MEASI
Institute *of* Management

FUNDAMENTALS OF BUSINESS ANALYTICS

534E2A

LECTURE NOTES

PREPARED BY

DR. CATHERENE JULIE AARTHY.C

MEASI
Institute *of* Management

VISION & MISSION STATEMENTS

Vision: To be an oasis of knowledge to the seeker, to nurture one's creativity and research acumen, and to instil a unique blend of leadership, innovative spirit and empathy in response to the ever-evolving business ecosystem.

Mission

- Provide a pedagogy that blends academic rigor and experiential learning. (PEO1)
- Inculcate an entrepreneurial mindset through curated activities. (PEO2)
- Establish a conducive environment for research. (PEO3)
- Foster a culture of innovation and collaboration to progress in a dynamic business landscape. (PEO2, PEO4)
- Promote humanistic values to produce socially responsible leaders. (PEO5)

Program Educational Objectives (PEOs)

PEO 1 – Employability:

To develop students with industry specific knowledge & skills to meet the industry requirements and also join public sector undertaking through competitive examinations.

PEO 2 - Entrepreneur:

To create effective business service owners, with a growth mindset by enhancing their critical thinking, problem solving and decision-making skills.

PEO3 – Research and Development:

To instil and grow a mindset that focusses efforts towards inculcating and encouraging the students in the field research and development.

PEO 4 – Contribution to Business World:

To produce ethical and innovative business professionals to enhance growth of the business world.

PEO 5 – Contribution to the Society:

To work and contribute towards holistic development of society by producing competent MBA professionals.

LIST OF PROGRAM OUTCOMES

Regulation	**2023-2024**
Batch	**2023-2025**
PO1	**Problem Solving Skill:** Application of tools and techniques relevant to management theories and practices in analysing & solving business problems
PO2	**Decision Making Skill:** Fostering analytical and critical thinking abilities for data-based decision making
PO3	**Ethical Value:** Ability to develop value-based leadership attributes.
PO4	**Communication Skill:** Ability to understand, analyse and effectively communicate global, economic, legal and ethical aspects of business
PO5	**Individual and Team Leadership Skill:** Ability to be self-motivated in leading and driving a team towards achievement of organisational goals and contributing effectively to establish industrial harmony
PO6	**Employability Skill:** Foster and enhance employability skills through relevant industry subject knowledge
PO7	**Entrepreneurial Skill:** Equipped with skills and competencies to become a global entrepreneur
PO8	**Contribution to Society:** Strive towards becoming a global influencer and motivating future generations towards building a legacy that contributes to overall growth of humankind

Program Specific Outcomes (PSO)

PSO1: Finance: The students should demonstrate proficiency in analysing financial statements, evaluating investment opportunities and making financial decision to maximize shareholders' value.

PSO2: Marketing: Students should be able to create a comprehensive marketing plan that integrates effective communication strategies, leading to customer success and the accomplishment of marketing objectives.

PSO3: Logistics: Students will acquire knowledge of inventory management for domestic and global supply chains, thereby developing problem solving skills in logistics to optimise supply chain efficiency

PSO4: Business Analytics: The students should be able to analyse data, communicate insights, take data-driven decisions and solve business problems efficiency

SYLLABUS

Subject Code	Subject Name	Category	L	T	P	O	Credits	Inst.	Marks		
									CIA	External	Total
534E2A	**Fundamental of Business Analytics**	Elective	3	-	-	-	3	3	25	75	100

	Course Objectives
C1	To enable the students to understand the basics of Business Analytics
C2	To create awareness and understanding on visualizing data through collecting, managing and analyzing data.
C3	To educate the students on data mining and multi-dimensional data analysis
C4	To educate the students on machine learning and AI.
C5	To elucidate the students on the analysis of various areas of business

UNIT	Details	No. of Hours	Course Objectives
I	**Introduction to Business Analytics:** Meaning - Historical overview of data analysis – Data Scientist Vs Data Engineer Vs Business Analyst – Career in Business Analytics – Introduction to data science – Applications for data science – Roles and Responsibilities of data scientists	9	C1
II	**Data Visualization:** Data Collection - Data Management - Big Data Management - Organization/sources of data - Importance of data quality - Dealing with missing or incomplete data - Data Visualization - Data Classification Data Science Project Life Cycle: Business Requirement - Data Acquisition – Data Preparation - Hypothesis and Modeling - Evaluation and Interpretation, Deployment, Operations, Optimization.	9	C2
III	**Data Mining:** Introduction to Data Mining - The origins of Data Mining - Data Mining Tasks - OLAP and Multidimensional data analysis - Basic concept of Association Analysis and Cluster Analysis.	9	C3
IV	**Machine Learning:** Introduction to Machine Learning - History and Evolution - AI Evolution - Statistics Vs Data Mining Vs, Data Analytics Vs,	9	C4

	Data Science - Supervised Learning, Unsupervised Learning, Reinforcement Learning – Frame works for building Machine Learning Systems.		
V	**Application of Business Analysis:** Retail Analytics - Marketing Analytics - Financial Analytics - Healthcare Analytics - Supply Chain Analytics.	9	C5
	Total	**45**	
Course Outcomes			
Course Outcomes	On completion of this course, students will;	**Program Outcomes**	
CO1	Be able to understand the basics of Business Analytics	PO1, PO2	
CO2	Possess awareness and understanding on visualizing data through collecting, managing and analyzing data.	PO1, PO2,	
CO3	Possess knowledge on data mining and multi-dimensional data analysis	PO2, P05, PO6	
CO4	Have knowledge on machine learning and AI.	PO4, PO5	
CO5	Possess knowledge on the analysis of various areas of business.	PO2, P05, PO6	
Reading List			
1.	https://ptgmedia.pearsoncmg.com/images/9780133552188/samplepages/0133552187.pdf		
2.	http://www.gerkoole.com/IBA/downloads/IBA_Koole_first_chapters.pdf		
3.	Jeen-Su Lim, John H. Heinrichs. (2021) Developing context-relevant project experiences for marketing analytics students. Decision Sciences Journal of Innovative Education 19:2, pages 150-156.		
4.	Wullianallur Raghupathi, Viju Raghupathi. (2021) Contemporary Business Analytics: An Overview. Data 6:8, pages 86.		
References Books			
1.	Majid Nabavi, David L.Olson, Introduction to Business Analytics, Business Expert Press,2018		
2.	Umesh R Hodeghatta and Umesha Nayak, Business Analytics Using R - A PracticalApproachApress, 2017.		
3.	Jeffery D.Camm, James J. Cochran, Michael J. Fry, Jeffrey W. Ohlmann, David R.Anderson, Essentials of Business Analytics, Cengage Learning, 2015		
4.	Sandhya Kuruganti, Business Analytics: Applications To Consumer Marketing, McGrawHill, 2015		
5.	Bernard Marr, Big Data: Using Smart Big Data, Analytics and Metrics to Make BetterDecisions and Improve Performance, Wiley, 2015		
Methods of Evaluation			
Internal Evaluation	Continuous Internal Assessment Test	25 Marks	
	Assignments		
	Seminars		
	Attendance and Class Participation		

External Evaluation	End Semester Examination	75 Marks
	Total	100 Marks
Methods of Assessment		
Recall (K1)	Simple definitions, MCQ, Recall steps, Concept definitions	
Understand / Comprehend (K2)	MCQ, True/False, Short essays, Concept explanations, Short summary or overview	
Application (K3)	Suggest idea/concept with examples, Suggest formulae, Solve problems, Observe, Explain	
Analyze (K4)	Problem-solving questions, Finish a procedure in many steps, Differentiate between various ideas, Map knowledge	
Evaluate (K5)	Longer essay/ Evaluation essay, Critique or justify with pros and cons	
Create (K6)	Check knowledge in specific or offbeat situations, Discussion, Debating or Presentations	

Unit I: Introduction to Business Analytics

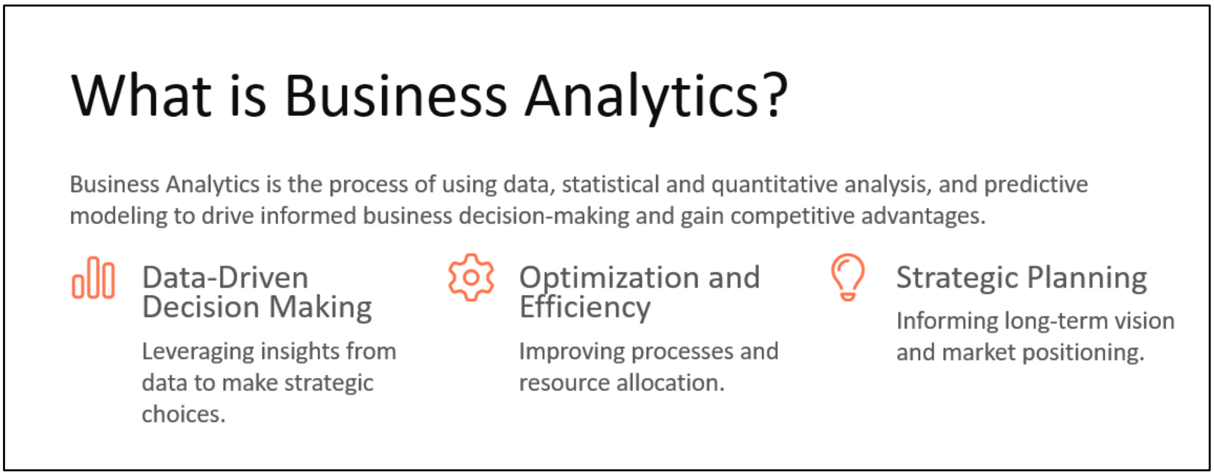
1. Meaning of Business Analytics

Business Analytics (BA) is the **practice of exploring data to uncover patterns, generate insights, and support decision-making**. It combines **statistics, data science, and business intelligence** to improve operational efficiency, increase revenue, and gain competitive advantage.

Key Points:

- Focuses on **data-driven decisions**.
- Helps in **predicting future trends** (predictive analytics) and identifying **key business drivers**.

Example:
A retail company analyzing past sales to forecast demand for the next season, thus optimizing inventory.



What is Business Analytics?

Business Analytics is the process of using data, statistical and quantitative analysis, and predictive modeling to drive informed business decision-making and gain competitive advantages.

Data-Driven Decision Making

Leveraging insights from data to make strategic choices.

Optimization and Efficiency

Improving processes and resource allocation.

Strategic Planning

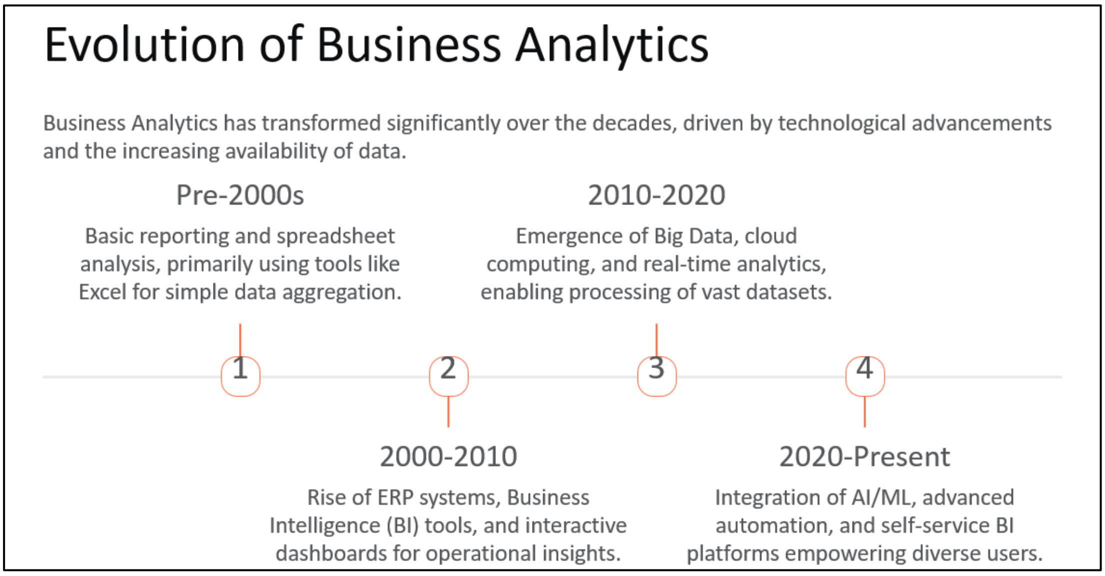
Informing long-term vision and market positioning.

---

2. Historical Overview of Data Analysis

- **1950s-60s:** Introduction of **statistical analysis** for business decisions.
- **1970s-80s:** Emergence of **Decision Support Systems (DSS)**.
- **1990s: Data Warehousing and OLAP** tools.
- **2000s-present:** Growth of **big data, machine learning, and predictive analytics**.

Observation:
The evolution reflects a shift from **manual calculations to automated, AI-powered analytics**.



Evolution of Business Analytics

Business Analytics has transformed significantly over the decades, driven by technological advancements and the increasing availability of data.

1 Pre-2000s

Basic reporting and spreadsheet analysis, primarily using tools like Excel for simple data aggregation.

2 2000-2010

Rise of ERP systems, Business Intelligence (BI) tools, and interactive dashboards for operational insights.

3 2010-2020

Emergence of Big Data, cloud computing, and real-time analytics, enabling processing of vast datasets.

4 2020-Present

Integration of AI/ML, advanced automation, and self-service BI platforms empowering diverse users.

Analytics Timeline

The journey of analytics reflects a continuous quest for better insights and more informed decision-making, from early statistical methods to advanced AI.

1 **1950s: OR & Stats**
Operations Research and statistical analysis laid the groundwork for data-driven problem solving.

2 **1980s: MIS Systems**
Management Information Systems (MIS) began centralizing data for business reporting.

3 **2000s: BI & Data Warehousing**
Business Intelligence (BI) tools and data warehouses emerged for deeper historical analysis.

4 **2010s: Big Data & Cloud**
The era of Big Data and cloud analytics revolutionized data storage and processing capabilities.

5 **2020s: AI/ML & Real-time**
Integration of AI/ML enables advanced real-time decisioning and automation.

Types of Analytics

Business analytics can be categorised into three main types, each serving a distinct purpose in understanding and influencing business outcomes.

1. Descriptive Analytics

What happened? Focuses on summarizing past data to understand historical events and trends.

Tools: Dashboards, Reports, Data Aggregation, Data Visualisation tools.

2. Predictive Analytics

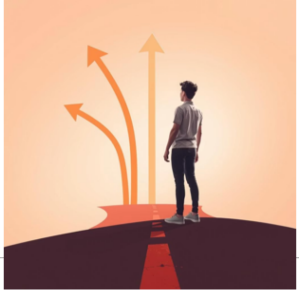
What could happen? Uses statistical models and machine learning to forecast future outcomes.

