UNIT-I

What is Data?

Data is distinct pieces of information, usually formatted in a special way". Data can be measured, collected, reported, and analyzed, whereupon it is often visualized using graphs, images, or other analysis tools. Raw data ("unprocessed data") may be a collection of numbers or characters before it's been "cleaned" and corrected by researchers.

What is Information ?

Information is data that has been processed , organized, or structured in a way that makes it meaningful, valuable and useful.

Categories of Data

Data can be catogeries into two main parts –

Structured Data: This type of data is organized data into specific format, making it easy to search , analyze and process. Structured data is found in a relational databases that includes information like numbers, data and categories.

UnStructured Data: Unstructured data does not conform to a specific structure or format. It may include some text documents , images, videos, and other data that is not easily organized or analyzed without additional processing.

What is Data Mining

Data Mining:

Definition: Data mining is the process of analyzing large datasets to discover patterns, relationships, correlations, or meaningful insights that can help in making informed decisions and predictions.

Purpose: The primary purpose of data mining is to extract valuable knowledge and information from large volumes of data that might be hidden or not readily apparent. It involves using advanced statistical and machine learning techniques to identify patterns and trends.

Functions: Data mining algorithms and techniques are applied to the data to identify associations, clusters, classifications, and anomalies. It helps in understanding customer behavior, predicting trends, detecting fraud, and making data-driven business decisions.

Usage: Data mining is widely used in areas such as marketing analysis, customer segmentation, recommendation systems, fraud detection, healthcare research, and financial forecasting.

Goals of Data Mining:

- The goal of data mining is to extract useful information from large datasets and use it to make predictions or inform decision-making.
- Data mining is important because it allows organizations to uncover insights and trends in their data that would be difficult or impossible to discover manually.
- This can help organizations make better decisions, improve their operations, and gain a competitive advantage.

Data Mining History and Origins

One of the earliest and most influential pioneers of data mining was Dr. Herbert Simon, a Nobel laureate in economics who is widely considered to be the father of artificial intelligence. In the 1950s and 1960s, Simon and his colleagues developed a number of algorithms and techniques for extracting useful information and insights from data, including clustering, classification, and decision trees.

In the 1980s and 1990s, the field of data mining continued to evolve, and new algorithms and techniques were developed to address the challenges of working with large and complex data sets. The development of data mining software and platforms, such as SAS, SPSS, and RapidMiner, made it easier for organizations to apply data mining techniques to their data.

In recent years, the availability of large data sets and the growth of cloud computing and big data technologies have made data mining even more powerful and widely used. Today, data mining is a crucial tool for many organizations and industries and is used to extract valuable insights and information from data sets in a wide range of domains.

Tasks of Data Mining

1. Classification: Categorizing data into predefined classes.

2. Clustering: Grouping similar data points together.

3. Regression: Predicting numerical values based on data relationships.

4. Association Rule Mining: Discovering interesting relationships between variables.

5. Anomaly Detection: Identifying unusual patterns in data.

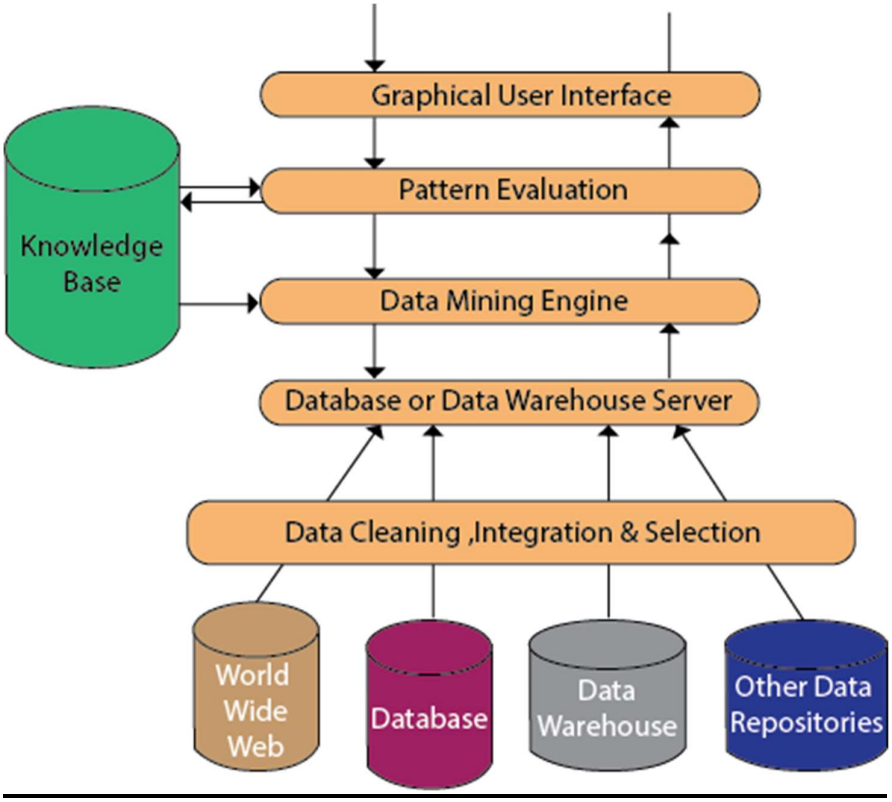
6. Text Mining: Extracting insights from unstructured text data.

7. Prediction and Forecasting: Predicting future trends based on historical data.

8. Pattern Mining: Identifying recurring patterns in sequential data.

9. Feature Selection and Dimensionality Reduction: Identifying relevant features and reducing dataset complexity.

Architecture of Data Mining



Data mining architecture typically consists of several components:

1. **Data Sources**: These are the repositories of data where the raw information resides. Sources can

include databases, data warehouses, websites, and more.

2. **Data Cleaning and Integration**: This stage involves preprocessing the data to ensure its qualityand compatibility for mining. It includes tasks like removing noise, handling missing values, and integrating data from different sources.

3. Data Selection and Transformation: Here, relevant data subsets are selected for analysis

based on the mining goals. The selected data may also undergo transformation to better suit the mining algorithms.

4. Data Mining Engine: This is the core component where various data mining algorithms are applied to the prepared data to discover patterns, trends, and insights.

5. Pattern Evaluation: Once patterns are discovered, they need to be evaluated for their relevance, validity, and usefulness. This step often involves statistical techniques and domain expertise.

6. Knowledge Presentation: Finally, the discovered knowledge is presented to users in a comprehensible format, such as reports, visualizations, or dashboards, to aid in decision making.

Throughout this process, feedback loops may exist where insights gained from the data mining results inform subsequent data selection, cleaning, or mining steps, creating a continuous improvement cycle.

Data Mining Process

The data mining process typically involves several key stages:

1. Understanding the Business Problem: The first step is to clearly understand the business problem or objective that data mining aims to address. This involves collaborating closely with domain experts to identify key questions and goals.

2. Data Collection: In this stage, relevant data is gathered from various sources such as databases, data warehouses, spreadsheets, or even web scraping. The data collected should be comprehensive and representative of the problem domain.

3. Data Preprocessing: Raw data often requires preprocessing to ensure its quality and suitability for analysis. This includes tasks such as cleaning data to remove errors and inconsistencies, handling missing values, and transforming data into a suitable format for analysis.

4. Exploratory Data Analysis (EDA): EDA involves examining the collected data to understand its characteristics, identify patterns, and detect outliers or anomalies. Techniques such as descriptive statistics, data visualization, and clustering may be used during this stage.

5. Feature Selection and Engineering: Feature selection involves identifying the most relevant variables (features) that will be used for analysis, while feature engineering may involve creating new features or transforming existing ones to enhance the predictive power of the model.

6. Model Selection and Training: Based on the nature of the problem and the available data,suitable data mining algorithms or models are selected. These may include techniques such as decision trees, neural networks, support vector machines, or clustering algorithms. The selected models are then trained on the prepared data.

7. Model Evaluation: Trained models need to be evaluated to assess their performance and generalization ability. This involves using evaluation metrics such as accuracy, precision, recall, or F1-score, and techniques such as cross-validation to ensure robustness.

8. Model Deployment: Once a satisfactory model is obtained, it is deployed into production to make predictions or generate insights on new, unseen data. This may involve integrating the model into existing systems or workflows.

9. Monitoring and Maintenance: Deployed models should be regularly monitored to ensure they continue to perform effectively over time. This may involve monitoring for concept drift (changes in the underlying data distribution) and updating the model or its parameters as necessary.

Throughout the entire data mining process, it's essential to maintain a clear focus on the business objectives and involve domain experts at each stage to ensure that the insights gained are relevant and actionable.

Classification of data mining

Classification Based on the mined Databases

A data mining system can be classified based on the types of databases that have been mined. A database system can be further segmented based on distinct principles, such as data models, types of data, etc., which further assist in classifying a data mining system.

For example, if we want to classify a database based on the data model, we need to select either relational, transactional, object-relational or data warehouse mining systems.

Classification Based on the type of Knowledge Mined

A data mining system categorized based on the kind of knowledge mind may have the following functionalities:

1. Characterization

2. Discrimination

3. Association and Correlation Analysis

4. Classification

5. Prediction

6. Outlier Analysis

7. Evolution Analysis

Classification Based on the Techniques Utilized

A data mining system can also be classified based on the type of techniques that are being incorporated.

These techniques can be assessed based on the involvement of user interaction involved or the methods of analysis employed.

Classification Based on the Applications Adapted

Data mining systems classified based on adapted applications adapted are as follows:

1. Finance

2. Telecommunications

3. DNA

4. Stock Markets

5. E-mail

What is KDD (Knowledge Discovery in Databases).

KDD is a computer science field specializing in extracting previously unknown and interesting information from raw data. KDD is the whole process of trying to make sense of data by developing appropriate methods or techniques. The following steps are included in KDD process:

Data Cleaning

Data cleaning is defined as removal of noisy and irrelevant data from collection.

- Cleaning in case of Missing values.
- Cleaning noisy data, where noise is a random or variance error.
- Cleaning with Data discrepancy detection and Data transformation tools.

Data Integration

Data integration is defined as heterogeneous data from multiple sources combined in a common source(DataWarehouse). Data integration using Data

Migration tools, Data Synchronization tools and ETL(Extract-Load-Transformation) process.

Data Selection

Data selection is defined as the process where data relevant to the analysis is decided and retrieved from the data collection. For this we can use Neural network, Decision Trees, Naive bayes, Clustering, and Regression methods.

Data Transformation

Data Transformation is defined as the process of transforming data into appropriate form required by mining procedure. Data Transformation is a two step process:

- Data Mapping: Assigning elements from source base to destination to capture transformations.
- Code generation: Creation of the actual transformation program.

Data Mining

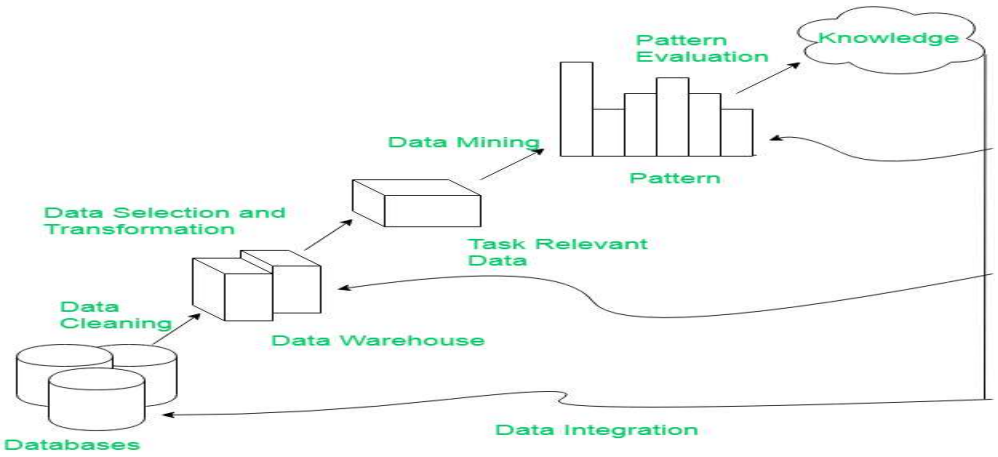
Data mining is defined as techniques that are applied to extract patterns potentially useful. It transforms task relevant data into patterns, and decides purpose of model using classification or characterization.

Pattern Evaluation

Pattern Evaluation is defined as identifying strictly increasing patterns representing knowledge based on given measures. It find interestingness score of each pattern, and uses summarization and Visualization to make data understandable by user.

Knowledge Representation

This involves presenting the results in a way that is meaningful and can be used to make decisions.



Difference between KDD and Data Mining

Parameter	KDD	Data Mining
Definition	KDD refers to a process of identifying valid, novel, potentially useful, and ultimately understandable patterns and relationships in data.	Data Mining refers to a process of extracting useful and valuable information or patterns from large data sets.
Objective	To find useful knowledge from data.	To extract useful information from data.
Techniques Used	Data cleaning, data integration, data selection, data transformation, data mining, pattern evaluation, and knowledge representation and visualization.	Association rules, classification, clustering, regression, decision trees, neural networks, and dimensionality reduction**.**
Output	Structured information, such as rules and models, that can be used to make decisions or predictions	Patterns, associations, or insights that can be used to improve decision-making or understanding
Focus	Focus is on the discovery of useful knowledge, rather than simply finding patterns in data.	Data mining focus is on the discovery of patterns or relationships in data.
Role of domain expertise	Domain expertise is important in KDD, as it helps in defining the goals of the process, choosing appropriate data, and interpreting the results.	Domain expertise is important in KDD, as it helps in defining the goals of the process, choosing appropriate data, and interpreting the results.

What is the difference between DBMS and Data mining?

Main Differences:

Scope: DBMS focuses on efficiently managing and storing data, ensuring data integrity and security. Data mining, on the other hand, focuses on analyzing data to discover meaningful patterns and insights.

Purpose: DBMS is used for data storage, retrieval, and management. Data mining is used for knowledge discovery and gaining insights from the data.

Functionality: DBMS provides functionalities for data storage, retrieval, and manipulation. Data mining employs algorithms and statistical techniques to identify patterns and relationships within the data.

Role: DBMS serves as the foundation for data storage and retrieval, enabling efficient data handling. Data mining is a process that builds on top of the data stored in the DBMS to extract valuable information.

In summary, DBMS is the infrastructure for storing and managing data, while data mining is a process of analyzing and extracting knowledge from the data stored in the DBMS.

What is OLAP?

OLAP stands for Online Analytical Processing. It is a computing method that allows users to extract useful information and query data in order to analyze it from different angles. For example, OLAP business intelligence queries usually aid in financial reporting, budgeting, predict future sales, trends analysis and other purposes. It enables the user to analyze database information from different database systems simultaneously. OLAP data is stored in multidimensional databases.

OLAP and data mining look similar since they operate on data to gain knowledge, but the major difference is how they operate on data. OLAP tools provide multidimensional data analysis and a summary of the data.

Key features of OLAP

- It supports complex calculations
- Time intelligence
- It has a multidimensional view of data
- Business-focused calculations
- Flexible and self-service reporting

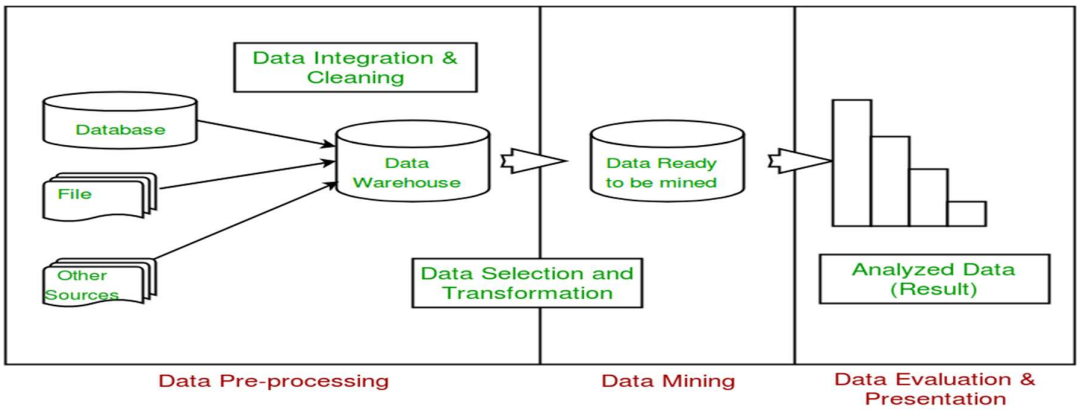
- Applications of OLAP
- Database Marketing
- Marketing and sales analysis

Data Mining	OLAP
Data mining refers to the field of computer science, which deals with the extraction of data, trends and patterns from huge sets of data.	OLAP is a technology of immediate access to data with the help of multidimensional structures.
It deals with the data summary.	It deals with detailed transaction-level data.
It is discovery-driven.	It is query driven.
It is used for future data prediction.	It is used for analyzing past data.
It has huge numbers of dimensions.	It has a limited number of dimensions.
Bottom-up approach.	Top-down approach.
It is an emerging field.	It is widely used.

