	It converts a string into upper case	str1.upper()	'DATA SCIENCE'
zfill()	It returns the string with filled by 0 for specified number of times in a string at the beginning	str3.zfill(10)	'0000Python'
+	It concatenates or join the two strings	str1+" "+str2+" "+str3	'Data Science using Python'
*	It repeated the string using n times	str3 * 3	'PythonPythonPython'
string[0]	It returns the first character of a string	str1[0]	'D'
string[7]	It returns the eighth character of a string	str1[7]	'i'
string[2:8]	It returns the string from third character to eighth character	str1[2:8]	'ta Sci'
string[3:]	It returns the string from fourth character to last character	str1[3:]	'a Science'
string[:8]	It returns the string from first character to eighth character	str1[:8]	'Data Sci'
string[-4:]	It returns the last four character of a sting	str1[-4:]	'ence'
string[:-4]	It removes the last four character of a string	str1[:-4]	'Data Sci'

7.11 REGULAR EXPRESSION

Regular expression or RegEx is generally used to identify whether a sequence or character or pattern is exists in a given string or not. It is also used to identify the position

of such pattern in a string or file. It mainly used to find and replace specific patterns in a string or file. It helps to manipulate text-based datasets.

Python has a built-in package called *re*, to work with Regular Expression.

The *re* package of Python has set of functions to search the string for matching. The common Python RegEx functions are as follow:

Function	Description
findall	It returns a list of all match values.
search	It returns a match object if match found anywhere in the string.
split	It returns a list and spit the string where each match found
sub	It replaces the one or more matches with a specified string.

7.11.1 The findall() function:

This function returns a list which contains all match values.

Example:

```
import re
string = "Working with Data Science using Python"
result = re.findall("th",string)
result
```

The list contains all match values in an order of they are found.

Output:

```
['th', 'th']
```

Example:

```
import re
string = "Working with Data Science using Python"
result = re.findall("ds",string)
result
```

If no matches found, an empty list will return.

Output:

```
[]
```

7.11.2 The search() function:

This function returns a match object if match found anywhere in the string. Only first occurrence of match will be returned, if there are more than one match found. The none will return if no match found.

Example:

```
import re
string = "Working with Data Science using Python"
result = re.search("wi",string)
result
```

It found the match value “**wi**”.

Output:

```
<re.Match object; span=(8, 10), match='wi'>
```

Example:

```
import re
string = "Working with Data Science using Python"
result = re.search("\s",string)
result
```

It found the match value “**\s**” i.e. space.

Output:

```
<re.Match object; span=(7, 8), match=' '>
```

7.11.3 The split() function:

This function returns a list of where the string has been split at each match found.

Example:

```
import re
string = "Working with Data Science using Python"
result = re.split("\s",string)
result
```

It found the match value “**\s**” i.e. space.

Output:

```
['Working', 'with', 'Data', 'Science', 'using', 'Python']
```

Example:

```
import re
string = "Working with Data Science using Python"
result = re.split("\s",string,2)
result
```

It found the match value “**\s**” i.e. space. But here the string will split into first 2 occurrence of found only. The remaining string will print as it is.

Output:

```
['Working', 'with', 'Data Science using Python']
```

7.11.4 The sub() function:

This function replaces the string with the specified text at each match found. It will print same string, if not match found.

Example:

```
import re
string = "Working with Data Science using Python"
result = re.sub("\s", "-",string)
result
```

It found the match value “\s” i.e. space. Every “\s” (space) will replace with the “-” (dash).

Output:

```
'Working-with-Data-Science-using-Python'
```

Example:

```
import re
string = "Working with Data Science using Python"
result = re.sub("\s", "-",string,2)
result
```

It found the match value “\s” i.e. space. But here “\s” (space) will replace with the “-” (dash) in first two occurrence of found only. The remaining string will print as it is.

Output:

```
'Working-with-Data Science using Python'
```

7.12 SUMMARY

The students will learn many things related to data pre-processing in this module and they will be able to perform the various data science related operation using Python.

- Ability to do the data wrangling which includes data discovery structuring, cleaning, enriching, validating and publishing.

- Ability to do the combining different datasets using concat, merge and join function with different arguments.
- Ability to do the reshaping of dataset using melt, stack and unstack and pivot functions.
- Ability to work with missing value, encoding categorical data into numerical value, splitting dataset etc.
- Ability to perform the various functions related to string and regular expression.

7.13 REFERENCES

Books

1. Davy Cielen, Arno D. B. Meysman, Mohamed Ali : Introducing Data Science, Manning Publications Co.
2. Stephen Klosterman (2019) : Data Science Projects with Python, Packt Publishing
3. Jake VanderPlas (2017) : Python Data Science Handbook: Essential Tools for Working with Data, O'Reilly
4. Wes McKinnery and Pandas Development Team (2021) : pandas : powerful Python data analysis toolkit, Release 1.2.3

Web References

1. <https://www.geeksforgeeks.org>
2. <https://www.tutorialspoint.com>
3. <https://www.w3schools.com>
4. <https://pandas.pydata.org>
5. <https://pbpython.com>

Practice Exercise

Short Answer:

1. What is data wrangling?
2. What is data cleaning?
3. List tools for data wrangling.
4. What is merging dataset?
5. What is reshaping dataset?

Long Answer:

1. Explain steps for data wrangling process.
2. Explain concat function with example.
3. Explain merge operation with syntax and example.
4. Explain reshaping dataset different functions.
5. Explain string function with example.
6. Explain regular expression with example.

PRACTICALS

1. Create and display a data frame.
2. Create a data frame and find null values and remove it.
3. Create a data frame and insert new column in data frame.
4. Create a data frame and convert categorical data into numerical values.
5. Create a data frame and split the column.
6. Create data frames and combine using concat function.
7. Create data frames and merge with different arguments.
8. Create data frame and reshape using melt function.
9. Perform string manipulation operations on string.
10. Perform regular expression functions on string.



The Motto of Our University
(SEWA)

SKILL ENHANCEMENT

EMPLOYABILITY

WISDOM

ACCESSIBILITY

SELF-INSTRUCTIONAL STUDY MATERIAL FOR JGND PSOU

ALL COPYRIGHTS WITH JGND PSOU, PATIALA

JAGAT GURU NANAK DEV
PUNJAB STATE OPEN UNIVERSITY, PATIALA

(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

B.Sc.(Data Science)

Semester III

BSDB32304T

Mathematical Foundation for Data Science

Head Quarter: C/28, The Lower Mall, Patiala-147001

Website: www.psou.ac.in

The Study Material has been prepared exclusively under the guidance of Jagat Guru Nanak Dev Punjab State Open University, Patiala, as per the syllabi prepared by Committee of Experts and approved by the Academic Council.

The University reserves all the copyrights of the study material. No part of this publication may be reproduced or transmitted in any form.

COURSE COORDINATOR AND EDITOR:

Dr. Amitoj Singh

Associate Professor

School of Sciences and Emerging Technologies

Jagat Guru Nanak Dev Punjab State Open University

LIST OF CONSULTANTS/ CONTRIBUTORS

Sr. No.	Name
1	Dr. A.B. Chandermohan
2	Dr. Reetu Malhotra



JAGAT GURU NANAK DEV PUNJAB STATE OPEN UNIVERSITY, PATIALA
(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

PREFACE

Jagat Guru Nanak Dev Punjab State Open University, Patiala was established in December 2019 by Act 19 of the Legislature of State of Punjab. It is the first and only Open University of the State, entrusted with the responsibility of making higher education accessible to all, especially to those sections of society who do not have the means, time or opportunity to pursue regular education.

In keeping with the nature of an Open University, this University provides a flexible education system to suit every need. The time given to complete a programme is double the duration of a regular mode programme. Well-designed study material has been prepared in consultation with experts in their respective fields.

The University offers programmes which have been designed to provide relevant, skill-based and employability-enhancing education. The study material provided in this booklet is self-instructional, with self-assessment exercises, and recommendations for further readings. The syllabus has been divided in sections, and provided as units for simplification.

The University has a network of 10 Learner Support Centres/Study Centres, to enable students to make use of reading facilities, and for curriculum-based counselling and practicals. We, at the University, welcome you to be a part of this institution of knowledge.

Prof. Anita Gill
Dean Academic Affairs



B.Sc. (Data Science)
Skill Enhancement Course (SEC)
Semester III

BSDB32304T: Mathematical Foundation for Data Science

Total Marks: 100

External Marks: 70

Internal Marks: 30

Credits: 4

Pass Percentage: 35%

Objective

This course helps student understand the basic concepts Mathematics that will be helpful to them in data science.

INSTRUCTIONS FOR THE PAPER SETTER/EXAMINER

1. The syllabus prescribed should be strictly adhered to.
2. The question paper will consist of three sections: A, B, and C. Sections A and B will have four questions each from the respective sections of the syllabus and will carry 10 marks each. The candidates will attempt two questions from each section.
3. Section C will have fifteen short answer questions covering the entire syllabus. Each question will carry 3 marks. Candidates will attempt any 10 questions from this section.
4. The examiner shall give a clear instruction to the candidates to attempt questions only at one place and only once. Second or subsequent attempts, unless the earlier ones have been crossed out, shall not be evaluated.
5. The duration of each paper will be three hours.

INSTRUCTIONS FOR THE CANDIDATES

Candidates are required to attempt any two questions each from the sections A, and B of the question paper, and any ten short answer questions from Section C. They have to attempt questions only at one place and only once. Second or subsequent attempts, unless the earlier ones have been crossed out, shall not be evaluated.

Section A

Unit I: Basics of Data Science- Introduction, Importance of data science, statistics and optimization from a data science perspective, structured thinking for solving data science problems using mathematics

Unit II: Linear Equations- Solution of Simultaneous Linear Equations (upto two variable case), Solution of Quadratic Equations. Series: Arithmetic Progression Series, Geometric Progression Series

Unit III: Permutations and Combinations, Binomial Theorem, Determinants with simple applications for solution of linear simultaneous equations using Cramer's Rule.

Unit IV: Matrices: with simple application for solution of linear simultaneous equations using matrix inversion method, Eigenvalues and eigenvectors; Matrix factorizations.

Section B

Unit V: Functions: Graphical representations of functions, limits and continuity of functions, first principle of differential calculus, derivations of simple algebraic functions

Unit VI: Applications: Applications of derivatives in Economic and Commerce. Maximum and minimum.

Unit VII: Linear Programming: General form, formulating Linear Programming Problems assumptions, Graphic Method.

Unit VIII: Exploring Linear Programming Problems: The Standard Maximum and Minimum Problems, Simplex Method, Duality, Dual Linear Programming Problems

Suggestive Readings

1. G. Strang, Introduction to Linear Algebra, Wellesley-Cambridge Press, Fifth edition, USA, 2016
2. Bendat, J. S. and A. G. Piersol, Random Data: Analysis and Measurement Procedures. 4th Edition. John Wiley & Sons, Inc., NY, USA, 2010.
3. Cathy O'Neil and Rachel Schutt, Doing Data Science, O'Reilly Media, 2013
4. Hadrien Jean , Essential Math for Data Science, O'Reilly, 2020



JAGAT GURU NANAK DEV PUNJAB STATE OPEN UNIVERSITY, PATIALA
(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

BSDB32304T: MATHEMATICAL FOUNDATION FOR DATA SCIENCE
COURSE COORDINATOR AND EDITOR: DR. AMITOJ SINGH

UNIT NO.	UNIT NAME
UNIT 1	BASICS OF DATA SCIENCE
UNIT 2	LINEAR EQUATIONS
UNIT 3	PERMUTATIONS AND COMBINATIONS
UNIT 4	MATRICES
UNIT 5	FUNCTIONS
UNIT 6	APPLICATIONS
UNIT 7	LINEAR PROGRAMMING
UNIT 8	EXPLORING LINEAR PROGRAMMING PROBLEMS

B.Sc.(DATA SCIENCE)

SEMESTER-III

MATHEMATICAL FOUNDATION FOR DATA SCIENCE

UNIT I: BASICS OF DATA SCIENCE

STRUCTURE

- 1.0 Objective**
- 1.1 Introduction**
- 1.2 Importance Of Data Science**
- 1.3 Optimization From A Data Science Perspective**
- 1.4 Solving Data Science Problems Using Mathematics**
- 1.5 Summary**
- 1.6 Useful Links**

1.0 OBJECTIVE

- Understand Importance of mathematics in data science.
- Optimization from a data science perspective, structured thinking for solving data science problems using mathematics

1.1 INTRODUCTION

In order to start the data science, first, we will learn the definitions of data science. Here, we listed few definitions of data science:

- **Wikipedia mentioned that** Data science is an interdisciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from noisy, structured and unstructured data, and apply knowledge and actionable insights from data across a broad range of application domains.
- **Oracle.com** pointed that Data science encompasses preparing data for analysis, including cleansing, aggregating, and manipulating the data to perform advanced data analysis. Analytic applications and data scientists can then review the results to uncover patterns and enable business leaders to draw informed insights.
- **DataRobot.com** explained that Data science is the field of study that combines domain expertise, programming skills, and knowledge of mathematics and statistics to extract meaningful insights from data.

From the above definitions, we listed few important key points about the data science:

- Data science is an interdisciplinary field - it means the data science is a combination of computer science, statistics, business model. Data Science is applied to variety of engineering application such as Banking, Financial Services, and Insurance (BFSI), Telecom and IT, Retail and eCommerce, Healthcare and Life sciences, Manufacturing, Energy and Utilities, Media and Entertainment, Transportation and Logistics, Government and many more applications.
- Data science is a systems to extract knowledge - from the huge data sets / big data, data scientist will extracts the knowledgeable information for improving our business / service.
- Insights from noisy, structured and unstructured data - the real time data available in the big data have noisy data, missing data, unwanted data and data which is not possible to frame in Relational Database model, it means

that it is impossible to represent in the tabular format.

- Data scientist requires many skills such as Programming Skills, Knowledge on Mathematics and Statistics, Domain Knowledge, and Business Model.



Fig 1.1 Data Science is a interdisciplinary Field (Courtesy: CleverTap)

The future of data science is expected to grow in the exponential manner. Because, most of the leading companies are focused on the data science. Major key players in the data science industry are shown below:

- Alphabet Inc. (Google) (US),
- Altair Engineering, Inc. (US),
- Alteryx, Inc. (US),
- Anaconda, Inc. (US),
- Civis Analytics (US),
- Cloudera, Inc. (US),
- Databricks (US).
- DataRobot, Inc. (US),
- Domino Data Lab, Inc. (US),
- H2O.ai. (US),
- IBM Corporation (US),
- MathWorks (Australia),
- Microsoft Corporation (US),
- Rapid Miner, Inc. (US),
- RStudio, Inc. (US),
- SAP (Germany),

- SAS Institute Inc. (US).

Simply says, the world has huge data and every business involves lots of data. The process required for improving the business is hidden in these data. Hence, extracting the hidden knowledge will improve the business. Therefore, the industry who wants to improve their business involves in data science.

1.2 IMPORTANCE OF DATA SCIENCE

In the Facebook account, you need to enter your hometown and your current location. In the users perspective, it is simply a data for finding your friends and well wishers in the easier way to find and connect with you. But it involves data analyzes, from these locations, the Facebook will identify global migration patterns, Social status based on Covid19 and where lot of fans available for T20 world-cup cricket. Based on these analyses, they will fine tune their business in the particular area. If Jaipur Town has higher cricket fans, they will spread more advertisement on cricket in Jaipur Town than other cities.

As a large retailer, Target tracks your purchases and interactions, both online and in-store. Based on these data, the company forms a predict model which will identify the sort of customers needs. For example, from the data, they will predict the customer who are pregnant, and then they will market baby-related products to them.

In 2012, the Obama campaign employed dozens of data scientists who data-mined and experimented their way to identifying voters who needed extra attention, choosing optimal donor-specific fund raising appeals and programs, and focusing get-out-the-vote efforts where they were most likely to be useful. It is generally agreed that these efforts played an important role in the president's re-election, which means it is a safe bet that political campaigns of the future will become more and more data-driven, resulting in a never- ending arms race of data science and data collection (Joel Grus, "Data Science from Scratch", O'Reilly, 2015). These kinds of data scientist are evolved in Indian Election for the past several years.

Therefore, we may conclude that the data science is useful in every part of our life, such as store, company, medical shop, retail shop, bank, insurance sector, and infinity.

1.3 OPTIMIZATION FROM A DATA SCIENCE PERSPECTIVE

In general, the data science on various business will represent the results of the analysis on any visualization techniques. Few graphical representation of data analyses are shown in figure 1.2.

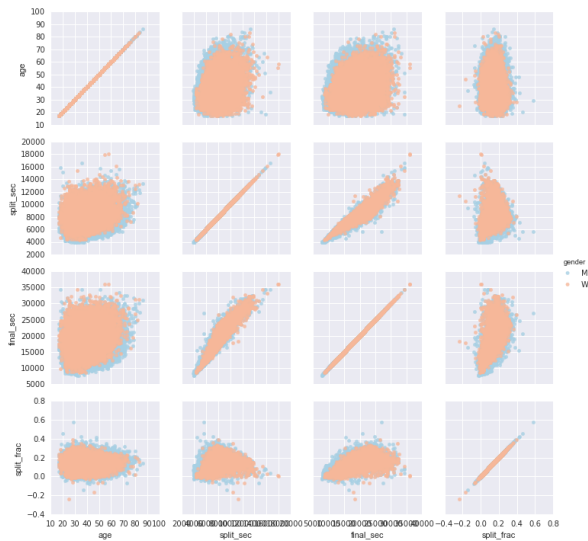


Figure 1.2 Results of Data science using variety of visualization methods

In hand writing key board available in the market requires to analyze the characters written by the customers. The main challenge in the task is the hand writing of the customers will vary. Figure 1.3 represents the classification of hand writing numbers. Now, it is understood, this application may extend to convert the non-readable doctors prescription into readable computer documents.

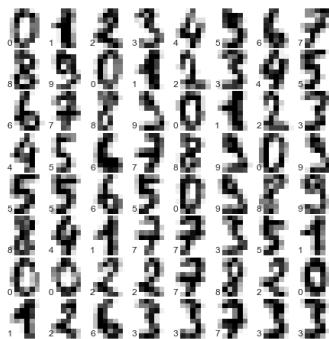


Figure 1.3 Classification of hand writing characters

In order to apply the data science in the practical applications shown in the above sections, this course framed to train the students to improve their mathematical knowledge required for data science and the subsequent coding skills using python. A Sample code for Simple Linear Equation with two variables is shown in figure 1.4. The mathematical way of solution for the given coding also explained in the same figure.

```
def solve_linear(equation,var='x'):
    expression = equation.replace("=","-("+"+")")
    grouped = eval(expression.replace(var,'1j'))
    return -grouped.real/grouped.imag
```

```
solve_linear("x - 2*x + 5*x - 46*(235-24) = x + 2")
```

```
3236.0
```

$$\begin{aligned}
 x - 2x + 5x - 46(235 - 24) &= x + 2 \\
 4x - 46(211) &= x + 2 \\
 4x - 9706 &= x + 2 \\
 3x &= 9708 \\
 x &= 3236
 \end{aligned}$$

Figure 1.4 Solving Linear Equation with two variables using python and its mathematical solution

1.4 SOLVING DATA SCIENCE PROBLEMS USING MATHEMATICS

The data science is part of Knowledge Discovery from Data base (KDD). The process of KDD is shown in the figure 1.5

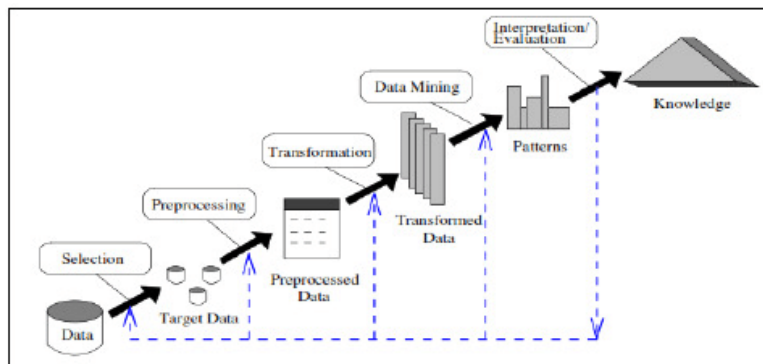


Figure 1.5 Process of Knowledge Extraction

The meaningful knowledge was extracted from the data after several stages, such as Pre-processing, Transformation, Data Mining and Visualization.

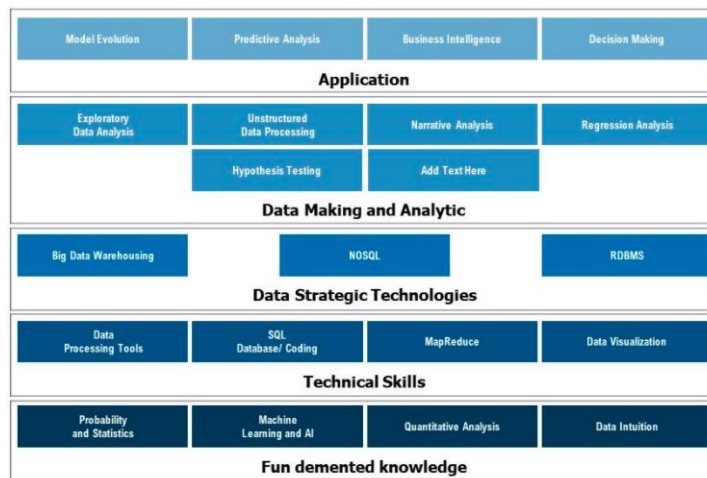


Figure 1.6 Skill Set required for a Data Scientist

The overall skill set required for the data scientist are shown in figure 1.6. In a large Multi National Company (MNC), there are several data scientist available in each parts of figure 1.6. Hence, the data scientist working in the application model carries designing business model (called Model Evolution), Predictive Analysis, Business Intelligence and Decision making using data mining techniques.