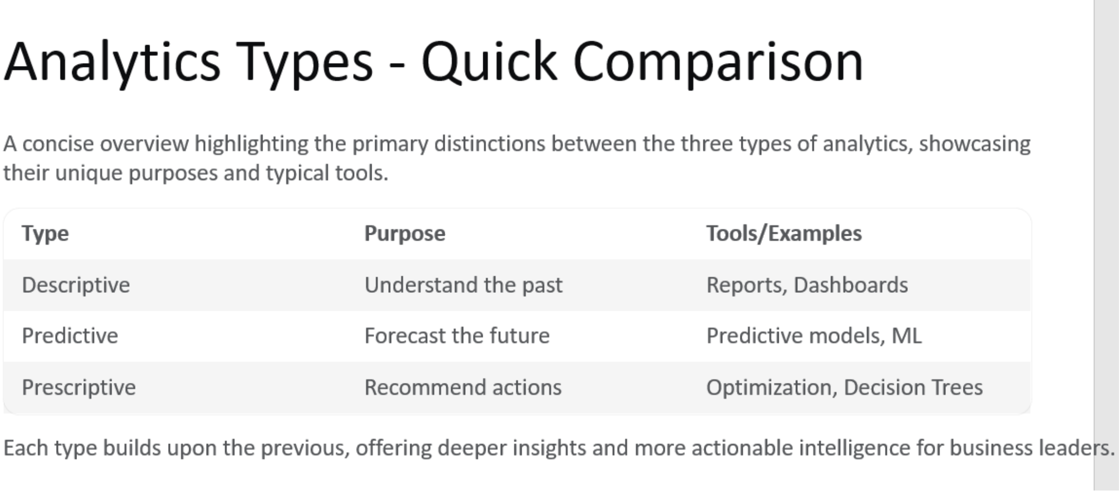
Tools: Regression, Forecasting, Machine Learning Models, Predictive Algorithms.

3. Prescriptive Analytics

What should we do? Recommends optimal actions to achieve desired outcomes, considering various constraints.

Tools: Optimization, Simulation, Decision Algorithms, A/B Testing.





Analytics Types - Quick Comparison

A concise overview highlighting the primary distinctions between the three types of analytics, showcasing their unique purposes and typical tools.

Type	Purpose	Tools/Examples
Descriptive	Understand the past	Reports, Dashboards
Predictive	Forecast the future	Predictive models, ML
Prescriptive	Recommend actions	Optimization, Decision Trees

Each type builds upon the previous, offering deeper insights and more actionable intelligence for business leaders.

---

3. Roles in Analytics

3.1 Data Scientist

- Designs **models** to extract insights.
- Performs **predictive analytics, machine learning, and statistical modeling**.
- Skills: Python, R, SQL, ML algorithms, visualization.

3.2 Data Engineer

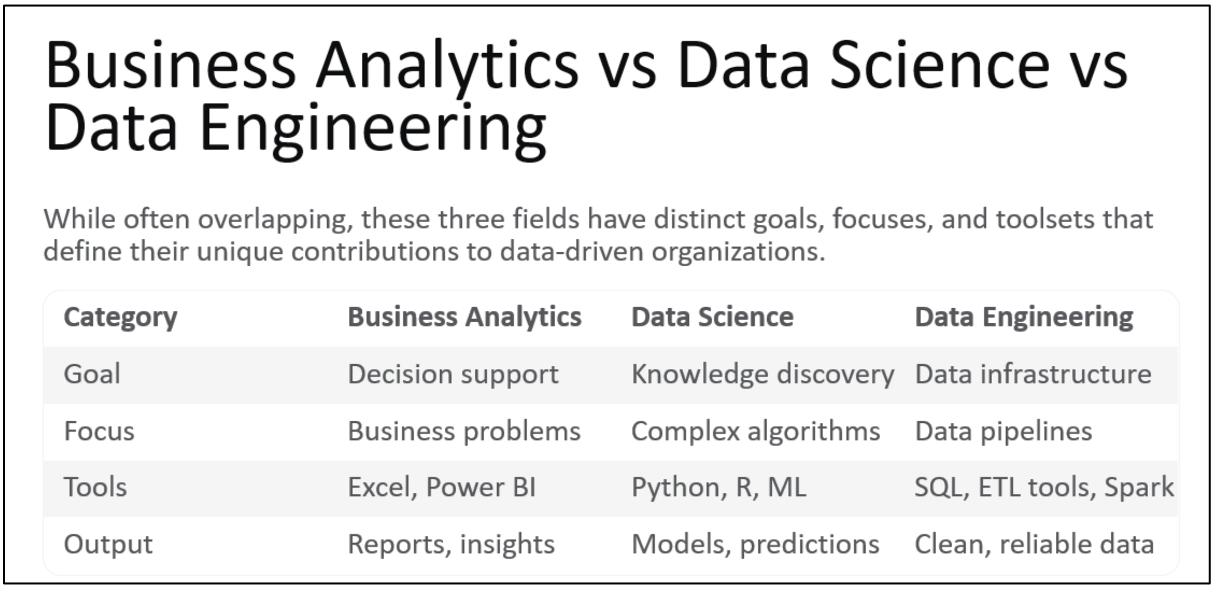
- Builds **data infrastructure and pipelines**.
- Ensures **data storage, quality, and accessibility**.
- Skills: SQL, Hadoop, Spark, ETL tools.

3.3 Business Analyst

- Focuses on **interpreting data for business decisions**.
- Connects **analytics insights with business strategy**.
- Skills: Excel, dashboards, reporting, basic statistics.

Career Path:

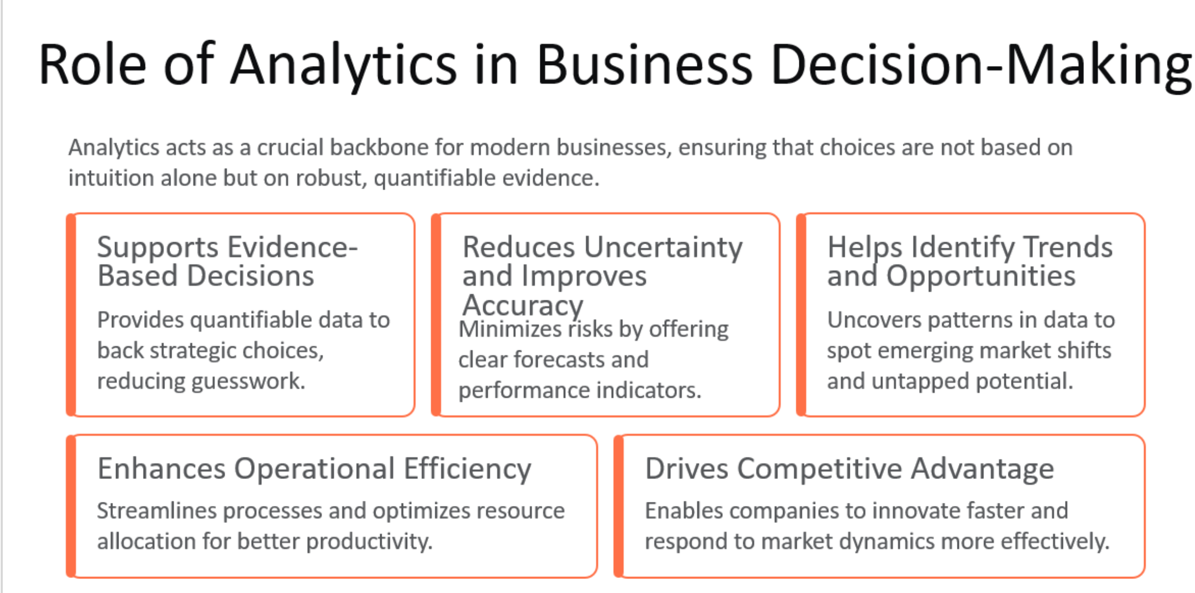
- **Data Scientist:** Advanced analytics, AI models, predictive insights.
- **Data Engineer:** Infrastructure, big data handling.
- **Business Analyst:** Decision-making, dashboards, KPIs.



Business Analytics vs Data Science vs Data Engineering

While often overlapping, these three fields have distinct goals, focuses, and toolsets that define their unique contributions to data-driven organizations.

Category	Business Analytics	Data Science	Data Engineering
Goal	Decision support	Knowledge discovery	Data infrastructure
Focus	Business problems	Complex algorithms	Data pipelines
Tools	Excel, Power BI	Python, R, ML	SQL, ETL tools, Spark
Output	Reports, insights	Models, predictions	Clean, reliable data



Role of Analytics in Business Decision-Making

Analytics acts as a crucial backbone for modern businesses, ensuring that choices are not based on intuition alone but on robust, quantifiable evidence.

Supports Evidence-Based Decisions
Provides quantifiable data to back strategic choices, reducing guesswork.

Reduces Uncertainty and Improves Accuracy
Minimizes risks by offering clear forecasts and performance indicators.

Helps Identify Trends and Opportunities
Uncovers patterns in data to spot emerging market shifts and untapped potential.

Enhances Operational Efficiency
Streamlines processes and optimizes resource allocation for better productivity.

Drives Competitive Advantage
Enables companies to innovate faster and respond to market dynamics more effectively.

4. Introduction to Data Science

Data Science integrates **statistics, computer science, and domain expertise** to extract knowledge from data.

Applications:

- **Retail:** Personalized recommendations, demand forecasting.
- **Finance:** Fraud detection, credit scoring.
- **Healthcare:** Patient outcome prediction, resource optimization.

- **Marketing:** Customer segmentation, campaign effectiveness.

Example:
Netflix uses viewing history (data) to recommend shows (insight) increasing **user engagement**.

---

5. Roles and Responsibilities of Data Scientists

- Collect, clean, and process large datasets.
- Build predictive and prescriptive models.
- Perform **data visualization** for insights communication.
- Collaborate with **business units** for decision-making.
- Ensure **data quality and ethical use** of analytics.

Case Study: Netflix Case Study

Title: Netflix Case Study: The Power of Data-Driven Content Strategy
Duration: 40 Minutes (Groups of 4-5)

Instructions: - Analyze Netflix's use of descriptive, predictive, and prescriptive analytics - Identify how analytics influenced their strategic business decisions - Present key takeaways

Guidance: Use the worksheet provided and assign one member to present

Key Learning Outcomes

By the end of this case study discussion, students should be able to:

1. **Explain** how Netflix uses different types of analytics (descriptive, predictive, prescriptive) in their business operations
2. **Analyse** the business value of data-driven decision-making in content creation and user experience
3. **Evaluate** the ethical implications of personalised recommendation systems
4. **Apply** Netflix's analytics principles to other industries and business contexts

Assess the limitations and challenges of heavy reliance on data analytics

Unit II: Data Visualization and Project Life Cycle

1. Data Collection

- Sources: **Primary** (surveys, experiments) and **Secondary** (company databases, industry reports).
- Importance: Data quality depends on correct, complete, and reliable collection.

---

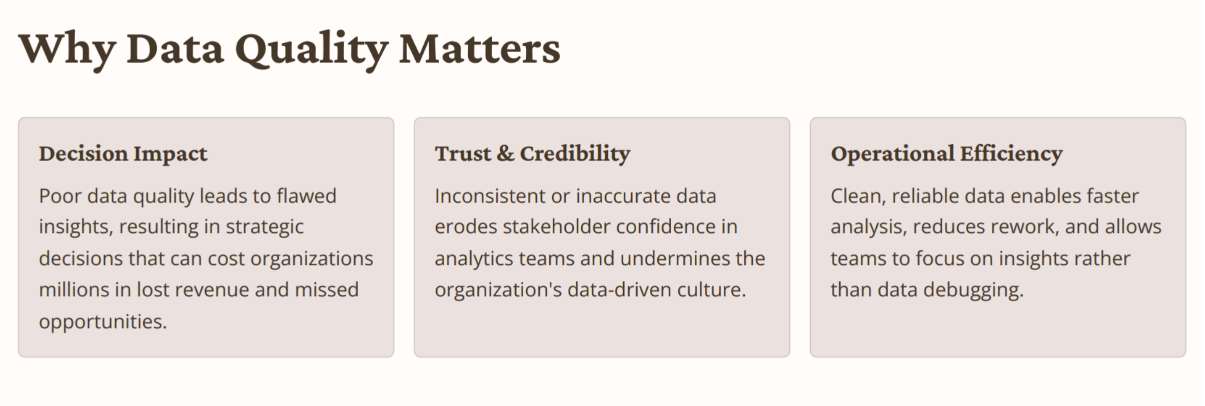
2. Data Management

- Organizing, storing, and **maintaining datasets** efficiently.
- Key challenges: **Volume, variety, velocity** (Big Data).
- Tools: SQL databases, NoSQL (MongoDB), cloud storage.

---

3. Importance of Data Quality

- **Completeness:** No missing records.
- **Consistency:** Same format across sources.
- **Accuracy:** Correct data values.



Why Data Quality Matters

Decision Impact

Poor data quality leads to flawed insights, resulting in strategic decisions that can cost organizations millions in lost revenue and missed opportunities.

Trust & Credibility

Inconsistent or inaccurate data erodes stakeholder confidence in analytics teams and undermines the organization's data-driven culture.

Operational Efficiency

Clean, reliable data enables faster analysis, reduces rework, and allows teams to focus on insights rather than data debugging.

Dealing with Missing Data:

- Remove incomplete records.
- Impute missing values using mean, median, or predictive models.

The Hidden Costs of Poor Data Quality

30%

Time Wasted

Data scientists spend up to 30% of their time cleaning and preparing data instead of generating insights.

$15M

Annual Cost

Average cost of poor data quality for large organizations, including lost productivity and wrong decisions.

1 in 3

Failed Projects

Analytics projects fail due to data quality issues, wasting resources and delaying critical business initiatives.

Common Data Quality Issues

Missing Data

- Incomplete customer records
- Gaps in time-series data
- Survey non-responses

Inconsistent Formats

- Mixed date formats (MM/DD vs DD/MM)
- Varying naming conventions
- Different measurement units

Duplicate Records

- Multiple customer entries
- System integration errors
- Manual data entry mistakes

Outdated Information

- Stale customer contact details
- Legacy system data
- Infrequent data refreshes

Essential Data Cleaning Techniques

01

Data Profiling

Analyze data distributions, identify patterns, detect anomalies, and understand data structure before cleaning begins.

02

Remove Duplicates

Use algorithms to identify and eliminate duplicate records while preserving data integrity and maintaining referential relationships.

