# Unit 1

# Contents

**Data Management**

- Design Data Architecture and manage the data for analysis, understand various sources, Internal Sources, External Sources, Data like Sensors/Signals/GPS**,** etc. Data Types, Methods of collecting data, Data Management, benefits of data management, Data Quality(noise, outliers, missing values, duplicate data) and Data Processing and processing, Data Cleaning, Data Transformation, Data Reduction

# Data, Data architecture design

- Data is one of the essential pillars of enterprise architecture through which it successfully executes business strategy.
- **Data architecture design** is important for creating a vision of interactions occurring between data systems, like for example if a data architect wants to implement data integration so that it will need interaction between two systems and by using data architecture the visionary model of data interaction during the process can be achieved.

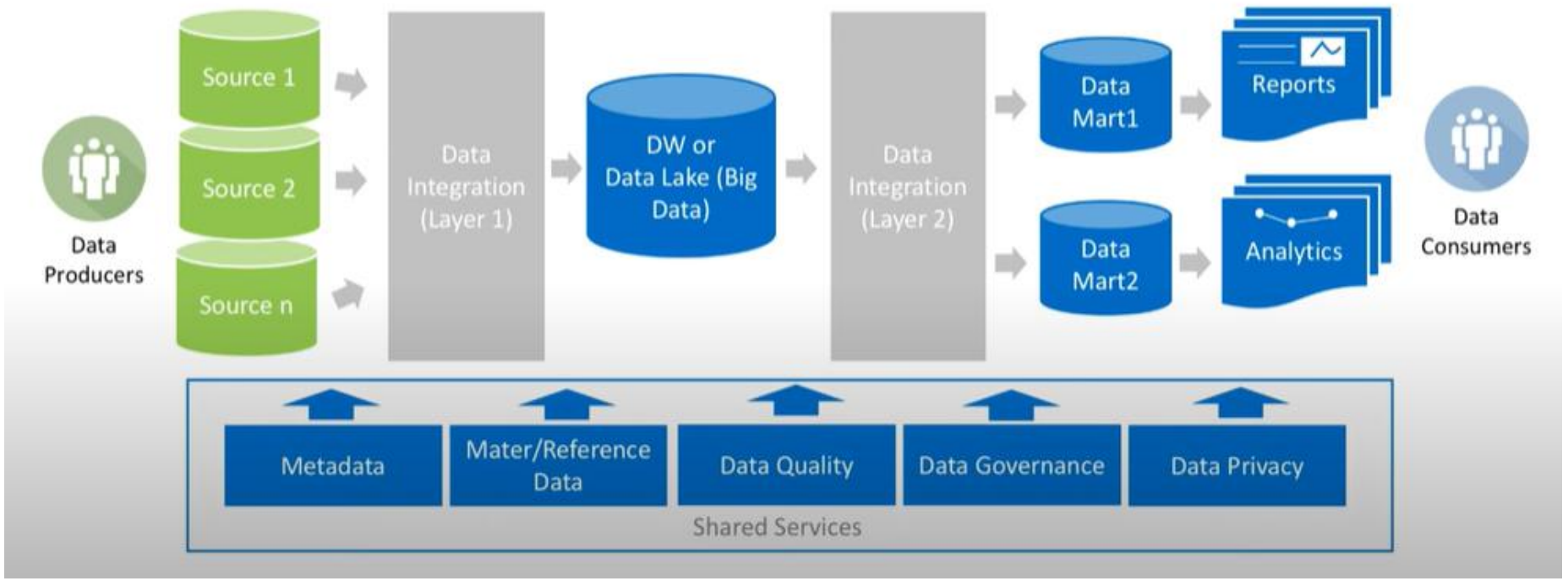
# Data Architecture

- Data architecture is composed of
  - models,
  - policies,
  - rules or standards

  that govern which data
  - is collected,
  - how it is stored,
  - arranged,
  - integrated,
  - put to use

  in data systems and organizations.
- Data is usually one of several architecture domains that form the pillars of an enterprise architecture or solution architecture

# Example: Generic Analytical Data Architecture



# Constraints and influences

- Various constraints and influences that will affect data architecture design are

❑Enterprise requirements

❑Technology drivers

❑Economics

❑Business Policies

❑Data processing needs.

# Enterprise requirements

- These will generally include such elements as
  - economical and effective system expansion,
  - acceptable performance levels (especially system access speed),
  - transaction reliability,
  - transparent data management.
- In addition, the conversion of raw data such as transaction records and image files into more useful information forms through such features as data warehouses is also a common organizational requirement,
- This enables managerial decision-making and other organizational processes. Architecture techniques
  - Split between managing transaction data and (master) reference data.
  - Splitting data capture systems from data retrieval systems.

# Technology drivers

- These are usually suggested by
  - completed data architecture
  - database architecture designs

- In addition, some technology drivers will derive from
  - existing organizational integration frameworks and standards,
  - organizational economics,
  - existing site resources (e.g. previously purchased software licensing).

# Economics

- These are also important factors that must be considered during the data architecture phase.
- It is possible that some solutions, while optimal in principle, may not be potential candidates due to cost.
- External factors such as
  - business cycle,
  - interest rates,
  - market conditions,
  - legal considerations could all affect decisions relevant to data architecture

# Business policies

- Business policies that also drive data architecture design include
  - internal organizational policies,
  - rules of regulatory bodies,
  - professional standards,
  - applicable governmental laws

that can vary by applicable agency.

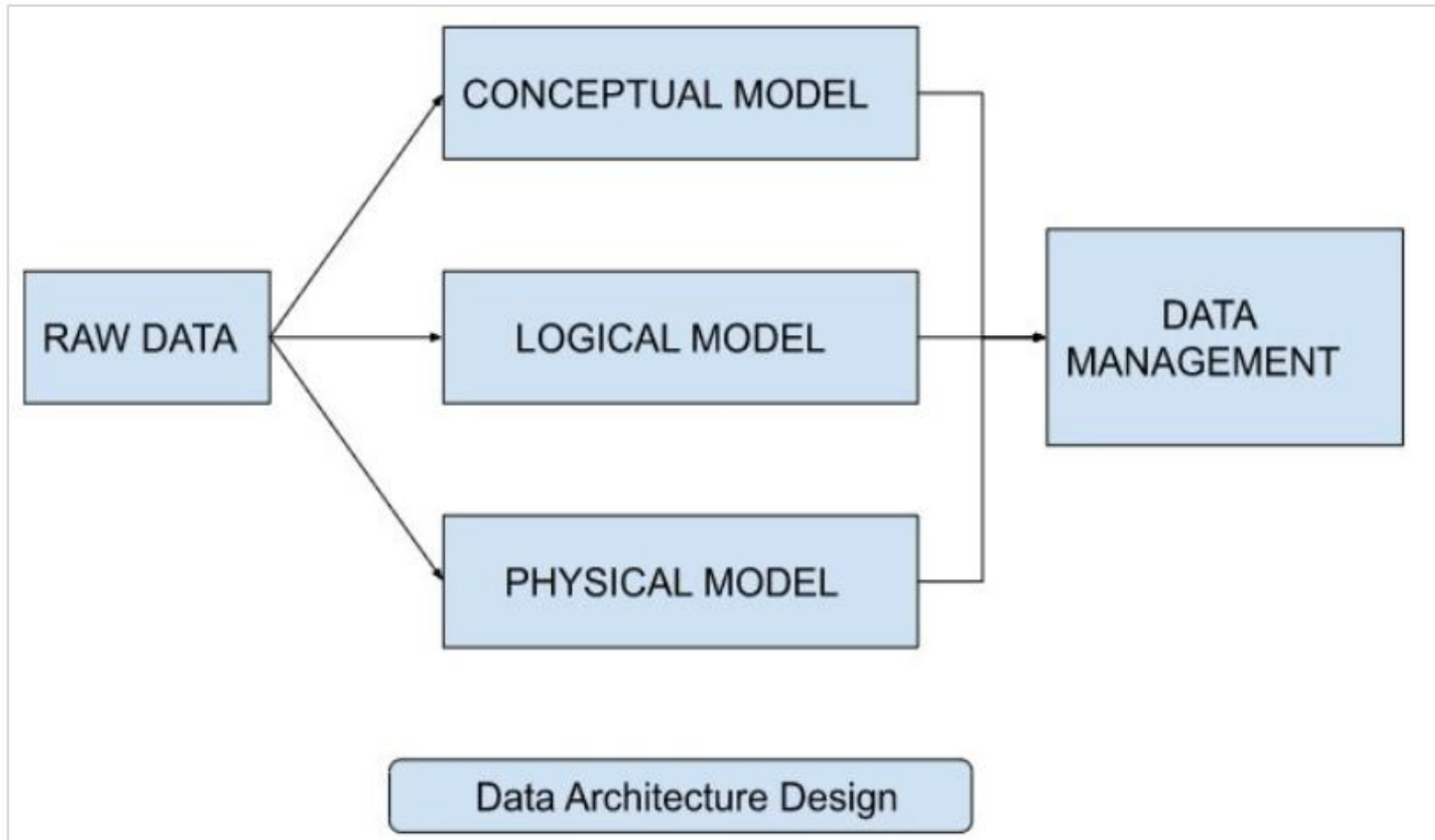
- These policies and rules will help describe how the enterprise wishes to process its data.

# Data processing needs

- These include
  - accurate and reproducible transactions performed in high volumes,
  - data warehousing for the support of management information systems (and potential data mining),
  - repetitive periodic reporting,
  - ad hoc reporting,
  - support various organizational initiatives as required (i.e. annual budgets, new product development).

# General Approach

- Data architecture also describes the type of data structures applied to manage data and it provides an easy way for data preprocessing.
- The data architecture is formed by dividing into three essential models and then are combined :
  - The Conceptual Model
  - The Logical Model
  - The Implementation/Physical  Model

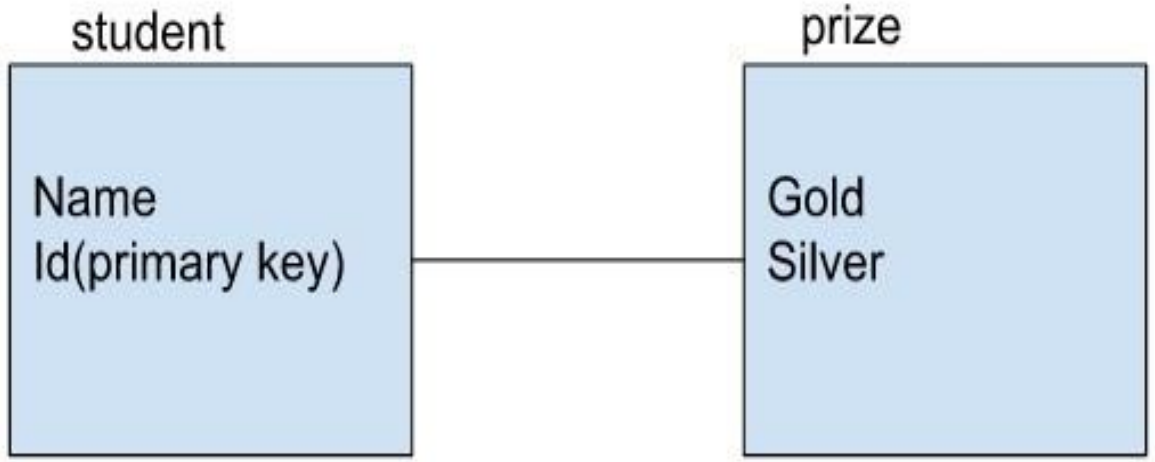

CONCEPTUAL MODEL
RAW DATA
LOGICAL MODEL
DATA MANAGEMENT
PHYSICAL MODEL
Data Architecture Design

# Conceptual model

- It is a model in which table entities and their attributes are related with the help of the ER model (Entity Relationship model). it also tells us what data is stored in the database along with the security and integrity information.
- Example
- Student and prize are two entities provided by their attributes.
- Rewarded is the relationship between the entities related

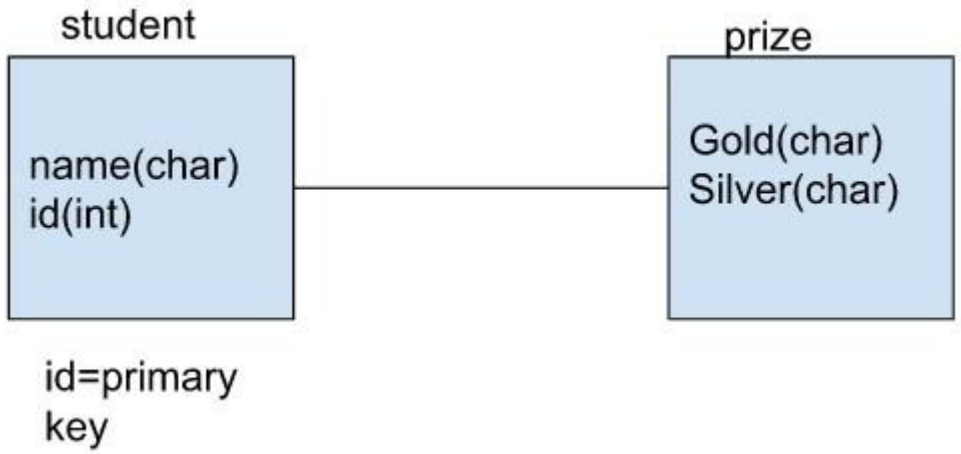
# Logical Model

- It is used to define problems in the form of logic. It adds more detail to the conceptual model i.e. presence of attributes for each entity, the presence of key and non-key attributes, and primary key-foreign key relationships are clearly defined.
- Example



# Implementation/Physical Model

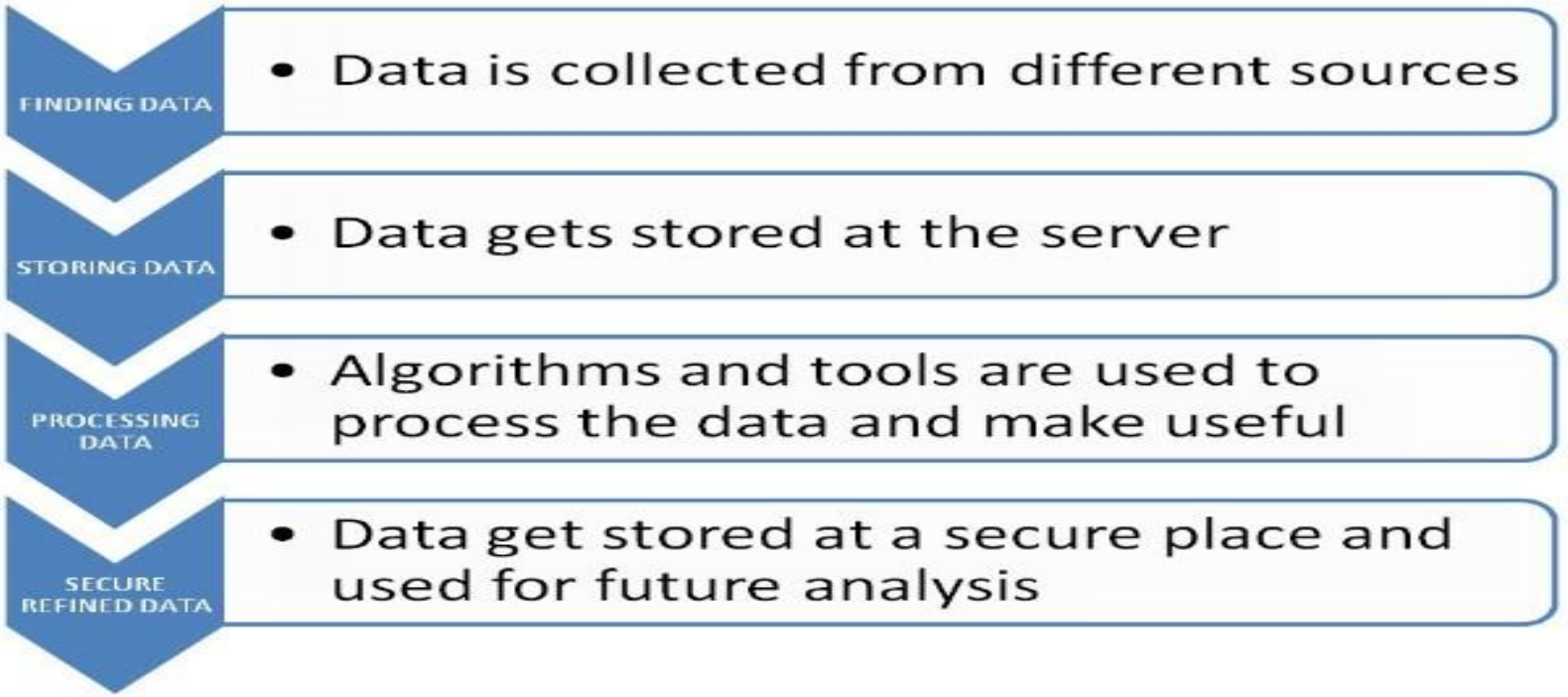
- While creating a physical model, we should be referring to entities and attributes as tables and columns respectively. Specific datatypes must be there to tell the type of data to be entered. More effort than a logical model is needed if any changes occur.
- Example



# Data Management

- Data management is the process of managing tasks like extracting data, storing data, transferring data, processing data, and then securing data with low-cost consumption.
- The main motive of data management is to manage and safeguard the people's and organization's data optimally so that they can easily create, access, delete, and update the data.
- Because data management is an essential process in every enterprise's growth, without which policies and decisions can't be made for business advancement. The better the data management the better productivity in business.
- Large volumes of data like big data are harder to manage traditionally so there must be the utilization of optimal technologies and tools for data management such as Hadoop, Scala, Tableau, AWS, etc. Which can be further used for big data analysis to achieve improvements in patterns.
- Data management can be achieved by training the employees necessary and maintenance by DBA, data analysts, and data architects.

# Data Management




It is the way of managing the data by performing tasks like finding, storing, processing, and then making the data secure.

# UNDERSTAND VARIOUS SOURCES OF THE DATA

## Two types of sources

Data can be generated from two types of sources

- Primary
- Secondary

- Sources of Primary Data
- The sources of generating primary data are
- Observation Method
- Survey Method
- Experimental Method

# Observation method

- The observation method involves human or mechanical observation of what people do or what events occur during a buying or consumption situation.

- "Information is collected by observing the process at work."

- The following are a few situations:
  - Service Stations-
    - Pose as a customer,
    - Go to a service station and observe.
  - To evaluate the effectiveness of the display of Dunlop Pillow Cushions-
  - In a departmental store, the observer notes:-
    - How many pass by;
    - How many stopped to look at the display;
    - How many decide to buy?
  - Super Market-
    - Which is the best location on the shelf? Hidden cameras are used.
    - To determine typical sales arrangements and find out sales enthusiasm shown by various salesmen-
  - Normally this is done by an investigator using a concealed tape recorder.

- Advantages of Observation Method
  - If the researcher observes and record events, it is not necessary to rely on the willingness and ability of respondents to report accurately.
  - The biasing effect of interviewers is either eliminated or reduced. Data collected by observation are, thus, more objective and generally more accurate.

- Disadvantages of Observation Method
  - The most limiting factor in the use of observation method is the inability to observe such things such as attitudes, motivations, customers/consumers' state of mind, their buying motives, and their images.
  - It also takes time for the investigator to wait for a particular action to take place.
  - Personal and intimate activities, such as watching television late at night, are more easily discussed with questionnaires than they are observed.
  - Cost is the final disadvantage of the observation method.
  - Under most circumstances, observational data are more expensive to obtain than other survey data.
  - The observer has to wait doing nothing, between events to be observed.
- The unproductive time is an increased cost

# Survey Method

There are mainly 4 methods by which we can collect data through the Survey Method

- Telephonic Interview
- Personal Interview
- Mail Interview
- Electronic Interview

# Telephonic Interview

- Best method for gathering quickly needed information.
- Responses are collected from the respondents by the researcher on telephone.

- Advantages of Telephonic Interview
  - It is very fast method of data collection.
  - It has the advantage over “Mail Questionnaire” of permitting the interviewer to talk to one or more persons and to clarifying his questions if they are not understood.
  - Response rate of telephone interviewing seems to be a little better than mail questionnaires
  - The quality of information is better
  - It is less costly method and there are less administration problems

- Disadvantages of Telephonic Interview
  - They cant handle interview which need props
  - It cant handle unstructured interview
  - It cant be used for those questions which requires long descriptive answers
  - Respondents cannot be observed
  - People are reluctant to disclose personal information on telephone
  - People who don’t have telephone facility cannot be approached

# Personal Interviewing

- It is the most versatile of the all methods. They are used when props are required along with the verbal response non-verbal responses can also be observed.

- Advantages of Personal Interview
  - The person interviewed can ask more questions and can supplement the interview with personal observation.
  - They are more flexible. Ord er of questions can be changed
  - In-depth research is possible.
  - Verification of data from other sources is possible.
  - The information obtained is very reliable and dependable and helps in establishing cause and effect relationship very early.
- Disadvantages of Personal Interview
  - It requires much more technical and administrative planning and supervision
  - It is more expensive
  - It is time consuming
  - The accuracy of data is influenced by the interviewer
  - A number of call banks may be required
  - Some people are not approachable

# Mail Survey

- Questionnaires are sent to the respondents; they fill it up and send it back.
- Advantages of Mail Survey
    - It can reach all types of people.
    - Response rate can be improved by offering certain incentives.
- Disadvantages of Mail Survey
    - It can not be used for unstructured study.
    - It is costly.
    - It requires established mailing list.
    - It is time consuming.
    - There is problem in case of complex questions.

# Electronic Interview

- Electronic interviewing is a process of recognizing and noting people, objects, occurrences rather than asking for information.
- For example-When you go to store, you notice which product people like to use.
- The Universal Product Code (UPC) is also a method of observing what people are buying.

- Advantages of Electronic Interview
  - There is no relying on willingness or ability of respondent.
  - The data is more accurate and objective.

- Disadvantages of Electronic Interview
  - Attitudes cannot be observed.
  - Those events which are of long duration cannot be observed.
  - There is observer bias. It is not purely objective.
  - If the respondents know that they are being observed, their response can be biased.
  - It is a costly method.

# Experimental Method

- There are several experimental designs that are used in carrying out an experiment.
- However, Market researchers have used 4 experimental designs most frequently.

- These are
  - CRD - Completely Randomized Design
  - RBD - Randomized Block Design
  - LSD - Latin Square Design
  - FD - Factorial Designs

# CRD

- Suppose you are conducting an experiment to test the effect of different fertilizers on plant growth.
- You randomly assign individual plants to different fertilizer treatments, ensuring that each plant has an equal chance of receiving any particular fertilizer.
- The growth of the plants is then measured and analyzed to determine if there are significant differences between the fertilizer treatments

# RBD

- A Randomized Block Design (RBD) is an experimental design that builds upon the principles of a Completely Randomized Design (CRD) but incorporates the concept of blocking to increase precision and account for known sources of variability.
- In an RBD, experimental units are first divided into homogeneous groups or blocks based on a known source of variability, and then randomization is applied within each block.
- Let's consider an example to illustrate the concept of a Randomized Block Design:
- **Experiment:** Suppose a researcher is investigating the effect of three different types of fertilizers (A, B, and C) on the yield of a specific crop. However, the field where the experiment is conducted has variations in soil fertility across different sections. To control for the variability in soil fertility, the researcher decides to use a Randomized Block Design.

# Randomized block design

**Steps in Randomized Block Design:**

1. **Blocking:**

   The field is divided into blocks based on soil fertility levels. Each block represents a group of experimental units (plots) with similar soil conditions.

2. **Randomization within Blocks:**

   Within each block, the three types of fertilizers (A, B, and C) are randomly assigned to different plots. This randomization helps ensure that the effects of soil fertility are evenly distributed among the fertilizer treatments.

3. **Data Collection:**

   The yield of the crops is measured for each plot within the experiment.

# Latin Square Design

Example:

- Consider a Latin Square Design with two factors: A and B. If there are three levels for each factor (A1, A2, A3, and B1, B2, B3), the design would look like the following:

- A1 B1 A2 B2 A3 B3
- A2 B3 A3 B1 A1 B2
- A3 B2 A1 B3 A2 B1
- B1 A1 B2 A2 B3 A3
- B2 A3 B3 A1 B1 A2
- B3 A2 B1 A3 B2 A1

- In this design, each level of factor A appears once in each row and each level of factor B appears once in each column.

# Factorial Designs

Consider an experiment investigating the effects of both the type of training method (A: traditional, B: online) and the duration of training (X: short, Y: long) on employee performance. The factorial design could be represented as follows:

- Factor A: Training Method (Levels: Traditional, Online)
- Factor B: Training Duration (Levels: Short, Long)
- The resulting experimental conditions (cells) would include all possible combinations of training methods and durations:

1. Traditional Training (A) + Short Duration (B)
2. Traditional Training (A) + Long Duration (B)
3. Online Training (A) + Short Duration (B)
4. Online Training (A) + Long Duration (B)

# SOURCES OF SECONDARY DATA

- The secondary data can be obtained through

1. Internal Sources - These are within the organization
2. External Sources - These are outside the organization

# Internal Sources of Data

- obtained with less time, effort, and money than the external secondary data.
- more pertinent to the situation since they are from within the organization.

The internal sources include

**Accounting resources-** This gives so much information which can be used by the marketing researcher. They give information about internal factors.

**Sales Force Report-** It gives information about the sale of a product. The information provided is of outside the organization.

**Internal Experts-** These are people who are heading the various departments. They can give an idea of how a particular thing is working

**Miscellaneous Reports-** These are what information you are getting from operational reports. If the data available within the organization are unsuitable or inadequate, the marketer should extend the search to external secondary data sources.

# External Sources of Data

- External Sources are sources that are outside the company in a larger environment.
- Collection of external data is more difficult because the data have much greater variety and the sources are much more numerous

**External data can be divided into the following classes.**

Government Publications- Government sources provide an extremely rich pool of data for the researchers. Data are available free of cost on internet websites.

There are number of government agencies generating data. These are:

**The Registrar General of India**

o It is an office that generates demographic data.

o It includes details of gender, age, occupation etc.

**Central Statistical Organization**

o publishes the national accounts statistics

o contains estimates of national income for several years, growth rate, and rate of major economic activities.

o Annual survey of Industries is also published.

o gives information about the total number of workers employed, production units, material used, and value added by the manufacturer.

# External Sources of Data

**The Director General of Commercial Intelligence**

O gives information about foreign trade i.e. import and export.

O These figures are provided region-wise and country-wise.

**Ministry of Commerce and Industries**

O provides information on the wholesale price index.

O indices related to several sectors like food, fuel, power, food grains etc.

O also generates All India Consumer Price Index numbers for industrial workers, urban, non-manual employees, and cultural laborers.

**Planning Commission**

O provides the basic statistics of the Indian Economy..

**Reserve Bank of India**

o provides information on Banking Savings and investment.

o also prepares currency and finance reports.

**National Sample Survey**

o provides social, economic, demographic, industrial and agricultural statistics.

**Labor Bureau**

o provides information on skilled, unskilled, white collared jobs etc

**Department of Economic Affairs**

o conducts economic survey and it also generates information on income, consumption, expenditure, investment, savings and foreign trade.

**State Statistical Abstract**

o gives information on various types of activities related to the state like - commercial activities, education, occupation etc.

**Non Government Publications**

o includes publications of various industrial and trade associations, such as The Indian Cotton Mill Association Various chambers of commerce

**The Bombay Stock Exchange**

o publishes a directory containing financial accounts, key profitability and other relevant matter

**Various Associations of Press Media.**

**Export Promotion Council.**

**Confederation of Indian Industries ( CII )**

**Small Industries Development Board of India**

**International Organization** includes

o The International Labour Organization (ILO) :

publishes data on the total and active population, employment, unemployment, wages and consumer prices

o The Organization for Economic Co-operation and development (OECD)

publishes data on foreign trade, industry, food, transport, and science and technology.

o The International Monetary Fund (IMA)

publishes reports on national and international foreign exchange regulations.

# What is Data Management?

- Data management is collecting, organizing, transforming, and storing data for an organization’s data analysis operations.
- The process only ensures clean, well-managed data for various purposes, like gaining insights and planning marketing campaigns.
- When data is easy to find, visualize, and tweak, it helps organizations gain actionable insights and make informed decision-making.

# Technologies and Tools for Data Management

- **Relational Databases and Data Warehouses**

To work in the data arena, you must understand relational database management systems. This software system manages relational databases, encompassing everything from organizing data with rows and columns to supporting structured query language (SQL) for data manipulation and retrieval. It is intended for maintaining data standards to the ideal extent.

- **Data Management Platforms and Software**

Data management tools help professionals create and maintain accurate and consistent data within the organization. These tools are designed to facilitate data profiling, matching, quality management, merging, and governance capabilities.

- **Data Integration and ETL (Extract, Transform, Load) tools**


Data integration tools allow organizations to combine data from multiple sources, making management more effective. These tools come with capabilities for data mapping, cleansing, and transformation. Moreover, ETL tools streamline extracting and changing data from various sources into a consistent format.

- **Data Governance and Metadata Management Solutions**

Organizations employ data governance tools to enhance data governance processes, policies, and standards. These tools strengthen the management by assisting in metadata management, classification, data lineage, and compliance with data regulations.

# Data management tools

From sources across the web


Informatica


Google

Profisee

Ataccama


Microsoft Azure


Amazon Web Services


Dell Boomi


Oracle


Tableau


Talend


SAS


Stitch data


Looker


SAP NetWeaver Master D...


Fivetran


IBM InfoSphere Master d...


Master data management

Microsoft Power BI

# Benefits of Data Management

| | |
|---|---|
| Eradicate data redundancy | Data stored in multiple locations can become inconsistent and is often saved in different formats. Data management systems aim to provide a single, accurate source of data for analysis. |
| Improve data sharing | Data management systems help ensure consistent data is shared securely among all users. Cloud-based data management systems facilitate sharing large amounts of data. |
| Tighten data privacy and security | Data management systems help the organization focus its security efforts on the most critical data assets. They help the company understand where sensitive data is stored, whether it should be encrypted and which users and groups should have access. |
| Improve backups and recovery | Data management helps companies create and manage backup approaches tailored to the data. It helps set appropriate backup frequencies based on the importance of the data and minimize recovery time in the event of a problem. |
| Real-time data consistency | Modern data management systems help businesses capture data streams and share information in real time. |

# Data Quality

- Data mining applications are often applied to data that was collected for another purpose, or for future, but unspecified applications.
- For that reason data mining cannot usually take advantage of the significant benefits of "addressing quality issues at the source."
- In contrast, much of statistics deals with the design of experiments or surveys that achieve a prespecified level of data quality.
- Because preventing data quality problems is typically not an option, data mining focuses on

1. Detection and correction (called data cleaning ) of data quality problems
2. Use of algorithms that can tolerate poor data quality.

# Measurement and Data Collection Errors

- It is unrealistic to expect that data will be perfect.
- There may be problems due to human error, limitations of measuring devices, or flaws in the data collection process.
- Values or even entire data objects may be missing.
- In other cases, there may be spurious or duplicate objects; i.e., multiple data objects that all correspond to a single "real" object.
- For example, there might be two different records for a person who has recently lived at two different addresses.
- Even if all the data is present and "looks fine," there may be inconsistencies-a person has a height of 2 meters but weighs only 2 kilograms.

# Measurement and Data Collection Errors

**Measurement error**

o Refers to any problem resulting from the measurement process.

o A common problem is that the value recorded differs from the true value to some extent.

o For continuous attributes, the numerical difference of the measured and true value is called the error.

**Data collection error**

o refers to errors such as omitting data objects or attribute values, or inappropriately including a data object.

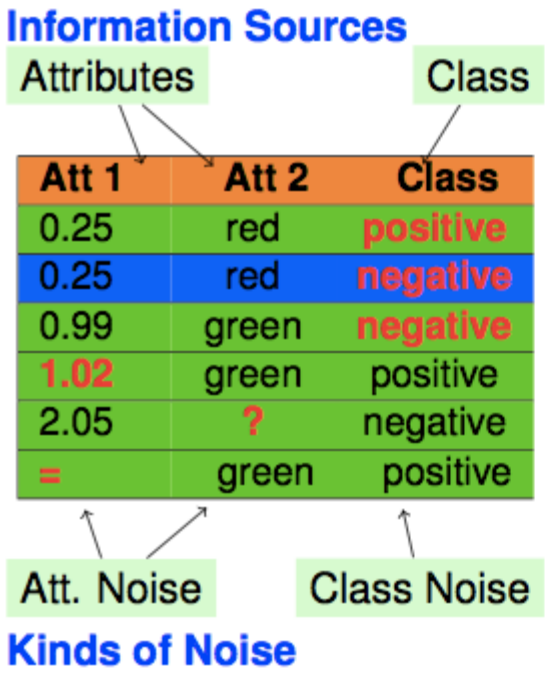
o For example, a study of animals of a certain species might include animals of a related species that are similar in appearance to the species of interest.

Both measurement errors and data collection errors can be either systematic or random.

# Examples of data quality problems:

- Noise
- Outliers
- Missing values
- Duplicate data

# Noise Example




**Class Noise**

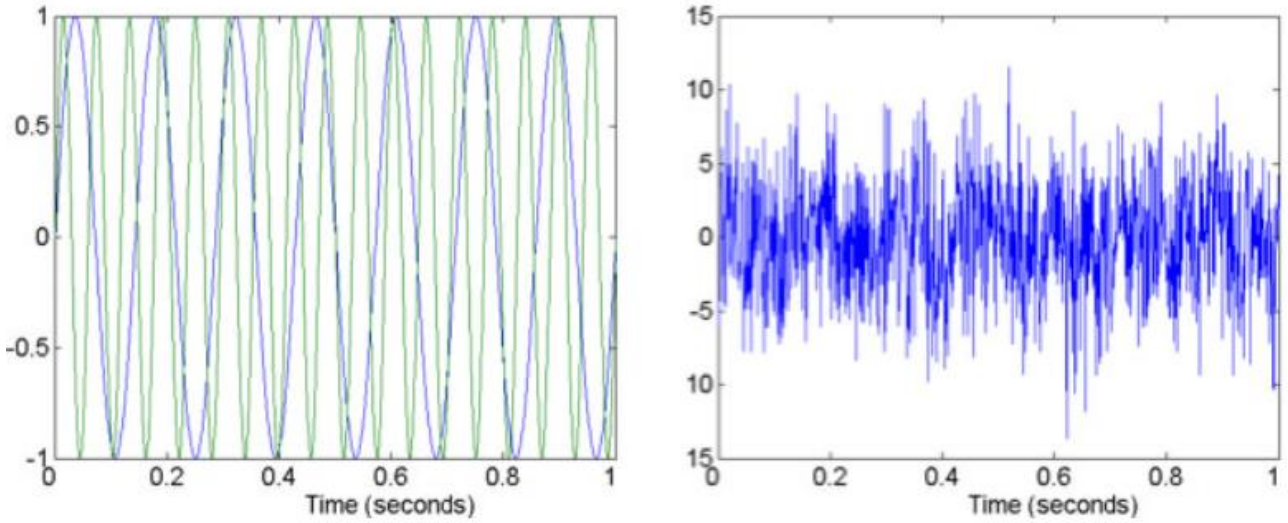
- Contradictory examples
- Mislabeled examples

**Attribute Noise**

- Erroneous values
- Missing values
- Don't care values

# Noise

- For objects, noise is considered an extraneous object.
- For attributes, noise refers to modification of original values.
- Examples: distortion of a person's voice when talking on a poor phone and "snow" on television screen
- We can talk about **signal to noise ratio.** Left image of 2 sine waves has low or zero SNR; the right image are the two waves combined with noise and has high SNR



# Outliers

- Outliers are either

1. data objects that have characteristics that are different from most of the other data objects in the data set, or
2. values of an attribute that are unusual with respect to the typical values for that attribute.

- Outliers can be legitimate data objects or values.
- Unlike noise, outliers may sometimes be of interest.

**Origins of noise**

- outliers -- values seemingly out of the normal range of data
- duplicate records -- good database design should minimize this (use DISTINCT on SQL retrievals)
- incorrect attribute values -- again good db design and integrity constraints should minimize this
- numeric only, deal with rogue strings or characters where numbers should be.
- null handling for attributes (nulls=missing values)

**Outliers--be careful!**

- Outliers are data objects with characteristics that are considerably different than most of the other data objects in the data set

  Case 1: Outliers are noise that interferes with data analysis

  Case 2: Recognizing outliers can be the goal of our analysis

- Credit card fraud
- Intrusion detection

# Missing values

| ID | Color | Weight | Broken | Class |
|---|---|---|---|---|
| 1 | Black | 80 | Yes | 1 |
| 2 | Yellow | 100 | No | 2 |
| 3 | Yellow | 120 | Yes | 2 |
| 4 | Blue | 90 | No | 2 |
| 5 | Blue | 85 | No | 2 |
| **6** | **?** | 60 | No | 1 |
| **7** | Yellow | 100 | **?** | 2 |
| **8** | **?** | 40 | **?** | 1 |

# Missing values

- It is not unusual for an object to be missing one or more attribute values.
- In some cases, the information was not collected; e.g., some people decline to give their age or weight.
- In other cases, some attributes are not applicable to all objects; e.g., often, forms have conditional parts that are filled out only when a person answers a previous question in a certain way, but for simplicity, all fields are stored.
- Missing values should be taken into account during the data analysis

# Missing values

Strategies for dealing with missing data, each of which may be appropriate in certain circumstances:

Eliminate Data Objects or Attributes

o A simple and effective strategy is to eliminate objects with missing values.

o However, even a partially specified data object contains some information, and if many objects have missing values, then a reliable analysis can be difficult or impossible.

o However, if a data set has only a few objects that have missing values, then it may be convenient to omit them.

o A related strategy is to eliminate attributes that have missing values.

o This should be done with caution, however, since the eliminated attributes may be the ones that are critical to the analysis.

# Inconsistent Values

- Data can contain inconsistent values.
- Consider an address field, where both a zip code and city are listed, but the specified zip code area is not contained in that city.
- It may be that the individual entering this information transposed two digits, or perhaps a digit was misread when the information was scanned from a handwritten form.
- it is important to detect and, if possible, correct such problems.
- Some types of inconsistences are easy to detect. For instance, a person's height should not be negative.
- In some cases, it can be necessary to consult an external source of information.
- For example, when an insurance company processes claims for reimbursement, it checks the names and addresses on the reimbursement forms against a database of its customers.
- Once an inconsistency has been detected, it is sometimes possible to correct the data.
- A product code may have "check" digits, or it may be possible to double-check a product code against a list of known product codes, and then correct the code if it is incorrect, but close to a known code.
- The correction of an inconsistency requires additional or redundant information.

# Duplicate Data

- Data set may include data objects that are duplicates, or almost duplicates of one another
- A major issue when merging data from multiple, heterogeneous sources
- Examples: Same person with multiple email addresses
- When should duplicate data not be removed?

# Duplicate Data

- A data set may include data objects that are duplicates, or almost duplicates, of one another.
- Many people receive duplicate mailings because they appear in a database multiple times under slightly different names.
- To detect and eliminate such duplicates, two main issues must be addressed.

o First, if there are two objects that actually represent a single object, then the values of corresponding attributes may differ, and these inconsistent values must be resolved.

o Second, care needs to be taken to avoid accidentally combining data objects that are similar, but not duplicates, such as two distinct people with identical names.

- The term deduplication is often used to refer to the process of dealing with these issues.
- In some cases, two or more objects are identical with respect to the attributes measured by the database, but they still represent different objects.
- Here, the duplicates are legitimate, but may still cause problems for some algorithms if the possibility of identical objects is not specifically accounted for in their design.

# ISSUES RELATED TO APPLICATIONS

- Data quality issues can also be considered from an application viewpoint.
- There are many issues that are specific to particular applications and fields.

**Timeliness**

- Some data starts to age as soon as it has been collected.
- In particular, if the data provides a snapshot of some ongoing phenomenon or process, such as the purchasing behavior of customers or Web browsing patterns, then this snapshot represents reality for only a limited time.
- If the data is out of date, then so are the models and patterns that are based on it.

**Relevance**

- The available data must contain the information necessary for the application.
- Consider the task of building a model that predicts the accident rate for drivers.
- If information about the age and gender of the driver is omitted, then it is likely that the model will have limited accuracy unless this information is indirectly available through other attributes.
- Making sure that the objects in a data set are relevant is also challenging.

**Sampling bias**

- which occurs when a sample does not contain different types of objects in proportion to their actual occurrence in the population.
- For example, survey data describes only those who respond to the survey.
- Because the results of a data analysis can reflect only the data that is present, sampling bias will typically result in an erroneous analysis.

**Knowledge about the Data**

- Ideally, data sets are accompanied by documentation that describes different aspects of the data.
- The quality of this documentation can either aid or hinder the subsequent analysis.
- For example, if the documentation identifies several attributes as being strongly related, these attributes are likely to provide highly redundant information, and we may decide to keep just one.
- If the documentation is poor, however, and fails to tell us, for example, that the missing values for a particular field are indicated with a -9999, then our analysis of the data may be faulty.
- Other important characteristics are the precision of the data, the type of features (nominal, ordinal, interval, ratio), the scale of measurement (e.g., meters or feet for length), and the origin of the data

# Data Preprocessing

# Data Preprocessing

- Data Preprocessing - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation and Data Discretization

# Data Quality: Why Preprocess the Data?

- Measures for data quality: A multidimensional view
  - Accuracy: correct or wrong, accurate or not
  - Completeness: not recorded, unavailable, …
  - Consistency: some modified but some not, dangling, …
  - Timeliness: timely update?
  - Believability: how trustworthy the data are correct?
  - Interpretability: how easily the data can be understood?

# Major Tasks in Data Preprocessing

- **Data cleaning**
  - Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies
- **Data integration**
  - Integration of multiple databases, data cubes, or files
- **Data reduction**
  - Dimensionality reduction
  - Numerosity reduction
  - Data compression
- **Data transformation and data discretization**
  - Normalization
  - Concept hierarchy generation

# Forms of Data Preprocessing

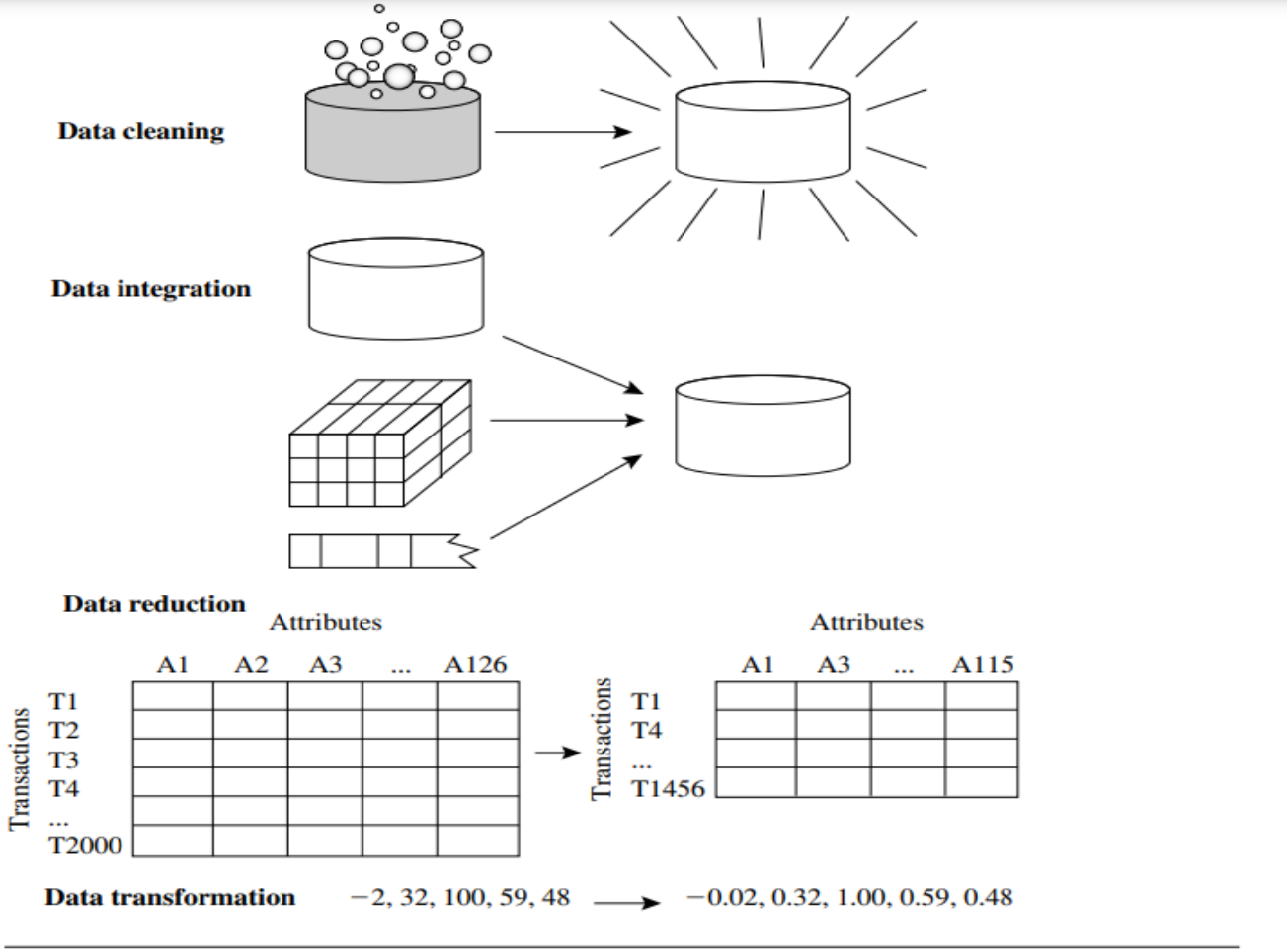



**Figure 3.1** Forms of data preprocessing.

# Data Preprocessing

- Data Preprocessing: An Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning 
- Data Integration
- Data Reduction
- Data Transformation and Data Discretization

# Data Cleaning

- Data in the Real World Is Dirty:

Lots of potentially incorrect data, e.g., instrument faulty, human or computer error, transmission error

- incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
  - e.g., *Occupation*=“ ” (missing data)
- noisy: containing noise, errors, or outliers
  - e.g., *Salary*=“−10” (an error)
- inconsistent: containing discrepancies in codes or names, e.g.,
  - *Age*=“42”, *Birthday*=“03/07/2010”
  - Was rating “1, 2, 3”, now rating “A, B, C”
  - discrepancy between duplicate records
- Intentional (e.g., *disguised missing* data)
  - Jan. 1 as everyone’s birthday?

# Incomplete (Missing) Data

- Data is not always available
  - E.g., many tuples have no recorded value for several attributes, such as customer income in sales data
- Missing data may be due to
  - equipment malfunction
  - inconsistent with other recorded data and thus deleted
  - data not entered due to misunderstanding
  - certain data may not be considered important at the time of entry
  - not register history or changes of the data
- Missing data may need to be inferred

# How to Handle Missing Data?

- Ignore the tuple: usually done when the class label is missing (when doing classification)—not effective when the % of missing values per attribute varies considerably
- Fill in the missing value manually: tedious + infeasible?
- Fill in it automatically with
  - a global constant: e.g., “unknown”, a new class?!
  - the attribute means
  - the attribute mean for all samples belonging to the same class: smarter
  - the most probable value: inference-based such as

    Bayesian formula or decision tree

# Noisy Data

- Noise: random error or variance in a measured variable
- Incorrect attribute values may be due to
  - Faulty data collection instruments
  - Data entry problems
  - Data transmission problems
  - Technology limitation
  - Inconsistency in the naming convention
- Other data problems that require data cleaning
  - duplicate records
  - incomplete data
  - inconsistent data

# How to Handle Noisy Data?

- Binning
  - first sort data and partition it into (equal-frequency) bins
  - then one can smooth by bin means,
    smooth by bin median,
    smooth by bin boundaries, etc.
- Regression
  - smooth by fitting the data into regression functions
- Clustering
  - detect and remove outliers
- Combined computer and human inspection
  - detect suspicious values and check by humans (e.g., deal with possible outliers)

- In smoothing by bin means, each value in a bin is replaced by the mean value of the bin. For example, the mean of the values 4, 8, and 15 in Bin 1 is 9. Therefore, each original value in this bin is replaced by the value 9.

- Similarly, smoothing by bin medians can be employed, in which each bin value is replaced by the bin median.

- In smoothing by bin boundaries, the minimum and maximum values in a given bin are identified as the bin boundaries. Each bin value is then replaced by the closest boundary value. In general, the larger the width, the greater the effect of the smoothing.

- Alternatively, bins may be equal in width, where the interval range of values in each bin is constant.

# Binning Methods for Data Smoothing

**Sorted data for *price* (in dollars):** 4, 8, 15, 21, 21, 24, 25, 28, 34

**Partition into (equal-frequency) bins:**

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

**Smoothing by bin means:**

Bin 1: 9, 9, 9
Bin 2: 22, 22, 22
Bin 3: 29, 29, 29

**Smoothing by bin boundaries:**

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

**Figure 3.2** Binning methods for data smoothing.

# Clustering Method for Outliers Identification

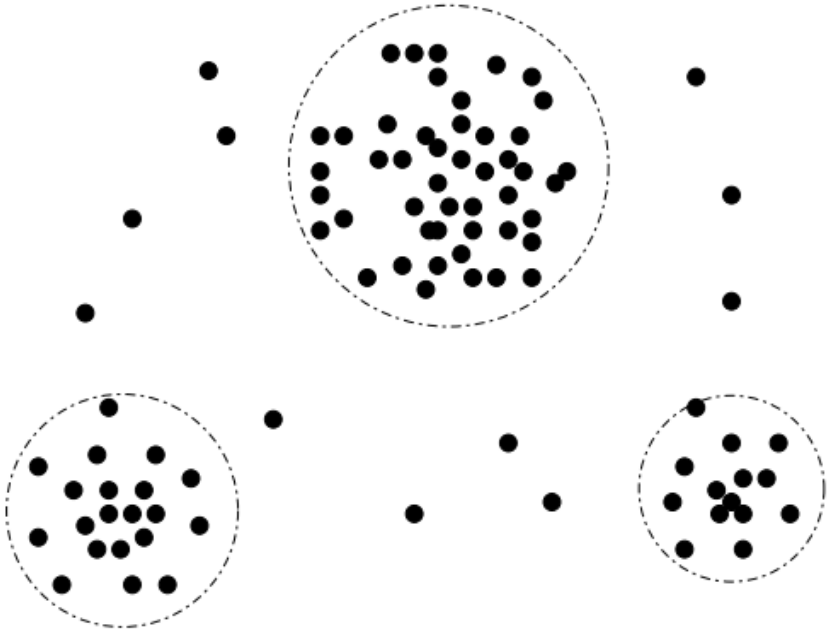


**Figure 3.3** A 2-D customer data plot with respect to customer locations in a city, showing three data clusters. Outliers may be detected as values that fall outside of the cluster sets.