The second layer data scientist responsible for data making and analytic. They involved in the task of exploratory data analysis, unstructured data data processing, narrative analysis, regression analysis, text mining and hypothesis testing.

The third layer data scientist responsible for data strategic technologies which involved in the task of Big data representation and big data warehousing, No SQL and RDBMS.

The fourth layer data scientist should have technical skill required for analysis. This layer involves the data base administration such as data processing tools, SQL coding, Map Reduce and Data Visualization for unstructured data.

In the fifth layer, finding the result in terms of probability and statistics, fine tuning the result using Machine Learning, Predicting the business using AI technique, Quantitative analysis, and data intuition are carried out by the fun demanded knowledge resources. These extracts of these five layers of data science skills are represented in figure 1.7.

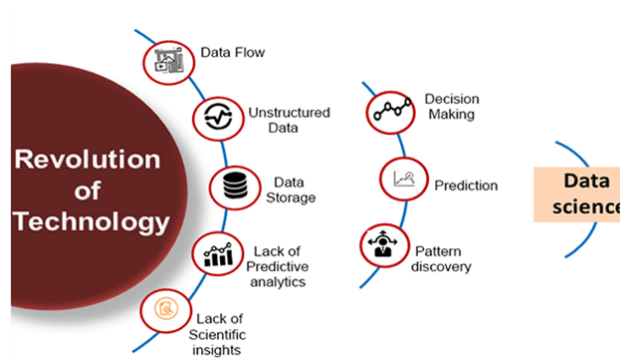


Figure 1.7 From Data to Data Science

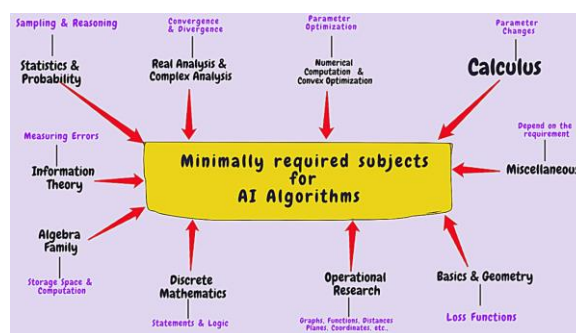


Fig 1.8 Mathematical Knowledge required for an advanced data scientist
(Courtesy:AnalyticsVidhya)

The mathematical knowledge required for an advanced data scientist are demonstrated in the figure 1.8. In this subject, the foundation for these mathematical model and subsequent coding skill are explored in the forthcoming units.

1.5 SUMMARY

Understanding how to construct linear equations is a fundamental component in developing central machine learning algorithms. These will be used to evaluate and observe data collections. Linear algebra is applied in machine learning algorithms in loss functions, regularisation, covariance matrices, Singular Value Decomposition (SVD), Matrix Operations, and support vector machine classification. It is also applied in machine learning algorithms like linear regression. These are the concepts that are needed for understanding the optimization methods used for machine learning

1.6 USEFUL LINKS

- <https://www.simplilearn.com/tutorials/data-science-tutorial/what-is-data-science>
- <https://www.coursera.org/specializations/mathematics-for-data-science>

B.Sc.(DATA SCIENCE)

SEMESTER-III

MATHEMATICAL FOUNDATION FOR DATA SCIENCE

UNIT II: LINEAR EQUATIONS

STRUCTURE

2.0 Objectives

2.1 Linear Equations

2.2 Solution of Simultaneous Linear Equations

2.3 Solution of Quadratic Equations

2.4 Sequences

2.5 Series

2.6 Arithmetic Progression Series

2.7 Geometric Progression Series

2.8 Summary

2.9 Practice Exercises

2.0 OBJECTIVES

- To understand and learn solution of Simultaneous Linear Equations (upto two variable case),
- To learn about the Quadratic Equations.
- Understand the concept of Series: Arithmetic Progression Series, Geometric Progression Series

2.1 LINEAR EQUATIONS

There are two kinds of mathematical model,

1) Expression

2) Equation.

The expression has mathematical model without equal sign and the equation has mathematical model with equal sign.

Some examples of expressions are:

- i. $20x$
- ii. $20x - 3y$
- iii. $30x + 30y$,
- iv. $x+y+z + x y + x y z$

Some examples of equations are:

- v. $5x = 20$
- vi. $20x - 3y = 30$
- vii. $30x + 30y = 60$
- viii. $x+y+z + x y + x y z = 60$

From the above example, it is identified that the mathematical equation uses the equality (=) sign; where as it is missing in the expressions. In the other terms, the equation has two parts, namely Left Hand Side (LHS) and Right Hand Side(RHS), and it assumes $LHS = RHS$.

The equation $5x = 20$ is a single variable equation, $20x - 3y = 30$ and $30x + 30y = 60$ are two variable equations, and $x+y+z + x y + x y z = 60$ is a three variable equation. Single variable equation are very easy to solve, for example, $5x = 20$, which may be converted as $x = 20 / 5$, hence x is 4.

If we have more than one variable, which is to be converted as single variable. In this section, we are going to solve one variable and two variable linear equation.

And the equation has a variable with the highest power of the variable is 1, then it is called as linear equation. For example, $2xy = 5$ is a linear equation, which means that the highest power of the variable appearing in the expression is 1. and the following equations are not linear equations, since highest power of variable > 1 :

- $x^2 + 1 = 10$
- $z + z^2 + z^3 = 100$

2.2 SOLUTION OF SIMULTANEOUS LINEAR EQUATIONS

Example 1: Find the solution of $2x - 3 = 7$

Solution:

Step 1: Add 3 to both sides.

$$2x - 3 + 3 = 7 + 3 \quad (\text{The balance is not disturbed})$$

or $2x = 10$

Step 2: Next divide both sides by 2.

$$2x/2 = 10/2$$

Now, the linear equation is solved, and the answer is $x = 5$.

Example 2: Solve $x/2 + 1/2 = 10$

Solution:

Step 1: Move the constant value to other side, $x/2 = 10 - 1/2$, $x/2 = 19/2$

Step 2: Multiply 2 on both sides $2 * (x/2) = 2 * (19/2)$, it means $x = 19$

The Answer of the given equation is 19.

Example 3: Solve the following equation and check your answer.

Solution:

$$4x - 7(2 - x) = 3x + 2$$

First, we need to clear out the parenthesis on the left side and then simplify the left side.

$$4x - 7(2 - x) = 3x + 2$$

$$4x - 14 + 7x = 3x + 2$$

$$11x - 14 = 3x + 2$$

Now we can subtract $3x$ on both side.

$$11x - 14 - 3x = 3x + 2 - 3x$$

$$8x - 14 = 2$$

Now, move 14 to other side,

$$8x = 2 + 14$$

$$8x = 16$$

Finally, divide by 8 on both sides to get the solution of $8x/8 = 16/8$, $x = 2$

Now, we will verify our solution:

Substitute the answer in the given equation.

$$4x - 7(2 - x) = 3x + 2$$

$$4(2) - 7(2 - 2) = 3(2) + 2$$

$$8 - 7(0) = 6 + 2$$

$$8 = 8$$

As the LHS = RHS, it is correct.

Example 4 : Solve the following equation : $(4 - 2z)/3 = 3/4 - 5z/6$

Solution:

The first step here is to multiply both sides by the LCD,

LCD of the 3,4,6 is 12.

$$12(4 - 2z)/3 = 12(3/4 - 5z/6)$$

$$12(4 - 2z)/3 = 12(3/4) - 12(5z/6)$$

$$4(4 - 2z) = 3(3) - 2(5z)$$

$$16 - 8z = 9 - 10z$$

Now move constant to one side and the variable part on another side.

$$\text{Hence, } 10z - 8z = 9 - 16$$

$$2z = -7$$

$$z = -7/2$$

Therefore, $z = -7/2$ or 3.5

2.3 SOLUTION OF QUADRATIC EQUATIONS

A quadratic equation in the variable x is of the form $ax^2 + bx + c = 0$, where a , b , c are real numbers and a not equal 0. The roots of a quadratic equation will be solved by the method of factorization.

If $b^2 - 4ac$ is not equal to 0, then the real roots of the quadratic equation $ax^2 + bx + c = 0$ are given by

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

The expression $b^2 - 4ac$ is called the discriminant of the quadratic equation. A quadratic equation $ax^2+bx+c=0$ has

two distinct real roots if $b^2 - 4ac > 0$

two equal real roots if $b^2 - 4ac = 0$

no real roots if $b^2 - 4ac < 0$.

- If $b^2 < 4ac$, then roots are complex (not real). For example roots of $x^2 + x + 1$, roots are $-0.5 + i1.73205$ and $-0.5 - i1.73205$.
- If $b^2 = 4ac$, then roots are real and both roots are same. For example, roots of $x^2 - 2x + 1$ are 1.
- If $b^2 > 4ac$, then roots are real and different. For example, roots of $x^2 - 7x - 12$ are 3 and 4.

Code1: Python Code for Quadratic equation:

```
import math

# Input Coordinates

a = int(input("Enter value of a"))
b = int(input("Enter value of b"))
c = int(input("Enter value of c"))

# If a is 0, then incorrect equation
if a == 0:

    print("Input correct quadratic equation")

else:

    # calculating discriminant using formula

    dis = b * b - 4 * a * c

    sqrt_val = math.sqrt(abs(dis))

    # checking condition for discriminant

    if dis > 0:

        print(" real and different roots ")

        print((-b + sqrt_val)/(2 * a))
```

```

        print((-b - sqrt_val)/(2 * a))
elif dis == 0:
    print(" real and same roots")
    print(-b / (2 * a))
# when discriminant is less than 0
else:
    print("Complex Roots")
    print(- b / (2 * a), " + i", sqrt_val)
    print(- b / (2 * a), " - i", sqrt_val)

```

Code2: Python Code for Quadratic equation using Function:

```

import math

# function for finding roots
def quaeqroots( a, b, c):
    # calculating discriminant using formula
    dis = b * b - 4 * a * c
    sqrt_val = math.sqrt(abs(dis))
    # checking condition for discriminant
    if dis > 0:
        print(" real and different roots ")
        print((-b + sqrt_val)/(2 * a))
        print((-b - sqrt_val)/(2 * a))
    elif dis == 0:
        print(" real and same roots")
        print(-b / (2 * a))

```

```

# when discriminant is less than 0

else:

    print("Complex Roots")

    print(- b / (2 * a), " + i", sqrt_val)

    print(- b / (2 * a), " - i", sqrt_val)

# Input Coordinates

a = int(input())

b = int(input())

c = int(input())

# If a is 0, then incorrect equation

if a == 0:

    print("Input correct quadratic equation")

else:

    quaeqnroots(a, b, c)

```

2.4 SEQUENCES

Sequence is a set of numbers which are written in some particular order. For example, take the numbers

$$1, 3, 5, 7, 9, \dots$$

Here, we seem to have a rule. We have a sequence of odd numbers. To put this another way, we start with the number 1, which is an odd number, and then each successive number is obtained by adding 2 to give the next odd number.

Here is another sequence:

$$1, 4, 9, 16, 25, \dots$$

This is the sequence of square numbers. And this sequence,

$$1, -1, 1, -1, 1, -1, \dots,$$

is a sequence of numbers alternating between 1 and 1. In each case, the dots written at the end indicate that we must consider the sequence as an infinite sequence, so that it

goes on for ever.

On the other hand, we can also have finite sequences. The numbers

$$1, 3, 5, 9$$

form a finite sequence containing just four numbers. The numbers

$$1, 4, 9, 16$$

also form a finite sequence. And so do these, the numbers

$$1, 2, 3, 4, 5, 6, \dots n .$$

These are the numbers we use for counting, and we have included n of them. Here, the dots indicate that we have not written all the numbers down explicitly. The n after the dots tells us that this is a finite sequence, and that the last number is n .

Here is a sequence that you might recognize:

$$1, 1, 2, 3, 5, 8, \dots$$

This is an infinite sequence where each term (from the third term onwards) is obtained by adding together the two previous terms. This is called the Fibonacci sequence.

We often use an algebraic notation for sequences. We might call the first term in a sequence u_1 , the second term u_2 , and so on. With this same notation, we would write u_n to represent the n -th term in the sequence. So

$$u_1, u_2, u_3 \dots, u_n$$

would represent a finite sequence containing n terms. As another example, we could use this notation to represent the rule for the Fibonacci sequence. We would write

$$u_n = u_{n-1} + u_{n-2}$$

to say that each term was the sum of the two preceding terms.

2.5 SERIES

A series is something we obtain from a sequence by adding all the terms together.

For example, suppose we have the sequence

$$u_1, u_2, u_3, \dots, u_n .$$

The series we obtain from this is

$$u_1 + u_2 + u_3 + \dots + u_n ,$$

and we write S_n for the sum of these n terms. So although the ideas of a ‘sequence’ and a ‘series’ are related, there is an important distinction between them.

For example, let us consider the sequence of numbers

$$1, 2, 3, 4, 5, 6, \dots, n .$$

Then $S_1 = 1$, as it is the sum of just the first term on its own. The sum of the first two terms is $S_2 = 1 + 2 = 3$. Continuing, we get

$$S_3 = 1 + 2 + 3 = 6 ,$$

$$S_4 = 1 + 2 + 3 + 4 = 10 ,$$

and so on.

2.6 ARITHMETIC PROGRESSION SERIES

Arithmetic Progression (AP) is a sequence of numbers, which follows some predefined order. The difference between two consecutive terms is always a constant in the AP. The difference of any two consecutive numbers in the AP is a constant value. For example, the series of natural numbers: 1, 2, 3, 4, 5, 6,... is an AP where the difference is 1, it means that the difference between any two consecutive number is $(n) - (n-1) = 1$.

Consider these two common sequences

$$1, 3, 5, 7, \dots$$

and

$$0, 10, 20, 30, 40$$

It is easy to see how these sequences are formed. They each start with a particular first term, and then to get successive terms we just add a fixed value to the previous term. In the first sequence we add 2 to get the next term, and in the second sequence we add 10. So the difference between consecutive terms in each sequence is a constant. We could also subtract a constant instead, because that is just the same as adding a negative constant. For example, in the sequence

$$8, 5, 2, -1, -4, \dots$$

the difference between consecutive terms is -3 . Any sequence with this property is called an arithmetic progression, or AP for short

We can use algebraic notation to represent an arithmetic progression. We shall let a stand for the first term of the sequence, and let d stand for the common difference between successive terms. For example, our first sequence could be written as

$$1, \quad 3, \quad 5, \quad 7, \quad 9, \quad \dots$$

$$1, \quad 1 + 2, \quad 1 + 2 \times 2, \quad 1 + 3 \times 2, \quad 1 + 4 \times 2, \quad \dots,$$

and this can be written as

$$a, \quad a + d, \quad a + 2d, \quad a + 3d, \quad a + 4d, \quad \dots$$

where $a = 1$ is the first term, and $d = 2$ is the common difference. If we wanted to write down the n -th term, we would have

$$a + (n - 1)d,$$

because if there are n terms in the sequence there must be $(n-1)$ common differences between successive terms, so that we must add on $(n-1)d$ to the starting value a . We also sometimes write ℓ for the last term of a finite sequence, and so in this case we would have

$$\ell = a + (n - 1)d.$$

In mathematics, there are three different types of progressions. They are:

- Arithmetic Progression (AP)
- Geometric Progression (GP)
- Harmonic Progression (HP)

A progression is a special type of sequence for which it is possible to obtain a formula for the n^{th} term. The Arithmetic Progression is the most commonly used sequence in maths with easy to understand formulas. Let's have a look at its three different types of definitions.

An arithmetic sequence or progression is defined as a sequence of numbers in which for every pair of consecutive terms, the second number is obtained by adding a fixed number to the first one.

The fixed number that must be added to any term of an AP to get the next term is known as the common difference of the AP. Now, let us consider the sequence, 1, 4, 7, 10, 13, 16, ... is considered as an arithmetic sequence with common difference 3.

If the AP is $a, a + d, a + 2d, a + 3d, \dots$,

- then the n^{th} term of AP is $X_n = a + (n - 1) \times d$
- Sum of n terms is $S = n/2[2a + (n - 1) \times d]$

- Sum of all terms in a finite AP with the last term as 'l' is $n/2(a + l)$

Example 1: Find the nth term of AP: 1, 2, 3, 4, 5....., X_n , if the number of terms are 15.

Solution:

Given data:

AP is 1, 2, 3, 4, 5....., X_n

$n=15$

First-term, $a = 1$

Common difference, $d=2-1 = 1$

$X_n = a + (n-1) d = 1+(15-1)1 = 1+14 = 15$

Example2: Find sum of natural numbers up to 15 numbers.

Solution:

Given data:

AP = 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15

$a = 1,$

$d = 2-1 = 1$

$X_n = 15$

Sum = $n/2 [2a + (n - 1) \times d]$

Sum = $15/2[2 \times 1+(15-1) \times 1]$

Sum = $15/2[2+14] = 15/2 [16] = 15 \times 8$

Sum = 120

Hence, the sum of the first 15 natural numbers is 120.

Example 3: Find the value of n. If $a = 10, d = 5, X_n = 95$.

Solution:

$X_n = a + (n - 1) \times d$

$$95 = 10 + (n - 1) \times 5$$

$$(n - 1) \times 5 = 95 - 10 = 85$$

$$(n - 1) = 85 / 5$$

$$(n - 1) = 17$$

$$n = 17 + 1$$

$$n = 18$$

Example 4: Find the 20th term in the given AP : 3, 5, 7, 9,

Solution:

Given data:

AP : 3, 5, 7, 9,

$$a = 3, d = 5 - 3 = 2, n = 20$$

$$X_n = a + (n - 1) \times d$$

$$X_{20} = 3 + (20 - 1) \times 2$$

$$X_{20} = 3 + 38$$

$$X_{20} = 41$$

2.7 GEOMETRIC PROGRESSION SERIES

A geometric sequence is a sequence such that any element after the first is obtained by multiplying the preceding element by a constant called the common ratio which is denoted by r .

The common ratio (r) is obtained by dividing any term by the preceding term, $r = a_2/a_1 = a_3/a_2 = a_4/a_3$

Where,

- r is the common ratio
- a_1 is the first term
- a_2 is the second term
- a_3 is the third term
- a_{n-1} is the term before the n^{th} term
- a_n is the n^{th} term

For example, the sequence 1, 3, 9, 27, 81 is a geometric sequence. Note that after the first term, the next term is obtained by multiplying the preceding element by 3.

To find the n th term of a geometric sequence we use the formula:

$$a_n = a_1 r^{n-1}$$

Where,

- r is the common ratio
- a_1 is the first term
- a_{n-1} is the term before the n^{th} term
- a_n is the n^{th} term

Finding the sum of terms in a geometric progression is easily obtained by applying the formulas:

$$S_n = \frac{a_1(1 - r^n)}{1 - r}, \quad r \neq 1$$

where

- S_n is the sum of GP with n terms
- a_1 is the first term
- r is the common ratio
- n is the number of terms

Example 1: Find the 8th term in the Geometric Progression 1, 3, 9, ...

Solution:

$$a_1 = 1$$

$$a_2 = 3$$

$$a_3 = 9$$

$$n = 8$$

$$r = a_2/a_1 = 3$$

$$a_8 = a_1 \times r^{(8-1)}$$

$$a_8 = 1 \times (3)^7$$

$$a_8 = 3^7$$

$$a_8 = 2187$$

Answer: 2187

Example 2: Find the sum of the first five terms of the geometric sequence (6,18,54,162,..)

Solution:

$$\text{Sum} = a \times (1 - r^n) / (1-r)$$

$$a = 6$$

$$r = 3$$

$$\text{So, Sum} = 6 \times (1 - 3^5) / (1-3)$$

$$\text{Sum} = 6 \times (-242) / (-2)$$

$$\text{Sum} = 6 \times 121$$

$$\text{Sum} = 726$$

2.8 SUMMARY

Linear equations are nothing but yet another subset of "equations". Any linear calculations requiring more than one variable can be done with the help of linear equations. The standard form of a linear equation in one variable is of the form $ax + b = 0$. Here, x is a variable, and a and b are constants. While the standard form of a linear equation in two variables is of the form $ax + by = c$. Here, x and y are variables, and a , b and c are constants.

A linear equation is an equation that is written for two different variables. This equation will be a linear combination of these two variables, and a constant can be present. Surprisingly, when any linear equation is plotted on a graph, it will necessarily produce a straight line - hence the name: Linear equations.

2.9 PRACTICE EXERCISES

1. A sequence is given by the formula $u_n = 3n + 5$, for $n = 1, 2, 3, \dots$. Write down the first five terms of this sequence.
2. A sequence is given by $u_n = 1/n^2$, for $n = 1, 2, 3, \dots$. Write down the first four terms of this sequence. What is the 10th term?
3. Write down the first five terms of the AP with first term 8 and common

difference 7.

4. Write down the first five terms of the AP with first term 2 and common difference -5 .
5. What is the common difference of the AP $11, -1, -13, -25, \dots$?
6. Find the 17th term of the arithmetic progression with first term 5 and common difference 2.
7. Write down the 10th and 19th terms of the APs
 - (i) $8, 11, 14, \dots$,
 - (ii) $8, 5, 2, \dots$

B.Sc.(DATA SCIENCE)

SEMESTER-III

MATHEMATICAL FOUNDATION FOR DATA SCIENCE

UNIT III: PERMUTATIONS AND COMBINATION

STRUCTURE

3.0 Objectives

3.1 Permutations and Combinations

3.1.1 Permutations

3.1.2 Combinations

3.2 Binomial Theorem

3.3 Solution of Linear Simultaneous Equations Using Cramer's Rule

3.3.1 Evaluating the Determinant of a Matrix

3.3.2 Properties of Determinants

3.3.2 Practical Applications

3.4 Solve a System of Two Equations in Two Variables

3.5 Practice Exercises

3.0 OBJECTIVES

- To understand the concept of Permutations and Combinations, Binomial Theorem.
- To understand the solution of linear simultaneous equations using Cramer's Rule.