03

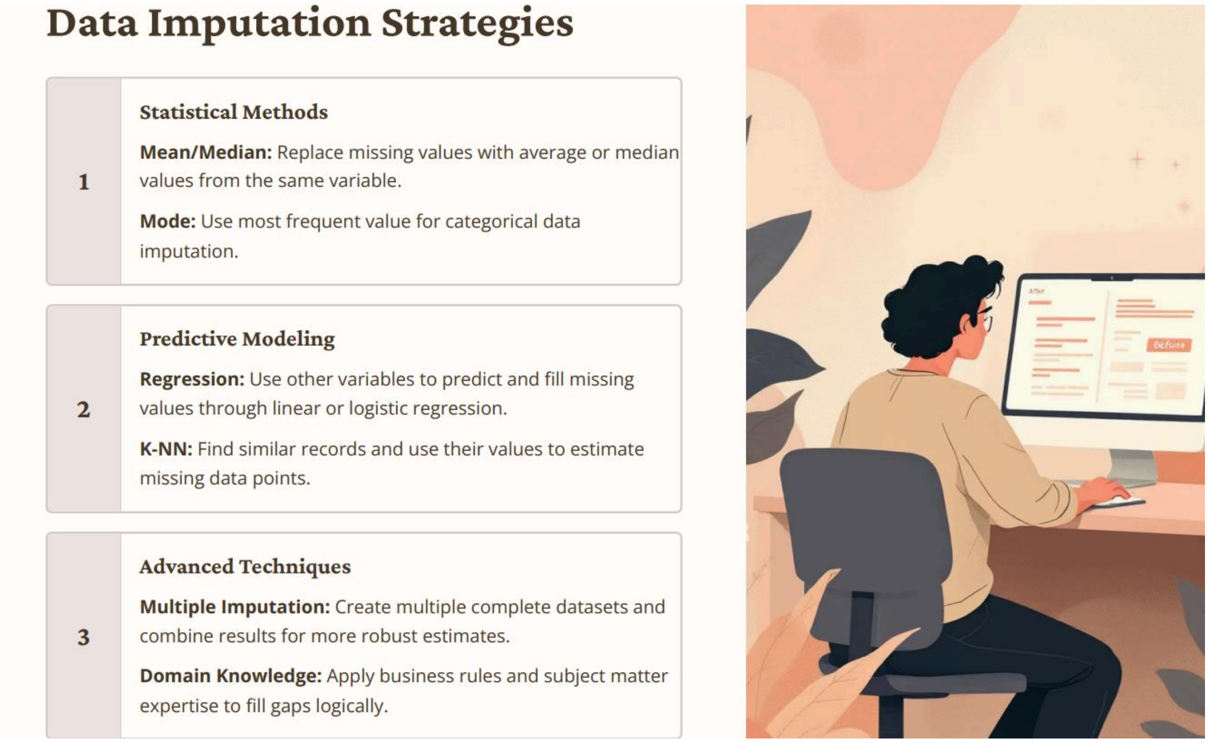
Standardize Formats

Convert data to consistent formats, normalize text cases, and establish uniform naming conventions across all datasets.

04

Handle Outliers

Detect statistical outliers, determine if they're errors or valid extreme values, and apply appropriate treatment strategies.



Data Imputation Strategies

1 Statistical Methods

Mean/Median: Replace missing values with average or median values from the same variable.

Mode: Use most frequent value for categorical data imputation.

2 Predictive Modeling

Regression: Use other variables to predict and fill missing values through linear or logistic regression.

K-NN: Find similar records and use their values to estimate missing data points.

3 Advanced Techniques

Multiple Imputation: Create multiple complete datasets and combine results for more robust estimates.

Domain Knowledge: Apply business rules and subject matter expertise to fill gaps logically.

4. Data Visualization

- Converts complex data into **graphs, charts, dashboards**.
- Techniques: Bar charts, line charts, heat maps, scatter plots.
- Tools: Tableau, Power BI, Python (Matplotlib, Seaborn), R (ggplot2).

---

5. Data Classification

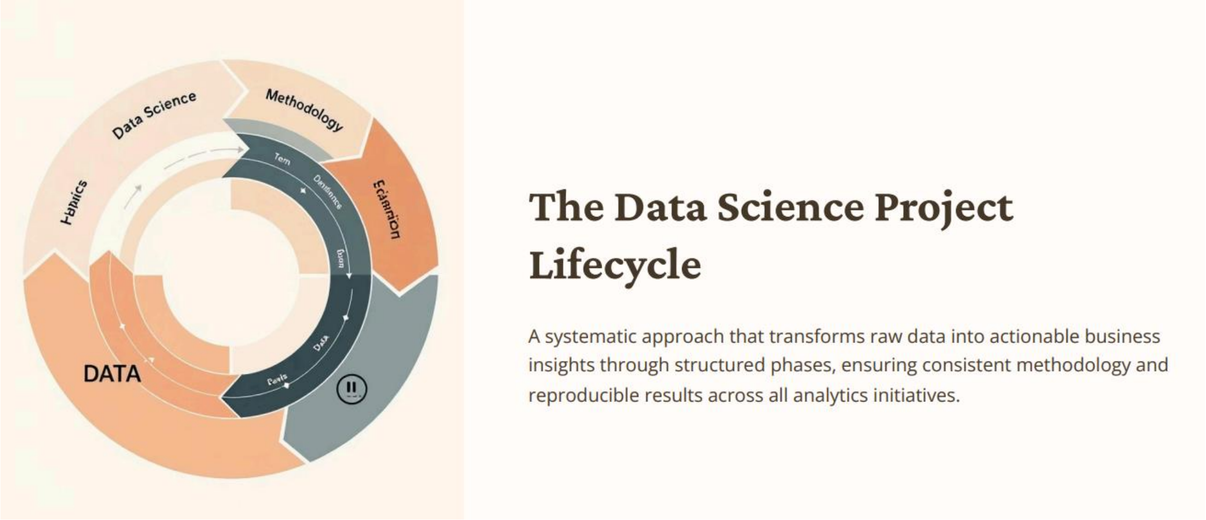
- **Organizing data** into categories for analysis.
- Examples: Customer segmentation by age, region, purchase behavior.

---

6. Data Science Project Life Cycle

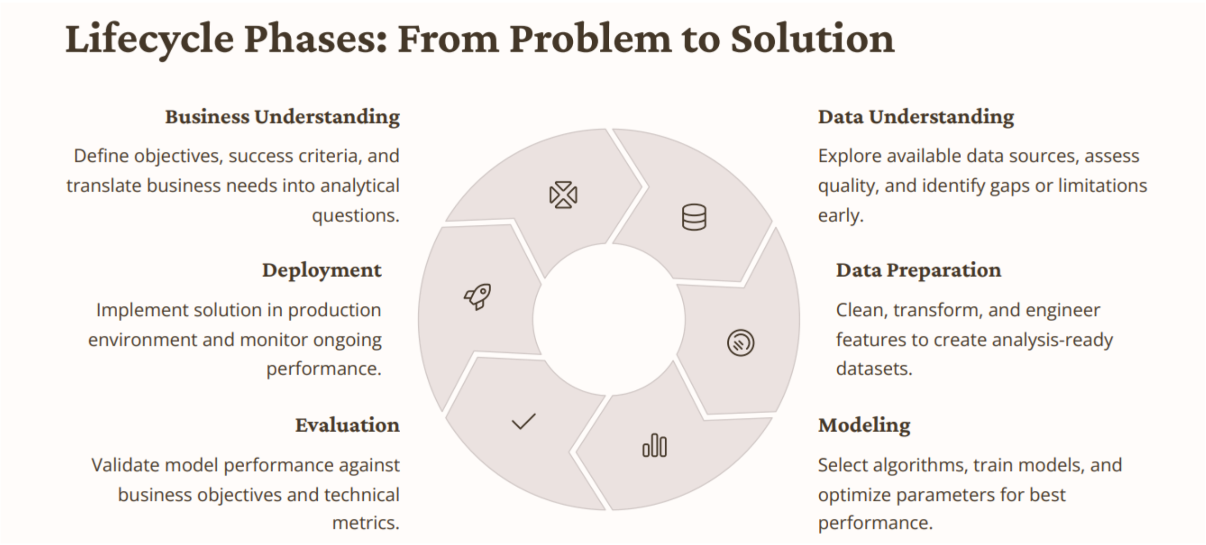
1. **Business Requirement:** Understand objectives and KPIs.
2. **Data Acquisition:** Collect relevant datasets.
3. **Data Preparation:** Clean, transform, and structure data.
4. **Hypothesis and Modeling:** Explore correlations, build predictive models.
5. **Evaluation and Interpretation:** Test model accuracy, interpret results.

6. **Deployment:** Implement models in operations.
7. **Operations and Optimization:** Monitor performance, refine models.



The Data Science Project Lifecycle

A systematic approach that transforms raw data into actionable business insights through structured phases, ensuring consistent methodology and reproducible results across all analytics initiatives.



Lifecycle Phases: From Problem to Solution

Business Understanding

Define objectives, success criteria, and translate business needs into analytical questions.

Data Understanding

Explore available data sources, assess quality, and identify gaps or limitations early.

Data Preparation

Clean, transform, and engineer features to create analysis-ready datasets.

Modeling

Select algorithms, train models, and optimize parameters for best performance.

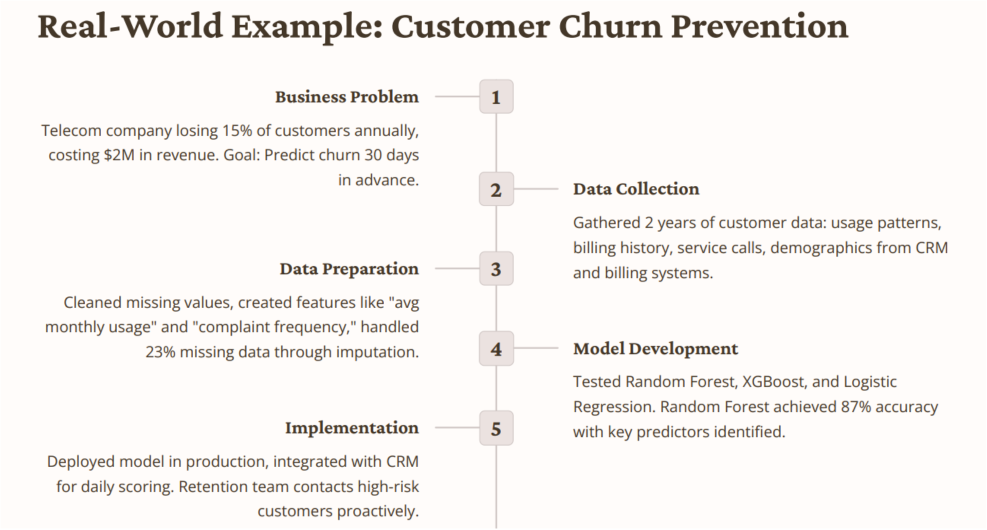
Evaluation

Validate model performance against business objectives and technical metrics.

Deployment

Implement solution in production environment and monitor ongoing performance.

Example:



Real-World Example: Customer Churn Prevention

1 **Business Problem**

Telecom company losing 15% of customers annually, costing $2M in revenue. Goal: Predict churn 30 days in advance.

2 **Data Collection**

Gathered 2 years of customer data: usage patterns, billing history, service calls, demographics from CRM and billing systems.

3 **Data Preparation**

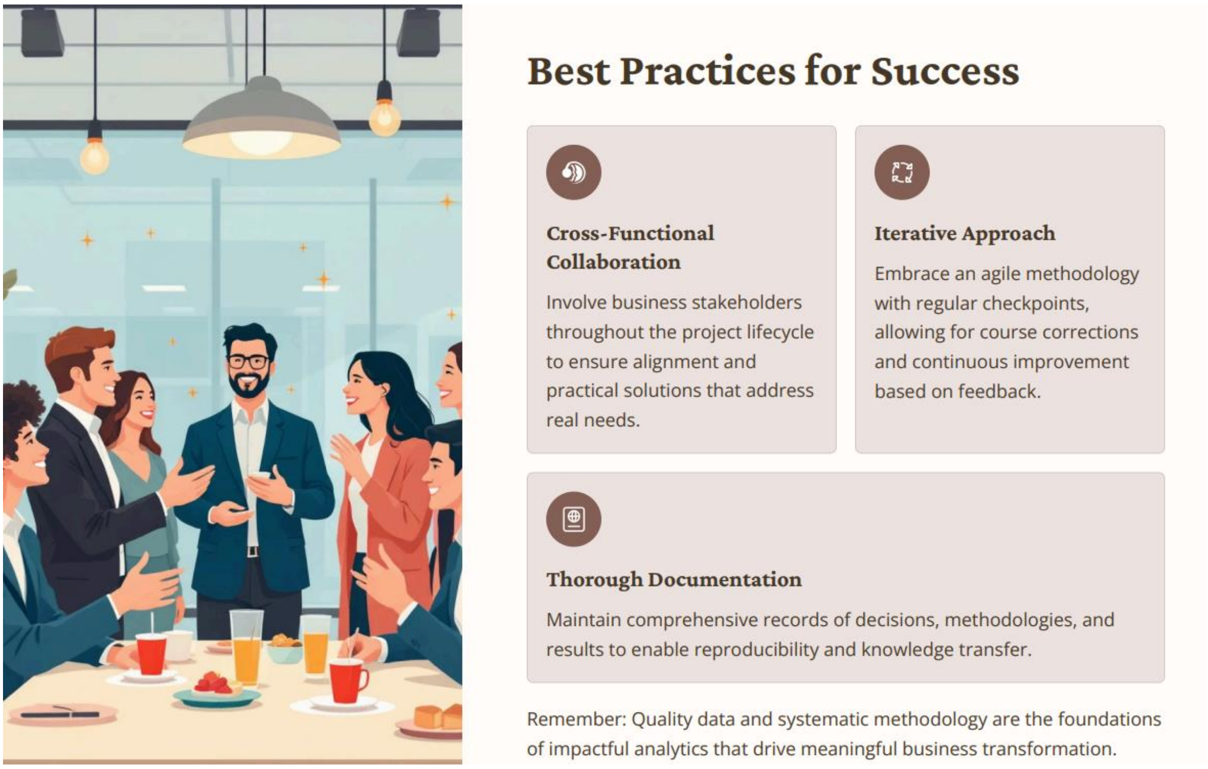
Cleaned missing values, created features like "avg monthly usage" and "complaint frequency," handled 23% missing data through imputation.

4 **Model Development**

Tested Random Forest, XGBoost, and Logistic Regression. Random Forest achieved 87% accuracy with key predictors identified.