# Data Preprocessing

- Data Preprocessing: An Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction 
- Data Transformation
- Summary



15

# Data Reduction Strategies

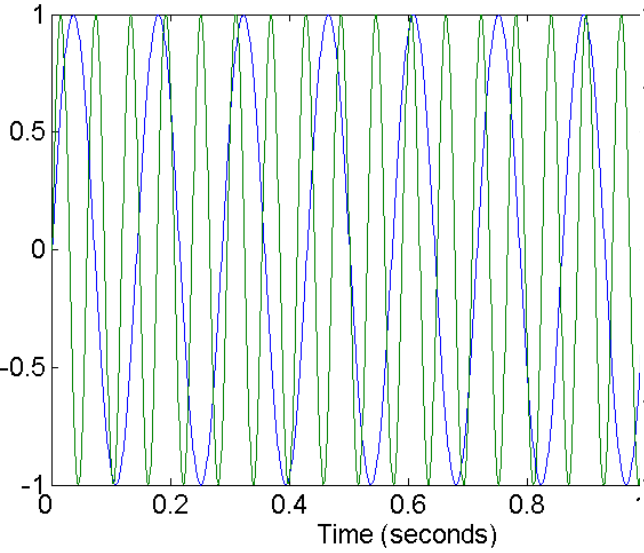
- **Data reduction**: Obtain a reduced representation of the data set that is much smaller in volume but yet produces the same (or almost the same) analytical results
- Why data reduction? — A database/data warehouse may store terabytes of data. Complex data analysis may take a very long time to run on the complete data set.
- Data reduction strategies
  - Dimensionality reduction, e.g., removing unimportant attributes
    - Wavelet transforms
    - Principal Components Analysis (PCA)
    - Feature subset selection, feature creation
  - Numerosity reduction (some simply call it: Data Reduction)
    - Regression and Log-Linear Models
    - Histograms, clustering, sampling
    - Data cube aggregation
  - Data compression

# Data Reduction : Dimensionality Reduction

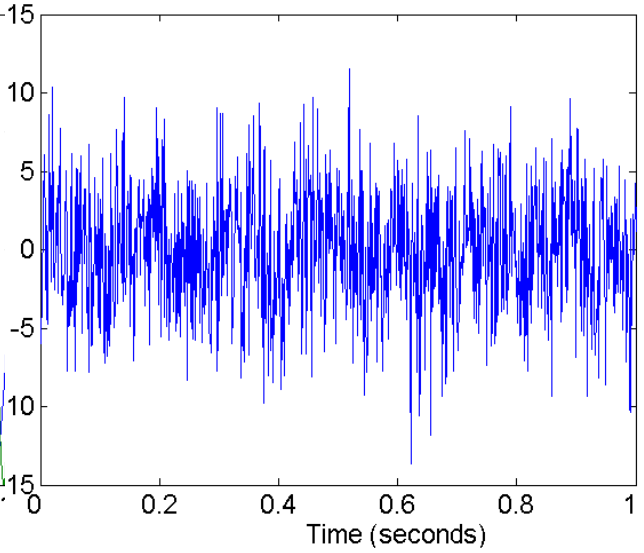
- Curse of dimensionality
  - When dimensionality increases, data becomes increasingly sparse
  - Density and distance between points, which is critical to clustering, and outlier analysis, becomes less meaningful
  - The possible combinations of subspaces will grow exponentially
- Dimensionality reduction
  - Avoid the curse of dimensionality
  - Help eliminate irrelevant features and reduce noise
  - Reduce time and space required in data mining
  - Allow easier visualization
- Dimensionality reduction techniques
  - Wavelet transforms
  - Principal Component Analysis
  - Supervised and nonlinear techniques (e.g., feature selection)

# Mapping Data to a New Space

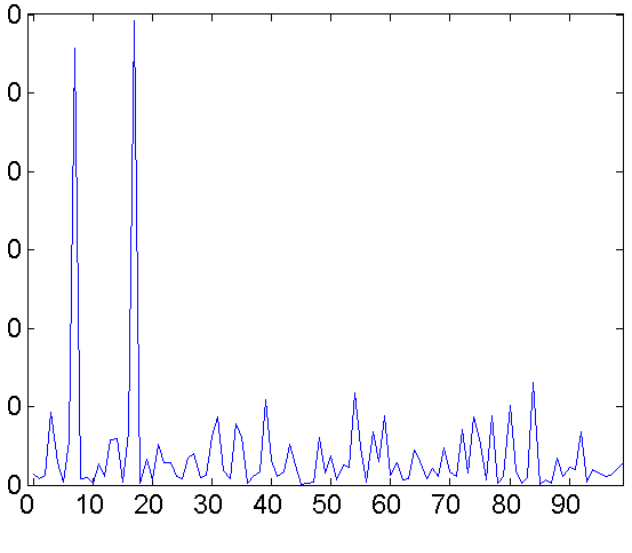
- Fourier transform
- Wavelet transform




Two Sine Waves



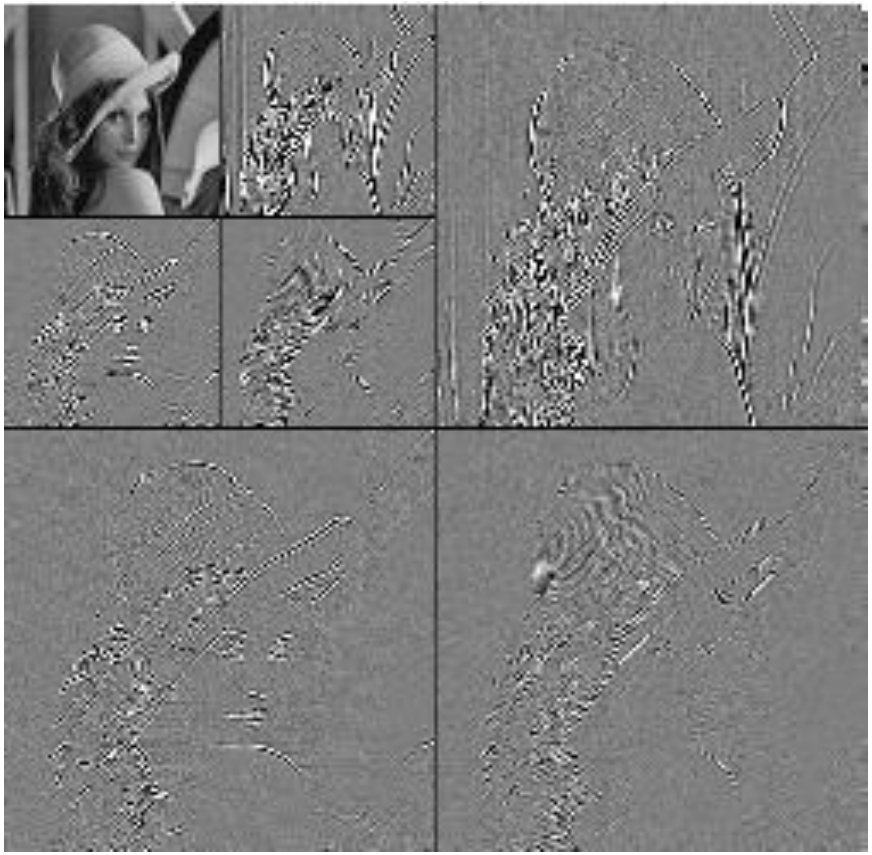

Two Sine Waves + Noise



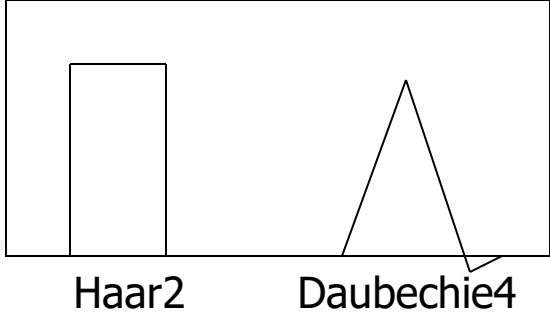

Frequency

# What Is Wavelet Transform?

- Decomposes a signal into different frequency subbands
  - Applicable to n-dimensional signals
- Data are transformed to preserve relative distance between objects at different levels of resolution
- Allow natural clusters to become more distinguishable
- Used for image compression



# Wavelet Transformation



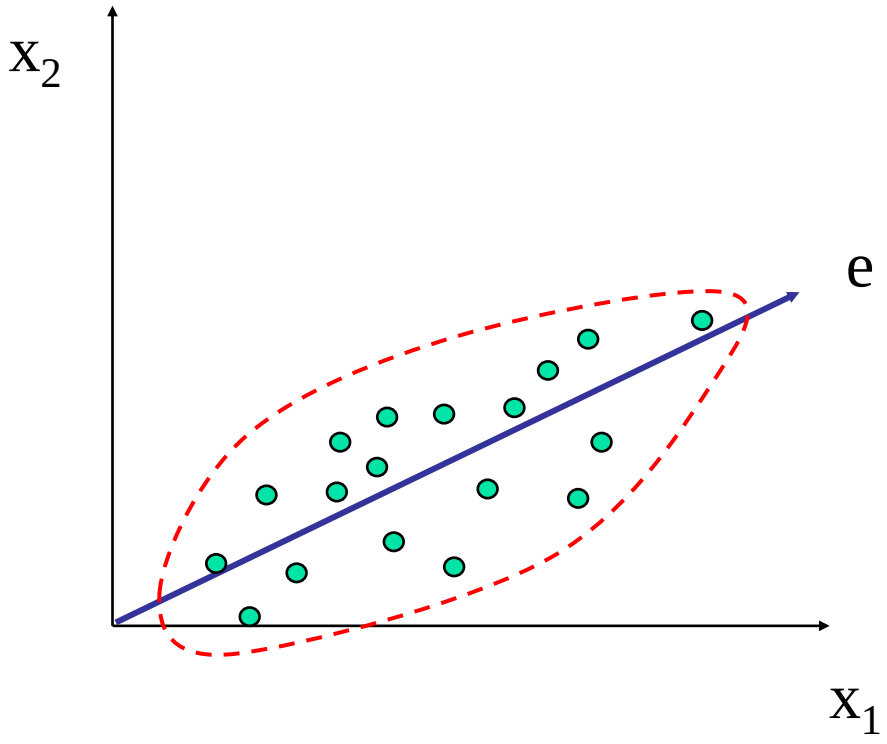

- Discrete wavelet transform (DWT) for linear signal processing, multi-resolution analysis
- Compressed approximation: store only a small fraction of the strongest of the wavelet coefficients
- Similar to discrete Fourier transform (DFT), but better lossy compression, localized in space
- Method:
  - Length, L, must be an integer power of 2 (padding with 0's, when necessary)
  - Each transform has 2 functions: smoothing, difference
  - Applies to pairs of data, resulting in two sets of data of length L/2
  - Applies two functions recursively, until it reaches the desired length

# Why Wavelet Transform?

- Use hat-shape filters
  - Emphasize the region where points cluster
  - Suppress weaker information in their boundaries
- Effective removal of outliers
  - Insensitive to noise, insensitive to input order
- Multi-resolution
  - Detect arbitrarily shaped clusters at different scales
- Efficient
  - Complexity O(N)
- Only applicable to low-dimensional data

# Principal Component Analysis (PCA)

- Find a projection that captures the largest amount of variation in data
- The original data are projected onto a much smaller space, resulting in dimensionality reduction. We find the eigenvectors of the covariance matrix, and these eigenvectors define the new space



# Principal Component Analysis (Steps)

- Given $N$ data vectors from $n$-dimensions, find $k \leq n$ orthogonal vectors (*principal components*) that can be best used to represent data
  - Normalize input data: Each attribute falls within the same range
  - Compute $k$ orthonormal (unit) vectors, i.e., *principal components*
  - Each input data (vector) is a linear combination of the $k$ principal component vectors
  - The principal components are sorted in order of decreasing "significance" or strength
  - Since the components are sorted, the size of the data can be reduced by eliminating the *weak components*, i.e., those with low variance (i.e., using the strongest principal components, it is possible to reconstruct a good approximation of the original data)
- Works for numeric data only

# Attribute Subset Selection

- Another way to reduce the dimensionality of data
- Redundant attributes
  - Duplicate much or all of the information contained in one or more other attributes
  - E.g., the purchase price of a product and the amount of sales tax paid
- Irrelevant attributes
  - Contain no information that is useful for the data mining task at hand
  - E.g., students' ID is often irrelevant to the task of predicting students' GPA

# Heuristic Search in Attribute Selection

- There are $2^d$ possible attribute combinations of $d$ attributes
- Typical heuristic attribute selection methods:
  - Best single attribute under the attribute independence assumption: choose by significance tests
  - Best step-wise feature selection:
    - The best single attribute is picked first
    - Then next best attribute condition to the first, ...
  - Step-wise attribute elimination:
    - Repeatedly eliminate the worst attribute
  - Best combined attribute selection and elimination
  - Optimal branch and bound:
    - Use attribute elimination and backtracking

# Attribute Creation (Feature Generation)

- Create new attributes (features) that can capture the important information in a data set more effectively than the original ones
- Three general methodologies
  - Attribute extraction
    - Domain-specific
  - Mapping data to new space (see: data reduction)
    - E.g., Fourier transformation, wavelet transformation, manifold approaches (not covered)
  - Attribute construction
    - Combining features (see: discriminative frequent patterns in Chapter 7)
    - Data discretization

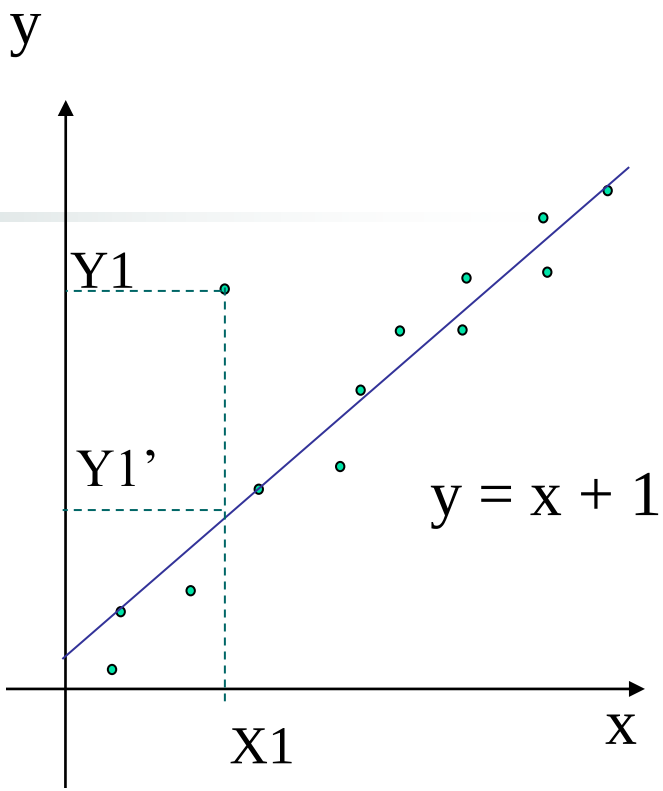
# 2. Data Reduction: Numerosity Reduction

- Reduce data volume by choosing alternative, *smaller forms* of data representation
- **Parametric methods** (e.g., regression)
  - Assume the data fits some model, estimate model parameters, store only the parameters, and discard the data (except possible outliers)
  - Ex.: Log-linear models—obtain value at a point in *m*-D space as the product on appropriate marginal subspaces
- **Non-parametric** methods
  - Do not assume models
  - Major families: histograms, clustering, sampling, …

# Parametric Data Reduction: Regression and Log-Linear Models

- **Linear regression**
  - Data modeled to fit a straight line
  - Often uses the least-square method to fit the line
- **Multiple regression**
  - Allows a response variable Y to be modeled as a linear function of the multidimensional feature vector
- **Log-linear model**
  - Approximates discrete multidimensional probability distributions

# Regression Analysis



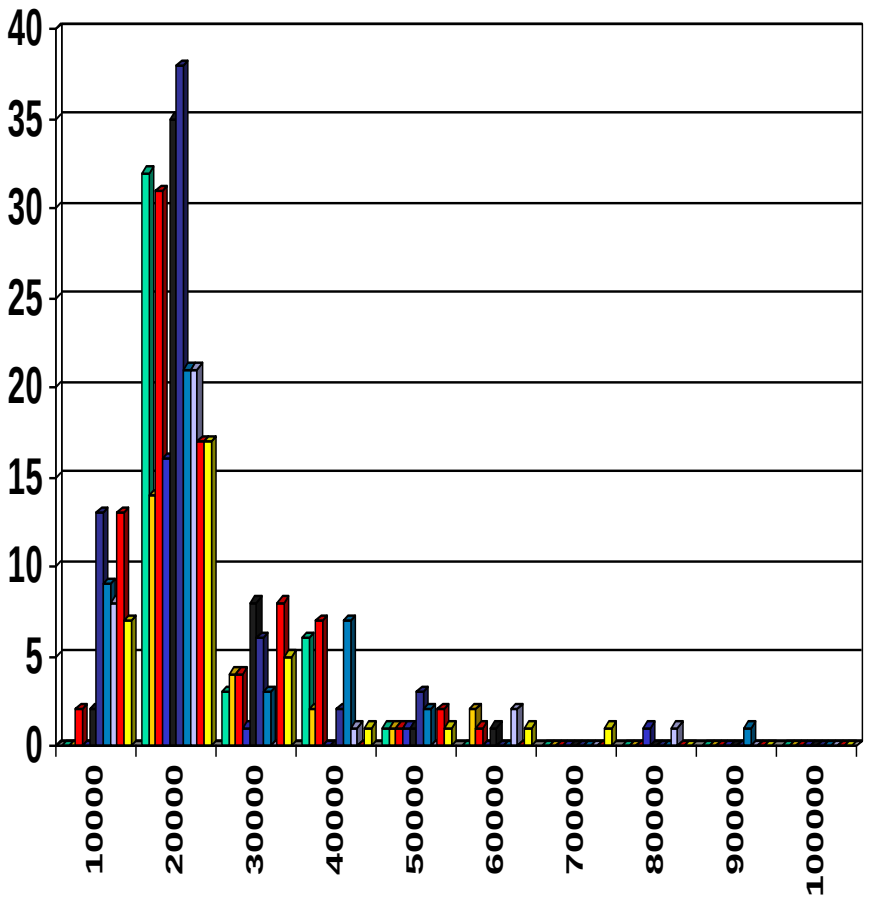

- Regression analysis: A collective name for techniques for the modeling and analysis of numerical data consisting of values of a ***dependent variable*** (also called ***response variable*** or *measurement*) and of one or more *independent variables* (aka. ***explanatory variables*** or ***predictors***)
- The parameters are estimated to give a "**best fit**" of the data
- Most commonly the best fit is evaluated by using the ***least squares method***, but other criteria have also been used
- Used for prediction (including forecasting of time-series data), inference, hypothesis testing, and modeling of causal relationships

# Regress Analysis and Log-Linear Models

- Linear regression: $Y = w X + b$
  - Two regression coefficients, $w$ and $b$, specify the line and are to be estimated by using the data at hand
  - Using the least squares criterion to the known values of $Y_1$, $Y_2$, …, $X_1$, $X_2$, ….
- Multiple regression: $Y = b_0 + b_1 X_1 + b_2 X_2$
  - Many nonlinear functions can be transformed into the above
- Log-linear models:
  - Approximate discrete multidimensional probability distributions
  - Estimate the probability of each point (tuple) in a multi-dimensional space for a set of discretized attributes, based on a smaller subset of dimensional combinations
  - Useful for dimensionality reduction and data smoothing

# Histogram Analysis

- Divide data into buckets and store average (sum) for each bucket
- Partitioning rules:
  - Equal-width: equal bucket range
  - Equal-frequency (or equal-depth)



# Clustering

- Partition data set into clusters based on similarity, and store cluster representation (e.g., centroid and diameter) only
- Can be very effective if data is clustered but not if data is “smeared”
- Can have hierarchical clustering and be stored in multi-dimensional index tree structures
- There are many choices of clustering definitions and clustering algorithms
- Cluster analysis will be studied in depth in Chapter 10

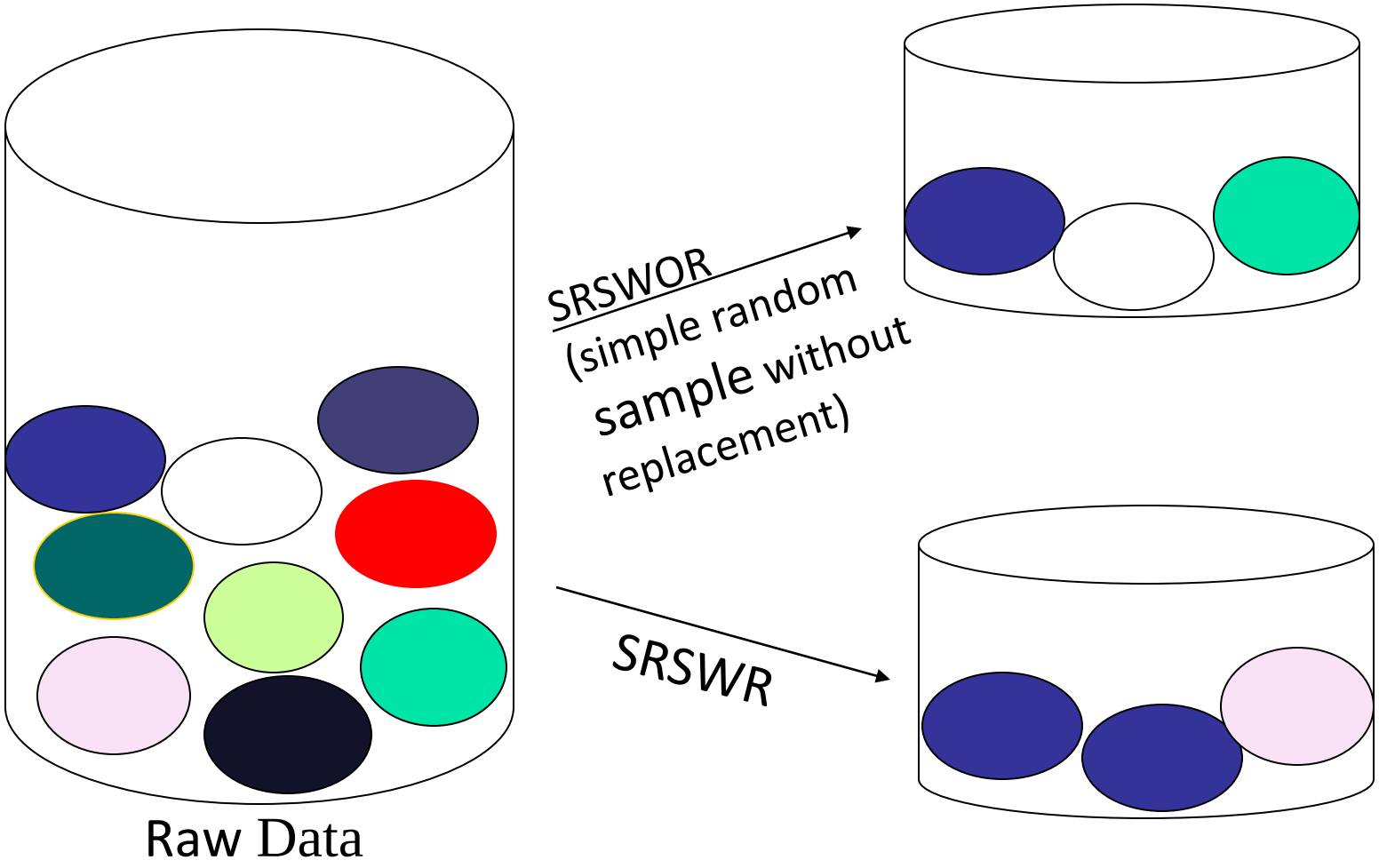
# Sampling

- Sampling: obtaining a small sample *s* to represent the whole data set *N*
- Allow a mining algorithm to run in complexity that is potentially sub-linear to the size of the data
- Key principle: Choose a representative subset of the data
  - Simple random sampling may have very poor performance in the presence of skew
  - Develop adaptive sampling methods, e.g., stratified sampling:
- Note: Sampling may not reduce database I/Os (page at a time)

# Types of Sampling

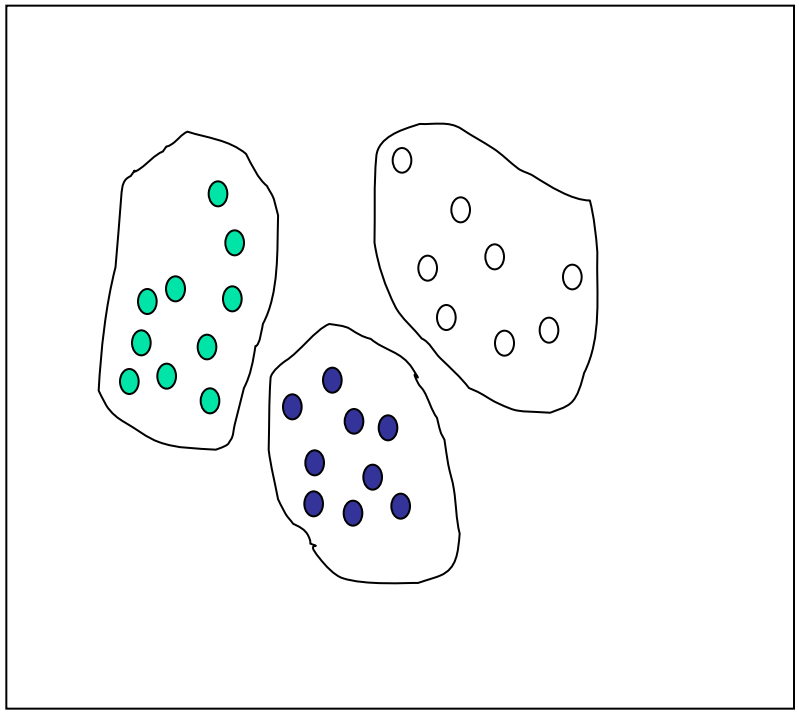
- **Simple random sampling**
  - There is an equal probability of selecting any particular item
- **Sampling without replacement**
  - Once an object is selected, it is removed from the population
- **Sampling with replacement**
  - A selected object is not removed from the population
- **Stratified sampling:**
  - Partition the data set, and draw samples from each partition (proportionally, i.e., approximately the same percentage of the data)
  - Used in conjunction with skewed data

# Sampling: With or without Replacement

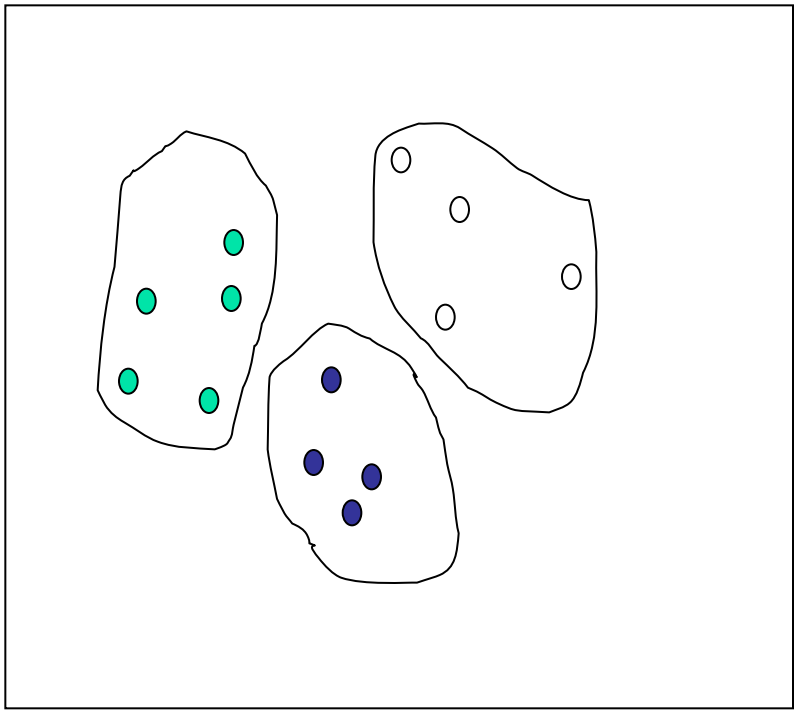


# Sampling: Cluster or Stratified Sampling

Raw Data



Cluster/Stratified Sample



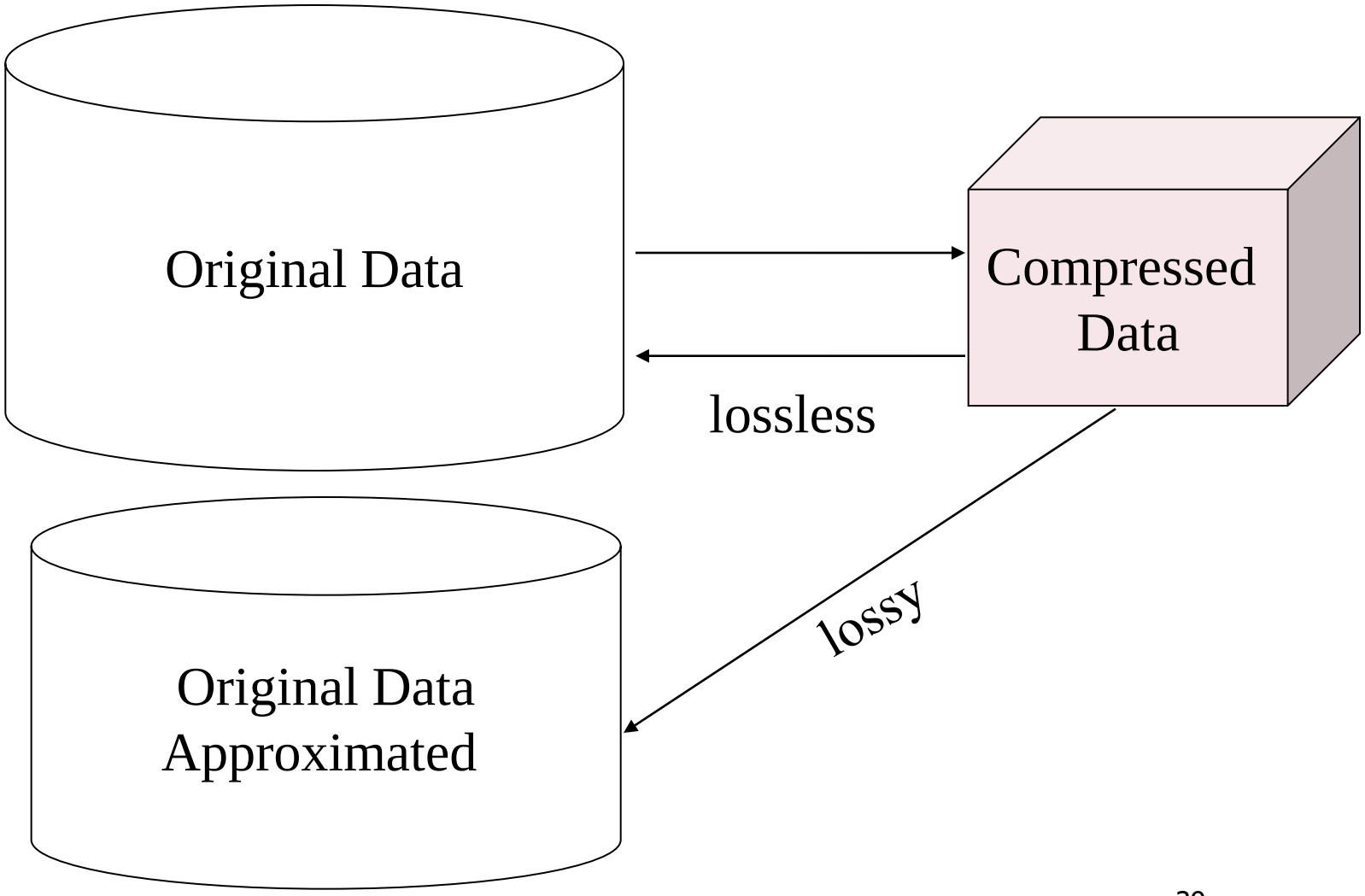
# Data Cube Aggregation

- The lowest level of a data cube (base cuboid)
  - The aggregated data for an individual entity of interest
  - E.g., a customer in a phone calling data warehouse
- Multiple levels of aggregation in data cubes
  - Further reduce the size of data to deal with
- Reference appropriate levels
  - Use the smallest representation which is enough to solve the task
- Queries regarding aggregated information should be answered using a data cube, when possible

# 3. Data Reduction : Data Compression

- String compression
  - There are extensive theories and well-tuned algorithms
  - Typically lossless, but only limited manipulation is possible without expansion
- Audio/video compression
  - Typically lossy compression, with progressive refinement
  - Sometimes small fragments of the signal can be reconstructed without reconstructing the whole
- Time sequence is no audio
  - Typically short and vary slowly with time
- Dimensionality and numerosity reduction may also be considered as forms of data compression

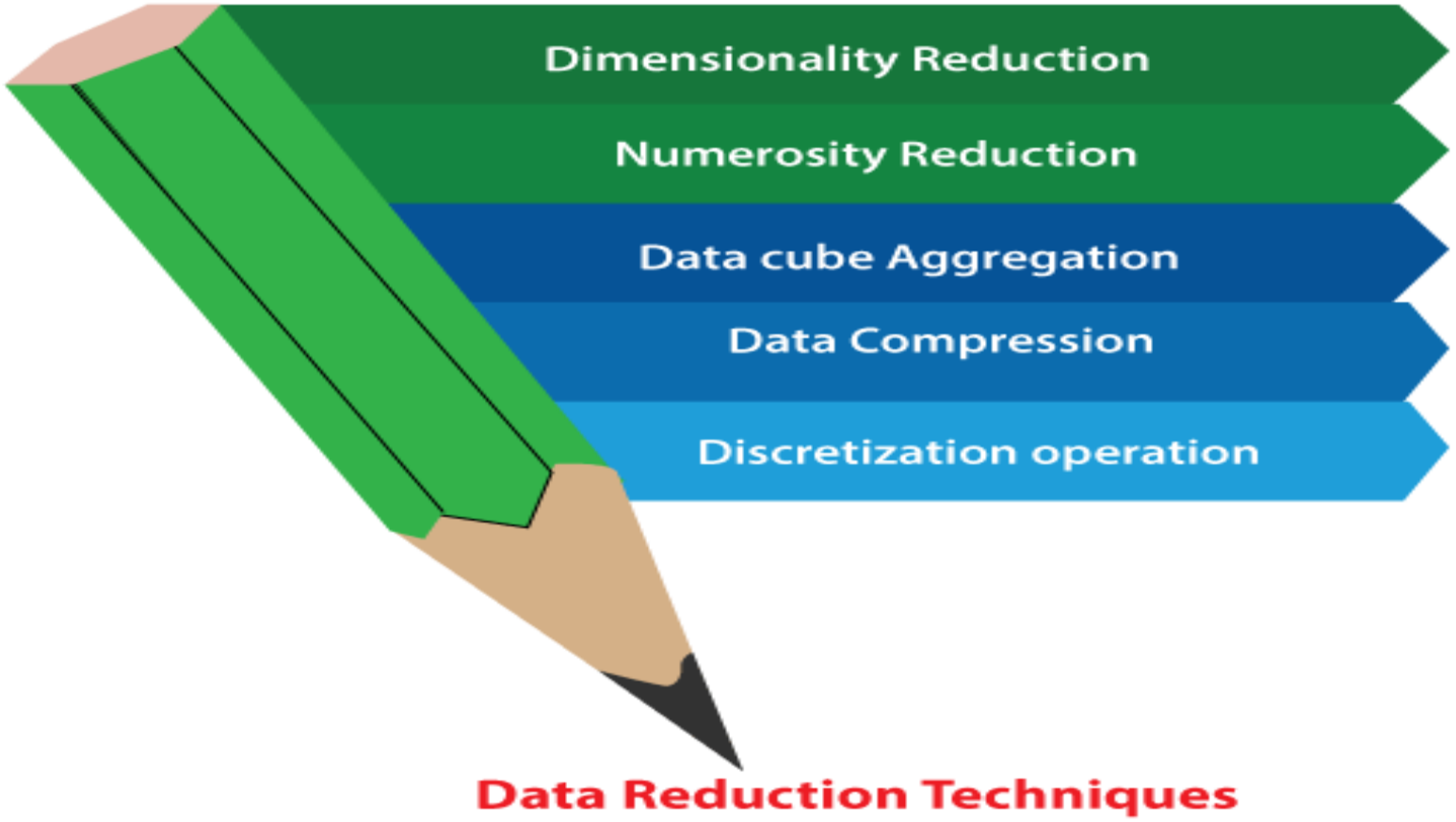
# Data Compression



# (Same again - in short)
# Data Reduction

- Data mining is applied to the selected data in a large amount database. When data analysis and mining is done on a huge amount of data, then it takes a very long time to process, making it impractical and infeasible.
- Data reduction techniques ensure the integrity of data while reducing the data. Data reduction is a process that reduces the volume of original data and represents it in a much smaller volume. Data reduction techniques are used to obtain a reduced representation of the dataset that is much smaller in volume by maintaining the integrity of the original data. By reducing the data, the efficiency of the data mining process is improved, which produces the same analytical results.
- Data reduction does not affect the result obtained from data mining. That means the result obtained from data mining before and after data reduction is the same or almost the same.
- Data reduction aims to define it more compactly. When the data size is smaller, it is simpler to apply sophisticated and computationally high-priced algorithms. The reduction of the data may be in terms of the number of rows (records) or terms of the number of columns (dimensions).

# Techniques


Dimensionality Reduction
Numerosity Reduction
Data cube Aggregation
Data Compression
Discretization operation
Data Reduction Techniques

# Dimensionality Reduction

- Whenever we encounter weakly important data, we use the attribute required for our analysis.
- Dimensionality reduction eliminates the attributes from the data set under consideration, thereby reducing the volume of original data. It reduces data size as it eliminates outdated or redundant features. Here are three methods of dimensionality reduction.
- **Wavelet Transform**
- **Principal Component Analysis**
- **Attribute Subset Selection**

# Wavelet Transform

- **Wavelet Transform:**
- In the wavelet transform, suppose a data vector A is transformed into a numerically different data vector A' such that both A and A' vectors are of the same length.
- Then how it is useful in reducing data because the data obtained from the wavelet transform can be truncated. The compressed data is obtained by retaining the smallest fragment of the strongest wavelet coefficients. Wavelet transform can be applied to data cubes, sparse data, or skewed data.

# Principal Component Analysis

- **Principal Component Analysis:** Suppose we have a data set to be analyzed that has tuples with n attributes. The principal component analysis identifies k independent tuples with n attributes that can represent the data set.

- In this way, the original data can be cast on a much smaller space, and dimensionality reduction can be achieved. Principal component analysis can be applied to sparse and skewed data.

# Attribute Subset Selection

- **Attribute Subset Selection:** The large data set has many attributes, some of which are irrelevant to data mining or some are redundant. The core attribute subset selection reduces the data volume and dimensionality.

- The attribute subset selection reduces the volume of data by eliminating redundant and irrelevant attributes.

- The attribute subset selection ensures that we get a good subset of original attributes even after eliminating the unwanted attributes. The resulting probability of data distribution is as close as possible to the original data distribution using all the attributes.

# Numerosity Reduction

- The numerosity reduction reduces the original data volume and represents it in a much smaller form. This technique includes two types parametric and non-parametric numerosity reduction.

i. **Parametric:** Parametric numerosity reduction incorporates storing only data parameters instead of the original data. One method of parametric numerosity reduction is the regression and log-linear method.

   i. **Regression and Log-Linear:** Linear regression models a relationship between the two attributes by modeling a linear equation to the data set. Suppose we need to model a linear function between two attributes.

   ***y = wx +b***

   Here, y is the response attribute, and x is the predictor attribute. If we discuss in terms of data mining, attribute x and attribute y are the numeric database attributes, whereas w and b are regression coefficients.

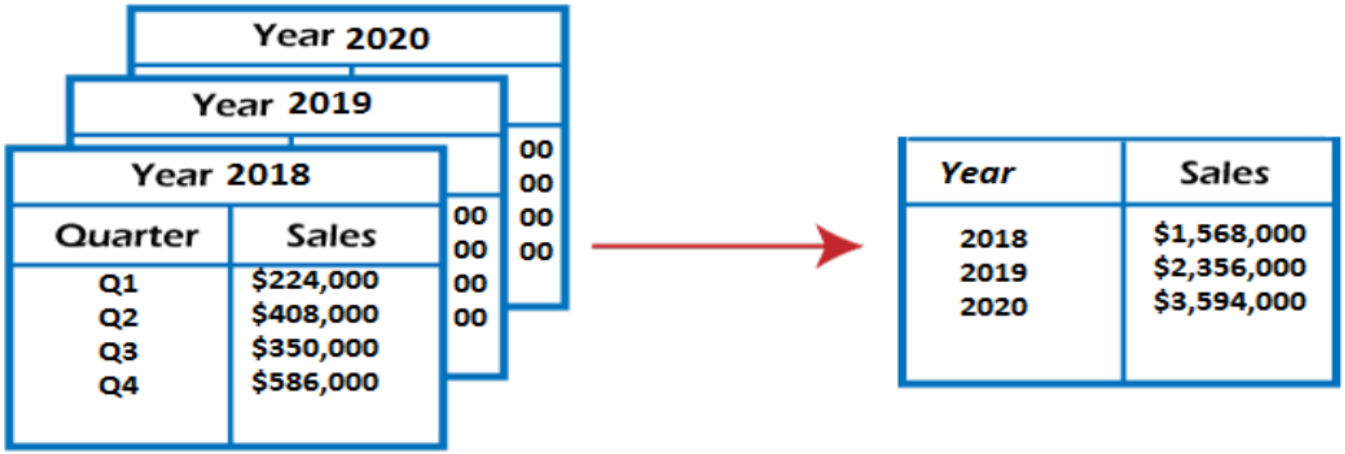
   Multiple linear regressions let the response variable y model linear function between two or more predictor variables.

   Log-linear model discovers the relation between two or more discrete attributes in the database. Suppose we have a set of tuples presented in n-dimensional space. Then the log-linear model is used to study the probability of each tuple in a multidimensional space. Regression and log-linear methods can be used for sparse data and skewed data.

1. **Sampling:** One of the methods used for data reduction is sampling, as it can reduce the large data set into a much smaller data sample. Below we will discuss the different methods in which we can sample a large data set D containing N tuples:**Simple random sample without replacement (SRSWOR) of size s:** In this s, some tuples are drawn from N tuples such that in the data set D (s<N). The probability of drawing any tuple from the data set D is 1/N. This means all tuples have an equal probability of getting sampled.
2. **Simple random sample with replacement (SRSWR) of size s:** It is similar to the SRSWOR, but the tuple is drawn from data set D, is recorded, and then replaced into the data set D so that it can be drawn again.
3. **Cluster sample:** The tuples in data set D are clustered into M mutually disjoint subsets. The data reduction can be applied by implementing SRSWOR on these clusters. A simple random sample of size s could be generated from these clusters where s<M.
4. **Stratified sample:** The large data set D is partitioned into mutually disjoint sets called 'strata'. A simple random sample is taken from each stratum to get stratified data. This method is effective for skewed data.

# Data Cube Aggregation

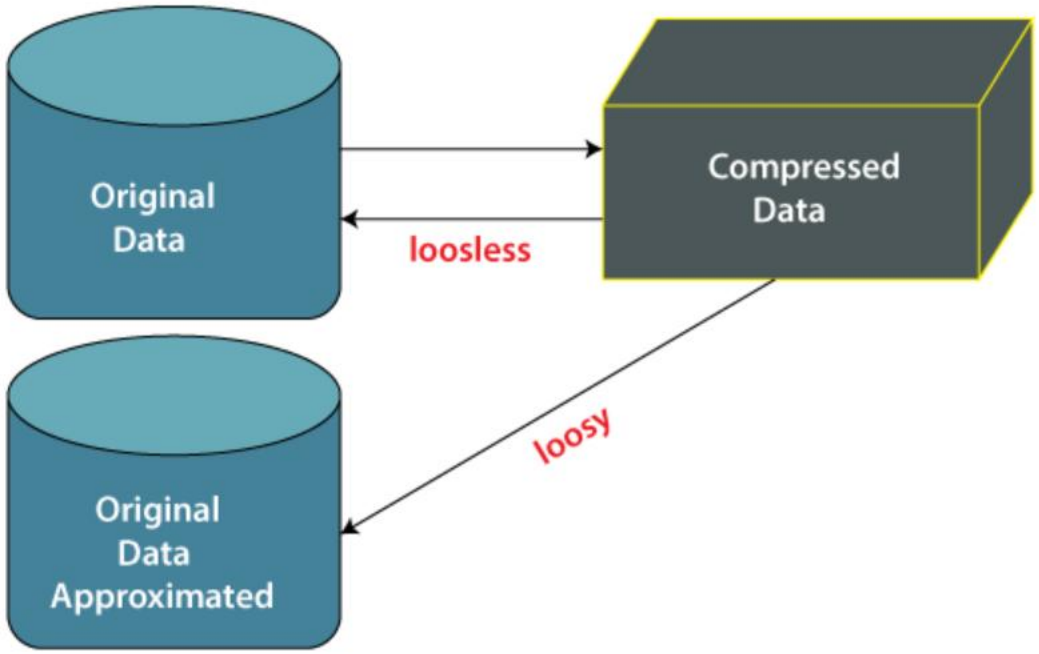
- This technique is used to aggregate data in a simpler form. Data Cube Aggregation is a multidimensional aggregation that uses aggregation at various levels of a data cube to represent the original data set, thus achieving data reduction.
- For example, suppose you have the data of All Electronics sales per quarter for the year 2018 to the year 2022. If you want to get the annual sale per year, you just have to aggregate the sales per quarter for each year. In this way, aggregation provides you with the required data, which is much smaller in size, and thereby we achieve data reduction even without losing any data.




Aggregated Data

# Data Compression

- Data compression employs modification, encoding, or converting the structure of data in a way that consumes less space. Data compression involves building a compact representation of information by removing redundancy and representing data in binary form. Data that can be restored successfully from its compressed form is called Lossless compression. In contrast, the opposite where it is not possible to restore the original form from the compressed form is Lossy compression. Dimensionality and numerosity reduction method are also used for data compression.



- This technique reduces the size of the files using different encoding mechanisms, such as Huffman Encoding and run-length Encoding. We can divide it into two types based on their compression techniques.

i. **Lossless Compression:** Encoding techniques (Run Length Encoding) allow a simple and minimal data size reduction. Lossless data compression uses algorithms to restore the precise original data from the compressed data.

ii. **Lossy Compression:** In lossy-data compression, the decompressed data may differ from the original data but are useful enough to retrieve information from them. For example, the JPEG image format is a lossy compression, but we can find the meaning equivalent to the original image. Methods such as the Discrete Wavelet transform technique PCA (principal component analysis) are examples of this compression.

# Data Discretization

- The data discretization technique is used to divide the attributes of the continuous nature into data with intervals. We replace many constant values of the attributes with labels of small intervals. This means that mining results are shown in a concise and easily understandable way.

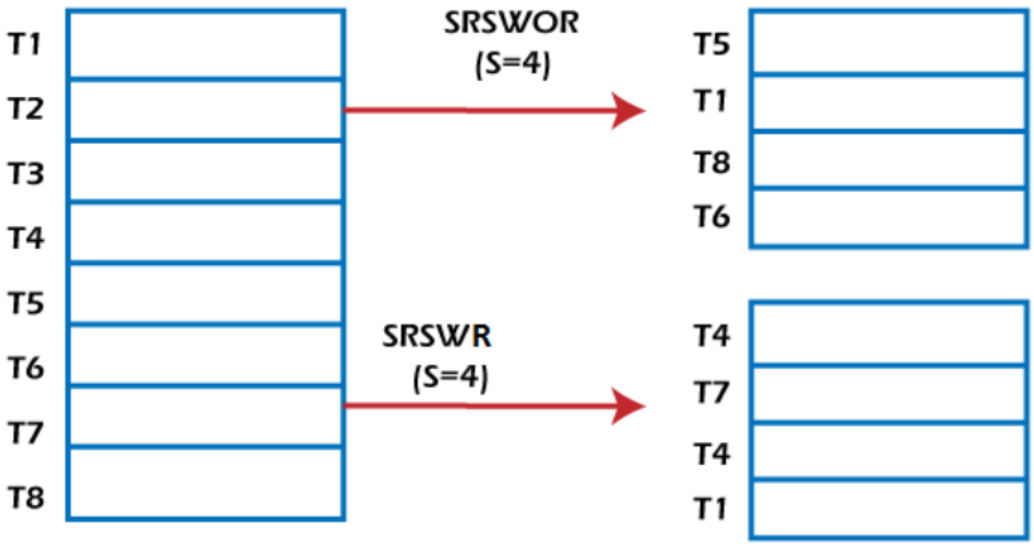
i. **Top-down discretization:** If you first consider one or a couple of points (so-called breakpoints or split points) to divide the whole set of attributes and repeat this method up to the end, then the process is known as top-down discretization, also known as splitting.

ii. **Bottom-up discretization:** If you first consider all the constant values as split points, some are discarded through a combination of the neighborhood values in the interval. That process is called bottom-up discretization.

# Benefits of Data Reduction

- The main benefit of data reduction is simple: the more data you can fit into a terabyte of disk space, the less capacity you will need to purchase. Here are some benefits of data reduction, such as:

- Data reduction can save energy.
- Data reduction can reduce your physical storage costs.
- And data reduction can decrease your data center track.
- Data reduction greatly increases the efficiency of a storage system and directly impacts your total spending on capacity.


T1
T2
T3
T4
T5
T6
T7
T8
SRSWOR
(S=4)
T5
T1
T8
T6
SRSWR
(S=4)
T4
T7
T4
T1

# Data Preprocessing

- Data Preprocessing: An Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation and Data Discretization 
- Summary

# Data Transformation

- A function that maps the entire set of values of a given attribute to a new set of replacement values s.t. each old value can be identified with one of the new values
- Methods
  - Smoothing: Remove noise from data
  - Attribute/feature construction
    - New attributes constructed from the given ones
  - Aggregation: Summarization, data cube construction
  - Normalization: Scaled to fall within a smaller, specified range
    - min-max normalization
    - z-score normalization
    - normalization by decimal scaling
  - Discretization: Concept hierarchy climbing

# Data Transformation by Normalization

- The measurement unit used can affect the data analysis. For example, changing measurement units from meters to inches for height, or from kilograms to pounds for weight, may lead to very different results.

- In general, expressing an attribute in smaller units will lead to a larger range for that attribute, and thus tend to give such an attribute greater effect or "weight."

- To help avoid dependence on the choice of measurement units, the data should be normalized or standardized.

- This involves transforming the data to fall within a smaller or common range such as [−1,1] or [0.0, 1.0]. (The terms standardize and normalize are used interchangeably in data preprocessing, although in statistics, the latter term also has other connotations.)

# Normalization

- Normalizing the data attempts to give all attributes an equal weight.
- Normalization is particularly useful for classification algorithms involving neural networks or distance measurements such as nearest-neighbor classification and clustering.
- If using the neural network backpropagation algorithm for classification mining, normalizing the input values for each attribute measured in the training tuples will help speed up the learning phase.

# Normalization

- **Min-max normalization**: to [new_min$_A$, new_max$_A$]

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$

  - Ex. Let income range \$12,000 to \$98,000 normalized to [0.0, 1.0]. Then \$73,000 is mapped to $\frac{73{,}600 - 12{,}000}{98{,}000 - 12{,}000}(1.0 - 0) + 0 = 0.716$

- **Z-score normalization** (μ: mean, σ: standard deviation):

$$v' = \frac{v - \mu_A}{\sigma_A}$$

  - Ex. Let μ = 54,000, σ = 16,000. Then $\frac{73{,}600 - 54{,}000}{16{,}000} = 1.225$

- **Normalization by decimal scaling**

$$v' = \frac{v}{10^j}$$ Where $j$ is the smallest integer such that Max(|v'|) < 1

# Discretization

- Three types of attributes
  - Nominal—values from an unordered set, e.g., color, profession
  - Ordinal—values from an ordered set, e.g., military or academic rank
  - Numeric—real numbers, e.g., integer or real numbers

- Discretization: Divide the range of a continuous attribute into intervals
  - Interval labels can then be used to replace actual data values
  - Reduce data size by discretization
  - Supervised vs. unsupervised
  - Split (top-down) vs. merge (bottom-up)
  - Discretization can be performed recursively on an attribute
  - Prepare for further analysis, e.g., classification

# Data Discretization Methods

- Typical methods: All the methods can be applied recursively
  - Binning
    - Top-down split, unsupervised
  - Histogram analysis
    - Top-down split, unsupervised
  - Clustering analysis (unsupervised, top-down split or bottom-up merge)
  - Decision-tree analysis (supervised, top-down split)
  - Correlation (e.g., $\chi^2$) analysis (unsupervised, bottom-up merge)

# Simple Discretization: Binning

- Equal-width (distance) partitioning
  - Divides the range into *N* intervals of equal size: uniform grid
  - if *A* and *B* are the lowest and highest values of the attribute, the width of intervals will be $W = (B - A)/N$.
  - The most straightforward, but outliers may dominate the presentation
  - Skewed data is not handled well
- Equal-depth (frequency) partitioning
  - Divide the range into *N* intervals, each containing approximately the same number of samples
  - Good data scaling
  - Managing categorical attributes can be tricky

# Binning Methods for Data Smoothing

- Sorted data for price (in dollars): 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

* Partition into equal-frequency (**equi-depth**) bins:
  - Bin 1: 4, 8, 9, 15
  - Bin 2: 21, 21, 24, 25
  - Bin 3: 26, 28, 29, 34
* Smoothing by **bin means**:
  - Bin 1: 9, 9, 9, 9
  - Bin 2: 23, 23, 23, 23
  - Bin 3: 29, 29, 29, 29
* Smoothing by **bin boundaries**:
  - Bin 1: 4, 4, 4, 15
  - Bin 2: 21, 21, 25, 25
  - Bin 3: 26, 26, 26, 34

# Discretization by Classification & Correlation Analysis

- Classification (e.g., decision tree analysis)
  - Supervised: Given class labels, e.g., cancerous vs. benign
  - Using *entropy* to determine the split point (discretization point)
  - Top-down, recursive split
  - Details to be covered in Chapter 7
- Correlation analysis (e.g., Chi-merge: $\chi^2$-based discretization)
  - Supervised: use class information
  - Bottom-up merge: find the best neighboring intervals (those having similar distributions of classes, i.e., low $\chi^2$ values) to merge
  - Merge performed recursively, until a predefined stopping condition

# Concept **Hierarchy Generation**

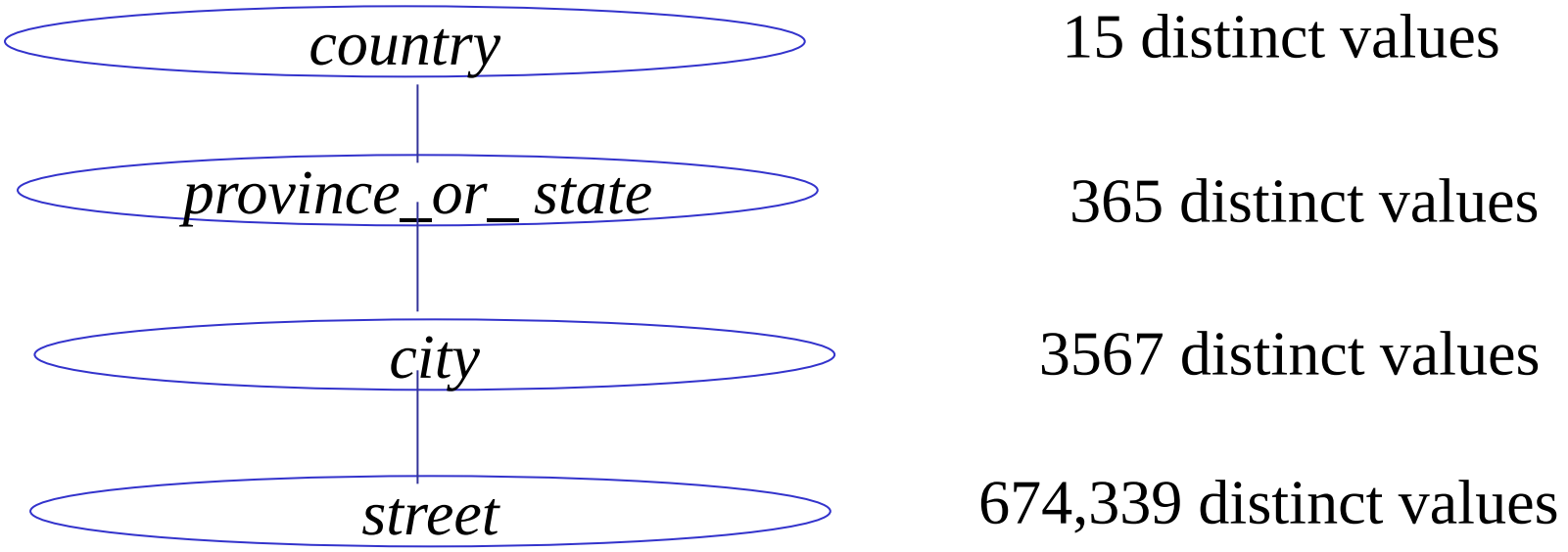
- **Concept hierarchy** organizes concepts (i.e., attribute values) hierarchically and is usually associated with each dimension in a data warehouse
- Concept hierarchies facilitate drilling and rolling in data warehouses to view data in multiple granularity
- Concept hierarchy formation: Recursively reduce the data by collecting and replacing low-level concepts (such as numeric values for *age*) by higher-level concepts (such as *youth, adult*, or *senior*)
- Concept hierarchies can be explicitly specified by domain experts and/or data warehouse designers
- Concept hierarchy can be automatically formed for both numeric and nominal data. For numeric data, use the discretization methods shown.

# Concept Hierarchy Generation for Nominal Data

- Specification of a partial/total ordering of attributes explicitly at the schema level by users or experts
  - *street* < *city* < *state* < *country*
- Specification of a hierarchy for a set of values by explicit data grouping
  - {Urbana, Champaign, Chicago} < Illinois
- Specification of only a partial set of attributes
  - E.g., only *street* < *city*, not others
- Automatic generation of hierarchies (or attribute levels) by the analysis of the number of distinct values
  - E.g., for a set of attributes: {*street, city, state, country*}

# Automatic Concept Hierarchy Generation

- Some hierarchies can be automatically generated based on the analysis of the number of distinct values per attribute in the data set
  - The attribute with the most distinct values is placed at the lowest level of the hierarchy
  - Exceptions, e.g., weekday, month, quarter, year



# Data **Preprocessing**

- Data Preprocessing: An Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation and Data Discretization
- Summary 

# Data **Transformation**

- In this preprocessing step, the data are transformed or consolidated so that the resulting mining process may be more efficient, and the patterns found may be easier to understand.

- Data discretization, a form of data transformation, is also discussed

# Data Transformation Strategies Overview

1. **Smoothing**, which works to remove noise from the data. Techniques include binning, regression, and clustering.
2. **Attribute construction** (or *feature construction*), where new attributes are constructed and added from the given set of attributes to help the mining process.
3. **Aggregation**, where summary or aggregation operations are applied to the data. For example, the daily sales data may be aggregated so as to compute monthly and annual total amounts. This step is typically used in constructing a data cube for data analysis at multiple abstraction levels.
4. **Normalization**, where the attribute data are scaled so as to fall within a smaller range, such as −1.0 to 1.0, or 0.0 to 1.0.
5. **Discretization**, where the raw values of a numeric attribute (e.g., *age*) are replaced by interval labels (e.g., 0–10, 11–20, etc.) or conceptual labels (e.g., *youth*, *adult*, *senior*). The labels, in turn, can be recursively organized into higher-level concepts, resulting in a *concept hierarchy* for the numeric attribute. Figure 3.12 shows a concept hierarchy for the attribute *price*. More than one concept hierarchy can be defined for the same attribute to accommodate the needs of various users.
6. **Concept hierarchy generation for nominal data**, where attributes such as *street* can be generalized to higher-level concepts, like *city* or *country*. Many hierarchies for nominal attributes are implicit within the database schema and can be automatically defined at the schema definition level.

# Concept **Hierarchy for price**

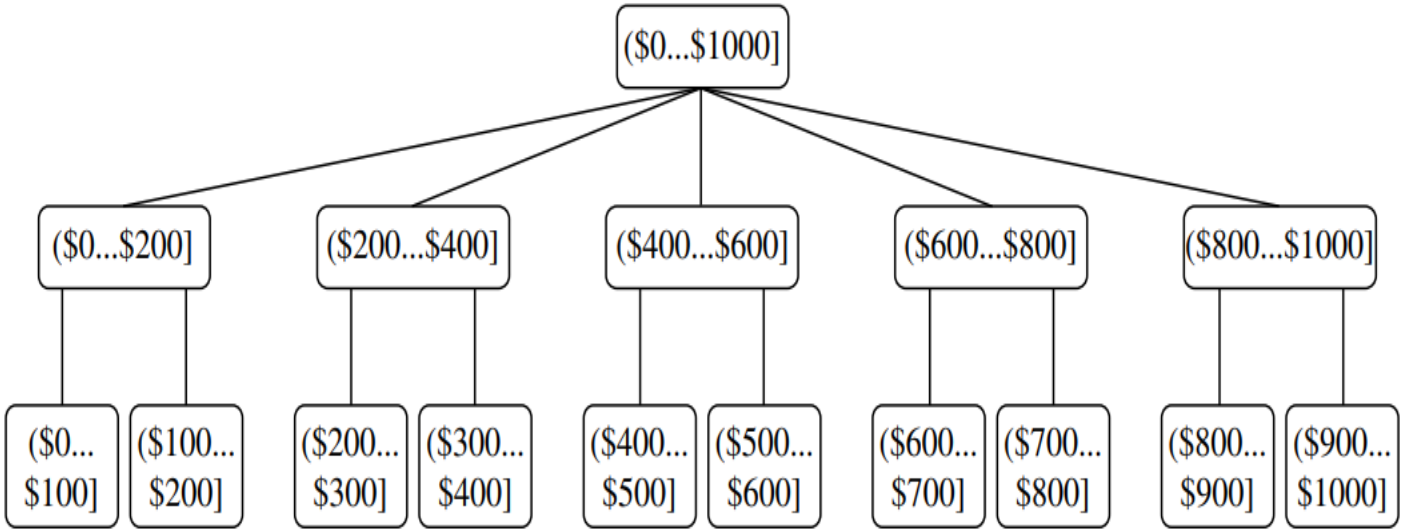



**Figure 3.12** A concept hierarchy for the attribute *price*, where an interval ($X...$Y] denotes the range from $X (exclusive) to $Y (inclusive).

# Summary

- **Data quality**: accuracy, completeness, consistency, timeliness, believability, interpretability
- **Data cleaning**: e.g. missing/noisy values, outliers
- **Data integration** from multiple sources:
  - Entity identification problem
  - Remove redundancies
  - Detect inconsistencies
- **Data reduction**
  - Dimensionality reduction
  - Numerosity reduction
  - Data compression
- **Data transformation and data discretization**
  - Normalization
  - Concept hierarchy generation

# References


- D. P. Ballou and G. K. Tayi. Enhancing data quality in data warehouse environments. Comm. of ACM, 42:73-78, 1999
- A. Bruce, D. Donoho, and H.-Y. Gao. Wavelet analysis. *IEEE Spectrum*, Oct 1996
- T. Dasu and T. Johnson. Exploratory Data Mining and Data Cleaning. John Wiley, 2003
- J. Devore and R. Peck. *Statistics: The Exploration and Analysis of Data*. Duxbury Press, 1997.
- H. Galhardas, D. Florescu, D. Shasha, E. Simon, and C.-A. Saita. Declarative data cleaning: Language, model, and algorithms. *VLDB'01*
- M. Hua and J. Pei. Cleaning disguised missing data: A heuristic approach. *KDD'07*
- H. V. Jagadish, et al., Special Issue on Data Reduction Techniques. Bulletin of the Technical Committee on Data Engineering, 20(4), Dec. 1997
- H. Liu and H. Motoda (eds.). *Feature Extraction, Construction, and Selection: A Data Mining Perspective*. Kluwer Academic, 1998
- J. E. Olson. *Data Quality: The Accuracy Dimension*. Morgan Kaufmann, 2003
- D. Pyle. Data Preparation for Data Mining. Morgan Kaufmann, 1999
- V. Raman and J. Hellerstein. Potters Wheel: An Interactive Framework for Data Cleaning and Transformation, VLDB'2001
- T. Redman. *Data Quality: The Field Guide*. Digital Press (Elsevier), 2001
- R. Wang, V. Storey, and C. Firth. A framework for analysis of data quality research. IEEE Trans. Knowledge and Data Engineering, 7:623-640, 1995

# Unit 2

## Data Analytics

- Introduction to Analytics, Introduction to Tools and Environment, R Language, Python, SPSS etc, Need for Business Modeling, Key Responsibilities of a Business Analyst, Purpose of business Modeling, Application of Modeling in Business, Databases & Types of Data and variables, Data Modeling Techniques, Conceptual, Logical, Physical Data Model, Missing imputations, MCAR, MAR, MNAR.

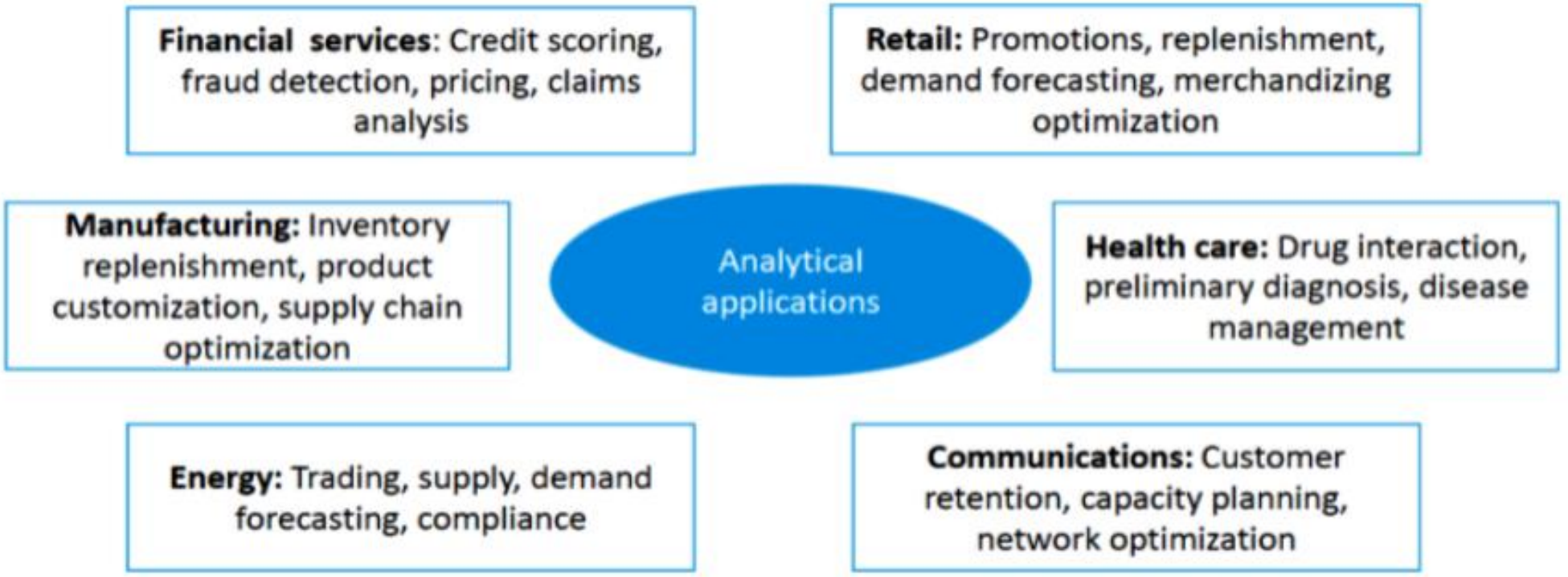
# Introduction to Analytics

- Predictive Analytics is an art of predicting the future based on past trends.
- It is a branch of Statistics comprising Modeling Techniques, Machine Learning & Data Mining.
- Predictive Analytics is primarily used in Decision Making.

# What and Why Analytics

- Analytics is a journey that involves a combination of potential skills, advanced technologies, applications, and processes used by firms to gain business insights from data and statistics.
- This is done to perform business planning.