3.1 PERMUTATIONS AND COMBINATIONS

Permutation and combination are the ways to represent a group of objects by selecting them in a set and forming subsets. It defines the various ways to arrange a certain group of data. When we select the data or objects from a certain group, it is said to be permutations, whereas the order in which they are represented is called combination. Both concepts are very important in Mathematics.

3.1.1 Permutations

Suppose you want to arrange your books on a shelf. If you have only one book, there is only one way of arranging it. Suppose you have two books, one of History and one of Geography.

You can arrange the Geography and History books in two ways. Geography book first and the History book next, GH or History book first and Geography book next; HG. In other words, there are two arrangements of the two books.

Now, suppose you want to add a Mathematics book also to the shelf. After arranging History and Geography books in one of the two ways, say GH, you can put Mathematics book in one of the following ways: MGH, GMH or GHM. Similarly, corresponding to HG, you have three other ways of arranging the books. So, by the Counting Principle, you can arrange Mathematics, Geography and History books in $2 \times 3 = 6$ ways

By permutation we mean an arrangement of objects in a particular order. In the above example, we were discussing the number of permutations of one book or two books. In general, if you want to find the number of permutations of n objects $n \geq 1$, how can you do it? Let us see if we can find an answer to this. Similar to what we saw in the case of books, there is one permutation of 1 object, 2×1 permutations of two objects and $3 \times 2 \times 1$ permutations of 3 objects. It may be that, there are $n \times (n - 1) \times (n - 2) \times \dots \times 2 \times 1$ permutations of n objects. In fact, it is so, as you will see when we prove the following result.

P is a permutation or arrangement of r things from a set of n things without replacement, P is defined as (When repetition is not allowed):

$${}_n P_r = \frac{n!}{(n-r)!}$$

P is a permutation or arrangement of r things from a set of n things when repetition is allowed. P is defined as (When repetition is allowed):

Derivation of Permutation Formula:

Let us assume that there are r boxes and each of them can hold one thing. There will be as many permutations as there are ways of filling in r vacant boxes by n objects.

- No. of ways the first box can be filled: n
- No. of ways the second box can be filled: $(n - 1)$
- No. of ways the third box can be filled: $(n - 2)$
- No. of ways the fourth box can be filled: $(n - 3)$
- No. of ways r^{th} box can be filled: $[n - (r - 1)]$

The number of permutations of n different objects taken r at a time, where $0 < r \leq n$ and the objects do not repeat is: $n(n - 1)(n - 2)(n - 3) \dots (n - r + 1)$

$$\Rightarrow {}_n P_r = n(n - 1)(n - 2)(n - 3) \dots (n - r + 1)$$

Multiplying and dividing by $(n - r)(n - r - 1) \dots 3 \times 2 \times 1$,

$${}_n P_r = \frac{[n(n-1)(n-2)(n-3)\dots(n-r+1)(n-r)(n-r-1)\dots 3 \times 2 \times 1]}{(n-r)(n-r-1)\dots 3 \times 2 \times 1} = \frac{n!}{(n-r)!}$$

$${}_n P_r = \frac{n!}{(n-r)!}$$

Example:

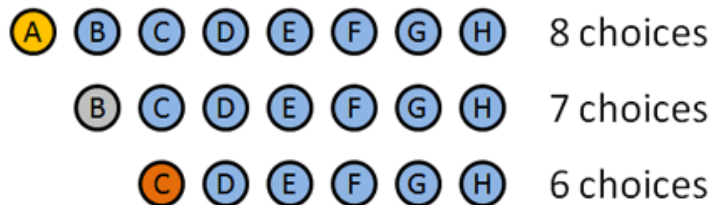
The following eight players playing a game:

- 1: Alice
- 2: Bob
- 3: Charlie
- 4: David
- 5: Eve

6: Frank
7: George
8: Horatio

How many possible ways to award 1st, 2nd and 3rd place (Gold / Silver / Bronze) prize among these eight contestants?

Solution:



Gold medal: 8 choices: A B C D E F G H . Let's say A wins the Gold.

Silver medal: 7 choices: B C D E F G H. Let's say B wins the silver.

Bronze medal: 6 choices: C D E F G H. Let's say... C wins the bronze.

The above consideration such as A wins Gold, B wins the silver and C wins the bronze are just an assumption. The assumption will not affect the computation.

Hence, no of ways are $8 \times 7 \times 6 = 336$ ways.

The same may be represented as $8! / 5! = 8 \times 7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1 / 5 \times 4 \times 3 \times 2 \times 1$, where 8 is the number of contestants and 5 is the (number of contestants - number of prize, $8 - 3$).

3.1.2 Combinations

Let us consider the example of shirts and trousers as stated in the introduction. There you have 4 sets of shirts and trousers and you want to take 2 sets with you while going on a trip. In how many ways can you do it?

Let us denote the sets by S1 , S2 , S3 , S4 . Then you can choose two pairs in the following ways :

1. {S1 , S2}
2. {S1 , S3}
3. {S1 , S4}
4. {S2 , S3}
5. {S2 ,S4}
6. {S3 , S4}

[Observe that {S1 , S2} is the same as {S2 ,S1}]. So, there are 6 ways of choosing the two sets that you want to take with you. Of course, if you had 10 pairs and you wanted to take 7 pairs, it will be much more difficult to work out the number of pairs in this way.

Now as you may want to know the number of ways of wearing 2 out of 4 sets for two days, say Monday and Tuesday, and the order of wearing is also important to you. We know that it can be done in ${}^4P_2 = 12$ ways. But note that each choice of 2 sets gives us two ways of wearing 2 sets out of 4 sets as shown below :

1. {S1 ,S2 } → S1 on Monday and S2 on Tuesday or S2 on Monday and S1 on Tuesday
2. {S1 ,S3 } → S1 on Monday and S3 on Tuesday or S3 on Monday and S1 on Tuesday
3. {S1 ,S4 } → S1 on Monday and S4 on Tuesday or S4 on Monday and S1 on Tuesday
4. {S2 ,S3 } → S2 on Monday and S3 on Tuesday or S3 on Monday and S2 on Tuesday
5. {S2 ,S4 } → S2 on Monday and S4 on Tuesday or S4 on Monday and S2 on Tuesday
6. {S3 ,S4 } → S3 on Monday and S4 on Tuesday or S4 on Monday and S3 on Tuesday

Thus, there are 12 ways of wearing 2 out of 4 pairs.

C is a combination of n distinct things taking r at a time, C is defined as (When repetition not allowed) as:

$${}^nC_r = \frac{{}^nP_r}{r!} = \frac{n!}{(n-r)!r!}$$

C is a combination of n distinct things taking r at a time with repetition, C is defined as (When repetition is allowed):

$${}^nC_r = \frac{(n + r - 1)!}{r!(n - 1)!}$$

Derivation of Combination Formula:

Let us assume that there are r boxes and each of them can hold one thing.

- No. of ways to select the first object from n distinct objects: **n**
- No. of ways to select the second object from (n-1) distinct objects: **(n-1)**
- No. of ways to select the third object from (n-2) distinct objects: **(n-2)**
- No. of ways to select rth object from [n-(r-1)] distinct objects: **[n-(r-1)]**

Completing the selection of r things from the original set of n things creates an

ordered subset of r elements.

\therefore The number of ways to make a selection of r elements of the original set of n elements is: n

$$(n - 1) (n - 2) (n - 3) \dots (n - (r - 1)) \text{ or } n (n - 1) (n - 2) \dots (n - r + 1).$$

Let us consider the ordered subset of r elements and all its permutations. The total number of all permutations of this subset is equal to $r!$ because r objects in every combination can be rearranged in $r!$ ways.

Hence, the total number of permutations of n different things taken r at a time is $(nC_r \times r!)$. It is nothing but nP_r .

$$nP_r = nC_r \times r!$$
$$nC_r = \frac{nP_r}{r!} = \frac{n!}{(n-r)!r!}$$

Example:

How many ways, three ice creams will be issued to eight people?

In this case, the order picking the people doesn't matter. If I give a can to Alice, Bob and then Charlie, it's the same as giving to Charlie, Alice and then Bob.

There are three choices for the first person, two choices for the second person, and only one choice for the third person, there are no choice for other five persons.

Therefore, $3 \times 2 \times 1$ choices = 6 choices.

Difference between Permutation and Combination:

Combination:

- Picking a team of 3 people from a group of 10 people. $C(10,3) = 10! / (7! \times 3!)$
 $= 10 \times 9 \times 8 / 3 \times 2 \times 1$
 $= 120$
- Similarity, Choosing 3 desserts from a menu of 10 desserts. $C(10,3) = 120$.

Permutation:

- Picking a President, VP and Waterboy from a group of 10 people. $P(10,3) = 10! / (10-3)!$

$$10! / 7! = 10 \times 9 \times 8 = 720$$

- Listing your 3 favorite desserts, in order, from a menu of 10. $P(10,3) = 720$.

3.2 BINOMIAL THEOREM

An expression consisting of two terms, connected by + or – sign is called a binomial expression. For example, $x + a$, $2x - 3y$, $1/x - 1/x^3$ are all binomial expressions.

If a and b are real numbers and n is a positive integer, then $(a + b)^n = {}^nC_0a^n b^0 + {}^nC_1a^{n-1}b^1 + {}^nC_2a^{n-2}b^2 + \dots + {}^nC_na^0b^n$

Some important observations:

1. The total number of terms in the binomial expansion of $(a + b)^n$ is $n + 1$, i.e. one more than the exponent n .
2. In the expansion, the first term is raised to the power of the binomial and in each subsequent terms the power of a reduces by one with simultaneous increase in the power of b by one, till power of b becomes equal to the power of binomial, i.e., the power of a is n in the first term, $(n - 1)$ in the second term and so on ending with zero in the last term. At the same time power of b is 0 in the first term, 1 in the second term and 2 in the third term and so on, ending with n in the last term.
3. In any term the sum of the indices (exponents) of ‘ a ’ and ‘ b ’ is equal to n (i.e., the power of the binomial).
4. The coefficients in the expansion follow a certain pattern known as pascal’s triangle, refer below table and the pascal’s triangle.

Index of Binomial	Coefficient of various terms													
0	1													
1	1		1											
2	1			2		1								
3	1				3			1						
4	1					4		6	4	1				
5	1						5			10		10	5	1

Pascal's Triangle

$$\begin{array}{rcccccc}
 n=0 & & \binom{0}{0} & & & & 1 \\
 n=1 & & \binom{1}{0} & \binom{1}{1} & & & 1 \quad 1 \\
 n=2 & & \binom{2}{0} & \binom{2}{1} & \binom{2}{2} & & 1 \quad 2 \quad 1 \\
 n=3 & & \binom{3}{0} & \binom{3}{1} & \binom{3}{2} & \binom{3}{3} & 1 \quad 3 \quad 3 \quad 1 \\
 n=4 & & \binom{4}{0} & \binom{4}{1} & \binom{4}{2} & \binom{4}{3} & \binom{4}{4} & 1 \quad 4 \quad 6 \quad 4 \quad 1
 \end{array}$$

Example 1: Simplify the following expressions:

$$\frac{(a+1)^4 + (a-1)^4}{(a+1)^4 - (a-1)^4}$$

Solution:

$$\begin{aligned}
 \frac{(a+1)^4 + (a-1)^4}{(a+1)^4 - (a-1)^4} &= \frac{a^4 + 4a^3 + 6a^2 + 4a + 1 + a^4 - 4a^3 + 6a^2 - 4a + 1}{a^4 + 4a^3 + 6a^2 + 4a + 1 - a^4 + 4a^3 - 6a^2 + 4a - 1} \\
 &= \frac{2a^4 + 12a^2 + 2}{8a^3 + 8a} = \frac{2(a^4 + 6a^2 + 1)}{2(4a^3 + 4a)} = \frac{a^4 + 6a^2 + 1}{4a^3 + 4a}
 \end{aligned}$$

Example 2: Find out the fourth member of following formula after expansion:

$$\left(x + \frac{2}{x}\right)^8$$

Solution:

$$A = x, B = \frac{2}{x}, k = 4, n = 8$$

$$M_k = \binom{n}{k-1} A^{n-k+1} B^{k-1}$$

$$M_4 = \binom{8}{3} x^5 \left(\frac{2}{x}\right)^3 = \frac{8!}{5!3!} x^5 \frac{8}{x^3} = \frac{8 \cdot 7 \cdot 6 \cdot 5!}{5! \cdot 6} x^5 \frac{8}{x^3} = 8 \cdot 8 \cdot 7 \cdot x^2 = 448x^2$$

$$M_4 = 448x^2$$

Example 3: Find out the member of the binomial expansion of $(x + x-1)^8$ not containing x .

Solution:

$$\begin{aligned} \binom{8}{k-1} x^{9-k} \cdot \left(\frac{1}{x}\right)^{k-1} &= 1 \\ \binom{8}{k-1} x^{9-k} \cdot \left(\frac{1}{x}\right)^{k-1} &= x^0 \Rightarrow \\ \Rightarrow x^{9-k} \cdot x^{1-k} &= x^0 \\ 9 - k + 1 - k &= 0 \\ 2k &= 10 \\ k &= 5 \end{aligned}$$

$$\begin{aligned} M_5 &= \binom{8}{4} x^4 \cdot x^{-4} \\ M_5 &= \binom{8}{4} \\ M_5 &= 70 \end{aligned}$$

The result is the number $M_5 = 70$.

Example 4: Which member of the binomial expansion of $(2x^3 + x^{-1})^{10}$ contains x^6 ?

Solution:

$$\begin{aligned} \binom{10}{k-1} (2x^3)^{11-k} \cdot (x^{-1})^{k-1} &= x^6 \Rightarrow \\ \Rightarrow x^{33-3k} \cdot x^{1-k} &= x^6 \\ 33 - 3k + 1 - k &= 6 \\ 4k &= 28 \\ k &= 7 \end{aligned}$$

$$\begin{aligned} M_7 &= \binom{10}{6} (2x^3)^4 \cdot (x^{-1})^6 = \binom{10}{6} 16 \cdot x^{12-6} = 16 \binom{10}{6} x^6 \\ M_7 &= 16 \binom{10}{6} x^6 \end{aligned}$$

Example 5: In the expansion of $(a + 2a^3)^n$ is the coefficient of the 3. expansion member greater by 44 than the 2. member's coefficient. Find out a positive integer meeting these conditions.

Solution:

$$\binom{n}{2} - \binom{n}{1} = 44$$

$$\frac{n!}{(n-2)! \cdot 2!} - n = 44$$

$$\frac{n(n-1)}{2} - n = 44 / .2$$

$$n^2 - n - 2n = 88$$

$$n^2 - 3n - 88 = 0$$

$$(n-11)(n+8) = 0$$

$$n_1 = 11$$

$$n_2 = -8 \in \emptyset$$

$$K = \{11\}$$

3.3 SOLUTION OF LINEAR SIMULTANEOUS EQUATIONS USING CRAMER'S RULE

From the previous sections, systems of equations in two variables was learned. Some of these methods are easier to apply to more appropriate in certain situations. In this section, two more strategies for solving systems of equations are focused.

3.3.1 Evaluating the Determinant of a 2×2 Matrix

A determinant is a real number that can be very useful in mathematics because it has multiple applications, such as calculating area, volume, and other quantities. Here, determinants are used to reveal whether a matrix is invertible by using the entries of a square matrix to determine whether there is a solution to the system of equations. Perhaps one of the more interesting applications, however, is their use in cryptography. Secure signals or messages are sometimes sent encoded in a matrix. The data can only be decrypted with an invertible matrix and the determinant.

The determinant of a 2×2 matrix, given

Matrix:	Determinant:
$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$	$\det(A) = A = \begin{vmatrix} a & b \\ c & d \end{vmatrix}$

is defined as

$$\det(A) = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - cb$$

For Example :

$$\begin{vmatrix} 2 & 4 \\ -5 & -1 \end{vmatrix} = 2(-1) - 4(-5) = -2 + 20 = 18$$

3.3.2 Properties of Determinants

- If the matrix is in upper triangular form, the determinant equals the product of entries down the main diagonal.
- When two rows are interchanged, the determinant changes sign.
- If either two rows or two columns are identical, the determinant equals zero.
- If a matrix contains either a row of zeros or a column of zeros, the determinant equals zero.
- The determinant of an inverse matrix A^{-1} is the reciprocal of the determinant of the matrix A .
- If any row or column is multiplied by a constant, the determinant is multiplied by the same factor.

3.3.2 Practical Applications

(i) The area of the triangle with vertices $(1, 0)$, $(6, 0)$, $(4, 3)$ is given by the relation,

$$\begin{aligned} \Delta &= \frac{1}{2} \begin{vmatrix} 1 & 0 & 1 \\ 6 & 0 & 1 \\ 4 & 3 & 1 \end{vmatrix} \\ &= \frac{1}{2} [1(0-3) - 0(6-4) + 1(18-0)] \\ &= \frac{1}{2} [-3 + 18] = \frac{15}{2} \text{ square units} \end{aligned}$$

(ii) The area of the triangle with vertices $(2, 7)$, $(1, 1)$, $(10, 8)$ is given by the relation,

$$\begin{aligned} \Delta &= \frac{1}{2} \begin{vmatrix} 2 & 7 & 1 \\ 1 & 1 & 1 \\ 10 & 8 & 1 \end{vmatrix} \\ &= \frac{1}{2} [2(1-8) - 7(1-10) + 1(8-10)] \\ &= \frac{1}{2} [2(-7) - 7(-9) + 1(-2)] \\ &= \frac{1}{2} [-14 + 63 - 2] = \frac{1}{2} [-16 + 63] \\ &= \frac{47}{2} \text{ square units} \end{aligned}$$

3.4 CRAMER'S RULE TO SOLVE A SYSTEM OF TWO EQUATIONS IN TWO VARIABLES

The second method is called Cramer's Rule, this technique is invented in the middle of the 18th century and is named for its innovator, the Swiss mathematician Gabriel Cramer (1704-1752), who introduced it in 1750. Cramer's Rule is a viable and efficient method for finding solutions to systems with an arbitrary number of unknowns, provided that we have the same number of equations as unknowns.

Cramer's Rule will give us the unique solution to a system of equations, if it exists. However, if the system has no solution or an infinite number of solutions, this will be indicated by a determinant of zero. To find out if the system is inconsistent or dependent, another method, such as elimination, will have to be used.

- Given a linear system

$$\begin{array}{ccc} \text{x-column} & & \text{constant column} \\ \downarrow & & \downarrow \\ a_1x + b_1y = c_1 \\ a_2x + b_2y = c_2 \\ \uparrow \\ \text{y-column} \end{array}$$

- Assign names for each matrix

coefficient matrix:

$$D = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix}$$

X – matrix:

$$D_x = \begin{bmatrix} c_1 & b_1 \\ c_2 & b_2 \end{bmatrix}$$

Y – matrix:

$$D_y = \begin{bmatrix} a_1 & c_1 \\ a_2 & c_2 \end{bmatrix}$$

From the above, x and y are:

$$x = \frac{D_x}{D} = \frac{\begin{vmatrix} c_1 & b_1 \\ c_2 & b_2 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}}$$

$$y = \frac{D_y}{D} = \frac{\begin{vmatrix} a_1 & c_1 \\ a_2 & c_2 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}}$$

Example 1: Solve the system with two variables by Cramer's Rule: $4x-3y=11$ and $6x+5y=7$

Solution:

Start by extracting the three relevant matrices: coefficient, x, and y. Then solve each corresponding determinant.

For coefficient matrix

$$D = \begin{bmatrix} 4 & -3 \\ 6 & 5 \end{bmatrix} \xrightarrow{\text{find its determinant}} |D| = \begin{vmatrix} 4 & -3 \\ 6 & 5 \end{vmatrix} = (4)(5) - (-3)(6)$$

$$= 20 - (-18)$$

$$= 20 + 18$$

$$= 38$$

For X – matrix:

$$D_x = \begin{bmatrix} 11 & -3 \\ 7 & 5 \end{bmatrix} \xrightarrow{\text{find its determinant}} |D_x| = \begin{vmatrix} 11 & -3 \\ 7 & 5 \end{vmatrix} = (11)(5) - (-3)(7)$$

$$= 55 - (-21)$$

$$= 55 + 21$$

$$= 76$$

For Y – matrix

$$D_y = \begin{bmatrix} 4 & 11 \\ 6 & 7 \end{bmatrix} \xrightarrow{\text{find its determinant}} |D_y| = \begin{vmatrix} 4 & 11 \\ 6 & 7 \end{vmatrix} = (4)(7) - (11)(6)$$

$$= 28 - (66)$$

$$= -38$$

Once all three determinants are calculated, it's time to solve for the values of x and y using the formula above.

Therefore, $x = Dx / D = 76 / 38 = 2$

And $y = Dy / D = -38/38 = 1$

3.5 PRACTICE EXERCISES

1. Evaluate (a) $3!$ (b) $2! + 4!$ (c) $2! \times 3!$
2. Suppose you want to arrange your English, Hindi, Mathematics, History, Geography and Science books on a shelf. In how many ways can you do it?
3. (a) 5 students are staying in a dormitory. In how many ways can you allot 5 beds to them?
(b) In how many ways can the letters of the word 'TRIANGLE' be arranged?
(c) How many four digit numbers can be formed with digits 1, 2, 3 and 4 and with distinct digits?
4. Find the number of subsets of the set $\{1,2,3,4,5,6,7,8,9,10,11\}$ having 4 elements.
5. 12 points lie on a circle. How many cyclic quadrilaterals can be drawn by using these points?