Data Mining as a Whole Process

The whole process of Data Mining consists of three main phases:

Data Pre-processing – Data cleaning, integration, selection, and transformation takes place
Data Extraction – Occurrence of exact data mining
Data Evaluation and Presentation – Analyzing and presenting results



What is Data Mining Techniques?

Data mining techniques are algorithms and methods used to extract information and insights from data sets.

1. Regression

Regression is a data mining technique that is used to model the relationship between a dependent variable and one or more independent variables. In regression analysis, the goal is to fit a mathematical model to the data that can be used to make predictions or forecasts about the dependent variable based on the values of the independent variables.

There are many different types of regression models, including linear regression, logistic regression, and non-linear regression. In general, regression models are used to answer questions such as:

- What is the relationship between the dependent and independent variables?
- How well does the model fit the data?
- How accurate are the predictions or forecasts made by the model?

2. Classification

Classification is a data mining technique that is used to predict the class or category of an item or instance based on its characteristics or attributes. There are many different types of classification models, including decision trees, k-nearest neighbours, and support vector machines. In general, classification models are used to answer questions such as:

- What is the relationship between the classes and the attributes
- How well does the model fit the data?

- How accurate are the predictions made by the model?

3. Clustering

Clustering is a data mining technique that is used to group items or instances in a data set into clusters or groups based on their similarity or proximity. In clustering analysis, the goal is to identify and explore the natural structure or organization of the data, and to uncover hidden patterns and relationships.

There are many different types of clustering algorithms, including k-means clustering, hierarchical clustering, and density-based clustering. In general, clustering is used to answer questions such as:

- What is the natural structure or organization of the data?
- What are the main clusters or groups in the data?
- How similar or dissimilar are the items in the data set?

4. Association rule mining

Association rule mining is a data mining technique that is used to identify and explore relationships between items or attributes in a data set. In association rule mining, the goal is to identify patterns and rules that describe the co-occurrence or occurrence of items or attributes in the data set and to evaluate the strength and significance of these patterns and rules.

There are many different algorithms and methods for association rule mining, including the Apriori algorithm and the FP-growth algorithm. In general, association rule mining is used to answer questions such as

- What are the main patterns and rules in the data?
- How strong and significant are these patterns and rules?
- What are the implications of these patterns and rules for the data set and the domain?

5. Dimensionality Reduction

Dimensionality reduction is a data mining technique that is used to reduce the number of dimensions or features in a data set while retaining as much information and structure as possible. There are many different methods for dimensionality reduction, including principal component analysis (PCA), independent component analysis (ICA), and singular value decomposition (SVD). In general, dimensionality reduction is used to answer questions such as:

- What are the main dimensions or features in the data set?

- How much information and structure can be retained in a lower-dimensional space?
- How can the data be visualized and analyzed in a lower-dimensional space?

6. Anomaly Detection: Anomaly detection identifies outliers or anomalies in data that deviate from normal patterns. It is used for detecting fraud, network intrusions, and equipment failures.Techniques include statistical methods, clustering-based approaches, and machine learning algorithms such as isolation forests and one-class SVM.

7. Sequential Pattern Mining: Sequential pattern mining discovers patterns that occur sequentially or temporally in data. It is used in applications such as analyzing customer behavior over time or identifying patterns in sequences of events.Examples include the Prefix Span algorithm and the GSP (Generalized Sequential Pattern)algorithm.

8. Text Mining: Text mining techniques extract useful information from unstructured text data. This includes tasks such as sentiment analysis, topic modeling, named entity recognition, and document classification. Techniques such as natural language processing (NLP) and machine learning algorithms are commonly used in text mining.

Benefits of Data Mining

Improved decision-making: Data mining can provide valuable insights that can help organizations make better decisions by identifying patterns and trends in large data sets.

Increased efficiency: Data mining can automate repetitive and time-consuming tasks, such as data cleaning and preparation, which can help organizations save time and resources.

Enhanced competitiveness: Data mining can help organizations gain a competitive edge by uncovering new business opportunities and identifying areas for improvement.

Improved customer service: Data mining can help organizations better understand their customers and tailor their products and services to meet their needs.

Fraud detection: Data mining can be used to identify fraudulent activities by detecting unusual patterns and anomalies in data.

Predictive modeling: Data mining can be used to build models that can predict future events and trends, which can be used to make proactive decisions.

New product development: Data mining can be used to identify new product opportunities by analyzing customer purchase patterns and preferences.

Risk management: Data mining can be used to identify potential risks by analyzing data on customer behavior, market conditions, and other factors.

Challenges and Issues in Data Mining

1]Data Quality

The quality of data used in data mining is one of the most significant challenges. The accuracy, completeness, and consistency of the data affect the accuracy of the results obtained. The data may contain errors, omissions, duplications, or inconsistencies, which may lead to inaccurate results.

To address these challenges, data mining practitioners must apply data cleaning and data preprocessing techniques to improve the quality of the data

2]Data Complexity

Data complexity refers to the vast amounts of data generated by various sources, such as sensors, social media, and the internet of things (IoT). The complexity of the data may make it challenging to process, analyze, and understand. In addition, the data may be in different formats, making it challenging to integrate into a single dataset.

To address this challenge, data mining practitioners use advanced techniques such as clustering, classification, and association rule mining.

3]Data Privacy and Security

Data privacy and security is another significant challenge in data mining. As more data is collected, stored, and analyzed, the risk of data breaches and cyber-attacks increases. The data may contain personal, sensitive, or confidential information that must be protected. Moreover, data privacy regulations such as GDPR, CCPA, and HIPAA impose strict rules on how data can be collected, used, and shared.

To address this challenge, data mining practitioners must apply data anonymization and data encryption techniques to protect the privacy and security of the data. Data anonymization involves removing personally identifiable information (PII) from the data, while data encryption involves using algorithms to encode the data to make it unreadable to unauthorized users.

4]Scalability

Data mining algorithms must be scalable to handle large datasets efficiently. As the size of the dataset increases, the time and computational resources required to perform data mining operations also increase.

To address this challenge, data mining practitioners use distributed computing frameworks such as Hadoop and Spark.

5]Interpretability

Data mining algorithms can produce complex models that are difficult to interpret. This is because the algorithms use a combination of statistical and mathematical techniques to identify patterns and relationships in the data.

To address this challenge, data mining practitioners use visualization techniques to represent the data and the models visually.

Data Mining Applications

Data mining is used by a wide range of organizations and individuals across many different industries and domains. Some examples of who uses data mining include:

Businesses and Enterprises – Many businesses and enterprises use data mining to extract useful insights and information from their data, in order to make better decisions, improve their operations, and gain a competitive advantage. For example, a retail company might use data mining to identify customer trends and preferences or to predict demand for its products.

Government Agencies and Organizations – Government agencies and organizations use data mining to analyze data related to their operations and the population they serve, in order to make better decisions and improve their services. For example, a health department might use data mining to identify patterns and trends in public health data or to predict the spread of infectious diseases.

Academic and Research Institutions – Academic and research institutions use data mining to analyze data from their research projects and experiments, in order to identify patterns, relationships, and trends in the data. For example, a university might use data mining to analyze data from a clinical trial or to explore the relationships between different variables in a social science study.

Individuals – Many individuals use data mining to analyze their own data, in order to better understand and manage their personal information and activities.

For example, a person might use data mining to analyze their financial data and identify patterns in their spending or to analyze their social media data and understand their online behavior and interactions.

UNIT-II

What Is a Data Warehouse?

A Data Warehouse (DW) is a relational database that is designed for query and analysis rather than transaction processing. It includes historical data derived from transaction data from single and multiple sources.

A Data Warehouse provides integrated, enterprise-wide, historical data and focuses on providing support for decision-makers for data modeling and analysis.

A Data Warehouse can be viewed as a data system with the following attributes:

- It is a database designed for investigative tasks, using data from various applications.
- It supports a relatively small number of clients with relatively long interactions.
- It includes current and historical data to provide a historical perspective of information.
- Its usage is read-intensive.
- It contains a few large tables.

Benefits of Data Warehouse

1.Understand business trends and make better forecasting decisions.

2.Data Warehouses are designed to perform well enormous amounts of data.

3.The structure of data warehouses is more accessible for end-users to navigate, understand, and query.

4.Queries that would be complex in many normalized databases could be easier to build and maintain in data warehouses.

5.Data warehousing is an efficient method to manage demand for lots of information from lots of users.

6.Data warehousing provide the capabilities to analyze a large amount of historical data.

What is Multi-Dimensional Data Model?

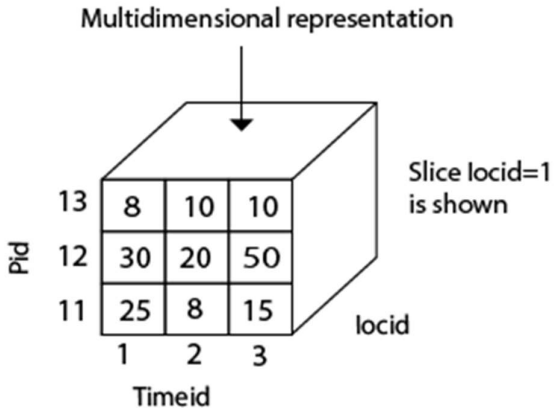
The dimensions are the perspectives or entities concerning which an organization keeps records. For example, a shop may create a sales data warehouse to keep records of the store's sales for the dimension time, item, and location. These dimensions allow the save to keep track of things, for example, monthly sales of items and the locations at which the items were sold. Each dimension has a table

related to it, called a dimensional table, which describes the dimension further. For example, a dimensional table for an item may contain the attributes item_name, brand, and type.

A multidimensional data model is organized around a central theme, for example, sales. This theme is represented by a fact table. Facts are numerical measures. The fact table contains the names of the facts or measures of the related dimensional tables.

Tabular representation

Pid	Timeid	locid	Sales
11	1	1	25
11	2	1	8
11	3	1	15
12	1	1	30
12	2	1	20
12	3	1	50
13	1	1	8
13	2	1	10
13	3	1	10
11	1	2	35



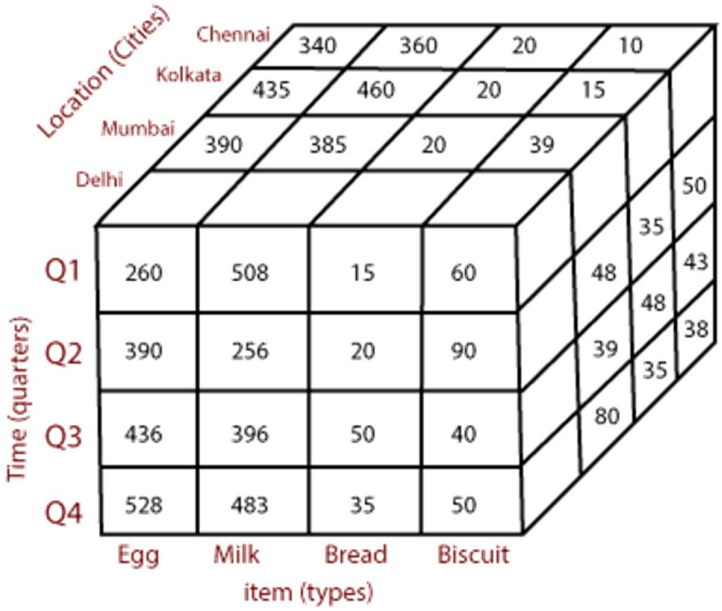
Consider the data of a shop for items sold per quarter in the city of Delhi. The data is shown in the table. In this 2D representation, the sales for Delhi are shown for the time dimension (organized in quarters) and the item dimension (classified according to the types of an item sold). The fact or measure displayed in rupee_sold (in thousands).

Location="Delhi"				
Time (quarter)	item (type)			
	Egg	Milk	Bread	Biscuit
Q1	260	508	15	60
Q2	390	256	20	90
Q3	436	396	50	40
Q4	528	483	35	50

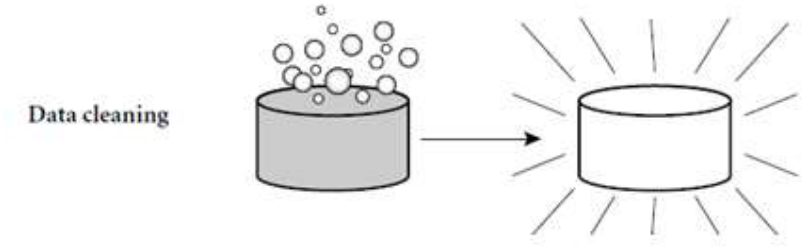
Now, if we want to view the sales data with a third dimension, For example, suppose the data according to time and item, as well as the location is considered for the cities Chennai, Kolkata, Mumbai, and Delhi. These 3D data are shown in the table. The 3D data of the table are represented as a series of 2D tables.

	Location="Chennai"				Location="Kolkata"				Location="Mumbai"				Location="Delhi"			
	item				item				item				item			
Time	Egg	Milk	Bread	Biscuit	Egg	Milk	Bread	Biscuit	Egg	Milk	Bread	Biscuit	Egg	Milk	Bread	Biscuit
Q1	340	360	20	10	435	460	20	15	390	385	20	39	260	508	15	60
Q2	490	490	16	50	389	385	45	35	463	366	25	48	390	256	20	90
Q3	680	583	46	43	684	490	39	48	568	594	36	39	436	396	50	40
Q4	535	694	39	38	335	365	83	35	338	484	48	80	528	483	35	50

Conceptually, it may also be represented by the same data in the form of a 3D data cube (What is Data Cube? When data is grouped or combined in multidimensional matrices called Data Cubes.), as shown in fig:



Data Cleaning :



• Real-world data tend to be incomplete, noisy, and inconsistent.

• Data cleaning routines attempt to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data.

• Data cleaning tasks include:

1. **Fill in missing values**

- The tuple is ignored when it includes several attributes with missing values.
- The values are filled manually for the missing value.
- The same global constant can fill the values.
- The attribute mean can fill the missing values.
- The most probable value can fill the missing values

2. **Identify outliers and smooth out noisy data**

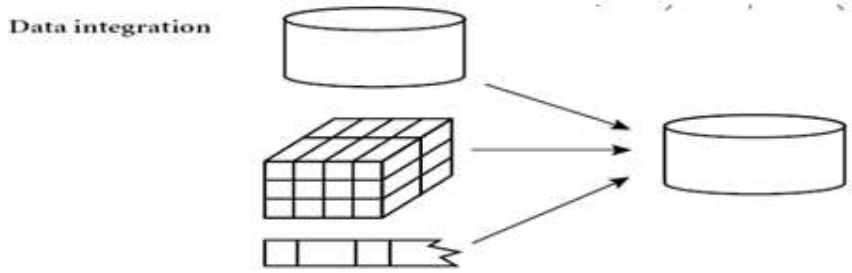
Noise is a random error or variance in a measured variable. And can be smoothened using the following steps:

- Binning − These methods smooth out a arrange data value by consulting its “neighborhood,” especially, the values around the noisy information. The arranged values are distributed into multiple buckets or bins. Because binning methods consult the neighborhood of values, they implement local smoothing.
- Regression − Data can be smoothed by fitting the information to a function, including with regression. Linear regression contains finding the “best” line to fit two attributes (or variables) so that one attribute can be used to forecast the other. Multiple linear regression is a development of linear regression, where more than two attributes are contained and the data are fit to a multidimensional area.
- Clustering − Clustering supports in identifying the outliers. The same values are organized into clusters and those values which fall outside the cluster are known as outliers.