# Application areas



# What is Data Analytics?

- Data analytics takes raw data and turns it into useful information. It uses various tools and methods to discover patterns and solve problems with data. Data analytics helps businesses make better decisions and grow.
- Companies around the globe generate vast volumes of data daily, in the form of log files, web servers, transactional data, and various customer-related data. In addition to this, social media websites also generate enormous amounts of data.
- Companies ideally need to use all of their generated data to derive value out of it and make impactful business decisions. Data analytics is used to drive this purpose.

# Why to Use Data Analytics

- let’s understand how we can use data analytics.

**1. Improved Decision Making:** Data Analytics eliminates guesswork and manual tasks. Be it choosing the right content, planning marketing campaigns, or developing products. Organizations can use the insights they gain from data analytics to make informed decisions. Thus, leading to better outcomes and customer satisfaction.

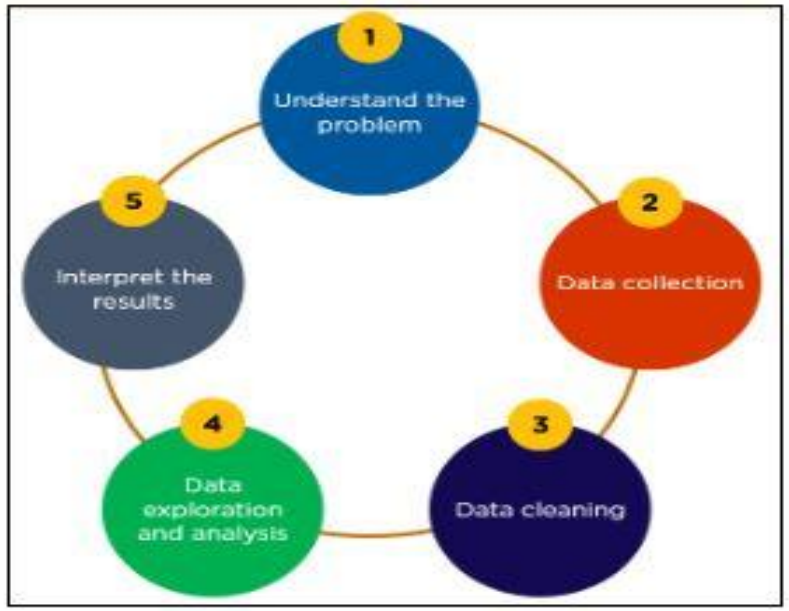
**2. Better Customer Service:** Data analytics allows you to tailor customer service to their needs. It also provides personalization and builds stronger relationships with customers. Analyzed data can reveal customers’ interests, concerns, and more. It helps you give better recommendations for products and services.

**3. Efficient Operations:** With the help of data analytics, you can streamline your processes, save money, and boost production. With an improved understanding of what your audience wants, you spend less time creating ads and content that aren't in line with your audience's interests.

**4. Effective Marketing:** Data analytics gives you valuable insights into how your campaigns are performing. This helps in fine-tuning them for optimal outcomes. Additionally, you can also find potential customers who are most likely to interact with a campaign and convert into leads.

# Steps Involved in Data Analytics

- Imagine you are running an e-commerce business and your company has nearly a million in customer base. Your aim is to figure out certain problems related to your business, and subsequently come up with data-driven solutions to grow your business.
- Below are the steps that you can take to solve your problems.



**1. Understand the problem:** Understanding the business problems, defining the organizational goals, and planning a lucrative solution is the first step in the analytics process. E-commerce companies often encounter issues such as predicting the return of items, giving relevant product recommendations, canceling orders, identifying frauds, optimizing vehicle routing, etc.

**2. Data Collection:** Next, you need to collect transactional business data and customer-related information from the past few years to address the problems your business is facing. The data can have information about the total units that were sold for a product, the sales, and profit that were made, and also when was the order placed. Past data plays a crucial role in shaping the future of a business.

**3. Data Cleaning:** Now, all the data you collect will often be disorderly, messy, and contain unwanted missing values. Such data is not suitable or relevant for performing data analysis. Hence, you need to clean the data to remove unwanted, redundant, and missing values to make it ready for analysis.

**4. Data Exploration and Analysis:** After you gather the right data, the next vital step is to execute exploratory data analysis. You can use data visualization and business intelligence tools, data mining techniques, and predictive modeling to analyze, visualize, and predict future outcomes from this data. Applying these methods can tell you the impact and relationship of a certain feature as compared to other variables.

**5. Interpret the results:** The final step is to interpret the results and validate if the outcomes meet your expectations. You can find out hidden patterns and future trends. This will help you gain insights that will support you with appropriate data-driven decision-making.

# Data Analytics Applications

Data analytics is used in almost every sector of business, let's discuss a few of them:

**1. Retail:** Data analytics helps retailers understand their customer needs and buying habits to predict trends, recommend new products, and boost their business. They optimize the supply chain, and retail operations at every step of the customer journey.

**2. Healthcare:** Healthcare industries analyze patient data to provide lifesaving diagnoses and treatment options. Data analytics help in discovering new drug development methods as well.

**3. Manufacturing:** Using data analytics, manufacturing sectors can discover new cost-saving opportunities. They can solve complex supply chain issues, labor constraints, and equipment breakdowns.

**4. Banking sector:** Banking and financial institutions use analytics to find out probable loan defaulters and customer churn-out rates. It also helps in detecting fraudulent transactions immediately.

**5. Logistics:** Logistics companies use data analytics to develop new business models and optimize routes. This, in turn, ensures that the delivery reaches on time in a cost-efficient manner.

# Introduction to tools and Environment

- Analytics is now a days used in all fields ranging from Medical Science to Aero science to Government Activities.
- Data Science and Analytics are used by Manufacturing companies as well as Real Estate firms to develop their business and solve various issues with the help of a historical database.
- Tools are the software that can be used for Analytics like SAS or R.
- While techniques are the procedures to be followed to reach up to a solution.
- Various steps involved in Analytics:

1. Access
2. Manage
3. Analyze
4. Report


Access
Report
Analytics
Manage
Analyze

# Data Analytics Tools

There are several DA tools available, let's discuss the popular 7 data analytics tools, including a couple of programming languages that can help you perform analytics better.

1. Python: Python is an object-oriented open-source programming language. It supports a range of libraries for data manipulation, data visualization, and data modeling.

2. R: R is an open-source programming language majorly used for numerical and statistical analysis. It provides a range of libraries for data analysis and visualization.

3. Tableau: It is a simplified data visualization and analytics tool. This helps you create a variety of visualizations to present the data interactively, and build reports, and dashboards to showcase insights and trends.

4. Power BI: Power BI is a business intelligence tool that has an easy ‘drag and drop functionality. It supports multiple data sources with features that visually appeal to data. Power BI supports features that help you ask questions to your data and get immediate insights.

5. QlikView: QlikView offers interactive analytics with in-memory storage technology to analyze vast volumes of data and use data discoveries to support decision making. It provides social data discovery and interactive guided analytics. It can manipulate colossal data sets instantly with accuracy.

6. SPSS (Statistical Package for the Social Sciences): SPSS is a software package used for statistical analysis. It provides a graphical user interface (GUI) for conducting descriptive and inferential statistics, data manipulation, and data visualization. SPSS is commonly used in social sciences, market research, and business analytics.

7. SAS: SAS is a statistical analysis software that can help you perform analytics, visualize data, write SQL queries, perform statistical analysis, and build machine learning models to make future predictions.

## R Language

R is a programming language and software environment for statistical computing and graphics. It is widely used by statisticians, data analysts, and researchers for data manipulation, exploration, and statistical modeling. Here are some key features of R:

**Data Manipulation**: R provides powerful tools for data manipulation, including functions for subsetting, merging, reshaping, and summarizing data.

**Statistical Modeling:** R offers a vast array of statistical techniques for regression analysis, hypothesis testing, time series analysis, clustering, and more. These functionalities are available through built-in functions and packages contributed by the R community.

**Graphics:** R has extensive capabilities for creating high-quality static and interactive visualizations, allowing users to explore data and communicate findings effectively.

**Packages**: R's functionality can be extended through packages, which are collections of R functions, data, and documentation. There are thousands of packages available on the Comprehensive R Archive Network (CRAN) covering various domains and analysis techniques.

# Python

Python is a versatile programming language known for its simplicity and readability. It has become increasingly popular in data science, machine learning, and scientific computing due to its rich ecosystem of libraries and frameworks. Here are some key features of Python:

**Readability:** Python's syntax is clear and expressive, making it easy to write and understand code, which is beneficial for collaboration and maintainability.

**Libraries:** Python has a vast ecosystem of libraries for data manipulation (e.g., Pandas), numerical computing (e.g., NumPy), machine learning (e.g., scikit-learn), and data visualization (e.g., Matplotlib, Seaborn).

**Versatility:** Python is a general-purpose language, meaning it can be used for a wide range of applications beyond data analysis, including web development, automation, and scripting.

**Community:** Python has a large and active community of developers, which contributes to the development of libraries, frameworks, and resources, making it easier for users to find solutions to their problems and stay updated with the latest developments.

## **SPSS** (Statistical Package for the Social Sciences)

SPSS is a software package used for statistical analysis and data visualization. It provides a user-friendly interface for performing various statistical procedures and generating reports. Here are some key features of SPSS:

**Graphical User Interface (GUI):** SPSS offers a point-and-click interface that allows users to perform statistical analyses without writing code. This makes it accessible to users with limited programming experience.

**Statistical Procedures:** SPSS provides a wide range of statistical procedures for descriptive statistics, hypothesis testing, regression analysis, factor analysis, and more. These procedures can be accessed through menus or syntax commands.

**Data Management:** SPSS includes tools for data manipulation, transformation, and cleaning, allowing users to prepare data for analysis efficiently.

**Output Management:** SPSS generates clear and customizable output reports, including tables, charts, and graphs, which can be exported to various formats for further analysis or presentation

# Summary

- In summary….

R, Python, and SPSS are powerful tools for data analysis and statistical modeling, each with its strengths and applications. The choice of tool depends on factors such as the specific requirements of the analysis, the user's familiarity with the tool, and the preferences of the user or organization.

# Business model

- The term business model refers to a company's plan for making a profit. It identifies the products or services the business plans to sell, its identified target market, and any anticipated expenses. Business models are important for both new and established businesses. They help new, developing companies attract investment, recruit talent, and motivate management and staff.

- Established businesses should regularly update their business model or they'll fail to anticipate trends and challenges ahead. Business models also help investors evaluate companies that interest them and employees understand the future of a company they may aspire to join.

# Need for Business Modelling

Business modeling is essential for several reasons:

**Strategic Planning:** Business modeling helps in defining the strategic direction of a company. It allows businesses to visualize their current state, set goals, and develop strategies to achieve those goals.

**Resource Allocation:** By creating a business model, organizations can identify the resources required for their operations. This includes financial resources, human capital, technology, and other assets. It helps in allocating resources efficiently to maximize productivity and profitability.

**Risk Management:** Business modeling enables companies to identify potential risks and uncertainties in their operations. By modeling different scenarios, businesses can assess the impact of risks and develop strategies to mitigate them.

**Decision Making:** A well-developed business model provides insights that support decision-making processes. Whether it's launching a new product, entering a new market, or making strategic investments, a business model helps in evaluating the potential outcomes and making informed decisions.

**Communication:** Business models serve as a communication tool, both internally and externally. Internally, it helps in aligning employees with the company's goals and objectives. Externally, it communicates the value proposition of the business to stakeholders, including customers, investors, and partners.

**Innovation and Adaptability:** Business modeling encourages innovation by exploring new business ideas and opportunities. It allows businesses to adapt to changing market conditions and customer preferences by quickly adjusting their strategies and operations.

**Financial Planning and Forecasting:** Business models are instrumental in financial planning and forecasting. They help in estimating revenues, expenses, and cash flows, which are essential for budgeting and financial decision-making.

**Measuring Performance:** A business model provides benchmarks for measuring performance and monitoring progress toward goals. It enables businesses to track key performance indicators (KPIs) and make adjustments as needed to stay on course.

# Responsibilities of a Business Analyst

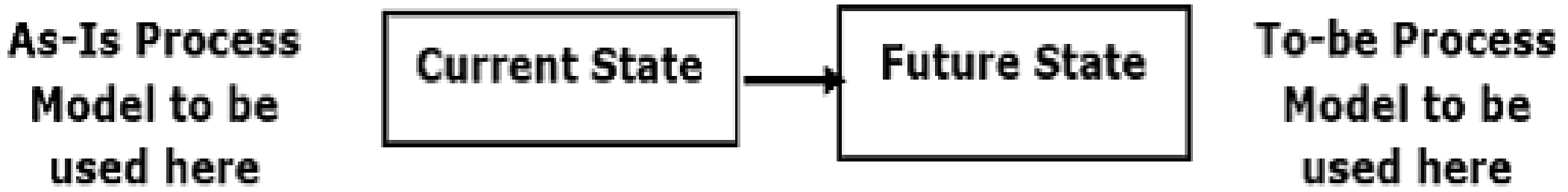
- **Understanding the Business Requirements:** A business analyst works in sync with the project stakeholders and typically understands their requirements. Once understood, they work with the developers to translate their requirements into relevant detail. Much time must be spent on meetings with the clients to know the project's requirements. This time spent saves the time of the development team.
- **Various Business Possibilities:** A business analyst must work with the primary stakeholders and other important people linked with the project. The business analyst does this by transforming the initial goal into something more realistic and workable outcome.
- **Presentation:** Now, the presentation of the work holds great significance. As a business analyst, all your capabilities are nullified if you cannot present them well. You'll be required to give many presentations about things like the project requirements, the designs of the applications made by the development team, and also many other kinds of business requirements. Remember, the audience is highly learned people who've immense knowledge about the subject you're discussing and are highly interested in knowing every word you say. So everything will matter; what you say and how you say it.
- **Project Detailing:** One of the important responsibilities is to be able to articulate the details of the project well. This usually involves working with many people in the organization, ensuring the understanding of their knowledge, and scaling the feasibility of the same. This is often taken up at the beginning to determine if the project mu

- **Supporting the Implementation:** A business analyst must be present until the project's end. The role is such that there has to be proper monitoring of the implementation of various requirements to ensure the whole team is on the same page and is working as per the stated client requirements. If there are any constraints related to the technical aspects, he or she has to work on the same with the rest of the team to meet the requirements of the business. This becomes even more pronounced in the case of new solutions where the analyst will have to understand the needs and requirements of the stakeholders. The business analysts must go to the extent of performance testing of the new product or solution. They might be required to prepare user documentation, training, and acceptance & business analyst responsibilities
- **Functional and Non-Functional Requirements:** A business analyst must work on the business's functional and non-functional aspects. While the former is related to what the project has to handle, the latter comprises how it'll be handled effectively. The non-functional requirements of the project are said to be far more important than the functional ones.
- **User Testing:** A business analysis doesn't end with delineating the needs and requirements of the project. The role also ensures that the product has been delivered and met the client's expectations. Thus, the business analyst should start putting it through various user-testing scenarios when the product reaches the final development stages. If done prudently, this almost ensures that the product will give the expected result.

- **Documentation:** This is another vital responsibility of a business analyst that one has to undertake with great care. They must prepare highly informative, coherent, and professional product documentation. The documentation has to be perfect and to the point. Remember, documentation of your work is highly important in this profile.
- **Team Building:** A business analyst is important, and nothing without his/her team. It is always up to the BA how he/she wants the team to be and how it should perform. This can take time, but once you have the right men who understand and can follow your instructions accordingly, there's no looking back. Good teams are not hired but are built. A business analyst has to build the team through constant reinforcement and encouraging them with a lot of energy and positivity.
- **Operation & System Maintenance:** A business analyst also assists in producing service reports, system validation & deactivation reports, other plans, and key documentation required for proper system maintenance.

# Purpose of Business Modelling

- Business modelling is used to design current and future state of an enterprise. This model is used by the Business Analyst and the stakeholders to ensure that they have an accurate understanding of the current “As-Is” model of the enterprise.
- It is used to verify if, stakeholders have a shared understanding of the proposed “To-be of the solution.



# Purpose of Business Modelling

- Analyzing requirements is a part of the business modelling process and it forms the core focus area.
- Functional Requirements are gathered during the “Current state”. These requirements are provided by the stakeholders regarding the business processes, data, and business rules that describe the desired functionality that will be designed in the Future State.

# Purpose of Business Modelling

The purpose of business modeling is multifaceted, encompassing various objectives aimed at enhancing decision-making, strategy development, and overall organizational performance. Here are some key purposes of business modeling:

1. **Strategic Planning**: Business modeling facilitates strategic planning by providing a structured framework for assessing current operations, analyzing market dynamics, and identifying opportunities for growth and innovation. By creating models that simulate different scenarios and outcomes, organizations can evaluate the potential impact of various strategies and make informed decisions about resource allocation and investments.
2. **Performance Analysis**: Business models enable organizations to measure and analyze their performance across different dimensions, such as financial metrics, operational efficiency, customer satisfaction, and market share. By quantifying key performance indicators (KPIs) and comparing actual results against predefined targets or benchmarks, businesses can identify areas for improvement and optimize their operations accordingly.
3. **Risk Management**: Business modeling helps organizations identify and mitigate risks by quantifying the potential impact of uncertainties and external factors on their operations. By incorporating risk factors into their models, businesses can assess their exposure to various risks, develop contingency plans, and make proactive decisions to minimize potential losses.

**4. Resource Optimization**: Business models support resource optimization by optimizing the allocation of resources, such as capital, manpower, and assets, to achieve strategic objectives and maximize efficiency. By modeling different scenarios and constraints, organizations can determine the most effective use of resources and streamline their processes to reduce costs and enhance productivity.

**5. Decision Support**: Business models serve as decision support tools by providing decision-makers with relevant information and insights to make informed choices. By synthesizing complex data and presenting it in a clear and understandable format, models enable stakeholders to evaluate alternative courses of action, assess their potential outcomes, and choose the most favorable option based on their objectives and constraints.

**6. Communication and Collaboration**: Business models facilitate communication and collaboration among stakeholders by providing a common language and framework for discussing organizational goals, strategies, and performance. By visualizing key relationships and dependencies within the business ecosystem, models help stakeholders align their objectives, coordinate their efforts, and work towards common goals more effectively.

# Data Modelling

**Common data modeling techniques and concepts**

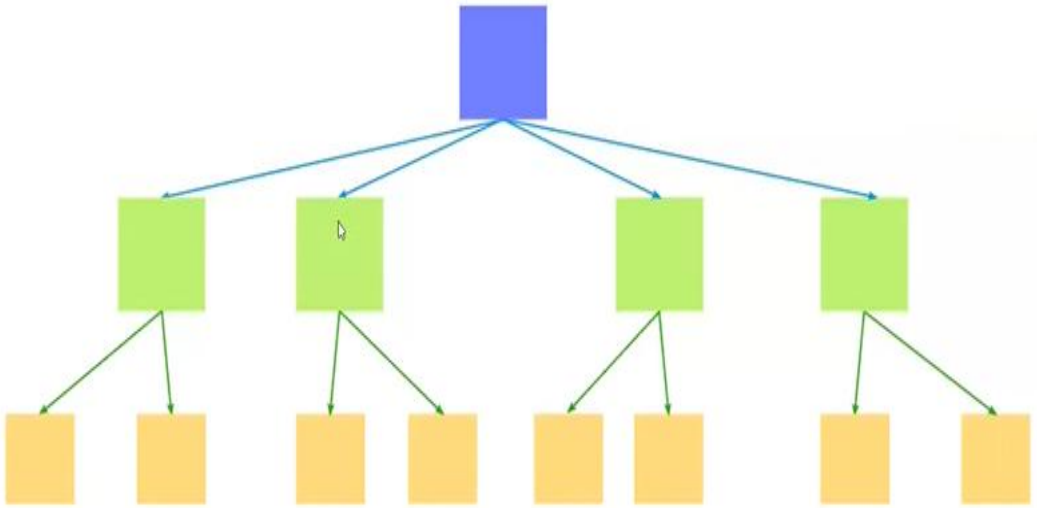
Among the seven data models described here, the first four types were used in the early days of databases and are still options, followed by the three models most widely used now.

# Application of Modelling in Business

- Data modelling is nothing but a process through which data is stored structurally in a format in a database. Data modelling is important because it enables organizations to make data-driven decisions and meet varied business goals.
- You are required to have a deeper understanding of the structure of an organization and then propose a solution that aligns with its end-goals and suffices it in achieving the desired objectives.

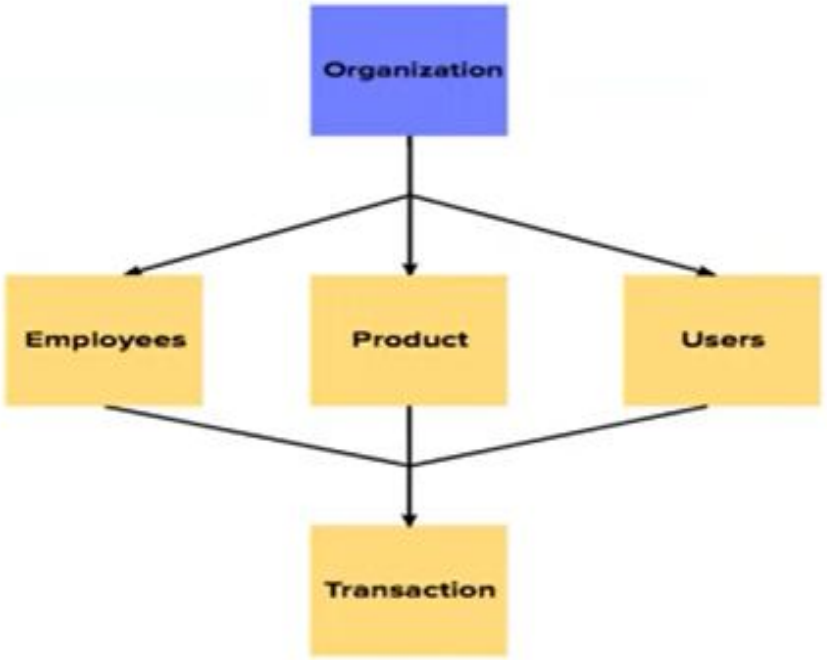
## 1. Hierarchical data model

Data is stored in a tree-like structure with parent and child records that comprise a collection of data fields. A parent can have one or more children, but a child record can have only one parent. The hierarchical model is also composed of links, which are the connections between records, and the types that specify the kind of data contained in the field. It originated in mainframe databases in the 1960s.



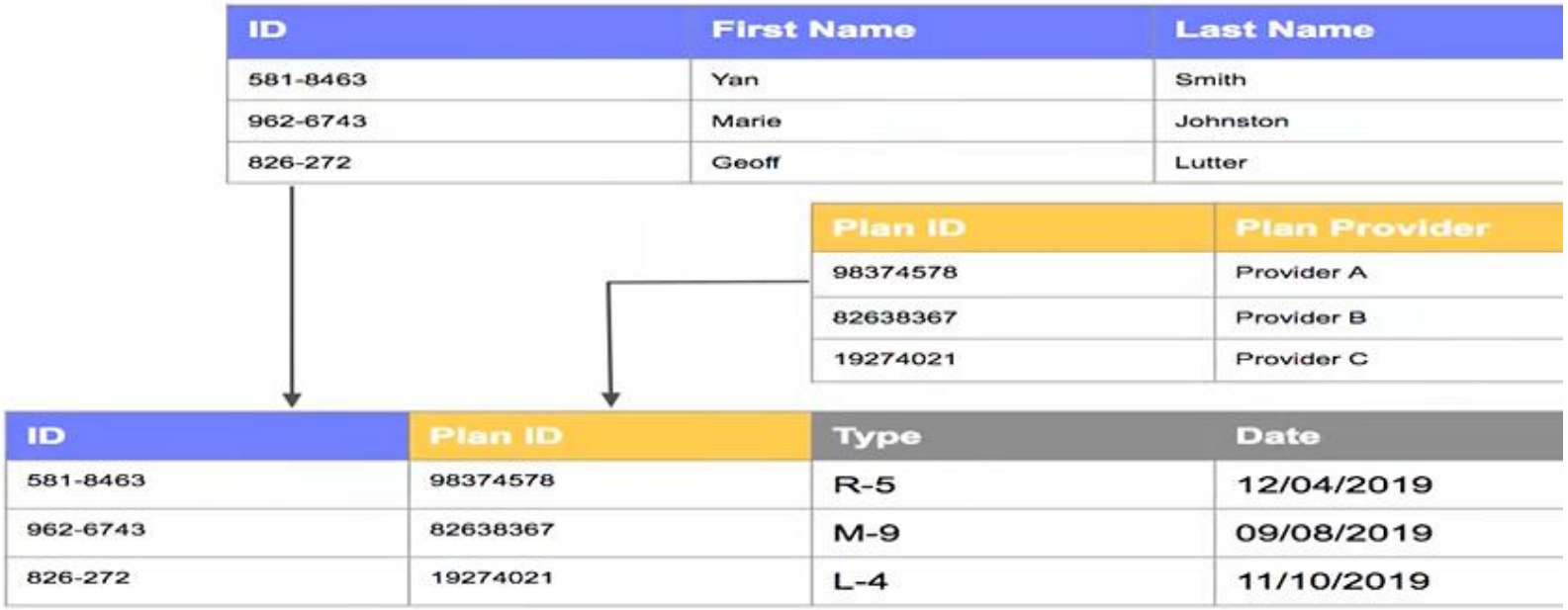
## 2. Network data model

This model extended the hierarchical model by allowing a child record to have one or more parents. A standard specification of the network model was adopted in 1969 by the Conference on Data Systems Languages, a now-defunct group better known as CODASYL. As a result, it's also referred to as the CODASYL model. The network technique is the precursor to a graph data structure, with a data object represented inside a node and the relationship between two nodes called an edge.



## 3. Relational data model

In this model, data is stored in tables and columns and the relationships between the data elements in them are identified. It also incorporates database management features such as constraints and triggers. The relational approach became the dominant data modeling technique during the 1980s. The entity-relationship and dimensional data models, currently the most prevalent techniques, are variations of the relational model but can also be used with non-relational databases.



| ID | First Name | Last Name |
|---|---|---|
| 581-8463 | Yan | Smith |
| 962-6743 | Marie | Johnston |
| 826-272 | Geoff | Lutter |

| Plan ID | Plan Provider |
|---|---|
| 98374578 | Provider A |
| 82638367 | Provider B |
| 19274021 | Provider C |

| ID | Plan ID | Type | Date |
|---|---|---|---|
| 581-8463 | 98374578 | R-5 | 12/04/2019 |
| 962-6743 | 82638367 | M-9 | 09/08/2019 |
| 826-272 | 19274021 | L-4 | 11/10/2019 |

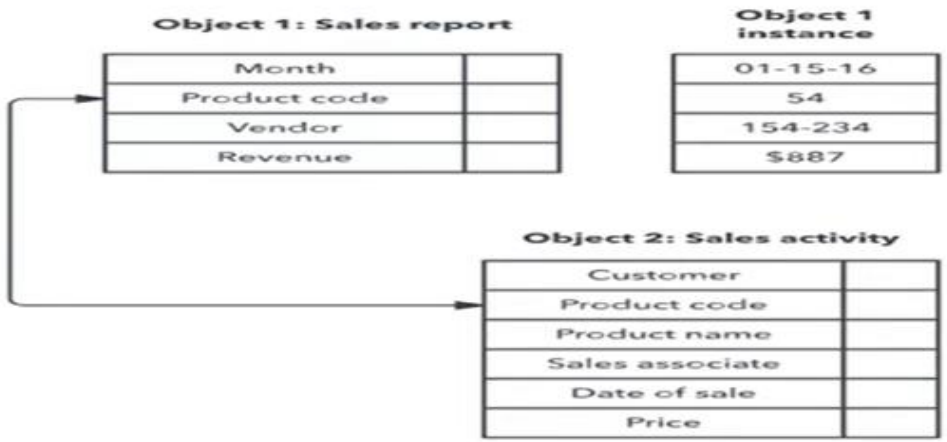
## 4. Object-oriented data model

This model combines aspects of object-oriented programming and the relational data model. The object-oriented model is also composed of the following:

Classes, which are collections of similar objects with shared attributes and behaviors.

Inheritance, which enables a new class to inherit the attributes and behaviors of an existing object.

It was created for use with object databases, which emerged in the late 1980s and early 1990s as an alternative to relational software. But they didn't make a big dent in relational technology's dominance.



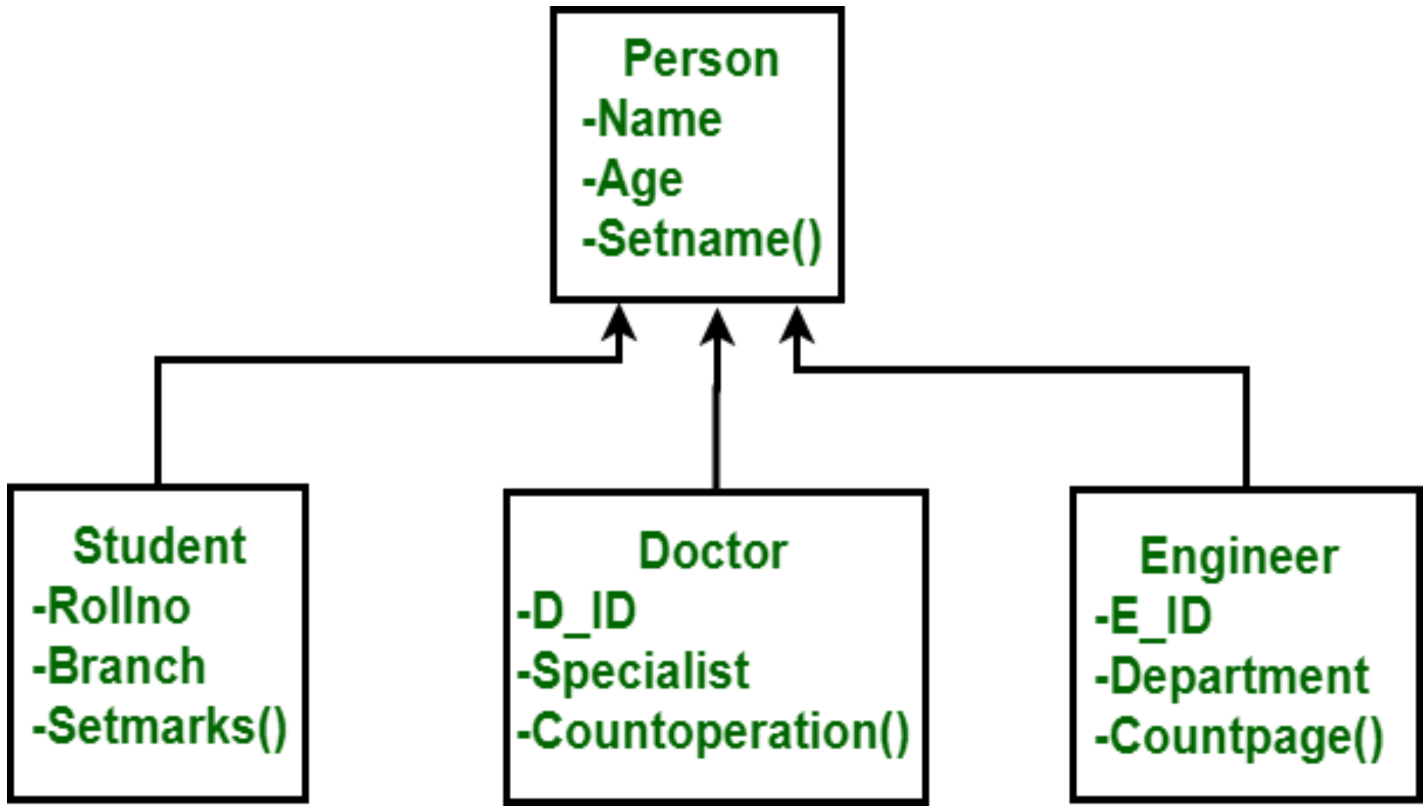
## 5. Entity-relationship data model

This model has been widely adopted for relational databases in enterprise applications, particularly for transaction processing. With minimal redundancy and well-defined relationships, it's very efficient for data capture and update processes. The model consists of the following:

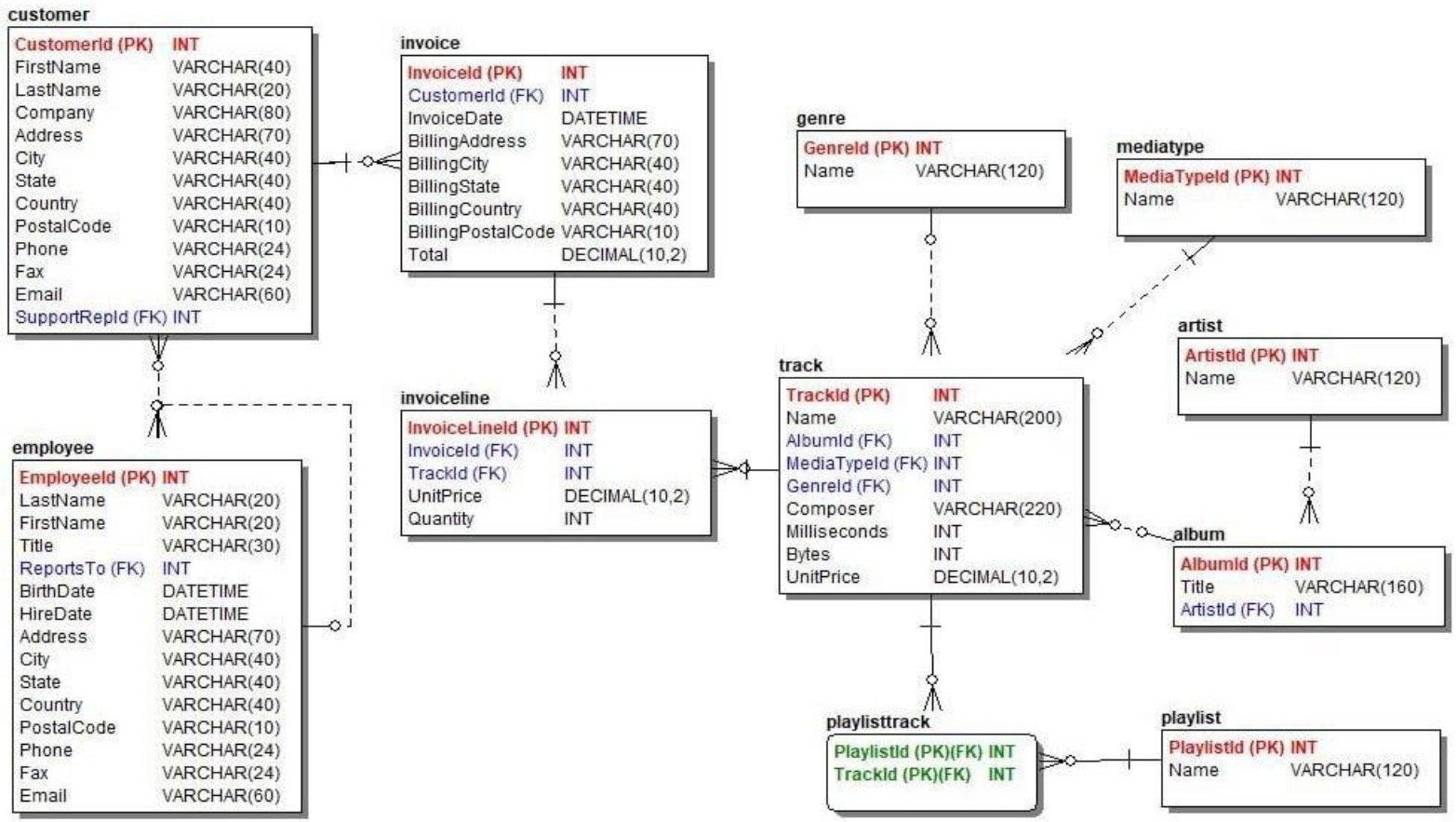
Entities that represent people, places, things, events or concepts on which data is processed and stored as tables.

Attributes, which are distinct characteristics or properties of an entity that are maintained and stored as data in columns.

Relationships, which define logical links between two entities that represent business rules or constraints.


Person
-Name
-Age
-Setname()
Student
-Rollno
-Branch
-Setmarks()
Doctor
-D_ID
-Specialist
-Countoperation()
Engineer
-E_ID
-Department
-Countpage()

# Sample entity-relationship data model



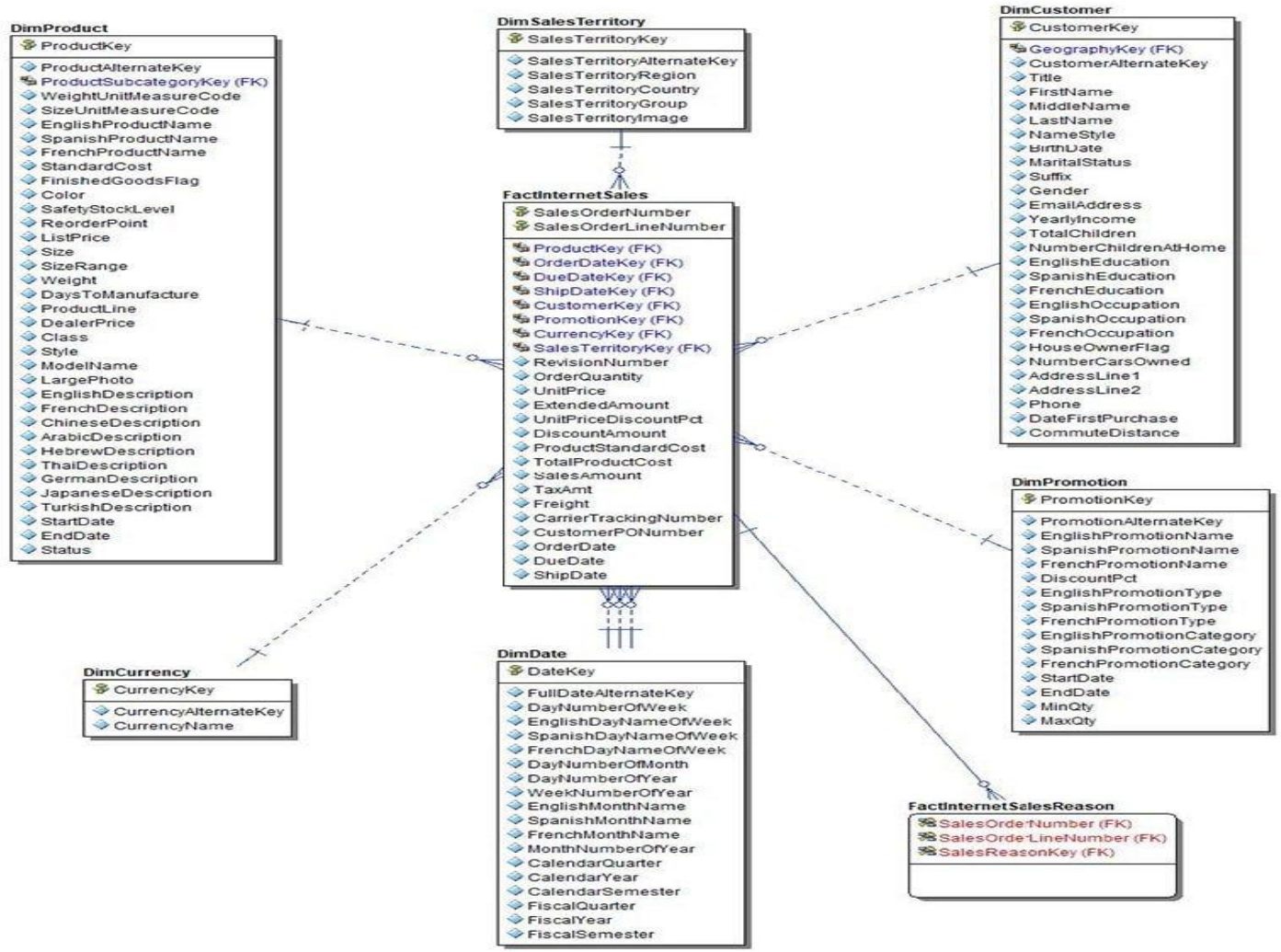
## 6. Dimensional data model

Like the entity-relationship model, the dimensional model includes attributes and relationships. But it features two core components.

**Facts**, which are measurements of an activity, such as a business transaction or an event involving a person or devices. Facts are generally numeric, and fact tables are normalized and contain little redundancy.

**Dimensions**, which are tables that contain the business context of the facts to define their who, what, where and why attributes. Dimensions are typically descriptive instead of numeric.

# Sample dimensional data model



## 7. Graph data model

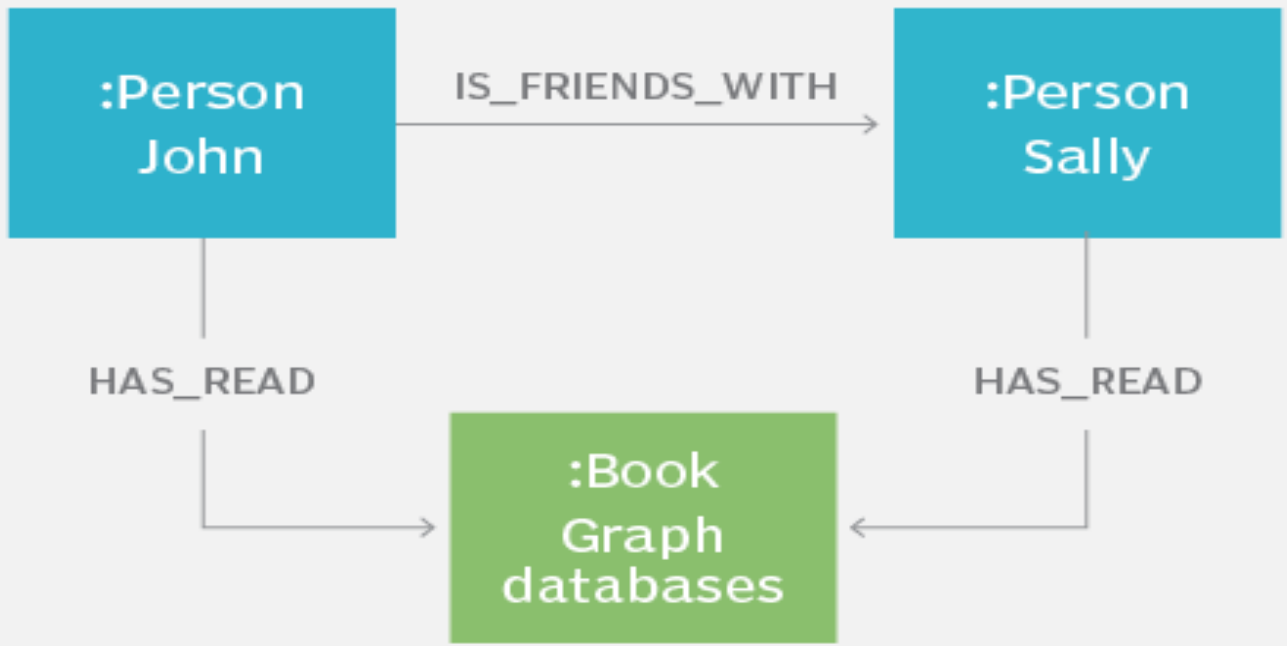
Graph data modeling has its roots in the network modeling technique. It's primarily used to model complex relationships in graph databases, but it can also be used for other NoSQL databases such as key-value and document types.

There are two fundamental elements in a graph data model.

**Nodes**, which represent entities with a unique identity. Each instance of an entity is a different node that's akin to a row in a table in the relational model.

**Edges**, also known as links or relationships. They connect nodes and define how the nodes are related. All nodes must have at least one edge connected to them. Edges can be undirected, with a bidirectional relationship between nodes, or directed, in which the relationship goes in a specified direction.

# Sample graph data model




SOURCE: NEO4J

# Application of modelling in Business

Modeling in business, particularly in the realm of data analytics, is pervasive and plays a crucial role in various aspects of decision-making and strategy development. Here are some key applications:

1. **Predictive Analytics**:
   - Predictive modeling uses historical data to predict future outcomes. In business, this could involve forecasting sales, customer behavior, market trends, or even operational metrics.
   - Businesses can use predictive models to anticipate customer needs, optimize inventory levels, manage supply chains more efficiently, and mitigate risks.
2. **Customer Segmentation**:
   - Businesses often use clustering techniques to segment customers based on their behavior, demographics, or preferences.
   - These segments can then be targeted with tailored marketing campaigns, product recommendations, or pricing strategies.
3. **Churn Prediction**:
   - Churn modeling helps businesses identify customers who are likely to leave or stop using their services.
   - By understanding the factors that contribute to churn, companies can take proactive measures to retain customers, such as offering incentives or improving service quality.

**4. Recommendation Systems**:

- Recommendation engines use algorithms to suggest products, services, or content to users based on their past behavior, preferences, or similarities with other users.
- Businesses across various sectors, including e-commerce, media streaming, and online advertising, leverage recommendation systems to personalize user experiences and drive engagement.

**5. Fraud Detection**:

- Modeling techniques can be applied to detect fraudulent activities in financial transactions, insurance claims, or online interactions.
- By analyzing patterns and anomalies in data, businesses can flag potentially fraudulent behavior and take appropriate measures to mitigate risks.

**6. Optimization**:

- Optimization models help businesses make better decisions regarding resource allocation, production planning, scheduling, and logistics.
- By formulating mathematical models that consider constraints and objectives, companies can optimize processes to minimize costs, maximize efficiency, or achieve other desired outcomes.

**7. Market Basket Analysis**:

- Market basket analysis examines the relationships between products that customers purchase together.
- By understanding these associations, businesses can optimize product placement, cross-selling strategies, and promotional efforts to increase sales and customer satisfaction.

# Types of data and variables

- An attribute is a data field, representing a characteristic or feature of a data object. The nouns attribute, dimension, feature, and variable are often used interchangeably in the literature. The term is commonly used in data warehousing. Machine learning literature tends to use dimensions term feature, while statisticians prefer the term variable. Data mining and database professionals commonly use the term attribute, and we do here as well. Attributes describing a customer object can include, for example, customer ID, name, and address.
- **Qualitative Data Type**
  - Nominal
  - Ordinal
- **Quantitative Data Type**
  - Discrete
  - Continuous

# Qualitative Data Type

- Qualitative or Categorical Data describes the object under consideration using a finite set of discrete classes. It means that this type of data can’t be counted or measured easily using numbers and therefore divided into categories. The gender of a person (male, female, or others) is a good example of this data type.
- These are usually extracted from audio, images, or text medium. Another example can be a smartphone brand that provides information about the current rating, the color of the phone, the category of the phone, and so on. All this information can be categorized as Qualitative data. There are two subcategories under this:

    - **Nominal**
    - **Ordinal**

## Nominal

- These are the set of values that don’t possess a natural ordering. Let’s understand this with some examples. The color of a smartphone can be considered as a nominal data type as we can’t compare one color with others.
- It is not possible to state that ‘Red’ is greater than ‘Blue’. The gender of a person is another one where we can’t differentiate between male, female, or others. Mobile phone categories whether it is midrange, budget segment, or premium smartphone is also nominal data type.
- Nominal data types in statistics are not quantifiable and cannot be measured through numerical units. Nominal types of statistical data are valuable while conducting qualitative research as it extends freedom of opinion to subjects.

# Ordinal

- These types of values have a natural ordering while maintaining their class of values. If we consider the size of a clothing brand then we can easily sort them according to their name tag in the order of small < medium < large. The grading system while marking candidates in a test can also be considered as an ordinal data type where A+ is definitely better than B grade.
- These categories help us deciding which encoding strategy can be applied to which type of data. Data encoding for Qualitative data is important because machine learning models can’t handle these values directly and needed to be converted to numerical types as the models are mathematical in nature.
- For nominal data type where there is no comparison among the categories, one-hot encoding can be applied which is similar to binary coding considering there are in less number and for the ordinal data type, label encoding can be applied which is a form of integer encoding.

| Aspect | Nominal Data | Ordinal Data |
| --- | --- | --- |
| Definition | Categories data into distinct classes or categories without any inherent order or ranking. | Categories data into ordered or ranked categories with meaningful differences between them. |
| Examples | Colors, gender, types of animals | Education levels, customer satisfaction ratings |
| Mathematical Operations | No meaningful mathematical operations can be performed (e.g., averaging categories). | Limited mathematical operations can be performed, such as determining the mode or median. |
| Order/ Ranking | No natural or meaningful order exists. | Categories have a specific order or ranking, but the magnitude of differences between ranks may not be uniform. |
| Central Tendency | Mode (most frequent category) | Mode, median (middle category), but mean is not typically used due to lack of uniform interval between ranks. |
| Example Use Case | Classifying objects, grouping data | Rating scales, survey responses, educational levels |

## Quantitative Data Type

- This data type tries to quantify things and it does by considering numerical values that make it countable in nature. The price of a smartphone, the discount offered, the number of ratings on a product, the frequency of the processor of a smartphone, or the RAM of that particular phone, all these things fall under the category of Quantitative data types.
- The key thing is that there can be an infinite number of values a feature can take. For instance, the price of a smartphone can vary from x amount to any value and it can be further broken down based on fractional values. The two subcategories that describe them are:

- Discrete
- Continuous

## Discrete

- The numerical values which fall under integers or whole numbers are placed under this category. The number of speakers in the phone, cameras, cores in the processor, and the number of SIMs supported are some examples of the discrete data type.
- Discrete data types in statistics cannot be measured – they can only be counted as the objects included in discrete data have a fixed value. The value can be represented in decimal, but it has to be whole. Discrete data is often identified through charts, including bar charts, pie charts, and tally charts.

## Continuous

- The fractional numbers are considered as continuous values. These can take the form of the operating frequency of the processors, the Android version of the phone, the wifi frequency, the temperature of the cores, and so on.
- Unlike discrete data types of data in research, with a whole and fixed value, continuous data can break down into smaller pieces and can take any value. For example, volatile values such as temperature and the weight of a human can be included in the continuous value. Continuous types of statistical data are represented using a graph that easily reflects value fluctuation by the highs and lows of the line through a certain period.

| Aspect | Discrete Data | Continuous Data |
| --- | --- | --- |
| Definition | Consists of distinct, separate values. | It can take any value within a given range. |
| Examples | Number of students in a class, coin toss outcomes (1, 2, 3), customer count. | Height, weight, temperature, time. |
| Nature | Usually involves whole numbers or counts. | Involves any value along a continuous spectrum. |
| Gaps in values | Gaps between values are common and meaningful. | Values can be infinitely divided without gaps. |
| Measurement | Often measured using integers. | Measured with decimal numbers or fractions. |
| Graphical representation | Typically represented with bar charts or histograms. | Represented with line graphs or smooth curves. |
| Mathematical Operations | Typically involves counting or summation. | Involves arithmetic operations, including fractions and decimals. |
| Probability Distribution | Typically represented using probability mass functions | Typically represented using probability density functions. |
| Example Use Case | Counting occurrences, tracking integers. | Measuring quantities and analyzing measurements. |

# Binary attributes or variables

- A binary attribute is a nominal attribute with only two categories or states: 0 or 1, where 0 typically means that the attribute is absent, and 1 means that it is present. Binary attributes are referred to as Boolean if the two states correspond to true and false.

Ex:

Binary attributes. Given the attribute smoker describing a patient object,

‘1’ indicates that the patient smokes, while ‘0’ indicates that the patient does not. Similarly, suppose the patient undergoes a medical test that has two possible outcomes. The attribute medical test is binary, where a value of 1 means the result of the test for the patient is positive, while 0 means the result is negative.

# Binary attribute: symmetric vs asymmetric

- A binary attribute is **symmetric** if both of its states are equally valuable and carry the same weight; that is, there is no preference on which outcome should be coded as 0 or 1. One such example could be the attribute gender having the states male and female.

- A binary attribute is **asymmetric** if the outcomes of the states are not equally important, such as the positive and negative outcomes of a medical test for HIV. By convention, we code the most important outcome, which is usually the rarest one, by 1 (e.g., HIV positive) and the other by 0 (e.g., HIV negative)

# Data Modelling

- Data modeling is the process of planning a structure to represent how information and relationships between information will be stored in your system.
- **Data modeling** is the process of creating a data model for the data to be stored in a database. This data model is a conceptual representation of Data objects, the associations between different data objects, and the rules.

  Data modeling helps in the visual representation of data and enforces business rules, regulatory compliances, and government policies on the data. Data Models ensure consistency in naming conventions, default values, semantics, security while ensuring quality of the data.

# Why is Data Modeling Important?

- Creating a data model is an important step in application development. Data modeling will force your team to make decisions about what data is necessary and how to collect and structure it.
- In fact, you can think of a data model as simply a “set of decisions”, assertions, and assumptions. Even if something is modeled incorrectly, those assumptions are written down, and help the team in the future to understand why it was modeled that way. With this baseline of information, the team in the future can more carefully consider if making a change is the right course of action.

# Conceptual vs Logical vs Physical Data Model

- There are three traditional levels of data modeling.

  Not every team will necessarily follow all three strictly. Often, all three – conceptual, logical, and physical data models – are compressed into one modeling exercise.

- However, breaking the process down into these three levels can be valuable. Each step lays down a foundation for the next:
  1. Conceptual – the “what” model
  2. Logical – the “how” of the details
  3. Physical – the “how” of the implementation
- Each level of conceptual, physical, and logical data models can involve different roles from your team.

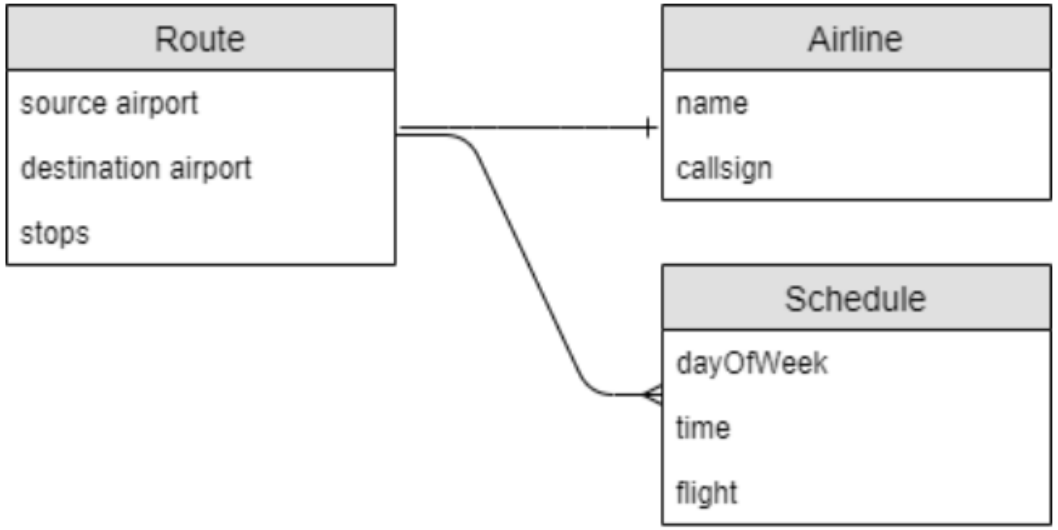
# Conceptual model

- The conceptual data model can be thought of as the “whiteboard” data model. This model does not go into the “how” at all.
- For this model, it’s important to focus on capturing all the types of data (or “entities”) that the system will need. In addition to entities, a conceptual data model will also capture:
  - **Attributes:** individual properties of an entity. For instance, a “person” entity may have “name” and “shoe size”. An “address” entity may have “zip code” and “city”.
  - **Relationships:** how an entity connects to other entities. For instance, a “person” entity may have one or more “addresses”.
- Along with the entities, their attributes, and relationships, a conceptual model can also:
  - **Organize scope:** which entities are included, but also which are explicitly NOT included.
  - **Define business concepts/rules:** For instance, are person entities allowed to have multiple addresses? What about multiple emails? Do they need to have a unique identifier?
- The conceptual data model is often created by architects in conjunction with business stakeholders and domain experts.

# Conceptual Data Model Example

There are many “languages” for describing a conceptual data model. But as long as it’s documented in an accessible way, it can be as easy as boxes and arrows.

Here’s an example of a conceptual diagram that involves two core entities, *travel routes* (and its associated schedules) and *airlines*:



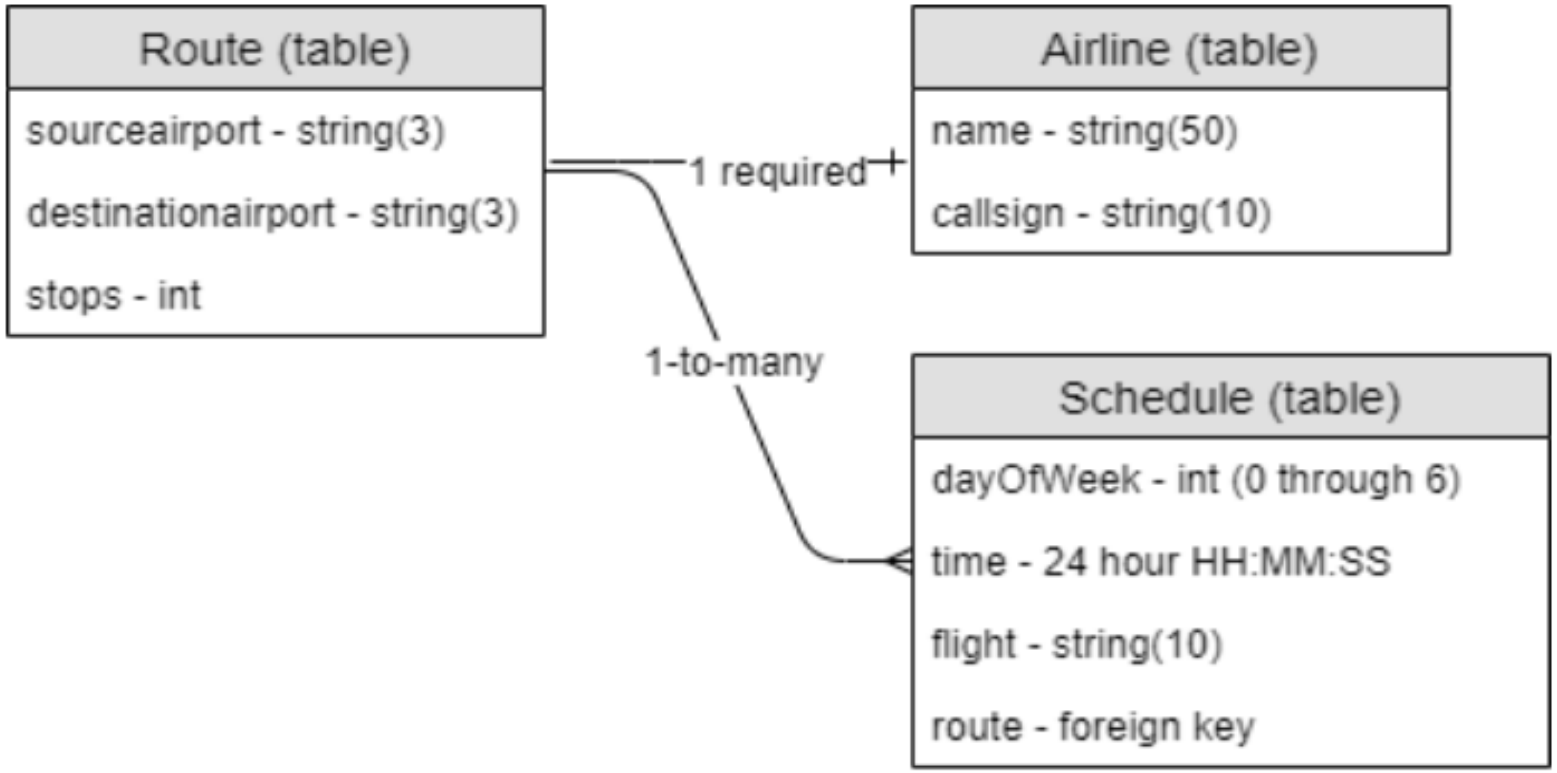

While these may look like tables in a relational database, the conceptual modeling stage is too early to make a determination about *how* the data will be stored. That determination comes later: it could be tables, JSON documents, graph nodes, CSV files, blockchain, or any other number of storage mediums.