REFERENCE

- Bendat, J. S. and A. G. Piersol, Random Data: Analysis and Measurement Procedures. 4th Edition. John Wiley & Sons, Inc., NY, USA, 2010.
- Cathy O'Neil and Rachel Schutt, Doing Data Science, O'Reilly Media, 2013
- https://nios.ac.in/media/documents/SrSec311NEW/311_Maths_Eng/311_Maths_Eng_Lesson11.pdf

B.Sc.(DATA SCIENCE)

SEMESTER-III

MATHEMATICAL FOUNDATION FOR DATA SCIENCE

UNIT IV: MATRIX

STRUCTURE

4.0 Objectives

4.1 Solving Linear Equations Using Matrix Inversion Method

4.2 Eigenvalues and Eigenvectors

4.2.1 Applications of Eigenvalues and Eigenvectors

4.2.2 Calculating Eigenvectors and Eigenvalues

4.3 Matrix Factorization

4.3.1 LU Matrix Decomposition

4.3.2 QR Matrix Decomposition

4.4 Practice Exercise

4.0 OBJECTIVE

This unit will help in understanding Matrices with simple application for solution of linear simultaneous equations using matrix inversion method, Eigenvalues and eigenvectors and Matrix factorizations.

4.1 SOLVING LINEAR EQUATIONS USING MATRIX INVERSION METHOD

The power of matrix algebra is seen in the representation of a system of simultaneous linear equations as a matrix equation.

Matrix algebra allows us to write the solution of the system using the inverse matrix of the coefficients. In practice the method is suitable only for small systems. Its main use is the theoretical insight into such problems which it provides.

If we have one linear equation $ax = b$ in which the unknown is x , a and b are constants, where a not equal to 0 then $x = b/a = a^{-1} b$.

If we have more than one equation and more than one unknown, the algebraic solution $x = a^{-1} b$ used for a single equation to solve a system of linear equations.

Consider the following system:

$$2x_1 + 3x_2 = 5$$

$$x_1 - 2x_2 = -1.$$

This is to be converted into matrix form, which becomes

$$\begin{bmatrix} 2 & 3 \\ 1 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 5 \\ -1 \end{bmatrix}$$

which is of the form $AX = B$.

If A^{-1} exists then the solution is $X = A^{-1}B$.

Example 1:

Use the inverse matrix method to solve $2x_1 + 3x_2 = 3$ and $5x_1 + 4x_2 = 11$

Solution:

$$AX = B \rightarrow$$

$$\begin{bmatrix} 2 & 3 \\ 5 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 11 \end{bmatrix}$$

$$|A| = 2 \times 4 - 3 \times 5 = -7$$

$$A^{-1} = -\frac{1}{7} \begin{bmatrix} 4 & -3 \\ -5 & 2 \end{bmatrix}$$

Using $X = A^{-1}B$:

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = -\frac{1}{7} \begin{bmatrix} 4 & -3 \\ -5 & 2 \end{bmatrix} \begin{bmatrix} 3 \\ 11 \end{bmatrix}$$

$$= -\frac{1}{7} \begin{bmatrix} 4 \times 3 - 3 \times 11 \\ -5 \times 3 + 2 \times 11 \end{bmatrix}$$

$$= -\frac{1}{7} \begin{bmatrix} -21 \\ 7 \end{bmatrix} = \begin{bmatrix} 3 \\ -1 \end{bmatrix}$$

So, $x_1 = 3$ and $x_2 = -1$.

Example 2: Solve the following system of linear equations, using matrix inversion method: $5x + 2y = 3$, and $3x + 2y = 5$.

Solution:

The matrix form of the system is $AX = B$, where

$$A = \begin{bmatrix} 5 & 2 \\ 3 & 2 \end{bmatrix}, X = \begin{bmatrix} x \\ y \end{bmatrix}, B = \begin{bmatrix} 3 \\ 5 \end{bmatrix}.$$

We find $|A| = \begin{vmatrix} 5 & 2 \\ 3 & 2 \end{vmatrix} = 10 - 6 = 4 \neq 0$. So, A^{-1} exists and $A^{-1} = \frac{1}{4} \begin{bmatrix} 2 & -2 \\ -3 & 5 \end{bmatrix}$.

Then, applying the formula $X = A^{-1}B$, we get

$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 2 & -2 \\ -3 & 5 \end{bmatrix} \begin{bmatrix} 3 \\ 5 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 6-10 \\ -9+25 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} -4 \\ 16 \end{bmatrix} = \begin{bmatrix} \frac{-4}{4} \\ \frac{16}{4} \end{bmatrix} = \begin{bmatrix} -1 \\ 4 \end{bmatrix}.$$

So the solution is $(x = -1, y = 4)$.

Example 3:

Solve the following system of equations, using matrix inversion method:

$$2x_1 + 3x_2 + 3x_3 = 5,$$

$$x_1 - 2x_2 + x_3 = -4,$$

$$3x_1 - x_2 - 2x_3 = 3$$

Solution:

The matrix form of the system is $AX = B$, where

$$A = \begin{bmatrix} 2 & 3 & 3 \\ 1 & -2 & 1 \\ 3 & -1 & -2 \end{bmatrix}, X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, B = \begin{bmatrix} 5 \\ -4 \\ 3 \end{bmatrix}.$$

$$\text{We find } |A| = \begin{vmatrix} 2 & 3 & 3 \\ 1 & -2 & 1 \\ 3 & -1 & -2 \end{vmatrix} = 2(4+1) - 3(-2-3) + 3(-1+6) = 10+15+15 = 40 \neq 0.$$

So, A^{-1} exists and

$$A^{-1} = \frac{1}{|A|} (\text{adj } A) = \frac{1}{40} \begin{bmatrix} +(4+1) & -(-2-3) & +(-1+6) \\ -(-6+3) & +(-4-9) & -(-2-9) \\ +(3+6) & -(2-3) & +(-4-3) \end{bmatrix}^T = \frac{1}{40} \begin{bmatrix} 5 & 3 & 9 \\ 5 & -13 & 1 \\ 5 & 11 & -7 \end{bmatrix}$$

Then, applying $X = A^{-1}B$, we get

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \frac{1}{40} \begin{bmatrix} 5 & 3 & 9 \\ 5 & -13 & 1 \\ 5 & 11 & -7 \end{bmatrix} \begin{bmatrix} 5 \\ -4 \\ 3 \end{bmatrix} = \frac{1}{40} \begin{bmatrix} 25-12+27 \\ 25+52+3 \\ 25-44-21 \end{bmatrix} = \frac{1}{40} \begin{bmatrix} 40 \\ 80 \\ -40 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix}.$$

So, the solution is ($x_1 = 1, x_2 = 2, x_3 = -1$).

$$\text{We find } AB = \begin{bmatrix} -4 & 4 & 4 \\ -7 & 1 & 3 \\ 5 & -3 & -1 \end{bmatrix} \begin{bmatrix} 1 & -1 & 1 \\ 1 & -2 & -2 \\ 2 & 1 & 3 \end{bmatrix} = \begin{bmatrix} -4+4+8 & 4-8+4 & -4-8+12 \\ -7+1+6 & 7-2+3 & -7-2+9 \\ 5-3-2 & -5+6-1 & 5+6-3 \end{bmatrix} = \begin{bmatrix} 8 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & 8 \end{bmatrix} = 8I_3$$

$$\text{and } BA = \begin{bmatrix} 1 & -1 & 1 \\ 1 & -2 & -2 \\ 2 & 1 & 3 \end{bmatrix} \begin{bmatrix} -4 & 4 & 4 \\ -7 & 1 & 3 \\ 5 & -3 & -1 \end{bmatrix} = \begin{bmatrix} -4+7+5 & 4-1-3 & 4-3-1 \\ -4+14-10 & 4-2+6 & 4-6+2 \\ -8-7+15 & 8+1-9 & 8+3-3 \end{bmatrix} = \begin{bmatrix} 8 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & 8 \end{bmatrix} = 8I_3.$$

So, we get $AB = BA = 8I_3$. That is, $\left(\frac{1}{8}A\right)B = B\left(\frac{1}{8}A\right) = I_3$. Hence, $B^{-1} = \frac{1}{8}A$.

Writing the given system of equations in matrix form, we get

$$\begin{bmatrix} 1 & -1 & 1 \\ 1 & -2 & -2 \\ 2 & 1 & 3 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 4 \\ 9 \\ 1 \end{bmatrix}. \text{ That is, } B \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 4 \\ 9 \\ 1 \end{bmatrix}.$$

$$\text{So, } \begin{bmatrix} x \\ y \\ z \end{bmatrix} = B^{-1} \begin{bmatrix} 4 \\ 9 \\ 1 \end{bmatrix} = \left(\frac{1}{8}A\right) \begin{bmatrix} 4 \\ 9 \\ 1 \end{bmatrix} = \frac{1}{8} \begin{bmatrix} -4 & 4 & 4 \\ -7 & 1 & 3 \\ 5 & -3 & -1 \end{bmatrix} \begin{bmatrix} 4 \\ 9 \\ 1 \end{bmatrix} = \frac{1}{8} \begin{bmatrix} -16+36+4 \\ -28+9+3 \\ 20-27-1 \end{bmatrix} = \frac{1}{8} \begin{bmatrix} 24 \\ -16 \\ -8 \end{bmatrix} = \begin{bmatrix} 3 \\ -2 \\ -1 \end{bmatrix}$$

Hence, the solution is ($x = 3, y = -2, z = -1$).

4.2 EIGENVALUES AND EIGENVECTORS

For a square matrix A , an Eigenvector and Eigenvalue make this equation true:

$$Av = \lambda v$$

The diagram shows the equation $Av = \lambda v$. Below the 'A' is the word 'Matrix'. Below the 'v' on the left is the word 'Eigenvector'. Below the ' λ ' is the word 'Eigenvalue'. Lines connect the labels to their respective symbols in the equation.

The German prefix “eigen” roughly translates to “self” or “own”. An eigenvector of A is a vector that is taken to a multiple of itself by the matrix transformation $T(x)=Ax$, which perhaps explains the terminology. On the other hand, “eigen” is often translated as “characteristic”; we may think of an eigenvector as describing an intrinsic, or characteristic, property of A .

Note: Eigenvalues and eigenvectors are only for square matrices.

Definition

Let A be an $n \times n$ matrix.

1. An **eigenvector** of A is a nonzero vector v in \mathbb{R}^n such that $Av=\lambda v$, for some scalar λ .
2. An **eigenvalue** of A is a scalar λ such that the equation $Av=\lambda v$ has a nontrivial solution.

If $Av=\lambda v$ for $v \neq 0$, we say that λ is the **eigenvalue for** v , and that v is an **eigenvector for** λ .

4.2.1 Applications of Eigenvalues and Eigenvectors

1. Eigenvalues and Eigenvectors have their importance in linear differential equations where you want to find a rate of change or when you want to maintain relationships between two variables.
2. Think of eigenvalues and eigenvectors as providing summary of a large matrix.
3. We can represent a large set of information in a matrix. Performing computations on a large matrix is a very slow process. To elaborate, one of the key methodologies to improve efficiency in computationally intensive tasks is to reduce the dimensions after ensuring most of the key information is maintained. Hence, one eigenvalue and eigenvector are used to capture key information that is stored in a large matrix. This technique can also be used to improve the performance of data churning components.
4. Component analysis is one of the key strategies that is utilized to reduce dimension space without losing valuable information. The core of component analysis (PCA) is built on the concept of eigenvalues and eigenvectors. The concept revolves around computing eigenvectors and eigenvalues of the covariance matrix of the features.

5. Additionally, eigenvectors and eigenvalues are used in facial recognition techniques such as EigenFaces.
6. They are used to reduce dimension space. The technique of Eigenvectors and Eigenvalues are used to compress the data. As mentioned above, many algorithms such as PCA rely on eigenvalues and eigenvectors to reduce the dimensions.
7. Eigenvalues are also used in regularisation and they can be used to prevent overfitting.
8. Eigenvectors and eigenvalues are used to reduce noise in data. They can help us improve efficiency in computationally intensive tasks. They also eliminate features that have a strong correlation between them and also help in reducing over-fitting.
9. Occasionally we gather data that contains a large amount of noise. Finding important or meaningful patterns within the data can be extremely difficult. Eigenvectors and eigenvalues can be used to construct spectral clustering. They are also used in singular value decomposition.
10. We can also use eigenvector to rank items in a dataset. They are heavily used in search engines and calculus.
11. In non-linear motion dynamics, eigenvalues and eigenvectors can be used to help us understand the data better as they can be used to transform and represent data into manageable sets.

4.2.2 Calculating Eigenvectors and Eigenvalues

Eigenvalues and Eigenvectors have following components:

- The eigenvector is an array with n entries where n is the number of rows (or columns) of a square matrix. The eigenvector is represented as \mathbf{x} . **Key Note:** *The direction of an eigenvector does not change when a linear transformation is applied to it.*
- Therefore, Eigenvector should be a non-null vector
- We are required to find a number of values, known as eigenvalues such that $\mathbf{A} * \mathbf{x} = \lambda * \mathbf{x}$
- The above equation states that we need to find eigenvalue (λ) and eigenvector (\mathbf{x}) such that when we multiply a scalar λ (eigenvalue) to the vector \mathbf{x} (eigenvector) then it should equal to the linear transformation of the matrix \mathbf{A} once it is scaled by vector \mathbf{x} (eigenvector).
- **Eigenvalues are represented as λ .** **Keynote:** **The above equation should not be invertible.**

- There are two special keywords which we need to understand: **Determinant of a matrix and an identity matrix.**
- The determinant of a matrix is a number that is computed from a square matrix. In a nutshell, the diagonal elements are multiplied by each other and then they are subtracted together. We need to ensure that the determinant of the matrix is 0.
- **Last component: We need an Identity Matrix.** An identity square matrix is a matrix that has 1 in diagonal and all of its elements are 0. **The identity matrix** is represented as **I**:

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

We can represent

$$A * x = \text{lambd}a * x$$

$$\text{As: } A * x - \text{lambd}a * x = 0$$

Now, we need to compute a characteristic equation.

$$|A - \text{Lambd}a * I| = 0$$

Subsequently

$$\text{Determinant}(A - \text{Lambd}a * I) = 0$$

The task is to find Eigenvalues of size n for a matrix A of size n.

Therefore, the aim is to find: Eigenvector and Eigenvalues of A such that:

$$A * \text{Eigenvector} - \text{Eigenvalue} * \text{EigenVector} = 0$$

Find Lambda Such that **Determinant(A — Lambd**a * **I) = 0**

Based on the concepts learned above:

1. **lambd**a * **I** is:

$$\begin{bmatrix} \text{LAMBDA} & 0 & 0 \\ 0 & \text{LAMBDA} & 0 \\ 0 & 0 & \text{LAMBDA} \end{bmatrix}$$

If A is:

$$\begin{bmatrix} A & B & C \\ D & E & F \\ G & H & I \end{bmatrix}$$

Then **A — lambd**a * **I** is:

$$\begin{bmatrix} A & B & C \\ D & E & F \\ G & H & I \end{bmatrix} - \begin{bmatrix} \text{LAMBDA} & 0 & 0 \\ 0 & \text{LAMBDA} & 0 \\ 0 & 0 & \text{LAMBDA} \end{bmatrix} = \begin{bmatrix} A - \text{LAMBDA} & B & C \\ D & E - \text{LAMBDA} & F \\ G & H & I - \text{LAMBDA} \end{bmatrix}$$

3. Finally calculate the determinant of (A-lambd*a*I) as:

$$A - \lambda I \begin{bmatrix} E - \lambda & F \\ H & I - \lambda \end{bmatrix} \begin{bmatrix} D \\ G \end{bmatrix} = 0$$

Once we solve the equation above, we will get the values of lambda. These values are the Eigenvalues.

Once we have calculated eigenvalues, we can calculate the Eigenvectors of matrix A by using Gaussian Elimination. Gaussian elimination is about converting the matrix to *row echelon form*. Finally, it is about solving the linear system by back substitution.

A detailed explanation of Gaussian elimination is out of the scope of this article so that we can concentrate on Eigenvalues and Eigenvectors.

Once we have the Eigenvalues, we can find Eigenvector for each of the Eigenvalues. We can substitute the eigenvalue in the lambda and we will achieve an eigenvector.

$$(A - \lambda I) \cdot x = 0$$

5. Practical Example: Let's calculate Eigenvalue and Eigenvector together

Let's find eigenvalue of the following matrix:

$$\begin{pmatrix} 2 & -1 \\ 4 & 3 \end{pmatrix}$$

First, multiply lambda to an identity matrix and then subtract the two matrices

$$\left(\begin{pmatrix} 2 & -1 \\ 4 & 3 \end{pmatrix} - \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right)$$

We need to compute a determinant of:

$$\begin{pmatrix} 2 - \lambda & -1 \\ 4 & 3 - \lambda \end{pmatrix}$$

Therefore:

$$\det \begin{pmatrix} 2 - \lambda & -1 \\ 4 & 3 - \lambda \end{pmatrix} = \lambda^2 - 5\lambda + 10$$

Once we solve the quadratic equation above, we will yield two Eigenvalues:

$$\frac{5}{2} + i \frac{\sqrt{15}}{2}, \frac{5}{2} - i \frac{\sqrt{15}}{2}$$

Now that we have computed Eigenvalues, let's calculate Eigenvectors together

Take the first Eigenvalue (Lambda) and substitute the eigenvalue into the following equation:

$$(A - \lambda I)$$

For the first eigenvalue, we will get the following Eigenvector:

Plug-in to get the matrix

$$\begin{pmatrix} 2 & -1 \\ 4 & 3 \end{pmatrix} - \left(\frac{5}{2} + i\frac{\sqrt{15}}{2}\right) \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} -\frac{1}{2} - i\frac{\sqrt{15}}{2} & -1 \\ 4 & \frac{1}{2} - i\frac{\sqrt{15}}{2} \end{pmatrix}$$

Gives:

$$\begin{pmatrix} -1 + \sqrt{15}i \\ 8 \end{pmatrix}$$

This Eigenvector now represents the key information of matrix A.

Eigenvalues and Eigenvectors in Python

We can use `numpy.linalg.eig` module. It takes in a square matrix as the input and returns eigenvalues and eigenvectors. It also raises an `LinAlgError` if the eigenvalue computation does not converge.

```
import numpy as np
from numpy import linalg as LA
```

```
input = np.array([[2,-1],[4,3]])
```

```
w, v = LA.eig(input)
print(w)
print(v)
```

4.3 MATRIX FACTORIZATION

A matrix decomposition is a way of reducing a matrix into its constituent parts. It is an approach that can simplify more complex matrix operations that can be performed on the decomposed matrix rather than on the original matrix itself.

A common analogy for matrix decomposition is the factoring of numbers, such as the factoring of 10 into 2 x 5. For this reason, matrix decomposition is also called matrix factorization. Like factoring real values, there are many ways to decompose a matrix, hence there are a range of different matrix decomposition techniques.

Two simple and widely used matrix decomposition methods are the LU matrix decomposition and the QR matrix decomposition. Next, we will take a closer look at each of these methods.

4.3.1 LU Matrix Decomposition

The LU decomposition is for square matrices and decomposes a matrix into L and U components.

$$A = L \cdot U$$

Or, without the dot notation.

$$A = LU$$

Where A is the square matrix that we wish to decompose, L is the lower triangle matrix and U is the upper triangle matrix.

The factors L and U are triangular matrices. The factorization that comes from elimination is $A = LU$. The LU decomposition is found using an iterative numerical process and can fail for those matrices that cannot be decomposed or decomposed easily.

A variation of this decomposition that is numerically more stable to solve in practice is called the LUP decomposition, or the LU decomposition with partial pivoting.

$$A = P . L . U$$

The rows of the parent matrix are re-ordered to simplify the decomposition process and the additional P matrix specifies a way to permute the result or return the result to the original order. There are also other variations of the LU.

The LU decomposition is often used to simplify the solving of systems of linear equations, such as finding the coefficients in a linear regression, as well as in calculating the determinant and inverse of a matrix.

The LU decomposition can be implemented in Python with the `lu()` function. More specifically, this function calculates an LPU decomposition.

The example below first defines a 3×3 square matrix. The LU decomposition is calculated, then the original matrix is reconstructed from the components.

```
# LU decomposition

from numpy import array

from scipy.linalg import lu

# define a square matrix

A = array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])

print(A)

# LU decomposition
```

```

P, L, U = lu(A)

print(P)

print(L)

print(U)

# reconstruct

B = P.dot(L).dot(U)

print(B)

```

Running the example first prints the defined 3×3 matrix, then the P, L, and U components of the decomposition, then finally the original matrix is reconstructed.

```

[[1 2 3]

 [4 5 6]

 [7 8 9]]

[[ 0.  1.  0.]

 [ 0.  0.  1.]

 [ 1.  0.  0.]]

[[ 1.          0.          0.          ]

 [ 0.14285714  1.          0.          ]

 [ 0.57142857  0.5         1.          ]]

[[ 7.00000000e+00  8.00000000e+00  9.00000000e+00]

 [ 0.00000000e+00  8.57142857e-01  1.71428571e+00]

 [ 0.00000000e+00  0.00000000e+00 -1.58603289e-16]]

[[ 1.  2.  3.]

 [ 4.  5.  6.]

 [ 7.  8.  9.]]