- Combined computer and human inspection − The outliers can also be recognized with the support of computer and human inspection. The outliers pattern can be descriptive or garbage. Patterns having astonishment value can be output to a list.

3.Inconsistence data − The inconsistency can be recorded in various transactions, during data entry, or arising from integrating information from multiple databases. Some redundancies can be recognized by correlation analysis. Accurate and proper integration of the data from various sources can decrease and avoid redundancy.

Data Integration



Data integration is one of the steps of data pre-processing that involves combining data residing in different sources and providing users with a unified view of these data.

• It merges the data from multiple data stores (data sources)

• It includes multiple databases, data cubes or flat files.

• Metadata, Correlation analysis, data conflict detection, and resolution of semantic heterogeneity contribute towards smooth data integration.

• There are mainly 2 major approaches for data integration - commonly known as "tight coupling approach" and "loose coupling approach".

Tight Coupling

o Here data is pulled over from different sources into a single physical location through the process of ETL - Extraction, Transformation and Loading.

o The single physical location provides an uniform interface for querying the data.

ETL layer helps to map the data from the sources so as to provide a uniform data

o warehouse. This approach is called tight coupling since in this approach the data is tightly coupled with the physical repository at the time of query.

ADVANTAGES:

- Independence (Lesser dependency to source systems since data is physically copied over)
- Faster query processing
- Complex query processing
- Advanced data summarization and storage possible
- High Volume data processing

DISADVANTAGES: 1. Latency (since data needs to be loaded using ETL)

2.Costlier (data localization, infrastructure, security)

Loose Coupling

o Here a virtual mediated schema provides an interface that takes the query from the user, transforms it in a way the source database can understand and then sends the query directly to the source databases to obtain the result.

o In this approach, the data only remains in the actual source databases.

o However, mediated schema contains several "adapters" or "wrappers" that can connect back to the source systems in order to bring the data to the front end.

ADVANTAGES:

Data Freshness (low latency - almost real time)

Higher Agility (when a new source system comes or existing source system changes - only the corresponding adapter is created or changed - largely not affecting the other parts of the system)

Less costlier (Lot of infrastructure cost can be saved since data localization not required)

DISADVANTAGES:

- Semantic conflicts
- Slower query response
- High order dependency to the data sources

For example, let's imagine that an electronics company is preparing to roll out a new mobile device. The marketing department might want to retrieve customer information from a sales department database and compare it to information from

the product department to create a targeted sales list. A good data integration system would let the marketing department view information from both sources in a unified way, leaving out any information that didn't apply to the search.

DATA TRANSFORMATION:

In data mining pre-processes and especially in metadata and data warehouse, we use data transformation in order to convert data from a source data format into destination data.

Data transformation $-2, 32, 100, 59, 48 \longrightarrow -0.02, 0.32, 1.00, 0.59, 0.48$

We can divide data transformation into 2 steps:

• Data Mapping:

It maps the data elements from the source to the destination and captures any transformation that must occur.

• Code Generation:

It creates the actual transformation program.

Data transformation:

• Here the data are transformed or consolidated into forms appropriate for mining.

• Data transformation can involve the following:

Smoothing:

- It works to remove noise from the data.
- It is a form of data cleaning where users specify transformations to correct data inconsistencies.
- Such techniques include binning, regression, and clustering.

Aggregation:

- Here summary or aggregation operations are applied to the data.
- This step is typically used in constructing a data cube for analysis of the data at multiple granularities.
- Aggregation is a form of data reduction.

Generalization :

• Here low-level or “primitive” (raw) data are replaced by higher-level concepts through the use of concept hierarchies.

• For example, attributes, like age, may be mapped to higher-level concepts, like youth, middle-aged, and senior.

• Generalization is a form of data reduction.

Normalization:

• Here the attribute data are scaled so as to fall within a small specified range, such as 1:0 to 1:0, or 0:0 to 1:0.

• Normalization is particularly useful for classification algorithms involving neural networks, or distance measurements such as nearest-neighbor classification and clustering

• For distance-based methods, normalization helps prevent attributes with initially large ranges (e.g., income).

• There are three methods for data normalization:

- min-max normalization
- z-score normalization
- normalization by decimal scaling:

Attribute construction:

- Here new attributes are constructed and added from the given set of attributes to help the mining process.
- Attribute construction helps to improve the accuracy and understanding of structure in high-dimensional data.
- By combining attributes, attribute construction can discover missing information about the relationships between data attributes that can be useful for knowledge discovery.

EG:The structure of stored data may vary between applications, requiring semantic mapping prior to the transformation process. For instance, two applications might store the same customer credit card information using slightly different structures:

APPLICATION A	EXAMPLE	APPLICATION B	EXAMPLE
Cardholder First Name	JOHN	Cardholder Name	JOHN DOE
Cardholder Last Name	DOE	Card Type	VISA
Card Type and Card Number	VISA 0123 4567 8910 1112	Card Number	0123 4567 8910 1112
Expiration Date	05/2012	Expiration Date	05/2012

To ensure that critical data isn't lost when the two applications are integrated, information from Application A needs to be reorganized to fit the data structure of Application B.

Data Reduction

Data reduction is a method of reducing the size of original data so that it may be represented in a much smaller space. While reducing data, data reduction techniques preserve data integrity.

Data Reduction Techniques

1.Dimensionality Reduction

Dimensionality reduction removes characteristics from the data set in question, resulting in a reduction in the size of the original data. It shrinks data by removing obsolete or superfluous characteristics.

1. **Wavelet Transform:** Assume that 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 using the wavelet transform. Then, because the data received from the wavelet transform may be abbreviated, how is it beneficial in data reduction? By keeping the smallest piece of the strongest wavelet coefficients, compressed data can be produced. Data cubes, sparse data, and skewed data can all benefit from the Wavelet transform.
2. **Principal Component Analysis:** Assume we have a data set with n properties that needs to be evaluated. The main component analysis finds k distinct tuples with n properties that may be used to describe the data collection. The original data may be cast on a considerably smaller space in this fashion, resulting in dimensionality reduction. Principal component analysis can be used on data that is sparse or skewed.

2. sNumerosity Reduction

The numerosity reduction decreases the size of the original data and expresses it in a much more compact format. There are two sorts of numerosity reduction techniques: parametric and non-parametric.

Parametric: Instead of keeping the original data, parametric numerosity reduction stores just data parameters. The regression and log-linear technique is one way for reducing parametric numerosity.

Non-Parametric: There is no model in a non-parametric numerosity reduction strategy. The non-parametric approach achieves a more uniform reduction, regardless of data size, but it may not accomplish the same volume of data reduction as the parametric technique. Histogram, Clustering, Sampling, Data Cube Aggregation, and Data Compression are at least four forms of Non-Parametric data reduction techniques.

3.Data Cube Aggregation

This method is used to condense data into a more manageable format. Data Cube Aggregation is a multidimensional aggregation that represents the original data set by aggregating at multiple layers of a data cube, resulting in data reduction.

Aggregation gives you with the needed data, which is considerably smaller in size, and we achieve data reduction without losing any data in this method.

4.Data Compression

Data compression is the process of altering, encoding, or transforming the structure of data in order to save space. By reducing duplication and encoding data in binary form, data compression creates a compact representation of information. Lossless compression refers to data that can be effectively recovered from its compressed state. Lossy compression, on the other hand, occurs when the original form cannot be restored from the compressed version. For data compression, dimensionality and numerosity reduction methods are also utilised.

5.**Discretization Operation**

Data discretization is a technique for dividing continuous nature qualities into data with intervals. We use labels of tiny intervals to replace several of the characteristics' constant values. This implies that mining results are presented in a clear and succinct manner.

Data Discretization:

Data discretization is a technique for dividing continuous nature qualities into data with intervals. We use labels of tiny intervals to replace several of the characteristics' constant values. This implies that mining results are presented in a clear and succinct manner.

Discretization techniques can be categorized depends on how the discretization is implemented, such as whether it uses class data or which direction it proceeds (i.e., top-down vs. bottom-up).

- If the process begins by first discovering one or a few points (known as split points or cut points) to split the whole attribute range, and then continue this recursively on the resulting intervals, it is known as **top-down discretization or splitting.**
- In **bottom-up discretization or merging**, it can start by considering all of the continuous values as potential split-points, removes some by merging neighbourhood values to form intervals, and then recursively applies this process to the resulting intervals.

UNIT-III

Mining Frequent Patterns:

Frequent pattern extraction is an essential mission in data mining that intends to uncover repetitive patterns or itemsets in a granted dataset. It encompasses recognizing collections of components that occur together frequently in a transactional or relational database. This procedure can offer valuable perceptions into the connections and affiliations among diverse components or features within the data.

The technique of frequent pattern mining is built upon a number of fundamental ideas.

Transactional and Relational Databases: The analysis is based on transaction databases, which include records or transactions that represent collections of objects. Items inside these transactions are grouped together as itemsets.

Support and Repeating Groupings: The importance of patterns is greatly influenced by support and confidence measurements. Support quantifies how frequently an itemset appears in the database, whereas confidence quantifies how likely it is that a rule generated from the itemset is accurate.

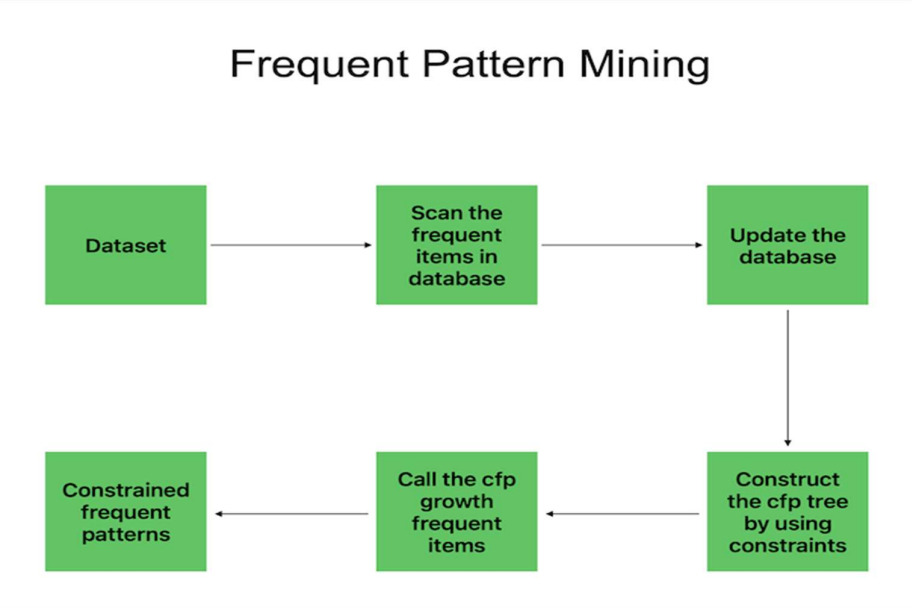
The Apriori algorithm, is one of the most well-known and widely used algorithms for repeating arrangement prospecting. It uses a breadth-first search strategy to discover repeating groupings efficiently. The algorithm works in multiple iterations. It starts by finding repeating individual objects by scanning the database once and counting the occurrence of each object. It then generates candidate groupings of size 2 by combining the repeating groupings of size 1. The support of these candidate groupings is calculated by scanning the database again. The process continues iteratively, generating candidate groupings of size k and calculating their support until no more repeating groupings can be found.

Support-based Pruning: During the Apriori algorithm's execution, aid-based pruning is used to reduce the search space and enhance efficiency. If an itemset is found to be rare (i.e., its aid is below the minimum aid threshold), then all its supersets are also assured to be rare. Therefore, these supersets are trimmed from further consideration. This trimming step significantly decreases the number of potential item sets that need to be evaluated in subsequent iterations..

Association Rule Mining: Frequent item sets can be further examined to discover association rules, which represent connections between different items. An association rule consists of an antecedent and a consequent (right-hand side), both of which are item sets. For instance, {milk, bread} => {eggs} is an

association rule. Association rules are produced from frequent itemsets by considering different combinations of items and calculating measures such as **aid, confidence**, and **lift**. Aid measures the frequency of both the antecedent and the consequent appearing together, while confidence measures the conditional probability of the consequent given the antecedent. Lift indicates the strength of the association between the antecedent and the consequent, considering their individual aid.

Applications: Frequent pattern mining has various practical uses in different domains. Some examples include market basket analysis, customer behavior analysis, web mining, bioinformatics, and network traffic analysis.



Applications of Frequent Pattern Mining

Market Basket Analysis

Market basket analysis frequently mines patterns to comprehend consumer buying patterns. Businesses get knowledge about product associations by recognizing itemsets that commonly appear together in transactions. This knowledge enables companies to improve recommendation systems and cross−sell efforts. Retailers can use this program to assist them in making data−driven decisions that will enhance customer happiness and boost sales.

Web usage mining

Web usage mining is examining user navigation patterns to learn more about how people use websites. In order to personalize websites and enhance their

performance, frequent pattern mining makes it possible to identify recurrent navigation patterns and session patterns. Businesses can change content, layout, and navigation to improve user experience and boost engagement by studying how consumers interact with a website.

Bioinformatics

The identification of relevant DNA patterns in the field of bioinformatics is made possible by often occurring pattern mining. Researchers can get insights into genetic variants, illness connections, and drug development by examining big genomic databases for recurrent patterns. In order to diagnose diseases, practice personalized medicine, and create innovative therapeutic strategies, frequent pattern mining algorithms help uncover important DNA sequences and patterns.

Frequent Item sets Mining Methods:

Association Rule Learning

Association rule learning is a type of unsupervised learning technique that checks for the dependency of one data item on another data item and maps accordingly so that it can be more profitable.

The association rule learning is one of the very important concepts of machine learning, and it is employed in Market Basket analysis, Web usage mining, continuous production, etc. Here market basket analysis is a technique used by the various big retailer to discover the associations between items.

For example, if a customer buys bread, he most likely can also buy butter, eggs, or milk, so these products are stored within a shelf or mostly nearby.

How does Association Rule Learning work?

Association rule learning works on the concept of If and Else Statement, such as if A then B.

Here the If element is called antecedent, and then statement is called as Consequent. These types of relationships where we can find out some association or relation between two items is known as single cardinality. It is all about creating rules, and if the number of items increases, then cardinality also increases accordingly. So, to measure the associations between thousands of data items, there are several metrics. These metrics are given below:

- Support
- Confidence
- Lift

Let's understand each of them:

Support

Support is the frequency of A or how frequently an item appears in the dataset. It is defined as the fraction of the transaction T that contains the itemset X. If there are X datasets, then for transactions T, it can be written as:

$$\text{Supp(X)}=\frac{Freq(X)}{T}$$

Confidence

Confidence indicates how often the rule has been found to be true. Or how often the items X and Y occur together in the dataset when the occurrence of X is already given. It is the ratio of the transaction that contains X and Y to the number of records that contain X.

$$\text{Confidence}=\frac{Freq(X,Y)}{Freq(X)}$$

Lift

It is the strength of any rule, which can be defined as below formula:

$$\text{Lift}=\frac{Supp(X,Y)}{Supp(X)\times Supp(Y)}$$

It is the ratio of the observed support measure and expected support if X and Y are independent of each other. It has three possible values:

- If Lift= 1: The probability of occurrence of antecedent and consequent is independent of each other.
- Lift>1: It determines the degree to which the two itemsets are dependent to each other.
- Lift<1: It tells us that one item is a substitute for other items, which means one item has a negative effect on another.

Apriori Algorithm:

Apriori algorithm refers to the algorithm which is used to calculate the association rules between objects. It means how two or more objects are related to one another. In other words, we can say that the apriori algorithm is an association rule leaning that analyzes that people who bought product A also bought product B.

The primary objective of the apriori algorithm is to create the association rule between different objects. The association rule describes how two or more objects are related to one another. Apriori algorithm is also called frequent pattern mining.

Components of Apriori algorithm

The given three components comprise the apriori algorithm.

- Support
- Confidence
- Lift

Support

Support refers to the default popularity of any product. You find the support as a quotient of the division of the number of transactions comprising that product by the total number of transactions. Hence, we get

Support (Biscuits) = (Transactions relating biscuits) / (Total transactions)

= 400/4000 = 10 percent.