# Logical Data Model

- A logical data model is the next step, once the stakeholders have agreed on a conceptual model.
- This step involves filling in the *details* of the conceptual model. It's still too early to pick a specific DBMS, but this step can help you decide which *class* of database to use (relational, document, etc). For instance, if you decide **relational**, then it's time to decide which tables to create. If you decide to **document**, then it's time to define the collections.
- Decide the details of each individual field/column and relationship as well. This includes data types, sizes, lengths, arrays, nested objects, etc.
- The logical model is typically created by architects and business analysts.

# Logical Data Model Example

For instance, if going with a relational model, the logical model might look like this:

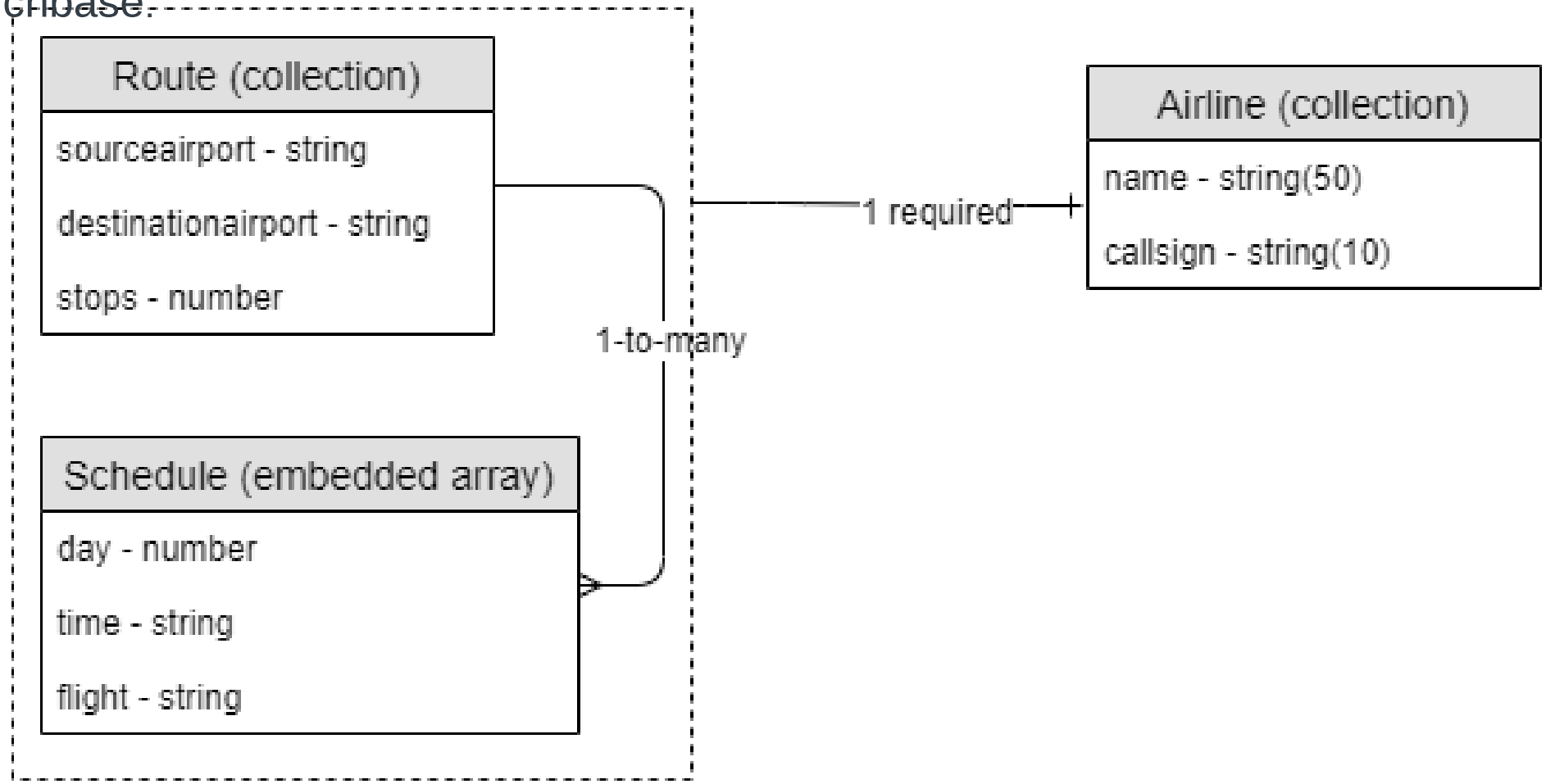


# Physical Data Model

- Once a logical model has been defined, it's now time to actually implement it into a real database.
- If you decided on a relational model, options include SQL Server, Oracle, PostgreSQL, MySQL, etc. However, if your modeling process reveals that your data model is likely to change frequently to adapt to new requirements, you might still consider going with a document database. One of the best choices for this is Couchbase, a "NoSQL" document database that supports familiar relational concepts like JOINs, ACID transactions , and flexible JSON data.
- The physical data model should include:
  - A specific DBMS (Couchbase, for instance)
  - How data is stored (On disk/RAM/hybrid/etc. Couchbase has a built-in cache to provide the speed of RAM with the durability of disk)
  - How to accommodate replications, shards, partitions, etc. (For Couchbase, sharding and partitioning is automatic. Replication is a drop-down box to select how many replicas you want).
- The physical data model is typically created by DBAs and/or developers.

# Physical Data Model Example

Here's an example of a physical model for Couchbase:



It's sometimes helpful to show sample data along with the physical model.

Here's a sample route document:

```
{
  "airlineid": "airline_137",
  "sourceairport": "TLV",
  "destinationairport": "MRS",
  "stops": 0,
  "schedule": [
    { "day": 0, "utc": "10:13:00", "flight": "AF198" },
    { "day": 0, "utc": "01:31:00", "flight": "AF943" },
    { "day": 1, "utc": "12:40:00", "flight": "AF356" },
    // ... etc ...
  ]
}
```

And here's a sample airline document:

```
key: airline_137
{
  "name": "Air France",
  "callsign": "AIRFRANS",
}
```

# Types of Data Models

- There are mainly three different types of data models: conceptual data models, logical data models, and physical data models, and each one has a specific purpose. The data models are used to represent the data and how it is stored in the database and to set the relationship between data items.

1. **Conceptual Data Model:** This Data Model defines **WHAT** the system contains. This model is typically created by Business stakeholders and Data Architects. The purpose is to organize, scope and define business concepts and rules.
2. **Logical Data Model:** Defines **HOW** the system should be implemented regardless of the DBMS. This model is typically created by Data Architects and Business Analysts. The purpose is to developed technical map of rules and data structures.
3. **Physical Data Model**: This Data Model describes **HOW** the system will be implemented using a specific DBMS system. This model is typically created by DBA and developers. The purpose is actual implementation of the database.

# Conceptual Data Model

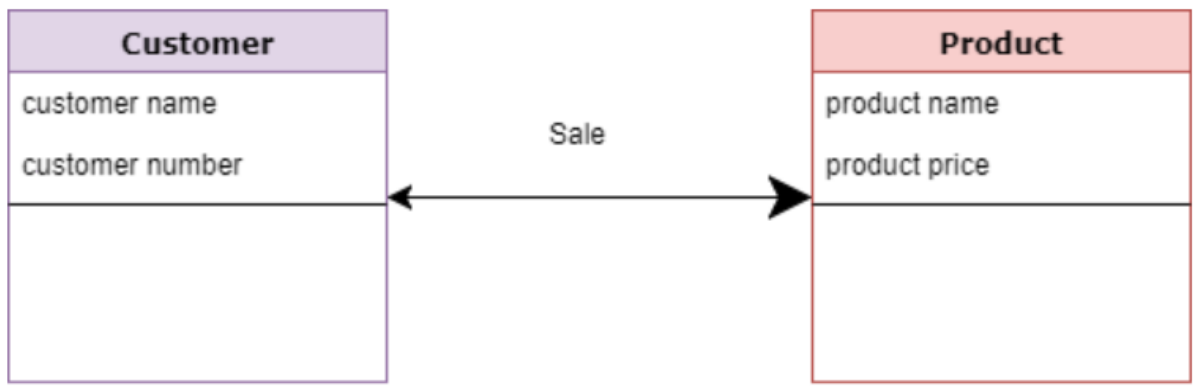
A **Conceptual Data Model** is an organized view of database concepts and their relationships. The purpose of creating a conceptual data model is to establish entities, their attributes, and relationships. In this data modeling level, there is hardly any detail available on the actual database structure. Business stakeholders and data architects typically create a conceptual data model.

The 3 basic tenants of Conceptual Data Model are

- **Entity**: A real-world thing
- **Attribute**: Characteristics or properties of an entity
- **Relationship**: Dependency or association between two entities

Data model example:

- Customer and Product are two entities. Customer number and name are attributes of the Customer entity
- Product name and price are attributes of product entity
- Sale is the relationship between the customer and product




Conceptual Data Model

# Logical Data Model

The **Logical Data Model** is used to define the structure of data elements and to set relationships between them. The logical data model adds further information to the conceptual data model elements. The advantage of using a Logical data model is to provide a foundation to form the base for the Physical model. However, the modeling structure remains generic.


Logical Data Model

At this Data Modeling level, no primary or secondary key is defined. At this Data modeling level, you need to verify and adjust the connector details that were set earlier for relationships.

# Physical Data Model

A **Physical Data Model** describes a database-specific implementation of the data model. It offers database abstraction and helps generate the schema. This is because of the richness of meta-data offered by a Physical Data Model. The physical data model also helps in visualizing database structure by replicating database column keys, constraints, indexes, triggers, and other RDBMS features.


Physical Data Model

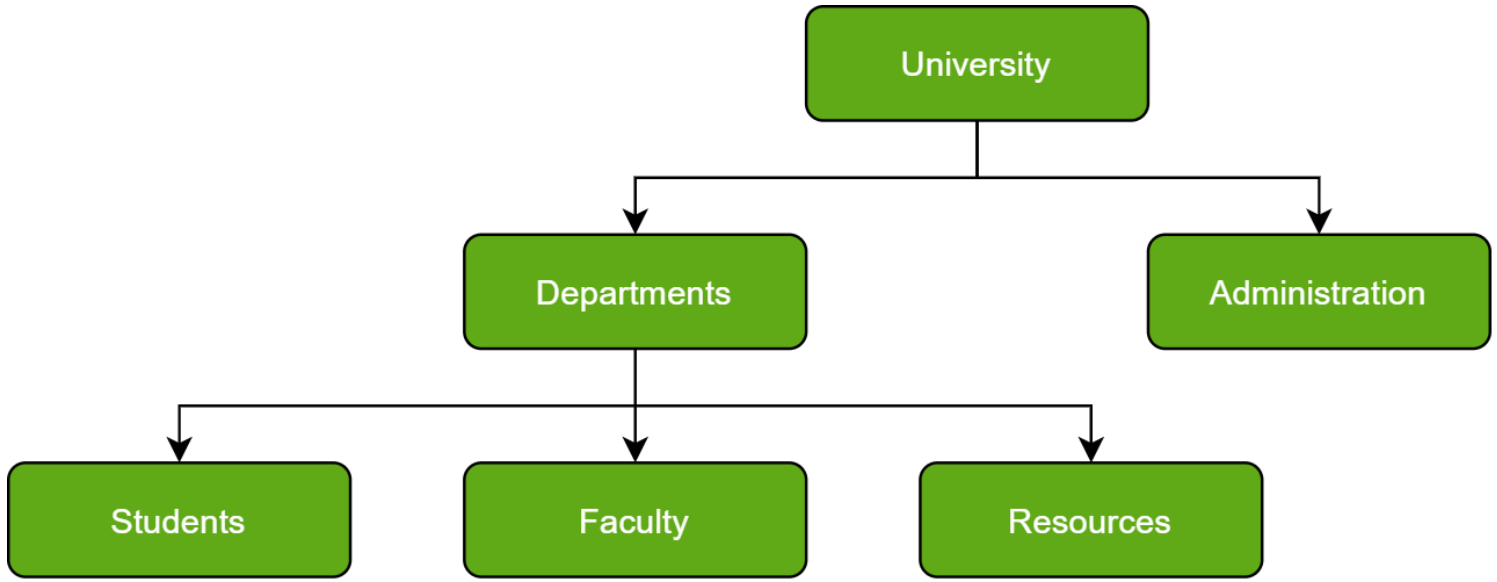
# Types of Databases

There are several types of databases, that are briefly explained below.

- Hierarchical databases
- Network databases
- Object-oriented databases
- Relational databases
- Cloud Database
- Centralized Database
- Operational Database
- NoSQL databases

# Hierarchical Databases

- Just as in any hierarchy, this database follows the progression of data being categorized in ranks or levels, wherein data is categorized based on a common point of linkage. As a result, two entities of data will be lower in rank and the commonality will assume a higher rank. Refer to the diagram below:

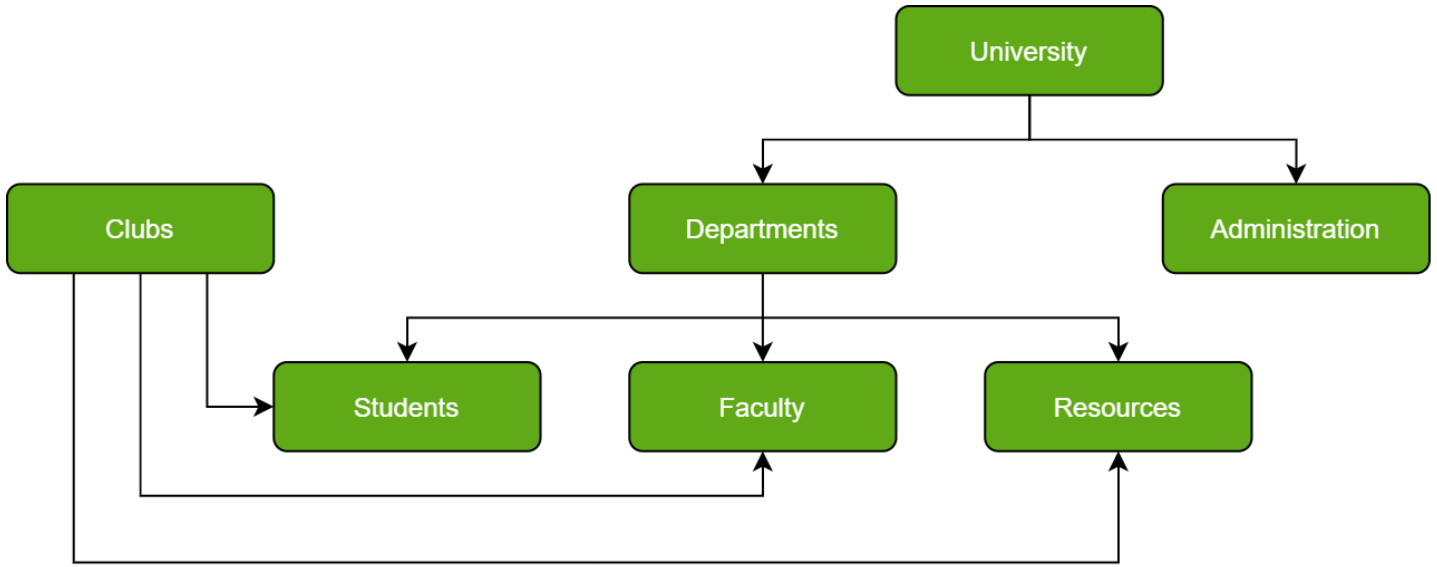


# Hierarchical Databases

- Do note how Departments and Administration are entirely unlike each other and yet fall under the domain of a University. They are elements that form this hierarchy.
- Another perspective advises visualizing the data being organized in a parent-child relationship, which upon the addition of multiple data elements would resemble a tree. The child records are linked to the parent record using a field, and so the parent record is allowed multiple child records. However, vice versa is not possible.
- Notice that due to such a structure, hierarchical databases are not easily scalable; the addition of data elements requires a lengthy traversal through the database.

# Network Databases

- In Layman's terms, a network database is a hierarchical database, but with a major tweak. The child records are given the freedom to associate with multiple parent records. As a result, a network or net of database files linked with multiple threads is observed. Notice how the Student, Faculty, and Resources elements each have two parent records, which are Departments and Clubs.

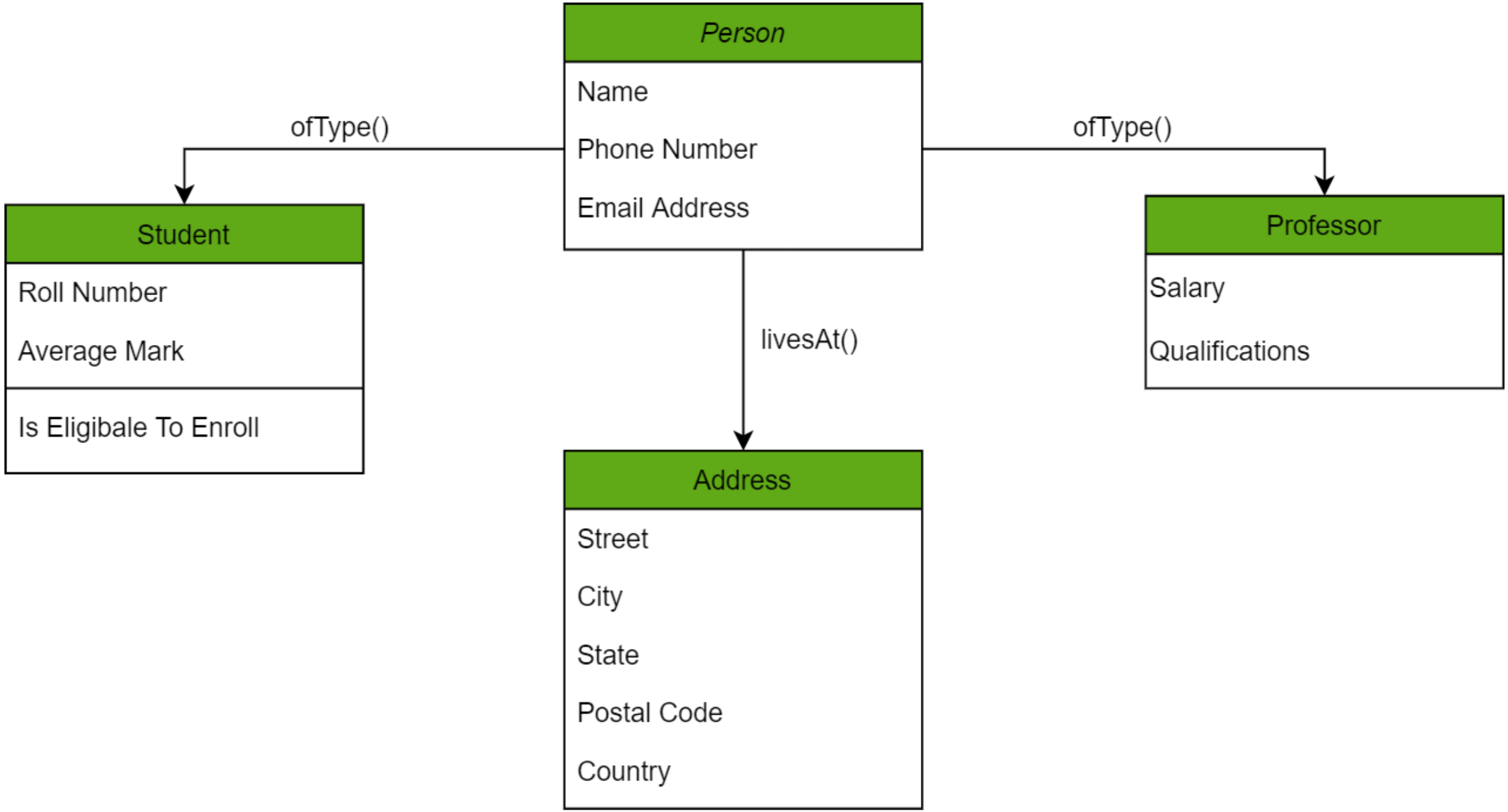


# Network Databases

- Certainly, in a complex framework, network databases are more capable of representing two-directional relationships. Also, conceptual simplicity favors the utilization of a simpler database management language.
- The disadvantage lies in the inability to alter the structure due to its complexity and also in it being highly structurally dependent.

# Object-Oriented Databases

- Those familiar with the Object-Oriented Programming Paradigm would be able to relate to this model of databases easily. Information stored in a database is capable of being represented as an object that responds as an instance of the database model. Therefore, the object can be referenced and called without any difficulty. As a result, the workload on the database is substantially reduced.


Person
Name
Phone Number
Email Address
ofType()
ofType()
Student
Roll Number
Average Mark
Is Eligibale To Enroll
livesAt()
Professor
Salary
Qualifications
Address
Street
City
State
Postal Code
Country

- In the chart above, we have different objects linked to one another using methods; one can get the address of the Person (represented by the Person Object) using the latest () method. Furthermore, these objects have attributes which are the data elements that need to be defined in the database.
- An example of such a model is the Berkeley DB software library which uses the same conceptual background to deliver quick and highly efficient responses to database queries from the embedded database.

# Relational Databases

- Considered the most mature of all databases, these databases lead in the production line along with their management systems. In this database, every piece of information has a relationship with every other piece of information. This is on account of every data value in the database having a unique identity in the form of a record.
- Note that all data is tabulated in this model. Therefore, every row of data in the database is linked with another row using a primary key. Similarly, every table is linked with another table using a foreign key.
- Refer to the diagram below and notice how the concept of ‘Keys’ is used to link two tables.

| Roll no. | Student Name | Marks Awarded |
|---|---|---|
| 1 | Raman Triphati | 86 |
| 2 | Rajan Govindan | 94 |
| 3 | Mahesh Nandalal | 94 |

Key = 94

| Marks Awarded | Student Name | Rank | Scholarship |
|---|---|---|---|
| 94 | Rajan Govindan | 17 | Yes |
| 94 | Mahesh Nandlal | 16 | Yes |

| Section | Student Name | Marks Awarded | Rank |
|---|---|---|---|
| A | Raman Tripathi | 86 | 43 |
| B | Rajan Govindan | 94 | 17 |
| C | Mahesh Nandlal | 94 | 16 |

- Due to this introduction of tables to organize data, it has become exceedingly popular. In consequence, they are widely integrated into Web-Ap interfaces to serve as ideal repositories for user data. What makes it further interesting is the ease in mastering it, since the language used to interact with the database is simple (SQL in this case) and easy to comprehend.
- It is also worth being aware of the fact that in Relational databases, scaling and traversing through data is quite a lightweight task in comparison to Hierarchical Databases.

# Cloud Databases

- A cloud database is used where data requires a virtual environment for storing and executing over the cloud platforms and there are so many cloud computing services for accessing the data from the databases (like SaaS, PaaS, etc).

There are some names of cloud platforms are-

- Amazon Web Services (AWS)
- Google Cloud Platform (GCP)
- Microsoft Azure
- ScienceSoft, etc.

# Centralized Databases

- A centralized database is a type of database that is stored, located as well as maintained at a single location and it is more secure when the user wants to fetch the data from the Centralized Database.
- **Advantages**
- Data Security
- Reduced Redundancy
- Consistency
- **Disadvantages**
- The size of the centralized database is large which increases the response and retrieval time.
- It is not easy to modify, delete and update.

# Personal Databases

- Collecting and Storing the data on its System and this type of database is designed for the single user.
- **Advantages**
- It is easy to handle
- It occupies less space

# Operational Databases

- It is used for creating, updating, and deleting the database in real-time and it is designed for executing and handling the daily data operation in organizations and businesses purposes.
- **Advantages**
- easy to fetch.
- Structured data
- Real-time processing

# NoSQL Databases

- A NoSQL originally referring to non SQL or non-relational is a database that provides a mechanism for storage and retrieval of data. This data is modeled in means other than the tabular relations used in relational databases.
- A NoSQL database includes simplicity of design, simpler horizontal scaling to clusters of machines, and finer control over availability. The data structures used by NoSQL databases are different from those used by default in relational databases which makes some operations faster in NoSQL.
- MongoDB falls in the category of NoSQL document-based database.

- **Advantages of NoSQL**
- There are many advantages of working with NoSQL databases such as MongoDB and Cassandra. The main advantages are high scalability and high availability.

- **Disadvantages of NoSQL**
- NoSQL has the following disadvantages.
- NoSQL is an open-source database.
- GUI is not available
- Backup is a weak point for some NoSQL databases like MongoDB.
- Large document size.

# What Kinds of Data Can Be Mined?

- As a general technology, data mining can be applied to any kind of data as long as the data are meaningful for a target application. The most basic forms of data for mining applications are **database data**, **data warehouse data**, and **transactional data**.
- Data mining can also be applied to other forms of data (e.g., data streams, ordered/sequence data, graph or networked data, spatial data, text data, multimedia data, and the WWW).

## 1.3.1 Database Data

A database system, also called a **database management system** (**DBMS**), consists of a collection of interrelated data, known as a **database**, and a set of software programs to manage and access the data. The software programs provide mechanisms for defining database structures and data storage; for specifying and managing concurrent, shared, or distributed data access; and for ensuring consistency and security of the information stored despite system crashes or attempts at unauthorized access.

A **relational database** is a collection of **tables**, each of which is assigned a unique name. Each table consists of a set of **attributes** (*columns* or *fields*) and usually stores a large set of **tuples** (*records* or *rows*). Each tuple in a relational table represents an object identified by a unique *key* and described by a set of attribute values. A semantic data model, such as an **entity-relationship** (**ER**) data model, is often constructed for relational databases. An ER data model represents the database as a set of entities and their relationships.

**Example 1.2** **A relational database for *AllElectronics*.** The fictitious *AllElectronics* store is used to illustrate concepts throughout this book. The company is described by the following relation tables: *customer, item, employee*, and *branch*. The headers of the tables described here are shown in Figure 1.5. (A header is also called the *schema* of a relation.)

- The relation *customer* consists of a set of attributes describing the customer information, including a unique customer identity number (*cust_ID*), customer name, address, age, occupation, annual income, credit information, and category.
- Similarly, each of the relations *item*, *employee*, and *branch* consists of a set of attributes describing the properties of these entities.
- Tables can also be used to represent the relationships between or among multiple entities. In our example, these include *purchases* (customer purchases items, creating a sales transaction handled by an employee), *items_sold* (lists items sold in a given transaction), and *works_at* (employee works at a branch of *AllElectronics*). ■

| | |
|---|---|
| *customer* | (*cust_ID, name, address, age, occupation, annual_income, credit_information, category*, . . . ) |
| *item* | (*item_ID, brand, category, type, price, place_made, supplier, cost*, . . . ) |
| *employee* | (*empl_ID, name, category, group, salary, commission*, . . . ) |
| *branch* | (*branch_ID, name, address*, . . . ) |
| *purchases* | (*trans_ID, cust_ID, empl_ID, date, time, method_paid, amount*) |
| *items_sold* | (*trans_ID, item_ID, qty*) |
| *works_at* | (*empl_ID, branch_ID*) |

## 1.3.2 Data Warehouses

Suppose that *AllElectronics* is a successful international company with branches around the world. Each branch has its own set of databases. The president of *AllElectronics* has asked you to provide an analysis of the company's sales per item type per branch for the third quarter. This is a difficult task, particularly since the relevant data are spread out over several databases physically located at numerous sites.

If *AllElectronics* had a data warehouse, this task would be easy. A **data warehouse** is a repository of information collected from multiple sources, stored under a unified schema, and usually residing at a single site. Data warehouses are constructed via a process of data cleaning, data integration, data transformation, data loading, and periodic data refreshing. This process is discussed in Chapters 3 and 4. Figure 1.6 shows the typical framework for construction and use of a data warehouse for *AllElectronics*.

To facilitate decision making, the data in a data warehouse are organized around *major subjects* (e.g., customer, item, supplier, and activity). The data are stored to provide information from a *historical perspective*, such as in the past 6 to 12 months, and are typically *summarized*. For example, rather than storing the details of each sales transaction, the data warehouse may store a summary of the transactions per item type for each store or, summarized to a higher level, for each sales region.

A data warehouse is usually modeled by a multidimensional data structure, called a **data cube**, in which each **dimension** corresponds to an attribute or a set of attributes in the schema, and each **cell** stores the value of some aggregate measure such as *count*

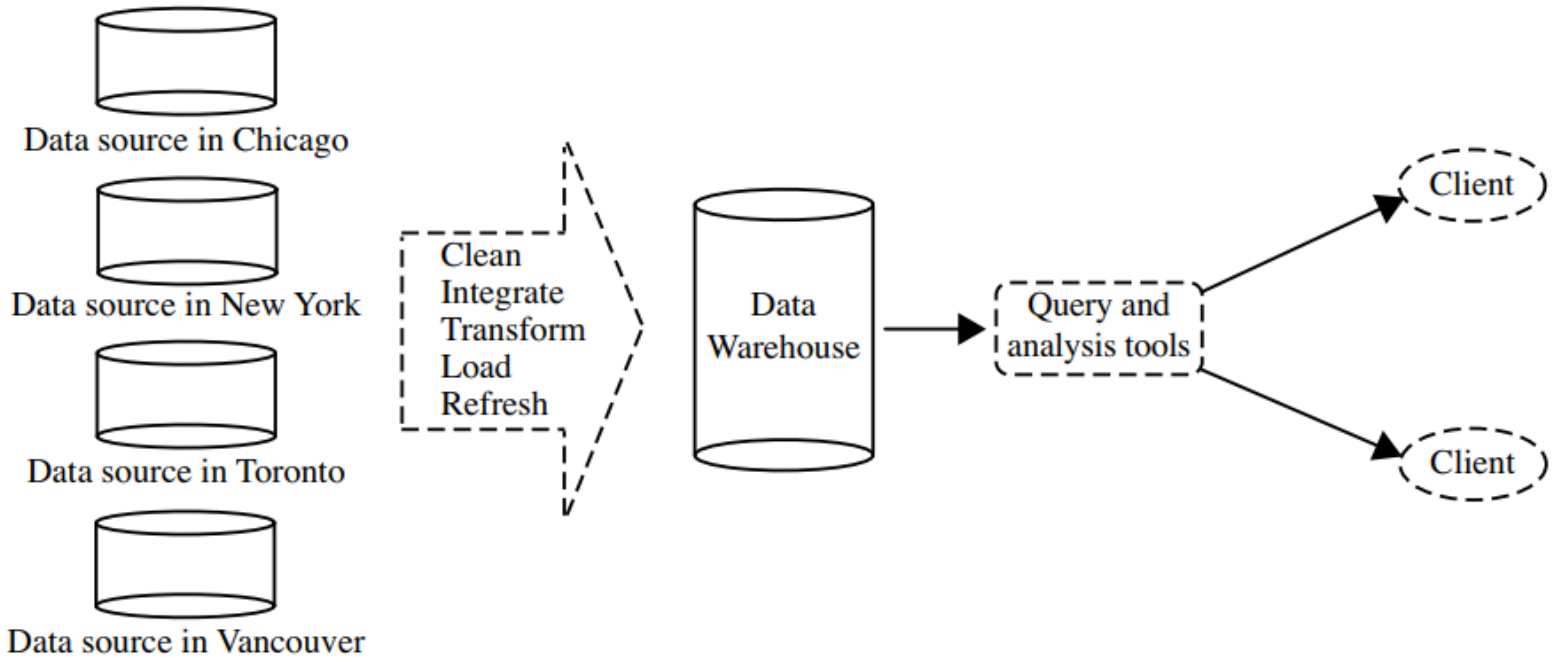



**Figure 1.6** Typical framework of a data warehouse for *AllElectronics*.

or *sum*(*sales_amount*). A data cube provides a multidimensional view of data and allows the precomputation and fast access of summarized data.

**Example 1.3** **A data cube for *AllElectronics*.** A data cube for summarized sales data of *AllElectronics* is presented in Figure 1.7(a). The cube has three dimensions: *address* (with city values *Chicago, New York, Toronto, Vancouver*), *time* (with quarter values *Q1, Q2, Q3, Q4*), and *item* (with item type values *home entertainment, computer, phone, security*). The aggregate value stored in each cell of the cube is *sales_amount* (in thousands). For example, the total sales for the first quarter, *Q1*, for the items related to security systems in Vancouver is $400,000, as stored in cell ⟨*Vancouver, Q1, security*⟩. Additional cubes may be used to store aggregate sums over each dimension, corresponding to the aggregate values obtained using different SQL group-bys (e.g., the total sales amount per city and quarter, or per city and item, or per quarter and item, or per each individual dimension). ■

By providing multidimensional data views and the precomputation of summarized data, data warehouse systems can provide inherent support for OLAP. Online analytical processing operations make use of background knowledge regarding the domain of the data being studied to allow the presentation of data at *different levels of abstraction*. Such operations accommodate different user viewpoints. Examples of OLAP operations include **drill-down** and **roll-up**, which allow the user to view the data at differing degrees of summarization, as illustrated in Figure 1.7(b). For instance, we can drill down on sales data summarized by *quarter* to see data summarized by *month*. Similarly, we can roll up on sales data summarized by *city* to view data summarized by *country*.

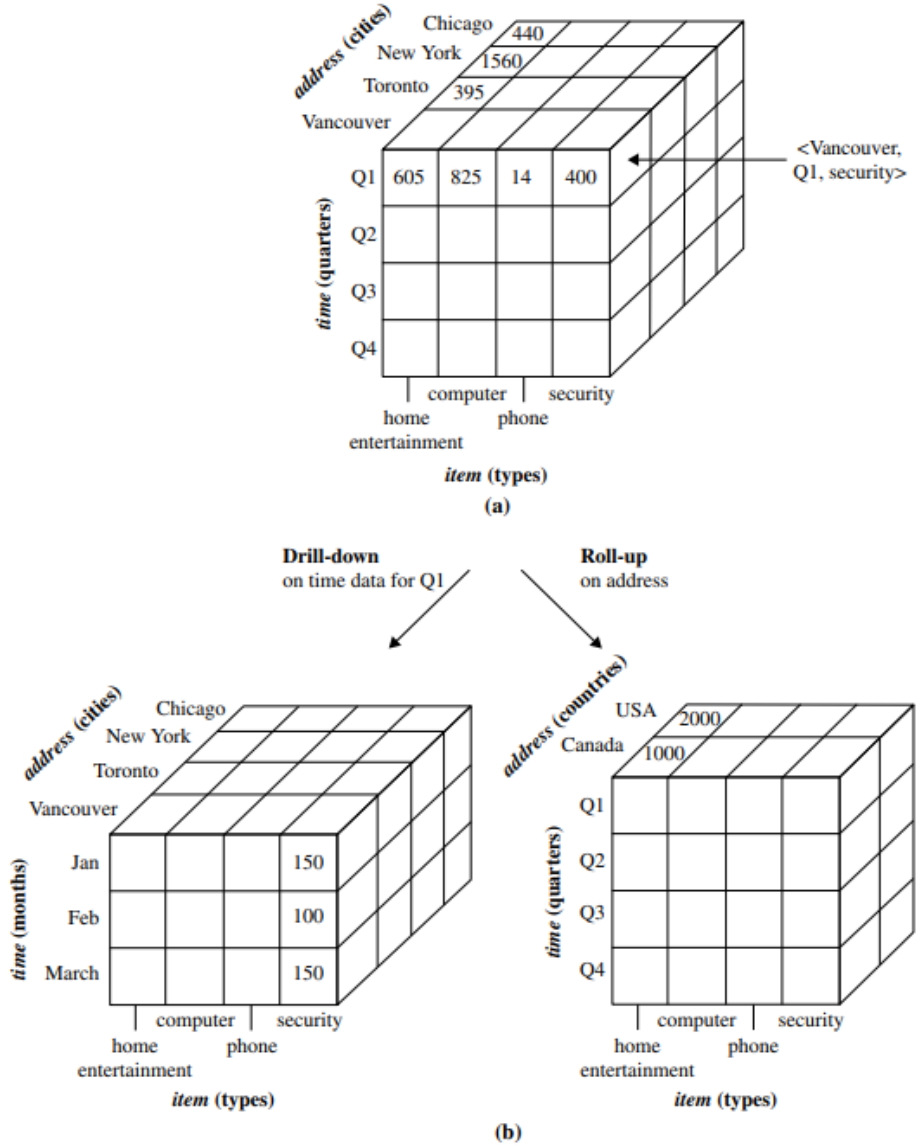



**Figure 1.7** A multidimensional data cube, commonly used for data warehousing, (a) showing summarized data for *AllElectronics* and (b) showing summarized data resulting from drill-down and roll-up operations on the cube in (a). For improved readability, only some of the cube cell values are shown.

### 1.3.3 Transactional Data

In general, each record in a **transactional database** captures a transaction, such as a customer's purchase, a flight booking, or a user's clicks on a web page. A transaction typically includes a unique transaction identity number (*trans_ID*) and a list of the **items** making up the transaction, such as the items purchased in the transaction. A transactional database may have additional tables, which contain other information related to the transactions, such as item description, information about the salesperson or the branch, and so on.

**Example 1.4** **A transactional database for *AllElectronics*.** Transactions can be stored in a table, with one record per transaction. A fragment of a transactional database for *AllElectronics* is shown in Figure 1.8. From the relational database point of view, the *sales* table in the figure is a nested relation because the attribute *list_of_item_IDs* contains a set of *items*. Because most relational database systems do not support nested relational structures, the transactional database is usually either stored in a flat file in a format similar to the table in Figure 1.8 or unfolded into a standard relation in a format similar to the *items_sold* table in Figure 1.5. ■

As an analyst of *AllElectronics*, you may ask,"*Which items sold well together?*" This kind of *market basket data analysis* would enable you to bundle groups of items together as a strategy for boosting sales. For example, given the knowledge that printers are commonly purchased together with computers, you could offer certain printers at a steep discount (or even for free) to customers buying selected computers, in the hopes of selling more computers (which are often more expensive than printers). A traditional database system is not able to perform market basket data analysis. Fortunately, data mining on transactional data can do so by mining *frequent itemsets*, that is, sets

| *trans_ID* | *list_of_item_IDs* |
|---|---|
| T100 | I1, I3, I8, I16 |
| T200 | I2, I8 |
| ... | ... |

**Figure 1.8** Fragment of a transactional database for sales at *AllElectronics*.

of items that are frequently sold together

## 1.3.4 Other Kinds of Data

Besides relational database data, data warehouse data, and transaction data, there are many other kinds of data that have versatile forms and structures and rather different semantic meanings. Such kinds of data can be seen in many applications: time-related or sequence data (e.g., historical records, stock exchange data, and time-series and biological sequence data), data streams (e.g., video surveillance and sensor data, which are continuously transmitted), spatial data (e.g., maps), engineering design data (e.g., the design of buildings, system components, or integrated circuits), hypertext and multimedia data (including text, image, video, and audio data), graph and networked data (e.g., social and information networks), and the Web (a huge, widely distributed information repository made available by the Internet). These applications bring about new challenges, like how to handle data carrying special structures (e.g., sequences, trees, graphs, and networks) and specific semantics (such as ordering, image, audio and video contents, and connectivity), and how to mine patterns that carry rich structures and semantics.

# Missing imputation

**Why missing data is a concern?**

- Missing data can create a major risk of producing incorrect conclusions due to the absence of relevant information leading to invalid results.
- Results of any statistical analysis used can be only as good as the quality of the data.
- missing data arises for many reasons such as non-response, lost data, or skip patterns in surveys.
- Let us familiarize ourselves with the most common types of missing data.

- Complete Data

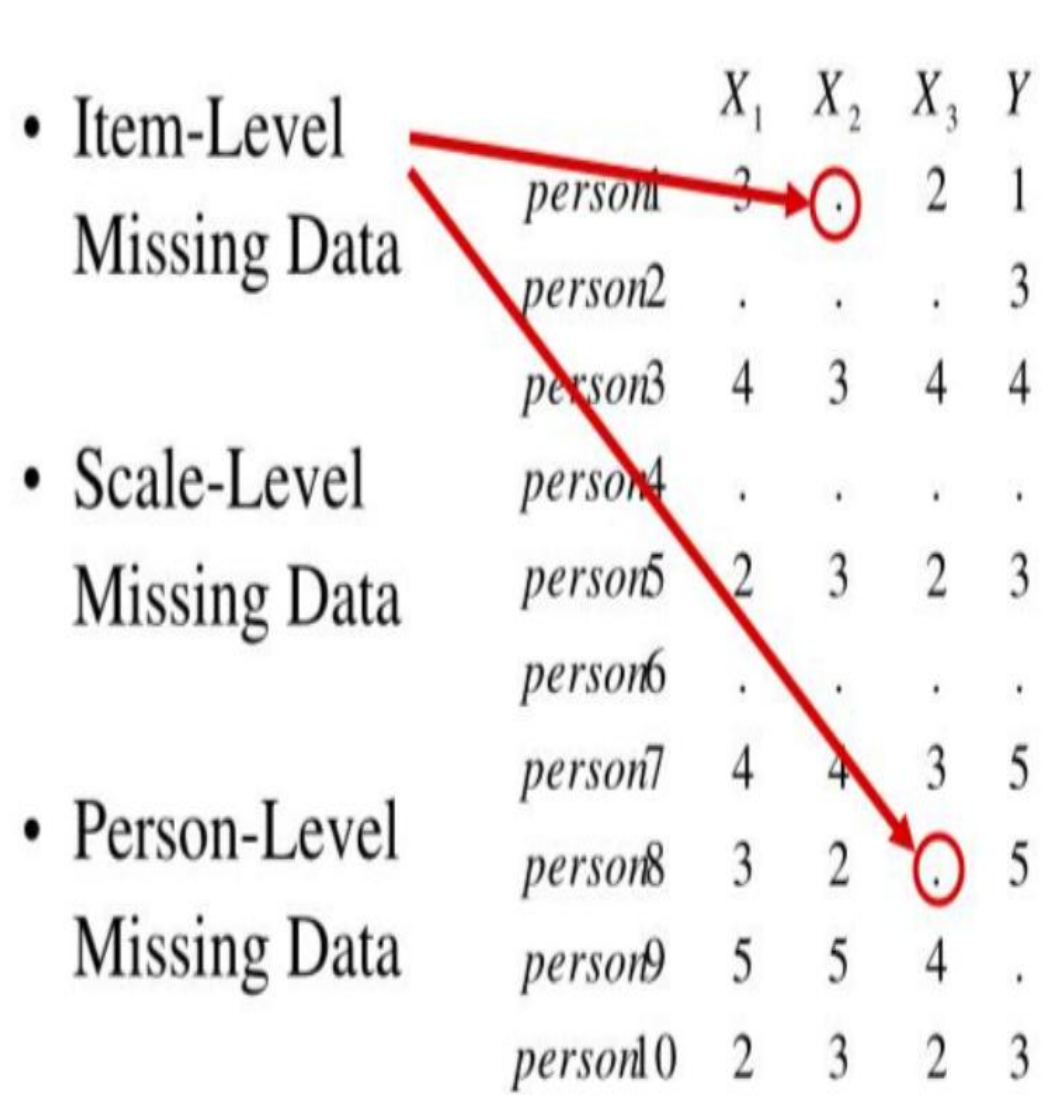
| | $X_1$ | $X_2$ | $X_3$ | $Y$ |
|---|---|---|---|---|
| person1 | 3 | 2 | 2 | 1 |
| person2 | 2 | 2 | 2 | 3 |
| person3 | 4 | 3 | 4 | 4 |
| person4 | 3 | 3 | 3 | 3 |
| person5 | 2 | 3 | 2 | 3 |
| person6 | 4 | 4 | 4 | 3 |
| person7 | 4 | 4 | 3 | 5 |
| person8 | 3 | 2 | 3 | 5 |
| person9 | 5 | 5 | 4 | 5 |
| person10 | 2 | 3 | 2 | 3 |

Incomplete Data

| | $X_1$ | $X_2$ | $X_3$ | $Y$ |
|---|---|---|---|---|
| person1 | 3 | . | 2 | 1 |
| person2 | . | . | . | 3 |
| person3 | 4 | 3 | 4 | 4 |
| person4 | . | . | . | . |
| person5 | 2 | 3 | 2 | 3 |
| person6 | . | . | . | . |
| person7 | 4 | 4 | 3 | 5 |
| person8 | 3 | 2 | . | 5 |
| person9 | 5 | 5 | 4 | . |
| person10 | 2 | 3 | 2 | 3 |

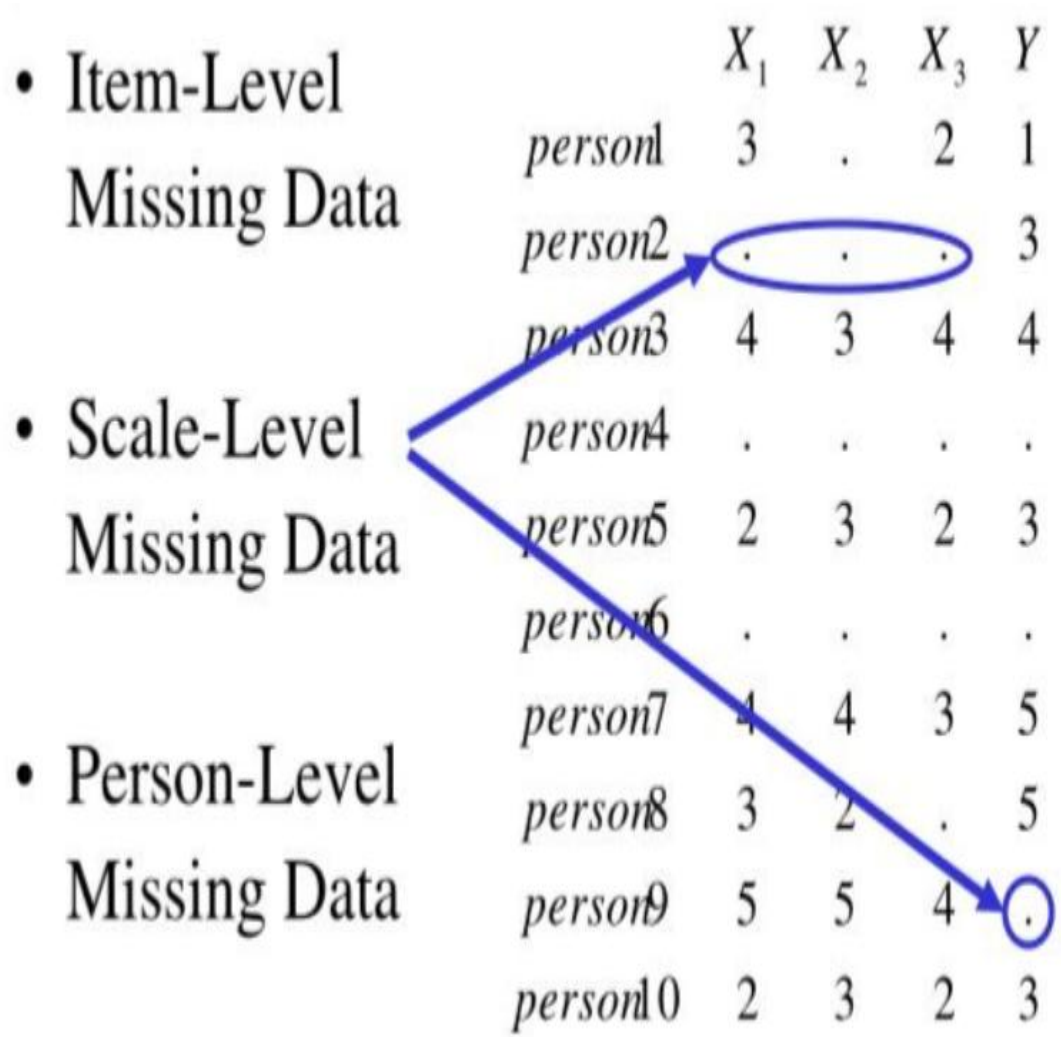
- Item-Level Missing Data
- Scale-Level Missing Data
- Person-Level Missing Data

| | $X_1$ | $X_2$ | $X_3$ | $Y$ |
|---|---|---|---|---|
| person1 | 3 | . | 2 | 1 |
| person2 | . | . | . | 3 |
| person3 | 4 | 3 | 4 | 4 |
| person4 | . | . | . | . |
| person5 | 2 | 3 | 2 | 3 |
| person6 | . | . | . | . |
| person7 | 4 | 4 | 3 | 5 |
| person8 | 3 | 2 | . | 5 |
| person9 | 5 | 5 | 4 | . |
| person10 | 2 | 3 | 2 | 3 |

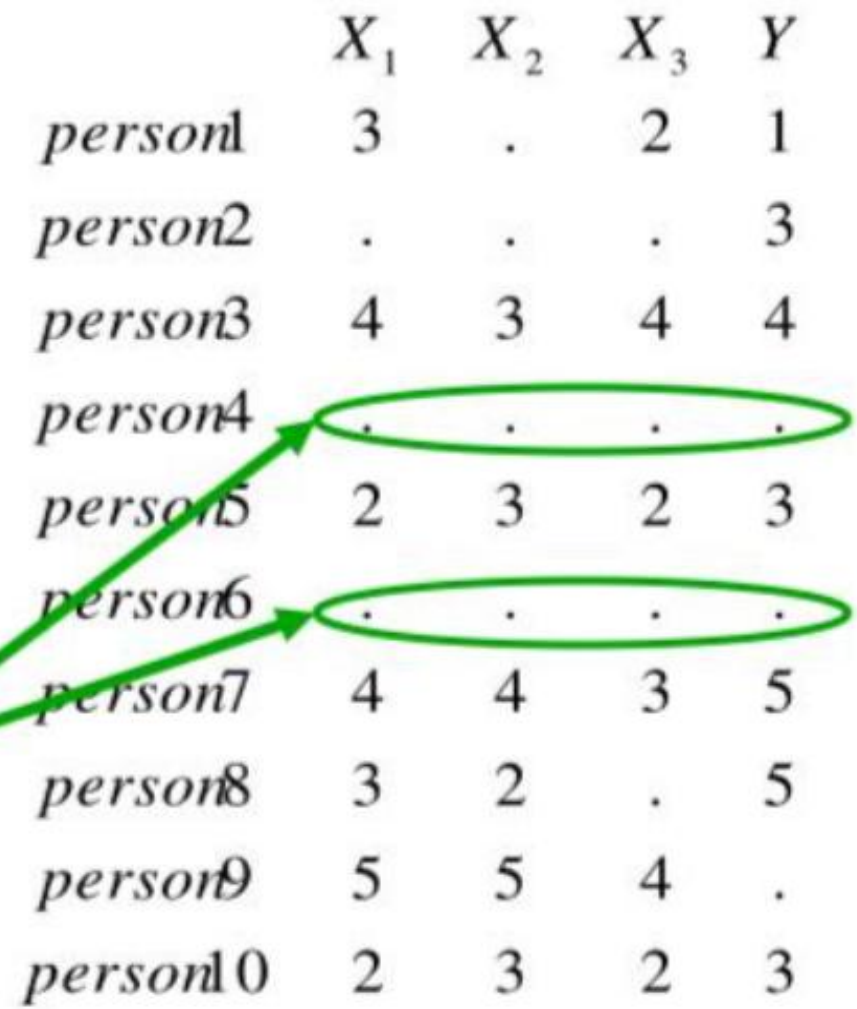



- Item-Level Missing Data
- Scale-Level Missing Data
- Person-Level Missing Data


| | $X_1$ | $X_2$ | $X_3$ | $Y$ |
|---|---|---|---|---|
| person1 | 3 | . | 2 | 1 |
| person2 | . | . | . | 3 |
| person3 | 4 | 3 | 4 | 4 |
| person4 | . | . | . | . |
| person5 | 2 | 3 | 2 | 3 |
| person6 | . | . | . | . |
| person7 | 4 | 4 | 3 | 5 |
| person8 | 3 | 2 | . | 5 |
| person9 | 5 | 5 | 4 | . |
| person10 | 2 | 3 | 2 | 3 |



- Item-Level Missing Data
- Scale-Level Missing Data
- Person-Level Missing Data



| | $X_1$ | $X_2$ | $X_3$ | $Y$ |
|---|---|---|---|---|
| person1 | 3 | . | 2 | 1 |
| person2 | . | . | . | 3 |
| person3 | 4 | 3 | 4 | 4 |
| person4 | . | . | . | . |
| person5 | 2 | 3 | 2 | 3 |
| person6 | . | . | . | . |
| person7 | 4 | 4 | 3 | 5 |
| person8 | 3 | 2 | . | 5 |
| person9 | 5 | 5 | 4 | . |
| person10 | 2 | 3 | 2 | 3 |

# Missing Data Mechanisms

- MCAR – *p*(missing) is *unrelated* to all variables, observed and unobserved

  $p(\text{missing}|\text{complete data}) = p(\text{missing})$

- MAR – *p*(missing) is related to observed variables [observed data] only

  $p(\text{missing}|\text{complete data}) = p(\text{missing}|\text{observed data})$

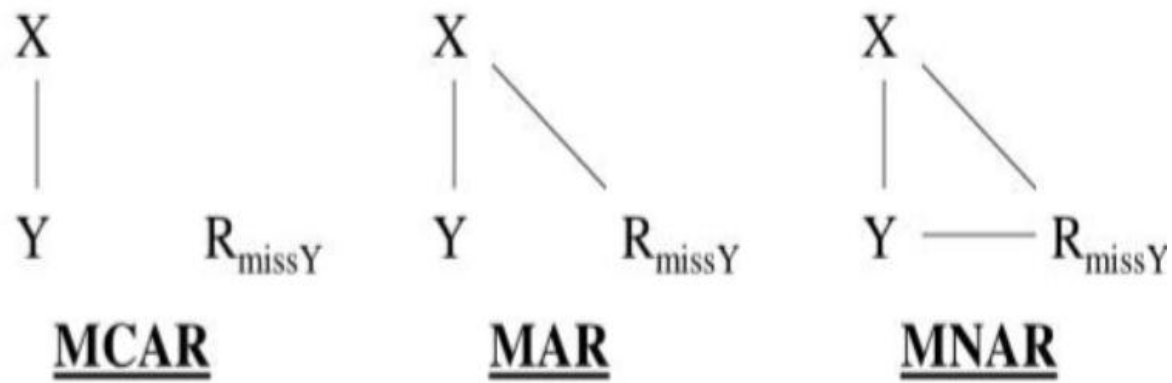
- MNAR – *p*(missing) is related to the unobserved/ missing variables [missing data]

  $p(\text{missing}|\text{complete data}) \neq p(\text{missing}|\text{observed data})$

(see Schafer & Graham, 2002)

| | $X_{complete}$ | $Y_{complete}$ | $R_{miss\_Y}$ | $Y_{incomplete}$ |
|---|---|---|---|---|
| person1 | 3 | 1 | 1 | 1 |
| person2 | 2 | 3 | 1 | 3 |
| person3 | 4 | 4 | 1 | 4 |
| person4 | 3 | 3 | 0 | . |
| person5 | 2 | 3 | 1 | 3 |
| person6 | 4 | 3 | 0 | . |
| person7 | 4 | 5 | 1 | 5 |
| person8 | 3 | 5 | 1 | 5 |
| person9 | 5 | 5 | 0 | . |
| person10 | 2 | 3 | 1 | 3 |

- MCAR – $R_{miss\_Y}$ is not related to X or Y
- MAR – $R_{miss\_Y}$ is related to X, but is not related to Y after controlling for X
- MNAR – $R_{miss\_Y}$ is related to Y



- Some missing data techniques (e.g., listwise deletion) assume missing data are MCAR
- Some missing data techniques (e.g., maximum likelihood, multiple imputation) assume MAR
- It is impossible to test whether missing data are MAR vs. MNAR, because we would need to compare observed values of Y against unobserved values of Y, and *unobserved* values of Y are *unknown*



Practically speaking …

- In the real world, MCAR almost never happens
  - One exception: "planned missingness" (Graham et al., 2006)
- Most missingness falls on a continuum between MAR and MNAR

# MCAR

- Missing completely at random (MCAR). When data are MCAR, the fact that the data are missing is independent of the observed and unobserved data.[15] In other words, no systematic differences exist between participants with missing data and those with complete data.
- For example, some participants may have missing laboratory values because a batch of lab samples was processed improperly. In these instances, the missing data reduce the analyzable population of the study and consequently, the statistical power, but do not introduce bias: when data are MCAR, the data which remain can be considered a simple random sample of the full data set of interest. MCAR is generally regarded as a strong and often unrealistic assumption.

# MAR

- Missing at random (MAR).

When data are MAR, the fact that the data are missing is systematically related to the observed but not the unobserved data.[15]

For example, a registry examining depression may encounter data that are MAR if male participants are less likely to complete a survey about depression severity than female participants. That is, if the probability of completion of the survey is related to their sex (which is fully observed) but not the severity of their depression, then the data may be regarded as MAR. Complete case analyses, which are based on only observations for which all relevant data are present and no fields are missing, of a data set containing MAR data may or may not result in bias. If the complete case analysis is biased, however, proper accounting for the known factors (in the above example, sex) can produce unbiased results in analysis.

# MNAR

- Missing not at random (MNAR).

When data are MNAR, the fact that the data are missing is systematically related to the unobserved data, that is, the missingness is related to events or factors not measured by the researcher. To extend the previous example, the depression registry may encounter data that are MNAR if participants with severe depression are more likely to refuse to complete the survey about depression severity. As with MAR data, a complete case analysis of a data set containing MNAR data may or may not result in bias; if the complete case analysis is biased, however, the fact that the sources of missing data are themselves unmeasured means that (in general) this issue cannot be addressed in analysis and the estimate of effect will likely be biased.

# Unit 3

# Contents

**Unit – III: Regression**

Concepts, Blue property assumptions, Linear regression, Logistic regression, Least Square Estimation, Variable Rationalization, Model Building, etc.

Logistic Regression: Binary, Multinomial regression, Model Theory, Model fit Statistics, Maximum Likelihood Estimation(MLE), Model Construction, Analytics applications to various Business Domains, Finance, marketing, credit card companies

# Regression Analysis

- Regression analysis is a statistical method to model the relationship between dependent (target) and independent (predictor) variables with one or more independent variables.
- More specifically, Regression analysis helps us to understand how the value of the dependent variable changes corresponding to an independent variable when other independent variables are held fixed.
- It predicts continuous/real values such as **temperature, age, salary, price,** etc.

**Example:** Suppose there is a marketing company A, which does various advertisements every year and gets sales on that. The below list shows the advertisement made by the company in the last 5 years and the corresponding sales:

| Advertisement | Sales |
|---|---|
| $90 | $1000 |
| $120 | $1300 |
| $150 | $1800 |
| $100 | $1200 |
| $130 | $1380 |
| $200 | ?? |

Now, the company wants to do the advertisement of $200 in the year 2023 **and wants to know the prediction about the sales for this year**. So to solve such type of prediction problems in machine learning, we need regression analysis.

- Regression is a supervised learning technique which helps in finding the correlation between variables and enables us to predict the continuous output variable based on the one or more predictor variables. **It is mainly used for prediction, forecasting, time series modeling, and determining the causal-effect relationship between variables.**
- In Regression, we plot a graph between the variables which best fits the given datapoints, using this plot, the machine learning model can make predictions about the data. In simple words, **"Regression shows a line or curve that passes through all the datapoints on target-predictor graph in such a way that the vertical distance between the datapoints and the regression line is minimum."** The distance between datapoints and line tells whether a model has captured a strong relationship or not.

Some examples of regression can be as:

- Prediction of rain using temperature and other factors
- Determining Market trends
- Prediction of road accidents due to rash driving.