5 **Implementation**

Deployed model in production, integrated with CRM for daily scoring. Retention team contacts high-risk customers proactively.



Best Practices for Success

Cross-Functional Collaboration

Involve business stakeholders throughout the project lifecycle to ensure alignment and practical solutions that address real needs.

Iterative Approach

Embrace an agile methodology with regular checkpoints, allowing for course corrections and continuous improvement based on feedback.

Thorough Documentation

Maintain comprehensive records of decisions, methodologies, and results to enable reproducibility and knowledge transfer.

Remember: Quality data and systematic methodology are the foundations of impactful analytics that drive meaningful business transformation.

Unit III – OLAP and Multi-Dimensional Data

What is OLAP?

Discover how OLAP (Online Analytical Processing) enhances data analysis by providing rapid, multidimensional exploration for informed business decision-making.

OLAP’s integration with data warehouses has allowed data analysts to find patterns and relationships within the data at a much faster speed and at a much higher level of efficiency. But the work of OLAP, which stands for Online Analytical Processing, happens under the hood, so to speak. So, you might be a skilled analyst working with data in a business intelligence tool and be supported by OLAP, but you do not necessarily know what OLAP is or how helpful it is.

What is Online Analytical Processing?

What is OLAP, or online analytical processing? In short, OLAP is a data processing technique that has changed **how we analyze massive amounts of data**. Because OLAP has become a big and important part of our data workflows, analysts might be working with OLAP advancements without even realizing it.

With OLAP, data analysts can rapidly analyze complex queries to transform enormous datasets into the actionable insights that our supervisors often ask for. With OLAP, analysts can perform multidimensional analysis, meaning they look at dimensions such as time, geography, and hierarchical categories to see patterns and trends that would probably be difficult to detect if you were working with tables and had to find associations without OLAP.

OLAP. Source: CSUS coursework online

How OLAP is involved in manipulating large datasets? The best part is you can perform these operations in real-time. I think about these three specific actions:

1. **Extract specific perspectives from the dataset.**
2. **Drill into granular details for in-depth analysis.**
3. **Aggregate it to generate high-level summaries.**

For instance, OLAP helps us extract sales data by region, drill into the details to analyze sales performance for each individual store, and then aggregate all the info to provide overall sales trends.

What Are OLAP Cubes?

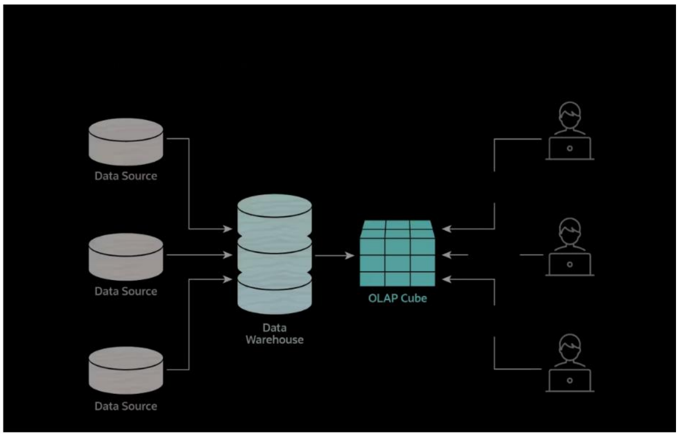
An OLAP cube, also known as a hypercube, is a kind of data structure that permits fast and multi-dimensional analysis of extensive data beyond what can be achieved by relational databases. Its key features include the following:

1. **Dimensions**: They represent different categories for analysis.
2. **Measures**: They are numerical values that you want to analyze.
3. **Pre-Aggregation**: Data is pre-calculated and stored for rapid querying here.
4. **Hierarchies**: They allow for drilling down within dimensions.
5. **Fast Query Performance**: They allow quick analysis of large datasets.

Simply put, OLAP cubes take our tables, which are flat and two-dimensional (think like spreadsheets), and elevate them into something that has three (or many more) dimensions. OLAP cubes add additional layers and associations. They organize data appropriately to support complex analytical queries and reporting at a much higher speed.

How OLAP Systems Work

Analyzing large volumes of multidimensional data begins with collecting data from various sources and storing it in data warehouses, followed by creating OLAP cubes for fast and efficient querying. Here's a visual so you can see OLAP's place in this:



OLAP working process. Source: NetSuite

Data collection and storage

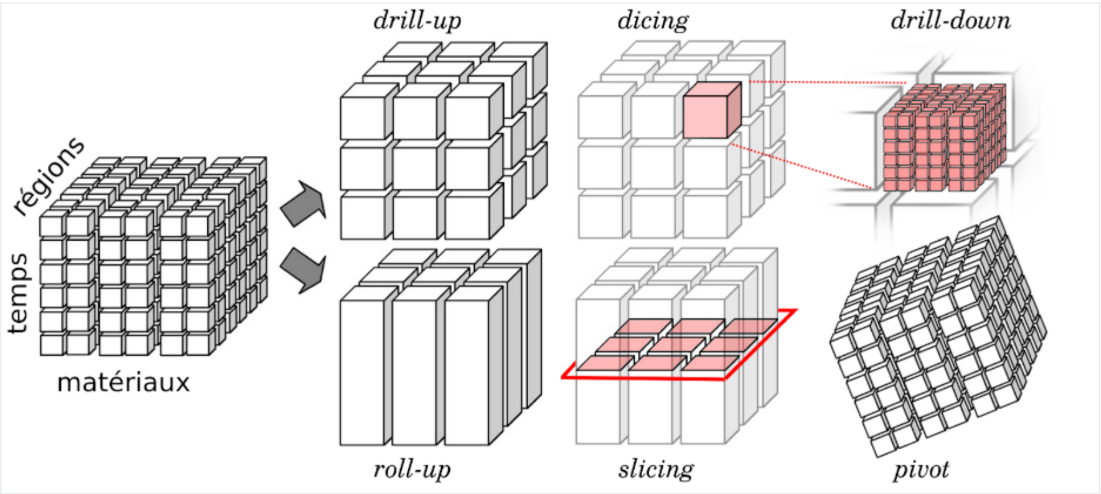
Data is extracted from multiple sources, such as transactional databases and external systems. Then, the extracted data undergoes a transformation process to maintain quality. It’s then loaded into a centralized data warehouse optimized for analytical queries and historical data storage.

Creation of OLAP cubes

OLAP cubes are pre-calculated, multidimensional data structures built from the data stored in the data warehouse. They organize data along various predefined dimensions and hierarchies (e.g., time, product, location) and measures (e.g., sales, quantity). This pre-calculation of aggregations is a key part of OLAP, and it's really what gives rapid query responses. Because OLAP cubes are pre-aggregated at various levels, so we can access summarized data without writing the same SQL query on raw data.

Fast and efficient data querying

OLAP cubes make it easier to quickly access summarized data at different granularity levels. You can perform complex analytical queries without impacting the source systems and enjoy more flexible data exploration. Here are some of the key OLAP operations that data analysts can perform:



Basic OLAP operations. Source: Wikimedia

- **Roll up**: The roll-up operation summarizes data to provide a less detailed view. For example, instead of viewing product sales by specific cities like Paris, Berlin, Madrid and Rome, a roll-up aggregates this data to show sales by regions such as Western Europe and Southern Europe.
- **Drill down**: The opposite of roll-up is drill-down. Here, the analyst can go deeper into the data hierarchy to retrieve lower-level information. For example, while looking at the data for annual sales, one can drill down to monthly sales.
- **Slice**: The slice operation extracts a specific subset of data based on one dimension. For example, slicing can analyze sales data for a particular product category.

- **Dice**: The dice operation lets analysts simultaneously select and analyze data from multiple dimensions. For example, they can examine sales data for specific products in certain regions during a given period.
- **Pivot**: The pivot operation rotates the OLAP cube in one of the dimensions. So a data analyst, interested in exploring a graph with a different configuration, might pivot the data using drag-and-drop actions in a **graphical user interface**.

Types of OLAP Systems

There are different types of OLAP systems, each with different usage and benefits. Let’s summarize the types with a table.

OLAP Type	Summary	Benefits
Multidimensional OLAP (MOLAP)	It uses a multidimensional database to store pre-computed data in cube structures.	Suitable for complex calculations. Optimized for multidimensional analysis.
Relational OLAP (ROLAP)	It works with relational databases to access data using SQL queries.	Can handle large data volumes. Leans on existing relational database technology. More flexible for changing data structures.
Hybrid OLAP (HOLAP)	HOLAP stores some data in a multidimensional format and some in a relational format.	Balances performance and scalability. Offers flexibility in data storage and access. Can be optimized based on specific use cases.

OLAP vs. OLTP

Now that we have looked at different types of OLAP systems, we can compare OLAP versus OLTP. Unlike OLAP, OTLP, which stands for online transaction processing, is designed for transaction processing rather than analytical processing. OLAP is focused on data analysis and querying, but OLTP is centered around real-time transaction processing and data integrity in operational databases.

Generally speaking, OLAP handles more complex queries so businesses can make better decisions. An OLTP system handles day-to-day transactions to maintain data integrity and service various applications, like order entry and inventory control.

More specifically, and in the language of data engineering and architecture, we can say that OLAP uses a multidimensional data model with a denormalized schema that is best for fast querying and read-heavy operations. OLTP, on the other hand, uses a normalized relational data model optimized for write-heavy operations to ensure consistency in real-time transactions. Let's capture the differences in a table:

Feature	**OLAP (Online Analytical Processing)**	**OLTP (Online Transaction Processing)**
Purpose	Complex queries and data analysis for decision-making	Handling day-to-day transactions and maintaining data integrity
Data Model	Multidimensional data model with denormalized schema	Normalized relational data model
Operations	Read-heavy operations	Write-heavy operations
Query Complexity	Supports complex queries and multidimensional analysis	Optimized for simple, short online transactions
Data Redundancy	Higher redundancy due to denormalized schema	Lower redundancy due to normalized schema
Performance Focus	Fast querying and analysis	Quick processing and maintaining data accuracy
Use Cases	Business intelligence, data warehousing, analytics	Order entry, inventory control, customer relationship management

Benefits of OLAP

Let’s now document the key benefits of OLAP, some of which we touched on above.

- **Faster Data Analysis**: OLAP speeds up querying and analyzing large amounts of data by reducing the time needed to generate reports and insights. This helps businesses make timely decisions based on the current available data.

- **Support for Non-Technical Users**: It makes data analysis more accessible for even non-technical users. As a result, aspiring data analysts can carry out complex analytical calculations and reports without learning advanced database usage.
- **Unified Data Perspective**: OLAP allows businesses to manage multiple functions like marketing, finance, and production on a single platform. This integrated view helps managers and decision-makers understand the greater context and address issues more effectively.
- **Complex Query Processing**: OLAP systems handle complex queries that involve many dimensions and vast volumes of data. This feature helps users analyze data in detail, which would be very difficult or impossible to do using traditional tools for data processing.

Implementing OLAP Solutions

Here are key factors to consider while implementing OLAP solutions within your organization:

1. **Business Requirements**: Clearly define analytical needs and objectives.
2. **Data Sources**: Identify and integrate relevant data sources.
3. **Scalability**: Ensure the solution can grow with increasing data volumes.
4. **User-Friendliness**: Choose tools that match users' technical skills.
5. **Integration**: Consider compatibility with existing systems.
6. **Cost**: Evaluate the total cost of ownership, including licensing and maintenance.

Some of the most common tools with OLAP capabilities include **IBM Cognos** and **Oracle OLAP**. These tools come with various advanced analytics capabilities to meet different needs.

Unit 4: Machine Learning

The machine learning unit covers basic and advanced concepts and is designed to cater to students and experienced working professionals.

This helps you gain a solid introduction to the fundamentals of machine learning and explore a wide range of techniques, including supervised, unsupervised, and reinforcement learning.

Machine learning (ML) is a subdomain of artificial intelligence (AI) that focuses on developing systems that learn—or improve performance—based on the data they ingest. Artificial intelligence is a broad word that refers to systems or machines that resemble human intelligence. Machine learning and AI are frequently discussed together, and the terms are occasionally used interchangeably, although they do not signify the same thing. A crucial distinction is that, while all machine learning is AI, not all AI is machine learning.

What is Machine Learning?

Machine Learning is the field of study that equips computers with the ability to learn without being explicitly programmed. ML is one of the most exciting technologies one has ever come across. As is evident from the name, it gives the computer that makes it more similar to humans: The ability to learn. Machine learning is actively being used today, perhaps in many more places than expected.

Features of Machine Learning

- Machine learning is a data-driven technology. Large amounts of data are generated by organizations daily. So, by notable relationships in data, organizations make better decisions.
- The machine can learn itself from past data and automatically improve.
- From the given dataset it detects various patterns on data.
- For big organizations branding is important and it will become easier to target a relatable customer base.
- It is similar to data mining because it also deals with a huge amount of data.

Why use Machine Learning?

Machine learning (ML) is essential across industries for several compelling reasons:

1. Automation and Efficiency:
   - ML automates tasks, freeing up human resources and improving operational efficiency.
2. Enhanced Data Insights:
   - Recognizes patterns and correlations in large datasets, enabling predictive analytics and informed decision-making.
3. Improved Accuracy:
   - ML algorithms deliver precise predictions and classifications, continuously learning and improving over time.

4. Personalization:
   - Creates tailored user experiences and targeted marketing strategies based on individual preferences and behaviors.
5. Cost Reduction:
   - Reduces operational costs through automation and fraud detection, saving resources and mitigating losses.
6. Innovation and Competitive Advantage:
   - Drives innovation by enabling new products and services, providing a competitive edge through data-driven strategies.
7. Real-World Applications:
   - Applies across healthcare, finance, retail, manufacturing, transportation, enhancing processes from diagnosis to supply chain management.
8. Handling Complex Data:
   - Processes high-dimensional data efficiently, extracting insights crucial for strategic decision-making.
9. Real-Time Decision Making:
   - Supports real-time analytics and adaptive systems, ensuring decisions are based on current, actionable data.
10. Interdisciplinary Impact:
    - Versatile applications span multiple disciplines, fostering collaboration and solving diverse, complex challenges.

Real-Life Examples of Machine Learning

Machine learning (ML) applications are ubiquitous in various industries, transforming how businesses operate and enhancing everyday experiences. Here are some compelling real-life examples:

1. Healthcare:
   - Medical Diagnosis: ML algorithms analyze patient data (such as symptoms and medical history) to assist doctors in diagnosing diseases accurately and early detection of illnesses.
   - Personalized Treatment: ML models predict optimal treatment plans based on genetic data, medical records, and patient demographics, improving patient outcomes.
2. Finance:

- Credit Scoring: Banks use ML to assess creditworthiness by analyzing past behavior and financial data, predicting the likelihood of loan repayment.
- Fraud Detection: ML algorithms detect unusual patterns in transactions, identifying and preventing fraudulent activities in real-time.