Confidence

Confidence refers to the possibility that the customers bought both biscuits and chocolates together. So, you need to divide the number of transactions that comprise both biscuits and chocolates by the total number of transactions to get the confidence.

Hence,

Confidence = (Transactions relating both biscuits and Chocolate) / (Total transactions involving Biscuits)

= 200/400

= 50 percent.

It means that 50 percent of customers who bought biscuits bought chocolates also.

Lift

Consider the above example; lift refers to the increase in the ratio of the sale of chocolates when you sell biscuits. The mathematical equations of lift are given below.

Lift = (Confidence (Biscuits - chocolates)/ (Support (Biscuits)

= 50/10 = 5

It means that the probability of people buying both biscuits and chocolates together is five times more than that of purchasing the biscuits alone. If the lift value is below one, it requires that the people are unlikely to buy both the items together. Larger the value, the better is the combination.

How does the Apriori Algorithm work in Data Mining?

Consider the following dataset and we will find frequent itemsets and generate association rules for them.

TID	items
T1	I1, I2 , I5
T2	I2,I4
T3	I2,I3
T4	I1,I2,I4
T5	I1,I3
T6	I2,I3
T7	I1,I3
T8	I1,I2,I3,I5
T9	I1,I2,I3

minimum support count is 2

minimum confidence is 60%

Step-1: K=1

(I) Create a table containing support count of each item present in dataset – Called C1(candidate set)

Itemset	sup_count
I1	6
I2	7
I3	6
I4	2
I5	2

(II) compare candidate set item's support count with minimum support count(here min_support=2 if support_count of candidate set items is less than min_support then remove those items). This gives us itemset L1.

Itemset	sup_count
I1	6
I2	7
I3	6
I4	2
I5	2

Step-2: K=2

- Generate candidate set C2 using L1 (this is called join step). Condition of joining Lk-1 and Lk-1 is that it should have (K-2) elements in common.
- Check all subsets of an itemset are frequent or not and if not frequent remove that itemset.(Example subset of{I1, I2} are {I1}, {I2} they are frequent.Check for each itemset)
- Now find support count of these itemsets by searching in dataset.

Itemset	sup_count
I1,I2	4
I1,I3	4
I1,I4	1
I1,I5	2
I2,I3	4
I2,I4	2
I2,I5	2
I3,I4	0
I3,I5	1
I4,I5	0

(II) compare candidate (C2) support count with minimum support count(here min_support=2 if support_count of candidate set item is less than min_support then remove those items) this gives us itemset L2.

Itemset	sup_count
I1,I2	4
I1,I3	4
I1,I5	2
I2,I3	4
I2,I4	2
I2,I5	2
I2,I5	2

Step-3:

- Generate candidate set C3 using L2 (join step). Condition of joining Lk-1 and Lk-1 is that it should have (K-2) elements in common. So here, for L2, first element should match.
- So itemset generated by joining L2 is {I1, I2, I3} {I1, I2, I5} {I1, I3, i5} {I2, I3, I4} {I2, I4, I5} {I2, I3, I5}
- Check if all subsets of these itemsets are frequent or not and if not, then remove that itemset.(Here subset of {I1, I2, I3} are {I1, I2},{I2, I3},{I1, I3} which are frequent. For {I2, I3, I4}, subset {I3, I4} is not frequent so remove it. Similarly check for every itemset)
- find support count of these remaining itemset by searching in dataset.

Itemset	sup_count
I1,I2,I3	2
I1,I2,I5	2

(II) Compare candidate (C3) support count with minimum support count(here min_support=2 if support_count of candidate set item is less than min_support then remove those items) this gives us itemset L3.

Itemset	sup_count
I1,I2,I3	2
I1,I2,I5	2

Step-4:

- Generate candidate set C4 using L3 (join step). Condition of joining Lk-1 and Lk-1 (K=4) is that, they should have (K-2) elements in common. So here, for L3, first 2 elements (items) should match.
- Check all subsets of these itemsets are frequent or not (Here itemset formed by joining L3 is {I1, I2, I3, I5} so its subset contains {I1, I3, I5}, which is not frequent). So no itemset in C4
- We stop here because no frequent itemsets are found further.

Thus, we have discovered all the frequent item-sets. Now generation of strong association rule comes into picture. For that we need to calculate confidence of each rule.

Advantages of Apriori Algorithm

- It is used to calculate large itemsets.
- Simple to understand and apply.

Disadvantages of Apriori Algorithms

- Apriori algorithm is an expensive method to find support since the calculation has to pass through the whole database.
- Sometimes, you need a huge number of candidate rules, so it becomes computationally more expensive.
- At each step, candidate sets have to be built.
- To build the candidate sets, the algorithm has to repeatedly scan the database.

Frequent Pattern Growth Algorithm

Frequent Pattern Growth Algorithm overcomes the disadvantages of the Apriori algorithm by storing all the transactions in a Trie Data. Consider the following data:-

Transaction ID	Items
T1	{E,K,M,N,O,Y}
T2	{D,E,K,N,O,Y}
T3	{A,E,K,M}
T4	{C,KM,U,Y}
T5	{C,E,K,O,O}

The above-given data is a hypothetical dataset of transactions with each letter representing an item. The frequency of each individual item is computed:-

ITEM	FREQUENCY
A	1
C	2
D	1
E	4
I	1
K	5
M	3
N	2
O	4
U	1
Y	3

Let the minimum support be 3. A Frequent Pattern set is built which will contain all the elements whose frequency is greater than or equal to the minimum support. These elements are stored in descending order of their respective frequencies. After insertion of the relevant items, the set L looks like this:-

L = {K : 5, E : 4, M : 3, O : 4, Y : 3}

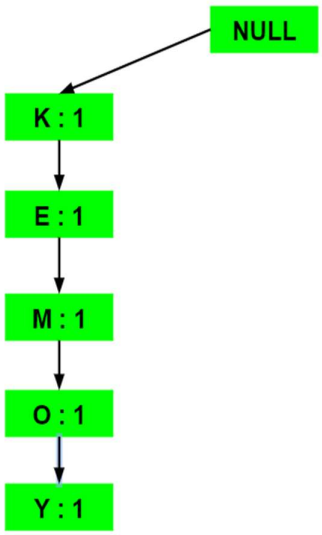
Now, for each transaction, the respective Ordered-Item set is built. It is done by iterating the Frequent Pattern set and checking if the current item is contained in the transaction in question. If the current item is contained, the item is inserted in the Ordered-Item set for the current transaction. The following table is built for all the transactions:

Transaction ID	Items	Ordered-Item Set
T1	{E,K,M,N,O,Y}	{K,E,M,O,Y}
T2	{D,E,K,N,O,Y}	{K,E,O,Y}
T3	{A,E,K,M}	{K,E,M}
T4	{C,KM,U,Y}	{K,M,Y}
T5	{C,E,K,O,O}	{K,E,O}

Now, all the Ordered-Item sets are inserted into a Trie Data Structure.

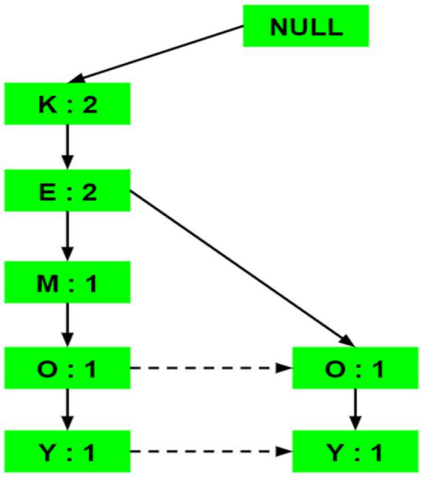
a) Inserting the set {K, E, M, O, Y}:

Here, all the items are simply linked one after the other in the order of occurrence in the set and initialize the support count for each item as 1.



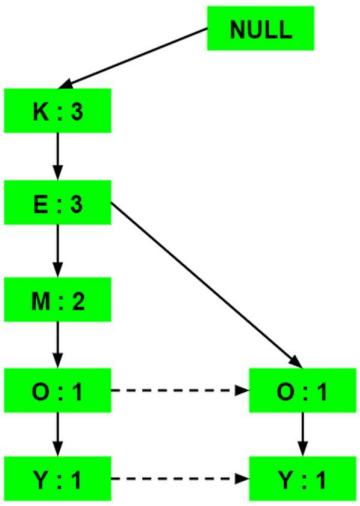
b) Inserting the set {K, E, O, Y}:

Till the insertion of the elements K and E, simply the support count is increased by 1. On inserting O we can see that there is no direct link between E and O, therefore a new node for the item O is initialized with the support count as 1 and item E is linked to this new node. On inserting Y, we first initialize a new node for the item Y with support count as 1 and link the new node of O with the new node of Y.

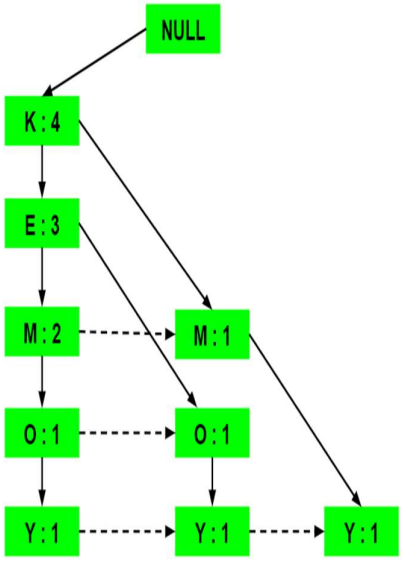


c) Inserting the set {K, E, M}:

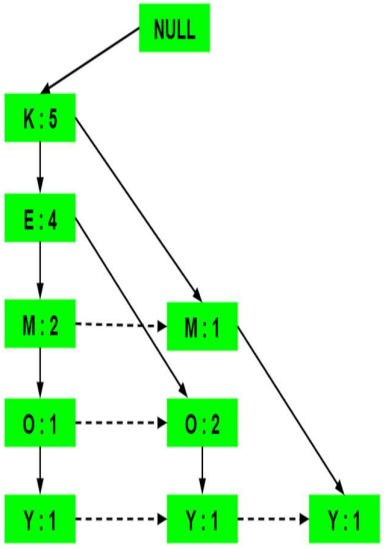
Here simply the support count of each element is increased by 1.



d) Inserting the set {K, M, Y}: Similar to step b), first the support count of K is increased, then new nodes for M and Y are initialized and linked accordingly.



e) Inserting the set {K, E, O}: Here simply the support counts of the respective elements are increased. Note that the support count of the new node of item O is increased.



Now, for each item, the **Conditional Pattern Base** is computed which is path labels of all the paths which lead to any node of the given item in the frequent-pattern tree. Note that the items in the below table are arranged in the ascending order of their frequencies.

Items	Conditional Pattern Base
Y	{{K,E,M,O : 1}, {K,E,O : 1}, {K,M : 1}}
O	{{K,E,M : 1}, {K,E : 2}}
M	{{K,E : 2}, {K : 1}}
E	{K : 4}
K	

Now for each item, the Conditional Frequent Pattern Tree is built. It is done by taking the set of elements that is common in all the paths in the Conditional Pattern Base of that item and calculating its support count by summing the support counts of all the paths in the Conditional Pattern Base.

Items	Conditional Pattern Base	Conditional Frequent Pattern Tree
Y	{{K,E,M,O : 1}, {K,E,O : 1}, {K,M : 1}}	{K : 3}
O	{{K,E,M : 1}, {K,E : 2}}	{K,E : 3}
M	{{K,E : 2}, {K : 1}}	{K : 3}
E	{K : 4}	{K : 4}
K		

From the Conditional Frequent Pattern tree, the Frequent Pattern rules are generated by pairing the items of the Conditional Frequent Pattern Tree set to the corresponding to the item as given in the below table.

Items	Frequent Pattern Generated
Y	{<K,Y : 3>}
O	{<K,O : 3>, <E,O : 3>, <E,K,O : 3>}
M	{<K,M : 3>}
E	{<E,K : 4>}
K	

For each row, two types of association rules can be inferred for example for the first row which contains the element, the rules K -> Y and Y -> K can be inferred. To determine the valid rule, the confidence of both the rules is calculated and the one with confidence greater than or equal to the minimum confidence value is retained.

Applications of Association Rule Learning

It has various applications in machine learning and data mining. Below are some popular applications of association rule learning:

- **Market Basket Analysis:** It is one of the popular examples and applications of association rule mining. This technique is commonly used by big retailers to determine the association between items.
- **Medical Diagnosis:** With the help of association rules, patients can be cured easily, as it helps in identifying the probability of illness for a particular disease.
- **Protein Sequence:** The association rules help in determining the synthesis of artificial Proteins.
- It is also used for the Catalog Design and Loss-leader Analysis and many more other applications.

UNIT-IV

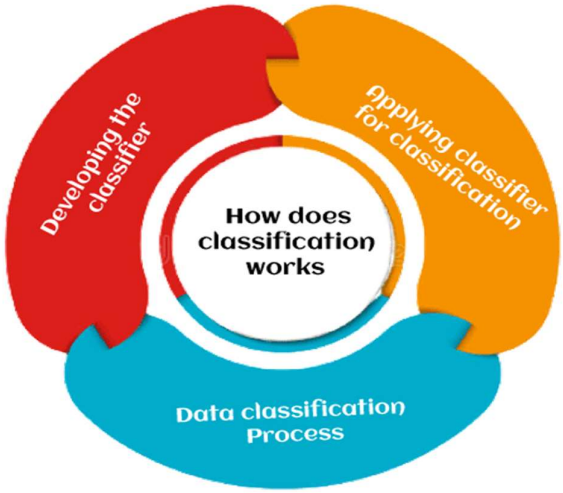
Basic Concept of Classification

Classification is a task in data mining that involves assigning a class label to each instance in a dataset based on its features. The goal of classification is to build a model that accurately predicts the class labels of new instances based on their features.

There are two main types of classification: binary classification and multi-class classification. Binary classification involves classifying instances into two classes, such as "spam" or "not spam", while multi-class classification involves classifying instances into more than two classes.

How does Classification Works?

There are two stages in the data classification system: classifier or model creation and classification classifier.



1.Developing the Classifier or model creation: This level is the learning stage or the learning process. The classification algorithms construct the classifier in this stage. A classifier is constructed from a training set composed of the records of databases and their corresponding class names. Each category that makes up the training set is referred to as a category or class. We may also refer to these records as samples, objects, or data points.

2.Applying classifier for classification: The classifier is used for classification at this level. The test data are used here to estimate the accuracy of the

classification algorithm. If the consistency is deemed sufficient, the classification rules can be expanded to cover new data records. It includes:

3.Sentiment Analysis: Sentiment analysis is highly helpful in social media monitoring. We can use it to extract social media insights. We can build sentiment analysis models to read and analyze misspelled words with advanced machine learning algorithms. The accurate trained models provide consistently accurate outcomes and result in a fraction of the time.

- **Document Classification:** We can use document classification to organize the documents into sections according to the content. Document classification refers to text classification; we can classify the words in the entire document. And with the help of machine learning classification algorithms, we can execute it automatically.
- **Image Classification:** Image classification is used for the trained categories of an image. These could be the caption of the image, a statistical value, a theme. You can tag images to train your model for relevant categories by applying supervised learning algorithms.
- **Machine Learning Classification:** It uses the statistically demonstrable algorithm rules to execute analytical tasks that would take humans hundreds of more hours to perform.

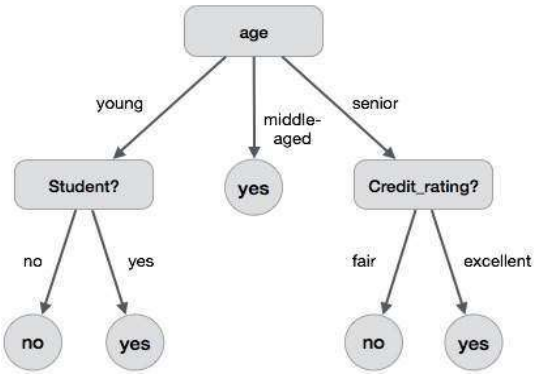
Data Classification Process: The data classification process can be categorized into five steps:

- Create the goals of data classification, strategy, workflows, and architecture of data classification.
- Classify confidential details that we store.
- Using marks by data labelling.
- To improve protection and obedience, use effects.
- Data is complex, and a continuous method is a classification.