# Terminologies

- **Dependent Variable:** The main factor in Regression analysis that we want to predict or understand is called the dependent variable. It is also called the **target variable**.
- **Independent Variable:** The factors which affect the dependent variables or which are used to predict the values of the dependent variables are called independent variables, also called **predictor**.
- **Outliers:** Outlier is an observation that contains either a very low value or very high value in comparison to other observed values. An outlier may hamper the result, so it should be avoided.
- **Multicollinearity:** If the independent variables are highly correlated with each other than other variables, then such condition is called Multicollinearity. It should not be present in the dataset, because it creates problems while ranking the most affecting variable.
- **Underfitting and Overfitting:** If our algorithm works well with the training dataset but not well with the test dataset, then such a problem is called **Overfitting**. And if our algorithm does not perform well even with a training dataset, then such a problem is called **underfitting**.

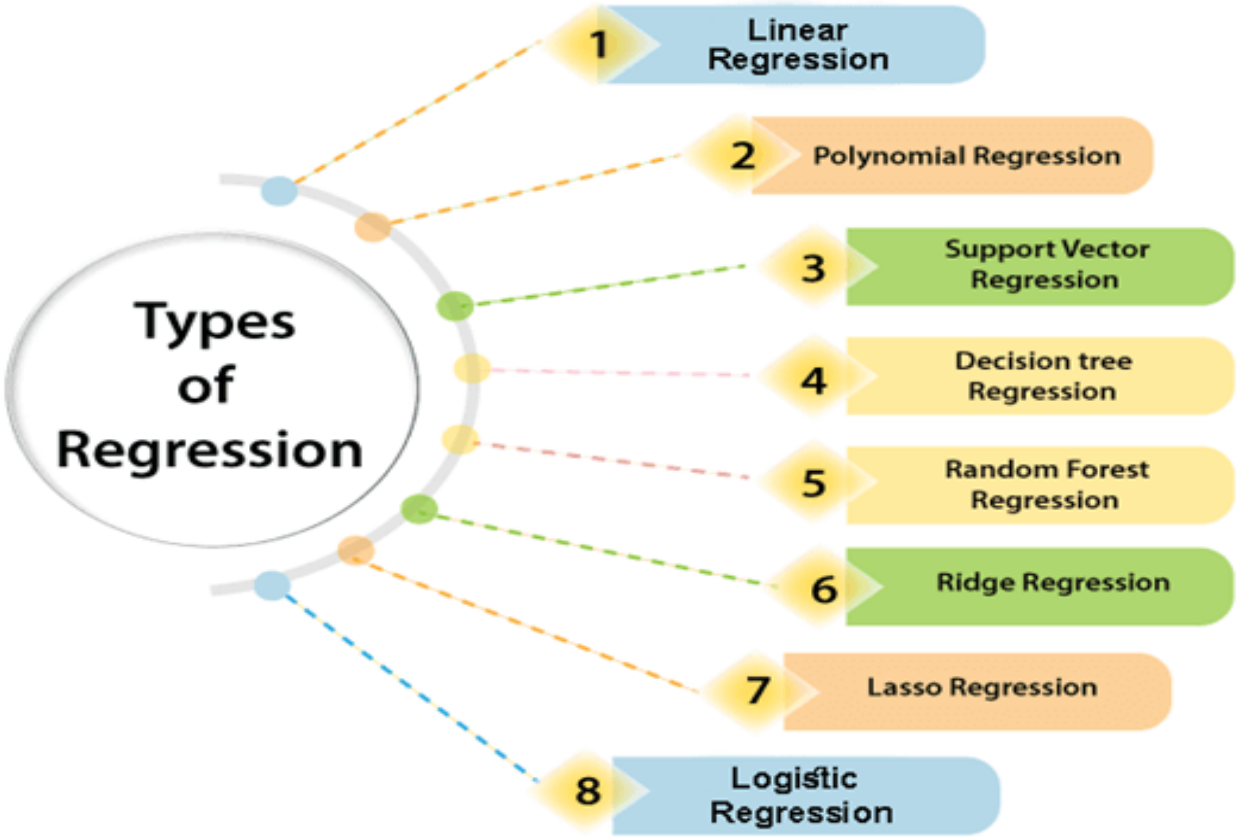
# Why do we use regression analysis?

- Regression analysis helps in the prediction of a continuous variable. There are various scenarios in the real world where we need some future predictions such as weather conditions, sales prediction, marketing trends, etc., for such a case we need some technology that can make predictions more accurately.
- So for such a case we need Regression analysis which is a statistical method used in machine learning and data science.

Below are some other reasons for using Regression analysis:

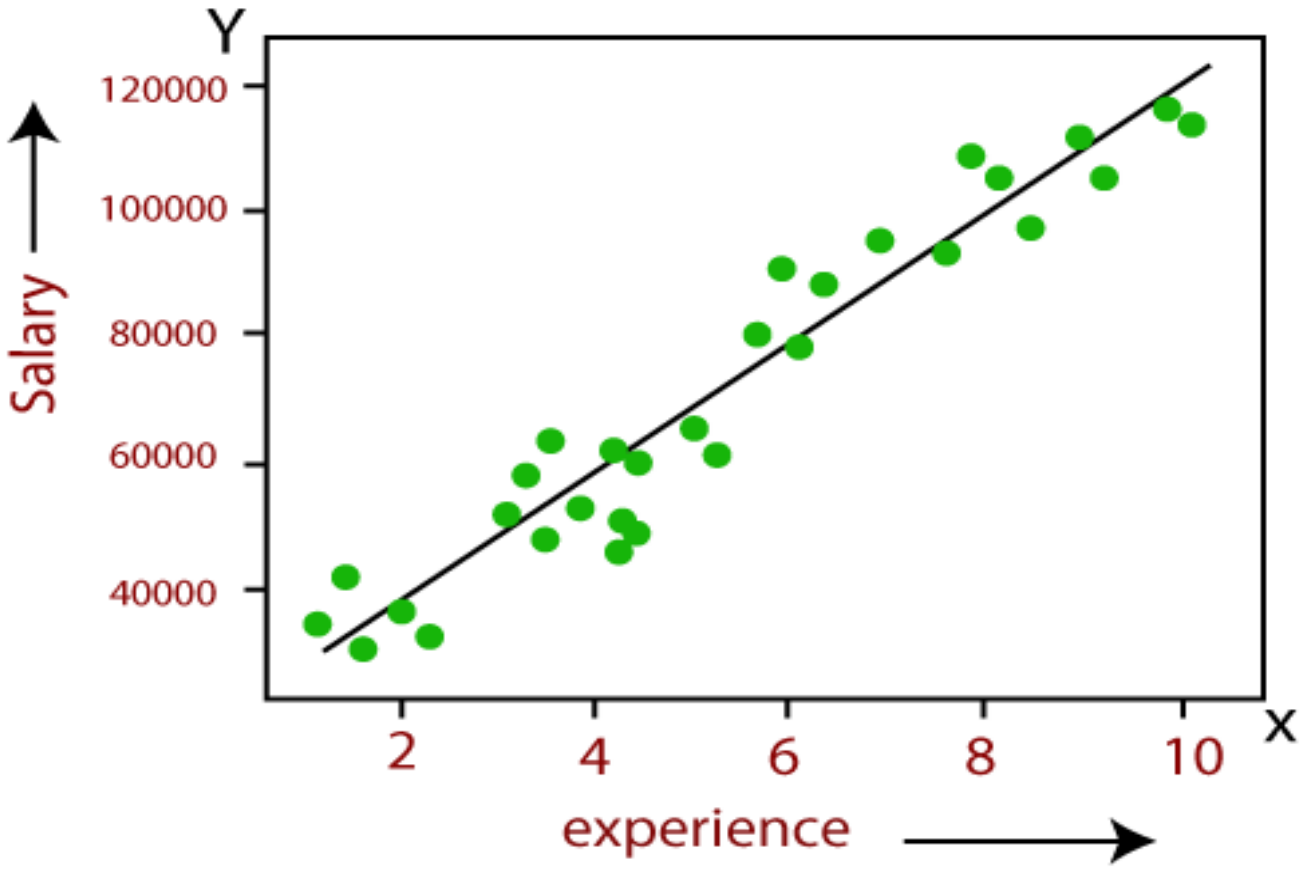
- Regression estimates the relationship between the target and the independent variable.
- It is used to find the trends in data.
- It helps to predict real/continuous values.
- By performing the regression, we can confidently determine the **most important factor, the least important factor, and how each factor is affecting the other factors**.

- There are various types of regressions that are used in data science and machine learning. Each type has its own importance in different scenarios, but at the core, all the regression methods analyze the effect of the independent variable on the dependent variables



# Linear Regression

- Linear regression is a statistical regression method that is used for predictive analysis.
- It is one of the very simple and easy algorithms that works on regression and shows the relationship between the continuous variables.
- It is used for solving the regression problem in machine learning.
- Linear regression shows the linear relationship between the independent variable (X-axis) and the dependent variable (Y-axis), hence called linear regression.
- If there is only one input variable (x), then such linear regression is called **simple linear regression**. If there is more than one input variable, then such linear regression is called **multiple linear regression**.
- The relationship between variables in the linear regression model can be explained using the below image. Here we are predicting the salary of an employee based on **the year of experience**.


Y
120000
100000
80000
60000
40000
Salary
2
4
6
8
10
X
experience

Below is the mathematical equation for Linear regression:

$$Y = aX + b$$

**Here,**

**Y**= Dependent variables (target variables), X= Independent variables (predictor variables), a and b are the linear coefficients

Some popular applications of linear regression are:

**Analyzing trends and sales estimates**

**Salary forecasting**

**Real estate prediction**

**Arriving at ETAs in traffic.**

# Logistic Regression

- Logistic regression is another supervised learning algorithm which is used to solve the classification problems. In **classification problems**, we have dependent variables in a binary or discrete format such as 0 or 1.
- Logistic regression algorithm works with categorical variables such as 0 or 1, Yes or No, True or False, Spam or not spam, etc.
- It is a predictive analysis algorithm that works on the concept of probability.
- Logistic regression is a type of regression, but it is different from the linear regression algorithm in terms of how they are used.
- Logistic regression uses a **sigmoid function** or logistic function which is a complex cost function. This sigmoid function is used to model the data in logistic regression. The function can be represented as:

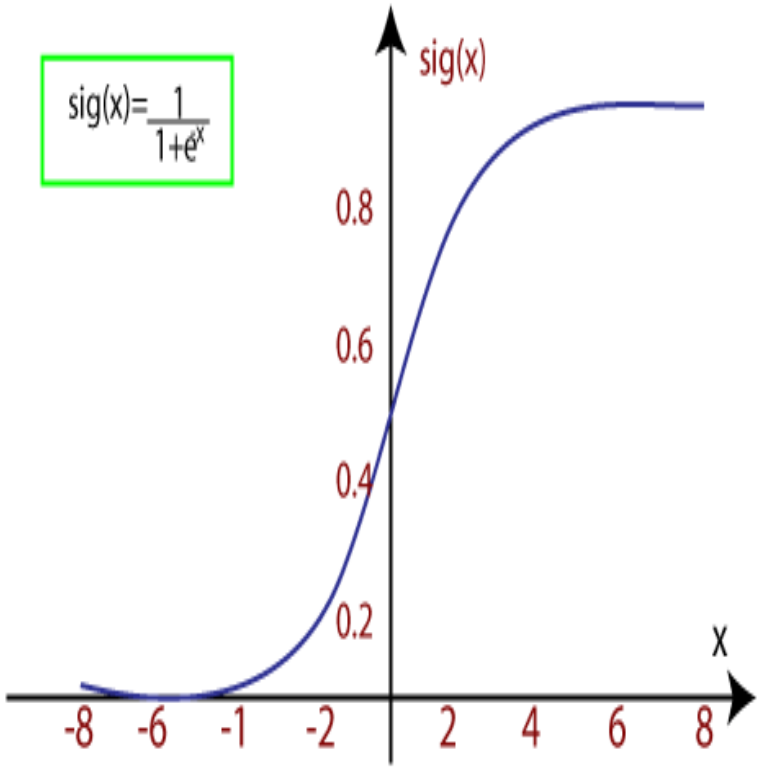
$$f(x)=\frac{1}{1+e^{-x}}$$

f(x)= Output between the 0 and 1 value.
x= input to the function
e= base of natural logarithm.

When we provide the input values (data) to the function, it gives the S-curve as follows:

It uses the concept of threshold levels, values above the threshold level are rounded up to 1, and values below the threshold level are rounded up to 0.

There are three types of logistic regression:
**Binary(0/1, pass/fail)**
**Multi(cats, dogs, lions)**
**Ordinal(low, medium, high)**



# Linear Regression

- Linear regression is a statistical method for modeling relationships between a dependent variable with a given set of independent variables.

Note: we refer to dependent variables as **responses** and independent variables as **features** for simplicity

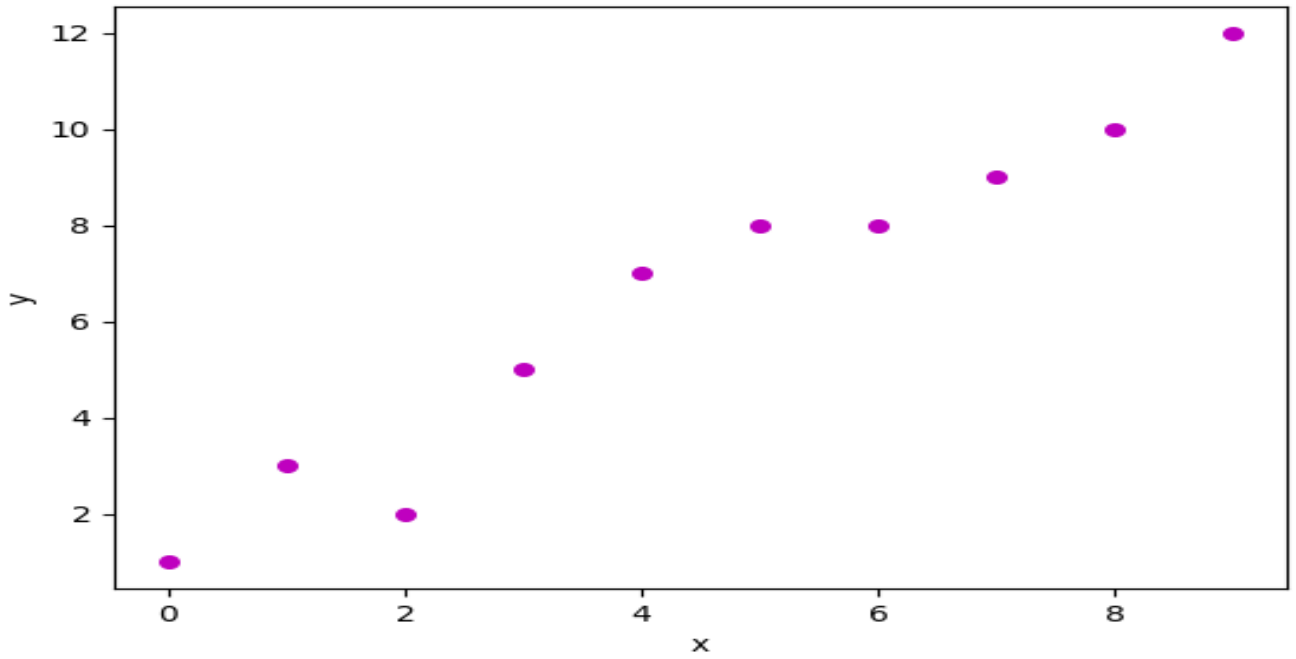
# Simple linear regression

- Simple linear regression is an approach for predicting a **response** using a **single feature**.
- It is assumed that the two variables are linearly related. Hence, we try to find a linear function that predicts the response value(y) as accurately as possible as a function of the feature or independent variable(x).
  Let us consider a dataset where we have a value of response y for every feature x:

| x | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| y | 1 | 3 | 2 | 5 | 7 | 8 | 8 | 9 | 10 | 12 |

- For generality, we define:

x as **feature vector**, i.e x = [x_1, x_2, …., x_n],
y as **response vector**, i.e y = [y_1, y_2, …., y_n]
for **n** observations (in the above example, n=10).
A scatter plot of the above dataset looks like:-



- Now, the task is to find a line that fits best in the above scatter plot so that we can predict the response for any new feature values. (i.e a value of x not present in a dataset)
  This line is called a regression line.
  The equation of regression line is represented as:

$$h(x_i) = \beta_0 + \beta_1 x_i$$

Here,

- h(x_i) represents the **predicted response value** for $i^{th}$ observation.
- b_0 and b_1 are regression coefficients and represent **y-intercept** and **slope** of regression line respectively.

$$\beta_1 = \frac{SS_{xy}}{SS_{xx}}$$

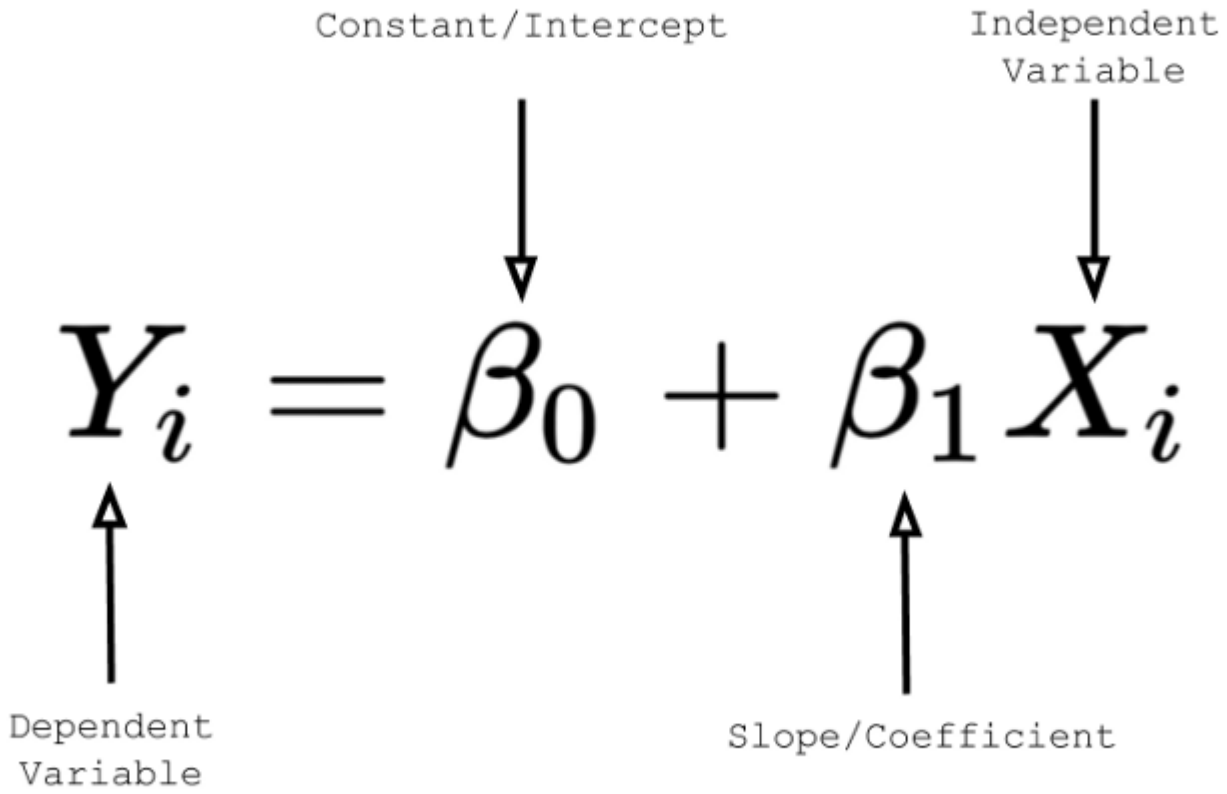
$$\beta_0 = \bar{y} - \beta_1 \bar{x}$$

where SS_xy is the sum of cross-deviations of y and x:

$$SS_{xy} = \sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) = \sum_{i=1}^{n} y_i x_i - n\bar{x}\bar{y}$$

and SS_xx is the sum of squared deviations of x:

$$SS_{xx} = \sum_{i=1}^{n}(x_i - \bar{x})^2 = \sum_{i=1}^{n} x_i^2 - n(\bar{x})^2$$




For example, let's say we have a regression equation of ***y = 2 + 0.5x***. For every 1-unit increase in the independent variable (***x***), there will be a 0.50 increase in the dependent variable (***y***).

To estimate the slope $\beta_1$ of the data we will use the following formula:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{n}(X_i - \bar{X})^2}$$

To estimate the intercept $\beta_0$, we can use the following formula:

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}$$

# Calculating How Well The Regression Line Fits

- To determine how well our regression line fits the data, we want to calculate the *correlation coefficient*, commonly referred to *just as* ***R***, and the *coefficient of determination*, otherwise known as $\mathbf{R^2}$ (R squared).

- **Coefficient of Determination (*$R^2$*)** — The percentage of variance explained by the independent variable (***x***) with values between 0 and 1. It cannot be negative because it is a square value. For example, if $\boldsymbol{R^2} = 0.81$, then this tells you that ***x*** explains 81% of the variance in ***y***. Otherwise known as the "goodness of fit".

- **Correlation Coefficient (*R*)** — The degree of relationship or correlation between two variables (***x*** and ***y*** in this case). ***R*** can range from -1 to 1 with values equal to 1 meaning a perfect positive correlation and values equal to -1 meaning a perfect negative correlation.
- Below is the formula for Pearson's correlation coefficient:

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n(\sum x^2) - (\sum x)^2]\,[n(\sum y^2) - (\sum y)^2]}}$$

Pearson Correlation Coefficient (R)

# Ex:

Formula for linear regression equation is given by:

$$y = a + bx$$

*a* and *b* are given by the following formulas:

$$a\,(intercept) = \frac{\sum y \sum x^2 - \sum x \sum xy}{(\sum x^2) - (\sum x)^2}$$

$$b\,(slope) = \frac{n\sum xy - (\sum x)(\sum y)}{n\sum x^2 - (\sum x)^2}$$

Where,

*x* and *y* are two variables on the regression line.

*b* = Slope of the line.

*a* = *y*-intercept of the line.

*x* = Values of the first data set.

*y* = Values of the second data set.

**Question:** Find linear regression equation for the following two sets of data:

| x | 2 | 4 | 6 | 8 |
|---|---|---|---|---|
| y | 3 | 7 | 5 | 10 |

| x | y | $x^2$ | xy |
|---|---|---|---|
| 2 | 3 | 4 | 6 |
| 4 | 7 | 16 | 28 |
| 6 | 5 | 36 | 30 |
| 8 | 10 | 64 | 80 |
| $\sum x$ = 20 | $\sum y$ = 25 | $\sum x^2$ = 120 | $\sum xy$ = 144 |

$$b = \frac{n\sum xy - (\sum x)(\sum y)}{n\sum x^2 - (\sum x)^2}$$

$$b = \frac{4\times 144 - 20\times 25}{4\times 120 - 400}$$

b = 0.95

$$a = \frac{\sum y \sum x^2 - \sum x \sum xy}{n(\sum x^2) - (\sum x)^2}$$

$$a = \frac{25\times 120 - 20\times 144}{4(120) - 400}$$

a = 1.5

Linear regression is given by:

y = a + bx

y = 1.5 + 0.95 x

# Another way

If you have a dataset with one independent variable, you can find the line of best fit by calculating **B**

$$B = \frac{\sum_{i=1}^{n}(x_i Y_i - \bar{Y} x_i)}{\sum_{i=1}^{n}(x_i^2 - \bar{x} x_i)}$$

Then substituting **B** into **a**

$$a = \bar{Y} - B\bar{x}$$

and finally, substituting **B** and **a** into the line of best fit!

$$\hat{Y}_i = a + Bx_i$$

If you have a dataset with one independent variable, you can find the line of best fit by calculating **B**

$$B = \frac{\sum_{i=1}^{n}(x_i Y_i - \bar{Y} x_i)}{\sum_{i=1}^{n}(x_i^2 - \bar{x} x_i)}$$

Then substituting **B** into **a**

$$a = \bar{Y} - B\bar{x}$$

and finally, substituting **B** and **a** into the line of best fit!

$$\hat{Y}_i = a + Bx_i$$

**Simple Linear Regression**

$$y = b_0 + b_1 * x_1$$

**Multiple Linear Regression**

$$y = b_0 + b_1 * x_1 + b_2 * x_2 + \ldots + b_n * x_n$$

# Least Square Method

$$m = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$c = \bar{y} - m\bar{x}$$

Here $\bar{x}$ is the mean of all the values in the input **X** and $\bar{y}$ is the mean of all the values in the desired output **Y**. This is the Least Squares method.

Least-square method is the curve that best fits a set of observations with a minimum sum of squared residuals or errors. Let us assume that the given points of data are $(x_1, y_1)$, $(x_2, y_2)$, $(x_3, y_3)$, ..., $(x_n, y_n)$ in which all x's are independent variables, while all y's are dependent ones. This method is used to find a linear line of the form $y = mx + b$, where y and x are variables, m is the slope, and b is the y-intercept. The formula to calculate slope m and the value of b is given by:

$$m = (n\sum xy - \sum y\sum x)/n\sum x^2 - (\sum x)^2$$

$$b = (\sum y - m\sum x)/n$$

Here, n is the number of data points.

Following are the steps to calculate the least square using the above formulas.

- Step 1: Draw a table with 4 columns where the first two columns are for x and y points.
- Step 2: In the next two columns, find xy and $(x)^2$.
- Step 3: Find $\sum x$, $\sum y$, $\sum xy$, and $\sum (x)^2$.
- Step 4: Find the value of slope m using the above formula.
- Step 5: Calculate the value of b using the above formula.
- Step 6: Substitute the value of m and b in the equation $y = mx + b$

**Example:** Let's say we have data as shown below.

| x | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| y | 2 | 5 | 3 | 8 | 7 |

**Solution:** We will follow the steps to find the linear line.

| x | y | xy | $x^2$ |
|---|---|---|---|
| 1 | 2 | 2 | 1 |
| 2 | 5 | 10 | 4 |
| 3 | 3 | 9 | 9 |
| 4 | 8 | 32 | 16 |
| 5 | 7 | 35 | 25 |
| $\sum x = 15$ | $\sum y = 25$ | $\sum xy = 88$ | $\sum x^2 = 55$ |

Find the value of m by using the formula,

$m = (n\sum xy - \sum y\sum x)/n\sum x^2 - (\sum x)^2$

$m = [(5\times88) - (15\times25)]/(5\times55) - (15)^2$

$m = (440 - 375)/(275 - 225)$

$m = 65/50 = 13/10$

Find the value of b by using the formula,

$b = (\sum y - m\sum x)/n$

$b = (25 - 1.3\times15)/5$

$b = (25 - 19.5)/5$

$b = 5.5/5$

So, the required equation of least squares is $y = mx + b = 13/10x + 5.5/5$.

# Summary

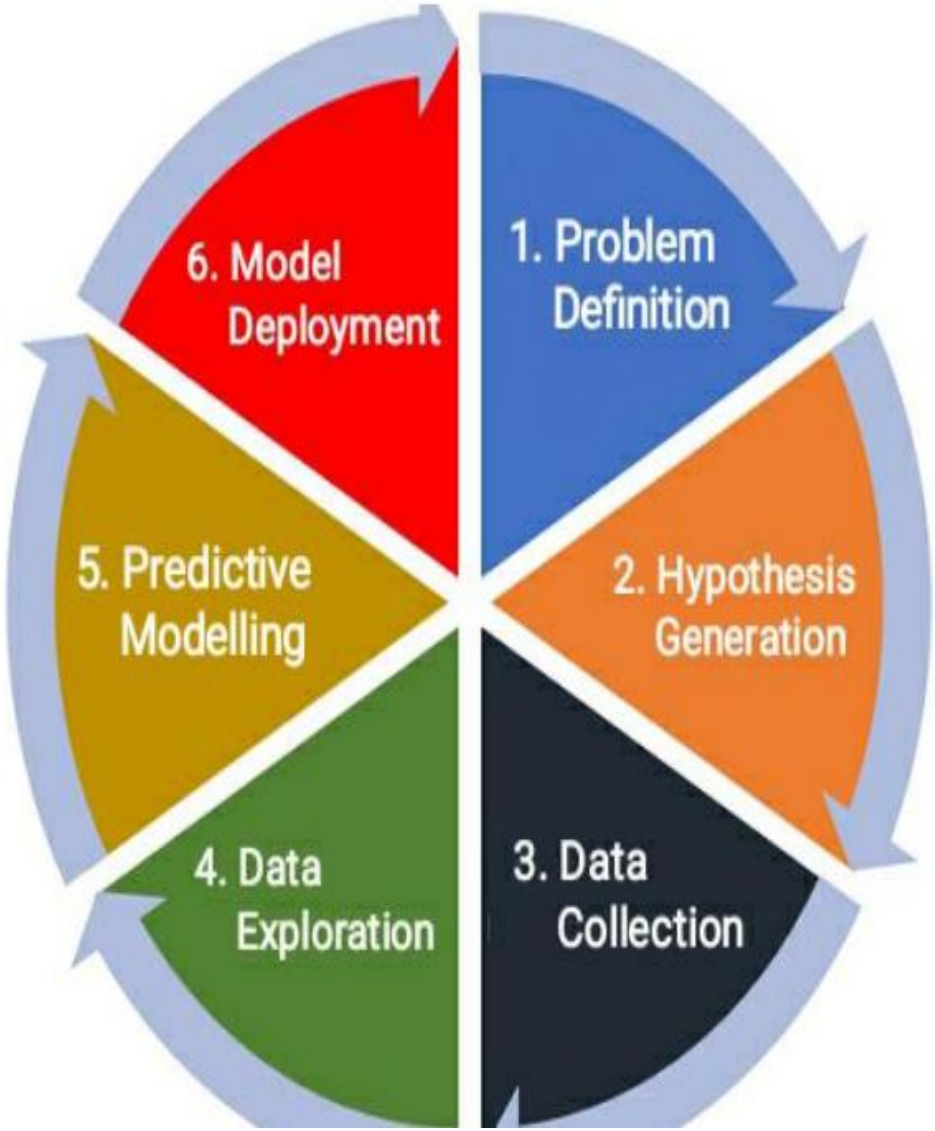
- The least-squares method is used to predict the behavior of the dependent variable with respect to the independent variable.
- The sum of the squares of errors is called variance.
- The main aim of the least-squares method is to minimize the sum of the squared errors.

# Model Building Life Cycle in Data Analytics:

- When we come across a business analytical problem, without acknowledging the stumbling blocks, we proceed towards the execution. Before realizing the misfortunes, we try to implement and predict the outcomes.

- The problem-solving steps involved in the data science model-building life cycle. Let’s understand every model-building step in-depth, The data science model-building life cycle includes some important steps to follow. The following are the steps to follow to build a Data Model

1. Problem Definition
2. Hypothesis Generation
3. Data Collection
4. Data Exploration/Transformation
5. Predictive Modelling
6. Model Deployment



# 1. Problem Definition

- The first step in constructing a model is to understand the industrial problem in a more comprehensive way. To identify the purpose of the problem and the prediction target, we must define the project objectives appropriately.
- Therefore, to proceed with an analytical approach, we have to recognize the obstacles first. Remember, excellent results always depend on a better understanding of the problem.

# 2. Hypothesis Generation

- Hypothesis generation is the guessing approach through which we derive some essential data parameters that have a significant correlation with the prediction target.
- Your hypothesis research must be in-depth, looking for every perceptive of all stakeholders into account. We search for every suitable factor that can influence the outcome.
- Hypothesis generation focuses on what you can create rather than what is available in the dataset

# 3. Data Collection

- Data collection is gathering data from relevant sources regarding the analytical problem, then we extract meaningful insights from the data for prediction.

The data gathered must have:

- Proficiency in answering hypothesis questions.
- Capacity to elaborate on every data parameter.
- Effectiveness to justify your research.
- Competency to predict outcomes accurately.

# 4. Data Exploration/Transformation

- The data you collected may be in unfamiliar shapes and sizes. It may contain unnecessary features, null values, unanticipated small values, or immense values. So, before applying any algorithmic model to data, we have to explore it first.
- By inspecting the data, we get to understand the explicit and hidden trends in data. We find the relation between data features and the target variable.
- Usually, a data scientist invests his 60–70% of project time dealing with data exploration only.
- There are several sub-steps involved in data exploration:

## o Feature Identification:

- You need to analyze which data features are available and which ones are not.
- Identify independent and target variables.
- Identify data types and categories of these variables.

## o Univariate Analysis:

- We inspect each variable one by one. This kind of analysis depends on the variable type whether it is categorical and continuous.
- Continuous variable: We mainly look for statistical trends like mean, median, standard deviation, skewness, and many more in the dataset.
- Categorical variable: We use a frequency table to understand the spread of data for each category. We can measure the counts and frequency of occurrence of values

## o Multi-variate Analysis:

- The bi-variate analysis helps to discover the relation between two or more variables.
- We can find the correlation in case of continuous variables and the case of categorical, we look for association and dissociation between them.

## o Filling Null Values:

- Usually, the dataset contains null values which lead to lower the potential of the model.
- With a continuous variable, we fill these null values using the mean or mode of that specific column.
- For the null values present in the categorical column, we replace them with the most frequently occurring categorical value.
- Remember, don’t delete those rows because you may lose the information.

# 5. Predictive Modeling

- Predictive modeling is a mathematical approach to create a statistical model to forecast future behavior based on input test data. Steps involved in predictive modeling:

- Algorithm Selection:

o When we have a structured dataset, and we want to estimate the continuous or categorical outcome then we use supervised machine learning methodologies like regression and classification techniques. When we have unstructured data and want to predict the clusters of items to which a particular input test sample belongs, we use unsupervised algorithms. An actual data scientist applies multiple algorithms to get a more accurate model.

## Train Model:

After assigning the algorithm and getting the data handy, we train our model using the input data applying the preferred algorithm. It is an action to determine the correspondence between independent variables, and the prediction targets.

## Model Prediction

- We make predictions by giving the input test data to the trained model. We measure the accuracy by using a cross-validation strategy or ROC curve which performs well to derive model output for test data.

# 6. Model Deployment

- There is nothing better than deploying the model in a real-time environment. It helps us to gain analytical insights into the decision-making procedure. You constantly need to update the model with additional features for customer satisfaction.
- To predict business decisions, plan market strategies, and create personalized customer interests, we integrate the machine learning model into the existing production domain.
- When you go through the Amazon website and notice the product recommendations completely based on your curiosities. You can experience the increase in the involvement of the customers utilizing these services. That’s how a deployed model changes the mindset of the customer and convince him to purchase the product.

# BLUE Property Assumptions

In data analytics, the term "BLUE" stands for "Best Linear Unbiased Estimator." BLUE properties refer to the desirable characteristics of an estimator that make it optimal for estimating parameters in a linear regression model. These properties are fundamental for statistical inference and regression analysis. Here's a breakdown of what each component means:

- Best: The estimator is considered "best" because it has the smallest variance among all unbiased estimators. In other words, it minimizes the spread or uncertainty of the estimated parameter values.
- Linear: The estimator is a linear function of the observed data. This means that it can be expressed as a linear combination of the independent variables.
- Unbiased: The estimator is unbiased if, on average, it produces estimates that are equal to the true parameter values. In other words, there is no systematic overestimation or underestimation.
- Estimator: An estimator is a statistical method or procedure used to estimate a population parameter (e.g., the slope or intercept in linear regression) based on sample data.

# Variable Rationalization

- "Variable rationalization" in data analytics typically refers to the process of evaluating and justifying the inclusion or exclusion of variables in a statistical model. It involves assessing the relevance, importance, and impact of different variables on the outcome of the analysis. Here's a more detailed explanation of variable rationalization:

1. **Variable Selection**: In many datasets, there can be numerous variables available for analysis. Variable rationalization involves selecting the subset of variables that are most relevant to the analysis at hand. This can be based on domain knowledge, statistical techniques such as feature selection algorithms, or a combination of both.
2. **Feature Engineering**: Sometimes, variable rationalization involves creating new variables through feature engineering. This could include transforming existing variables, combining them to create new features, or deriving additional variables that may capture important information.
3. **Multicollinearity Assessment**: Multicollinearity occurs when two or more predictor variables in a regression model are highly correlated. Variable rationalization includes assessing multicollinearity to identify and potentially remove redundant variables to avoid issues with interpretation and estimation in regression analysis.
4. **Validation and Iteration**: Variable rationalization is often an iterative process that involves validating the chosen variables and model through techniques such as cross-validation or out-of-sample testing. This helps ensure that the selected variables are robust and generalize well to unseen data.

# Variable Rationalization:

- The data set may have a large number of attributes. But some of those attributes can be irrelevant or redundant. The goal of Variable Rationalization is to improve the Data Processing optimally through attribute subset selection.
- This process is to find a minimum set of attributes such that dropping of those irrelevant attributes does not much affect the utility of data and the cost of data analysis could be reduced.
- Mining on a reduced data set also makes the discovered pattern easier to understand. As part of Data processing, we use the below methods of **Attribute subset selection**

1. Stepwise Forward Selection
2. Stepwise Backward Elimination
3. Combination of Forward Selection and Backward Elimination
4. Decision Tree Induction.

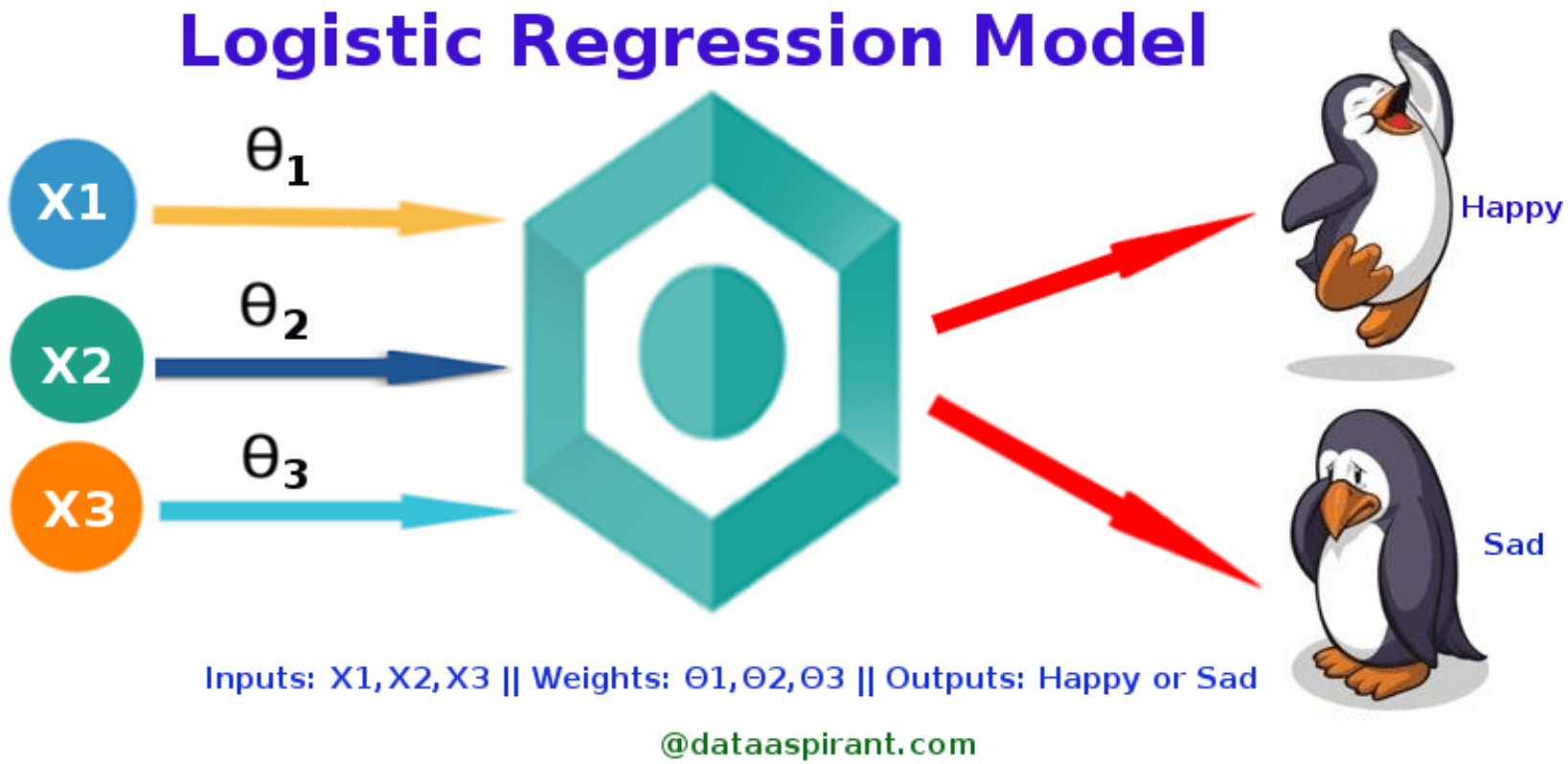
- Stepwise Forward Selection: This procedure starts with an empty set of attributes as the minimal set. The most relevant attributes are chosen (having minimum p-value) and are added to the minimal set. In each iteration, one attribute is added to a reduced set.
- Stepwise Backward Elimination: Here all the attributes are considered in the initial set of attributes. In each iteration, one attribute is eliminated from the set of attributes whose p-value is higher than the significance level.
- Combination of Forward Selection and Backward Elimination: The stepwise forward selection and backward elimination are combined to select the relevant attributes most efficiently. This is the most common technique which is generally used for attribute selection.
- Decision Tree Induction: This approach uses a decision tree for attribute selection. It constructs a flow chart-like structure having nodes denoting a test on an attribute. Each branch corresponds to the outcome of the test and leaf nodes are a class prediction. The attribute that is not part of the tree is considered irrelevant and hence discarded.

# Logistic regression

- Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, logistic regression is a predictive analysis. It is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

- Logistic Regression is another statistical analysis method borrowed from Machine Learning. It is used when our dependent variable is dichotomous or binary. It just means a variable that has only 2 outputs,
- For example, whether A person will survive this accident or not, The student will pass this exam or not. The outcome can either be yes or no (2 outputs). This regression technique is similar to linear regression and can be used to predict the Probabilities for classification problems.

# Logistic regression



# Types of Logistic Regression

- **Here are the three main types of logistic regression**:

**Binary logistic regression**

- Binary logistic regression is used to predict the probability of a binary outcome, such as yes or no, true or false, or 0 or 1. For example, it could be used to predict whether a patient has a disease or not, or whether a loan will be repaid or not**.**

**Multinomial logistic regression**

- Multinomial logistic regression is used to predict the probability of one of three or more possible outcomes, such as the type of product a customer will buy, the rating a customer will give a product, or the political party a person will vote for.

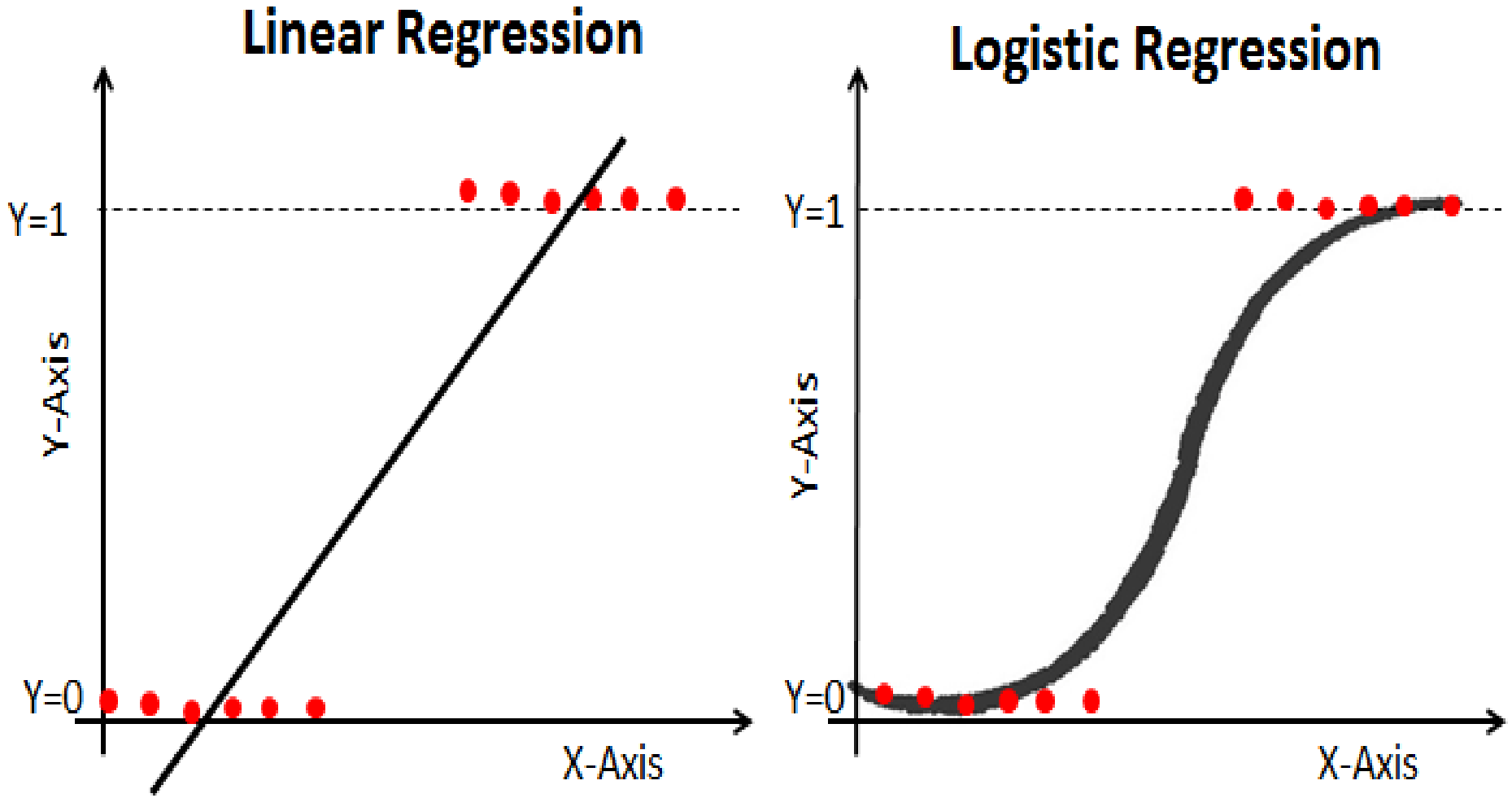
**Ordinal logistic regression**

- **i**s used to predict the probability of an outcome that falls into a predetermined order, such as the level of customer satisfaction, the severity of a disease, or the stage of cancer

# How does Logistic Regression work?

**Logistic regression works in the following steps:**

1. **Prepare the data: T**he data should be in a format where each row represents a single observation and each column represents a different variable. The target variable (the variable you want to predict) should be binary (yes/no, true/false, 0/1).
2. **Train the model:** We teach the model by showing it the training data. This involves finding the values of the model parameters that minimize the error in the training data.
3. **Evaluate the model:** The model is evaluated on the held-out test data to assess its performance on unseen data.
4. **Use the model to make predictions:** After the model has been trained and assessed, it can be used to forecast outcomes on new data.


Linear Regression
Y=1
Y=0
Y-Axis
X-Axis
Logistic Regression
Y=1
Y=0
Y-Axis
X-Axis

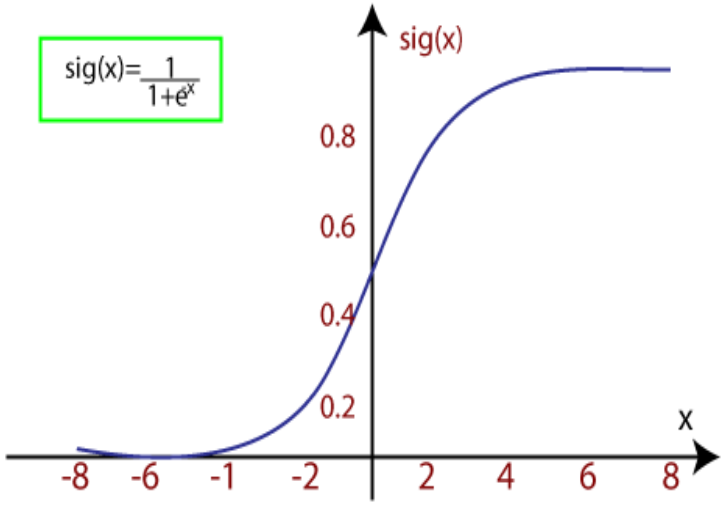
# Logistic Regression

- Logistic regression uses a **sigmoid function** or logistic function which is a complex cost function. This sigmoid function is used to model the data in logistic regression. The function can be represented as:

$$f(x)= \frac{1}{1+e^{-x}}$$

f(x)= Output between the 0 and 1 value.
x= input to the function
e= base of natural logarithm.

When we provide the input values (data) to the function, it gives the S-curve as follows:




It uses the concept of threshold levels, values above the threshold level are rounded up to 1, and values below the threshold level are rounded up to 0.

# Logistic Regression: Model Theory

- Logistic regression is a technique used when the dependent variable is categorical (or nominal). Examples: 1) Consumers make a decision to buy or not to buy, 2) a product may pass or fail quality control, 3) there are good or poor credit risks, and 4) an employee may be promoted or not.
- **Binary logistic regression** - determines the impact of multiple independent variables presented simultaneously to predict membership of one or other of the two dependent variable categories.
- Since the dependent variable is dichotomous we cannot predict a numerical value for it using logistic regression so the usual regression least squares deviations criteria for the best fit approach of minimizing error around the line of best fit is inappropriate (It's impossible to calculate deviations using binary variables!).
- Instead, logistic regression employs binomial probability theory in which there are only two values to predict: that probability (p) is 1 rather than 0, i.e. the event/person belongs to one group rather than the other.

- Logistic regression forms a best fitting equation or function using the maximum likelihood (ML) method, which maximizes the probability of classifying the observed data into the appropriate category given the regression coefficients.
- Like multiple regression, logistic regression provides a coefficient ‘b’, which measures each independent variable’s partial contribution to variations in the dependent variable.
- The goal is to correctly predict the category of outcome for individual cases using the most parsimonious model.
- To accomplish this goal, a model (i.e. an equation) is created that includes all predictor variables that are useful in predicting the response variable.

# The Purpose of Binary Logistic Regression

1. The logistic regression predicts group membership

- Since logistic regression calculates the probability of success over the probability of failure, the results of the analysis are in the form of an odds ratio.
- Logistic regression determines the impact of multiple independent variables presented simultaneously to predict membership of one or other of the two dependent variable categories.

2. The logistic regression also provides the relationships and strengths among the variables

**## Assumptions of (Binary) Logistic Regression**

- Logistic regression does not assume a linear relationship between the dependent and independent variables.
    - Logistic regression assumes linearity of independent variables and log odds of dependent variables.
- The independent variables need not be interval, nor normally distributed, nor linearly related, nor of equal variance within each group
    - Homoscedasticity is not required. The error terms (residuals) do not need to be normally distributed.
- The dependent variable in logistic regression is not measured on an interval or ratio scale.
    - The dependent variable must be dichotomous ( 2 categories) for the binary logistic regression.
- The categories (groups) as a dependent variable must be mutually exclusive and exhaustive; a case can only be in one group and every case must be a member of one of the groups.
- Larger samples are needed than for linear regression because maximum coefficients using an ML method are large sample estimates. A minimum of 50 cases per predictor is recommended (Field, 2013)
- Hosmer, Lemeshow, and Sturdivant (2013) suggest a minimum sample of 10 observations per independent variable in the model but caution that 20 observations per variable should be sought if possible.
- Leblanc and Fitzgerald (2000) suggest a minimum of 30 observations per independent variable.

# Log Transformation

The log transformation is, arguably, the most popular among the different types of transformations used to transform skewed data to approximately conform to normality.

- Log transformations and sq. root transformations moved skewed distributions closer to normality. So what we are about to do is common.

This log transformation of the p values to a log distribution enables us to create a link with the normal regression equation. The log distribution (or logistic transformation of p) is also called the logit of p or logit(p).

In logistic regression, a logistic transformation of the odds (referred to as logit) serves as the depending variable:

$$\log(odds) = \text{logit}(P) = \ln\left(\frac{P}{1-P}\right)$$

If we take the above dependent variable and add a regression equation for the independent variables, we get a logistic regression:

$$logit(p) = a + b_1x_1 + b_2x_2 + b_3x_3 + \ldots$$

# Equation

$$P = \frac{\exp(a + b_1x_1 + b_2x_2 + b_3x_3 + \ldots)}{1 + \exp(a + b_1x_1 + b_2x_2 + b_3x_3 + \ldots)}$$

In the equation above: *P* can be calculated with the following formula

where:

*P* = the probability that a case is in a particular category,

*exp* = the exponential function (approx. 2.72),

*a* = the constant (or intercept) of the equation and,

*b* = the coefficient (or slope) of the predictor variables.

# Logistic Function (Sigmoid Function):

- The sigmoid function is a mathematical function that maps the predicted values to probabilities.
- The sigmoid function maps any real value into another value within a range of 0 and 1, forming forma S-Form curve.
- The value of the logistic regression must be between 0 and 1, which cannot go beyond this limit, so it forms a curve like the "S" form

The below image is showing the logistic function:

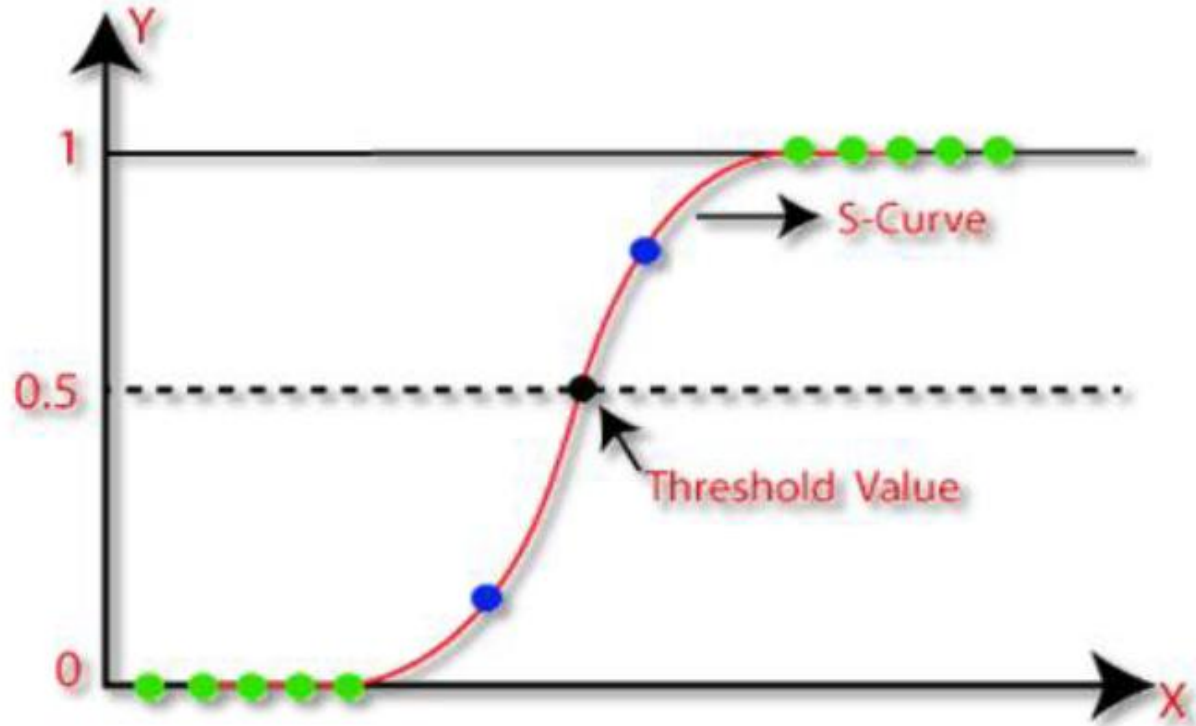



Fig: Sigmoid Function Graph

# Hypothesis Test

- In logistic regression, hypotheses are of interest:
- **the null hypothesis**, which is when all the coefficients in the regression equation take the value zero, and

- **the alternate hypothesis** that the model currently under consideration is accurate and differs significantly from the null of zero, i.e. gives significantly better than the chance or random prediction level of the null hypothesis.

- The null hypothesis in logistic regression states that there is no relationship between the independent variables and the outcome variable. In other words, it suggests that the coefficients of the independent variables in the regression equation are all equal to zero. This implies that the independent variables do not have any effect on predicting the outcome variable, and any observed relationship is due to random chance.
- On the other hand, the alternate hypothesis in logistic regression asserts that the model being evaluated is accurate and significantly different from the null hypothesis. This means that the independent variables included in the model are meaningful predictors of the outcome variable, and the model provides a better fit to the data than would be expected by chance alone. In essence, the alternate hypothesis suggests that the regression model has predictive power beyond random chance and is capable of making meaningful predictions about the outcome variable based on the values of the independent variables.

# Model Statistics

**Likelihood Ratio Test**: This test compares how well our full model with all predictors fits the data compared to a simpler model with fewer predictors. It helps us see if adding more predictors significantly improves our model's fit.

Example: Let's say we have a logistic regression model predicting whether customers will purchase a product based on their age, income, and education level. We compare this full model to a simpler model that only includes age and income. If the likelihood ratio test shows a significant improvement in fit for the full model over the reduced model, it suggests that including education level improves our ability to predict purchases.

# Model Statistics

**Deviance**: Deviance tells us how far our model's predictions are from a perfect fit. A lower deviance means our model fits the data better.

Example: Suppose we have a logistic regression model predicting whether patients will develop a certain disease based on their health indicators. Lower deviance indicates that our model's predicted probabilities are closer to the actual outcomes. For instance, a deviance value of 1000 suggests better fit compared to a deviance value of 1500.

# Model Statistics

**AIC (Akaike Information Criterion)** and **BIC (Bayesian Information Criterion)**: These are measures that balance how well our model fits the data with how complex it is. Smaller AIC and BIC values suggest better models.

Example: Continuing with the disease prediction example, if we have two competing logistic regression models, one with five predictors and another with ten predictors, we can compare their AIC and BIC values. If the model with five predictors has lower AIC and BIC values compared to the ten-predictor model, it suggests that the simpler model is preferable as it balances goodness of fit with model complexity.

# Model Statistics

**Pseudo R-squared**: Instead of using the traditional R-squared from linear regression, logistic regression uses pseudo R-squared values like McFadden's R-squared or Nagelkerke's R-squared. These values help us understand how much of the variability in the data our model explains.

Example: In a study investigating factors influencing student performance, a logistic regression model predicts whether students will pass an exam based on variables like study hours, attendance, and prior GPA. McFadden's R-squared might indicate that our model explains 30% of the variation in exam pass rates. This helps us gauge how well our model captures the relationship between predictors and the outcome compared to a null model.

# Model Construction:

1. **Data Collection and Preparation**: Gather and preprocess your data, ensuring that independent and dependent variables are correctly formatted.
2. **Model Specification**: Choose the appropriate independent variables based on domain knowledge and exploratory data analysis.
3. **Model Estimation**: Use an algorithm (often maximum likelihood estimation) to estimate the coefficients of the logistic regression model.
4. **Model Evaluation**: Evaluate the model using fit statistics, cross-validation, or other techniques to assess its performance and generalization ability.
5. **Model Interpretation**: Interpret the coefficients of the model to understand the relationship between the independent variables and the log odds of the outcome.
6. **Model Deployment**: Deploy the model for making predictions on new data or for use in decision-making processes.

# Analytics applications to various Business Domains:

- Applications of Data Modelling can be termed as Business analytics.
- Business analytics involves the collecting, sorting, processing, and studying of

business-related data using statistical models and iterative methodologies. The

goal of BA is to narrow down which datasets are useful and which can increase revenue, productivity, and efficiency.