```

4.3.2 QR Matrix Decomposition

The QR decomposition is for $m \times n$ matrices (not limited to square matrices) and decomposes a matrix into Q and R components.

$$A = Q \cdot R$$

Or, without the dot notation.

$$A = QR$$

Where A is the matrix that we wish to decompose, Q a matrix with the size $m \times m$, and R is an upper triangle matrix with the size $m \times n$. The QR decomposition is found using an iterative numerical method that can fail for those matrices that cannot be decomposed, or decomposed easily. Like the LU decomposition, the QR decomposition is often used to solve systems of linear equations, although is not limited to square matrices.

The QR decomposition can be implemented in NumPy using the `qr()` function. By default, the function returns the Q and R matrices with smaller or 'reduced' dimensions that is more economical. We can change this to return the expected sizes of $m \times m$ for Q and $m \times n$ for R by specifying the mode argument as 'complete', although this is not required for most applications.

The example below defines a 3×2 matrix, calculates the QR decomposition, then reconstructs the original matrix from the decomposed elements.

```
# QR decomposition

from numpy import array

from numpy.linalg import qr

# define a 3x2 matrix

A = array([[1, 2], [3, 4], [5, 6]])

print(A)

# QR decomposition

Q, R = qr(A, 'complete')

print(Q)

print(R)
```



```
# reconstruct
```

```
B = Q.dot(R)
```

```
print(B)
```

Running the example first prints the defined 3×2 matrix, then the Q and R elements, then finally the reconstructed matrix that matches what we started with.

```
[[1 2]
```

```
 [3 4]
```

```
 [5 6]]
```

```
[[ -0.16903085  0.89708523  0.40824829]
```

```
 [ -0.50709255  0.27602622 -0.81649658]
```

```
 [ -0.84515425 -0.34503278  0.40824829]]
```

```
[[ -5.91607978 -7.43735744]
```

```
 [ 0.          0.82807867]
```

```
 [ 0.          0.          ]]
```

```
[[ 1.  2.]
```

```
 [ 3.  4.]
```

```
 [ 5.  6.]]
```

And the python code for Cholesky decomposition is shown below:

```
from numpy import array
```

```
from numpy.linalg import cholesky
```

```
# define a 3x3 matrix
```

```
A = array([[2, 1, 1], [1, 2, 1], [1, 1, 2]])
```

```
print(A)
```

```
# Cholesky decomposition
```

```
L = cholesky(A)
```

```
print(L)
```

```
# reconstruct
```

```
B = L.dot(L.T)
```

```
print(B)
```

Running the example first prints the symmetric matrix, then the lower triangular matrix from the decomposition followed by the reconstructed matrix.

```
[[2 1 1]
```

```
 [1 2 1]
```

```
 [1 1 2]]
```

```
[[ 1.41421356  0.          0.          ]
```

```
 [ 0.70710678  1.22474487  0.          ]
```

```
 [ 0.70710678  0.40824829  1.15470054]]
```

```
[[ 2.  1.  1.]
```

```
 [ 1.  2.  1.]
```

```
 [ 1.  1.  2.]]
```

4.4 PRACTICE EXERCISE

- a. Use the inverse matrix method to solve

$$2x_1 + 3x_2 = 3$$

$$5x_1 + 4x_2 = 11$$

- b. What can you conclude about the solutions of the following systems?

a. $x_1 - 2x_2 = 1, 3x_1 - 6x_2 = 3$

b. $3x_1 + 2x_2 = 7, -6x_1 - 4x_2 = 5$

B.Sc.(DATA SCIENCE)
SEMESTER-III
MATHEMATICAL FOUNDATION FOR DATA SCIENCE

UNIT V: FUNCTIONS, LIMIT AND CONTINUITY

STRUCTURE

5.0 Objectives

5.1 Introduction

5.2 Review of the Basic Concepts

5.3 Functions

5.3.1 Graph of a Function

5.3.2 Bounded Functions and Their Bounds

5.3.3 Monotone Function

5.3.4 Inverse Function

5.3.5 Types of Function

5.4 Concept of Limit

5.5 Continuity

5.6 Key Words

5.7 Practice Exercises

5.0 OBJECTIVES

After going through this unit, you will be able to understand the notions of:

- Function, limit and continuity;
- Familiarise yourself with their forms of simple derivation; and presentation.

5.1 INTRODUCTION

Addressing problems of economics with the aid of mathematical language and techniques have proved quite useful. Therefore, the present discussion is devoted to introduce you to some of basic concepts, which are adopted widely in the formulation of problems and analysing the results. We will discuss the ideas of function, limit and continuity confining ourselves to the system of real numbers (i.e., zero, integers, rational and irrational numbers (positive or negative)). We do not intend to enter the sophisticated technicalities of language as well as derivation of these. To present these themes, therefore, we take the help of examples.

5.2 REVIEW OF THE BASIC CONCEPTS

Set: A **Set** is a collection of objects that are distinct and well defined. These objects are termed as **elements** of the **set**.

Example: The set of integers that are less than or equal to 3. The set is written as –

$$S = \{x < x \text{ is an integer}, 3\}.$$

The **elements** of such a set are = $\{3, 2, 1, -1, -2, -3\}$

The **set** of all possible values of a variable x is called the **domain** of x .

Example: In the equation $x^2 - 3 = 0$, the **domain** of x is = $\{-3, 3\}$.

- **Variable:** It is a representation of a number, which may take different numerical values during a mathematical operation. **Variables** are represented by such as letters of the alphabet, $n, j, s, u, v, w, <, <, \text{etc.}$
- **Continuous variable:** If x takes all possible real values from a given number a to another given number b , then x is called a **continuous variable**. The **domain** of x (or the **interval** of x in this case is denoted by a, x, b or $x \in [a, b]$). The symbol ' \in ' means 'belongs to'.
- **Range:** This is the set of values which $f(x)$ can assume. Thus, **range** of the function $f(x) = \sin x$ is $[-1, 1]$

Range of the function $f(x) = x^2 + 2x + 3$ is entire real line and so on.

Note that $[\]$ is called **closed interval** as it includes both the terminal values a and b . If x does not take either of the two terminal values or both, then we say the **interval is open**, notational $y -a < x < b$ or, $x \in (a, b)$

$a \leq x \leq b$ or, $x \in [a, b]$

$a < x \leq b$ or, $x \in (a, b]$

Example:

In the expression $\cos^{-1}x$, the **interval** of x is $[-1, 1]$ or, $-1 \leq x \leq 1$.

In the expression $x \in [-5, 3] \sqrt{(x-3)(x+5)}$ the **interval** of x is $-5 \leq x \leq 3$ or, $x \in [-5, 3]$.

In the expression $\sqrt[1]{(x-1)(x+1)}$ the interval of x is $-1 < x < 1$ or, $x \in (-1, 1)$.

Constant: A representation, which retains the same numerical value throughout a mathematical operation other than numerical constants (like 1, -2, π , e). Constants are denoted by the earlier letters of the alphabet, such as a, b, c, \dots , etc.

Absolute value: It is represented by the notation $|x|$ and is defined as

$$|x| = x \text{ if } x \geq 0 = -x \text{ if } x < 0$$

Thus, $|-3| = 3$, and $|3| = 3$ and so on.

Note: ‘**Algebraic value**’ is different from the ‘**numerical value**’.

Two results that follow are

i) $|a \pm b \pm c \pm d \pm \dots| \leq |a| + |b| + |c| + |d| + \dots$ Try to check with different numerical values.

ii) $|a \pm b| \leq |a| + |b|$

where $|a| - |b| = | |a| - |b| |$

Try to check with different numerical values.

Think of the meaning of the symbol $|x - a| \leq k$. Check that the symbol $|x - a| \leq k$ means, $a - k \leq x \leq a + k$ or $x \in [a - k, a + k]$.

5.3 FUNCTIONS

By a **function** of x , defined for a given **domain** is a quantity y , which has a single and definite value for every value of x in its **domain**.

The variable x , to which we may arbitrarily assign different values in the given **domain**, is called the **independent variable** (or **argument**) and y is called the **dependent variable** (or **function**).

A **function** may be undefined for some particular value or values of x in a given interval. The forms $a/0$ or $0/0$ are meaningless. In other words, we cannot define division by zero.

If $y = f(x)$ is a **function** of x , then in case $f(x)$ is a mathematical expression involving x , $f(a)$, i.e., the value of the function for $x = a$ in general be obtained by putting a for x in the expression for $f(x)$.

Example:

i) $f(x) = \overline{\sin^2 x} - \cos x, f(0) = -1.$

However, note that, $f(-1)$ is undefined.

Sometimes functions may be given in an implicit form.

For example, $x^3 - Iy^2 = 0$. From this we can write y as a function of x or, x as a function of y , i.e., $x = (Iy^2)^{1/3}$.

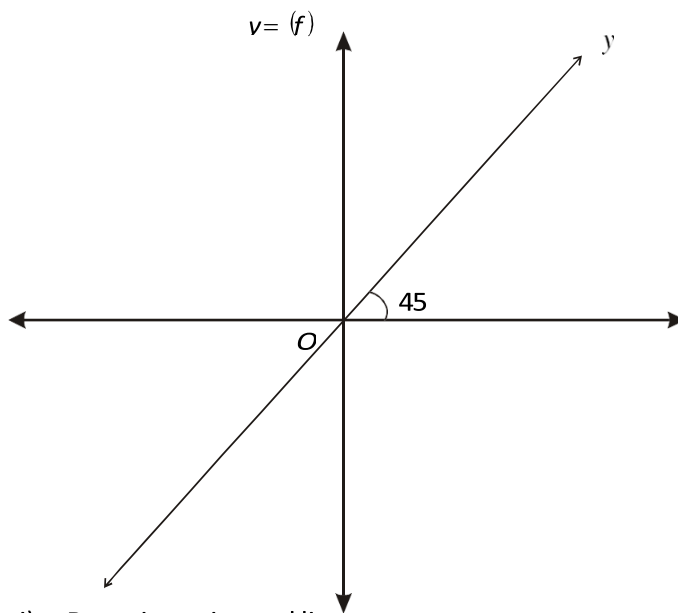
5.3.1 Graph of a Function

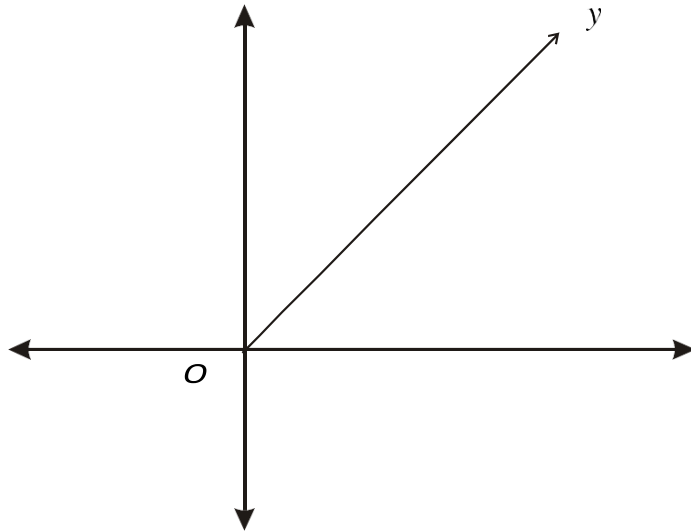
In drawing the graph, it is sufficient if we know the definite value of y corresponding to every value of x in the defined **domain**. The graph shows the way in which the function is related to and changes with the argument.

Exercise

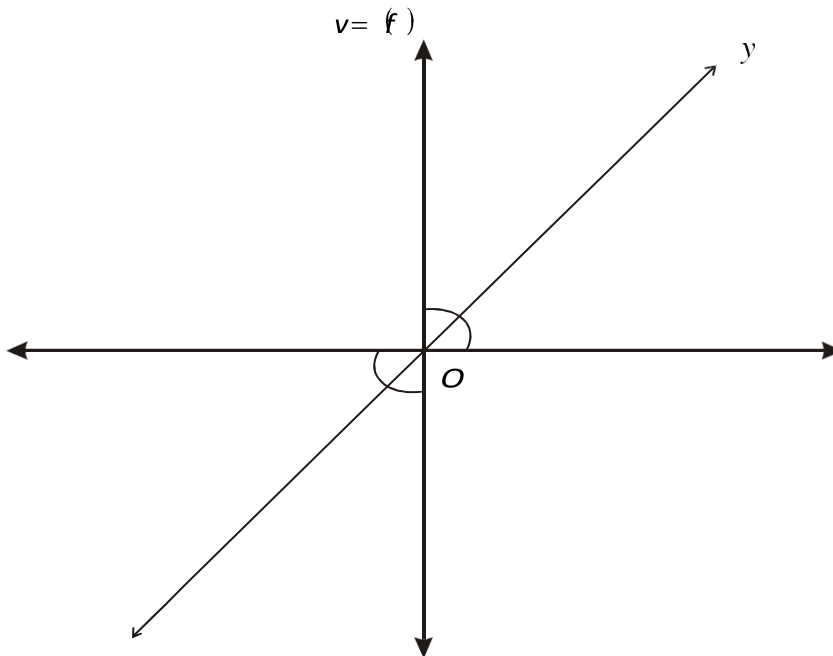
Draw the graph for the following functions:

- i) $f(x) = x$ ii) $f(x) = < x, x \dots 0$ iii) $f(x) = x^2/x$





ii. Domain positive values of x and $x=0$



iii. Domain, entire real line except $x=0$.
The function is undefined when $x = 0$.

5.3.2 Bounded Functions And Their Bounds

Let $f(x)$ be a function defined in the interval (a, b) . If a finite number k is there such that $f(x) \leq k$ for every x , then $f(x)$ is **bounded above** in the interval and k is an **upper bound** of the function $f(x)$.

Similarly, if a finite number m is there such that $f(x) \geq m$ for every x , then $f(x)$ is **bounded below** in the interval and m is a **lower bound** of the function $f(x)$. If $f(x)$ is both **bounded above** and **bounded below**, then $f(x)$ is said to be simply **bounded**.

It may be useful to note that a function need not have an upper bound or a lower bound.

Example:

$$f(x) = x + 1; x \in (-1, 1)$$

Thus, upper bound = 0

lower bound = 2

and the function is bounded.

Exercise

Think of a function, which is not bounded above or bounded below. The above function when x can take any real value.

5.3.3 Monotone Function

Broadly speaking, a function which always increases or decreases can be called monotonically increasing (or decreasing) function. Let x_1, x_2 be two numbers such that $x_1 < x_2$. $f(x)$ is **monotonically increasing** if $f(x_1) \leq f(x_2)$, for all $(x_1, x_2) \in \text{domain of } x$.

Example: $f(x) = x + 1$ is **monotonically increasing**.

5.3.4 Inverse Function

The **inverse** of the **function** $y = f(x)$ is defined as $x = f^{-1}(y)$.

Example: i) $y = x^3; f^{-1}(y) = y^{1/3}$; ii) $y = \sin x; x = f^{-1}(y) = \sin^{-1}y$ and so on.

5.3.5 Types of Function

The expression $y = f(x)$ is actually a rule. Now we specify several types of function, each representing a different rule

I. Algebraic Function

a) **Constant function:** A function whose range consists of only one element is called a **constant function**, e.g., $y = 1$.

b) **Polynomial function:** The above function is a specific case of a more general function, known as **polynomial function**. The function is of the form –

$$y = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$

a_i = coefficient of the term x^i
 $i = 0, 1, 2, \dots, n$

n = a positive real number

Depending on the value of n , we get several types of functions: when $n = 0$, $y = a_0$ (constant function)

when $n = 1$, $y = a_0 + a_1x$ (linear function)

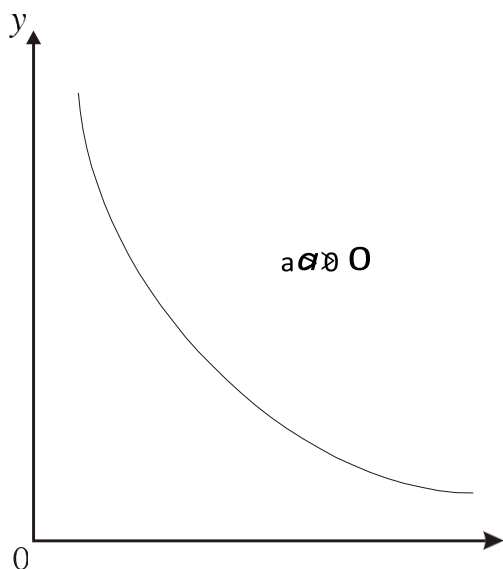
when $n = 2$, $y = a_0 + a_1x + a_2x^2$ (quadratic function)

... and so on.

Plotted in a coordinate plane, a **linear function** will appear as a straight line, a_0 is the **y intercept** (or the **vertical intercept**) of the function. The term a_1 measures the **slope** (or the steepness) of the function. When $a_1 > 0$, the line will be upward sloping and when $a_1 < 0$, the line will be downward sloping

Remember that a **quadratic function** is a parabola.

c. **Rational function:** This is an expression of ratio of two **polynomial functions**. Thus, **polynomial function** is also a **rational function**,



The curve approaches both the axis as x or y value is large, but never touches either axis.

II. Non-algebraic functions: Any function expressed in terms of polynomials and/or roots of polynomials is an **algebraic function**.

We will state some functions, which are not expressible as above. Broadly, they may be called **non-algebraic functions** or **transcendental functions**

- a. Exponential function: $y = ab^x$
- b. Trigonometric function: It involves expressions like $\sin x$, $\cos x$, $\tan x$ or their inverse.

5.4 CONCEPT OF LIMIT

By the phrase 'x tends to the value a', we mean $|x - a|$ is gradually smaller and smaller. (The successive values of x may either be greater or less than the value 'a'). Then we can write this as $x \rightarrow a$.

If the successive values of x are always greater than a, then we say 'x tends to a from right' and write this as $x \rightarrow a+0$ or $x \rightarrow a+$. Similarly, if the successive values of x are always less than a, then we say 'x tends to a from left' and write it as $x \rightarrow a-0$ or $x \rightarrow a-$.

5.4.1 Limit of a Function

$\lim_{x \rightarrow a} f(x)$ means the following:

When x approaches a constant quantity a from either side, there exists a definite finite number t towards which f(x) tends. Such a numerical difference of f(x) and t can be made as small as we can think by taking x sufficiently close to a. Then t is defined as the limit of f(x) as x tends to a. This is written

$$\text{as } \lim_{x \rightarrow a} f(x) = t$$

Analytically, $\lim_{x \rightarrow a} f(x) = t$ means, given any pre-assigned positive quantity ϵ (as small as we like), we can find another positive quantity δ such that $|f(x) - t| < \epsilon \forall x$ such that $0 < |x - a| < \delta$.

Example

$$\lim_{x \rightarrow 1} (2x - 1) = 1$$

$$|f(x) - 1| = |2x - 2|. \text{ Let } \epsilon \text{ be a very small positive quantity.}$$

$$\text{Then } |f(x) - 1| < \epsilon \text{ if } 0 < |x - 1| < \epsilon/2$$

Take

$\epsilon/2 = \delta > 0$. Then 2

$$|f(x) - 1| < \epsilon \text{ if } 0 < |x - 1| < \delta$$

Thus, $\lim_{x \rightarrow a} (2x - 1) = 1$

a) Right Hand Limit of a Function

$\lim_{x \rightarrow a^+} f(x) = t_1$ means the following

Given any pre-fixed positive quantity ϵ (however small), we can determine a positive quantity δ such that $|f(x) - t_1| < \epsilon$ whenever $0 < x - a < \delta$, i.e., $a < x < a + \delta$.

This **right hand limit** is also denoted by the symbol $f(a + 0)$.

The above limit is so called because as x approaches the value a from the right (i.e., from bigger values), the numerical difference between $f(x)$ and t_1 can be made as small as we please by taking x closer to a , keeping x always greater than a .

b) Left Hand Limit of a Function

Analogously, we may define **left-hand limit** of a function as $\lim_{x \rightarrow a^-} f(x) = t_2$

When a quantity t_2 can be obtained such that given any pre-assigned quantity ϵ , however small, we can determine a positive quantity δ , so that $|f(x) - t_2| < \epsilon$ whenever $0 < a - x < \delta$, i.e., $a - \delta < x < a$

The **left hand limit** is also denoted by the symbol $f(a - 0)$.

When $\lim_{x \rightarrow a^+} f(x) = \lim_{x \rightarrow a^-} f(x)$, each of these is equal to $\lim_{x \rightarrow a} f(x)$, i.e., for $\lim_{x \rightarrow a} f(x)$ to exist, each of $\lim_{x \rightarrow a^+} f(x)$ and $\lim_{x \rightarrow a^-} f(x)$ must exist.

c) Functions Tending to Infinity

If corresponding to any pre-assigned quantity N , however large, we can determine a positive quantity δ , such that $f(x) > N$, whenever $0 < |x - a| < \delta$, we say $\lim_{x \rightarrow a} f(x) = \infty$.

5.5 CONTINUITY

Intuitively, a function $f(x)$ is **continuous** provided its graph is continuous, i.e., a **continuous** function does not have any break at any point of its graph.

More formally, a function $f(x)$ is said to be **continuous** for $x = a$, provided $\lim_{x \rightarrow a} f(x)$ exists, finite and is equal to $f(a)$. i.e., for $f(x)$ to be **continuous** at $x = a$, we should have

$$\lim_{x \rightarrow a} f(x) = f(a)$$

If $f(x)$ is **continuous** for every value of x in its **domain**, it is said to be **continuous** throughout the interval. A function, which is not continuous at a point, is said to have a **discontinuity** at that point.

Example:

In the examples of the graph of a function, the first function is **continuous** and 2nd and 3rd functions are **discontinuous** at $x = 0$. Otherwise, they are **continuous** at other points of the **domain**. This is because the **limit** is non-existent for both the function at $x = 0$.