Decision Tree Induction

A decision tree is a structure that includes a root node, branches, and leaf nodes. Each internal node denotes a test on an attribute, each branch denotes the outcome of a test, and each leaf node holds a class label. The top most node in the tree is the root node.

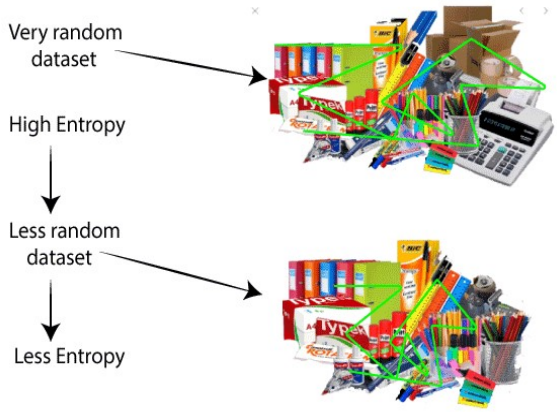
The following decision tree is for the concept buy_computer that indicates whether a customer at a company is likely to buy a computer or not. Each internal node represents a test on an attribute. Each leaf node represents a class.



Key factors:

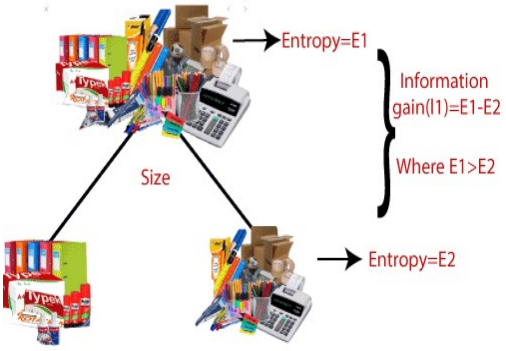
Entropy:

Entropy refers to a common way to measure impurity. In the decision tree, it measures the randomness or impurity in data sets.



Information Gain:

Information Gain refers to the decline in entropy after the dataset is split. It is also called Entropy Reduction. Building a decision tree is all about discovering attributes that return the highest data gain.



The benefits of having a decision tree are as follows

- It does not require any domain knowledge.
- It is easy to comprehend.
- The learning and classification steps of a decision tree are simple and fast.

Decision Tree Induction Algorithm

A machine researcher named J. Ross Quinlan in 1980 developed a decision tree algorithm known as ID3 (Iterative Dichotomiser). Later, he presented C4.5, which was the successor of ID3. ID3 and C4.5 adopt a greedy approach. In this algorithm, there is no backtracking; the trees are constructed in a top-down recursive divide-and-conquer manner.

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 node.

Attribute Selection Measure

Entropy:

Entropy is the measure of the degree of randomness or uncertainty in the dataset. In the case of classifications, It measures the randomness based on the distribution of class labels in the dataset.

The entropy for a subset of the original dataset having K number of classes for the ith node can be defined as:

$$H_i = -\sum_{k \epsilon K}^{n} p(i,k) \log_2 p(i,k)$$

Where,

- S is the dataset sample.
- k is the particular class from K classes
- p(k) is the proportion of the data points that belong to class k to the total number of data points in dataset sample S.
- Here p(i,k) should not be equal to zero.

Information Gain:

Information gain measures the reduction in entropy or variance that results from splitting a dataset based on a specific property.

$$\text{Information Gain(H, A)} = H - \sum \frac{|H_V|}{|H|} H_v$$

where

- A is the specific attribute or class label
- |H| is the entropy of dataset sample S
- |HV| is the number of instances in the subset S that have the value v for attribute A

Gini Impurity or index:

Gini Impurity is a score that evaluates how accurate a split is among the classified groups. The Gini Impurity evaluates a score in the range between 0 and 1, where 0 is when all observations belong to one class, and 1 is a random distribution of the elements within classes.

$$\text{Gini Impurity} = 1 - \sum p_i^2$$

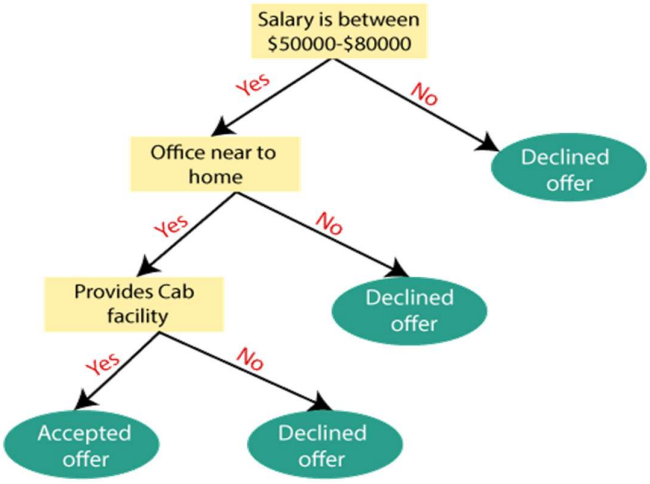
where

- A is the specific attribute or class label
- |H| is the entropy of dataset sample S
- |HV| is the number of instances in the subset S that have the value v for attribute A

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 -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.

Example: Suppose there is a candidate who has a job offer and wants to decide whether he should accept the offer or Not. So, to solve this problem, the decision tree starts with the root node (Salary attribute by ASM). The root node splits further into the next decision node (distance from the office) and one leaf node based on the corresponding labels. The next decision node further gets split into one decision node (Cab facility) and one leaf node. Finally, the decision node splits into two leaf nodes (Accepted offers and Declined offer). Consider the below diagram:



Tree Pruning

Tree pruning is performed in order to remove anomalies in the training data due to noise or outliers. The pruned trees are smaller and less complex.

Tree Pruning Approaches

There are two approaches to prune a tree −

- Pre-pruning − The tree is pruned by halting its construction early.
- Post-pruning - This approach removes a sub-tree from a fully grown tree.

Advantages of Decision Tree Induction

1. Easy to understand and interpret: Decision trees are a visual and intuitive model that can be easily understood by both experts and non-experts.
2. Handle both numerical and categorical data: Decision trees can handle a mix of numerical and categorical data, which makes them suitable for many different types of datasets.
3. Can handle large amounts of data: Decision trees can handle large amounts of data and can be updated with new data as it becomes available.
4. Can be used for both classification and regression tasks: Decision trees can be used for both classification, where the goal is to predict a discrete outcome, and regression, where the goal is to predict a continuous outcome.

Disadvantages of Decision Tree Induction

1. **Prone to overfitting:** Decision trees can become too complex and may not generalize well to new data. This can lead to poor performance on unseen data.
2. **Sensitive to small changes in the data:** Decision trees can be sensitive to small changes in the data, and a small change in the data can result in a significantly different tree.
3. **Biased towards attributes with many levels:** Decision trees can be biased towards attributes with many levels, and may not perform well on attributes with a small number of levels.

CART (Classification And Regression Tree) Algorithm:

CART(Classification And Regression Trees) is a variation of the decision tree algorithm. CART is a predictive algorithm used in Machine learning and it explains how the target variable's values can be predicted based on other matters. 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.

Gini index/Gini impurity

The Gini index is a metric for the classification tasks in CART. It stores the sum of squared probabilities of each class. It computes the degree of probability of a specific variable that is wrongly being classified when chosen randomly and a variation of the Gini coefficient. It works on categorical variables, provides outcomes either "successful" or "failure" and hence conducts binary splitting only.

The degree of the Gini index varies from 0 to 1,

- Where 0 depicts that all the elements are allied to a certain class, or only one class exists there.
- The Gini index of value 1 signifies that all the elements are randomly distributed across various classes, and
- A value of 0.5 denotes the elements are uniformly distributed into some classes.

Mathematically, we can write Gini Impurity as follows:

$$Gini = 1 - \sum_{i=1}^{n} (p_i)^2$$

where pi is the probability of an object being classified to a particular class.

CART Algorithm

Classification and Regression Trees (CART) is a decision tree algorithm that is used for both classification and regression tasks. It is a supervised learning algorithm that learns from labelled data to predict unseen data.

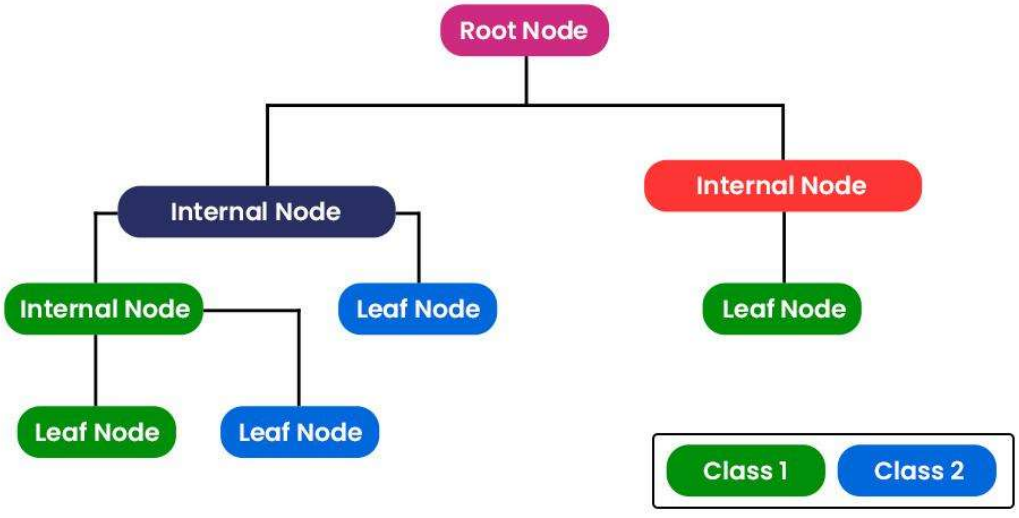
- **Tree structure:** CART builds a tree-like structure consisting of nodes and branches. The nodes represent different decision points, and the branches represent the possible outcomes of those decisions. The leaf nodes in the tree contain a predicted class label or value for the target variable.
- **Splitting criteria:** CART uses a greedy approach to split the data at each node. It evaluates all possible splits and selects the one that best reduces the impurity of the resulting subsets. For classification tasks, CART uses Gini impurity as the splitting criterion. The lower the Gini impurity, the more pure the subset is. For regression tasks, CART uses residual reduction as the splitting criterion. The lower the residual reduction, the better the fit of the model to the data.

- **Pruning:** To prevent overfitting of the data, pruning is a technique used to remove the nodes that contribute little to the model accuracy. Cost complexity pruning and information gain pruning are two popular pruning techniques. Cost complexity pruning involves calculating the cost of each node and removing nodes that have a negative cost. Information gain pruning involves calculating the information gain of each node and removing nodes that have a low information gain.

How does CART algorithm works?

The CART algorithm works via the following process:

- The best-split point of each input is obtained.
- Based on the best-split points of each input in Step 1, the new "best" split point is identified.
- Split the chosen input according to the "best" split point.
- Continue splitting until a stopping rule is satisfied or no further desirable splitting is available.



CART algorithm uses Gini Impurity to split the dataset into a decision tree .It does that by searching for the best homogeneity for the sub nodes, with the help of the Gini index criterion.

Data set

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Gini index

Gini index is a metric for classification tasks in CART. It stores sum of squared probabilities of each class. We can formulate it as illustrated below.

Gini = 1 – Σ $(Pi)^2$ for i=1 to number of classes

Outlook

Outlook is a nominal feature. It can be sunny, overcast or rain. I will summarize the final decisions for outlook feature.

Outlook	Yes	No	Number of instances
Sunny	2	3	5
Overcast	4	0	4
Rain	3	2	5

Gini(Outlook=Sunny) = 1 – $(2/5)^2$ – $(3/5)^2$ = 1 – 0.16 – 0.36 = 0.48

Gini(Outlook=Overcast) = 1 – $(4/4)^2$ – $(0/4)^2$ = 0

Gini(Outlook=Rain) = 1 – $(3/5)^2$ – $(2/5)^2$ = 1 – 0.36 – 0.16 = 0.48

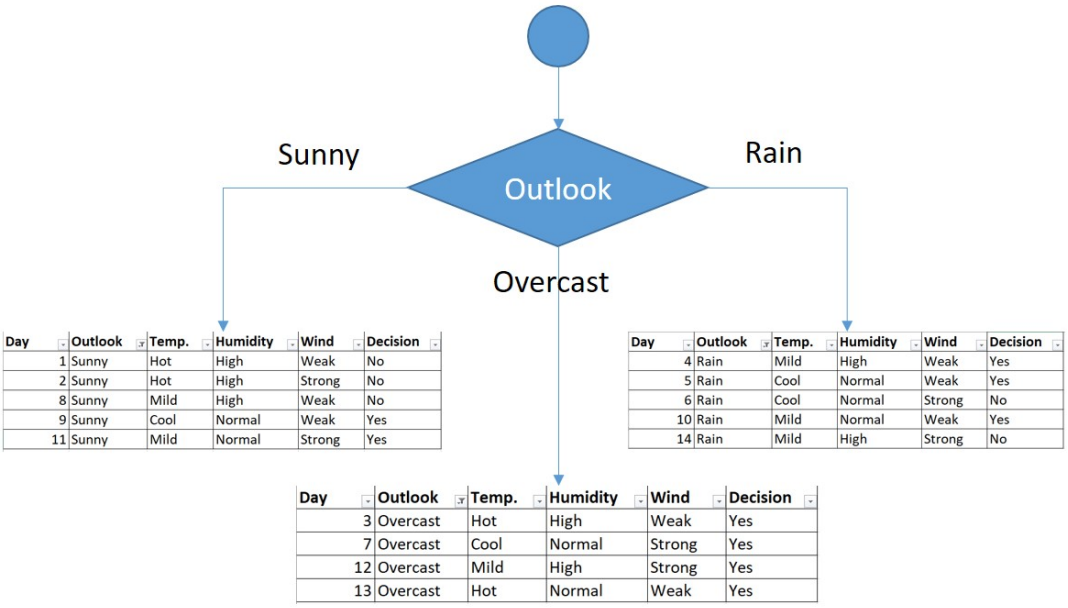
Then, we will calculate weighted sum of gini indexes for outlook feature.

Gini(Outlook) = (5/14) x 0.48 + (4/14) x 0 + (5/14) x 0.48 = 0.171 + 0 + 0.171 = 0.342

Similarly calculate index for humidity,Temperature and wind. We've calculated gini index values for each feature. The winner will be outlook feature because its cost is the lowest.

Feature	Gini index
Outlook	0.342
Temperature	0.439
Humidity	0.367
Wind	0.428

We'll put outlook decision at the top of the tree.

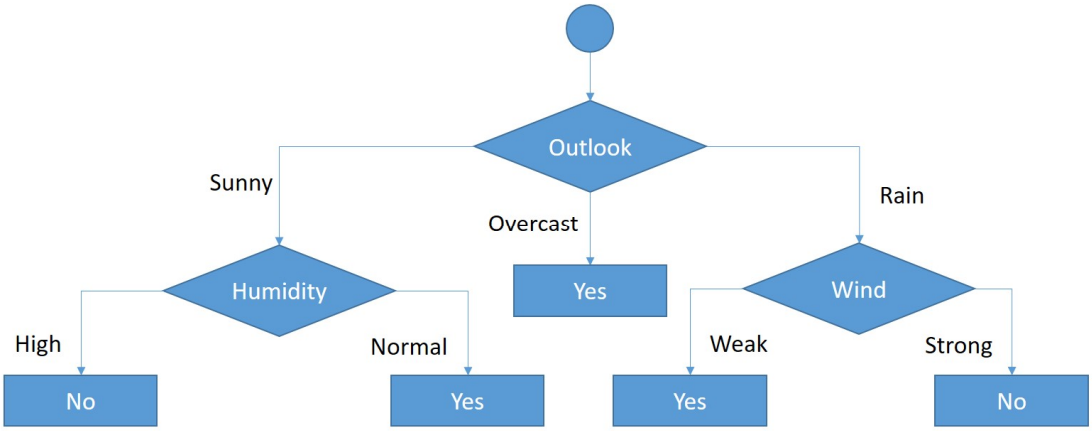


Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes

Day	Outlook	Temp.	Humidity	Wind	Decision
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
10	Rain	Mild	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Day	Outlook	Temp.	Humidity	Wind	Decision
3	Overcast	Hot	High	Weak	Yes
7	Overcast	Cool	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes

We will apply same principles to those sub datasets in the following steps. We end the algotithm when no more subset is possible.