- Business analytics (BA) is the combination of skills, technologies, and practices

used to examine an organization's data and performance as a way to gain insights

and make data-driven decisions in the future using statistical analysis.

Although business analytics is being leveraged in most commercial sectors and industries, the following applications are the most common.

1. Credit Card Companies

Credit and debit cards are an everyday part of consumer spending, and they are an ideal way of gathering information about a purchaser's spending habits, financial situation, behavior trends, demographics, and lifestyle preferences.

2. Customer Relationship Management (CRM)

Excellent customer relations is critical for any company that wants to retain customer loyalty to stay in business for the long haul. CRM systems analyze important performance indicators such as demographics, buying patterns, socio-economic information, and lifestyle.

3. Finance

The financial world is a volatile place, and business analytics helps to extract

insights that help organizations maneuver their way through tricky terrain.

Corporations turn to business analysts to optimize budgeting, banking, financial

planning, forecasting, and portfolio management.

4. Human Resources

Business analysts help the process by pouring through data that characterizes high-performingcandidates, such as educational background, attrition rate, the average

length of employment, etc. By working with this information, business analysts help

HR by forecasting the best fits between the company and candidates

5. Manufacturing:

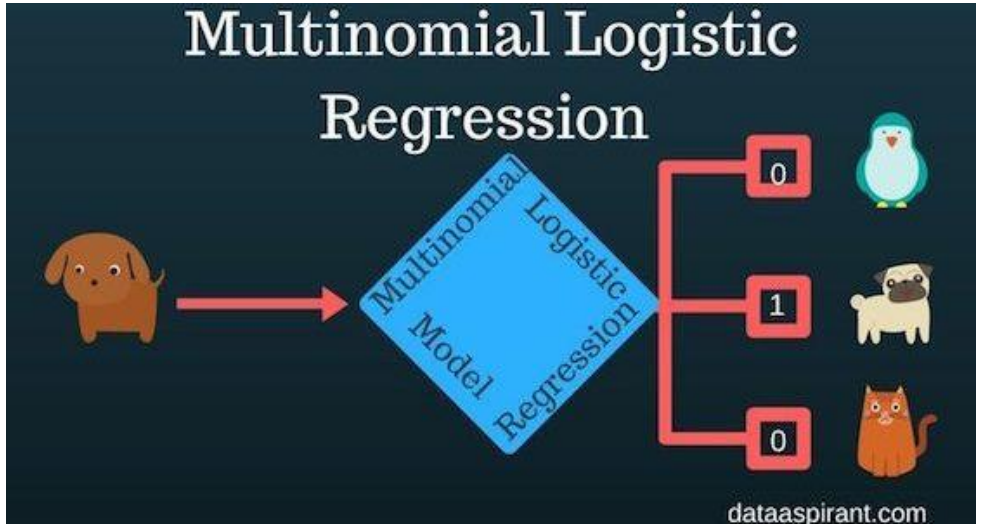
Business analysts work with data to help stakeholders understand the things that affect operations and the bottom line. Identifying things like equipment downtime, inventory levels, and maintenance costs helps companies streamline inventory management, risks, and supply-chain management to create maximum efficiency.

6. Marketing

Business analysts help answer these questions and so many more, by measuring marketing and advertising metrics, identifying consumer behavior and the target audience, and analyzing market trends.

# Multinomial Logistic Regression

- A multinomial logistic regression (or multinomial regression for short) is used when the outcome variable being predicted is nominal and has more than two categories that do not have a given rank or order.
- This model can be used with any number of independent variables that are categorical or continuous.



# What MNLR?

- Multinomial logistic regression is also a classification algorithm same like the logistic regression for binary classification. Whereas in logistic regression for binary classification the classification task is to predict the target class which is of binary type. Like **Yes/NO, 0/1, Male/Female.**

- When it comes to multinomial logistic regression. The idea is to use the logistic regression techniques to predicted the target class for **more than** 2 target classes.
- The underline technique will be same like the **logistic regression for binary classification** until calculating the probabilities for each target. Once the probabilities were calculated. We need to transfer them into **one hot encoding** and uses the cross entropy methods in the training process for getting the proper weights.

# Example

- Using the multinomial logistic regression. We can address different types of classification problems. Where the trained model is used to predict the target class from more than 2 target classes. Below are few examples to understand what kind of problems we can solve using the multinomial logistic regression.
    - Predicting the Iris flower species type
        - **Targets:** different species
    - Predicting the animal category using the given animal features
        - **Targets:** Dog, Cat, Tiger, Lion

# Assumptions

1. Independence of observations
2. Categories of the outcome variable must be mutually exclusive and exhaustive
3. No multicollinearity between independent variables
4. Linear relationship between continuous variables and the logit transformation of the outcome variable
5. No outliers or highly influential points

1. **Mutually Exclusive**: The categories or groups into which the variable is divided should not overlap. In other words, each observation should only fall into one and only one category. There should be no ambiguity or possibility of an observation belonging to multiple categories simultaneously.

For example, if you're categorizing people by their education level into "High School Graduate," "College Graduate," and "Postgraduate," these categories should be mutually exclusive. A person cannot be both a "College Graduate" and a "Postgraduate" simultaneously; they should fit into only one category.

**2. Exhaustive**: This implies that all possible outcomes or scenarios should be covered by the categories defined within the variable. No residual category should be needed to account for observations that do not fit into any defined categories.

For instance, if you're categorizing people by their employment status into "Employed," "Unemployed," and "Student," these categories should be exhaustive. Every person's employment status should fit into one of these categories, leaving no one unaccounted for. There should not be any other potential employment status that is not covered by these categories.

# Multinomial Logistic Regression Workflow/ Stages:

- Inputs
- Linear model
- Logits
- Softmax Function
- Cross Entropy
- One-Hot-Encoding

# Inputs

- The inputs to the multinomial logistic regression are the features we have in the dataset. Suppose if we are going to **predict the Iris flower species type**, the features will be the flower **sepal length, width** and **petal length and width** parameters will be our features. These features will treat as the inputs for the multinomial logistic regression.
- The keynote to remember here is the features values always numerical. If the features are not numerical, we need to convert them into numerical values using the proper categorical data analysis techniques.
- **Just a simple example:** If the feature is color and having different attributes of the color features are RED, BLUE, YELLOW, ORANGE. Then we can assign an integer value to each attribute of the features like for RED we can assign **1.** For BLUE we can assign the value **2** likewise of the other attributes for the color feature. Later we can use the numerically converted values as the inputs for the classifier.

# Linear Model

- The linear model equation is the same as the linear equation in the **linear regression** model. You can see this linear equation in the image. Where the **X** is the set of inputs, Suppose from the image we can say X is a matrix. Which contains all the feature( numerical values) X = [x1,x2,x3]. Where W is another matrix includes the same input number of weights W = [w1,w2,w3].
- In this example, the linear model output will be the **w1*x1, w2*x2, w3*x3**
- The weights w1, w2, w3, w4 will update in the training phase. We will learn about this in the parameters optimization section of this article.

# Logits

- The Logits also called as scores. These are just the outputs of the linear model. The Logits will change with the changes in the calculated weights.

# Softmax function

- The Softmax function is a probabilistic function that calculates the probabilities for the given score. Using the softmax function return the high probability value for the high scores and fewer probabilities for the remaining scores. This we can observe from the image. For the logits **0.5, 1.5, 0.1** the calculated probabilities using the softmax function are **0.2, 0.7, 0.1**
- For Logit **1.5**, we are getting a high probability value **0.7** and a very less probability value for the remaining Logits 0.5 and 0.1

# Cross Entropy

- The cross entropy is the last stage of multinomial logistic regression. It uses the **cross-entropy function** to find the **similarity distance** between the probabilities calculated from the softmax function and the target one-hot-encoding matrix.
- Before we learn more about Cross Entropy, let's understand what it means by a One-Hot-Encoding matrix.

# One hot encoding

- One-Hot Encoding is a method to represent the target values or categorical attributes into a binary representation. From this article main image, where the input is the dog image, the target having 3 possible outcomes like **bird, dog, cat.** Where you can find the one-hot-encoding matrix like [0, 1, 0].
- The one-hot-encoding matrix is so simple to create. For every input features (x1, x2, x3) the one-hot-encoding matrix is with the values of 0 and the 1 for the target class. The total number of values in the one-hot-encoding matrix and the unique target classes are the same.
- Suppose if we have 3 input features like x1, x2, and x3 and one target variable (With 3 target classes). Then the one-hot-encoding matrix will have 3 values. Out of **3** values, one value will be **1** and all other will be **0s.**
- You will know where to place the 1 and where to place the 0 value from the training dataset. Let's take one observation from the training dataset which contains values for x1, x2, x3 and what will be the target class for that observation. The one-hot-encoding matrix will be having 1 for the target class for that observation and 0s for other.

| id | color |
|---|---|
| 1 | red |
| 2 | blue |
| 3 | green |
| 4 | blue |


| id | color_red | color_blue | color_green |
|---|---|---|---|
| 1 | 1 | 0 | 0 |
| 2 | 0 | 1 | 0 |
| 3 | 0 | 0 | 1 |
| 4 | 0 | 1 | 0 |

One Hot Encoding a simple categorical feature (Image by author)

# Cross-entropy

- The Cross-entropy is a **distance calculation function** which takes the calculated probabilities from softmax function and the created one-hot-encoding matrix to calculate the distance. For the right target class, the distance value will be less, and the distance values will be larger for the wrong target class.
- With this, we discussed each stage of the multinomial logistic regression. All the stages/ workflow will happen for each observation in the training set.

# Unit 4

## Unit – IV: Object Segmentation:

Regression Vs Segmentation – Supervised and Unsupervised Learning, Tree Building – Regression, Classification, Overfitting, Pruning and Complexity, Multiple Decision Trees, etc.

Time Series Methods: Arima, Measures of Forecast Accuracy, STL approach, Extract features from the generated model as Height, Average Energy, etc and Analyze for prediction.

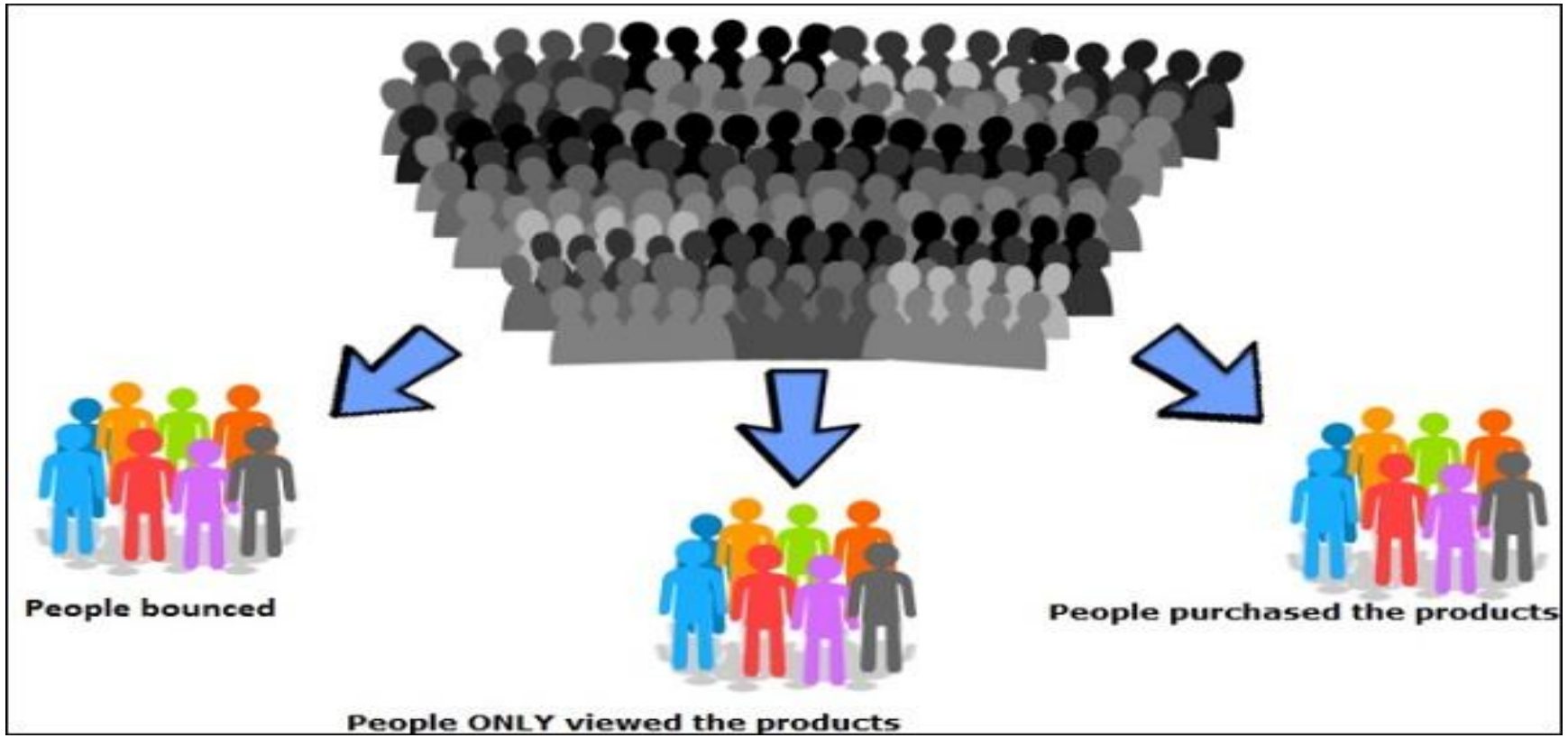
# Regression Vs Segmentation

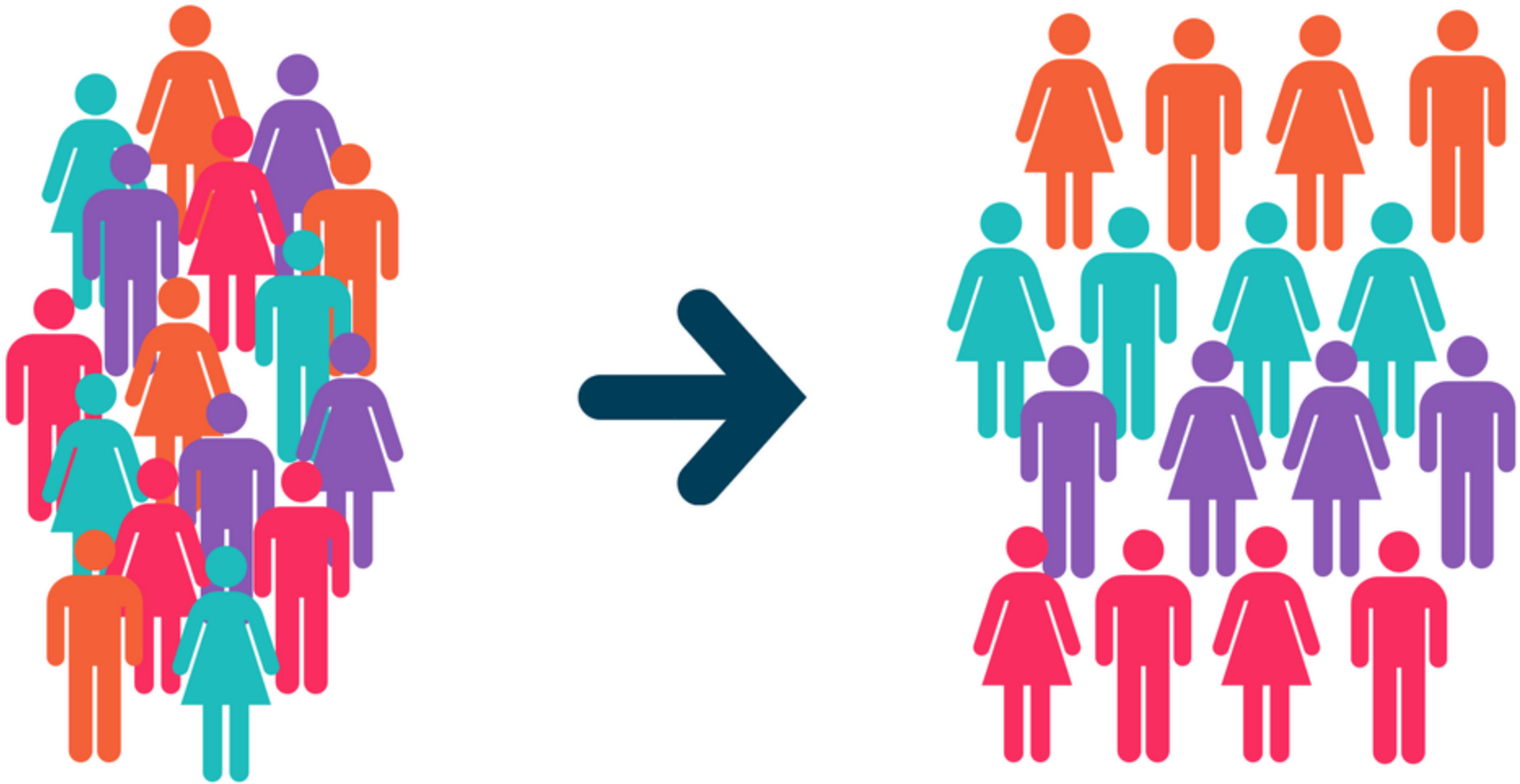
- Segmentation and regression are both statistical techniques used in data analysis, but they serve different purposes and have different methodologies:

# Segmentation

- Segmentation involves dividing a dataset into distinct groups or segments based on certain characteristics or variables.
- The goal of segmentation is to identify meaningful patterns or clusters within the data that can help in understanding customer behavior, market trends, or other phenomena.
- Segmentation can be performed using various techniques, such as clustering algorithms (e.g., k-means clustering, hierarchical clustering) or decision trees.
- These techniques group observations into segments based on similarities in their attributes or characteristics.
- Segmentation is often used in marketing to identify target customer segments, in healthcare to classify patient groups, or in finance to group investors based on risk profiles.

- Segmentation is the process that segregates the data to find the actionable items. For example, you can categorize your entire website traffic data as one segment for a “Country,” and one for a specific City. For the users, you can make the segments as one who purchased your products; one who only visited your website, and likewise. During the remarketing, you can target those audiences with the help of this segment.





- There are two broad sets of methodologies for segmentation:
- Objective (supervised) segmentation
- Non-objective (unsupervised) segmentation

**Objective Segmentation**

- Segmentation to identify the type of customers who would respond to a particular
- offer.
- Segmentation to identify high spenders among customers who will use the e-
- commerce channel for festive shopping.
- Segmentation to identify customers who will default on their credit obligation for a
- loan or credit card.

# Non-Objective Segmentation

- Segmentation of the customer base to understand the specific profiles which exist within the customer base so that multiple marketing actions can be personalized for each segment
- Segmentation of geographies on the basis of affluence and lifestyle of people living in each geography so that sales and distribution strategies can be formulated accordingly.
- Hence, it is critical that the segments created on the basis of an objective segmentation methodology must be different with respect to the stated objective (e.g. response to an offer).
- However, in case of a non-objective methodology, the segments are different with respect to the “generic profile” of observations belonging to each segment, but not with regards to any specific outcome of interest.
- The most common techniques for building non-objective segmentation are cluster

  analysis, K nearest neighbor techniques etc

# Regression

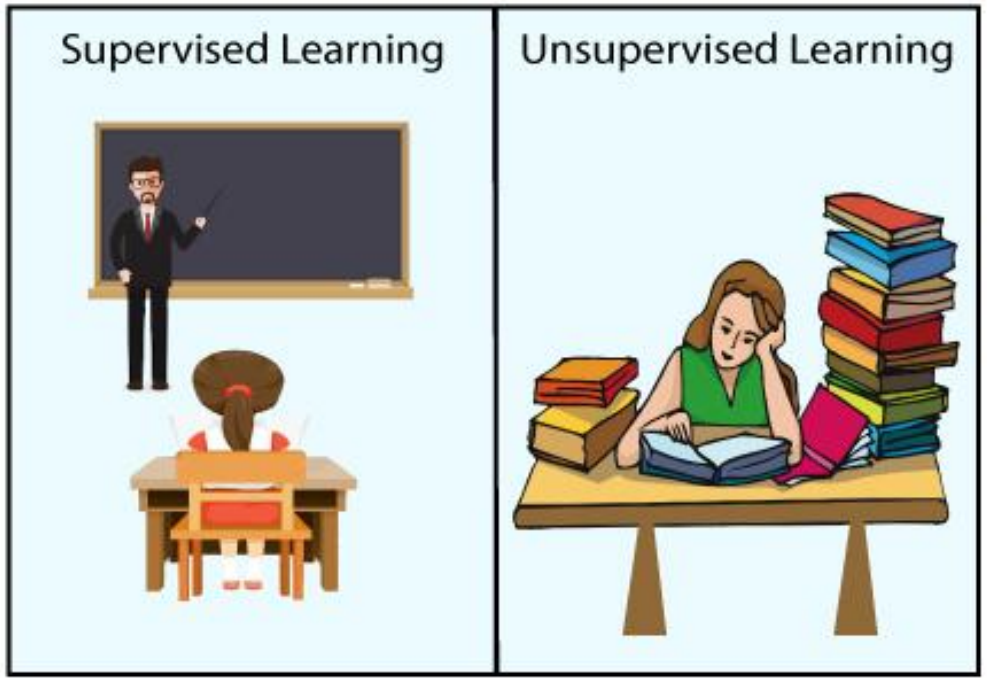
- Regression analysis is a statistical method used to model the relationship between a dependent variable and one or more independent variables.
- The goal of regression analysis is to quantify the effect of independent variables on the dependent variable and make predictions or inferences based on the model.
- Regression analysis provides estimates of the coefficients of the independent variables, which represent the strength and direction of their relationship with the dependent variable. Common regression techniques include linear regression, logistic regression, and polynomial regression.
- Regression analysis is used in various fields, including economics, social sciences, engineering, and finance, to analyze relationships between variables, make predictions, and test hypotheses.

# Summary

- Segmentation involves grouping data into distinct segments or clusters based on similarities.
- Regression analysis focuses on modeling the relationship between variables and making predictions based on that relationship.
- Both techniques are valuable tools in data analysis and are used in different contexts to gain insights from data.

# Supervised and Unsupervised Learning

- Supervised and Unsupervised learning are the two techniques of machine learning. But both the techniques are used in different scenarios and with different datasets.



# Supervised learning

- Supervised learning is a machine learning method in which models are trained using labeled data. In supervised learning, models need to find the mapping function to map the input variable (X) with the output variable (Y).

$$Y= f(X)$$

- Supervised learning needs supervision to train the model, which is similar to as a student learns things in the presence of a teacher. Supervised learning can be used for two types of problems: Classification and Regression.

# Example: Supervised learning

**Example:** Suppose we have an image of different types of fruits. The task of our supervised learning model is to identify the fruits and classify them accordingly. So to identify the image in supervised learning, we will give the input data as well as output for that, which means we will train the model by the shape, size, color, and taste of each fruit.

Once the training is completed, we will test the model by giving the new set of fruit. The model will identify the fruit and predict the output using a suitable algorithm.

# Unsupervised learning

- Unsupervised learning is another machine learning method in which patterns are inferred from the unlabeled input data.
- The goal of unsupervised learning is to find the structure and patterns from the input data. Unsupervised learning does not need any supervision. Instead, it finds patterns from the data on its own.
- Unsupervised learning can be used for two types of problems: Clustering and Association.

Example: Unlike supervised learning, here we will not provide any supervision to the model. We will just provide the input dataset to the model and allow the model to find the patterns from the data. With the help of a suitable algorithm, the model will train itself and divide the fruits into different groups according to the most similar features between them.

## Clustering

- Clustering is the type of Unsupervised Learning where we find hidden patterns in the data based on their similarities or differences. These patterns can relate to the shape, size, or color and are used to group data items or create clusters.
- There are several types of clustering algorithms, such as exclusive, overlapping, hierarchical, and probabilistic.

## Association

- Association is the kind of Unsupervised Learning where we can find the relationship of one data item to another data item. We can then use those dependencies and map them in a way that benefits us—e.g., understanding consumers' habits regarding our products can help us develop better cross-selling strategies.
- The association rule is used to find the probability of co-occurrence of items in a collection. These techniques are often utilized in customer behavior analysis in e-commerce websites and OTT platforms.

| Supervised Learning | Unsupervised Learning |
| --- | --- |
| Supervised learning algorithms are trained using labeled data. | Unsupervised learning algorithms are trained using unlabeled data. |
| Supervised learning model takes direct feedback to check if it is predicting correct output or not. | Unsupervised learning model does not take any feedback. |
| Supervised learning model predicts the output. | Unsupervised learning model finds the hidden patterns in data. |
| In supervised learning, input data is provided to the model along with the output. | In unsupervised learning, only input data is provided to the model. |

| | |
|---|---|
| Supervised learning needs supervision to train the model. | Unsupervised learning does not need any supervision to train the model. |
| Supervised learning can be categorized in Classification and Regression problems. | Unsupervised Learning can be classified in Clustering and Associations problems. |
| Supervised learning can be used for those cases where we know the input as well as corresponding outputs. | Unsupervised learning can be used for those cases where we have only input data and no corresponding output data. |
| Supervised learning model produces an accurate result. | Unsupervised learning model may give less accurate result as compared to supervised learning. |
| Supervised learning is not close to true Artificial intelligence as in this, we first train the model for each data, and then only it can predict the correct output. | Unsupervised learning is more close to the true Artificial Intelligence as it learns similarly as a child learns daily routine things by his experiences. |
| It includes various algorithms such as Linear Regression, Logistic Regression, Support Vector Machine, Multi-class Classification, Decision tree, Random Forest, Decision Trees , Bayesian Logic, etc. | In Unsupervised Learing we have K-means clustering. KNN (k-nearest neighbors), Hierarchal clustering, Anomaly detection, Neural Networks, Principle Component Analysis, Independent Component Analysis, Apriori algorithms, etc. |

# Decision Tree Classification Algorithm

- Decision Tree is a supervised learning technique that can be used for both classification and Regression problems, but it is mostly preferred for solving Classification problems.
- Decision Trees usually mimic human thinking ability while making a decision, making it easy to understand.
- A decision tree asks a question, and based on the answer (Yes/No), it further splits the tree into subtrees.
- It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.

- It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.
- In a Decision tree, there are two nodes, which are the Decision Node and the Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.

Basic Decision Tree Learning Algorithm:

- Now that we know what a Decision Tree is, we'll see how it works internally. Many algorithms out there construct Decision Trees, but one of the best

is called as ID3 Algorithm. ID3 Stands for Iterative Dichotomiser 3.

## There are two main types of Decision Trees:

1. Classification trees (Yes/No types)

What we've seen above is an example of a classification tree, where the outcome was a variable like 'fit' or 'unfit'. Here the decision variable is Categorical.

2. Regression trees (Continuous data types)

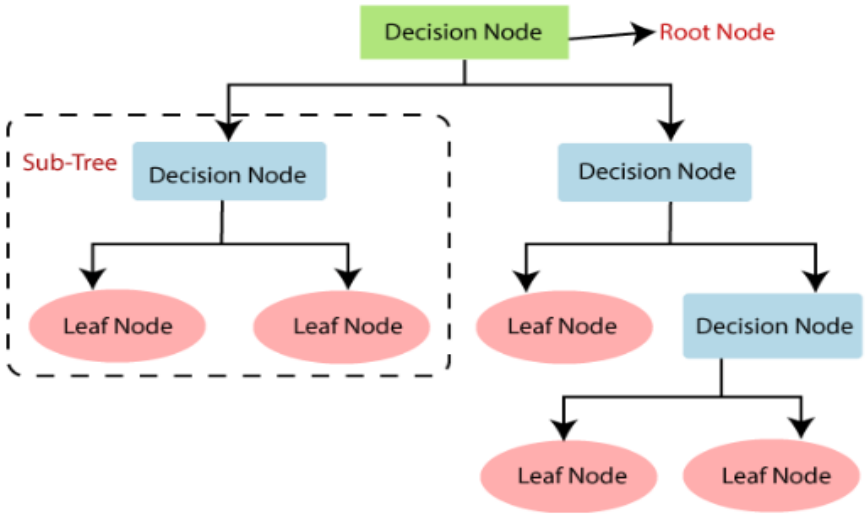
Here the decision or the outcome variable is Continuous, e.g. a number like 123.

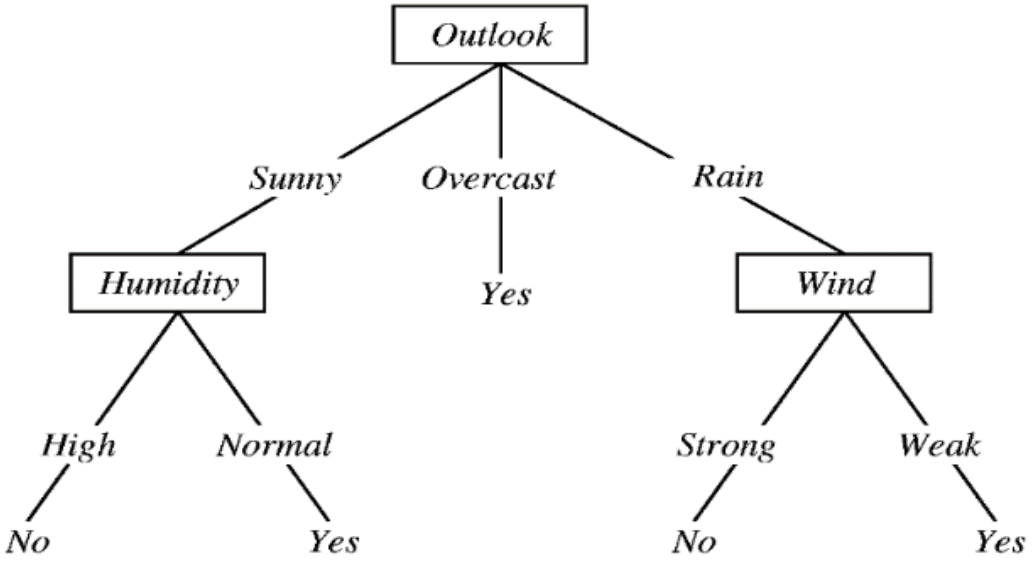
# Decision Tree Terminologies

- Root Node: Root node is from where the decision tree starts. It represents the entire dataset, which further gets divided into two or more homogeneous sets.
- Leaf Node: Leaf nodes are the final output node, and the tree cannot be segregated further after getting a leaf node.
- Splitting: Splitting is the process of dividing the decision node/root node into sub-nodes according to the given conditions.
- Branch/Sub Tree: A tree formed by splitting the tree.
- Pruning: Pruning is the process of removing the unwanted branches from the tree.
- Parent/Child node: The root node of the tree is called the parent node, and other nodes are called the child nodes.

# Decision Tree Representation

- Each non-leaf node is connected to a test that splits its set of possible answers into subsets corresponding to different test results.
- Each branch carries a particular test result's subset to another node.
- Each node is connected to a set of possible answers.



# Appropriate Problems for Decision Tree Learning

Decision tree learning is generally best suited to problems with the following characteristics:

- Instances are represented by attribute-value pairs.
  - There is a finite list of attributes (e.g. hair colour) and each instance stores a value for that attribute (e.g. blonde).
  - When each attribute has a small number of distinct values (e.g. blonde, brown, red) it is easier for the decision tree to reach a useful solution.
  - The algorithm can be extended to handle real-valued attributes (e.g. a floating point temperature)
- The target function has discrete output values.
  - A decision tree classifies each example as one of the output values.
    - Simplest case exists when there are only two possible classes (Boolean classification).
    - However, it is easy to extend the decision tree to produce a target function with more than two possible output values.

# Appropriate Problems for Decision Tree Learning

  - Although it is less common, the algorithm can also be extended to produce a target function with real-valued outputs.
- Disjunctive descriptions may be required.
  - Decision trees naturally represent disjunctive expressions.
- The training data may contain errors.
  - Errors in the classification of examples, or in the attribute values describing those examples are handled well by decision trees, making them a robust learning method.
- The training data may contain missing attribute values.
  - Decision tree methods can be used even when some training examples have unknown values (e.g., humidity is known for only a fraction of the examples).

Tree Building: Decision tree learning is the construction of a decision tree from class-labeled training tuples. A decision tree is a flow-chart-like structure, where each internal (non-leaf) node denotes a test on an attribute, each branch represents the outcome of a test, and each leaf (or terminal) node holds a class label. The topmost node in a tree is the root node. There are many specific decision-tree algorithms. Notable ones include the following.

ID3 → (extension of D3)

C4.5 → (successor of ID3)

CART → (Classification And Regression Tree)

CHAID → (Chi-square automatic interaction detection Performs multi-level splits when computing classification trees)

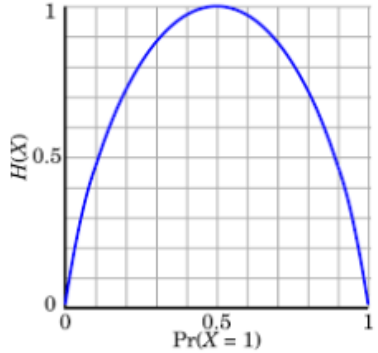
MARS → (multivariate adaptive regression splines): Extends decision trees to handle numerical data better

Conditional Inference Trees → Statistics-based approach that uses non-parametric tests as splitting criteria, corrected for multiple testing to avoid over fitting.

The complete process can be better understood using the below algorithm:

- Step-1: Begin the tree with the root node, says S, which contains the complete dataset.
- Step-2: Find the best attribute in the dataset using Attribute Selection Measure (ASM).
- Step-3: Divide the S into subsets that contains possible values for the best attributes.
- Step-4: Generate the decision tree node, which contains the best attribute.
- Step-5: Recursively make new decision trees using the subsets of the dataset created in
- Step -6: Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.

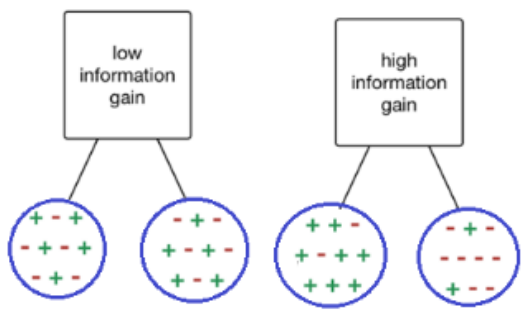
Entropy:




Entropy is a measure of the randomness in the information being processed. The higher the entropy, the harder it is to draw any conclusions from that information. Flipping a coin is an example of an action that provides information that is random.

From the graph, it is quite evident that the entropy H(X) is zero when the probability is either 0 or 1. The Entropy is maximum when the probability is 0.5 because it projects perfect randomness in the data and there is no chance if perfectly determining the outcome.

Information Gain




Information gain or IG is a statistical property that measures how well a given attribute separates the training examples according to their target classification. Constructing a decision tree is all about finding an attribute that returns the highest information gain and the smallest entropy.

ID3 follows the rule — A branch with an entropy of zero is a leaf node and A branch with entropy more than zero needs further splitting.

# Advantages of Decision Tree

- Simple to understand and interpret. People can understand decision tree models after a brief explanation.
- Requires little data preparation. Other techniques often require data normalization, dummy variables need to be created and blank values to be removed.
- Able to handle both numerical and categorical data. Other techniques are usually specialized in analyzing datasets that have only one type of variable. (For example, relation rules can be used only with nominal variables while neural networks can be used only with numerical variables.)

# Advantages of Decision Tree:

- Uses a white box model. If a given situation is observable in a model the explanation for the condition is easily explained by Boolean logic. (An example of a black box model is an artificial neural network since the explanation for the results is difficult to understand.)
- Possible to validate a model using statistical tests. That makes it possible to account for the reliability of the model.
- Robust: Performs well with large datasets. Large amounts of data can be analyzed

using standard computing resources in a reasonable time

# Example

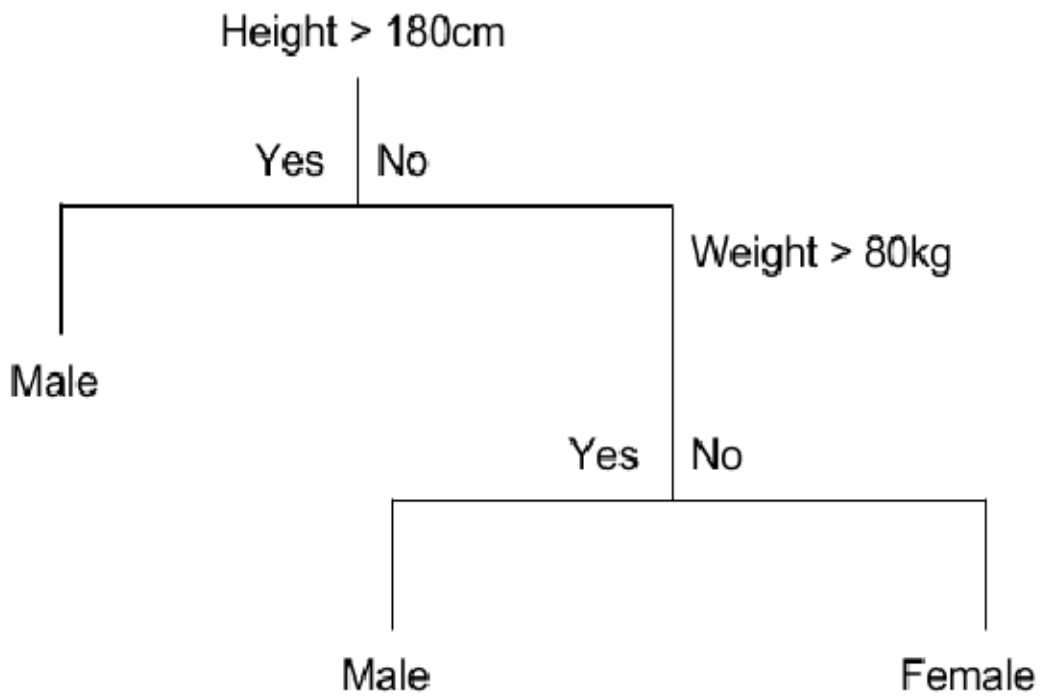
- Decision Tree Example: Solved Problem (PDF)

# Multiple Decision Trees: Classification & Regression Trees:

- Classification and regression trees is a term used to describe decision tree algorithms that are used for classification and regression learning tasks.
- The Classification and Regression Tree methodology, also known as the CART were introduced in 1984 by Leo Breiman, Jerome Friedman, Richard Olshen, and Charles Stone

Classification Trees:

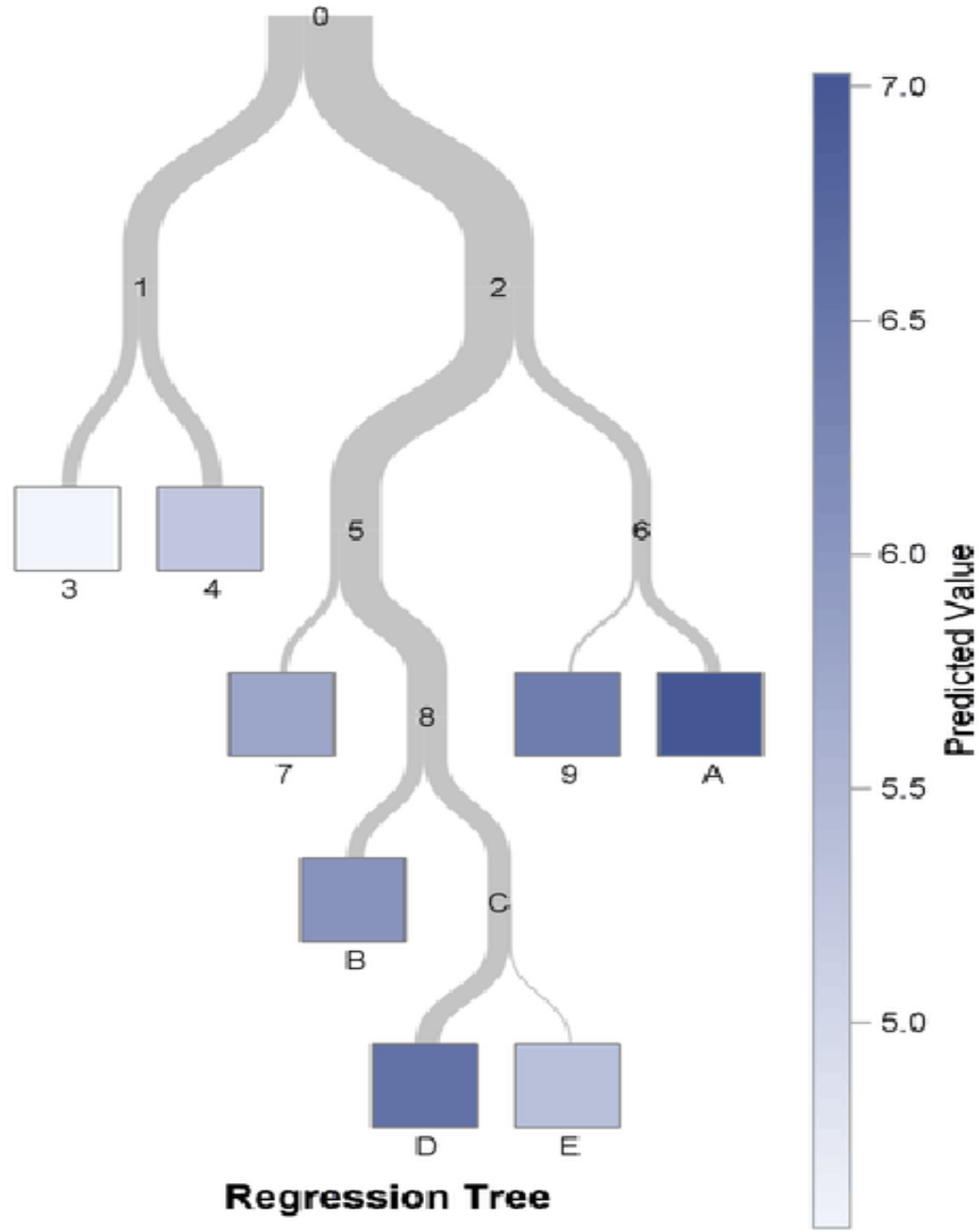
A classification tree is an algorithm where the target variable is fixed or categorical. The algorithm is then used to identify the "class" within which a target variable would most likely fall.




- ✓ An example of a classification-type problem would be determining who will or will not subscribe to a digital platform; or who will or will not graduate from high school.
- ✓ These are examples of simple binary classifications where the categorical dependent variable can assume only one of two, mutually exclusive values.

# Regression Trees

- ✓ A regression tree refers to an algorithm where the target variable is and the algorithm is used to predict its value which is a continuous variable.
- ✓ As an example of a regression type problem, you may want to predict the selling prices of a residential house, which is a continuous dependent variable.
- ✓ This will depend on both continuous factors like square footage as well as categorical factors.



Regression Tree

## Difference Between Classification and Regression Trees

- ✓ Classification trees are used when the dataset needs to be split into classes that belong to the response variable. In many cases, the classes Yes or No.
- ✓ In other words, they are just two and mutually exclusive. In some cases, there may be more than two classes in which case a variant of the classification tree algorithm is used.
- ✓ Regression trees, on the other hand, are used when the response variable is continuous. For instance, if the response variable is something like the price of a property or the temperature of the day, a regression tree is used.
- ✓ In other words, regression trees are used for prediction-type problems while classification trees are used for classification-type problems.

# CART (Classification And Regression Tree)

- CART( Classification And Regression Trees) is a variation of the decision tree algorithm.
- It can handle both classification and regression tasks. Scikit-Learn uses the Classification And Regression Tree (CART) algorithm to train Decision Trees (also called “growing” trees).
- CART was first produced by Leo Breiman, Jerome Friedman, Richard Olshen, and Charles Stone in 1984.

# CART(Classification And Regression Tree) for Decision Tree

- CART is a predictive algorithm used in Machine learning and it explains how the target variable's values can be predicted based on other variables. It is a decision tree where each fork is split into a predictor variable and each node has a prediction for the target variable at the end.
- The term CART serves as a generic term for the following categories of decision trees:
- Classification Trees: The tree is used to determine which "class" the target variable is most likely to fall into when it is categorical.
- Regression trees: These are used to predict a continuous variable's value.

# CART for Classification

- A classification tree is an algorithm where the target variable is categorical. The algorithm is then used to identify the “Class” within which the target variable is most likely to fall. Classification trees are used when the dataset needs to be split into classes that belong to the response variable(like yes or no)

# How Does CART for Classification Work?

- CART for classification works by recursively splitting the training data into smaller and smaller subsets based on certain criteria. The goal is to split the data in a way that minimizes the impurity within each subset. Impurity is a measure of how mixed up the data is in a particular subset. For classification tasks, CART uses Gini impurity
- **Gini Impurity-** Gini impurity measures the probability of misclassifying a random instance from a subset labeled according to the majority class. Lower Gini impurity means more purity of the subset.
- **Splitting Criteria-** The CART algorithm evaluates all potential splits at every node and chooses the one that best decreases the Gini impurity of the resultant subsets. This process continues until a stopping criterion is reached, like a maximum tree depth or a minimum number of instances in a leaf node.

# CART for Regression

- A Regression tree is an algorithm where the target variable is continuous and the tree is used to predict its value. Regression trees are used when the response variable is continuous. For example, if the response variable is the temperature of the day.

- CART for regression is a decision tree learning method that creates a tree-like structure to predict continuous target variables. The tree consists of nodes that represent different decision points and branches that represent the possible outcomes of those decisions. Predicted values for the target variable are stored in each leaf node of the tree.

# How Does CART works for Regression?

Regression CART works by splitting the training data recursively into smaller subsets based on specific criteria. The objective is to split the data in a way that minimizes the residual reduction in each subset.

- **Residual Reduction-** Residual reduction is a measure of how much the average squared difference between the predicted values and the actual values for the target variable is reduced by splitting the subset. The lower the residual reduction, the better the model fits the data.
- **Splitting Criteria-** CART evaluates every possible split at each node and selects the one that results in the greatest reduction of residual error in the resulting subsets. This process is repeated until a stopping criterion is met, such as reaching the maximum tree depth or having too few instances in a leaf node.

# GINI Impurity

- Gini impurity is a measure of the quality of a split in a decision tree algorithm. It measures the probability of misclassification of a random sample. The Gini impurity measures how often a randomly chosen element from the set would be incorrectly labeled if it were randomly labeled according to the distribution of labels in the subset.
- The Gini impurity of a node can be calculated as follows:

$$G = 1 - \sum_{i=1}^{c} p_i^2$$

where $c$ is the number of classes, and $p_i$ is the probability of a randomly chosen element in the node being labeled as class $i$.

- To illustrate how the Gini impurity is calculated, consider the following example. Suppose we have a binary classification problem, where we want to predict whether a person will buy a particular product based on their age and income.

| Age | Income | Buy Product? |
|---|---|---|
| 30 | 20000 | Yes |
| 40 | 50000 | Yes |
| 20 | 30000 | No |
| 50 | 60000 | No |
| 60 | 80000 | Yes |

- We want to build a decision tree to predict whether a person will buy the product based on their age and income. Suppose we want to split the data based on age, with a threshold of 35. The left child node will contain the data where age is less than or equal to 35, and the right child node will contain the data where age is greater than 35.

- The Gini impurity for the left child node can be calculated as follows:

$$G_{\text{left}} = 1 - (\frac{1}{2})^2 - (\frac{1}{2})^2 = 0.5$$

  There are two data points in the left child node, one of which buys the product, and one of which does not.
- Therefore, the probability of a randomly chosen element being labeled as "Yes" is 0.5, and the probability of it being labeled as "No" is also 0.5.
- The Gini impurity for the right child node can be calculated as follows:

$$G_{\text{right}} = 1 - (\frac{2}{3})^2 - (\frac{1}{3})^2 \approx 0.444$$

  There are three data points in the right child node, two of which buy the product, and one of which does not.
- Therefore, the probability of a randomly chosen element being labeled as "Yes" is 2/3, and the probability of it being labeled as "No" is 1/3.

- The overall Gini impurity for the split can be calculated as a weighted average of the Gini impurities for the child nodes, where the weights are proportional to the number of data points in each node.

$$G_{\text{split}} = \frac{2}{5} G_{\text{left}} + \frac{3}{5} G_{\text{right}} \approx 0.48$$

- The decision tree algorithm will try different thresholds and different features to find the split that minimizes the Gini impurity.

# Advantages of CART

- Can handle both numerical and categorical data. Can also handle multi-output problems.
- Decision trees require relatively little effort from users for data preparation.
- Nonlinear relationships between parameters do not affect tree performance.

# Disadvantages of CART

- Cannot guarantee to return the globally optimal decision tree.
- Can Over-fit on the training dataset.
- The tree structure may be unstable.

Example:

https://sumit-kr-sharma.medium.com/understanding-decision-trees-cart-algorithm-machine-learning-748f0c2249a6

# Decision tree pruning:

Pruning is a data compression technique in machine learning and search algorithms that reduces the size of decision trees by removing sections of the tree that are non-critical and redundant to classify instances. Pruning reduces the complexity of the final classifier, and hence improves predictive accuracy by the reduction of overfitting.

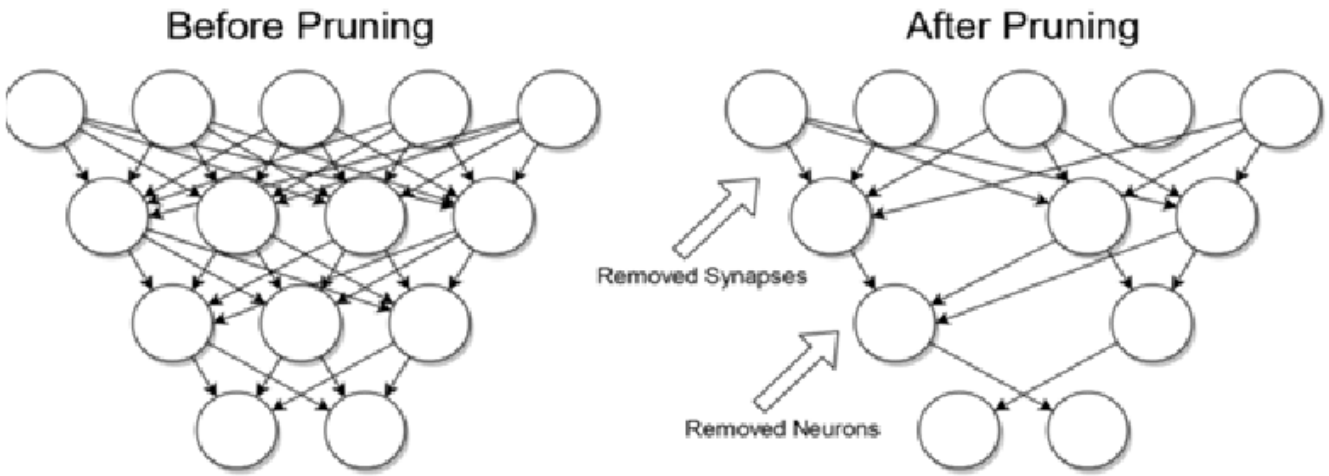



Fig: Before and After pruning

- One of the questions that arises in a decision tree algorithm is the optimal size of the final tree.
- A tree that is too large risks overfitting the training data and poorly generalizing to new samples.
- A small tree might not capture important structural information about the

  sample space. However, it is hard to tell when a tree algorithm should stop

  because it is impossible Before and After pruning to tell if the addition of a single extra node will dramatically decrease error. This problem is known as the horizon effect.
- A common strategy is to grow the tree until each node contains a small number of instances then use pruning to remove nodes that do not provide additional information.
- Pruning should reduce the size of a learning tree without reducing predictive accuracy as measured by a cross- validation set. There are many techniques for tree pruning that differ in the measurement that is used to optimize performance

Pruning Techniques:

Pruning processes can be divided into two types: PrePruning & Post Pruning

- Pre-pruning procedures prevent a complete induction of the training set by replacing a stop () criterion in the induction algorithm (e.g. max. Tree depth or information gain (Attr)> minGain). They considered to be more efficient because they do not induce an entire set, but rather trees remain small from the start.
- Post-Pruning (or just pruning) is the most common way of simplifying trees. Here, nodes and subtrees are replaced with leaves to reduce complexity.

The procedures are differentiated on the basis of their approach in the tree: Top-down approach & Bottom-Up approach

Bottom-up pruning approach:

- These procedures start at the last node in the tree (the lowest point).
- Following recursively upwards, they determine the relevance of each individual node.
- If the relevance for the classification is not given, the node is dropped or replaced by a leaf.
- The advantage is that no relevant sub-trees can be lost with this method.
- These methods include Reduced Error Pruning (REP), Minimum Cost Complexity Pruning (MCCP), or Minimum Error Pruning (MEP).

Top-down pruning approach:

- In contrast to the bottom-up method, this method starts at the root of the tree. Following the structure below, a relevance check is carried out which decides whether a node is relevant for the classification of all n items or not.
- By pruning the tree at an inner node, it can happen that an entire sub-tree (regardless of its relevance) is dropped.
- One of these representatives is pessimistic error pruning (PEP), which brings quite good results with unseen items.

# Overfitting

## What is Overfitting?

- Overfitting is an undesirable machine learning behavior that occurs when the machine learning model gives accurate predictions for training data but not for new data.
- When data scientists use machine learning models for making predictions, they first train the model on a known data set. Then, based on this information, the model tries to predict outcomes for new data sets.
- An overfit model can give inaccurate predictions and cannot perform well for all types of new data.

# Why does overfitting occur?

- You only get accurate predictions if the machine learning model generalizes to all types of data within its domain. Overfitting occurs when the model cannot generalize and fits too closely to the training dataset instead.

Overfitting happens due to several reasons, such as:

- The training data size is too small and does not contain enough data samples to accurately represent all possible input data values.
- The training data contains large amounts of irrelevant information, called noisy data.
- The model trains for too long on a single sample set of data.
- The model complexity is high, so it learns the noise within the training data.

## Overfitting examples

- Consider a use case where a machine learning model has to analyze photos and identify the ones that contain dogs in them. If the machine learning model was trained on a data set that contained majority photos showing dogs outside in parks , it may may learn to use grass as a feature for classification, and may not recognize a dog inside a room.
- Another overfitting example is a machine learning algorithm that predicts a university student's academic performance and graduation outcome by analyzing several factors like family income, past academic performance, and academic qualifications of parents. However, the test data only includes candidates from a specific gender or ethnic group. In this case, overfitting causes the algorithm's prediction accuracy to drop for candidates with gender or ethnicity outside of the test dataset.

## How can you detect overfitting?

- The best method to detect overfit models is by testing the machine learning models on more data with with comprehensive representation of possible input data values and types. Typically, part of the training data is used as test data to check for overfitting. A high error rate in the testing data indicates overfitting. One method of testing for overfitting is given below.

## K-fold cross-validation

Cross-validation is one of the testing methods used in practice. In this method, data scientists divide the training set into K equally sized subsets or sample sets called folds. The training process consists of a series of iterations. During each iteration, the steps are:

1. Keep one subset as the validation data and train the machine learning model on the remaining K-1 subsets.
2. Observe how the model performs on the validation sample.
3. Score model performance based on output data quality.

Iterations repeat until you test the model on every sample set. You then average the scores across all iterations to get the final assessment of the predictive model.

# How can you prevent overfitting?

- You can prevent overfitting by diversifying and scaling your training data set or using some other data science strategies, like those given below.

**Early stopping**

Early stopping pauses the training phase before the machine learning model learns the noise in the data. However, getting the timing right is important; else the model will still not give accurate results.

**Pruning**

You might identify several features or parameters that impact the final prediction when you build a model. Feature selection—or pruning—identifies the most important features within the training set and eliminates irrelevant ones. For example, to predict if an image is an animal or human, you can look at various input parameters like face shape, ear position, body structure, etc. You may prioritize face shape and ignore the shape of the eyes.

## Ensembling

Ensembling combines predictions from several separate machine learning algorithms. Some models are called weak learners because their results are often inaccurate. Ensemble methods combine all the weak learners to get more accurate results. They use multiple models to analyze sample data and pick the most accurate outcomes. The two main ensemble methods are bagging and boosting. Boosting trains different machine learning models one after another to get the final result, while bagging trains them in parallel.

## Data augmentation

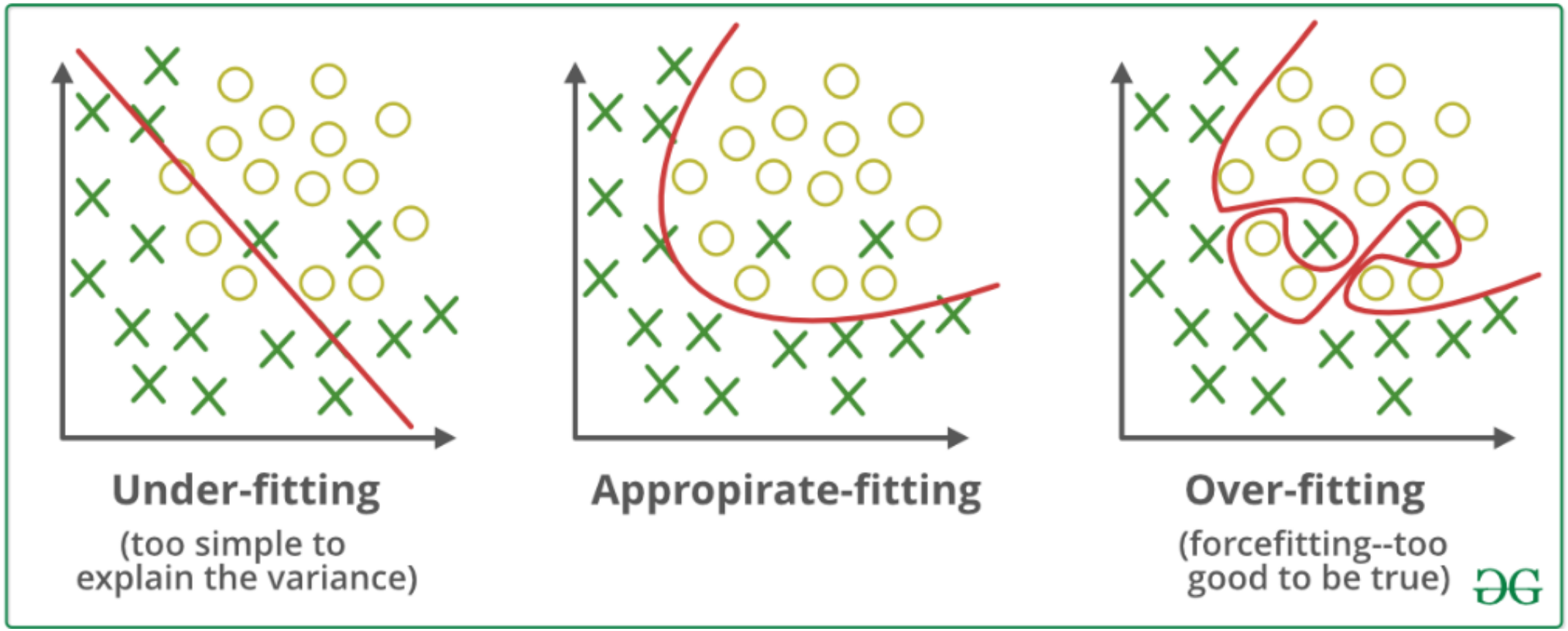
Data augmentation is a machine learning technique that changes the sample data slightly every time the model processes it. You can do this by changing the input data in small ways. When done in moderation, data augmentation makes the training sets appear unique to the model and prevents the model from learning their characteristics. For example, applying transformations such as translation, flipping, and rotation to input images.