3. Retail:
   - Recommendation Systems: E-commerce platforms employ ML to suggest products based on customer browsing history, purchase patterns, and preferences, enhancing user experience and increasing sales.
   - Inventory Management: ML predicts demand trends and optimizes inventory levels, reducing stockouts and overstock situations.
4. Manufacturing:
   - Predictive Maintenance: ML models analyze sensor data from machinery to predict equipment failure before it occurs, enabling proactive maintenance and minimizing downtime.
   - Quality Control: ML algorithms inspect products on production lines, identifying defects with greater accuracy and consistency than human inspection.
5. Transportation:
   - Autonomous Vehicles: ML powers self-driving cars by interpreting real-time data from sensors (like cameras and radar) to navigate roads, detect obstacles, and make driving decisions.
   - Route Optimization: Logistics companies use ML to optimize delivery routes based on traffic conditions, weather forecasts, and historical data, reducing delivery times and costs.
6. Marketing:
   - Customer Segmentation: ML clusters customers into segments based on behavior and demographics, enabling targeted marketing campaigns and personalised promotions.
   - Sentiment Analysis: ML algorithms analyze social media and customer feedback to gauge public sentiment about products and brands, informing marketing strategies.
7. Natural Language Processing (NLP):

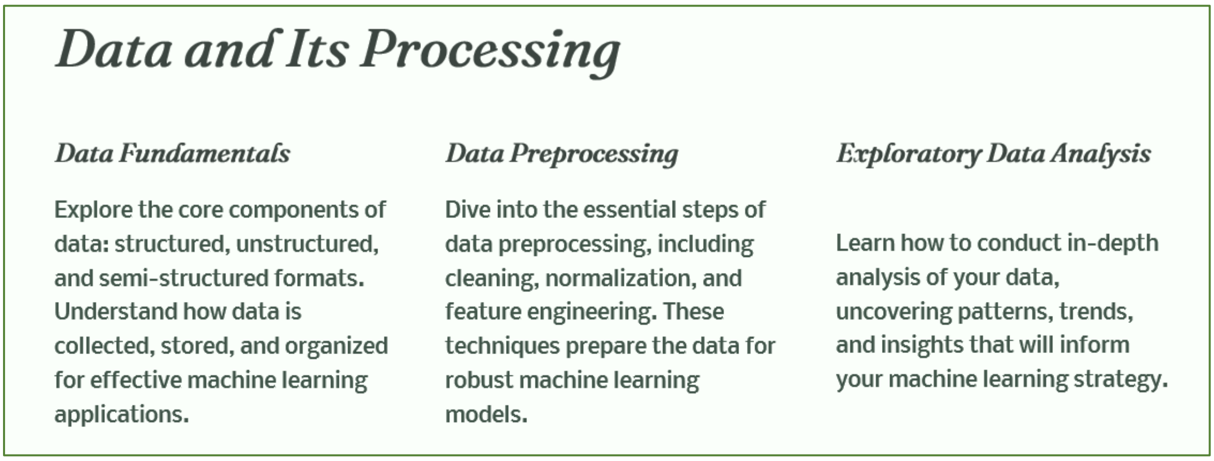
- Chatbots and Virtual Assistants: NLP models power conversational interfaces that understand and respond to natural language queries, enhancing customer support and service interactions.
- Language Translation: ML-driven translation tools translate text and speech between languages, facilitating global communication and collaboration.

8. Entertainment:
    - Content Recommendation: Streaming platforms use ML to recommend movies, TV shows, and music based on user preferences, viewing history, and ratings, improving content discovery.
9. Energy:
    - Smart Grids: ML optimizes energy distribution and consumption by predicting demand patterns, managing renewable energy sources, and improving grid stability and efficiency.
10. Education:
    - Adaptive Learning: ML algorithms personalize educational content and pathways based on student performance and learning styles, enhancing learning outcomes and engagement.

Roadmap to Learn Machine Learning

- **Phase 1: Fundamentals:** Mastering the fundamentals of mathematics, statistics, and programming lays the groundwork for a solid understanding of machine learning. From linear algebra and calculus to probability and Python programming, these foundational skills provide the essential toolkit for manipulating data, understanding algorithms, and optimising models. By delving into these areas, aspiring data scientists and machine learning enthusiasts build the necessary expertise to tackle complex problems and drive innovation.

- Phase 2: Data Handling and Visualization: Phase 2 focuses on mastering essential techniques for data acquisition, preparation, and exploration, crucial for effective machine learning. From collecting diverse data formats such as CSV, JSON, and XML, to utilizing SQL for database access and leveraging web scraping and APIs for data extraction, this phase equips learners with the tools to gather comprehensive datasets. Furthermore, it emphasizes the critical steps of cleaning and preprocessing data, including handling missing values, encoding categorical variables, and standardizing data for consistency. Exploratory Data Analysis (EDA) techniques, such as visualization through histograms, scatter plots, and box plots, alongside summary statistics, uncover valuable insights and patterns within the data, laying the foundation for informed decision-making and robust machine learning models.



Data and Its Processing

Data Fundamentals

Explore the core components of data: structured, unstructured, and semi-structured formats. Understand how data is collected, stored, and organized for effective machine learning applications.

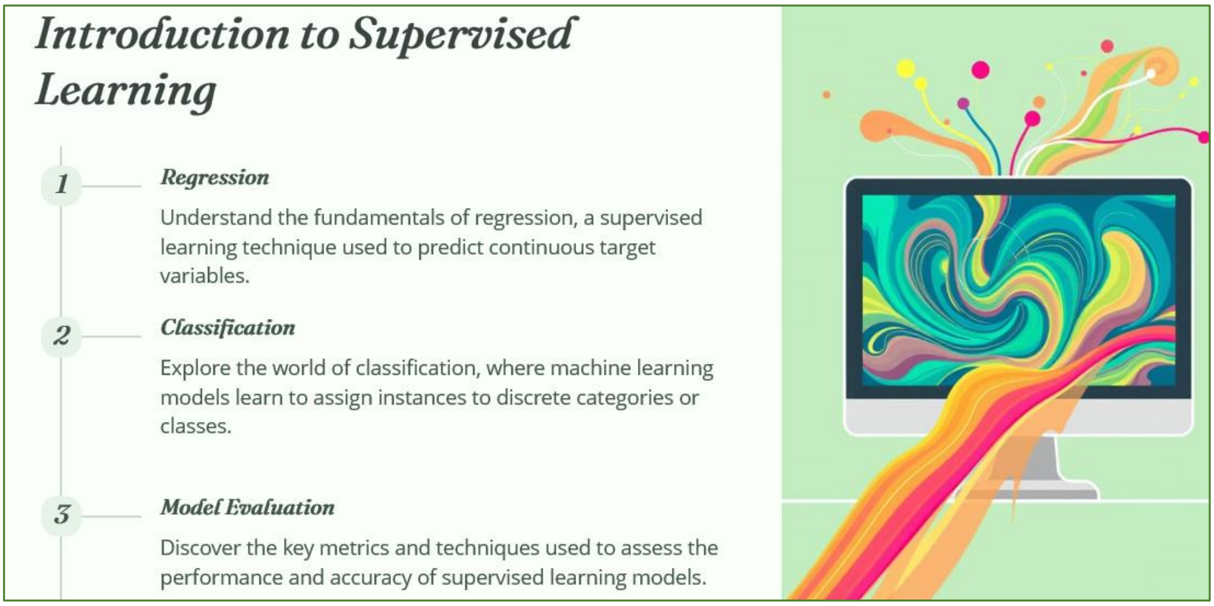
Data Preprocessing

Dive into the essential steps of data preprocessing, including cleaning, normalization, and feature engineering. These techniques prepare the data for robust machine learning models.

Exploratory Data Analysis

Learn how to conduct in-depth analysis of your data, uncovering patterns, trends, and insights that will inform your machine learning strategy.

- **Phase 3: Core Machine Learning Concepts:** In Phase 3, delving into core machine learning concepts opens doors to understanding and implementing various learning paradigms and algorithms. **Supervised learning focuses on predicting outcomes with labelled data,** while **unsupervised learning uncovers hidden patterns in unlabelled data**. Reinforcement learning, inspired by behavioural psychology, teaches algorithms through trial-and-error interactions. Common algorithms like **linear regression and decision trees empower predictive modelling**, while evaluation metrics like **accuracy and F1-score gauge model performance**. Together with cross-validation techniques, these components form the bedrock for developing robust machine-learning solutions.



Introduction to Supervised Learning

1 *Regression*

Understand the fundamentals of regression, a supervised learning technique used to predict continuous target variables.

2 *Classification*

Explore the world of classification, where machine learning models learn to assign instances to discrete categories or classes.

3 *Model Evaluation*

Discover the key metrics and techniques used to assess the performance and accuracy of supervised learning models.

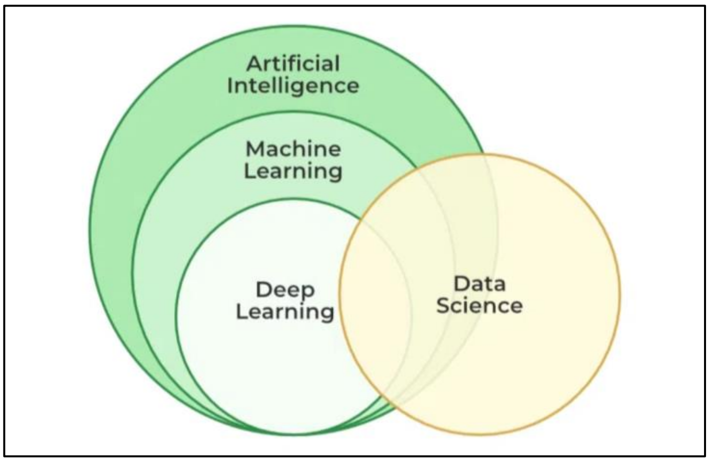
- **Phase 4: Advanced Machine Learning Topics:** Phase 4 delves into advanced machine-learning techniques for handling complex data and deploying sophisticated models. It covers deep learning fundamentals such as neural networks, CNNs for image recognition, and RNNs for sequential data. Frameworks like **TensorFlow, Keras, and PyTorch** are explored. In **natural language processing (NLP)**, topics include text preprocessing (tokenization, stemming, lemmatization), techniques like **Bag of Words, TF-IDF, and Word Embeddings (Word2Vec, GloVe), and applications such as sentiment analysis** and text classification. Model deployment strategies encompass saving/loading models, creating APIs with Flask or FastAPI, and utilizing cloud platforms (AWS, Google Cloud, Azure) for scalable model deployment. This phase equips learners with advanced skills crucial for applying machine learning in diverse real-world scenarios
- Phase 5: Practical Projects and Hands-On Experience: Phase 5 focuses on applying theoretical knowledge to real-world scenarios through practical projects. These hands-on experiences not only reinforce concepts learned but also build proficiency in implementing machine-learning solutions. From beginner to intermediate levels, these projects span diverse applications, from predictive analytics to deep learning techniques, showcasing the versatility and impact of machine learning in solving complex problems across various domains

What is Machine Learning? With Examples

Machine learning is a branch of artificial intelligence that enables algorithms to uncover hidden patterns within datasets, making predictions on new, similar data without explicit programming for each task. Traditional machine learning combines data with statistical tools to predict outputs, yielding actionable insights. This technology finds applications in diverse fields such as image and speech recognition, natural language processing, recommendation systems, fraud detection, portfolio optimization, and automating tasks.

For instance, recommender systems use historical data to personalize suggestions. Netflix, for example, employs collaborative and content-based filtering to recommend movies and TV shows based on user viewing history, ratings, and genre preferences. Reinforcement learning further enhances these systems by enabling agents to make decisions based on environmental feedback, continually refining recommendations.

Machine learning's impact extends to autonomous vehicles, drones, and robots, enhancing their adaptability in dynamic environments. This approach marks a breakthrough where machines learn from data examples to generate accurate outcomes, closely intertwined with data mining and data science.



Difference between Machine Learning and Traditional Programming

The Difference between Machine Learning and Traditional Programming is as follows:

Machine Learning	Traditional Programming	Artificial Intelligence

Machine Learning is a subset of artificial intelligence(AI) that focus on learning from data to develop an algorithm that can be used to make a prediction.	In traditional programming, rule-based code is written by the developers depending on the problem statements.	Artificial Intelligence involves making the machine as much capable, So that it can perform the tasks that typically require human intelligence.
Machine Learning uses a data-driven approach, It is typically trained on historical data and then used to make predictions on new data.	Traditional programming is typically rule-based and deterministic. It hasn't self-learning features like Machine Learning and AI.	AI can involve many different techniques, including Machine Learning and Deep Learning, as well as traditional rule-based programming.
ML can find patterns and insights in large datasets that might be difficult for humans to discover.	Traditional programming is totally dependent on the intelligence of developers. So, it has very limited capability.	Sometimes AI uses a combination of both Data and Pre-defined rules, which gives it a great edge in solving complex tasks with good accuracy which seem impossible to humans.
Machine Learning is the subset of AI. And Now it is used in various AI-based tasks like Chatbot Question answering, self-driven car., etc.	Traditional programming is often used to build applications and software systems that have specific functionality.	AI is a broad field that includes many different applications, including natural language processing, computer vision, and robotics.

A machine learning algorithm works by learning patterns and relationships from data to make predictions or decisions without being explicitly programmed for each task. Here's a simplified overview of how a typical machine-learning algorithm works:

1. **Data Collection**: Collect and curate relevant data. This data could include examples, features, or attributes that are important for the task at hand, such as images, text, numerical data, etc.

2. **Data Preprocessing**: Before feeding the data into the algorithm, it needs to be pre-processed. This step may involve cleaning the data (handling missing values, and outliers), transforming the data (normalization, scaling), and splitting it into training and test sets.

3. **Choosing a Model**: Depending on the task (e.g., **classification, regression, clustering**), a suitable machine-learning model is chosen. Examples include

decision trees, neural networks, support vector machines, and more advanced models like deep learning architectures.

4. **Training the Model:** The selected model is trained using the training data. During training, the algorithm learns patterns and relationships in the data. This involves adjusting model parameters iteratively to minimize the difference between predicted outputs and actual outputs (labels or targets) in the training data.

5. **Evaluating the Model:** Once trained, the model is evaluated using the test data to assess its performance. Metrics such as accuracy, precision, recall, or mean squared error are used to evaluate how well the model generalizes to new, unseen data.