Analytically, the function $f(x)$ is **continuous** at $x = a$, provided $f(a)$ exists and given any pre-assigned positive quantity ϵ , however small, we can determine a positive quantity δ such that $|f(x) - f(a)| < \epsilon$ for all values of x such that $a - \delta < x < a + \delta$.

Some Properties of Continuous Functions

- i) The sum or difference of two continuous function is a continuous function. This result is valid for any finite number of functions.
- ii) The product of two continuous functions is a continuous function. This result is also valid for any finite number of functions.
- iii) The quotient of two continuous functions is a continuous function, provided the denominator is not zero anywhere for the range of values considered.
- iv) If $f(x)$ is continuous at $x = a$ and $f(a) \neq 0$, then in the neighbourhood of $x = a$, $f(x)$ has the same sign as that of $f(a)$, i.e., we can get a positive quantity δ such that $f(x)$ preserves the same sign as that of $f(a)$ for every value of $f(x)$ in the interval $a - \delta < x < a + \delta$.
- v) If $f(x)$ is continuous throughout the interval (a, b) , and $f(a)$ and $f(b)$ are of opposite signs, then there is at least one value, say ξ of x within the interval for which $f(\xi) = 0$.
- vi) Let $f(x)$ is continuous throughout the interval (a, b) and $f(a) \neq f(b)$. Then $f(x)$ assumes every value between $f(a)$ and $f(b)$ at least once in the interval.
- vii) A function, which is continuous throughout a closed interval, is bounded therein.
- viii) A continuous function in an interval actually attains its upper and lower bounds, at least once each, in the interval.
- ix) A function $f(x)$, continuous in a closed interval $[a, b]$, attains every intermediate

value between its upper and lower bounds in the interval, atleast once.

5.6 KEY WORDS

Set: A collection of objects that are distinct and well defined. These objects are termed as **elements** of the **set**.

Variable: Representation of a number, which may take different numerical values during a mathematical operation.

Continuous Variable: If x takes all possible real values from a given number a to another given number b , then x is called a **continuous variable**.

Constant: A representation, which retains the same numerical value throughout a mathematical operation other than numerical constants (like 1, -2, π , e).

Functions: **Function** of x , is defined for a given **domain** is a quantity y , which has a single and definite value for every value of x in its **domain**.

Bounded Functions and their Bounds: Let $f(x)$ be a function defined in the interval (a, b) . If a finite number K is there such that $f(x) \leq K$ for every x , then $f(x)$ is **bounded above** in the interval and K is an **upper bound** of the function $f(x)$.

Similarly, if a finite number m is there such that $f(x) \geq m$ for every x , then $f(x)$ is **bounded below** in the interval and m is a **lower bound** of the function $f(x)$. If $f(x)$ is both **bounded above** and **bounded below**, then $f(x)$ is said to be simply **bounded**.

Monotone Function: A function which always increases or decreases can be called monotonically increasing (or decreasing) function.

Inverse Function: The **inverse** of the **function** $y = f(x)$ is defined as $x = f^{-1}(y)$.

Constant Function: A function whose range consists of only one element is called a **constant function**

Polynomial Function: A function that is a sum of power functions, with positive integer exponents, multiplied by constants, such as $f(x) = x^3 - 2x^2 + 4x + 7 = 1x^3 + -2x^2 + 4x + 7x^0$ and a power function is given by an exponential formula, where the independent variable is the base of the exponential, such as $x^3, x^{-2}, x^{1/2}, x^{\square}$, etc.

B.Sc.(DATA SCIENCE)
SEMESTER-III
MATHEMATICAL FOUNDATION FOR DATA SCIENCE

UNIT VII: LINEAR PROGRAMMING

STRUCTURE

7.0 Objective

7.1 Linear Programming

7.2 Formulation of the Linear Programming Model

7.2.1 Terminology Used In LPP

7.2.2 The Mathematical Expression of the LP Model

7.3 Formulation of LPP Steps

7.3.1 Example 1. Production Allocation Problem

7.3.2 Example 2: Product Mix Problem

7.3.3 Example: 3: Product Production

7.3.4 Example 4: Portfolio Selection (Investment Decisions)

7.3.5 Example 5: Inspection Problem

7.3.6 Example 6: Trim Loss Problem

7.3.7 Example 7: Media Selection

7.3.8 Example 8: Diet Problem

7.4 Merits of LPP

7.5 Demerits of LPP

7.6 Practice Exercise

7.0 OBJECTIVE

To understand General form of linear Programming and formulating Linear Programming Problems assumptions

7.1 LINEAR PROGRAMMING

Linear Programming is a problem solving approach that has been developed to help managers to make decisions. Linear Programming is a mathematical technique for determining the optimum allocation of resources and obtaining a particular objective when there are alternative uses of the resources, money, manpower, material, machine and other facilities.

The Nature Of Linear Programming Problem

Two of the most common are:

1. The product-mix problem
2. The blending Problem

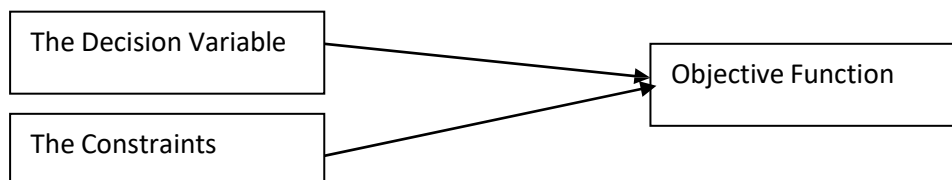
In the product- mix problem there are two or more product also called candidates or activities competing for limited resources. The problem is to find out which products to include in production plan and in what quantities these should be produced or sold in order to maximize profit, market share or sales revenue.

The blending problem involves the determination of the best blend of available ingredients to form a certain quantity of a product under strict specifications. The best blend means the least cost blend of the required inputs.

7.2 FORMULATION OF THE LINEAR PROGRAMMING MODEL

Three components are:

1. The decision variable
2. The environment (uncontrollable) parameters
3. The result (dependent) variable



Linear Programming Model is composed of the same components

7.2.1 Terminology Used In LPP

- 1. Components of LP Problem:** Every LPP is composed of a. Decision Variable, b. Objective Function, c. Constraints.
- 2. Optimization:** Linear Programming attempts to either maximise or minimize the values of the objective function.
- 3. Profit or Cost Coefficient:** The coefficient of the variable in the objective function express the rate at which the value of the objective function increases or decreases by including in the solution one unit of each of the decision variable.
- 4. Constraints:** The maximisation (or minimisation) is performed subject to a set of constraints. Therefore LP can be defined as a constrained optimisation problem. They reflect

the limitations of the resources.

5. Input-Output coefficients: The coefficient of constraint variables are called the Input-Output Coefficients. They indicate the rate at which a given resource is utilized or depleted. They appear on the left side of the constraints.

6. Capacities: The capacities or availability of the various resources are given on the right hand side of the constraints.

7.2.2 The Mathematical Expression of the LP Model

The general LP Model can be expressed in mathematical terms as shown below:

Let

O_j = Input-Output

Coefficient C_j = Cost (Profit)

Coefficient

b_i = Capacities (Right Hand

Side) X_j = Decision Variables

Find a vector $(x_1, x_2, x_3, \dots, x_n)$ that minimise or maximise a linear objective function

$F(x)$ where $F(x) = c_1x_1 + c_2x_2 + \dots + c_nx_n$

subject to linear constraints

$a_1x_1 + a_2x_2 + \dots + a_nx_n = b_2$

$a_1x_1 + a_2x_2 + \dots + a_nx_n \leq b_2$

.....

.....

.....

.....

$a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \leq b_2$

and non-negativity constraints

$x_1 \geq 0, x_2 \geq 0, \dots, x_n \geq 0$

7.3 FORMULATION OF LPP STEPS

1. Identify decision variables
2. Write objective function
3. Formulate constraints

7.3.1 Example 1. Production Allocation Problem

A firm produces three products. These products are processed on three different machines. The time required to manufacture one unit of each of the three products and the daily capacity of the three machines are given in the table below:

Machine	Time per unit (Minutes)			Machine Capacity (minutes/day)
	Product 1	Product 2	Product 3	
M1	2	3	2	440
M2	4	-	3	470
M3	2	5	-	430

It is required to determine the daily number of units to be manufactured for each product. The

profit per unit for product 1, 2 and 3 is Rs. 4, Rs.3 and Rs.6 respectively. It is assumed that all the amounts produced are consumed in the market. Formulate the mathematical (L.P.) model that will maximise the daily profit.

Formulation of Linear Programming Model

Step 1

From the study of the situation find the key-decision to be made. In the given situation key decision is to decide the extent of products 1, 2 and 3, as the extents are permitted to vary.

Step 2

Assume symbols for variable quantities noticed in step 1. Let the extents (amounts) of products 1, 2 and 3 manufactured daily be x_1 , x_2 and x_3 units respectively.

Step 3

Express the feasible alternatives mathematically in terms of variable. Feasible alternatives are those which are physically, economically and financially possible. In the given situation feasible alternatives are sets of values of x_1 , x_2 and x_3 units respectively.

where x_1, x_2 and $x_3 \geq 0$.

since negative production has no meaning and is not feasible.

Step 4

Mention the objective function quantitatively and express it as a linear function of variables. In the present situation, objective is to maximize the profit.

i.e., $Z = 4x_1 + 3x_2 + 6x_3$

Step 5

Put into words the influencing factors or constraints. These occur generally because of constraints on availability (resources) or requirements (demands). Express these constraints also as linear equations/inequalities in terms of variables.

Here, constraints are on the machine capacities and can be mathematically expressed as

$$2x_1 + 3x_2 + 2x_3 \leq 440$$

$$4x_1 + 0x_2 + 3x_3 \leq 470$$

$$2x_1 + 5x_2 + 0x_3 \leq 430$$

7.3.2 Example 2: Product Mix Problem

A factory manufactures two products A and B. To manufacture one unit of A, 1.5 machine hours and 2.5 labour hours are required. To manufacture product B, 2.5 machine hours and 1.5 labour hours are required. In a month, 300 machine hours and 240 labour hours are available.

Profit per unit for A is Rs. 50 and for B is Rs. 40. Formulate as LPP.

Solution:

Products	Resource/unit	
	Machine	Labour
A	1.5	2.5
B	2.5	1.5
Availability	300 hrs	240 hrs

There will be two constraints. One for machine hours availability and for labour hours availability.

Decision variables

X_1 = Number of units of A manufactured per month.
 X_2 = Number of units of B manufactured per month.

The objective function:

$$\text{Max } Z = 50x_1 + 40x_2$$

Subjective Constraints

For machine hours

$$1.5x_1 + 2.5x_2 \leq 300$$

For labour hours

$$2.5x_1 + 1.5x_2 \leq 240$$

Non negativity

$$x_1, x_2 \geq 0$$

7.3.3 Example: 3: Product Production

A company produces three products A, B, C.

For manufacturing three raw materials P, Q and R are used.

Profit per unit

A - Rs. 5, B - Rs. 3, C - Rs. 4

Resource requirements/unit

Raw Material Product	P	Q	R
A	-	20	50
B	20	30	-
C	30	20	40

Maximum raw material availability:

P - 80 units; Q - 100 units; R - 150 units. Formulate LPP.

Solution:

Decision variables:

x_1 = Number of units of A

x_2 = Number of units of B

x_3 = Number of units of C

Objective Function

Since Profit per unit is given, objective function is maximisation

$$\text{Max } Z = 5x_1 + 3x_2 + 4x_3$$

Constraints:

For P: $0x_1 + 20x_2 + 30x_3 \leq 80$

For Q: $20x_1 + 30x_2 + 20x_3 \leq 100$

For R: $50x_1 + 0x_2 + 40x_3 \leq 150$

(for B, R is not required)

$$X_1, X_2, X_3 \geq 0$$

7.3.4 Example 4: Portfolio Selection (Investment Decisions)

An investor is considering investing in two securities 'A' and 'B'. The risk and return associated with these securities is different.

Security 'A' gives a return of 9% and has a risk factor of 5 on a scale of zero to 10. Security 'B' gives return of 15% but has risk factor of 8.

Total amount to be invested is Rs. 5, 00, 000/- Total minimum returns on the investment should be 12%. Maximum combined risk should not be more than 6. Formulate as LPP.

Solution:

Decision Variables:

X_1 = Amount invested in Security A X_2 = Amount invested in

Objective Function: Security B

The objective is to maximise the return on total investment.

$$\therefore \text{Max } Z = 0.09 X_1 + 0.15 X_2 \quad (\% = 0.09, 15\% = 0.15)$$

Constraints:

1. Related to Total Investment:

$$X_1 + X_2 = 5, 00, 000$$

2. Related to Risk:

$$5X_1 + 8X_2 = (6 \times 5, 00, 000)$$

$$5X_1 + 8X_2 = 30, 00, 000$$

3. Related to Returns:

$$0.09X_1 + 0.15X_2 = (0.12 \times 5, 00, 000)$$

$$\therefore 0.09X_1 + 0.15X_2 = 60, 000$$

4. Non-negativity

$$X_1, X_2 \geq 0$$

7.3.5 Example 5: Inspection Problem

A company has two grades of inspectors, I and II to undertake quality control inspection. At least 1, 500 pieces must be inspected in an 8-hour day. Grade I inspector can check 20 pieces in an hour with an accuracy of 96%. Grade II inspector checks 14 pieces an hour with an accuracy of 92%.

Wages of grade I inspector are Rs. 5 per hour while those of grade II inspector are Rs. 4 per hour. Any error made by an inspector costs Rs. 3 to the company. If there are, in all, 10 grade I inspectors and 15 grade II inspectors in the company, find the optimal assignment of inspectors that minimise the daily inspection cost.

Solution:

Let x_1 and x_2 denote the number of grade I and grade II inspectors that may be assigned the job of quality control inspection.

The objective is to minimise the daily cost of inspection. Now the company has to incur two types of costs; wages paid to the inspectors and the cost of their inspection errors. The cost of grade I inspector/hour is

$$\text{Rs. } (5 + 3 \times 0.04 \times 20) = \text{Rs. } 7.40.$$

Similarly, cost of grade II inspector/hour is

$$\text{Rs. } (4 + 3 \times 0.08 \times 14) = \text{Rs. } 7.36.$$

∴ The objective function is

$$\text{minimise } Z = 8(7.40x_1 + 7.36x_2) = 59.20x_1 + 58.88x_2.$$

Constraints are

on the number of grade I inspectors: $x_1 \leq 10$,

on the number of grade II inspectors: $x_2 \leq 15$

on the number of pieces to be inspected daily: $20 \times 8x_1 + 14 \times 8x_2 \geq 1500$

or $160x_1 + 112x_2 \geq 1500$

where, $x_1, x_2 \geq 0$.

7.3.6 Example 6: Trim Loss Problem

A manufacturer of cylindrical containers receives tin sheets in widths of 30 cm and 60 cm respectively. For these containers the sheets are to be cut to three different widths of 15 cm, 21 cm and 27 cm respectively. The number of containers to be manufactured from these three widths are 400, 200 and 300 respectively. The bottom plates and top covers of the containers are purchased directly from the market. There is no limit on the lengths of standard tin sheets. Formulate the LPP for the production schedule that minimises the trim losses.

Solution:

Key decision is to determine how each of the two standard widths of tin sheets be cut to the require widths so that trim losses are minimum.

From the available widths of 30 cm and 60 cm, several combinations of the three required widths of 15 cm, 21 cm and 27 cm are possible.

Let x_{ij} represent these combinations. Each combination results in certain trim

loss. Constraints can be formulated as follows:

The possible cutting combinations (plans) for both types of sheets are shown in the table below:

Width (cm)	i = I (30 cm)			i = II (60 cm)					
	X11	X12	X13	X21	X22	X23	X24	X25	X26
15	2	0	0	4	2	2	1	0	0
21	0	1	0	0	1	0	2	1	0
27	0	0	1	0	0	1	0	1	2
Trim Loss (cm)	0	9	3	0	9	3	3	12	6

Thus, the constraints are

$$2x_{11} + 4x_{21} + 2x_{22} + 2x_{23} + x_{24} \geq$$

$$400x_{12} + x_{22} + 2x_{24} + x_{25} \geq 200$$

$$x_{13} + x_{23} + x_{25} + x_{26} \geq 300$$

Objective is to maximise the trim losses.

i.e., minimise $Z = 9x_{12} + 3x_{13} + 9x_{22} + 3x_{23} + 3x_{24} + 12x_{25} + 6x_{26}$ where $x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{24}, x_{25}, x_{26} \geq 0$.

7.3.7 Example 7: Media Selection

An advertising agency is planning to launch an ad campaign. Media under consideration are T.V., Radio & Newspaper. Each medium has different reach potential and different cost.

Minimum 10, 000, 000 households are to be reached through T.V. Expenditure on newspapers should not be more than Rs. 10, 00, 000. Total advertising budget is Rs. 20 million.

Following data is available:

Medium	Cost per Unit (Rs.)	Reach per unit (No. of households)
Television	2, 00, 000	20, 00, 000
Radio	80, 000	10, 00, 000
Newspaper	40, 000	2, 00, 000

Solution:

Decision Variables:

x_1 = Number of units of T.V. ads, x_2 = Number of units of Radio ads,

x_3 = Number of units of Newspaper ads.

Objective function: (Maximise reach)

$$\text{Max. } Z = 20, 00, 000 x_1 + 10, 00, 000 x_2 + 2, 00, 000 x_3$$

Subject to constraints:

$$20, 00, 000 x_1 \geq 10, 000, 000 \dots\dots\dots (\text{for T.V.})$$

$$40, 000 x_3 \leq 10, 00, 000 \dots\dots\dots (\text{for Newspaper})$$

$$2, 00, 000 x_1 + 80, 000 x_2 + 40, 000 x_3 \leq 20, 000, 000 \dots\dots\dots (\text{Ad. budget})$$

$$x_1, x_2, x_3 \geq 0$$

∴ Simplifying constraints:

$$\text{for T.V.} \quad 2 x_1 \geq 10 \quad \therefore x_1 \geq 5$$

$$\text{for Newspaper} \quad 4 x_3 \leq 100 \quad \therefore x_3 \leq 25$$

Ad. Budget

$$20 x_1 + 8 x_2 + 4 x_3 \leq 2000$$

$$5 x_1 + 2 x_2 + x_3 \leq 500$$

$$x_1, x_2, x_3 \geq 0$$

7.3.8 Example 8: Diet Problem

Vitamins B1 and B2 are found in two foods F1 and F2. 1 unit of F1 contains 3 units of B1 and 4 units of B2. 1 unit of F2 contains 5 units of B1 and 3 units of B2 respectively.

Minimum daily prescribed consumption of B1 & B2 is 50 and 60 units respectively. Cost per unit of F1 & F2 is Rs. 6 & Rs. 3 respectively.

Formulate as LPP.

Solution:

Vitamins	Foods		Minimum Consumption
	F1	F2	
B1	3	5	30
B2	5	7	40

Decision Variables:

x_1 = No. of units of P1 per day.

x_2 = No. of units of P2 per day.

Objective function:

$$\text{Min. } Z = 100x_1 + 150x_2$$

Subject to constraints:

$$3x_1 + 5x_2 \geq 30 \text{ (for } N_1\text{)}$$

$$5x_1 + 7x_2 \geq 40 \text{ (for } N_2\text{)}$$

$$x_1, x_2 \geq 0$$

7.3.9 Example 9: Blending Problem

A manager at an oil company wants to find optimal mix of two blending processes.

Formulate LPP.

Data:

Process	Input (Crude Oil)		Output (Gasoline)	
	Grade A	Grade B	X	Y
P1	6	4	6	9
P2	5	6	5	5

Profit per operation: Process 1 (P1) = Rs. 4,000

Process 2 (P2) = Rs. 5,000

Maximum availability of crude oil: Grade A = 500 units

Grade B = 400 units

Minimum Demand for Gasoline: X = 300 units

$$Y = 200 \text{ units}$$

Solution:

Decision Variables:

$$x_1 = \text{No. of operations of P1}$$

$$x_2 = \text{No. of operations of P2}$$

Objective Function:

$$\text{Max. } Z = 4000 x_1 + 5000 x_2$$

Subjective to constraints:

$$6x_1 + 5x_2 \leq 500$$

$$4x_1 + 6x_2 \leq 400$$

$$6x_1 + 5x_2 \geq 300$$

$$9x_1 + 5x_2 \geq 200$$

$$x_1, x_2 \geq 0$$

7.3.10 Example 10: Farm Planning

A farmer has 200 acres of land. He produces three products X, Y & Z. Average yield per acre for X, Y & Z is 4000, 6000 and 2000 kg.

Selling price of X, Y & Z is Rs. 2, 1.5 & 4 per kg respectively. Each product needs fertilizers. Cost of fertilizer is Rs. 1 per kg. Per acre need for fertilizer for X, Y & Z is 200, 200 & 100 kg respectively. Labour requirements for X, Y & Z is 10, 12 & 10 man hours per acre. Cost of labour is Rs. 40 per man hour. Maximum availability of labour is 20, 000 man hours. Formulate as LPP to maximise profit.

Solution:

Decision variables:

The production/yield of three products X, Y & Z is given as per acre.