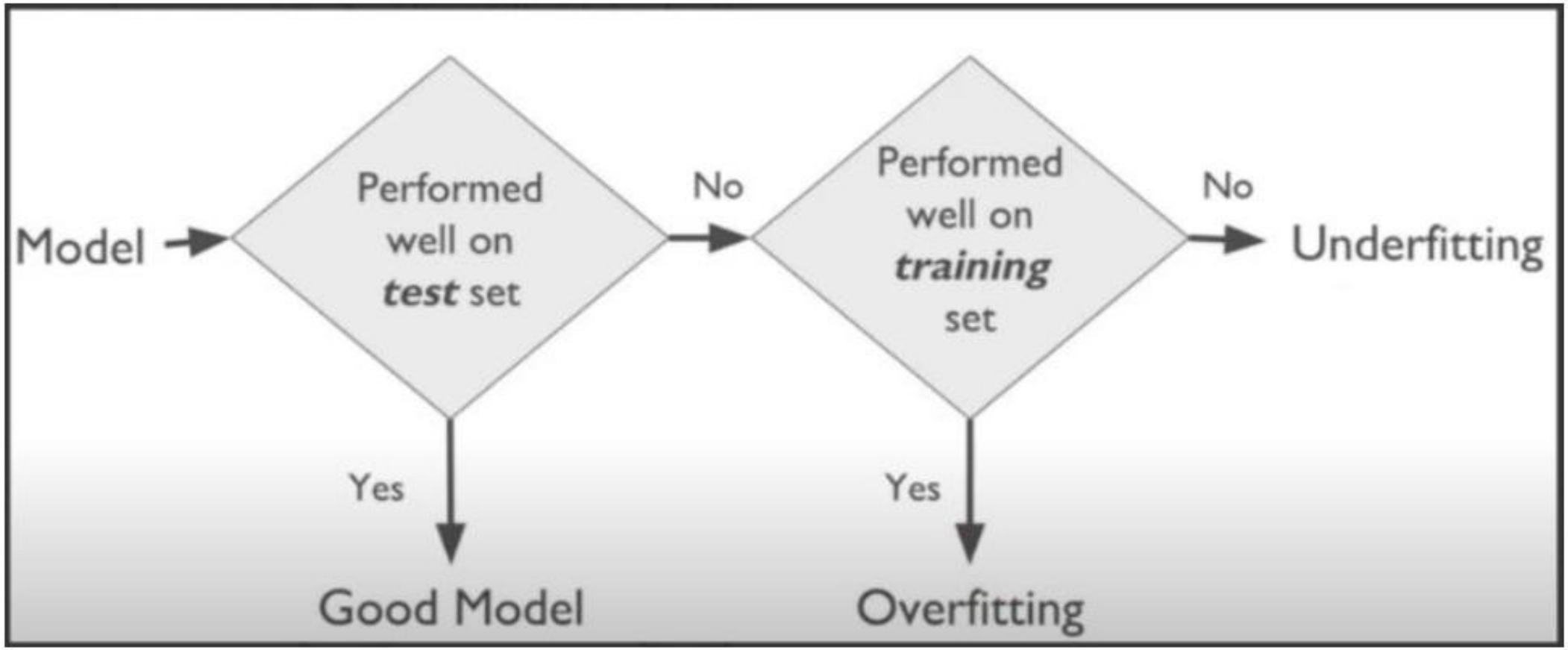
## What is underfitting?

- Underfitting is another type of error that occurs when the model cannot determine a meaningful relationship between the input and output data. You get underfit models if they have not trained for the appropriate length of time on a large number of data points.

## Underfitting vs. overfitting

Underfit models experience high bias—they give inaccurate results for both the training data and test set. On the other hand, overfit models experience high variance—they give accurate results for the training set but not for the test set. More model training results in less bias but variance can increase. Data scientists aim to find the sweet spot between underfitting and overfitting when fitting a model. A well-fitted model can quickly establish the dominant trend for seen and unseen data sets.


Under-fitting
(too simple to explain the variance)
Appropirate-fitting
Over-fitting
(forcefitting--too good to be true)


Model
Performed well on *test* set
No
Performed well on *training* set
No
Underfitting
Yes
Good Model
Yes
Overfitting

# Time Series Methods

- Time series forecasting focuses on analyzing data changes across equally spaced time intervals.
- Time series analysis is used in a wide variety of domains, ranging from econometrics to geology and earthquake prediction; it’s also used in almost all applied sciences and engineering.
- Time-series databases are highly popular and provide a wide spectrum of numerous applications such as stock market analysis, economic and sales forecasting, budget analysis, to name a few.
- They are also useful for studying natural phenomena like atmospheric pressure, temperature, wind speeds, earthquakes, and medical prediction for treatment.

- Time series data is data that is observed at different points in time. Time Series Analysis finds hidden patterns and helps obtain useful insights from the time series data.
- Time Series Analysis is useful in predicting future values or detecting anomalies from the data. Such analysis typically requires many data points to be present in the dataset to ensure consistency and reliability.

The different types of models and analyses that can be created through time series analysis are:

- Classification: To Identify and assign categories to the data.
- Curve fitting: Plot the data along a curve and study the relationships of variables present within the data.
- Descriptive analysis: Help Identify certain patterns in time-series data such as trends, cycles, or seasonal variation.
- Explanative analysis: To understand the data and its relationships, the dependent features, and cause and effect and its tradeoff.
- Exploratory analysis: Describe and focus on the main characteristics of the time series data, usually in a visual format.
- Forecasting: Predicting future data based on historical trends. Using the historical data as a model for future data and predicting scenarios that could happen along with the future plot points.
- Intervention analysis: The Study of how an event can change the data.
- Segmentation: Splitting the data into segments to discover the underlying properties from the source information.

## Components of Time Series:

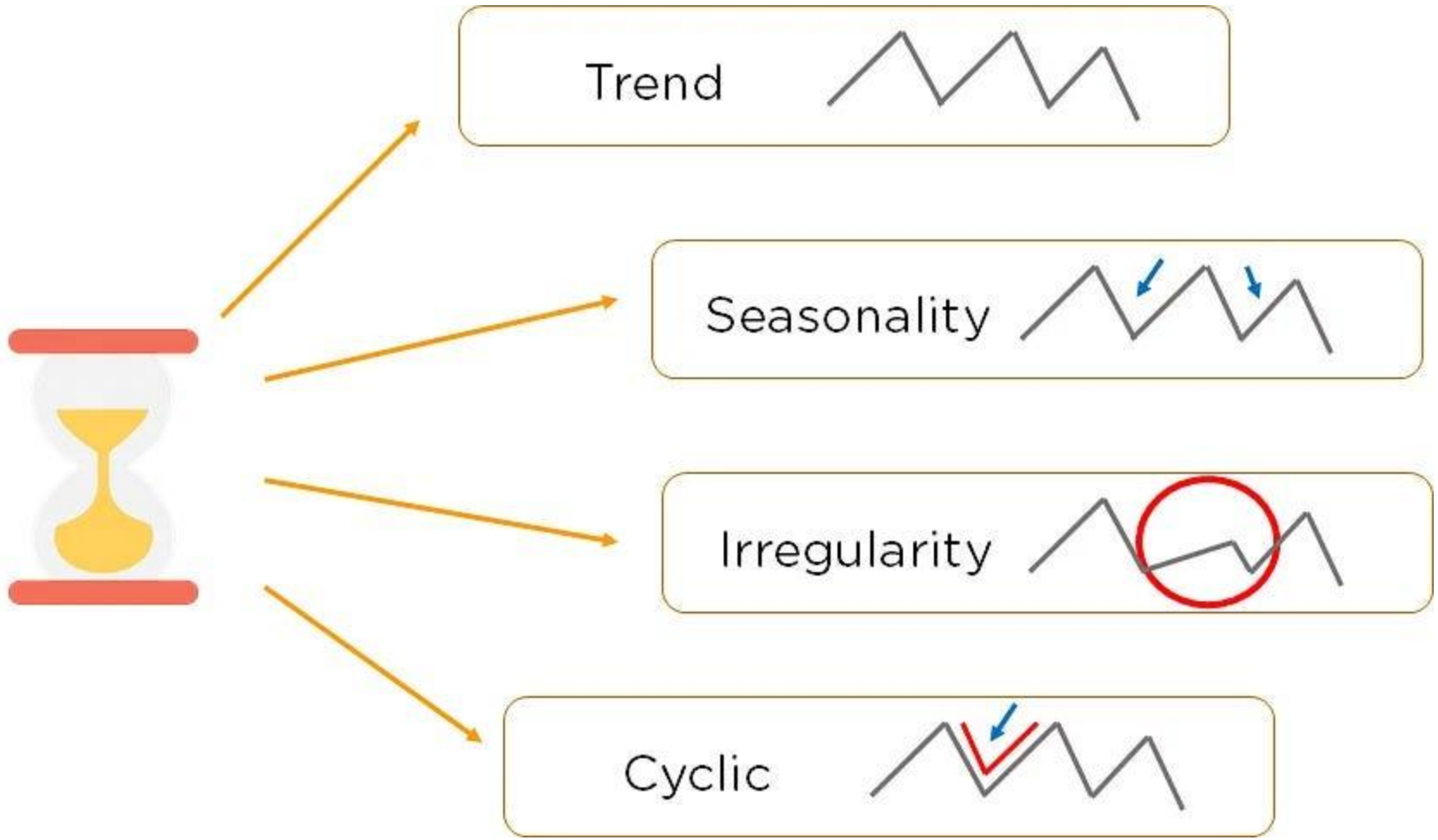
**Long term trend** – The smooth long term direction of time series where the data can increase or decrease in some pattern.

**Seasonal variation** – Patterns of change in a time series within a year which tends to repeat every year.

**Cyclical variation** – Its much alike seasonal variation but the rise and fall of time series over periods are longer than one year.

**Irregular variation** – Any variation that is not explainable by any of the three above mentioned components. They can be classified into – stationary and non – stationary variation.

**Stationary Variation:** When the data neither increases nor decreases, i.e. it's completely random it's called stationary variation. Or When the data has some explainable portion remaining and can be analyzed further then such case is called non – stationary variation.


Trend
Seasonality
Irregularity
Cyclic

# What Is an Autoregressive Integrated Moving Average (ARIMA)?

- An autoregressive integrated moving average, or ARIMA, is a statistical analysis model that uses time series data to either better understand the data set or to predict future trends.

- A statistical model is autoregressive if it predicts future values based on past values. For example, an ARIMA model might seek to predict a stock's future prices based on its past performance or forecast a company's earnings based on past periods.

- An autoregressive integrated moving average model is a form of regression analysis that gauges the strength of one dependent variable relative to other changing variables. The model's goal is to predict future securities or financial market moves by examining the differences between values in the series instead of through actual values.

An ARIMA model can be understood by outlining each of its components as follows:

- Autoregression (AR): refers to a model that shows a changing variable that regresses on its own lagged, or prior, values.
- Integrated (I): represents the differencing of raw observations to allow the time series to become stationary (i.e., data values are replaced by the difference between the data values and the previous values).
- Moving average (MA): incorporates the dependency between an observation and a residual error from a moving average model applied to lagged observations.

# ARIMA Parameters

- Each component in ARIMA functions as a parameter with a standard notation. For ARIMA models, a standard notation would be ARIMA with p, d, and q, where integer values substitute for the parameters to indicate the type of ARIMA model used. The parameters can be defined as:
- p: the number of lag observations in the model, also known as the lag order.
- d: the number of times the raw observations are differenced; also known as the degree of differencing.
- q: the size of the moving average window, also known as the order of the moving average.

For example,

A linear regression model includes the number and type of terms. A value of zero (0), which can be used as a parameter, would mean that particular component should not be used in the model. This way, the ARIMA model can be constructed to perform the function of an ARMA model, or even simple AR, I, or MA models.

## ARIMA and Stationary Data

- In an autoregressive integrated moving average model, the data are differenced in order to make it stationary. A model that shows stationarity is one that shows there is constancy to the data over time. Most economic and market data show trends, so the purpose of differencing is to remove any trends or seasonal structures.

Explanation:

1. **Seasonality**: This refers to the presence of regular and predictable patterns in data that occur over the course of a calendar year. For example, retail sales might increase during the holiday season or decrease during certain months due to seasonal factors.
2. **Negative Effect on Regression Model**: When there's seasonality in the data, it can negatively impact a regression model. Regression models attempt to identify relationships between variables, but if there are strong seasonal patterns in the data, the model might incorrectly attribute changes to these patterns rather than to the variables of interest.
3. **Trend and Stationarity**: A trend is a general direction in which data points are moving over time. Stationarity refers to a property of time series data where statistical properties such as mean and variance remain constant over time. If there's a trend in the data and it's not stationary, it means that the data's statistical properties are changing over time.
4. **Impact on Computations**: If there's a trend in the data and it's not stationary, it can affect computations in the regression modeling process. Stationarity is often an assumption in regression analysis, and if it's violated, the results may not be reliable or meaningful. This could lead to inaccurate predictions or interpretations of the data.

# How to Build an ARIMA Model

- To begin building an ARIMA model for an investment, you download as much of the price data as you can. Once you've identified the trends for the data, you identify the lowest order of differencing (d) by observing the autocorrelations. If the lag-1 autocorrelation is zero or negative, the series is already differenced. You may need to difference the series more if the lag-1 is higher than zero.
- Next, determine the order of regression (p) and order of moving average (q) by comparing autocorrelations and partial autocorrelations. Once you have the information you need, you can choose the model you'll use.

# Pros and Cons of ARIMA

ARIMA models have strong points and are good at forecasting based on past circumstances, but there are more reasons to be cautious when using ARIMA. In stark contrast to investing disclaimers that state "past performance is not an indicator of future performance...," ARIMA models assume that past values have some residual effect on current or future values and use data from the past to forecast future events.

**Pros**

- Good for short-term forecasting
- Only needs historical data
- Models non-stationary data

**Cons**

- Not built for long-term forecasting
- Poor at predicting turning points
- Computationally expensive
- Parameters are subjective

# What Is ARIMA Used for?

- ARIMA is a method for forecasting or predicting future outcomes based on a historical time series. It is based on the statistical concept of serial correlation, where past data points influence future data points.

**Measure of Forecast Accuracy:** Forecast Accuracy can be defined as the deviation of Forecast or Prediction from the actual results.

Error = Actual demand – Forecast OR $e_i = A_t - F_t$

We measure Forecast Accuracy by 2 methods : 1. Mean Forecast Error (MFE) For n time periods where we have actual demand and forecast values:

$$MFE = \frac{\sum_{i=1}^{n}(e_i)}{n}$$

Ideal value = 0; MFE > 0, model tends to under-forecast MFE < 0, model tends to over-forecast 2. Mean Absolute Deviation (MAD) For n time periods where we have actual demand and forecast values:

$$MAD = \frac{\sum_{i=1}^{n}(e_i)}{n}$$

While MFE is a measure of forecast model bias, MAD indicates the absolute size of the errors

# Uses of Forecast error:

Forecast error, or the discrepancy between predicted values and actual outcomes, serves several important purposes in various fields. Here are some of its key uses:

1. **Model Evaluation**: Forecast error helps assess the performance of forecasting models. By comparing predicted values with actual outcomes, analysts can determine the accuracy and reliability of the model. This evaluation is crucial for selecting the most appropriate model for future forecasts.
2. **Improving Forecasting Models**: Analyzing forecast errors provides insights into the strengths and weaknesses of forecasting models. By understanding where and why errors occur, analysts can refine models, adjust parameters, or incorporate additional variables to improve accuracy in future forecasts.
3. **Decision Making**: Forecast error informs decision-making processes by providing a measure of uncertainty associated with forecasts. Decision-makers can use this information to assess the risks and potential impact of relying on forecasted values for planning, resource allocation, inventory management, and other strategic decisions.
4. **Performance Monitoring**: Monitoring forecast errors over time helps track the performance of forecasting systems. Consistently high or increasing errors may signal changes in underlying patterns, such as shifts in demand or market dynamics, prompting organizations to adapt their strategies accordingly.
5. **Setting Realistic Expectations**: Forecast error highlights the inherent uncertainty in predictions and helps stakeholders set realistic expectations. Communicating forecast accuracy and potential variability in outcomes based on historical error analysis fosters transparency and avoids overreliance on overly optimistic forecasts.
6. **Continuous Improvement**: Utilizing forecast error as feedback fosters a culture of continuous improvement within organizations. By regularly analyzing errors and implementing corrective measures, teams can refine forecasting processes, enhance data quality, and ultimately achieve more accurate predictions over time.

# Seasonal Decomposition of Time Series by Loess (STL)

- Researchers can uncover trends, seasonal patterns, and other temporal linkages by employing time series data, which gathers information across many time-periods.
- Understanding the underlying patterns and components within time series data is crucial for making informed decisions and predictions.
- One common challenge in analyzing time series data is dealing with seasonality which is periodic fluctuations that occur at regular intervals.
- Seasonal patterns can obscure the overall trend and make it challenging to extract meaningful insights from the data.

# What is Seasonal Decomposition?

- Seasonal decomposition is a statistical technique for breaking down a time series into its essential components, which often include the trend, seasonal patterns, and residual (or error) components. The goal is to separate the different sources of variation within the data to understand better and analyze each component independently. The fundamental components are discussed below:
- **Trend:** The underlying long-term progression or direction in the data.
- **Seasonal:** The repeating patterns or cycles that occur at fixed intervals like daily, monthly or yearly.
- **Residual:** The random fluctuations or noise in the data that cannot be attributed to the trend or seasonal patterns.

# What is Loess (STL)?

- Locally Weighted Scatterplot Smoothing or Loess is a non-parametric regression method used for smoothing data.
- In the context of time series analysis, Seasonal-Trend decomposition using Loess (STL) is a specific decomposition method that employs the Loess technique to separate a time series into its trend, seasonal, and residual components.
- STL is particularly effective in handling time series data with complex and non-linear patterns. In STL, the decomposition is performed by iteratively applying Loess smoothing to the time series.
- This process helps capture both short-term and long-term variations in the data, making it a robust method for decomposing time series with irregular or changing patterns.

# Why to perform seasonal decomposition?

- There are various reasons for performing seasonal decomposition in Time-series data which are discussed below:

1. **Pattern Identification:** Seasonal decomposition allows analysts to identify and understand the underlying patterns within a time series. This is crucial for recognizing recurring trends, seasonal effects and overall data behavior.
2. **Forecasting:** Separating a time series into its components facilitates more accurate forecasting. By modeling the trend, seasonal patterns and residuals separately, it becomes possible to make predictions and projections based on the individual components.
3. **Anomaly Detection:** Detecting anomalies or unusual events in a time series is more effective when the data is decomposed. Anomalies are easier to identify when they stand out against the background of the trend and seasonal patterns.
4. **Statistical Analysis:** Seasonal decomposition aids in statistical analysis by providing a clearer picture of the structure of the time series. This, in turn, enables the application of various statistical methods and models.

# Components and Features

- STL is a popular method for decomposing time series data into three main components: trend, seasonal, and residual.

- Trend Component: This component captures the long-term trend or directionality of the data. It represents the underlying pattern that persists over time, abstracting away shorter-term fluctuations and noise.
- Seasonal Component: The seasonal component captures the periodic fluctuations in the data that occur at fixed intervals. It represents recurring patterns that repeat over a specific period, such as daily, weekly, monthly, or yearly cycles.
- Residual Component: The residual component represents the variability in the data that cannot be explained by the trend or seasonal components. It includes random fluctuations, noise, and any irregular patterns that remain after removing the trend and seasonal effects.

## Feature Extraction:

- Once you have decomposed your time series data using the STL approach, you can extract various features from each component for further analysis and prediction.

1. **Height (Trend Component)**:
    1. The Height of the time series can be interpreted as the range of the decomposed trend component. It represents the magnitude of the overall trend in the data.
    2. Calculate it by finding the difference between the maximum and minimum values of the trend component.
2. **Average Energy (Residual Component)**:
    1. The Average Energy of the time series can be calculated from the residual component. Since energy is related to the squared magnitude of the data points, you can compute the mean squared value of the residual component.
    2. It represents the overall level of variability or noise in the data that remains after accounting for the trend and seasonal effects.
3. **Other features**:
    1. You can explore additional features such as the strength of the seasonal component, autocorrelation of the residual component, or higher-order statistics like kurtosis and skewness.
    2. These features can provide further insights into the underlying structure and characteristics of the time series data.

## Height:

- The "Height" of the time series can be interpreted as the range of the decomposed trend component. You can calculate it by finding the difference between the maximum and minimum values of the trend component.

```
import numpy as np
import statsmodels.api as sm

# Assuming 'time_series_data' is your time series data
# Perform STL decomposition
stl_result = sm.tsa.STL(time_series_data).fit()

# Extract trend component
trend_component = stl_result.trend

# Calculate height
height = np.max(trend_component) - np.min(trend_component)
```

## Average Energy:

The "Average Energy" of the time series can be calculated from the residual component. Since energy is related to the squared magnitude of the data points, you can compute the mean squared value of the residual component

```
# Extract residual component
residual_component = stl_result.resid

# Calculate average energy
average_energy = np.mean(np.square(residual_component))
```

## Other features:

In addition to Height and Average Energy, you can explore other features such as the strength of the seasonal component, autocorrelation of the residual component, or higher-order statistics like kurtosis and skewness.

# Analysis for Prediction:

Once vou have extracted these features. vou can analvze them to understand the patterns and dynamics present in the data and use them for prediction purposes.

1. **Feature Importance**:
    1. Evaluate the importance of each feature in predicting future observations. Some features mav have more predictive power than others and can significantly impact the accuracy of your prediction models.
2. **Model Training**:
    1. Use the extracted features as input variables for training prediction models such as autoregressive integrated moving average (ARIMA). seasonal ARIMA (SARIMA), or machine learning models like random forests or gradient boosting machines.
    2. Train the models on historical data with the extracted features and use them to make predictions for future observations.
3. **Model Evaluation**:
    1. Validate the predictive performance of vour models using appropriate evaluation techniques such as cross-validation or out-of-sample testing.
    2. Assess how well the models capture the underlving patterns and dynamics present in the data and make accurate predictions for future time points

# Applications of STL in Real-World Scenarios

- STL decomposition has a wide range of applications across various industries and domains. Some examples of real-world scenarios where STL can be applied include:

1. **Energy consumption forecasting:** The ability to predict energy demand is of paramount importance in managing resources and optimizing energy production and distribution. By applying STL decomposition to energy consumption data, analysts can identify the underlying trend and seasonal patterns, enabling accurate forecasts of future energy needs. This knowledge facilitates efficient resource allocation, maintenance planning, and the implementation of demand response strategies.

**2. Financial market analysis:** In the fast-paced world of finance, understanding the trends and patterns in stock prices, exchange rates, and other financial indicators is crucial for making informed investment decisions. STL decomposition aids analysts in extracting the trend and seasonal components from the data, shedding light on long-term trends and cyclical patterns. By incorporating these insights, financial professionals can better gauge market movements, identify opportunities, and mitigate risks.

**3. Weather forecasting:** Weather patterns exhibit various seasonal and long-term trends, making accurate forecasting a challenging task. STL decomposition assists meteorologists in isolating the seasonal and trend components in climate data, providing a clearer understanding of the underlying patterns. By incorporating this knowledge into forecasting models, meteorologists can improve the accuracy of weather predictions, resulting in more reliable information for emergency preparedness, agriculture, and climate change analysis.

**4. Retail and sales forecasting:** Retailers and businesses heavily rely on accurate sales forecasting for inventory management, marketing strategies, and resource allocation. STL decomposition enables the identification and modeling of seasonal patterns in sales data, allowing for precise predictions of future sales volumes during specific time periods. Armed with this information, businesses can optimize stock levels, plan promotions, and efficiently allocate resources to meet consumer demand.

# Limitations and Considerations

- While STL decomposition is a powerful and flexible technique for time series analysis, it is essential to be aware of its limitations and considerations:

1. **Assumptions:** STL decomposition assumes a linear trend and additive seasonality in the time series data. However, in reality, time series may exhibit nonlinear trends or multiplicative seasonality. In such cases, STL may not yield accurate results, and alternative methods like seasonal-trend decomposition using LOESS (STL-LOESS) may be more suitable. It is essential to assess whether the assumptions align with the characteristics of the data at hand.
2. **Parameter selection:** The choice of parameters in STL decomposition, such as the seasonal cycle length and smoothing parameters, holds significant influence over its performance. Careful consideration and selection of these parameters are crucial, taking into account the unique characteristics of the data and the specific problem being addressed. Trial and error or robust parameter tuning techniques can aid in finding the optimal values.
3. **Model selection:** While STL decomposition facilitates the isolation of components in a time series, it is essential to remember that the forecasting models applied to each component play a critical role in achieving accurate predictions. The choice of appropriate forecasting models for the trend, seasonal, and residual components must be based on careful consideration of the data properties, statistical techniques, and the specific goals of the analysis.

# Unit 5

# Unit – V: Data Visualization

Importance of Data Visualization, Data Visualization techniques, Box plots, histograms, charts, tree maps, Pixel- Oriented Visualization Techniques, Geometric Projection Visualization Techniques, Icon-Based Visualization Techniques, Hierarchical Visualization Techniques, Visualizing Complex Data and Relations.

# Introduction

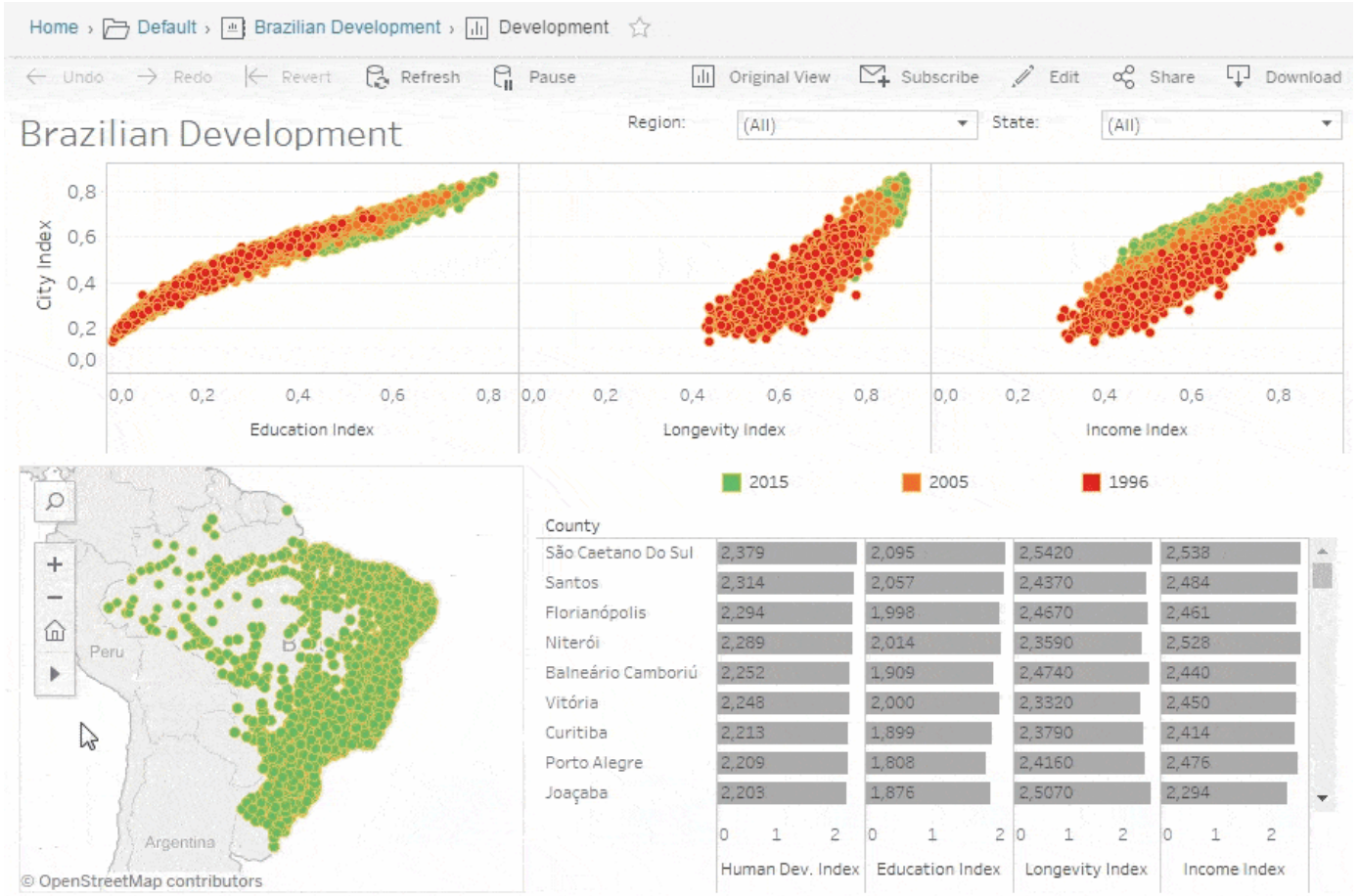
- In our increasingly data-driven world, it’s more important than ever to have accessible ways to view and understand data. After all, the demand for data skills in employees is steadily increasing each year. Employees and business owners at every level need to have an understanding of data and its impact.
- That’s where data visualization comes in handy. To make data more accessible and understandable, data visualization in the form of dashboards is the go-to tool for many businesses to analyze and share information.

# What is data visualization?

- 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. Additionally, it provides an excellent way for employees or business owners to present data to non-technical audiences without confusion.

- In the world of Big Data, data visualization tools and technologies are essential to analyze massive amounts of information and make data-driven decisions.

# What is data visualization?



# What is data visualization?

- Data visualization is the art and practice of gathering, analyzing, and graphically representing empirical information.
- They are sometimes called information graphics, or even just charts and graphs.
- The goal of visualizing data is to tell the story in the data.
- Telling the story is predicated on understanding the data at a very deep level, and gathering insight from comparisons of data points in the numbers

# What are the advantages and disadvantages of data visualization?

- Something as simple as presenting data in graphic format may have no downsides. But sometimes data can be misrepresented or misinterpreted when placed in the wrong data visualization style. When creating a data visualization, it’s best to keep both the advantages and disadvantages in mind.

## Advantages

Our eyes are drawn to colors and patterns. We can quickly identify red from blue, and squares from circles. Our culture is visual, including everything from art and advertisements to TV and movies. Data visualization is another form of visual art that grabs 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 data spreadsheet and couldn't see a trend, you know how much more effective a visualization can be.

Some other advantages of data visualization include:

- Easily sharing information.
- Interactively explore opportunities.
- Visualize patterns and relationships.

## Disadvantages

- While there are many advantages, some disadvantages may seem less obvious.

For example, viewing a visualization with many different data points makes it easy to make an inaccurate assumption. Sometimes the visualization is just designed wrong, so it's biased or confusing.

Some other disadvantages include:

- Biased or inaccurate information.
- Correlation doesn't always mean causation.
- Core messages can get lost in translation.

# Why data visualization is important

The importance of data visualization is simple:

it helps people see, interact with, and better understand data. Whether simple or complex, the right visualization can bring everyone on the same page, regardless of their level of expertise.

- **Communication**: Visualizations provide an intuitive and effective way to communicate complex information and insights to a diverse audience, including stakeholders, decision-makers, and the general public. Visual representations, such as charts, graphs, and maps, make it easier for people to understand trends, patterns, and relationships in the data at a glance.
- **Insight Discovery**: Visualizations help uncover hidden patterns, trends, and correlations in the data that may not be immediately apparent from raw numbers or text alone. By visualizing data, analysts and researchers can gain deeper insights into the underlying structure and dynamics of the data, leading to better-informed decisions and actions.

# Why data visualization is important

- **Decision Making**: Visualizations aid decision-making processes by providing clear and actionable insights derived from data analysis. Decision-makers can use visualizations to identify key trends, outliers, and areas of concern, enabling them to make informed choices and develop effective strategies.
- **Storytelling**: Visualizations are powerful tools for storytelling and narrative-building. By presenting data in a compelling and visually engaging manner, storytellers can convey complex concepts, convey messages, and evoke emotions more effectively, enhancing audience engagement and understanding.
- **Exploration and Explanation**: Visualizations facilitate data exploration and exploration by allowing users to interact with the data dynamically. Through interactive features such as zooming, filtering, and drilling down, users can explore different aspects of the data, uncover insights, and ask new questions, fostering a deeper understanding of the data and driving further analysis.

# Why data visualization is important

- **Detection of Errors and Anomalies**: Visualizations can help detect errors, outliers, and anomalies in the data more easily than manual inspection of raw data. By visualizing data distributions, trends, and relationships, analysts can quickly identify data quality issues and take corrective actions to ensure data integrity and reliability.
- **Presentation and Reporting**: Visualizations enhance the effectiveness of presentations, reports, and dashboards by making them more engaging, informative, and memorable. Well-designed visualizations can captivate audiences, convey key messages more effectively, and leave a lasting impression, thereby increasing the impact of the communication.

## Data Visualization techniques

- Data visualization techniques encompass various methods and tools for visually representing data to facilitate understanding, analysis, and communication. Here are some common data visualization techniques:

# General Types of Visualizations

- Chart: Information presented in a tabular, graphical form with data displayed along two axes. Can be in the form of a graph, diagram, or map.
- Table: A set of figures displayed in rows and columns.
- Graph: A diagram of points, lines, segments, curves, or areas that represent certain variables in comparison, usually along two axes at a right angle.
- Geospatial: A visualization that shows data in map form using different shapes and colors to show the relationship between pieces of data and specific locations.
- Infographic: A combination of visuals and words that represent data. It usually uses charts or diagrams.
- Dashboards: A collection of visualizations and data displayed in one place to help with analyzing and presenting data.

# More specific examples

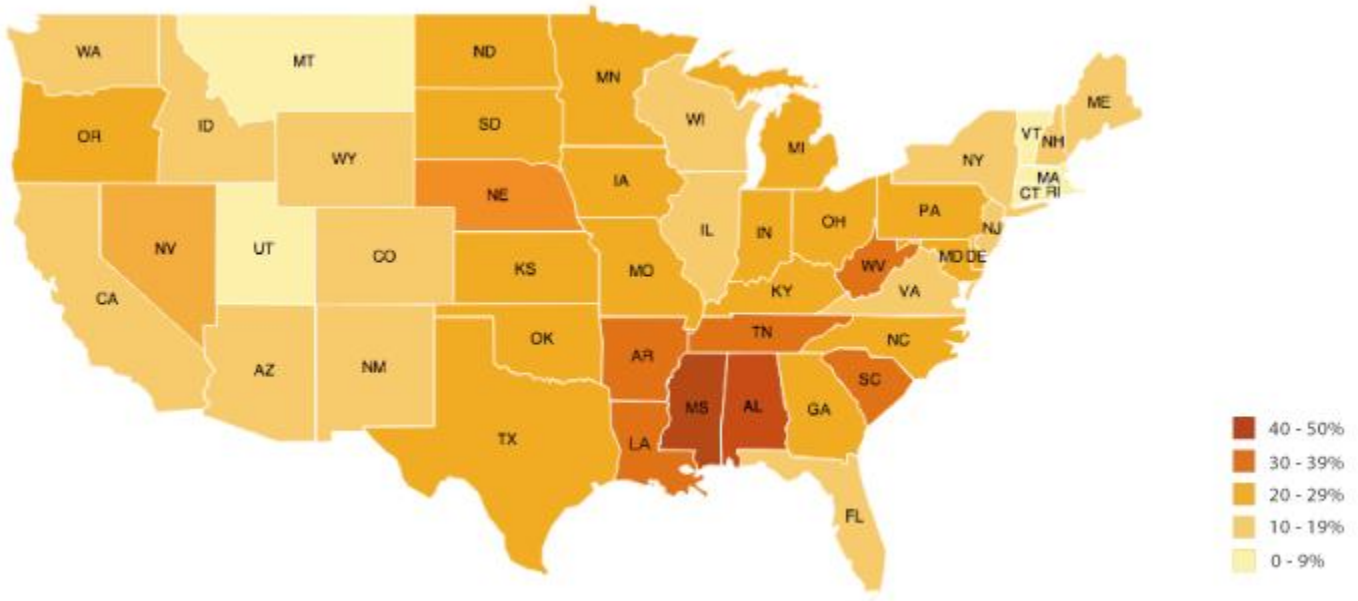
- Area Map: A form of geospatial visualization, area maps are used to show specific values set over a map of a country, state, county, or any other geographic location. Two common types of area maps are choropleths and isopleths.
- Bar Chart: Bar charts represent numerical values compared to each other. The length of the bar represents the value of each variable.
- Box-and-whisker Plots: These show a selection of ranges (the box) across a set measure (the bar).
- Bullet Graph: A bar marked against a background to show progress or performance against a goal, denoted by a line on the graph.
- Gantt Chart: Typically used in project management, Gantt charts are a bar chart depiction of timelines and tasks.
- Heat Map: A type of geospatial visualization in map form that displays specific data values as different colors (this doesn't need to be temperatures, but that is a common use).

# More specific examples

- Highlight Table: A form of table that uses color to categorize similar data, allowing the viewer to read it more easily and intuitively.
- Histogram: A type of bar chart that split a continuous measure into different bins to help analyze the distribution. Learn more.
- Pie Chart: A circular chart with triangular segments that shows data as a percentage of a whole. Learn more.
- Tree map: A type of chart that shows different, related values in the form of rectangles nested together. Learn more.

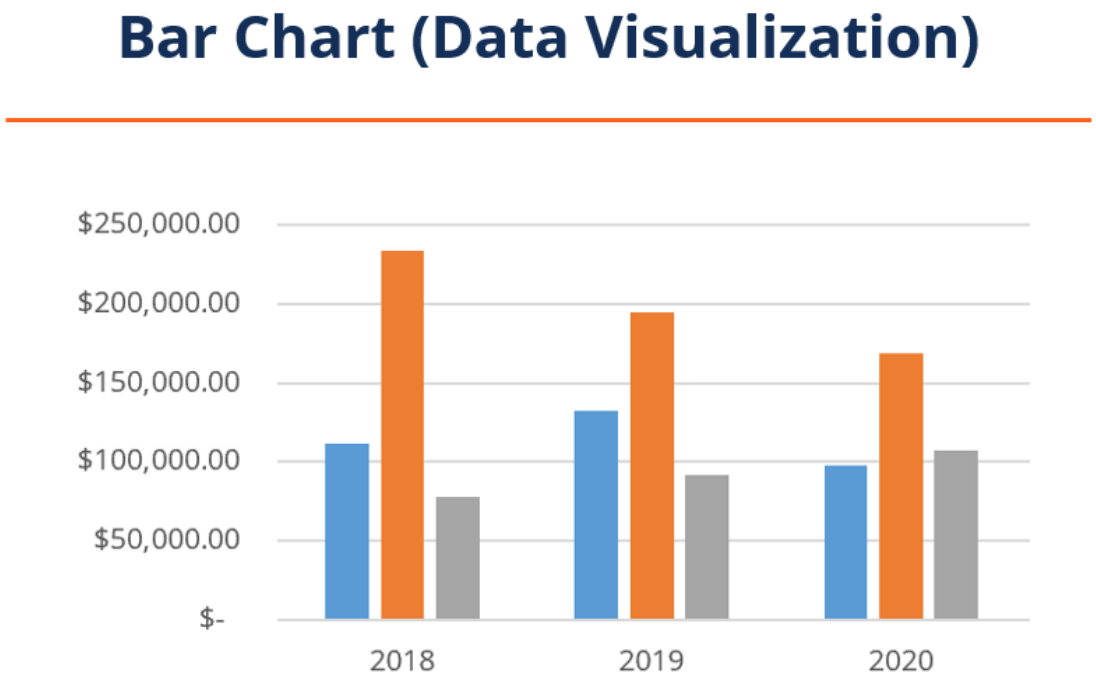
# Area Map

- **Area Map:** A form of geospatial visualization, area maps are used to show specific values set over a map of a country, state, or any other geographic location. Two common types of area maps are **choropleths** and **isopleths**.



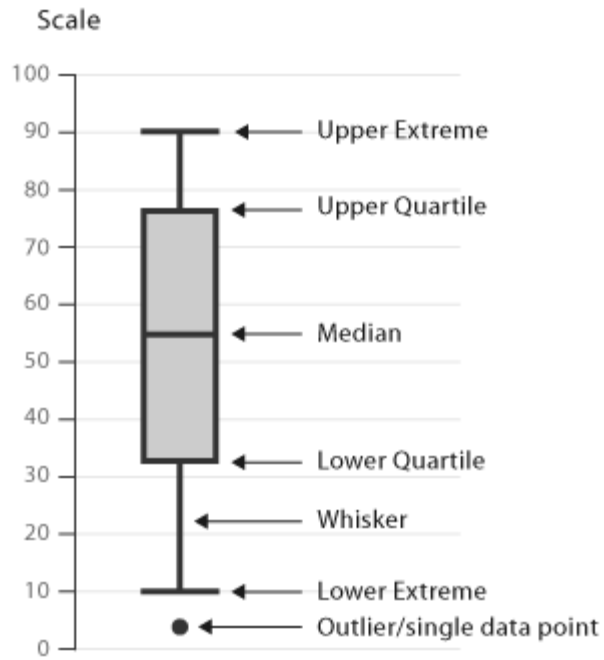
# Bar Chart

- Bar charts represent numerical values compared to each other. The length of the bar represents the value of each variable



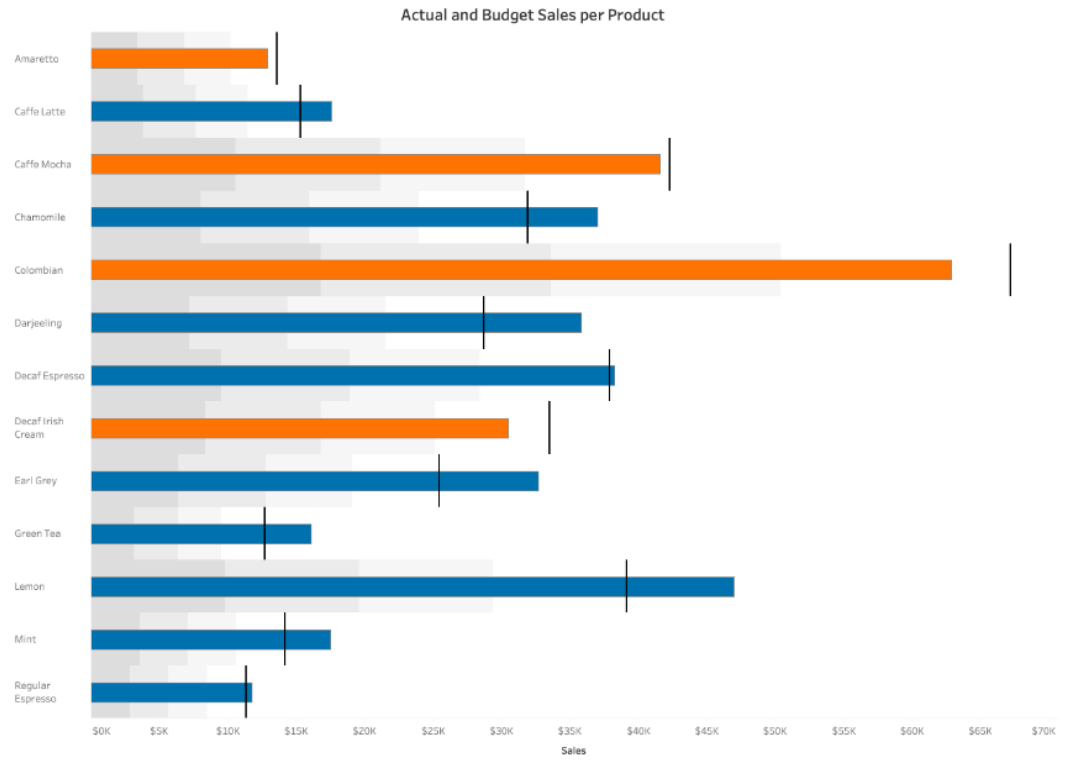
# Box-and-whisker Plots

- These show a selection of ranges (the box) across a set measure (the bar)



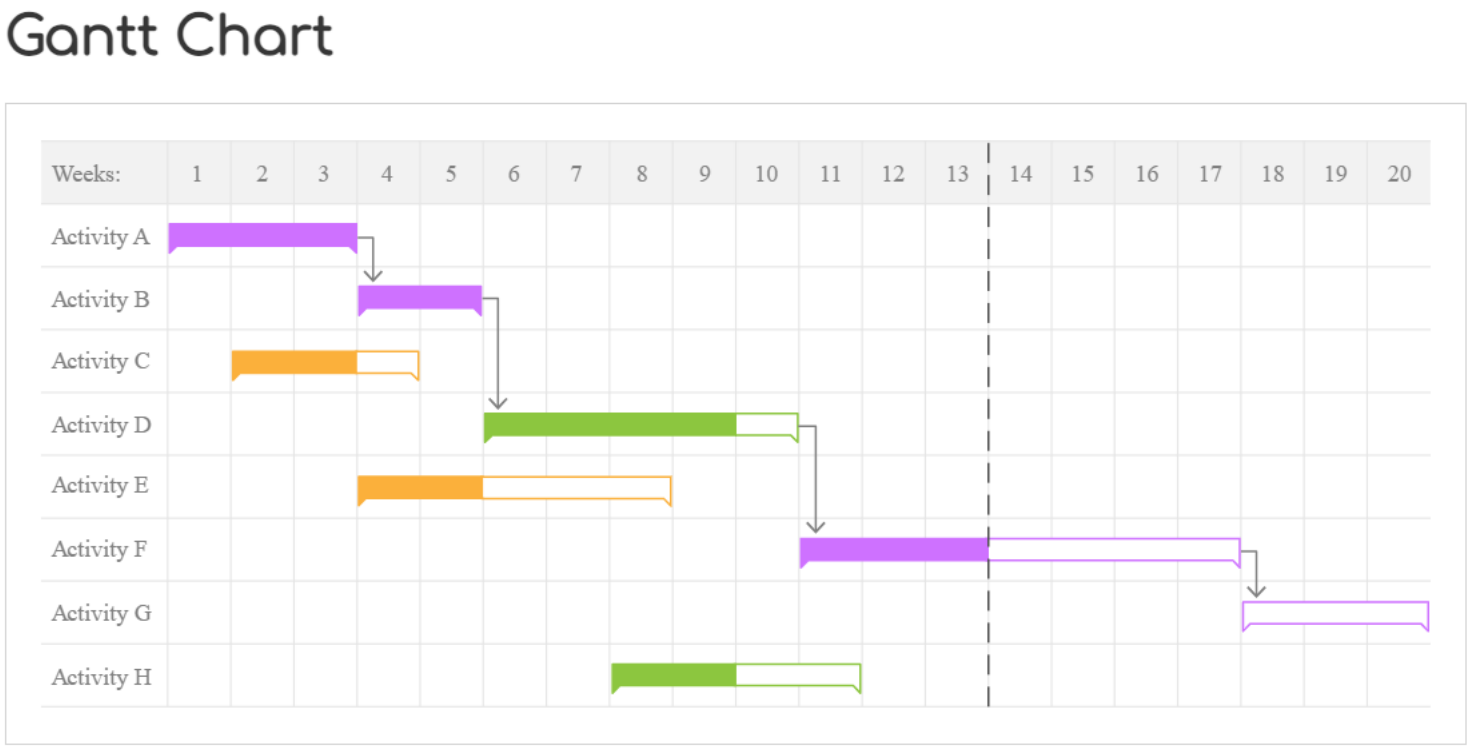
# Bullet Graph:

- A bar marked against a background to show progress or performance against a goal, denoted by a line on the graph.



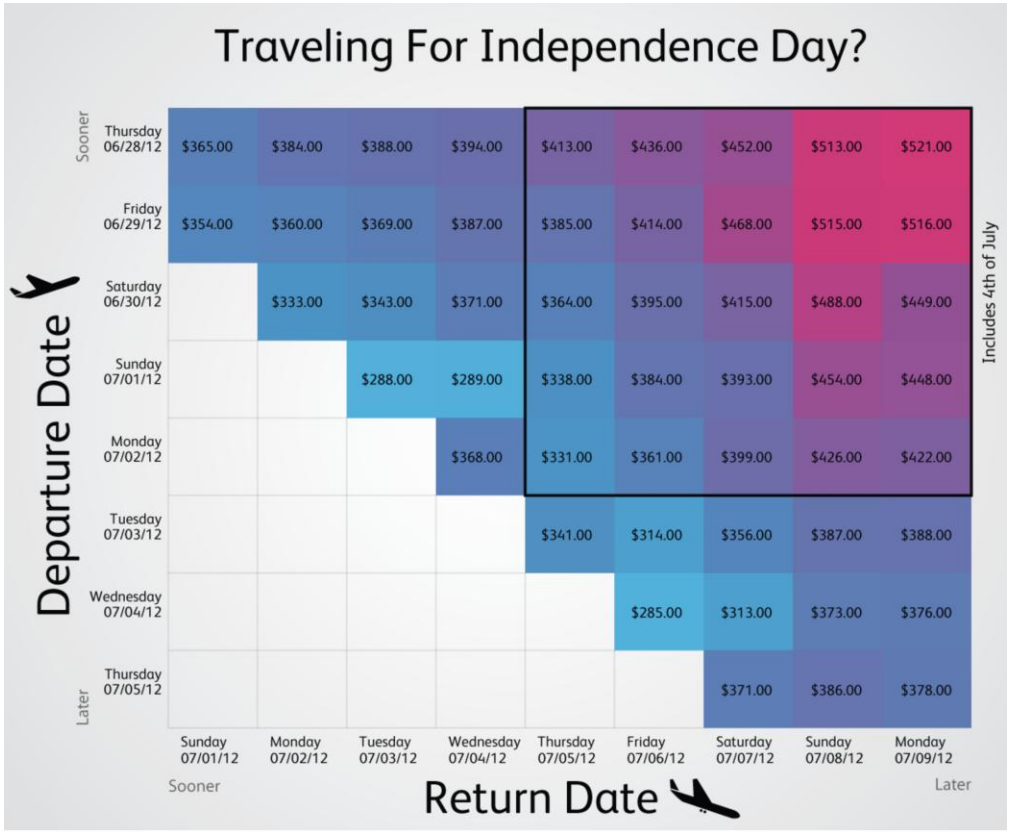
# Gantt Chart

- Typically used in project management, Gantt charts are a bar chart depiction of timelines and tasks.



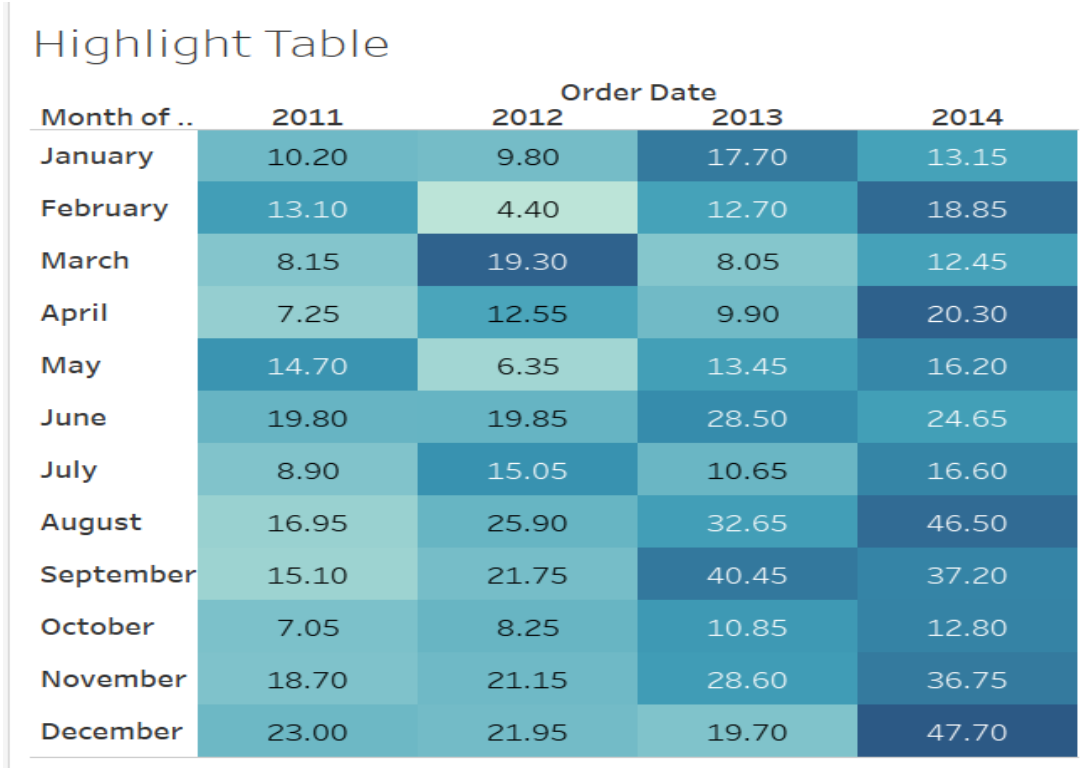
# Heat Map:

- A type of geospatial visualization in map form which displays specific data values as different colors (this doesn't need to be temperatures, but that is a common use).



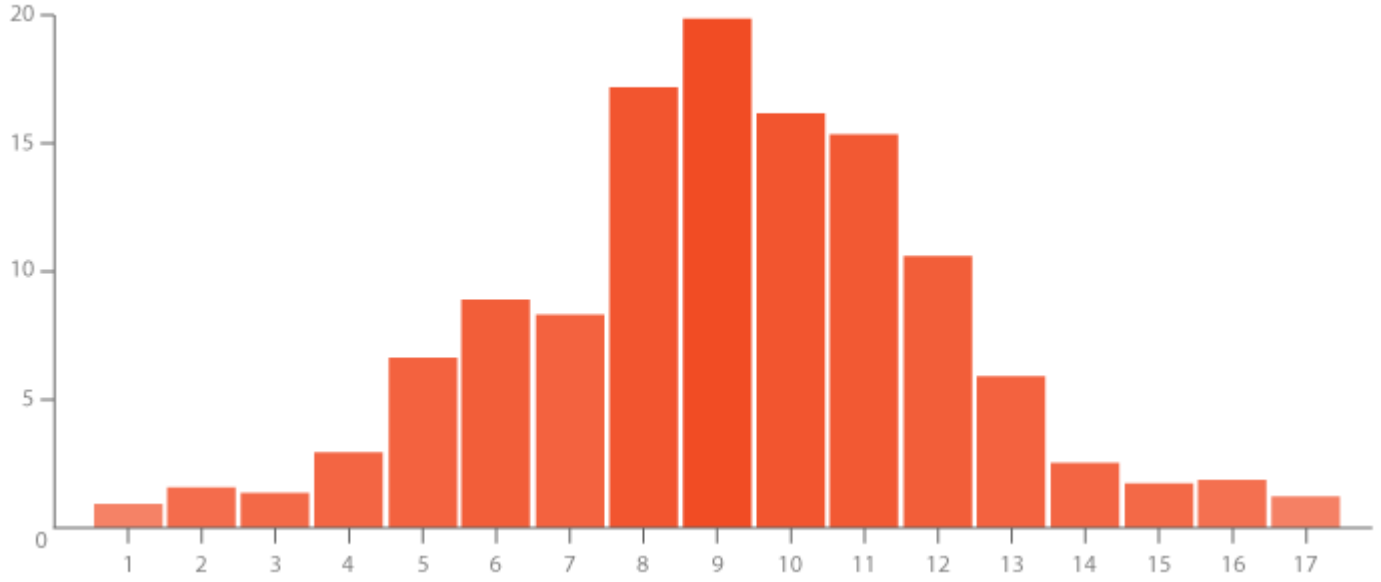
# Highlight Table

- A form of table that uses color to categorize similar data, allowing the viewer to read it more easily and intuitively



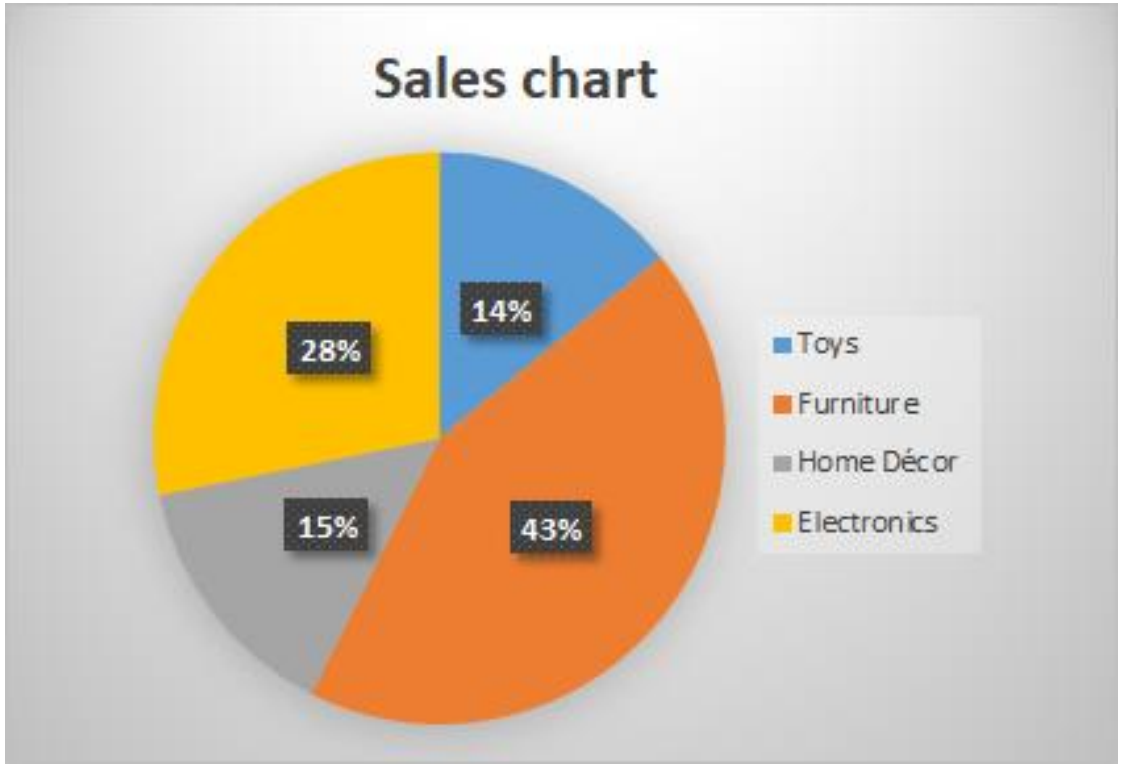
# Histogram

- A type of bar chart that split a continuous measure into different bins to help analyze the distribution



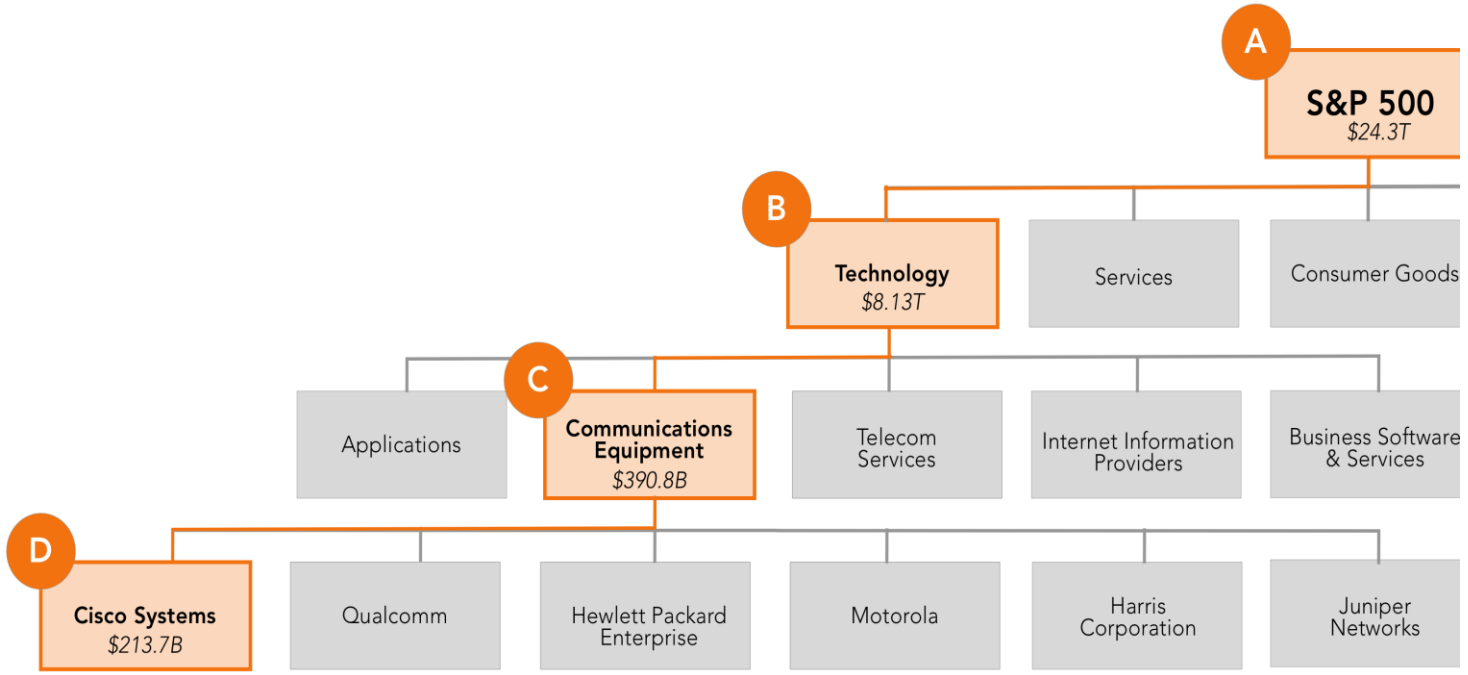
# Pie Chart:

- A circular chart with triangular segments that shows data as a percentage of a whole.