6. **Fine-tuning:** Models may be fine-tuned by adjusting hyperparameters (parameters that are not directly learned during training, like learning rate or number of hidden layers in a neural network) to improve performance.

7. **Prediction or Inference:**

Finally, the trained model is used to make predictions or decisions on new data. This process involves applying the learned patterns to new inputs to generate outputs, such as class labels in classification tasks or numerical values in regression tasks.

Types of Machine Learning

- Environmental Supervised Machine Learning
- Unsupervised Machine Learning
- Reinforcement Machine Learning

1. Supervised Machine Learning:

Supervised learning is a type of machine learning in which the algorithm is trained on the labeled dataset. It learns to map input features to targets based on labeled training data. In supervised learning, the algorithm is provided with input features and corresponding output labels, and it learns to generalize from this data to make predictions on new, unseen data.

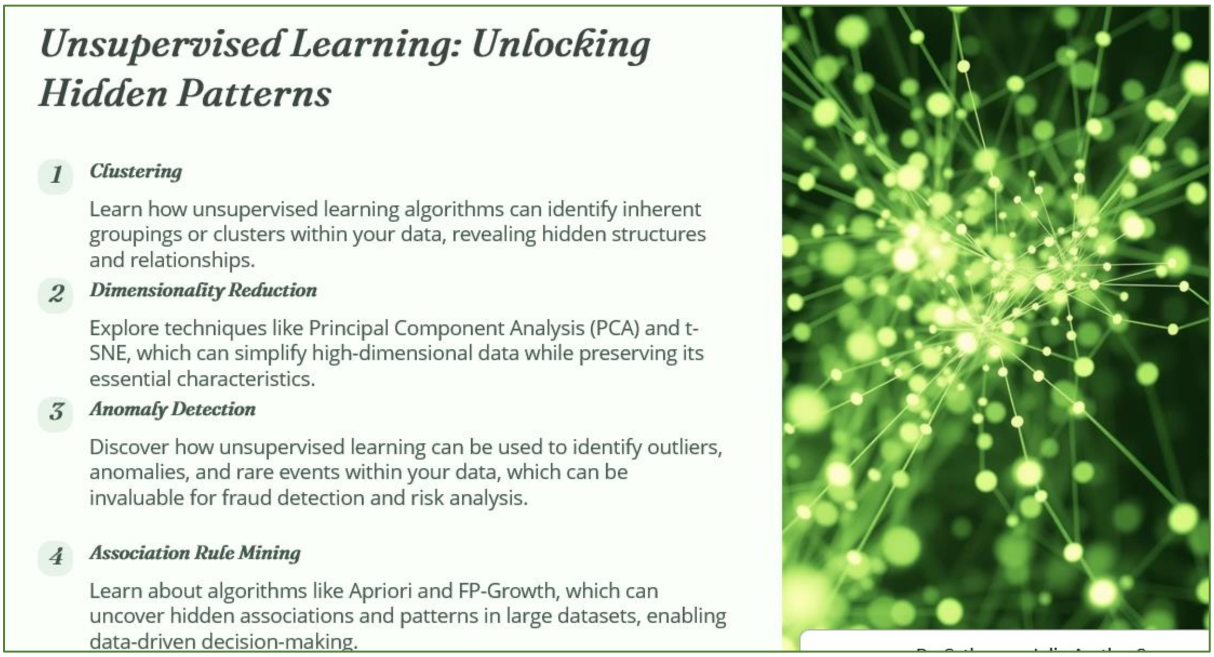
There are two main types of supervised learning:

Regression: Regression is a type of supervised learning where the algorithm learns to predict continuous values based on input features. The output labels in regression are continuous values, such as stock prices, and housing prices. The different regression algorithms in machine learning are: Linear Regression, Polynomial Regression, Ridge Regression, Decision Tree Regression, Random Forest Regression, Support Vector Regression, etc

Classification: Classification is a type of supervised learning where the algorithm learns to assign input data to a specific category or class based on input features. The output labels in classification are discrete values. Classification algorithms can be binary, where the output is one of two possible classes, or multiclass, where the output can be one of several classes. The different Classification algorithms in machine learning are: Logistic Regression, Naive Bayes, Decision Tree, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), etc

2. Unsupervised Machine Learning:

Unsupervised learning is a type of machine learning where the algorithm learns to recognize patterns in data without being explicitly trained using labeled examples. The goal of unsupervised learning is to discover the underlying structure or distribution in the data.



Unsupervised Learning: Unlocking Hidden Patterns

1 *Clustering*

Learn how unsupervised learning algorithms can identify inherent groupings or clusters within your data, revealing hidden structures and relationships.

2 *Dimensionality Reduction*

Explore techniques like Principal Component Analysis (PCA) and t-SNE, which can simplify high-dimensional data while preserving its essential characteristics.

3 *Anomaly Detection*

Discover how unsupervised learning can be used to identify outliers, anomalies, and rare events within your data, which can be invaluable for fraud detection and risk analysis.

4 *Association Rule Mining*

Learn about algorithms like Apriori and FP-Growth, which can uncover hidden associations and patterns in large datasets, enabling data-driven decision-making.

There are two main types of unsupervised learning:

Clustering: Clustering algorithms group similar data points together based on their characteristics. The goal is to identify groups, or clusters, of data points that are similar to each other, while being distinct from other groups. Some popular clustering algorithms include K-means, Hierarchical clustering, and DBSCAN.

Dimensionality reduction: Dimensionality reduction algorithms reduce the number of input variables in a dataset while preserving as much of the original information as possible. This is useful for reducing the complexity of a dataset and making it easier to visualize and analyze. Some popular dimensionality reduction algorithms include Principal Component Analysis (PCA), t-SNE, and Autoencoders.

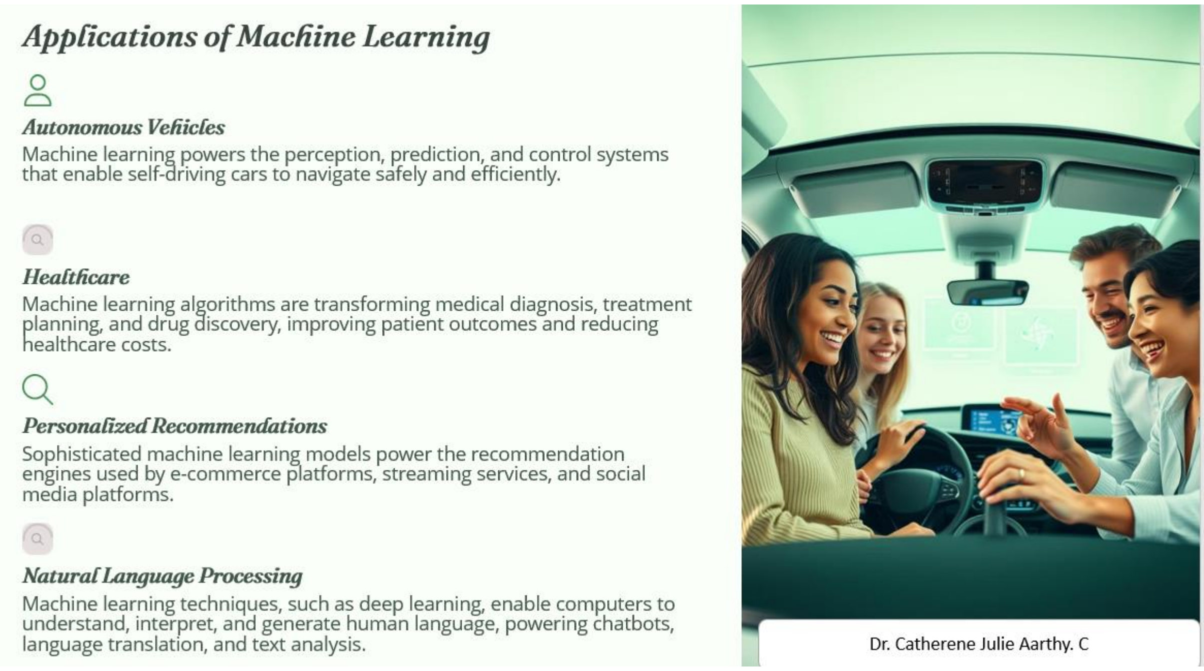
3. **Reinforcement Machine Learning**

Reinforcement learning is a type of machine learning where an agent learns to interact with an environment by performing actions and receiving rewards or penalties based on its actions. The goal of reinforcement learning is to learn a policy, which is a mapping from states to actions, that maximizes the expected cumulative reward over time.

There are two main types of reinforcement learning:

Model-based reinforcement learning: In model-based reinforcement learning, the agent learns a model of the environment, including the transition probabilities between states and the rewards associated with each state-action pair. The agent then uses this model to plan its actions to maximize its expected reward. Some popular model-based reinforcement learning algorithms include Value Iteration and Policy Iteration.

Model-free reinforcement learning: In model-free reinforcement learning, the agent learns a policy directly from experience without explicitly building a model of the environment. The agent interacts with the environment and updates its policy based on the rewards it receives. Some popular model-free reinforcement learning algorithms include Q-Learning, SARSA, and Deep Reinforcement Learning.



Applications of Machine Learning

Autonomous Vehicles

Machine learning powers the perception, prediction, and control systems that enable self-driving cars to navigate safely and efficiently.

Healthcare

Machine learning algorithms are transforming medical diagnosis, treatment planning, and drug discovery, improving patient outcomes and reducing healthcare costs.

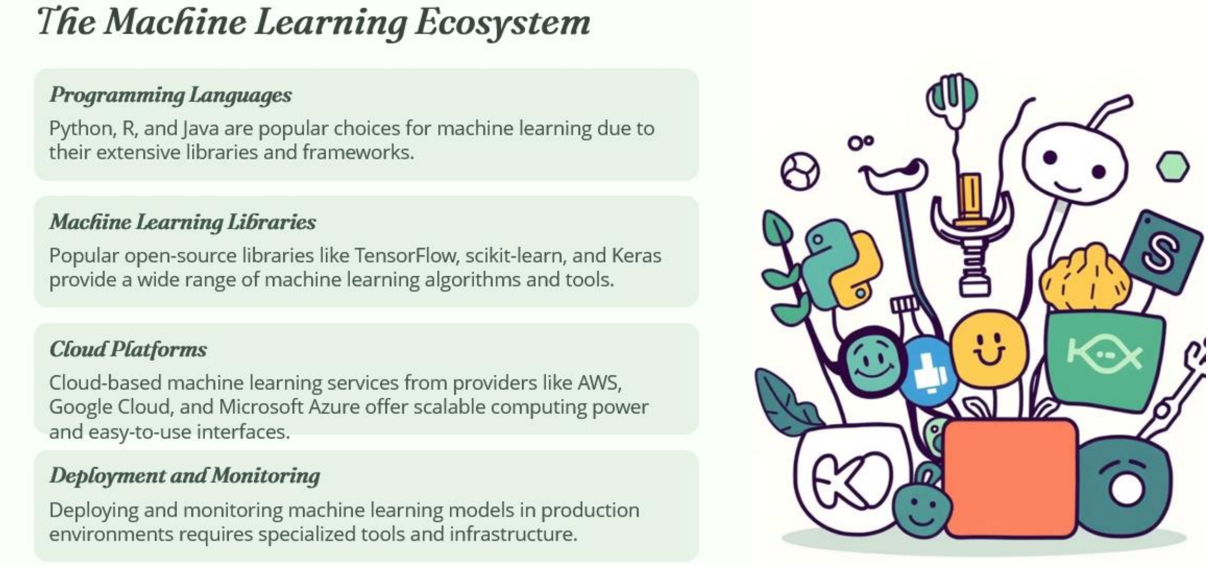
Personalized Recommendations

Sophisticated machine learning models power the recommendation engines used by e-commerce platforms, streaming services, and social media platforms.

Natural Language Processing

Machine learning techniques, such as deep learning, enable computers to understand, interpret, and generate human language, powering chatbots, language translation, and text analysis.

Dr. Catherene Julie Aarthy. C



The Machine Learning Ecosystem

Programming Languages

Python, R, and Java are popular choices for machine learning due to their extensive libraries and frameworks.

Machine Learning Libraries

Popular open-source libraries like TensorFlow, scikit-learn, and Keras provide a wide range of machine learning algorithms and tools.

Cloud Platforms

Cloud-based machine learning services from providers like AWS, Google Cloud, and Microsoft Azure offer scalable computing power and easy-to-use interfaces.

Deployment and Monitoring

Deploying and monitoring machine learning models in production environments requires specialized tools and infrastructure.

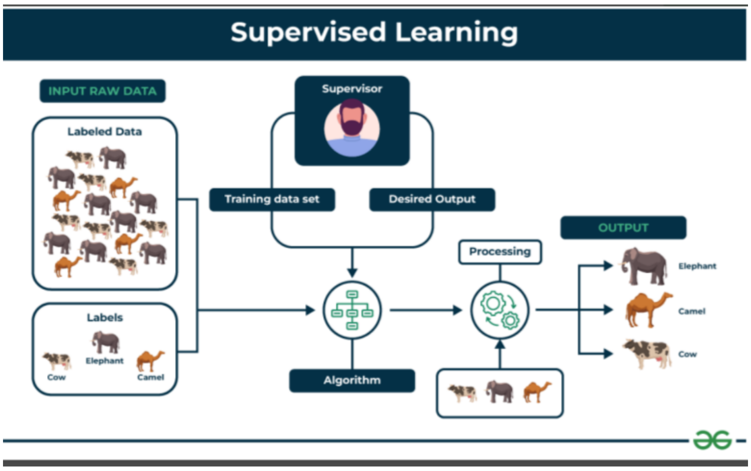
Limitations of Machine Learning-

- The primary challenge of machine learning is the lack of data or the diversity in the dataset.
- A machine cannot learn if there is no data available. Besides, a dataset with a lack of diversity gives the machine a hard time.
- A machine needs to have heterogeneity to learn meaningful insight.
- It is rare that an algorithm can extract information when there are no or few variations.
- It is recommended to have at least 20 observations per group to help the machine learn. This constraint leads to poor evaluation and prediction.