Advantages of CART

- Results are simplistic.
- Classification and regression trees are Nonparametric and Nonlinear.
- Classification and regression trees implicitly perform feature selection.
- Outliers have no meaningful effect on CART.
- It requires minimal supervision and produces easy-to-understand models.

Limitations of CART

- Overfitting.
- High Variance.
- low bias.
- the tree structure may be unstable.

Applications of the CART algorithm

- For quick Data insights.
- In Blood Donors Classification.
- For environmental and ecological data.
- In the financial sectors.

Bayesian Classification

Bayesian classification uses Bayes theorem to predict the occurrence of any event. Bayesian classifiers are the statistical classifiers with the Bayesian

probability understandings. The theory expresses how a level of belief, expressed as a probability.

Bayes' Theorem is named after Thomas Bayes. He first makes use of conditional probability to provide an algorithm which uses evidence to calculate limits on an unknown parameter. Bayes' Theorem has two types of probabilities :

1. Prior Probability [P(H)]
2. Posterior Probability [P(H/X)]

Where,

- **X** – X is a data tuple.
- **H** – H is some Hypothesis.

1. Prior Probability

Prior Probability is the probability of occurring an event before the collection of new data. It is the best logical evaluation of the probability of an outcome which is based on the present knowledge of the event before the inspection is performed.

2. Posterior Probability

When new data or information is collected then the Prior Probability of an event will be revised to produce a more accurate measure of a possible outcome. This revised probability becomes the Posterior Probability and is calculated using Bayes' theorem. So, the Posterior Probability is the probability of an event **X** occurring given that event **H** has occurred.

Formula

Bayes' Theorem, can be mathematically represented by the equation given below :

$P(H/X)= P(X/H)P(H) / P(X)$

Where,

- **H** and **X** are the events and,
- **P (X) ≠ 0**
- **P(H/X) –** Conditional probability of H.
  Given that X occurs.
- **P(X/H) –** Conditional probability of X.
  Given that H occurs.
- **P(H) and P(X) –** Prior Probabilities of occurring H and X independent of each other.This is called the marginal probability.

Bayesian Belief Network

Bayesian Belief Networks specify joint conditional probability distributions. They are also known as Belief Networks, Bayesian Networks, or Probabilistic Networks.

- A Belief Network allows class conditional independencies to be defined between subsets of variables.
- It provides a graphical model of causal relationship on which learning can be performed.
- We can use a trained Bayesian Network for classification.

There are two components that define a Bayesian Belief Network −

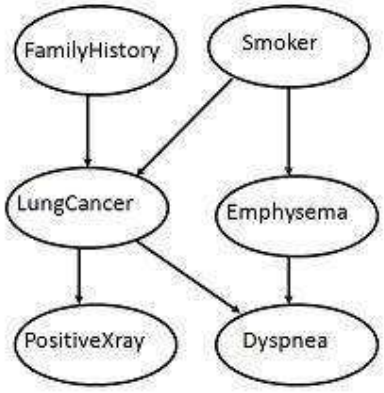
- Directed acyclic graph
- A set of conditional probability tables

Directed Acyclic Graph

- Each node in a directed acyclic graph represents a random variable.
- These variable may be discrete or continuous valued.
- These variables may correspond to the actual attribute given in the data.

Directed Acyclic Graph Representation

The following diagram shows a directed acyclic graph for six Boolean variables.



The arc in the diagram allows representation of causal knowledge. For example, lung cancer is influenced by a person's family history of lung cancer, as well as whether or not the person is a smoker. It is worth noting that the variable PositiveXray is independent of whether the patient has a family history of lung cancer or that the patient is a smoker, given that we know the patient has lung cancer.

Conditional Probability Table

The conditional probability table for the values of the variable LungCancer (LC) showing each possible combination of the values of its parent nodes, FamilyHistory (FH), and Smoker (S) is as follows –

	FH,S	FH,-S	-FH,S	-FH,S
LC	0.8	0.5	0.7	0.1
-LC	0.2	0.5	0.3	0.9

Naive bayesian classification:

Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

Why is it called Naïve Bayes?

The Naïve Bayes algorithm is comprised of two words Naïve and Bayes, Which can be described as:

- **Naïve**: It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other features. Such as if the fruit is identified on the bases of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple. Hence each feature individually contributes to identify that it is an apple without depending on each other.
- **Bayes**: It is called Bayes because it depends on the principle of Bayes' Theorem.

Bayes' Theorem:

- Bayes' theorem is also known as **Bayes' Rule** or **Bayes' law**, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.
- The formula for Bayes' theorem is given as:

$$P(A|B)=\frac{P(B|A)P(A)}{P(B)}$$

Where,

P(A|B) is Posterior probability: Probability of hypothesis A on the observed event B.

P(B|A) is Likelihood probability: Probability of the evidence given that the probability of a hypothesis is true.

P(A) is Prior Probability: Probability of hypothesis before observing the evidence.

P(B) is Marginal Probability: Probability of Evidence.

Working of Naïve Bayes' Classifier:

1. Convert the given dataset into frequency tables.
2. Generate Likelihood table by finding the probabilities of given features.
3. Now, use Bayes theorem to calculate the posterior probability.

Problem: If the weather is sunny, then the Player should play or not?

Solution: To solve this, first consider the below dataset:

	Outlook	Play
0	Rainy	Yes
1	Sunny	Yes
2	Overcast	Yes
3	Overcast	Yes
4	Sunny	No
5	Rainy	Yes
6	Sunny	Yes
7	Overcast	Yes
8	Rainy	No
9	Sunny	No
10	Sunny	Yes
11	Rainy	No
12	Overcast	Yes
13	Overcast	Yes

Frequency table for the Weather Conditions:

Weather	Yes	No
Overcast	5	0
Rainy	2	2
Sunny	3	2
Total	10	5

Likelihood table weather condition:

Weather	No	Yes	
Overcast	0	5	5/14= 0.35
Rainy	2	2	4/14=0.29
Sunny	2	3	5/14=0.35
All	4/14=0.29	10/14=0.71	

Applying Bayes'theorem:

P(Yes|Sunny)= P(Sunny|Yes)*P(Yes)/P(Sunny)

P(Sunny|Yes)= 3/10= 0.3

P(Sunny)= 0.35

P(Yes)=0.71

So P(Yes|Sunny) = 0.3*0.71/0.35= 0.60

P(No|Sunny)= P(Sunny|No)*P(No)/P(Sunny)

P(Sunny|NO)= 2/4=0.5

P(No)= 0.29

P(Sunny)= 0.35

So P(No|Sunny)= 0.5*0.29/0.35 = 0.41

So as we can see from the above calculation that P(Yes|Sunny)>P(No|Sunny)
Hence on a Sunny day, Player can play the game.

Rule Based Classification

IF-THEN Rules

Rule-based classifier makes use of a set of IF-THEN rules for classification. We can express a rule in the following from −

IF condition THEN conclusion

Let us consider a rule R1,

R1: IF age = youth AND student = yes
THEN buy_computer = yes

- The IF part of the rule is called **rule antecedent** or **precondition**.
- The THEN part of the rule is called **rule consequent**.
- The antecedent part the condition consist of one or more attribute tests and these tests are logically ANDed.
- The consequent part consists of class prediction. **Assessment of Rule**
- In rule-based classification in data mining, there are two factors based on which we can access the rules. These are:

- **Coverage of Rule:** The fraction of the records which satisfy the antecedent conditions of a particular rule is called the coverage of that rule.
  We can calculate this by dividing the number of records satisfying the rule(n1) by the total number of records(n).
  Coverage(R)=n1/n

- **Accuracy of a rule:** The fraction of the records that satisfy the antecedent conditions and meet the consequent values of a rule is called the accuracy of that rule.
  We can calculate this by dividing the number of records satisfying the consequent values(n2) by the number of records satisfying the rule(n1).
  Accuracy(R) = n2/n1

Rule Extraction

Here we will learn how to build a rule-based classifier by extracting IF-THEN rules from a decision tree.

To extract a rule from a decision tree –

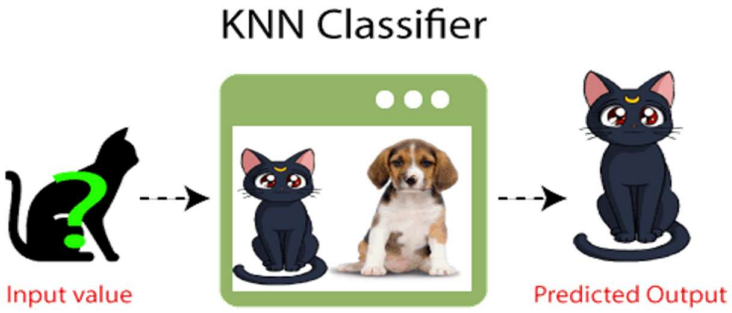
- One rule is created for each path from the root to the leaf node.
- To form a rule antecedent, each splitting criterion is logically ANDed.
- The leaf node holds the class prediction, forming the rule consequent

Lazy Learners: Lazy Learners are also known as instance-based learners, lazy learners do not learn a model during the training phase. Instead, they simply store the training data and use it to classify new instances at prediction time. It is very fast at prediction time because it does not require computations during the predictions. it is less effective in high-dimensional spaces or when the number of training instances is large. Examples of lazy learners include k-nearest neighbors and case-based reasoning.

K-Nearest Neighbor(KNN) Algorithm

- K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

- K-NN is a **non-parametric algorithm**, which means it does not make any assumption on underlying data.
- It is also called a **lazy learner algorithm** because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.
- **Example:** Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category.



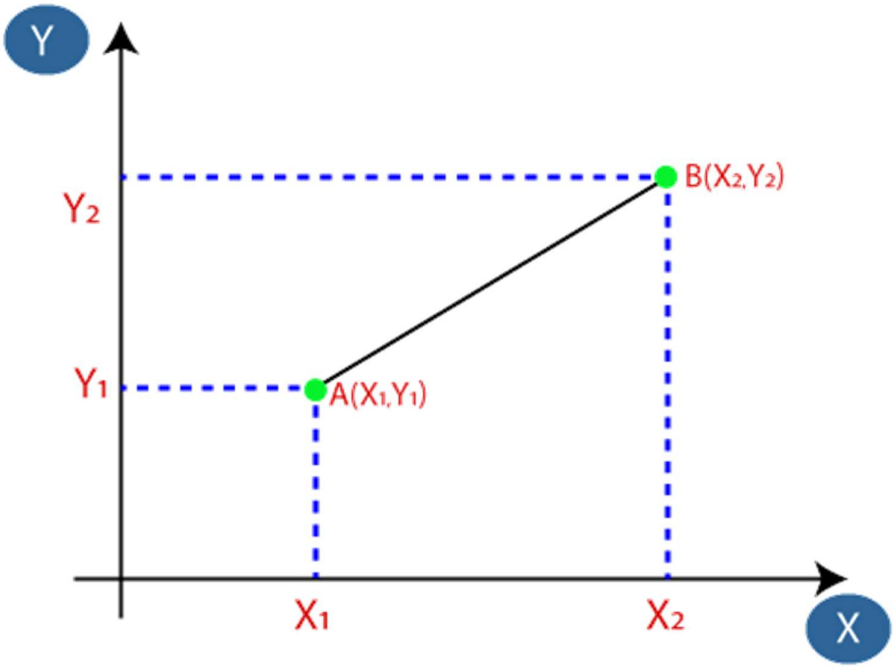
How does K-NN work?

The K-NN working can be explained on the basis of the below algorithm:

- **Step-1:** Select the number K of the neighbours
- **Step-2:** Calculate the Euclidean distance of **K number of neighbours**
- **Step-3:** Take the K nearest neighbours as per the calculated Euclidean distance.
- **Step-4:** Among these k neighbours, count the number of the data points in each category.
- **Step-5:** Assign the new data points to that category for which the number of the neighbour is maximum.
- **Step-6:** Our model is ready.

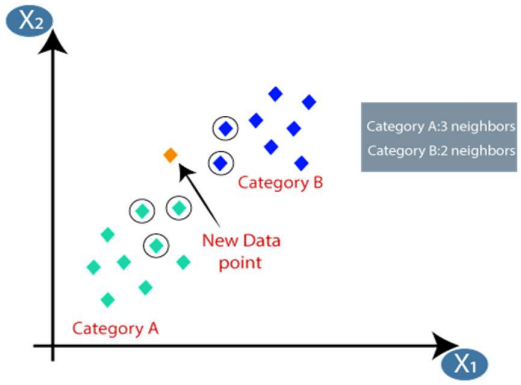
Suppose we have a new data point and we need to put it in the required category. Consider the below image:

- Firstly, we will choose the number of neighbours, so we will choose the k=5.
- Next, we will calculate the **Euclidean distance** between the data points. The Euclidean distance is the distance between two points, which we have already studied in geometry. It can be calculated as:



Euclidean Distance between A_1 and B_2 = $\sqrt{(X_2-X_1)^2+(Y_2-Y_1)^2}$

- By calculating the Euclidean distance we got the nearest neighbors, as three nearest neighbors in category A and two nearest neighbors in category B. Consider the below image:



- As we can see the 3 nearest neighbors are from category A, hence this new data point must belong to category A.

Advantages of KNN Algorithm:

- It is simple to implement.
- It is robust to the noisy training data
- It can be more effective if the training data is large.

Disadvantages of KNN Algorithm:

- Always needs to determine the value of K which may be complex some time.
- The computation cost is high because of calculating the distance between the data points for all the training samples.

What is Prediction?

Another process of data analysis is prediction. It is used to find a numerical output. Same as in classification, the training dataset contains the inputs and corresponding numerical output values. The algorithm derives the model or a predictor according to the training dataset. The model should find a numerical output when the new data is given. Unlike in classification, this method does not have a class label. The model predicts a continuous-valued function or ordered value.

Regression is generally used for prediction. Predicting the value of a house depending on the facts such as the number of rooms, the total area, etc., is an example for prediction.

For example, suppose the marketing manager needs to predict how much a particular customer will spend at his company during a sale. We are bothered to forecast a numerical value in this case. Therefore, an example of numeric prediction is the data processing activity. In this case, a model or a predictor will be developed that forecasts a continuous or ordered value function.

What is accuracy?

Accuracy is a metric that measures how often a machine learning model correctly predicts the outcome. You can calculate accuracy by dividing the number of correct predictions by the total number of predictions.

In other words, accuracy answers the question: how often the model is right?

What is precision?

Precision is a metric that measures how often a machine learning model correctly predicts the positive class. You can calculate precision by dividing the number of correct positive predictions (true positives) by the total number of instances the model predicted as positive (both true and false positives).

In other words, precision answers the question: how often the positive predictions are correct?

What is recall?

Recall is a metric that measures how often a machine learning model correctly identifies positive instances (true positives) from all the actual positive samples in the dataset. You can calculate recall by dividing the number of true positives by the number of positive instances. The latter includes true positives (successfully identified cases) and false negative results (missed cases).

In other words, recall answers the question: can an ML model find all instances of the positive class?

UNIT V
CLUSTERING

CLUSTER ANALYSIS:

Cluster analysis, also known as clustering, is a method of data mining that groups similar data points together. The goal of cluster analysis is to divide a dataset into groups (or clusters) such that the data points within each group are more similar to each other than to data points in other groups.

Cluster Analysis is the process to find similar groups of objects in order to form clusters. It is an unsupervised machine learning-based algorithm that acts on unlabelled data. A group of data points would comprise together to form a cluster in which all the objects would belong to the same group.

For example, consider a dataset of vehicles given in which it contains information about different vehicles like cars, buses, bicycles, etc. As it is unsupervised learning there are no class labels like Cars, Bikes, etc for all the vehicles, all the data is combined and is not in a structured manner.

Properties of Clustering :

1. Clustering Scalability: Nowadays there is a vast amount of data and should be dealing with huge databases. In order to handle extensive databases, the clustering algorithm should be scalable to get appropriate results.