# Tree Map:

- A type of chart that shows different, related values in the form of rectangles nested together.



# Data Visualization Tools

- There are numerous data visualization tools available, ranging from simple, user-friendly applications to more advanced and customizable platforms. Here's a list of popular data visualization tools:
- Tableau: Tableau is a powerful and widely used data visualization tool that offers a range of features for creating interactive dashboards, charts, and graphs. It supports various data sources and provides intuitive drag-and-drop functionality.
- Microsoft Power BI: Power BI is a business analytics tool by Microsoft that enables users to visualize and share insights from their data. It offers interactive dashboards, reports, and data exploration capabilities.
- Google Data Studio: Google Data Studio is a free tool that allows users to create customizable reports and dashboards using data from Google Analytics, Google Ads, and other sources. It offers a user-friendly interface and integration with other Google products.
- QlikView/Qlik Sense: QlikView and Qlik Sense are business intelligence and data visualization platforms that offer powerful data discovery, analysis, and visualization capabilities. They support interactive dashboards, self-service analytics, and data storytelling.

# Data Visualization Tools

- D3.js: D3.js is a JavaScript library for creating dynamic and interactive data visualizations in web browsers. It provides a low-level approach to visualization, allowing developers to create custom visualizations using HTML, SVG, and CSS.
- Plotly: Plotly is a Python graphing library that offers interactive charts, dashboards, and data visualization tools. It supports a wide range of chart types and can be used in conjunction with other Python libraries like Pandas and Matplotlib.
- Matplotlib: Matplotlib is a Python library for creating static, interactive, and animated visualizations. It provides a wide range of plotting functions and customization options for creating publication-quality figures.

# Data Visualization Tools

- Seaborn: Seaborn is a Python data visualization library built on top of Matplotlib that provides a higher-level interface for creating statistical graphics. It offers built-in themes, color palettes, and functions for visualizing complex data structures.
- High charts: High charts is a JavaScript charting library that offers a wide range of interactive and customizable charts, including line charts, bar charts, pie charts, and more. It is commonly used for creating dynamic data visualizations on the web.
- ggplot2: ggplot2 is an R package for creating elegant and sophisticated data visualizations based on the grammar of graphics principles. It offers a flexible and consistent syntax for creating a wide range of statistical graphics.

# Box plots

- Box plots, also known as box-and-whisker plots, are a standardized way of displaying the distribution of data based on a five-number summary: minimum, first quartile (Q1), median, third quartile (Q3), and maximum.
- It is a graphical representation of the distribution of a dataset. It is particularly useful for understanding the central tendency, spread, and skewness of the data along with identifying outliers.

## Key Components of a Box Plot:

1. **Median (Q2)**: The middle value of the dataset, which divides the data into two equal halves.
2. **Quartiles (Q1, Q3)**: The quartiles divide the dataset into four equal parts. Q1 represents the value below which 25% of the data falls, and Q3 represents the value below which 75% of the data falls.
3. **Interquartile Range (IQR)**: The range between the first and third quartiles (IQR = Q3 - Q1). It gives a measure of the spread of the middle 50% of the dataset.
4. **Whiskers**: The lines extending from the box indicate the range of the data. By default, they typically extend to 1.5 times the IQR from the quartiles. Any data points beyond the whiskers are considered outliers.
5. **Outliers**: Individual data points that fall outside the whiskers and are typically considered as anomalies or extreme values.

## Advantages of Box Plots:

- They provide a visual summary of the dataset's central tendency, spread, and outliers.
- They are useful for comparing distributions between different groups or datasets.
- They are resistant to the effects of outliers, providing a robust representation of the data.

# Example

```
#bar plot
import matplotlib.pyplot as plt
import numpy as np

# Sample data
data = np.random.normal(loc=0, scale=1, size=100)

# Creating the box plot
plt.boxplot(data)
plt.title('Box Plot')
plt.xlabel('Data')
plt.ylabel('Values')
plt.show()
```

# Histograms

- Histograms represent the distribution of numerical data by dividing the data into bins and displaying the frequency of each bin as a bar.

- A histogram is a graphical representation of the distribution of numerical data. It consists of a series of contiguous rectangles, or bins, where the area of each rectangle represents the frequency (or proportion) of data points falling within the corresponding interval.

## Key Components of a Histogram:

1. **Bins**: The intervals into which the data is divided. Each bin represents a range of values. and the height of the bin corresponds to the frequency of data points falling within that range.
2. **Frequency**: The number of data points falling within each bin. It gives an indication of how often certain values occur within the dataset.
3. **X-axis (Horizontal)**: Represents the range of values or categories being measured.
4. **Y-axis (Vertical)**: Represents the frequency or proportion of data points falling within each bin.

## Advantages of Histograms:

- They provide a visual summary of the distribution of the data, showing its central tendency, spread, and skewness.
- They are useful for identifying patterns, trends, and outliers within the dataset.
- They are flexible and can be customized by adjusting the number of bins. bin width. and other parameters to suit the characteristics of the data.

# Example

```
#histogram
import matplotlib.pyplot as plt
import numpy as np

# Sample data
data = np.random.normal(loc=0, scale=1, size=1000)

# Creating the histogram
plt.hist(data, bins=30, edgecolor='black')
plt.title('Histogram')
plt.xlabel('Data')
plt.ylabel('Frequency')
plt.show()
```

# Charts (line plot)

- A line plot is a basic type of chart that displays data points as markers connected by straight line segments. It's often used to visualize data trends over time.
- A line plot, also known as a line chart or line graph, is a type of chart that displays data points connected by straight line segments. It is particularly useful for showing trends and changes in data over time or across different categories.

## Key Components of a Line Plot:

1. **Data Points**: Individual data values are plotted on the chart.
2. **Lines**: Straight lines connecting consecutive data points. These lines help visualize the trend or pattern in the data.
3. **X-axis (Horizontal)**: Represents the independent variable, such as time, categories, or any other continuous variable being measured.
4. **Y-axis (Vertical)**: Represents the dependent variable, or the values being measured or observed.

## Advantages of Line Plots:

- They provide a clear visual representation of trends and patterns in the data, making it easy to identify relationships and correlations.
- They are effective for displaying time series data, showing how variables change over time.
- They are versatile and can be used to compare multiple datasets or variables on the same chart.

# Example

```
#line plot
import matplotlib.pyplot as plt
import numpy as np
# Sample data
x = np.linspace(0, 10, 100)
y = np.sin(x)
# Creating the line plot
plt.plot(x, y)
plt.title('Line Plot')
plt.xlabel('X')
plt.ylabel('Y')
plt.show()
```

# Tree maps

- Tree maps display hierarchical data as a set of nested rectangles, where each rectangle's size represents a certain attribute of the data.
- A tree map is a type of hierarchical visualization that displays hierarchical data as a set of nested rectangles. Each rectangle represents a node in the hierarchy, with the area of the rectangle proportional to a specific attribute of the data, such as size, value, or frequency.

## Key Components of a TreeMap:

1. **Nested Rectangles**: The main visual element of the tree map, where each rectangle represents a node in the hierarchical structure. The size of the rectangle corresponds to a certain attribute of the data.
2. **Hierarchy**: The hierarchical structure of the data, which is represented by the nesting of rectangles within each other. Typically, the parent-child relationships are indicated by the containment of smaller rectangles within larger ones.
3. **Color**: Optional visual encoding used to represent additional information, such as category or value, within the tree map. Color can be used to enhance the understanding of the data or highlight specific patterns or outliers.

## Advantages of TreeMaps:

- They provide an effective way to visualize hierarchical data structures, allowing users to explore relationships between different levels of the hierarchy.
- They offer a space-efficient representation of data, utilizing the area of rectangles to convey information about the data attributes.
- They are versatile and can be customized to represent various types of data and attributes, making them suitable for a wide range of applications.

# Example

```
import matplotlib.pyplot as plt
!pip install squarify
import squarify

# Sample data
sizes = [50, 30, 15, 5]
labels = ['A', 'B', 'C', 'D']

# Creating the tree map
squarify.plot(sizes=sizes, label=labels, alpha=0.7)
plt.axis('off')
plt.title('Tree Map')
plt.show()
```

# Data Visualization

- Why data visualization?
  - Gain insight into an information space by mapping data onto graphical primitives
  - Provide qualitative overview of large data sets
  - Search for patterns, trends, structure, irregularities, relationships among data
  - Help find interesting regions and suitable parameters for further quantitative analysis
  - Provide a visual proof of computer representations derived
- Categorization of visualization methods:
  - Pixel-oriented visualization techniques
  - Geometric projection visualization techniques
  - Icon-based visualization techniques
  - Hierarchical visualization techniques
  - Visualizing complex data and relations

# References

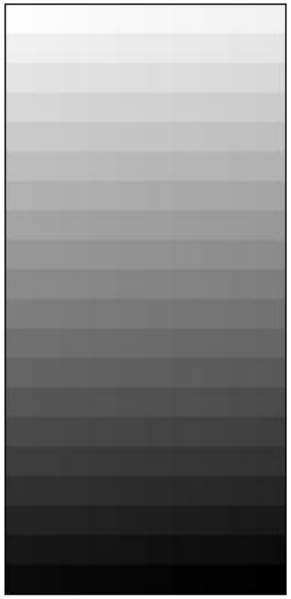

- https://datavizcatalogue.com/index.html

# Data Visualization

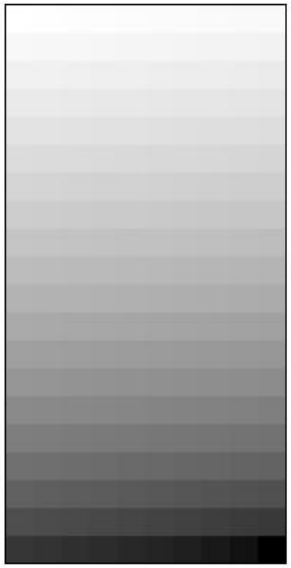
- Why data visualization?
  - Gain insight into an information space by mapping data onto graphical primitives
  - Provide a qualitative overview of large data sets
  - Search for patterns, trends, structure, irregularities, and relationships among data
  - Help find interesting regions and suitable parameters for further quantitative analysis
  - Provide a visual proof of computer representations derived
- **Categorization of visualization methods:**
  - Pixel-oriented visualization techniques
  - Geometric projection visualization techniques
  - Icon-based visualization techniques
  - Hierarchical visualization techniques
  - Visualizing complex data and relations

# Pixel-Oriented Visualization Techniques

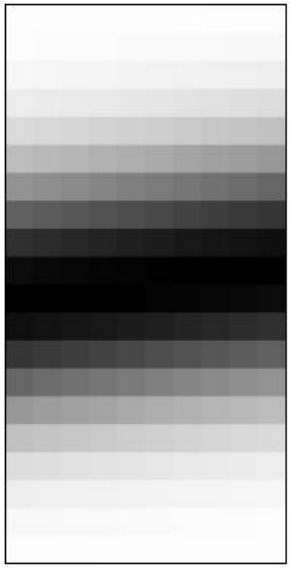
- For a data set of m dimensions, create m windows on the screen, one for each dimension
- The m dimension values of a record are mapped to m pixels at the corresponding positions in the windows
- The colors of the pixels reflect the corresponding values



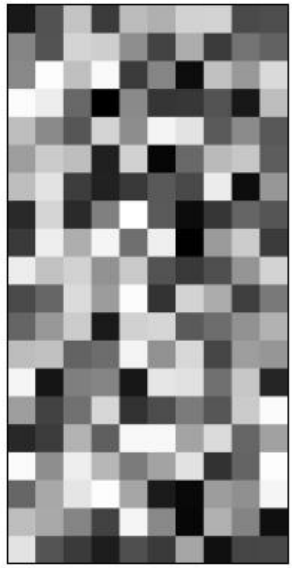
(a) Income



(b) Credit Limit



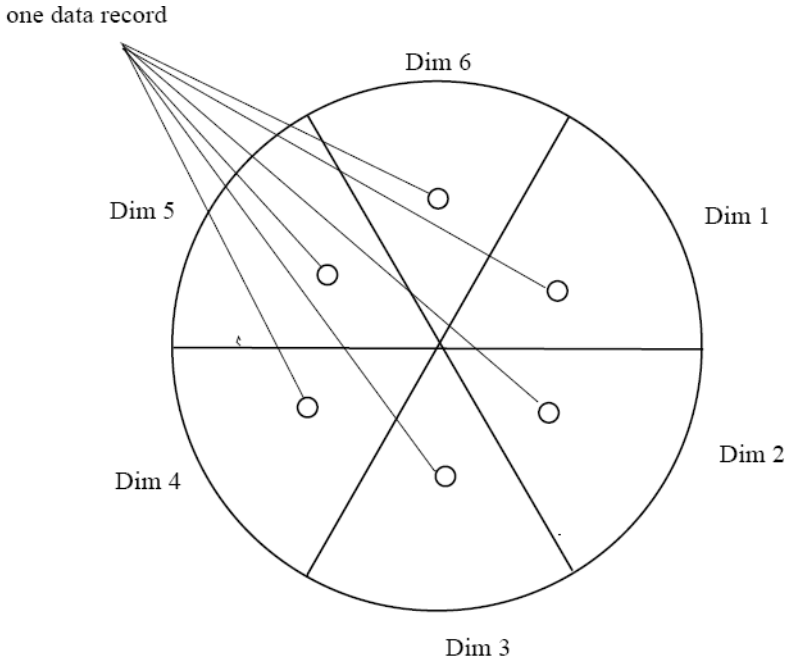
(c) transaction volume



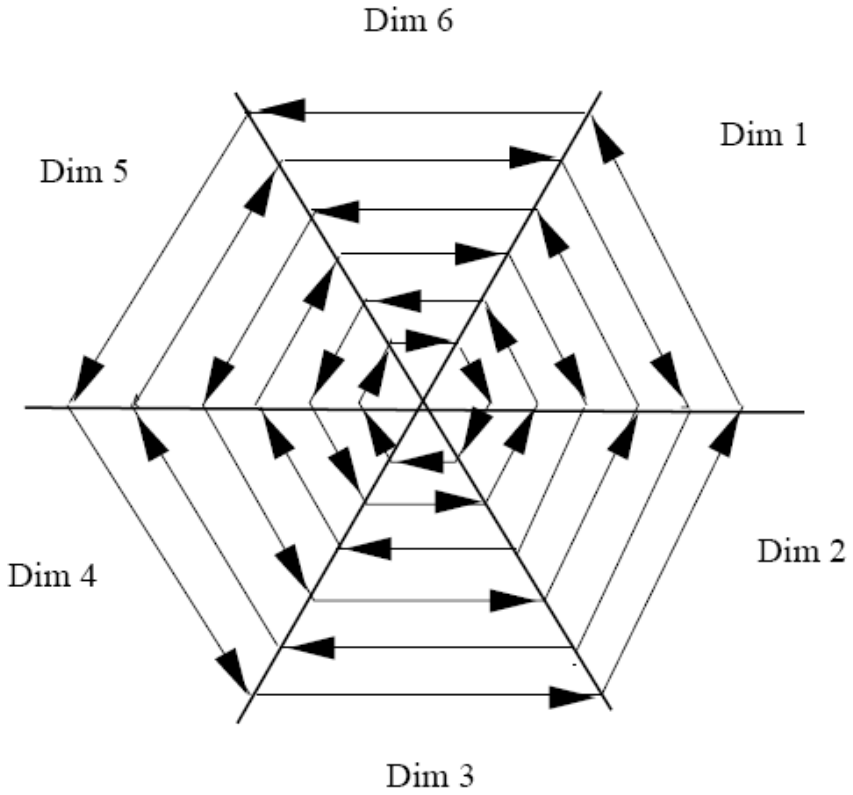
(d) age

# Laying Out Pixels in Circle Segments

- To save space and show the connections among multiple dimensions, space filling is often done in a circle segment




(a) Representing a data record in circle segment



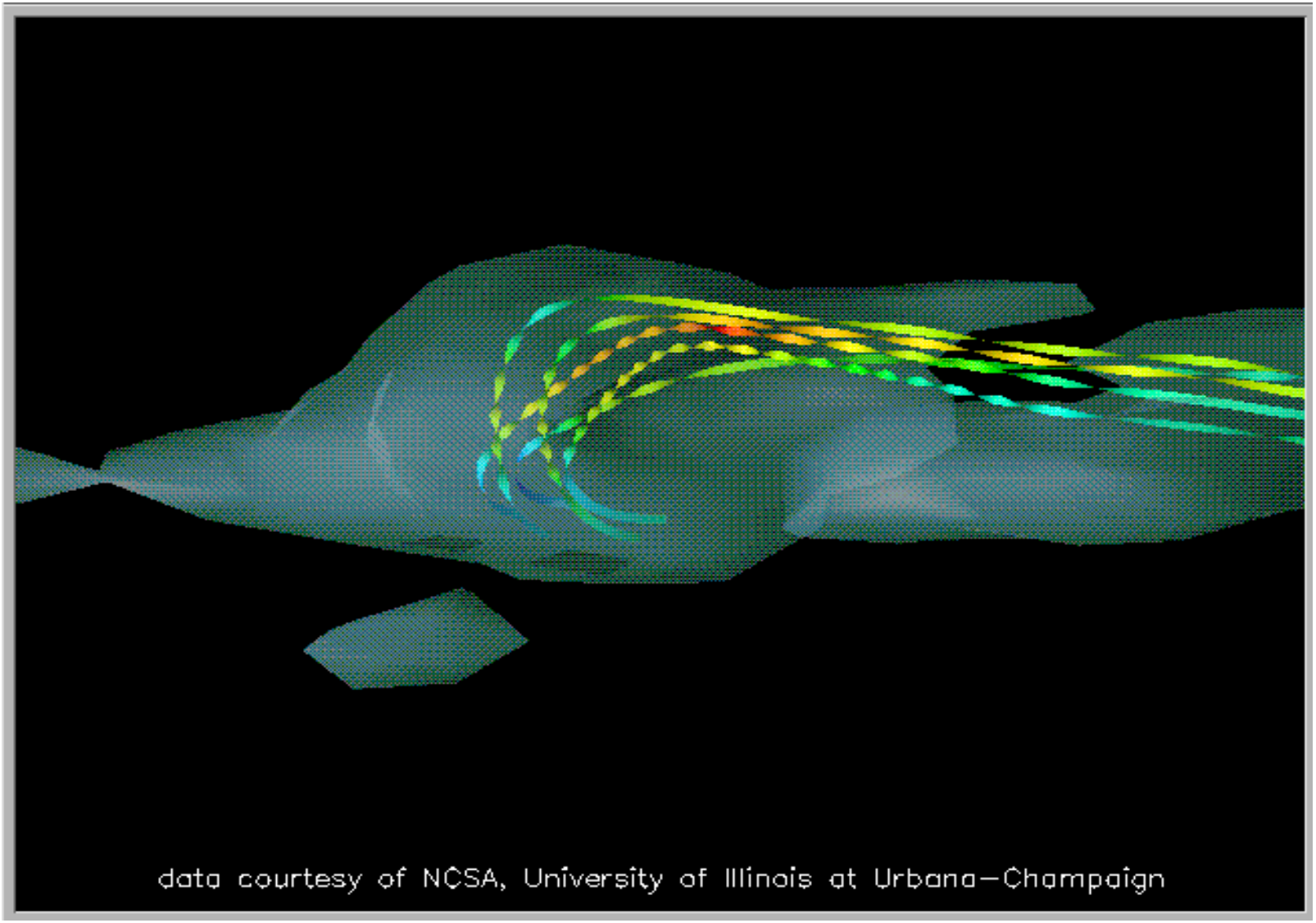

(b) Laying out pixels in circle segment

# Geometric Projection Visualization Techniques

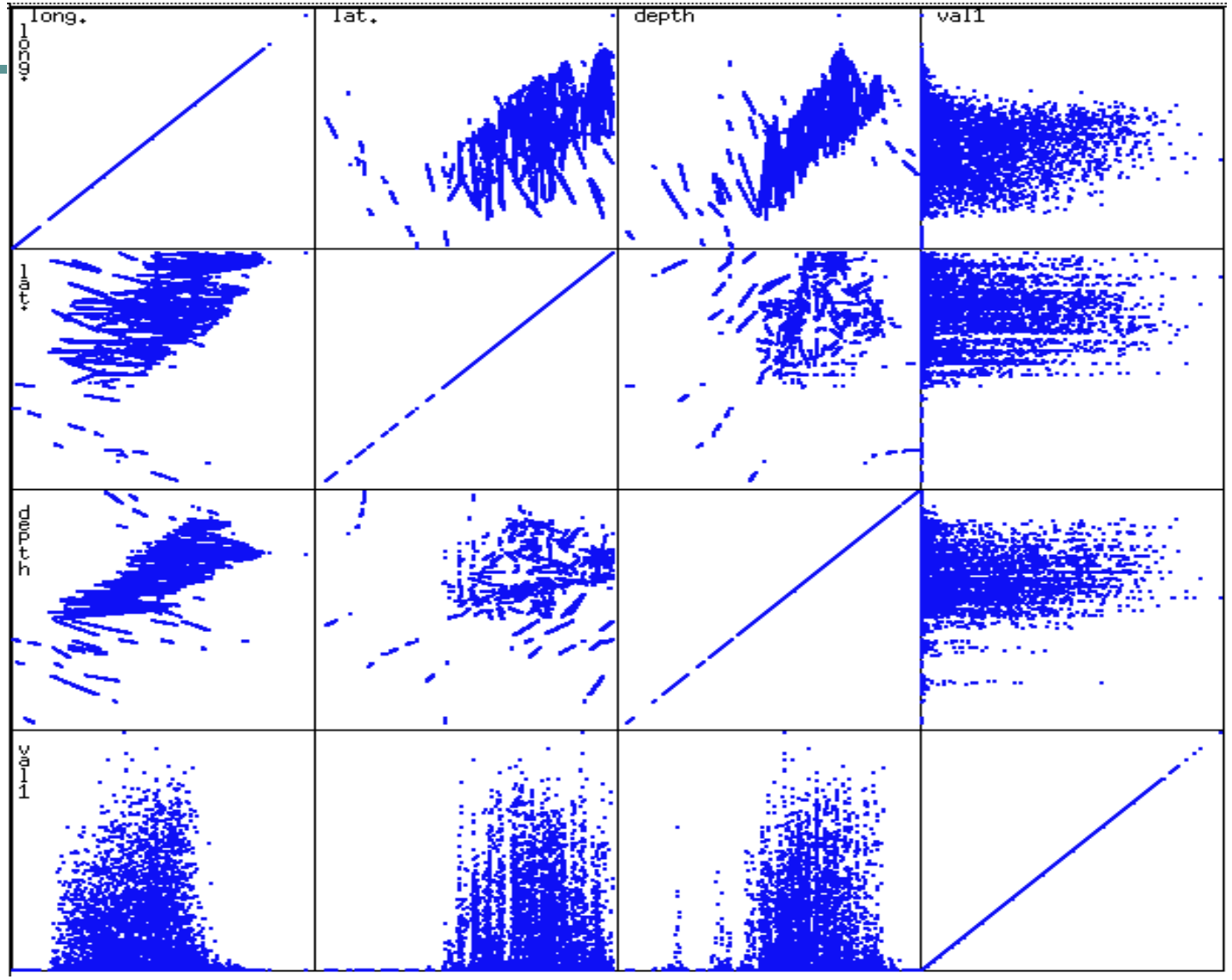
- Visualization of geometric transformations and projections of the data
- Methods
  - Direct visualization
  - Scatterplot and scatterplot matrices
  - Landscapes
  - Projection pursuit technique: Help users find meaningful projections of multidimensional data
  - Prosection views
  - Hyperslice
  - Parallel coordinates

# Direct Data Visualization

Ribbons with Twists Based on Vorticity



# Scatterplot Matrices

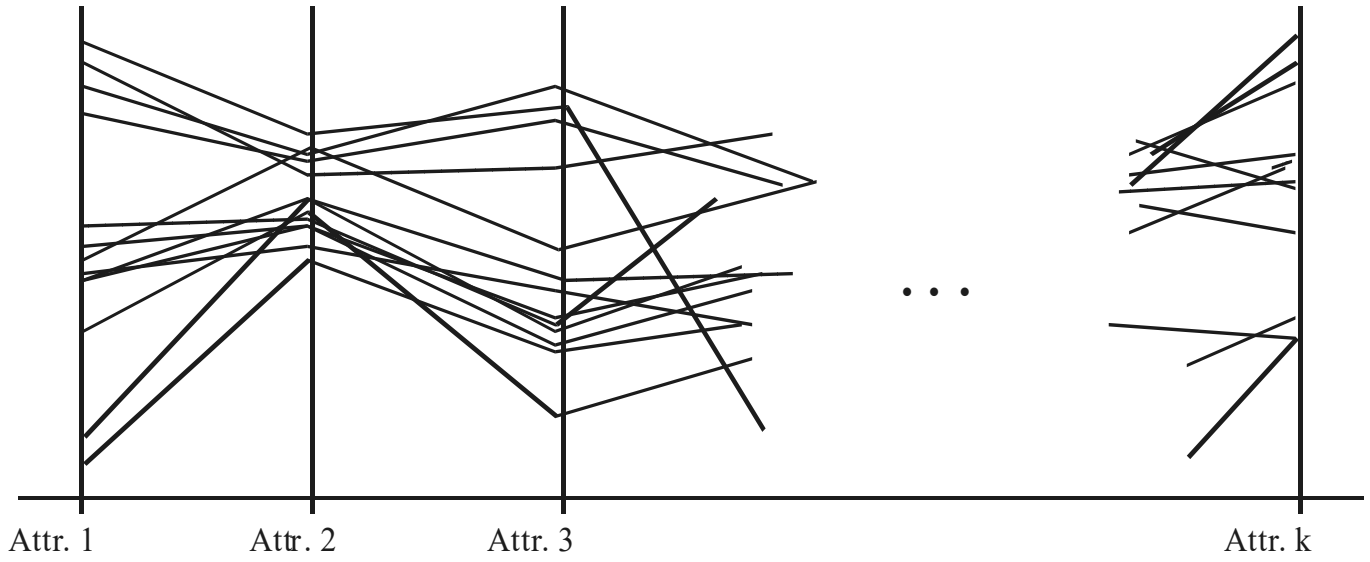




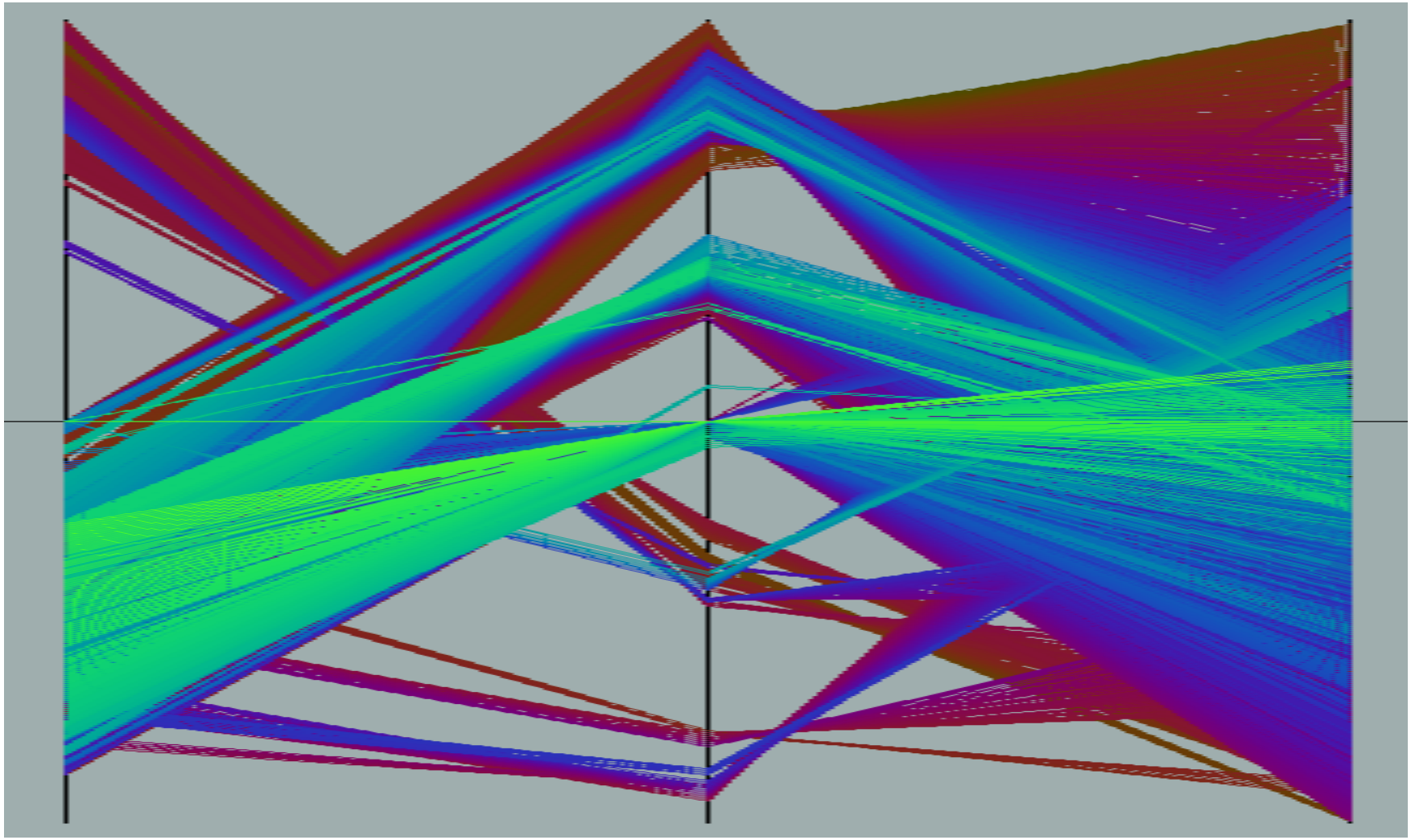

Matrix of scatterplots (x-y-diagrams) of the k-dim. data [total of (k2/2-k) scatterplots]

# Parallel Coordinates

- n equidistant axes which are parallel to one of the screen axes and correspond to the attributes
- The axes are scaled to the [minimum, maximum]: range of the corresponding attribute
- Every data item corresponds to a polygonal line which intersects each of the axes at the point which corresponds to the value for the attribute



# Parallel Coordinates of a Data Set

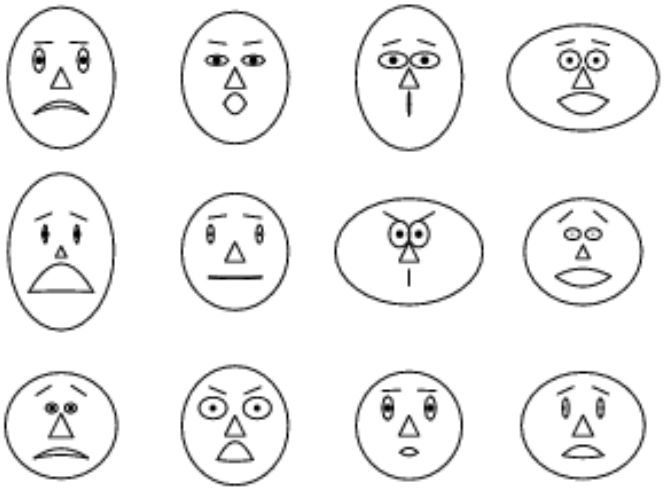


# Icon-Based Visualization Techniques

- Visualization of the data values as features of icons
- Typical visualization methods
  - Chernoff Faces
  - Stick Figures
- General techniques
  - Shape coding: Use shape to represent certain information encoding
  - Color icons: Use color icons to encode more information
  - Tile bars: Use small icons to represent the relevant feature vectors in document retrieval

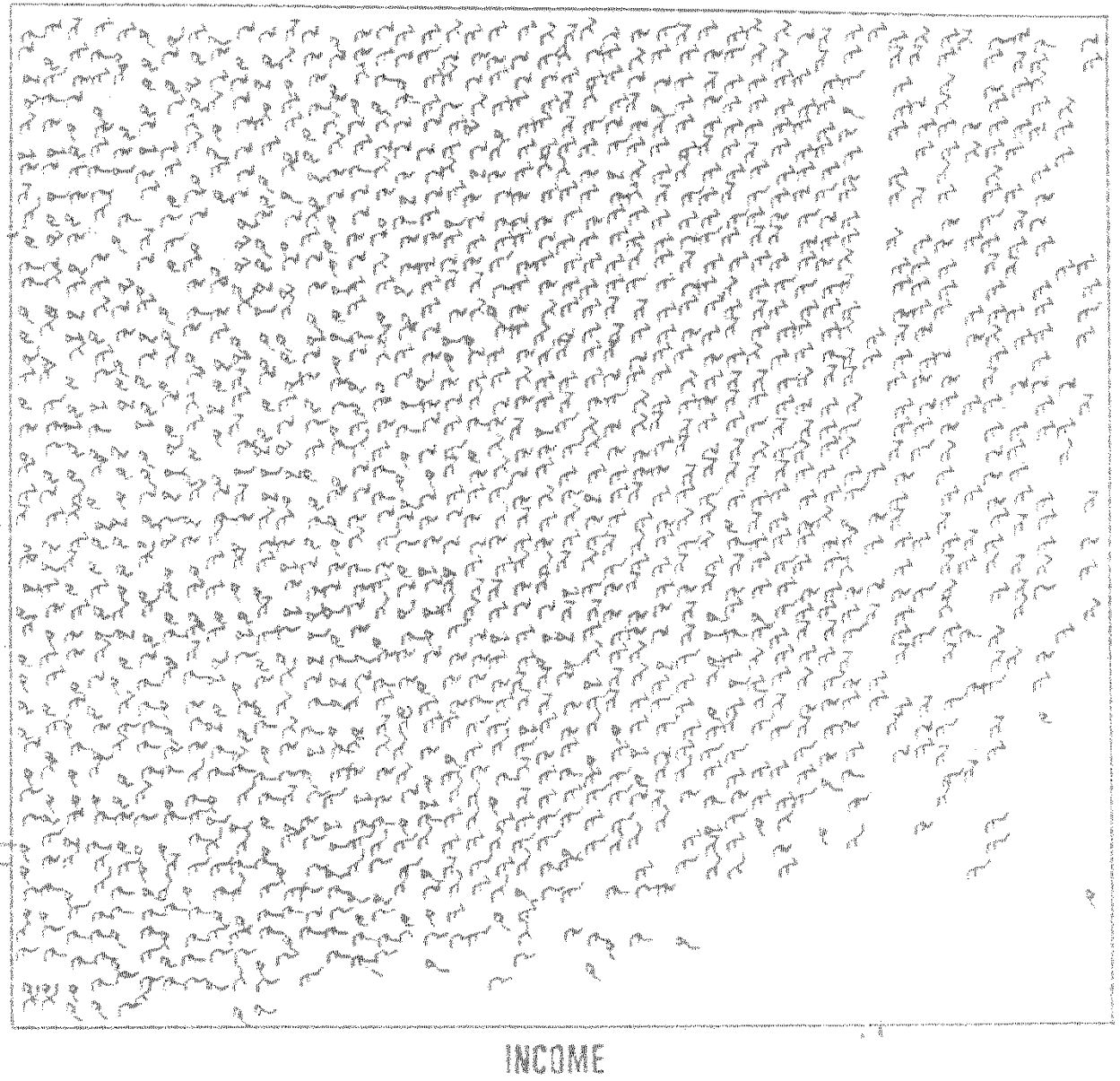
# Chernoff Faces

- A way to display variables on a two-dimensional surface, e.g., let x be eyebrow slant, y be eye size, z be nose length, etc.
- The figure shows faces produced using 10 characteristics--head eccentricity, eye size, eye spacing, eye eccentricity, pupil size, eyebrow slant, nose size, mouth shape, mouth size, and mouth opening): Each assigned one of 10 possible values, generated using *Mathematica* (S. Dickson)



- REFERENCE: Gonick, L. and Smith, W. *The Cartoon Guide to Statistics.* New York: Harper Perennial, p. 212, 1993
- Weisstein, Eric W. "Chernoff Face." From *MathWorld*--A Wolfram Web Resource. mathworld.wolfram.com/ChernoffFace.html

# Stick Figure




A census data figure showing age, income, gender, education, etc.

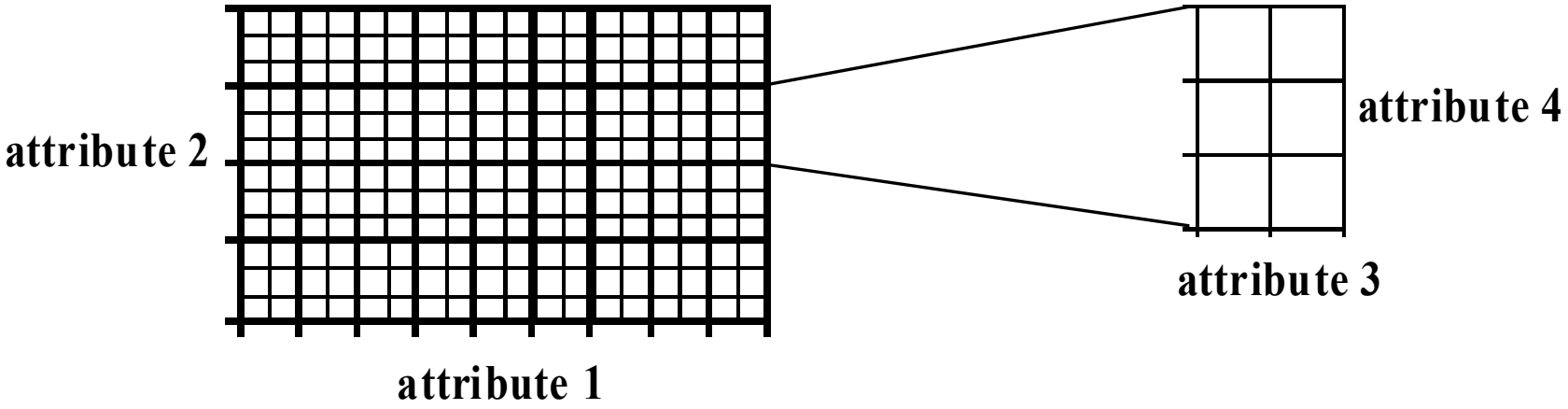
A 5-piece stick figure (1 body and 4 limbs w. different angle/length)

Two attributes mapped to axes, remaining attributes mapped to angle or length of limbs". Look at texture pattern

# Hierarchical Visualization Techniques

- Visualization of the data using a hierarchical partitioning into subspaces
- Methods
  - Dimensional Stacking
  - **Worlds-within-Worlds**
  - **Tree-Map**
  - Cone Trees
  - InfoCube

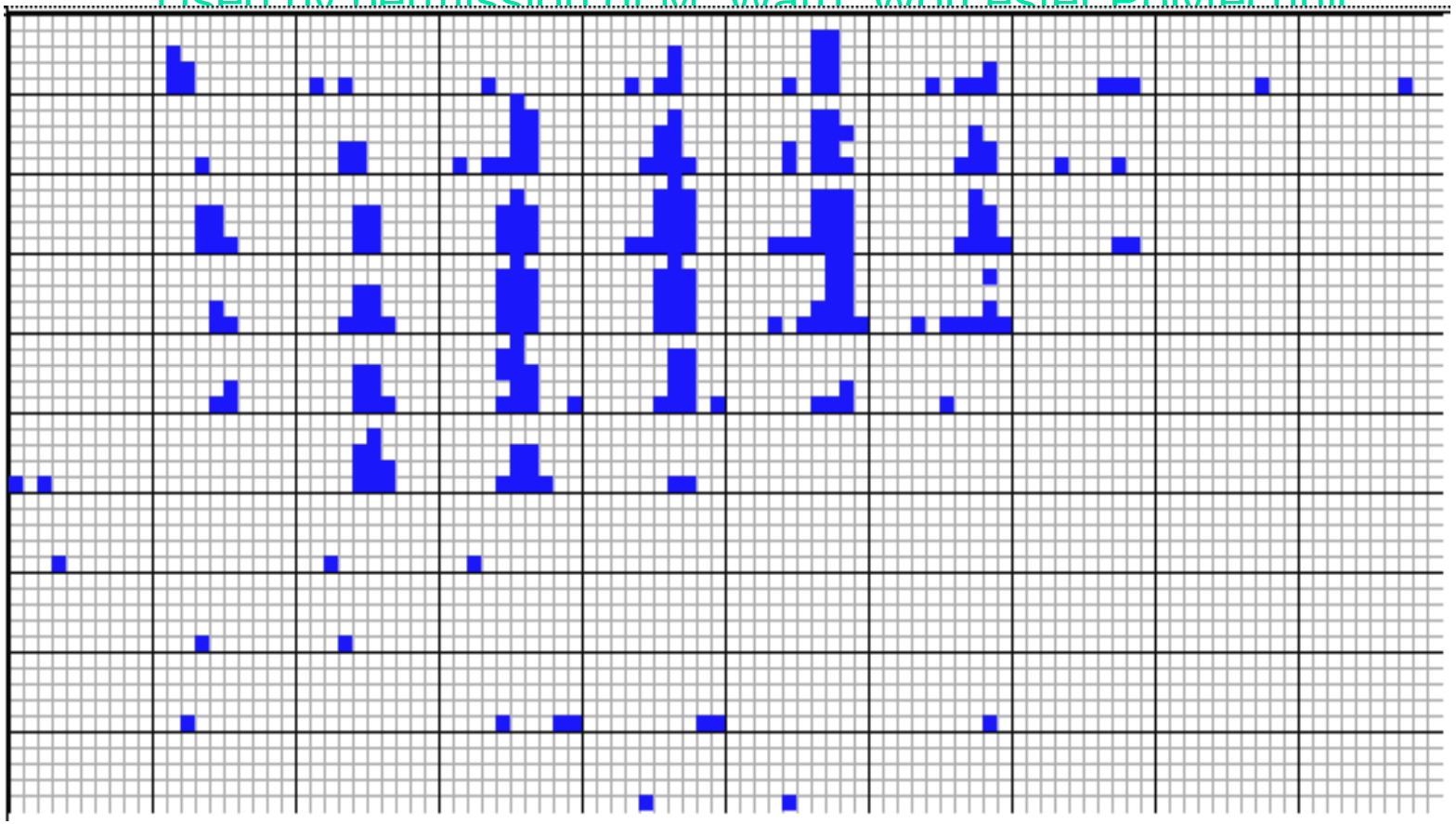
# Dimensional Stacking




- Partitioning of the n-dimensional attribute space in 2-D subspaces, which are ‘stacked’ into each other
- Partitioning of the attribute value ranges into classes. The important attributes should be used on the outer levels.
- Adequate for data with ordinal attributes of low cardinality
- But, difficult to display more than nine dimensions
- Important to map dimensions appropriately

# Dimensional Stacking

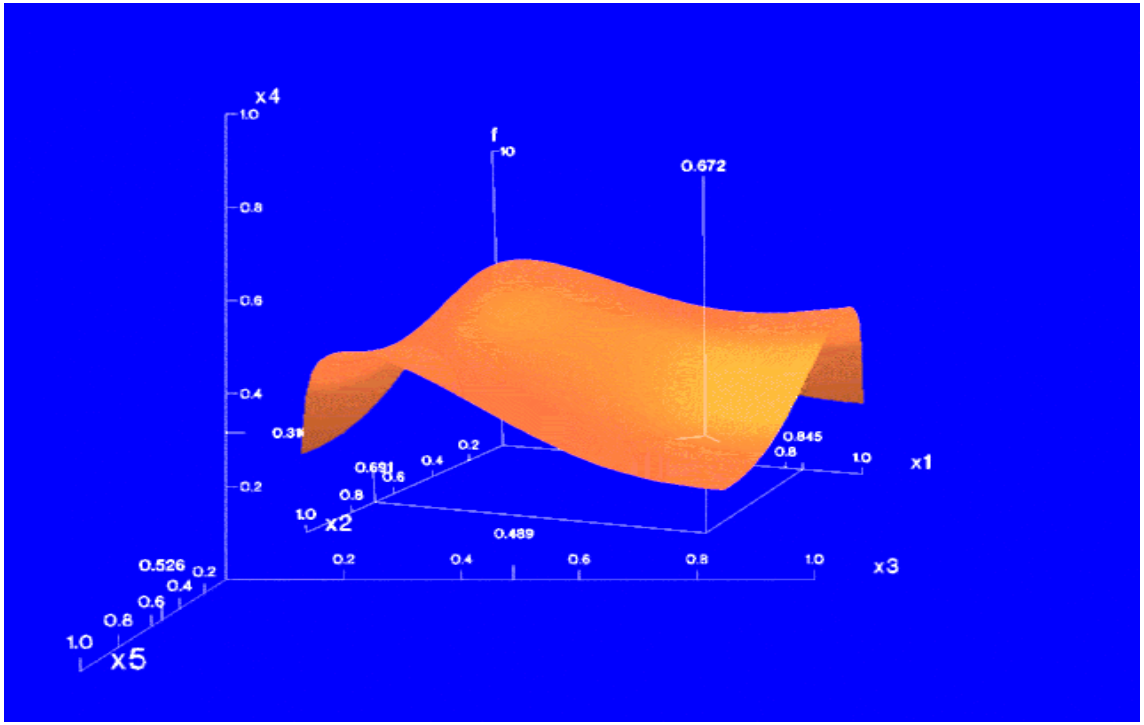
Used by permission of M. Ward, Worcester Polytechnic



Visualization of oil mining data with longitude and latitude mapped to the outer x-, y-axes and ore grade and depth mapped to the inner x-, y-axes

# Worlds-within-Worlds

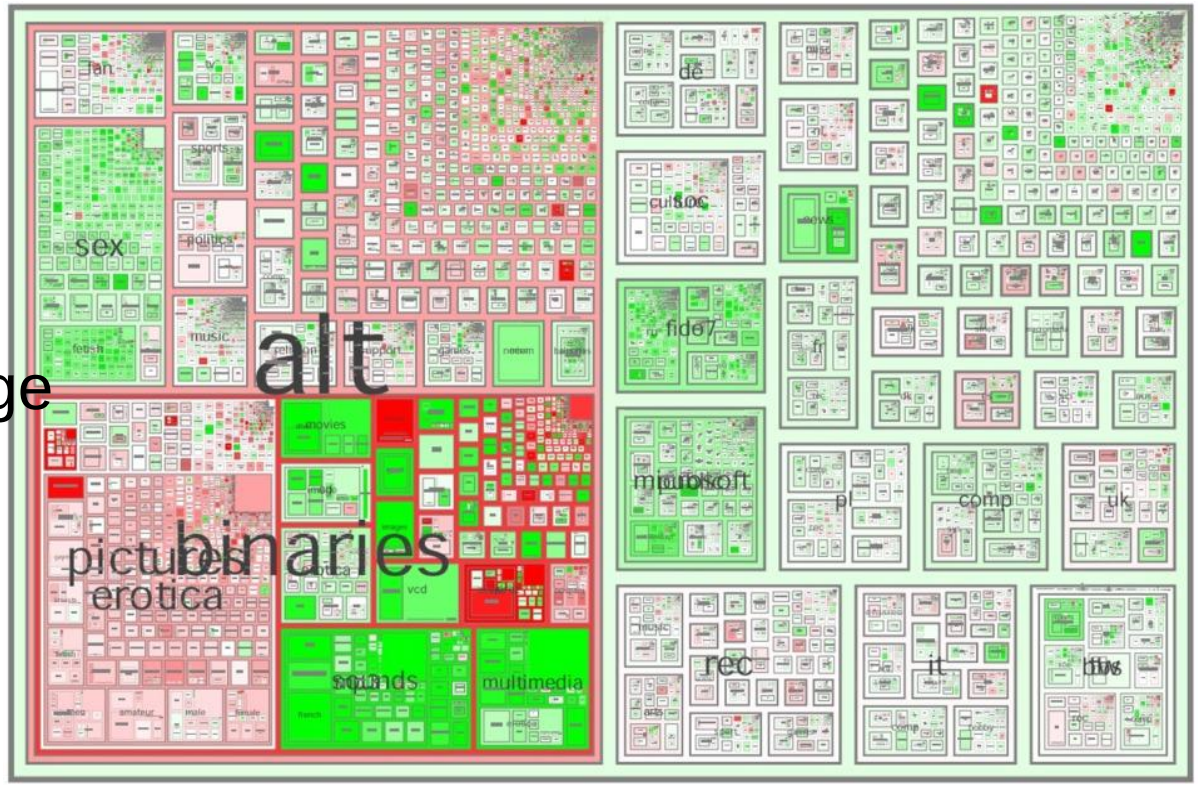
- Assign the function and two most important parameters to innermost world
- Fix all other parameters at constant values - draw other (1 or 2 or 3 dimensional worlds choosing these as the axes)
- Software that uses this paradigm
  - N–vision: Dynamic interaction through data glove and stereo displays, including rotation, scaling (inner) and translation (inner/outer)
  - Auto Visual: Static interaction by means of queries



# Tree-Map

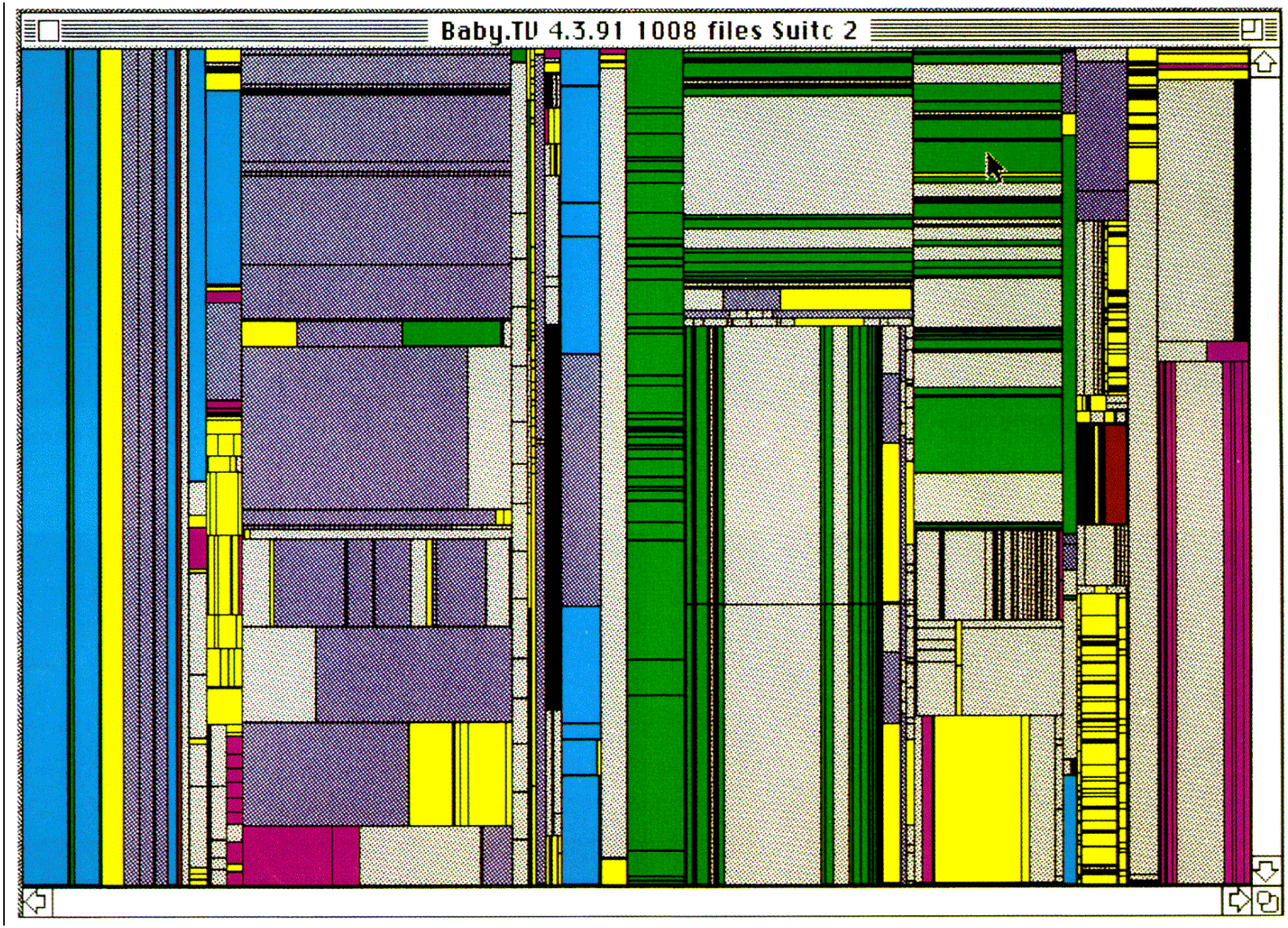
- Screen-filling method which uses a hierarchical partitioning of the screen into regions depending on the attribute values
- The x- and y-dimension of the screen are partitioned alternately according to the attribute values (classes)

MSR Netscan Image



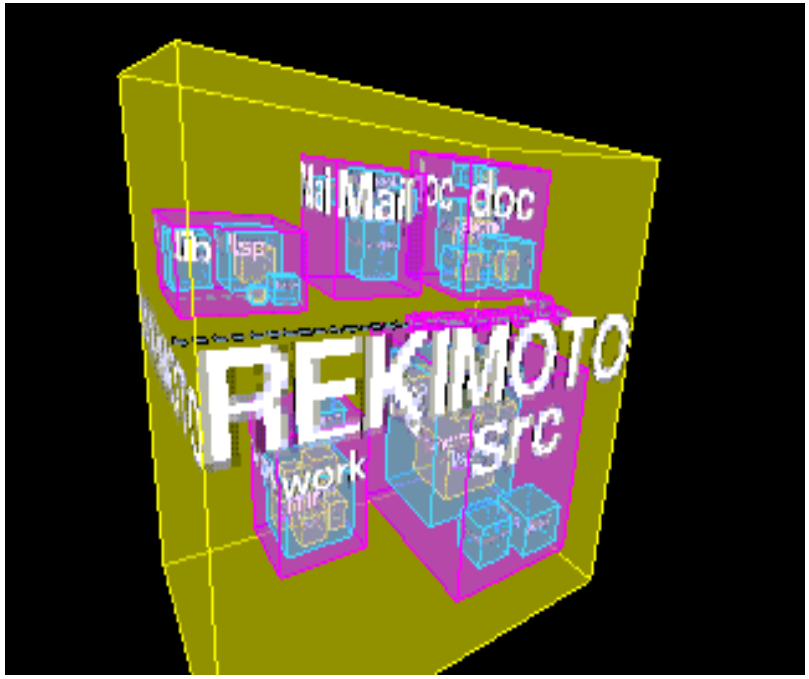

Ack.: http://www.cs.umd.edu/hcil/treemap-history/all102001.jpg

# Tree-Map of a File System (Schneiderman)



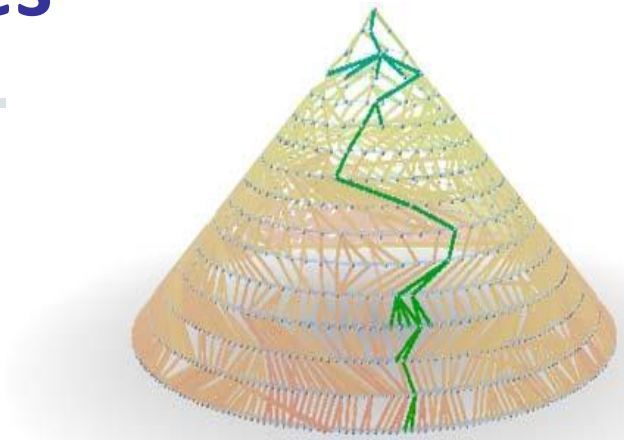
# InfoCube

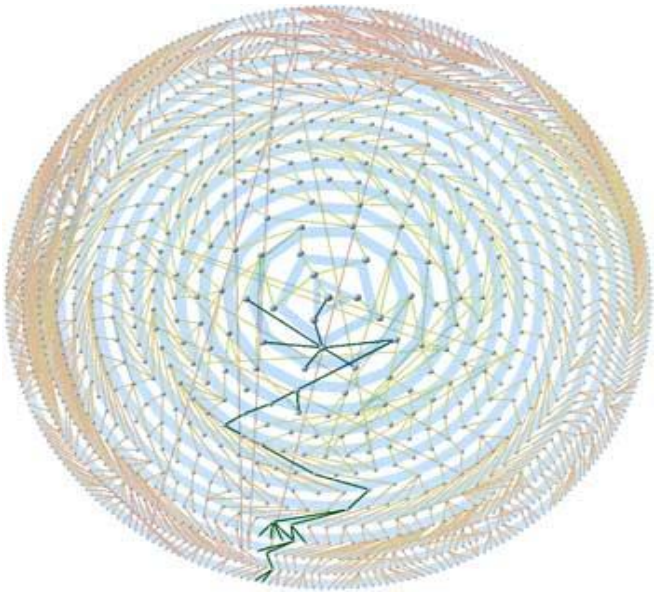
- A 3-D visualization technique where hierarchical information is displayed as nested semi-transparent cubes
- The outermost cubes correspond to the top-level data, while the subnodes or the lower-level data are represented as smaller cubes inside the outermost cubes, and so on



# Three-D Cone Trees

- *3D cone tree* visualization technique works well for up to a thousand nodes or so
- First build a *2D circle tree* that arranges its nodes in concentric circles centered on the root node
- Cannot avoid overlaps when projected to 2D
- G. Robertson, J. Mackinlay, S. Card. “Cone Trees: Animated 3D Visualizations of Hierarchical Information”, *ACM SIGCHI'91*
- Graph from Nadeau Software Consulting website: Visualize a social network data set that models the way an infection spreads from one person to





Ack.: http://nadeausoftware.com/articles/visualization

# Visualizing Complex Data and Relations

- Visualizing non-numerical data: text and social networks
- Tag cloud: visualizing user-generated tags
  - The importance of the tag is represented by font size/color
- Besides text data, there are also methods to visualize relationships, such as visualizing social networks


Newsmap: Google News Stories in 2005