Hence,

$$x_1 = \text{No. of acres allocated to}$$

$$X$$

$$x_2 = \text{No. of acres allocated to}$$

$$Y$$

$$x_3 = \text{No. of acres allocated to}$$

$$Z$$

Objective Function:

$$\text{Profit} = \text{Revenue} - \text{Cost}$$

$$\text{Profit} = \text{Revenue} - (\text{Fertiliser Cost} + \text{Labour Cost})$$

Product	X	Y	Z
Revenue	2 (4000) x ₁	1.5 (6000) x ₂	4 (2000) x ₃
(-) Less:			
Fertiliser Cost	1 (200) x ₁	1 (200) x ₂	1 (100) x ₃
Labour Cost	40 (10) x ₁	40 (12) x ₂	40 (10) x ₃
Profit	7400 x ₁	8320 x ₂	7500 x ₃

∴ Objective function

$$\text{Max.} = 7400 x_1 + 8320 x_2 + 7500 x_3$$

Subject to constraints:

$$x_1 + x_2 + x_3 = 200 \quad (\text{Total Land})$$

$$10 x_1 + 12 x_2 + 10 x_3 \leq 20,000 \quad (\text{Max Man hours})$$

$$x_1, x_2, x_3 \geq 0$$

7.4 MERITS OF LPP

1. Helps management to make efficient use of resources.
2. Provides quality in decision making.
3. Excellent tools for adjusting to meet changing demands.
4. Fast determination of the solution if a computer is used.
5. Provides a natural sensitivity analysis.
6. Finds solution to problems with a very large or infinite number of possible solution.

7.5 DEMERITS OF LPP

- 1. Existence of non-linear equation:** The primary requirements of Linear Programming is the objective function and constraint function should be linear. Practically linear relationship do not exist in all cases.
- 2. Interaction between variables:** LP fails in a situation where non-linearity in the equation emerge due to joint interaction between some of the activities like total effectiveness.
- 3. Fractional Value:** In LPP fractional values are permitted for the decision variable.
- 4. Knowledge of Coefficients of the equation:** It may not be possible to state all coefficients in the objective function and constraints with certainty.

7.6 PRACTICE EXERCISE

1. Explain what is meant by decision variables, objective function and constraints in Linear Programming.
2. Give the mathematical formulation of the linear programming problems.
3. What are the components of LPP? What is the significance of non-negativity restriction?
4. State the limitations of LPP.
5. Give the assumptions and advantages of LPP.
6. An investor wants to identify how much to invest in two funds, one equity and one debt. Total amount available is Rs. 5,00,000. Not more than Rs. 3,00,000 should be invested in a single fund. Returns expected are 30% in equity and 8% in debt. Minimum return on total investment should be 15%. Formulate as LPP.

B.Sc.(DATA SCIENCE)

SEMESTER-III

MATHEMATICAL FOUNDATION FOR DATA SCIENCE

UNIT VIII: MAXIMUM AND MINIMUM PROBLEMS AND SIMPLEX METHOD

STRUCTURE

8.0 Objective

8.1 Linear Programming Solution - Graphical Method

8.2 Methodology of Graphical Method

8.2.1 Maximisation

8.2.2 Minimisation

8.2.3 Maximisation-Mixed Constraints

8.3 Minimisation Mixed Constraints

8.4 Principle Of Simplex Method

8.5 Practice Exercises

8.0 Objective

To understand the concept of Standard Maximum and Minimum Problem, Graphical Method and Simplex Method.

8.1 LINEAR PROGRAMMING SOLUTION - GRAPHICAL METHOD

There are two methods available to find optimal solution to a Linear Programming Problem.

One is graphical method and the other is simplex method.

Graphical method can be used only for a two variables problem i.e. a problem which involves two decision variables. The two axes of the graph (X & Y axis) represent the two decision variables X_1 & X_2 .

8.2 METHODOLOGY OF GRAPHICAL METHOD

Step 1: Formulation of LPP (Linear Programming Problem)

Use the given data to formulate the LPP.

8.2.1 Maximisation

Example 1

A company manufactures two products A and B. Both products are processed on two machines M1 & M2.

	M1	M2
A	6 Hrs/Unit	2 Hrs/Unit
B	4 Hrs/Unit	4 Hrs/Unit
Availability	7200 Hrs/month	4000 Hrs/month

Profit per unit for A is Rs. 100 and for B is Rs. 80. Find out the monthly production of A and B to maximise profit by graphical method.

Formulation of LPP

X_1 = No. of units of

A/Month

X_2 = No. of units of

B/Month

Max $Z = 100 X_1 +$

$80 X_2$

Subject to constraints:

$6 X_1 + 4 X_2 \leq 7200$

$2 X_1 + 4 X_2 \leq$

4000

$X_1, X_2 \geq 0$

Step 2: Determination of each axis

Horizontal (X) axis: Product A (X_1)

Vertical (Y) axis: Product B (X_2)

Step 3: Finding co-ordinates of constraint lines to represent constraint lines on the graph.

The constraints are presently in the form of inequality (\leq). We should convert them into equality to obtain co-ordinates.

Constraint No. 1: $6 X_1 + 4 X_2 \leq 7200$

Converting into equality:

$$6 X_1 + 4 X_2 \leq 7200$$

X_1 is the intercept on X axis and X_2 is the intercept on Y axis.

To find X_1 , let $X_2 = 0$

$$6 X_1 = 7200$$

$$6 X_1 = 7200$$

$$\therefore X_1 = 1200; X_2 = 0 (1200, 0)$$

To find X_2 , let $X_1 = 0$

$$4 X_2 = 7200$$

$$X_2 = 1800; X_1 = 0 (0, 1800)$$

Hence the two points which make the constraint line are:

(1200, 0) and (0, 1800)

Note: When we write co-ordinates of any point, we always write (X_1 , X_2). The value of X_1 is written first and then value of X_2 . Hence, if for a point X_1 is 1200 and X_2 is zero, then its co-ordinates will be (1200, 0).

Similarly, for second point, X_1 is 0 and X_2 is 1800. Hence, its co-ordinates are (0, 1800).

Constraint No. 2:

$$2 X_1 + 4 X_2 \leq 4000$$

To find X_1 , let $X_2 = 0$

$$2 X_1 = 4000$$

$$\therefore X_1 = 2000; X_2 = 0 (2000, 0)$$

To find X_2 , let $X_1 = 0$

$$4 X_2 = 4000$$

$$\therefore X_2 = 1000; X_1 = 0 (0, 1000)$$

Each constraint will be represented by a single straight line on the graph. There are two constraints, hence there will be two straight lines.

The co-ordinates of points are:

1. Constraint No. 1: (1200, 0) and (0, 1800)

2. Constraint No. 2: (2000, 0) and (0, 1000)

Step 4: Representing constraint lines on graph

To mark the points on the graph, we need to select appropriate scale. Which scale to take will depend on maximum value of X_1 & X_2 from co-ordinates.

For X_1 , we have 2 values \longrightarrow 1200 and 2000

\therefore Max. value for $X_2 = 2000$

For X_2 , we have 2 values \longrightarrow 1800 and 1000

\therefore Max. value for $X_2 = 1800$

Assuming that we have a graph paper 20 X 30 cm. We need to accommodate our lines such that for X-axis, maximum value of 2000 contains in 20 cm.

\therefore Scale 1 cm = 200 units

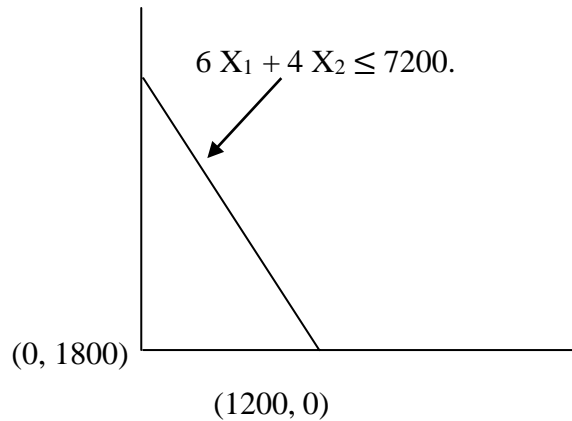
\therefore 2000 units = 10 cm (X-axis)

1800 units = 9 cm (Y-axis)

The scale should be such that the diagram should not be too small.

Constraint No. 1:

The line joining the two points (1200, 0) and (0, 1800) represents the constraint $6 X_1 + 4 X_2 \leq 7200$.

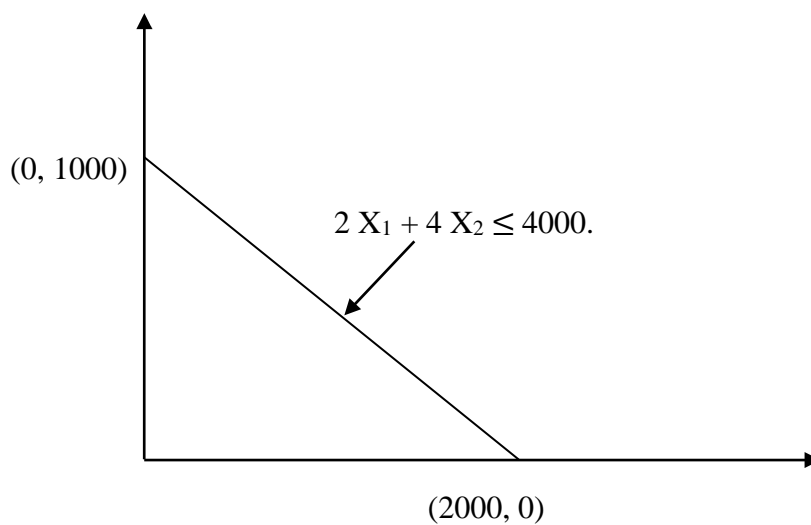


Every point on the line will satisfy the equation (equality) $6 X_1 + 4 X_2 \leq 7200$. Every point below the line will satisfy the inequality (less than) $6 X_1 + 4 X_2 \leq 7200$.

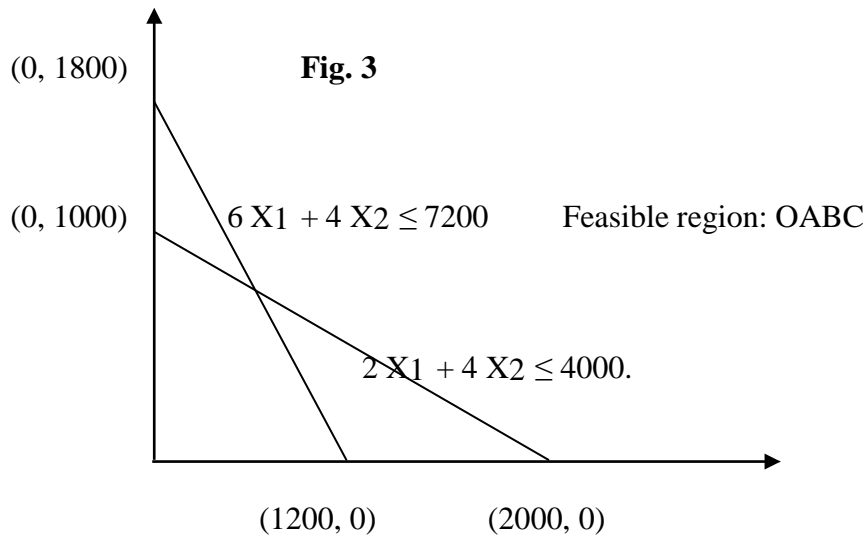
Constraint No. 2:

The line joining the two points (2000, 0) and (0, 1000) represents the constraint $2 X_1 + 4 X_2 \leq 4000$

Every point on the line will satisfy the equation (equality) $2 X_1 + 4 X_2 \leq 4000$. Every point below the line will satisfy the inequality (less than) $2 X_1 + 4 X_2 \leq 4000$.



Now the final graph will look like this:

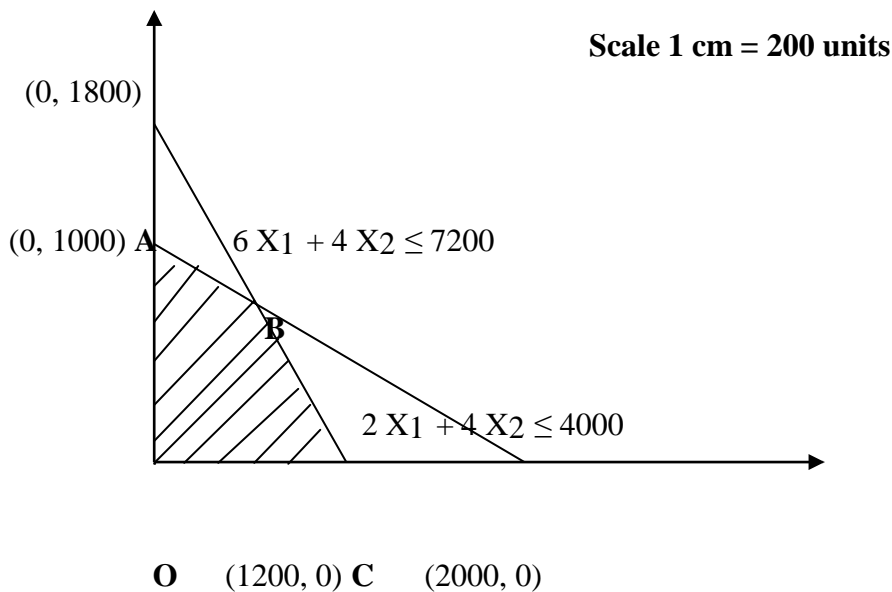


Step 5: Identification of Feasible Region

The feasible region is the region bounded by constraint lines. All points inside the feasible region or on the boundary of the feasible region or at the corner of the feasible region satisfy all constraints.

Both the constraints are 'less than or equal to' (\leq) type. Hence, the feasible region should be inside both constraint lines.

Hence, the feasible region is the polygon OABC. 'O' is the origin whose coordinates are (0, 0). O, A, B and C are called vertices of the feasible region.



Step 6: Finding the optimal Solution

The optimal solution always lies at one of the vertices or corners of the feasible region.

To find optimal solution:

We use corner point method. We find coordinates (X_1, X_2 Values) for each vertex or corner point. From this we find 'Z' value for each corner point.

Vertex	Co-ordinates	Z = 100 X ₁ + 80 X ₂
O	X ₁ = 0, X ₂ = 0 From Graph	Z = 0
A	X ₁ = 0, X ₂ = 1000 From Graph	Z = Rs. 80, 000
B	X ₁ = 800, X ₂ = 600 From Simultaneous equations	Z = Rs. 1, 28, 000
C	X ₁ = 1200, X ₂ = 0 From Graph	Z = Rs. 1, 20, 000

Max. Z = Rs. 1, 28, 000 (At point B)

For B → B is at the intersection of two constraint lines $6 X_1 + 4 X_2 \leq 7200$ and $2 X_1 + 4 X_2 \leq 4000$. Hence, values of X₁ and X₂ at B must satisfy both the equations.

We have two equations and two unknowns, X₁ and X₂. Solving

$$6 X_1 + 4 X_2 \leq 7200 \quad (1)$$

$$2 X_1 + 4 X_2 \leq 4000 \quad (2)$$

$$4 X_1 = 3200 \quad \text{Subtracting (2) from (1)}$$

$$X_1 = 800$$

Substituting value of X₁ in equation (1), we

$$4 X_2 = 2400 \quad \therefore X_2 = 600$$

Solution

Optimal Profit = Max Z = Rs. 1, 28, 000

Product Mix:

$$X_1 = \text{No. of units of A / Month} =$$

$$800 X_2 = \text{No. of units of A / Month} =$$

$$600$$

ISO Profit line:

ISO profit line is the line which passes through the points of optimal solution (Maximum Profit). The slope of the iso-profit line depends on the objective function.

In the above example, the objective function is:

$$\text{Max. } Z = 100 X_1 + 80 X_2$$

How to find slope of iso-profit line:

$$\text{Equation of a straight line: } y = mx + c$$

where, m = slope of the straight line

In our case, y means 'X₂' and x means '

X₁'. c means 'Z'.

$$\therefore X_2 = m \cdot X_1 + Z$$

Converting original objective function in this format:

$$\text{Max. } Z = 100 X_1 + 80 X_2$$

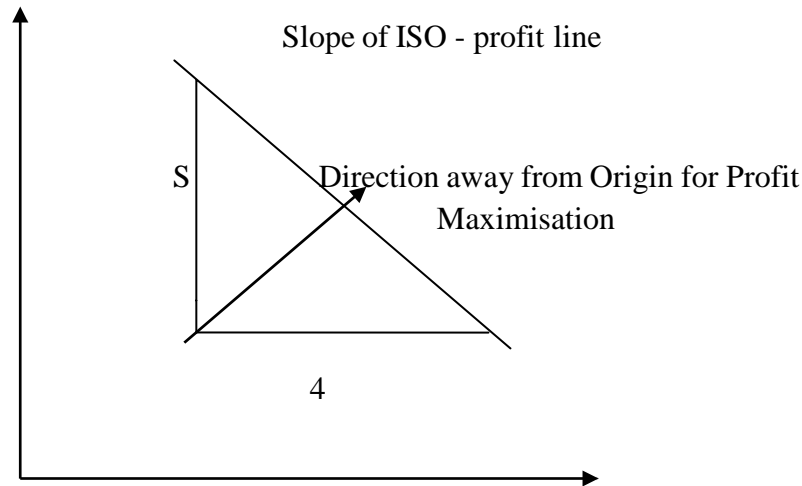
$$\therefore 80 X_2 = Z - 100 X_1 = -100 X_1 + Z$$

$$\therefore X_2 = \frac{-100}{80} X_1 + \frac{Z}{80}$$

$$\therefore X_2 = \frac{-5}{4} X_1 + \frac{Z}{80}$$

$$\therefore \text{Slope of ISO profit line} = \frac{-5}{4}$$

Negative sign indicates that the line will slope from left to right downwards. And slope will be 5/4. Every 4 units on X-axis for 5 units on Y-axis.



As we start from the origin and go away from origin to maximise profit, 'B' is the last point on the feasible region that is intersected by the iso-profit line. Hence, B is the optimal solution.

8.2.2 Minimisation

Example 2

A firm is engaged in animal breeding. The animals are to be given nutrition supplements everyday. There are two products A and B which contain the three required nutrients.

Nutrients	Quantity/unit		Minimum Requirement
	A	B	
1	72	12	216
2	6	24	72
3	40	20	200

Product cost per unit are: A: rs. 40; B: Rs. 80. Find out quantity of product A & B to be given to provide minimum nutritional requirement.

Step 1: Formulation as LPP

X₁ - Number of units of A

X₂ - Number of units of B

Z - Total Cost

$$\text{Min. } Z = 40 X_1 + 80 X_2$$

Subject to constraints:

$$72 X_1 + 12 X_2 \geq 216$$

$$6 X_1 + 24 X_2 \geq 72$$

$$40 X_1 + 20 X_2 \geq 200$$

$$X_1, X_2 \geq 0.$$

Step 2: Determination of each axis

Horizontal (X) axis: Product A

(X1) Vertical (Y) axis: Product B

(X2)

Step 3: Finding co-ordinates of constraint lines to represent the graph

All constraints are 'greater than or equal to' type. We should convert them into equality:

1. Constraint No. 1: $72 X_1 + 12 X_2 \geq 216$

Converting into equality

$$72 X_1 + 12 X_2 = 216$$

To find X_1 , let $X_2 = 0$

$$72 X_1 = 216 \quad \therefore X_1 = 3, X_2 = 0 \quad (3, 0)$$

To find X_2 , let $X_1 = 0$

$$12 X_2 = 216 \quad \therefore X_1 = 0, X_2 = 18 \quad (0, 18)$$

2. Constraint No. 2:

$$6 X_1 + 24 X_2 \geq 72$$

To find X_1 , let $X_2 = 0$

$$6 X_1 = 72 \quad \therefore X_1 = 12, X_2 = 0 \quad (12, 0)$$

$$6 X_1 = 72$$

To find X_2 , let $X_1 = 0$

$$24 X_2 = 72 \quad \therefore X_1 = 0, X_2 = 3 \quad (0, 3)$$

3. Constraint No. 3:

$$40 X_1 + 20 X_2 \geq 200$$

To find X_1 , let $X_2 = 0$

$$40 X_1 = 200 \quad \therefore X_1 = 5, X_2 = 0 \quad (5, 0)$$

$$40 X_1 = 200$$

To find X_2 , let $X_1 = 0$

$$20 X_2 = 200 \quad \therefore X_1 = 0, X_2 = 10 \quad (0, 10)$$

The co-ordinates of points are:

1. Constraint No. 1: (3, 0) & (0, 18)

2. Constraint No. 2: (12, 0) & (0, 3)

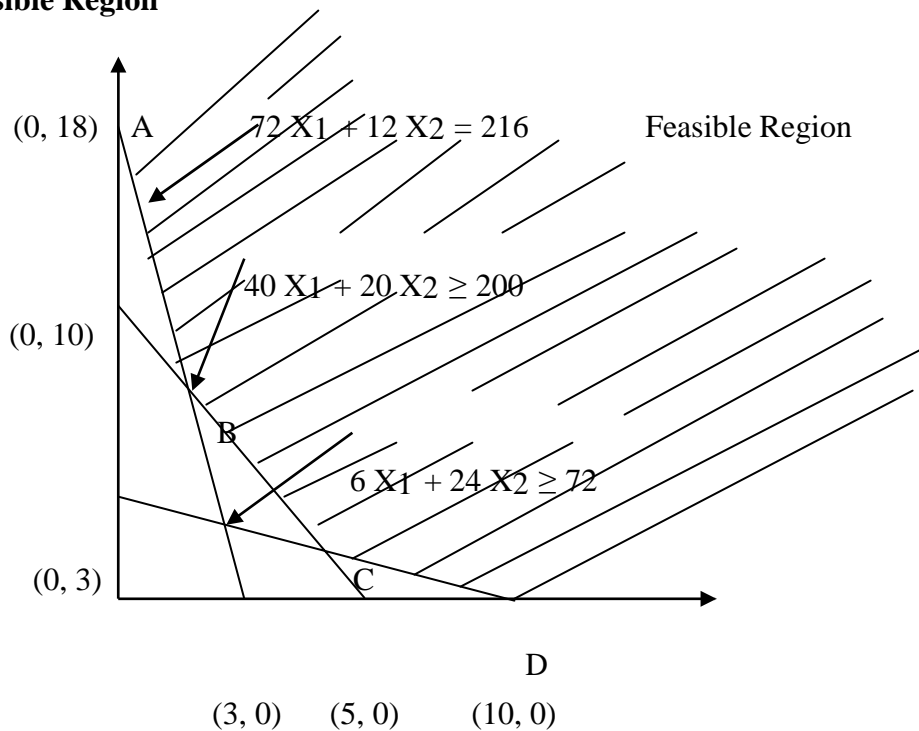
3. Constraint No. 3: (5, 0) & (0, 10)

Every point on the line will satisfy the equation (equality) $72 X_1 + 12 X_2 = 216$.