Conclusion

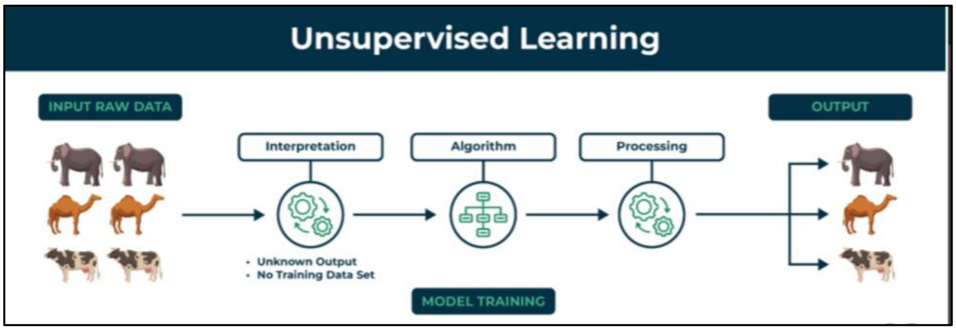
In conclusion, understanding what is machine learning opens the door to a world where computers not only process data but learn from it to make decisions and predictions. It represents the intersection of computer science and statistics, enabling systems to improve their performance over time without explicit programming. As machine learning continues to evolve, its applications across industries promise to redefine how we interact with technology, making it not just a tool but a transformative force in our daily lives.

Supervised Learning



Unsupervised Learning

What is unsupervised learning when there are no defined independent and dependent variables patterns in the data are used to identify groups that have civil or observations when we do not have a single column that is required to be a predictor in the future or which cannot be used in the future it comes under unsupervised it we have to differentiate between supervised and unsupervised learning. Supervised learning has clearly defined X and Y variables that are dependent on one or more independent variables you can predict the continuous response which comes under regression or classification. In unsupervised learning, we have unlabelled data. You cannot separate them into dependent and independent variables. Emerging patterns based on similarity, are identified by a clustering algorithm or association rules algorithm in a specific market basket analysis.



Key Points

- Unsupervised learning allows the model to discover patterns and relationships in unlabeled data.
- Clustering algorithms group similar data points together based on their inherent characteristics.

- Feature extraction captures essential information from the data, enabling the model to make meaningful distinctions.
- Label association assigns categories to the clusters based on the extracted patterns and characteristics.

Example

Imagine you have a machine learning model trained on a large dataset of unlabeled images, containing both dogs and cats. The model has never seen an image of a dog or cat before, and it has no pre-existing labels or categories for these animals. Your task is to use unsupervised learning to identify the dogs and cats in a new, unseen image.

For instance, suppose it is given an image having both dogs and cats which it has never seen.

Thus the machine has no idea about the features of dogs and cats so we can't categorize it as 'dogs and cats '. But it can categorize them according to their similarities, patterns, and differences, i.e., we can easily categorize the above picture into two parts. The first may contain all pics having **dogs** in them and the second part may contain all pics having **cats** in them. Here you didn't learn anything before, which means no training data or examples.

It allows the model to work on its own to discover patterns and information that was previously undetected. It mainly deals with unlabelled data.

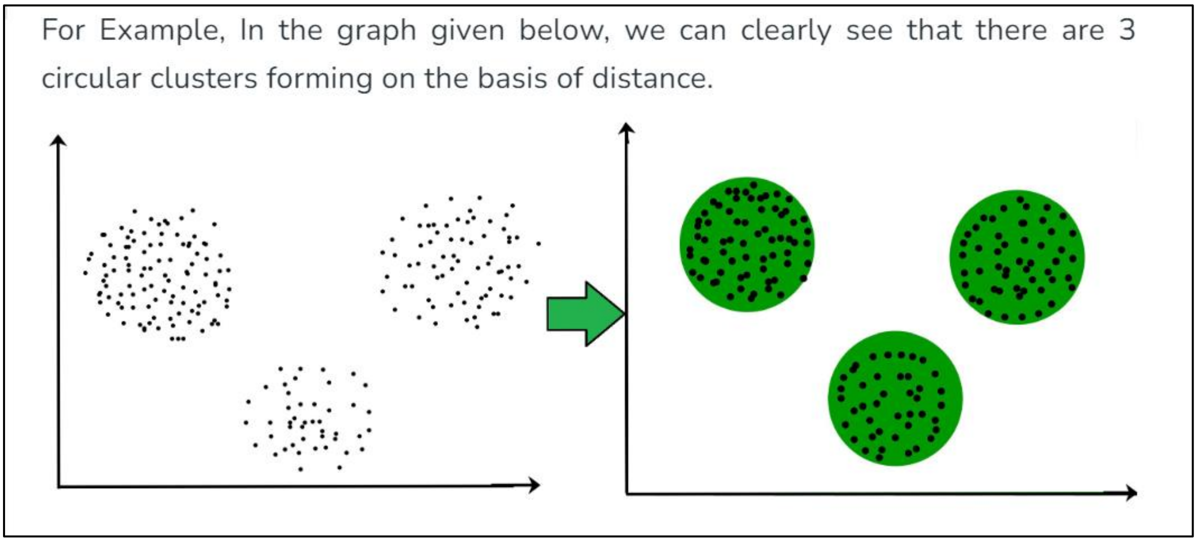
Types of Unsupervised Learning

Unsupervised learning is classified into two categories of algorithms:

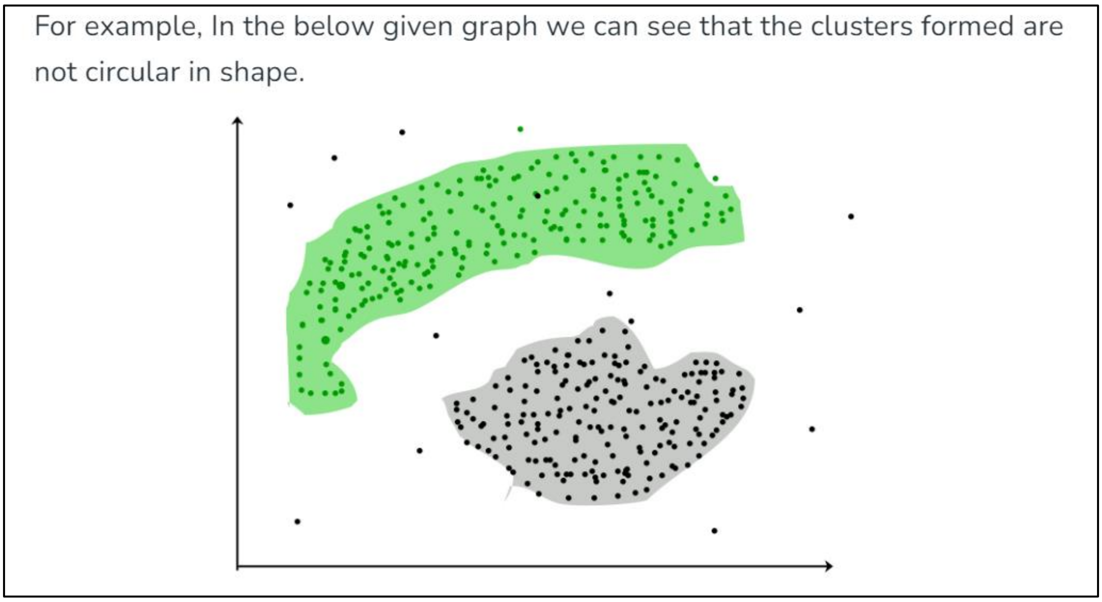
- **Clustering**: A clustering problem is where you want to discover the inherent groupings in the data, such as grouping customers by purchasing behavior.
- **Association**: An association rule learning problem is where you want to discover rules that describe large portions of your data, such as people that buy X also tend to buy Y.

What is clustering? clustering on an overall level trying to identify groups of similar objects that are highly similar to other objects heterogeneity between the group and homogeneous within the group that is a single concept in the class string technique followed. Different approaches are taken by each of the clustering techniques. The core understanding is to end up with clusters that are homogenous within the group. In simple terms when clustering is performed within a data set, we say that between group variance i.e between the different clusters if we compute the variance between one cluster and another cluster, we can find how far away they are from each other.

This distance should be greater than the variance within the clusters so that some of the squares of between-group variance should be greater than some of the squares of within-group variance. This is the objective that the clustering technique tries to apply and perform.



For Example, In the graph given below, we can clearly see that there are 3 circular clusters forming on the basis of distance.



For example, In the below given graph we can see that the clusters formed are not circular in shape.

The classical understanding of unsupervised learning is the identification of groups. Intra-cluster distances are minimised i.e. the distance between observations within a cluster is minimised, and Inter-cluster distance is maximised i.e. distance between an observation on one cluster to another observation on another cluster.

Minimisation of intra-cluster distance and maximization of inter-cluster distance is the objective of clustering.

What is the need for clustering? The need for clustering is multifold (many) . Sometimes we would like to find out what are the different patterns of similarity are that exist within the data. e.g. for customer segmentation problems we might want to identify customers who exhibit similar behavior so that we can conduct marketing activities on different behavior observations that we notice on various customer data so once a customer segmentation is complete we would be able to conduct target at marketing activities to the difference segment of customers that would be one of the reasons why we do class studying sometimes, we might want to identify patterns of similarities and dissimilarities through visualization. Visualisation of data sets gives us some additional insights into the data. Sometimes clustering is also a preliminary step before a supervised learning technique can be applied. Specifically, once we do clustering, we can identify a label column or a cluster-level column in addition to the existing columns of the data that cluster-labelled column will be a discrete variable. This discrete variable can be used as a target for the dependent variable for a supervised learning method, especially a classification technique. So sometimes clustering techniques are the research problem themselves or sometimes it is a preprocessing step or precursor to a future research problem or probably a supervised learning technique.

There are multiple use cases when it comes to clustering. So in image processing - we might want to understand images and their similarities based on visual content. When a company like Google or Facebook tries to perform cluster analysis on images they come up with patterns of similarities or dissimilarities. That is how google photos can give us collages that have similar images. This is one of the practical examples of image processing.

In the web and internet, we can think of groups of access patterns on web pages, the heatmap usage, the mouse movements, trigger actions, types of purchases they make, the type of browsing behavior that every customer has, the cookie information tracked by various websites on every customer, etc. These are some of the certain trace points on the web using which you can identify customers who have similar patterns, and similar behavior at the end of the day.

Uses of Clustering

Now, before we begin with types of clustering algorithms, we will go through the use cases of Clustering algorithms. Clustering algorithms are mainly used for:

Market Segmentation – Businesses use clustering to group their customers and use targeted advertisements to attract more customers.

Market Basket Analysis – Shop owners analyse their sales and figure out which items are bought together by the customers. For example, In the USA, according to a study, diapers and beer were usually bought together by fathers.

Social Network Analysis – Social media sites use your data to understand your browsing behaviour and provide you with targeted friend recommendations or content recommendations.

Medical Imaging – Doctors use Clustering to find diseased areas in diagnostic images like X-rays.

Anomaly Detection – We can find outliers in a stream of real-time datasets or forecast fraudulent transactions using clustering.

Simplify working with large datasets – After clustering is complete, each cluster is given a cluster ID. Now, you may reduce a feature set's whole feature set into its cluster-ID. Clustering is effective when representing a complicated case with a straightforward cluster ID. Using the same principle, clustering data can make complex datasets simpler.

There are many more use cases for clustering but there are some of the major and common use cases of clustering. Moving forward, we will discuss Clustering Algorithms that will help you perform the above tasks.

Types of Clustering Algorithms

At the surface level, clustering helps in the analysis of unstructured data. Graphing, the shortest distance, and the density of the data points are a few of the elements that influence cluster formation. Clustering is the process of determining how related the objects are based on a metric called the similarity measure. Similarity metrics are easier to locate in smaller sets of features. It gets harder to create similarity measures as the number of features increases. Depending on the type of clustering algorithm being utilized in data mining, several techniques are employed to group the data from the datasets. In this part, the clustering techniques are described. Various types of clustering algorithms are:

- Centroid-based Clustering (Partitioning methods)
- Density-based Clustering (Model-based methods)
- Connectivity-based Clustering (Hierarchical clustering)
- Distribution-based Clustering

We will be going through each of these types in brief.

1. Centroid-based Clustering (Partitioning methods)

Partitioning methods are the easiest clustering algorithms. They group data points based on their closeness. Generally, the similarity measures chosen for these algorithms are Euclidian distance, Manhattan Distance or Minkowski Distance. The datasets are separated into a predetermined number of clusters, and each cluster is referenced by a vector of values. When compared to the vector value, the input data variable shows no difference and joins the cluster.

The primary drawback for these algorithms is the requirement that we establish the number of clusters, “k,” either intuitively or scientifically (using the Elbow Method) before any clustering machine learning system starts allocating the data points. Despite this, it is still the most popular type of clustering. K-means and K-medoid clustering are some examples of this type of clustering.

2. Density-based Clustering (Model-based methods)

Density-based clustering, a model-based method, finds groups based on the density of data points. Contrary to centroid-based clustering, which requires that the number of clusters be predefined and is sensitive to initialization, density-based clustering determines the number of clusters automatically and is less susceptible to beginning positions. They are great at handling clusters of different sizes and forms, making them ideally suited for datasets with irregularly shaped or overlapping clusters. These methods manage both dense and sparse data regions by focusing on local density and can distinguish clusters with a variety of morphologies.

In contrast, centroid-based grouping, like k-means, has trouble finding arbitrarily shaped clusters. Due to its preset number of cluster requirements and extreme sensitivity to the initial positioning of centroids, the outcomes can vary. Furthermore, the tendency of centroid-based approaches to produce spherical or convex clusters restricts their capacity to handle complicated or irregularly shaped clusters. In conclusion, density-based clustering overcomes the drawbacks of centroid-based techniques by autonomously choosing cluster sizes, being resilient to initialization, and successfully capturing clusters of various sizes and forms. The most popular density-based clustering algorithm is DBSCAN.

3. Connectivity-based Clustering (Hierarchical clustering)

A method for assembling related data points into hierarchical clusters is called hierarchical clustering. Each data point is initially taken into account as a separate cluster, which is subsequently combined with the clusters that are the most similar to form one large cluster that contains all of the data points.

Think about how you may arrange a collection of items based on how similar they are. Each object begins as its own cluster at the base of the tree when using hierarchical clustering, which creates a dendrogram, a tree-like structure. The closest pairings of clusters are then combined into larger clusters after the algorithm examines how similar the objects are to one another. When every object is in one cluster at the top of the tree, the merging process has finished. Exploring various granularity levels is one of the fun things about hierarchical clustering. To obtain a given number of clusters, you can select to cut the dendrogram at a particular height. The more similar two objects are within a cluster, the closer they are. It's comparable to classifying items according to their family trees, where the nearest relatives are clustered together and the wider branches signify more general connections. There are 2 approaches for Hierarchical clustering:

Divisive Clustering: It follows a top-down approach, here we consider all data points to be part of one big cluster and then this cluster is divided into smaller groups.

Agglomerative Clustering: It follows a bottom-up approach, here we consider all data points to be part of individual clusters and then these clusters are clubbed together to make one big cluster with all data points.