2. High Dimensionality: The algorithm should be able to handle high dimensional space along with the data of small size.

3. Algorithm Usability with multiple data kinds: Different kinds of data can be used with algorithms of clustering. It should be capable of dealing with different types of data like discrete, categorical and interval-based data, binary data etc.

4. Dealing with unstructured data: There would be some databases that contain missing values, and noisy or erroneous data. If the algorithms are sensitive to such data then it may lead to poor quality clusters. So it should be able to handle unstructured data and give some structure to the data by organising it into groups of similar data objects. This makes the job of the data expert easier in order to process the data and discover new patterns.

5. Interpretability: The clustering outcomes should be interpretable, comprehensible, and usable. The interpretability reflects how easily the data is understood.

Applications Of Cluster Analysis:

- It is widely used in image processing, data analysis, and pattern recognition.
- It helps marketers to find the distinct groups in their customer base and they can characterize their customer groups by using purchasing patterns.

- It can be used in the field of biology, by deriving animal and plant taxonomies and identifying genes with the same capabilities.
- It also helps in information discovery by classifying documents on the web.

Advantages of Cluster Analysis:

1) It can help identify patterns and relationships within a dataset that may not be immediately obvious.\
2) It can be used for exploratory data analysis and can help with feature selection.
3) It can be used to reduce the dimensionality of the data.
4) It can be used for anomaly detection and outlier identification.
5) It can be used for market segmentation and customer profiling.

Disadvantages of Cluster Analysis:

1. It can be sensitive to the choice of initial conditions and the number of clusters.
2. It can be sensitive to the presence of noise or outliers in the data.
3. It can be difficult to interpret the results of the analysis if the clusters are not well-defined.
4. It can be computationally expensive for large datasets.
5. The results of the analysis can be affected by the choice of clustering algorithm used.
6. It is important to note that the success of cluster analysis depends on the data, the goals of the analysis, and the ability of the analyst to interpret the results.

Partitioning Method

It is used to make partitions on the data in order to form clusters. If "n" partitions are done on "p" objects of the database then each partition is represented by a cluster and n < p. The two conditions which need to be satisfied with this Partitioning Clustering Method are:

- One objective should only belong to only one group.
- There should be no group without even a single purpose.

In the partitioning method, there is one technique called iterative relocation, which means the object will be moved from one group to another to improve the partitioning. There are many algorithms that come under partitioning method some of the popular ones are K-Mean, PAM(K-Medoids), CLARA algorithm (Clustering Large Applications) etc.

Algorithm: K mean:

- The K means algorithm takes the input parameter K from the user and partitions the dataset containing N objects into K clusters so that resulting similarity among the data objects inside the group (intracluster) is high but the similarity of data objects with the data objects from outside the cluster is low (intercluster).
- The similarity of the cluster is determined with respect to the mean value of the cluster. It is a type of square error algorithm.
- At the start randomly k objects from the dataset are chosen in which each of the objects represents a cluster mean(centre).
- For the rest of the data objects, they are assigned to the nearest cluster based on their distance from the cluster mean.
- The new mean of each of the cluster is then calculated with the added data objects.

Algorithm:

Input:

K: The number of clusters in which the dataset has to be divided

D: A dataset containing N number of objects

Output:

A dataset of K clusters

Method:

1. Randomly assign K objects from the dataset(D) as cluster centres(C)
2. (Re) Assign each object to which object is most similar based upon mean values.
3. Update Cluster means, i.e., Recalculate the mean of each cluster with the updated values.
4. Repeat Step 2 until no change occurs.

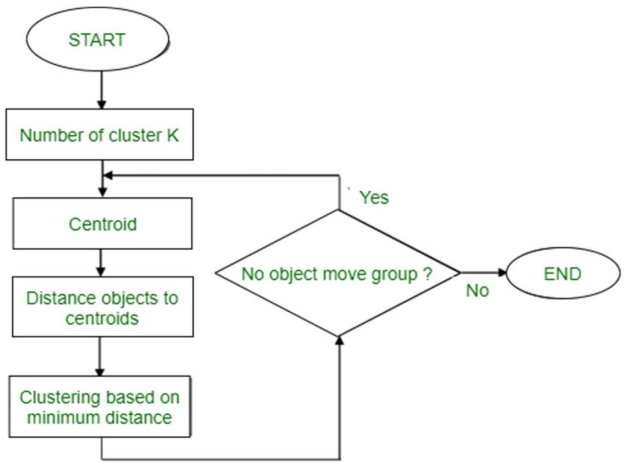


Figure – K-mean Clustering**Example:** Suppose we want to group the visitors to a website using just their age as follows:
16, 16, 17, 20, 20, 21, 21, 22, 23, 29, 36, 41, 42, 43, 44, 45, 61, 62, 66
Initial Cluster:
K=2
Centroid(C1) = 16 [16]
Centroid(C2) = 22 [22]
Note: These two points are chosen randomly from the dataset. **Iteration-1:**
C1 = 16.33 [16, 16, 17]
C2 = 37.25 [20, 20, 21, 21, 22, 23, 29, 36, 41, 42, 43, 44, 45, 61, 62, 66]
Iteration-2:
C1 = 19.55 [16, 16, 17, 20, 20, 21, 21, 22, 23]
C2 = 46.90 [29, 36, 41, 42, 43, 44, 45, 61, 62, 66]
Iteration-3:
C1 = 20.50 [16, 16, 17, 20, 20, 21, 21, 22, 23, 29]
C2 = 48.89 [36, 41, 42, 43, 44, 45, 61, 62, 66]
Iteration-4:
C1 = 20.50 [16, 16, 17, 20, 20, 21, 21, 22, 23, 29]
C2 = 48.89 [36, 41, 42, 43, 44, 45, 61, 62, 66]
No change Between Iteration 3 and 4, so we stop. Therefore we get the clusters **(16-29)** and **(36-66)** as 2 clusters we get using K Mean Algorithm.

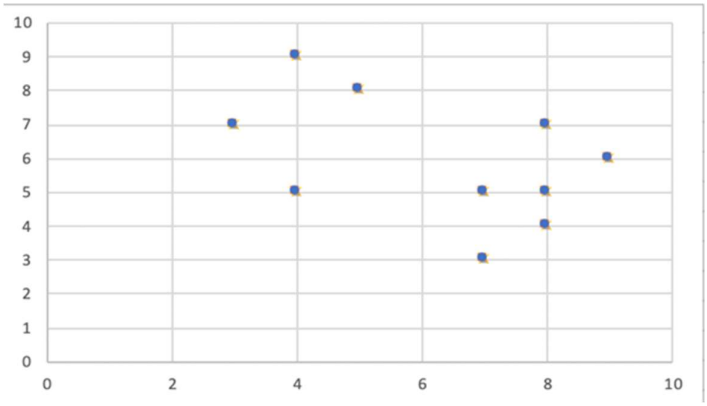
K-Medoids clustering with solved example

K-Medoids (also called Partitioning Around Medoid) algorithm was proposed in 1987 by Kaufman and Rousseeuw. A medoid can be defined as a point in the cluster, whose dissimilarities with all the other points in the cluster are minimum. The dissimilarity of the medoid(Ci) and object(Pi) is calculated by using
$E = |Pi - Ci|$
Algorithm:

1. Initialize: select *k* random points out of the *n* data points as the medoids.
2. Associate each data point to the closest medoid by using any common distance metric methods.
3. While the cost decreases: For each medoid m, for each data o point which is not a medoid:
    - Swap m and o, associate each data point to the closest medoid, and recompute the cost.
    - If the total cost is more than that in the previous step, undo the swap.

	X	Y
0	8	7
1	3	7
2	4	9
3	9	6
4	8	5
5	5	8
6	7	3
7	8	4
8	7	5
9	4	5

Let's consider the following example: If a graph is drawn using the above data points, we obtain the following:



Step 1: Let the randomly selected 2 medoids, so select k = 2, and let **C1 -(4, 5)** and **C2 -(8, 5)** are the two medoids.

Step 2: Calculating cost. The dissimilarity of each non-medoid point with the medoids is calculated and tabulated:

	X	Y	Dissimilarity from C1	Dissimilarity from C2
0	8	7	6	2
1	3	7	3	7
2	4	9	4	8
3	9	6	6	2
4	8	5	-	-
5	5	8	4	6
6	7	3	5	3
7	8	4	5	1
8	7	5	3	1
9	4	5	-	-

Here we have used Manhattan distance formula to calculate the distance matrices between medoid and non-medoid points. That formula tell that

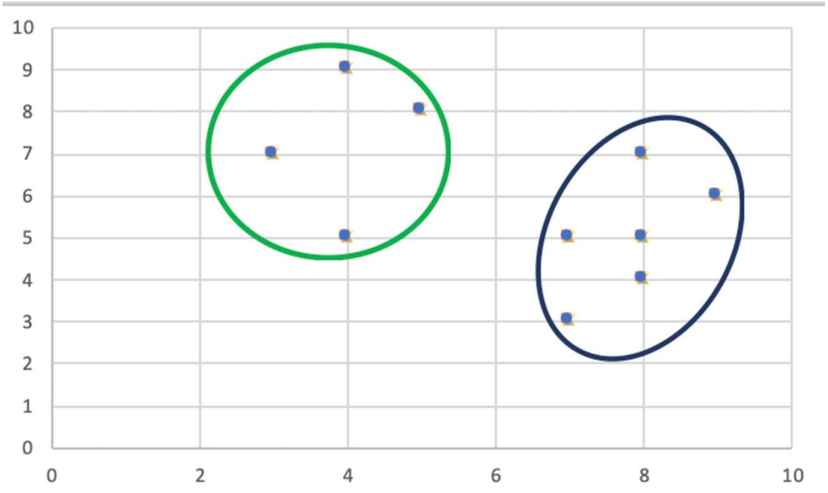
Distance = |X1-X2| + |Y1-Y2|.

Each point is assigned to the cluster of that medoid whose dissimilarity is less. Points 1, 2, and 5 go to cluster C1 and 0, 3, 6, 7, 8 go to cluster C2. The Cost = (3 + 4 + 4) + (3 + 1 + 1 + 2 + 2) = 20

Step 3: randomly select one non-medoid point and recalculate the cost. Let the randomly selected point be (8, 4). The dissimilarity of each non-medoid point with the medoids – C1 (4, 5) and C2 (8, 4) is calculated and tabulated.

	X	Y	Dissimilarity from C1	Dissimilarity from C2
0	8	7	6	3
1	3	7	3	8
2	4	9	4	9
3	9	6	6	3
4	8	5	4	1
5	5	8	4	7
6	7	3	5	2
7	8	4	-	-
8	7	5	3	2
9	4	5	-	-

Each point is assigned to that cluster whose dissimilarity is less. So, points 1, 2, and 5 go to cluster C1 and 0, 3, 6, 7, 8 go to cluster C2. The New cost = (3 + 4 + 4) + (2 + 2 + 1 + 3 + 3) = 22 Swap Cost = New Cost – Previous Cost = 22 – 20 and **2 >0** As the swap cost is not less than zero, we undo the swap. Hence (4, 5) and (8, 5) are the final medoids.



CLARA (Clustering Large Applications)ALGORITHM is an extension to k-medoids (PAM) methods to deal with data containing a large number of objects

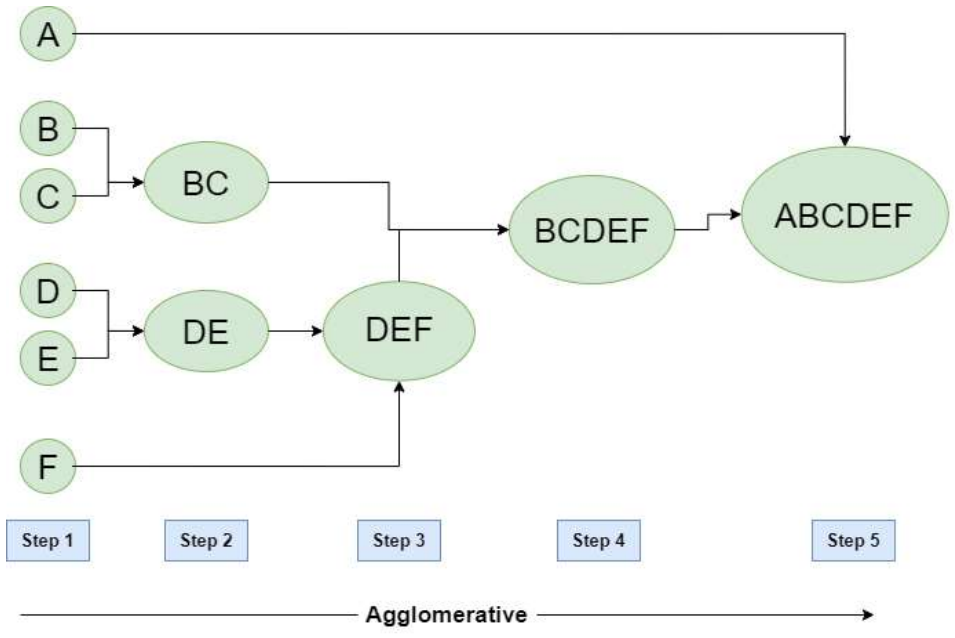
(more than several thousand observations) in order to reduce computing time and RAM storage problem.

The algorithm is as follow:

1. Create randomly, from the original dataset, multiple subsets with fixed size (sampsize)
2. Compute PAM algorithm on each subset and choose the corresponding k representative objects (medoids). Assign each observation of the entire data set to the closest medoid.
3. Calculate the mean (or the sum) of the dissimilarities of the observations to their closest medoid. This is used as a measure of the goodness of the clustering.
4. Retain the sub-dataset for which the mean (or sum) is minimal. A further analysis is carried out on the final partition.

Hierarchical Method: In this method, a hierarchical decomposition of the given set of data objects is created. We can classify hierarchical methods and will be able to know the purpose of classification on the basis of how the hierarchical decomposition is formed. There are two types of approaches for the creation of hierarchical decomposition, they are:

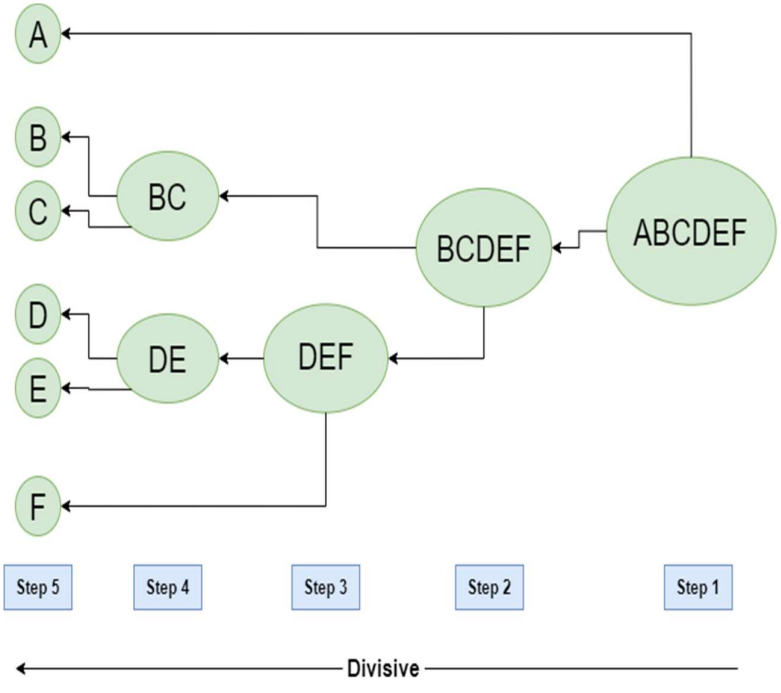
- **Agglomerative Approach:** The agglomerative approach is also known as the bottom-up approach. Initially, the given data is divided into which objects form separate groups. Thereafter it keeps on merging the objects or the groups that are close to one another which means that they exhibit similar properties. This merging process continues until the termination condition holds.



Steps:

- Consider each alphabet as a single cluster and calculate the distance of one cluster from all the other clusters.
- In the second step, comparable clusters are merged together to form a single cluster. Let's say cluster (B) and cluster (C) are very similar to each other therefore we merge them in the second step similarly to cluster (D) and (E) and at last, we get the clusters [(A), (BC), (DE), (F)]
- We recalculate the proximity according to the algorithm and merge the two nearest clusters([(DE), (F)]) together to form new clusters as [(A), (BC), (DEF)]
- Repeating the same process; The clusters DEF and BC are comparable and merged together to form a new cluster. We're now left with clusters [(A), (BCDEF)].
- At last, the two remaining clusters are merged together to form a single cluster [(ABCDEF)].