Every point above the line will satisfy the inequality (greater than) $72 X_1 + 12 X_2 = 216$.

Similarly, we can draw lines for other two constraints.

Step 5: Feasible Region



All constraints are greater than or equal to (\geq) type. Hence, feasible region should be above (to the right of) all constraints.

The vertices of the feasible region are A, B, C & D.

Step 6: Finding the optimal solution

Corner Point Method

Vertex	Co-ordinates	$Z = 40 X_1 + 80 X_2$
A	$X_1 = 0, X_2 = 18$ From Graph	$\therefore Z = 1,440$
B	$X_1 = 2, X_2 = 6$ From Simultaneous Equations	$\therefore Z = 560$
C	$X_1 = 4, X_2 = 2$ From Simultaneous Equations	$\therefore Z = 320$
D	$X_1 = 12, X_2 = 0$ From graph	$\therefore Z = 480$

\therefore Min. $Z = \text{Rs. } 320$ (At point 'C')

For B - Point B is at intersection of constraint lines ' $72 X_1 + 12 X_2 \geq 216$ ' and ' $40 X_1 + 20 X_2 \geq 200$ '. Hence, point B should satisfy both the equations.

$$72 X_1 + 12 X_2 = 216 \quad (1)$$

$$40 X_1 + 20 X_2 = 200 \quad (2)$$

$$\therefore 360 X_1 + 60 X_2 = 1080 \quad (1) \times 5$$

$$120 X_1 + 60 X_2 = 600 \quad (2) \times 3$$

$$\therefore 240 X_1 = 480$$

$$X_1 = 2$$

Substituting value of X_1 in equation (1), we get:

$$12 X_2 = 216 - 144 = 72$$

$$X_2 = 6$$

For C - Point C is at intersection of constraint lines ' $6 X_1 + 24 X_2 = 72$ ' and ' $40 X_1 + 20 X_2 = 200$ '. Hence, point C should satisfy both the equations.

$$6 X_1 + 24 X_2 = 72 \quad (1)$$

$$40 X_1 + 20 X_2 = 200 \quad (2)$$

$$30 X_1 + 120 X_2 = 360 \quad (1) \times 5$$

$$240 X_1 + 120 X_2 = 1200 \quad (2) \times 6$$

$$210 X_1 = 840$$

$$X_1 = 4$$

Substituting value of X_1 in equation (1), we

$$\text{get } 24 X_2 = 72 - 24 = 48$$

$$X_2 = 2$$

Solution

Optimal Cost = $Z \text{ min} = \text{Rs. } 320$

Optimal Product Mix:

$X_1 = \text{No. of units of product A} =$

$4 X_2 = \text{No. of units of product B} =$

2

ISO Cost Line

ISO Cost line passes through the point of optimal solution (Minimum cost)

Objective function: $Z = 40 X_1 + 80 X_2$

Equation of straight line: $y = mx + c$

where $m = \text{slope}$

In this case, y is X_2 & x is

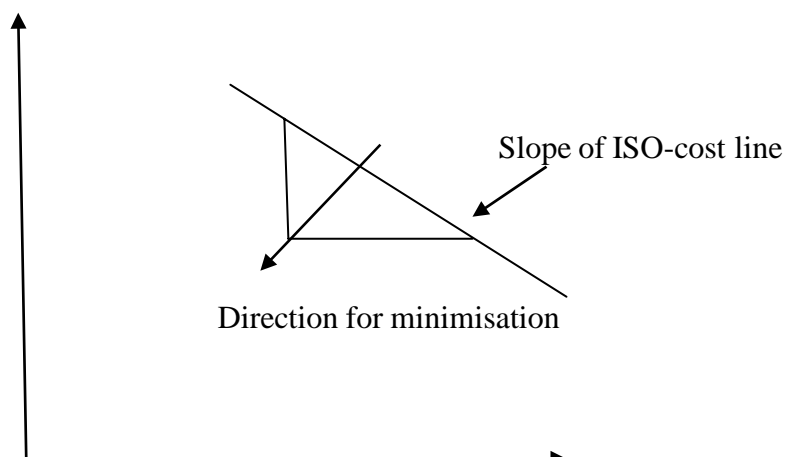
$$X_1 Z = 40 X_1 + 80 X_2$$

$$\therefore 80 X_2 = -40 X_1 + Z$$

$$\therefore X_2 = -1/2 x_1 + Z/80$$

$$\therefore \text{Slope of ISO-cost line} = -1/2$$

Sloping from left to right downwards



∴ Point C is the nearest to the origin.

8.2.3 Maximisation-Mixed Constraints

Example 1

A firm makes two products P1 & P2 and has production capacity of 18 tonnes per day. P1 & P2 require same production capacity. The firm must supply at least 4 t of P1 & 6 t of P2 per day. Each tonne of P1 & P2 requires 60 hours of machine work each. Maximum machine hours available are 720. Profit per tonne for P1 is Rs. 160 & P2 is Rs. 240. Find optimal solution by graphical method.

LPP Formulation

X_1 = Tonnes of P1 / Day

X_2 = Tonnes of P2 / Day

Max. $Z = 160 X_1 + 240$

X_2

Subject to constraints

$X_1 \geq 4$

$X_2 \geq 6$

$X_1 + X_2 \leq 18$

$60 X_1 + 60 X_2 \leq 720$

$X_1, X_2 \geq 0$

Coordinates for constraint lines:

1. $X_1 \geq 4$ (4, 0)..... No value for X_2 , ∴ $X_2 = 0$

2. $X_2 \geq 6$ (0, 6)..... No value for X_1 , ∴ $X_1 = 0$

3. $X_1 + X_2 \leq 18$ (18, 0) (0, 18)

4. $60 X_1 + 60 X_2 \leq 720$ (12, 0) (0, 12)

If $X_1 = 0$, $60 X_2 = 720$ ∴ $X_2 = 12$ (0, 12)

If $X_2 = 0$, $60 X_1 = 720$ ∴ $X_1 = 12$ (12, 0)

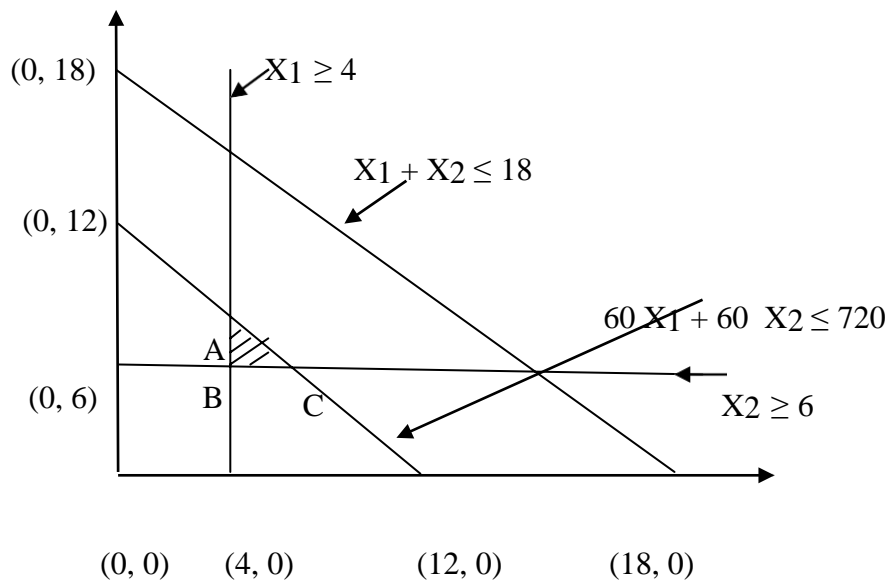
Graph: X_1 : X Axis

X_2 : Y Axis

Scale:

Maximum value for $X_1 = 18$; Maximum value for $X_2 = 18$; ∴ Scale: 1 cm = 2 Tonnes.

Two constraints are 'greater than or equal to' type. Hence, feasible region will be above or to the right of these constraint lines. Two constraints are 'less than or equal to' type. Hence, feasible region will be below or to the left of these constraint lines. Hence, feasible region is ABC.



Optimal Solution
Corner Point Method

Vertex	Coordinates	$Z = 160 X_1 + 240 X_2$
A	$X_1 = 4, X_2 = 8$ Simultaneous Equation	$\therefore Z = \text{Rs. } 2, 560$
B	$X_1 = 4, X_2 = 6$ From Graph	$\therefore Z = \text{Rs. } 2, 080$
C	$X_1 = 6, X_2 = 6$ Simultaneous Equations	$\therefore Z = \text{Rs. } 2, 400$

For A $\rightarrow X_1 = 4$ from graph

A is on the line $60 X_1 + 60 X_2 = 720$

$$60 X_2 = 720 - 60 (4) = 480 \quad \therefore X_2 = 8$$

For C $\rightarrow X_2 = 6$ from graph

A is on the line $60 X_1 + 60 X_2 = 720$

$$60 X_1 = 720 - 60 (6) = 360 \quad \therefore X_1 = 6$$

$\therefore Z = \text{Rs. } 2, 560$ (At point 'A')

Solution

Optimal Profit Z. Max = Rs. 2, 560

X_1 = Tonnes of P1 = 4 tonnes

X_2 = Tonnes of P2 = 8 tonnes.

8.3 MINIMISATION MIXED CONSTRAINTS

Example 1:

A firm produces two products P and Q. Daily production upper limit is 600 units for total production. But at least 300 total units must be produced every day. Machine hours consumption per unit is 6 for P and 2 for Q. At least 1200 machine hours must be used daily. Manufacturing costs per unit are Rs. 50 for P and Rs. 20 for Q. Find optimal solution for the LPP graphically.

LPP formulation

X_1 = No. of Units of P / Day

X_2 = No. of Units of Q /

Day Min. $Z = 50 X_1 + 20$

X_2

Subject to constraints

$$X_1 + X_2 \leq 600$$

$$X_1 + X_2 \geq 300$$

$$6 X_1 + 2 X_2 \geq 1200$$

$$X_1, X_2 \geq 0$$

Coordinates for Constraint lines

1. $X_1 + X_2 = 600$

If $X_1 = 0, X_2 = 600$ $\therefore (0, 600)$

If $X_2 = 0, 60 X_1 = 600$ $\therefore (600, 0)$

2. $X_1 + X_2 = 300$

If $X_1 = 0, X_2 = 300$ $\therefore (0, 300)$

If $X_2 = 0, 60 X_1 = 300$ $\therefore (300, 0)$

3. $6 X_1 + 2 X_2 \geq 1200$

If $X_1 = 0, 2X_2 = 1200$ $\therefore X_2 = 600$ $(0, 600)$

If $X_2 = 0, 6 X_1 = 1200$ $\therefore X_1 = 200$ $(200, 0)$

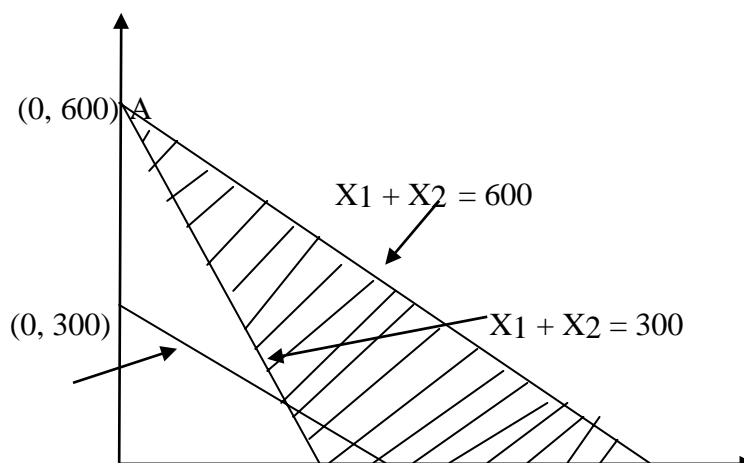
Graph: X_1 : X Axis

X_2 : Y Axis

Scale:

Maximum value for $X_1 = 600$; Maximum value for $X_2 = 600$; \therefore Scale: 1 cm = 50 units.

Feasible region is ABCD.



$$6 X_1 + 2 X_2 \geq 1200 \text{ B}$$

$$(0, 0) \quad (200, 0) \quad \text{C} \quad (300, 0) \quad \text{D} \quad (600, 0)$$

Two constraints are 'greater than or equal to' type. Hence, feasible region will be above or to the right of these constraint lines. Two constraints are 'less than or equal to' type. Hence, feasible region will be below or to the left of these constraint lines. Hence, feasible region is ABCD.

Optimal Solution

Corner Point Method

Vertex	Coordinates	Z = 160 X ₁ + 240 X ₂
A	X ₁ = 0, X ₂ = 600 From Graph	∴ Z = Rs. 12, 000
B	X ₁ = 150, X ₂ = 150 Simultaneous Equations	∴ Z = Rs. 10, 500
C	X ₁ = 300, X ₂ = 0 From Graph	∴ Z = Rs. 15, 000
D	X ₁ = 600, X ₂ = 0 From Graph	∴ Z = Rs. 30, 000

$$\text{Min. } Z = \text{Rs. } 10, 500$$

For B - B is at intersection of two constraint lines '6 X₁ + 2 X₂ ≥ 1200' and 'X₁ + X₂ =

$$300' \quad 6 X_1 + 2 X_2 \geq 1200 \quad (1)$$

$$X_1 + X_2 = 300 \quad (2)$$

$$2X_1 + 2X_2 = 600 \quad (2) \times 2$$

$$2X_1 = 600$$

$$X_1 = 150$$

Substituting value in Equation (2), X₂ =

$$150. \text{Solution}$$

$$\text{Optimal Cost} = \text{Rs. } 10, 500/-$$

$$X_1 = \text{No. of Units of P} = 150$$

$$X_2 = \text{No. of Units of P} =$$

$$150.$$

8.4 PRINCIPLE OF SIMPLEX METHOD

We explain the principle of the **Simplex method** with the help of the two variable linear programming problem .

Example I

$$\text{Maximise } 50x_1 + 60x_2$$

Subject to :

$$2x_1 + x_2 \leq 300$$

$$3x_1 + 4x_2 \leq 509$$

$$4x_1 + 7x_2 \leq 812$$

$$x_1 \geq 0, x_2 \geq 0$$

Solution

We introduce variables $x_3 \geq 0, x_4 \geq 0, x_5 \geq 0$ So that the constraints become equations

$$2x_1 + x_2 + x_3 = 300$$

$$3x_1 + 4x_2 + x_4 = 509$$

$$4x_1 + 7x_2 + x_5 = 812$$

The variables x_3, x_4, x_5 are **known as slack variables** corresponding to the three constraints. The system of **equations** has five variables (including the slack variables) and three equations.

Basic Solution

In the system of equations as presented above we may equate any two variables to zero. The system then consists of three equations with three variables. If this system of three equations with three variables is solvable such a solution is known as a **basic solution**.

In the example considered above suppose we take $x_1 = 0, x_2 = 0$. The solution of the system with remaining three variables is $x_3 = 300, x_4 = 509, x_5 = 812$. This is a **basic solution** of the system. The variables x_3, x_4 and x_5 are known as **basic variables** while the variables x_1, x_2 are known as **non basic variables** (variables which are equated to zero).

Since there are three equations and five variables the two non basic variables can be chosen in ${}^5C_2 = 10$ ways. Thus, the maximum number of basic solutions is 10, for in some cases the three variable three equation problem may not be solvable.

In the general case, if the number of constraints of the linear programming problem is m and the number of variables (including the slack variables) is n then there are **at most** ${}^nC_{n-m} = {}^nC_m$ basic solutions.

Basic Feasible Solution

A basic solution of a linear programming problem is a **basic feasible solution** if it is feasible, i.e. all the variables are non negative. The solution $x_3 = 300, x_4 = 509, x_5 = 812$ is a basic feasible solution of the problem. Again, if the number of constraints is m and the number of variables (including the slack variables) is n , the **maximum** number of basic feasible solution is ${}^nC_{n-m} = {}^nC_m$

The following result (Hadley, 1969) will help you to identify the extreme points of the convex set of feasible solutions analytically.

Every **basic feasible solution** of the problem is an **extreme point** of the convex set of feasible solutions and every extreme point is a basic feasible solution of the set of Constraints.

When several variables are present in a linear programming problem it is not possible to identify the extreme points geometrically. But we can identify them through the basic feasible solutions. Since one of the basic feasible solutions will maximise or minimise the objective function, we can carry out this search starting from one basic feasible solution to another. The simplex method provides a systematic search so that the objective function increases (in the case of maximisation) progressively until the basic feasible solution has been identified where the objective function is maximised.

8.5 PRACTICE EXERCISES

1. What is meant by feasible region in graphical method.
2. What is meant by 'iso-profit' and 'iso-cost line' in graphical solution.
3. Mr. A. P. Ravi wants to invest Rs. 1, 00, 000 in two companies 'A' and 'B' so as not to exceed Rs. 75, 000 in either of the company. The company 'A' assures average return of 10% in whereas the average return for company 'B' is 20%. The risk factor rating of company 'A' is 4 on 0 to 10 scale whereas the risk factor rating for 'B' is 9 on similar scale. As Mr. Ravi wants to maximise his returns, he will not accept an average rate of return below 12% risk or a risk factor above 6.

Formulate this as LPP and solve it graphically.

4. Solve the following LPP graphically and interpret the result.

$$\text{Max. } Z = 8X_1 + 16 X_2$$

Subject to:

$$X_1 + X_2 \leq$$

$$200X_2 \leq 125$$

$$3X_1 + 6X_2 \leq 900$$

$$X_1, X_2 \geq 0$$

5. A furniture manufacturer makes two products - tables and chairs.

Processing of these products is done on two types of machines A and B. A chair requires 2 hours on machine type A and 6 hours on machine type B. A table requires 5 hours on machine type A and no time on Machine type B. There are 16 hours/day available on machine type A and 30 hours/day on machine type B. Profits gained by the manufacturer from a chair and a table are Rs. 2 and Rs. 10 respectively. What should be the daily production of each of the two products? Use graphical method of LPP to find the solution

ਜਗਤ ਗੁਰੂ ਨਾਨਕ ਦੇ ਨੂਰ ਚ ਰੌਸ਼ਨ ਹੈ ਇਹ ਵਿਸ਼ਵ ਵਿਦਿਆਲਾ

ਜਗਤ ਗੁਰੂ ਨਾਨਕ ਦੇਵ

ਸ਼ਬਦ ਗੁਰੂ ਨਾਨਕ ਦੇਵ

ਕਿਰਤ ਕਰਮ ਦੀ

ਸ਼ਬਦ ਸੁਰਤ ਦੀ

ਸੰਗਤ ਪੰਗਤ

ਵੰਡ ਛਕਣ ਦੀ

ਖੋਜ, ਵਿਵੇਕ ਅਤੇ ਸਿਰਜਣ ਦੀ

ਕਰਤਾ ਪੁਰਖ ਰਹੱਸ ਦਰਸ਼ਨ ਦੀ

ਸਿੱਖਿਆ ਦੇਵਣ ਵਾਲਾ

ਰੌਸ਼ਨ ਹੈ ਇਹ ਵਿਸ਼ਵ ਵਿਦਿਆਲਾ

ਗਗਨ ਮੰਡਲ ਵਿਚ ਜਗਦੇ ਤਾਰੇ

ਦੀਪਕ ਸੋਹਣ ਦੁਆਰੇ ਦੁਆਰੇ

ਕਾਇਆ ਕਾਗਦ ਅੱਖਰ ਜਗਦੇ

ਪੁਸ਼ਪ ਸੁਹਾਵਣ ਧਰਤੀ ਹਿਰਦੇ

ਇਹ ਤੇਰੀ ਲੀਲਾ ਵਿਸਮਾਦੀ

ਨਿਤ ਨਵੇਲੀ ਆਦਿ ਜੁਗਾਦੀ

ਇਸ ਲੀਲਾ ਦੇ ਕਰਮ ਖੰਡ ਵਿਚ

ਤੇਰਾ ਸ਼ਬਦ ਸਵਾਰਨਹਾਰਾ

ਤੇਰਾ ਨਾਦ ਉਜਾਲਾ

ਰੌਸ਼ਨ ਹੈ ਇਹ ਵਿਸ਼ਵ ਵਿਦਿਆਲਾ

ਜਗਤ ਗੁਰੂ ਨਾਨਕ ਦੇਵ

ਸ਼ਬਦ ਗੁਰੂ ਨਾਨਕ ਦੇਵ



ਜਗਤ ਗੁਰੂ ਨਾਨਕ ਦੇਵ
ਪੰਜਾਬ ਸਟੇਟ ਓਪਨ ਯੂਨੀਵਰਸਿਟੀ
ਪਟਿਆਲਾ

JAGAT GURU NANAK DEV
PUNJAB STATE OPEN UNIVERSITY, PATIALA
(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)