- **Divisive Approach:** The divisive approach is also known as the top-down approach. In this approach, we would start with the data objects that are in the same cluster. The group of individual clusters is divided into small clusters by continuous iteration. The iteration continues until the condition of termination is met or until each cluster contains one object.



Once the group is split or merged then it can never be undone as it is a rigid method and is not so flexible. The two approaches which can be used to improve the Hierarchical Clustering Quality in Data Mining are: –

- One should carefully analyze the linkages of the object at every partitioning of hierarchical clustering.
- One can use a hierarchical agglomerative algorithm for the integration of hierarchical agglomeration. In this approach, first, the objects are grouped into micro-clusters. After grouping data objects into microclusters, macro clustering is performed on the microcluster.

Density-Based Method: The density-based method mainly focuses on density. In this method, the given cluster will keep on growing continuously as long as the density in the neighbourhood exceeds some threshold, i.e, for each data point within a given cluster. The radius of a given cluster has to contain at least a minimum number of points.

DBSCAN Algorithm

Parameters Required For DBSCAN Algorithm

Density-Based Clustering - Background

There are two different parameters to calculate the density-based clustering

EPS: It is considered as the maximum radius of the neighborhood.

MinPts: MinPts refers to the minimum number of points in an Eps neighborhood of that point.

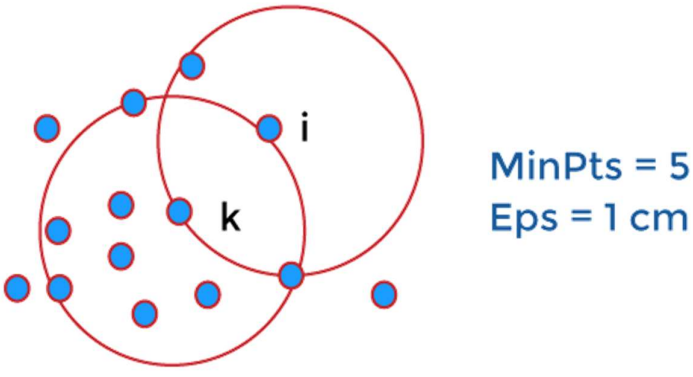
NEps (i) : { k belongs to D and dist (i,k) < = Eps}

Directly density reachable:

A point i is considered as the directly density reachable from a point k with respect to Eps, MinPts if i belongs to NEps(k)

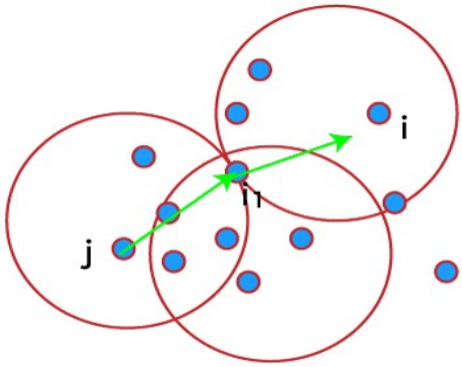
Core point condition:

NEps (k) >= MinPts



Density reachable:

A point denoted by i is a density reachable from a point j with respect to Eps, MinPts if there is a sequence chain of a point i1,…., in, i1 = j, pn = i such that i_i + 1 is directly density reachable from i_i.



Working of Density-Based Clustering

Suppose a set of objects is denoted by D', we can say that an object I is directly density reachable form the object j only if it is located within the ε neighborhood of j, and j is a core object.

An object i is density reachable form the object j with respect to ε and MinPts in a given set of objects, D' only if there is a sequence of object chains point i1,…., in, i1 = j, pn = i such that i_i + 1 is directly density reachable from i_i with respect to ε and MinPts.

An object i is density connected object j with respect to ε and MinPts in a given set of objects, D' only if there is an object o belongs to D such that both point i and j are density reachable from o with respect to ε and MinPts.

Major Features of Density-Based Clustering

The primary features of Density-based clustering are given below.

- It is a scan method.
- It requires density parameters as a termination condition.
- It is used to manage noise in data clusters.
- Density-based clustering is used to identify clusters of arbitrary size.

Density-Based Clustering Methods

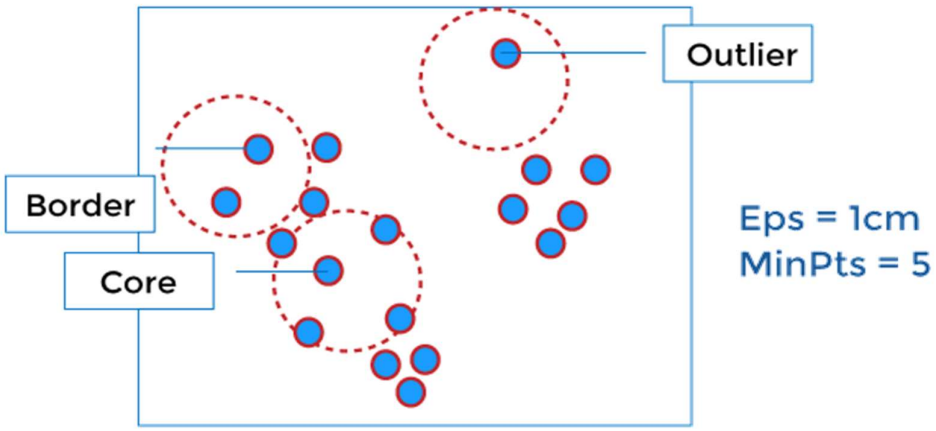
DBSCAN

DBSCAN stands for Density-Based Spatial Clustering of Applications with Noise. It depends on a density-based notion of cluster. It also identifies clusters of arbitrary size in the spatial database with outliers.

Steps Used In DBSCAN Algorithm

1. Find all the neighbor points within eps and identify the core points or visited with more than MinPts neighbors.

2. For each core point if it is not already assigned to a cluster, create a new cluster.
3. Find recursively all its density-connected points and assign them to the same cluster as the core point. A point *a* and *b* are said to be density connected if there exists a point *c* which has a sufficient number of points in its neighbors and both points *a* and *b* are within the *eps distance*. This is a chaining process. So, if *b* is a neighbor of *c*, *c* is a neighbor of *d*, and *d* is a neighbor of *e*, which in turn is neighbor of *a* implying that *b* is a neighbor of *a*.
4. Iterate through the remaining unvisited points in the dataset. Those points that do not belong to any cluster are noise.



OPTICS

OPTICS stands for Ordering Points To Identify the Clustering Structure. It gives a significant order of database with respect to its density-based clustering structure. The order of the cluster comprises information equivalent to the density-based clustering related to a long range of parameter settings. OPTICS methods are beneficial for both automatic and interactive cluster analysis, including determining an intrinsic clustering structure.

DENCLUE

Density-based clustering by Hinnebirg and Kiem. It enables a compact mathematical description of arbitrarily shaped clusters in high dimension state of data, and it is good for data sets with a huge amount of noise.

Grid-Based Method: In the Grid-Based method a grid is formed using the object together, i.e., the object space is quantized into a finite number of cells that form

a grid structure. One of the major advantages of the grid-based method is fast processing time and it is dependent only on the number of cells in each dimension in the quantized space.

Statistical Information Grid(STING):

- A STING is a grid-based clustering technique. It uses a multidimensional grid data structure that quantifies space into a finite number of cells. Instead of focusing on data points, it focuses on the value space surrounding the data points.
- In STING, the spatial area is divided into rectangular cells and several levels of cells at different resolution levels. High-level cells are divided into several low-level cells.
- In STING Statistical Information about attributes in each cell, such as mean, maximum, and minimum values, are precomputed and stored as statistical parameters. These statistical parameters are useful for query processing and other data analysis tasks.

Working of CLIQUE Algorithm:

The CLIQUE algorithm first divides the data space into grids. It is done by dividing each dimension into equal intervals called units. After that, it identifies dense units. A unit is dense if the data points in this are exceeding the threshold value.

Once the algorithm finds dense cells along one dimension, the algorithm tries to find dense cells along two dimensions, and it works until all dense cells along the entire dimension are found.

After finding all dense cells in all dimensions, the algorithm proceeds to find the largest set ("cluster") of connected dense cells. Finally, the CLIQUE algorithm generates a minimal description of the cluster. Clusters are then generated from all dense subspaces using the apriori approach.

Model based Methods

Model-based clustering is a statistical approach to data clustering. The observed (multivariate) data is considered to have been created from a finite combination of component models. Each component model is a probability distribution, generally a parametric multivariate distribution.

There are the following types of model-based clustering are as follows −

1.Statistical approach − Expectation maximization is a popular iterative refinement algorithm. An extension to k-means −

- It can assign each object to a cluster according to weight (probability distribution).
- New means are computed based on weight measures.

The basic idea is as follows −

- It can start with an initial estimate of the parameter vector.
- It can be used to iteratively rescore the designs against the mixture density made by the parameter vector.
- It is used to rescored patterns are used to update the parameter estimates.
- It can be used to pattern belonging to the same cluster if they are placed by their scores in a particular component.

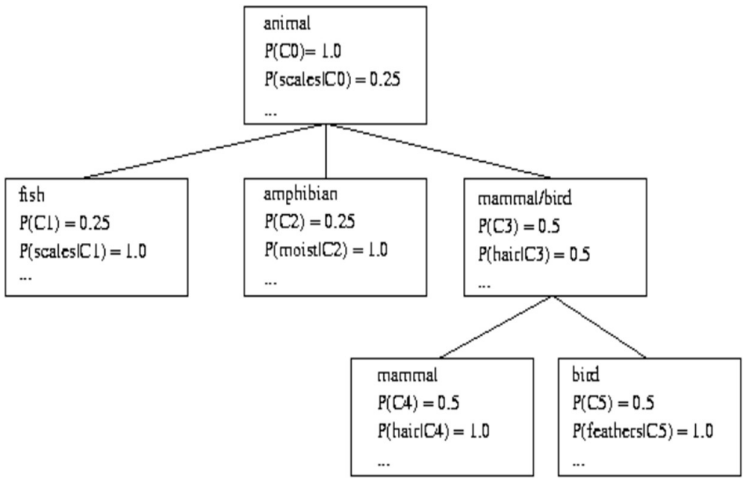
2.Conceptual clustering

- Conceptual clustering is a form of clustering in machine learning.
- It produces a classification scheme for a set of unlabeled objects and finds characteristic description for each concept (class).

COBWEB (Fisher'87)

COBWEB is a popular a simple method of incremental conceptual learning.
It creates a hierarchical clustering in the form of a classification tree.
Each node refers to a concept and contains a probabilistic description of that concept.

Classification Tree



3.Neural Network Approach − The neural network approach represents each cluster as an example, acting as a prototype of the cluster. The new objects are distributed to the cluster whose example is the most similar according to some distance measure.

Evaluation of clustering

According to their attributes and goals, many measures have therefore been developed to assess the effectiveness of clustering methods.
Many often employed metrics include −

- **Silhouette Score**

Based on its closeness to other data points in that cluster as well as to data points in other clusters, each data point's silhouette score evaluates how well it fits into the cluster to which it has been allocated. A score of 1 means the data point is well-clustered, whereas a value of -1 means it has been misclassified. The silhouette score goes from -1 to 1.

- **Calinski-Harabasz Index**
  A higher index value implies greater clustering performance. The Calinski-Harabasz index evaluates the ratio of between-cluster variation to within-cluster variance.
- **Davies-Bouldin index**
  A lower Davies-Bouldin index suggests greater clustering performance since it gauges the average similarity between each cluster and its most comparable cluster.
- **Rand Index**
  A higher Rand index denotes better clustering performance. It quantifies the similarity between the anticipated grouping and the ground truth clustering.
- **Adjusted Mutual Information (AMI)**
  A higher index implies greater clustering performance. The AMI evaluates the mutual information between the expected clustering and the ground truth clustering, corrected for the chance.

Choosing the Right Evaluation Metric

The nature and objectives of a clustering problem will dictate the most appropriate assessment measure to employ. If the goal of clustering is to group similar data points together, the Calinski-Harabasz index or the silhouette score can be beneficial. If the clustering results need to be compared to ground truth clustering, however, the Rand index or AMI would be more appropriate. So, it is important to consider the objectives and constraints of the clustering issue while selecting the evaluation metric.

Choosing appropriate distance metrics

Choosing an appropriate distance metric is a critical step in cluster analysis as it determines how similarity or dissimilarity is calculated between data points. The choice of distance metric should align with the characteristics of the data and the objectives of the clustering analysis. Here are some guidelines to consider when selecting distance metrics for cluster analysis:

- **Understand the nature of the data:** Consider the type of data you are working with. Is it numerical, categorical, binary, or a mix of different types? Different distance metrics are suitable for different types of data.

- **Euclidean distance:** Euclidean distance is commonly used for continuous or numerical data. It measures the straight-line distance between two points in Euclidean space. It assumes that all variables have equal importance and are on the same scale. Euclidean distance is widely used in algorithms like k-means and hierarchical clustering.
- **Manhattan distance:** Manhattan distance, also known as city block distance or L1 distance, calculates the sum of absolute differences between the coordinates of two points. It is appropriate for numerical data when the variables have different scales or represent different units. Manhattan distance is robust to outliers and is used in clustering algorithms like k-medians.
- **Minkowski distance:** Minkowski distance is a generalized distance metric that includes both Euclidean and Manhattan distances as special cases. It is defined as the nth root of the sum of the absolute values raised to the power of n. By varying the value of the parameter "n," different distance metrics can be obtained. When n=1, it is equivalent to Manhattan distance, and when n=2, it is equivalent to Euclidean distance.
- **Hamming distance:** Hamming distance is suitable for categorical or binary data. It measures the number of positions at which two strings of equal length differ. It is commonly used for clustering tasks involving text data, DNA sequences, or binary feature vectors.
- **Jaccard distance:** Jaccard distance is used to measure dissimilarity between sets. It is commonly used for binary or categorical data where presence or absence of items is of interest. Jaccard distance is defined as the ratio of the difference of the sizes of the intersection and union of two sets. It is often used in clustering tasks like text document clustering or item-based recommendation systems.

Selecting appropriate clustering algorithms

You also want to make sure that you select the appropriate clustering algorithms to align best with your data and objectives. Consider factors such as scalability, interpretability, and the ability to handle specific types of data (e.g., k-means for numerical data, DBSCAN for density-based clusters).

Evaluating cluster quality

Make sure you employ appropriate validation techniques to assess the quality of the clustering solution. Use both quantitative measures and visualizations to evaluate the cohesion, separation, and interpretability of the clusters. Some methods you can use include:

- **Silhouette Coefficient:** The silhouette coefficient measures how well each data point fits within its assigned cluster compared to other clusters.

It ranges from -1 to +1, with higher values indicating better-defined and well-separated clusters.

- **Davies-Bouldin Index:** The Davies-Bouldin index evaluates the compactness and separation of clusters. It calculates the average dissimilarity between each cluster and its most similar cluster, with lower values indicating better clustering.
- **Calinski-Harabasz Index:** The Calinski-Harabasz index quantifies the ratio of between-cluster dispersion to within-cluster dispersion. Higher values suggest well-separated and compact clusters.

Evaluating the Stability of Clustering Results

Clustering has certain challenges since the parameters of the algorithm and the initial conditions may affect the results. It is essential to execute the clustering technique repeatedly using multiple random initializations or settings in order to judge the sustainability of the clustering findings. One can evaluate the stability of the clustering results using metrics such as the Jaccard index or the variance of information.

Visualizing the Clustering Results

An understanding of the data's structure and patterns can be gained by visualizing the clustering findings. Using scatter plots or heat maps, where each data point is depicted as a point or a cell with a color-coded depending on its cluster assignment, is one approach to see the clustering findings. In order to project the high-dimensional data into a lower-dimensional space and show the clusters, dimensionality reduction techniques like principal component analysis (PCA) or t-SNE can be used. In addition, visualization tools like dendrograms or silhouette plots are frequently included in cluster analysis software packages allowing users to explore the clustering outcomes.