4. Distribution-based Clustering

Using distribution-based clustering, data points are generated and organized according to their propensity to fall into the same probability distribution (such as a Gaussian, binomial, or other) within the data. The data elements are grouped using a probability-based distribution that is based on statistical distributions. Included are data objects that have a higher likelihood of being in the cluster. A data point is less likely to be included in a cluster the further it is from the cluster's central point, which exists in every cluster.

A notable drawback of density and boundary-based approaches is the need to specify the clusters a priori for some algorithms, and primarily the definition of the cluster form for the bulk of algorithms. There must be at least one tuning or hyper-parameter selected, and while doing so should be simple, getting it wrong could have unanticipated repercussions. Distribution-based clustering has a definite advantage over proximity and centroid-based clustering approaches in terms of flexibility, accuracy, and cluster structure. The key issue is that, to avoid overfitting, many clustering methods only work with simulated or manufactured data, or when the bulk of the data points certainly belong to a preset distribution. The most popular distribution-based clustering algorithm is the Gaussian Mixture Model.

Applications of Clustering in different fields:

Marketing: It can be used to characterize & discover customer segments for marketing purposes.

Biology: It can be used to classify different species of plants and animals.

Libraries: It is used to classify different books based on topics and information.

Insurance: It is used to acknowledge the customers, and their policies and identify the frauds.

City Planning: It is used to make groups of houses and to study their values based on their geographical locations and other factors present.

Earthquake studies: By learning the earthquake-affected areas we can determine the dangerous zones.

Image Processing: Clustering can be used to group similar images together, classify images based on content, and identify patterns in image data.

Genetics: Clustering is used to group genes that have similar expression patterns and identify gene networks that work together in biological processes.

Finance: Clustering is used to identify market segments based on customer behavior, identify patterns in stock market data, and analyze risk in investment portfolios.

Customer Service: Clustering is used to group customer inquiries and complaints into categories, identify common issues, and develop targeted solutions.

Manufacturing: Clustering is used to group similar products together, optimize production processes, and identify defects in manufacturing processes.

Medical diagnosis: Clustering is used to group patients with similar symptoms or diseases, which helps in making accurate diagnoses and identifying effective treatments.

Fraud detection: Clustering is used to identify suspicious patterns or anomalies in financial transactions, which can help in detecting fraud or other financial crimes.

Traffic analysis: Clustering is used to group similar patterns of traffic data, such as peak hours, routes, and speeds, which can help in improving transportation planning and infrastructure.

Social network analysis: Clustering is used to identify communities or groups within social networks, which can help in understanding social behavior, influence, and trends.

Cybersecurity: Clustering is used to group similar patterns of network traffic or system behavior, which can help in detecting and preventing cyberattacks.

Climate analysis: Clustering is used to group similar patterns of climate data, such as temperature, precipitation, and wind, which can help in understanding climate change and its impact on the environment.

Sports analysis: Clustering is used to group similar patterns of player or team performance data, which can help in analyzing player or team strengths and weaknesses and making strategic decisions.

Crime analysis: Clustering is used to group similar patterns of crime data, such as location, time, and type, which can help in identifying crime hotspots, predicting future crime trends, and improving crime prevention strategies.

Conclusion

In this we have discussed Clustering, its types, and its applications in the real world. There is much more to be covered in unsupervised learning and Cluster Analysis is just the first step. This article can help you get started with Clustering algorithms and help you get a new project that can be added to your portfolio.

Frequently Asked Questions (FAQs) on Clustering

Q. What is the best clustering method?

The top 10 clustering algorithms are:

1. *K-means Clustering*
2. *Hierarchical Clustering*
3. *DBSCAN (Density-Based Spatial Clustering of Applications with Noise)*
4. *Gaussian Mixture Models (GMM)*
5. *Agglomerative Clustering*
6. *Spectral Clustering*
7. *Mean Shift Clustering*
8. *Affinity Propagation*
9. *OPTICS (Ordering Points to Identify the Clustering Structure)*
10. *Birch (Balanced Iterative Reducing and Clustering using Hierarchies)*

Q. What is the difference between clustering and classification?

The main difference between clustering and classification is that classification is a supervised learning algorithm and clustering is an unsupervised learning algorithm. That is, we apply clustering to those datasets without a target variable.

Q. What are the advantages of clustering analysis?

Data can be organized into meaningful groups using the strong analytical tool of cluster analysis. You can use it to pinpoint segments, find hidden patterns, and improve decisions.

Q. Which is the fastest clustering method?

K-means clustering is often considered the fastest method due to its simplicity and computational efficiency. It iteratively assigns data points to the nearest cluster centroid, making it suitable for large datasets with low dimensionality and a moderate number of clusters.

Q. What are the limitations of clustering?

Limitations of clustering include sensitivity to initial conditions, dependence on the choice of parameters, difficulty in determining the optimal number of clusters, and challenges handling high-dimensional or noisy data.

Q. What does the quality of the result of clustering depend on?

The quality of clustering results depends on factors such as the choice of algorithm, distance metric, number of clusters, initialization method, data preprocessing techniques, cluster evaluation metrics, and domain knowledge. These elements collectively influence the effectiveness and accuracy of the clustering outcome.

Evaluating Non-Supervised Learning Models

Evaluating non-supervised learning models is an important step in ensuring that the model is effective and useful. However, it can be more challenging than evaluating supervised learning models, as there is no ground truth data to compare the model's predictions to.

There are a number of different metrics that can be used to evaluate non-supervised learning models, but some of the most common ones include:

Silhouette score: The silhouette score measures how well each data point is clustered with its own cluster members and separated from other clusters. It ranges from -1 to 1, with higher scores indicating better clustering.

Calinski-Harabasz score: The Calinski-Harabasz score measures the ratio between the variance between clusters and the variance within clusters. It ranges from 0 to infinity, with higher scores indicating better clustering.

Adjusted Rand index: The adjusted Rand index measures the similarity between two clusterings. It ranges from -1 to 1, with higher scores indicating more similar clusterings.

Davies-Bouldin index: The Davies-Bouldin index measures the average similarity between clusters. It ranges from 0 to infinity, with lower scores indicating better clustering.

F1 score: The F1 score is a weighted average of precision and recall, which are two metrics that are commonly used in supervised learning to evaluate classification models. However, the F1 score can also be used to evaluate non-supervised learning models, such as clustering models.

K means Clustering – Introduction

K-Means Clustering is an Unsupervised Machine Learning algorithm, which groups the unlabeled dataset into different clusters. The article aims to explore the fundamentals and workings of k mean clustering along with the implementation.

Table of Content

What is K-means Clustering?

Unsupervised Machine Learning is the process of teaching a computer to use unlabeled, unclassified data and enabling the algorithm to operate on that data without supervision. Without any previous data training, the machine’s job in this case is to organize unsorted data according to parallels, patterns, and variations.

K means clustering, assigns data points to one of the K clusters depending on their distance from the center of the clusters. It starts by randomly assigning the clusters centroid in the space. Then each data point is assigned to one of the clusters based on its distance from the centroid of the cluster. After assigning each point to one of the clusters, new cluster centroids are assigned. This process runs iteratively until it finds a good cluster. In the analysis, we assume that a number of clusters is given in advance and we have to put points in one of the groups.

In some cases, K is not clearly defined, and we have to think about the optimal number of K. K Means clustering performs best data is well separated. When data points overlap, this clustering is not suitable. K Means is faster as compared to other clustering techniques. It provides a strong coupling between the data points. K Means clusters do not provide clear information regarding the quality of clusters. Different initial assignments of cluster centroids may lead to different clusters. Also, the K Means algorithm is sensitive to noise. It may have stuck in local minima.

What is the objective of k-means clustering?

The goal of clustering is to divide the population or set of data points into a number of groups so that the data points within each group are more comparable to one another and different from the data points within the other groups. It is essentially a grouping of things based on how similar and different they are to one another.

How does K-means clustering work?

We are given a data set of items, with certain features, and values for these features (like a vector). The task is to categorize those items into groups. To achieve this, we will use the K-means algorithm, an unsupervised learning algorithm. 'K' in the name of the algorithm represents the number of groups/clusters we want to classify our items into.

(It will help if you think of items as points in an n-dimensional space). The algorithm will categorize the items into k groups or clusters of similarity. To calculate that similarity, we will use the Euclidean distance as a measurement.

The algorithm works as follows:

First, we randomly initialize k points, called means or cluster centroids.

We categorize each item to its closest mean, and we update the mean's coordinates, which are the averages of the items categorized in that cluster so far.

We repeat the process for a given number of iterations and at the end, we have our clusters.

The "points" mentioned above are called means because they are the mean values of the items categorized in them. To initialize these means, we have a lot of options. An intuitive method is to initialize the means at random items in the data set. Another method is to initialize the means at random values between the boundaries of the data set (if for a feature x, the items have values in [0,3], we will initialize the means with values for x at [0,3]).

The above algorithm in pseudocode is as follows:

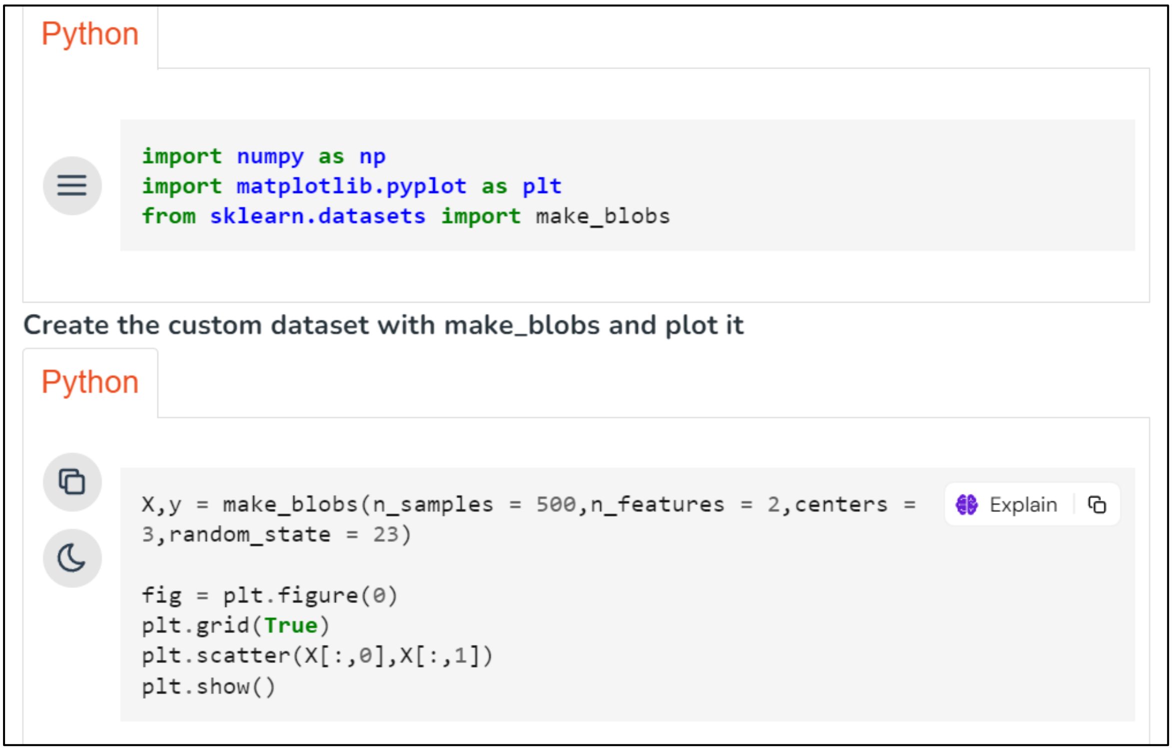
```
Initialize k means with random values
--> For a given number of iterations:

    --> Iterate through items:

        --> Find the mean closest to the item by calculating
        the euclidean distance of the item with each of the means

        --> Assign item to mean

        --> Update mean by shifting it to the average of the items
in that cluster
```

Implementation of K-Means Clustering in Python

Example 1

Import the necessary Libraries

We are importing Numpy for statistical computations, Matplotlib to plot the graph, and make_blobs from sklearn datasets.

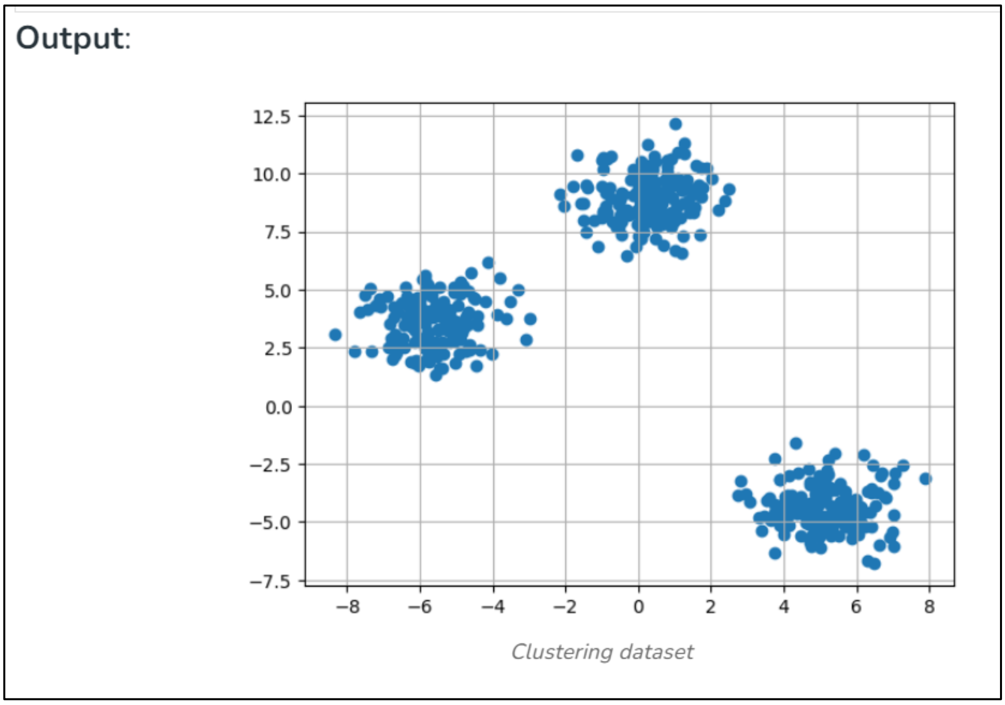


```python
import numpy as np
import matplotlib.pyplot as plt
from sklearn.datasets import make_blobs
```

Create the custom dataset with make_blobs and plot it

```python
X,y = make_blobs(n_samples = 500,n_features = 2,centers = 
3,random_state = 23)

fig = plt.figure(0)
plt.grid(True)
plt.scatter(X[:,0],X[:,1])
plt.show()
```

Output:

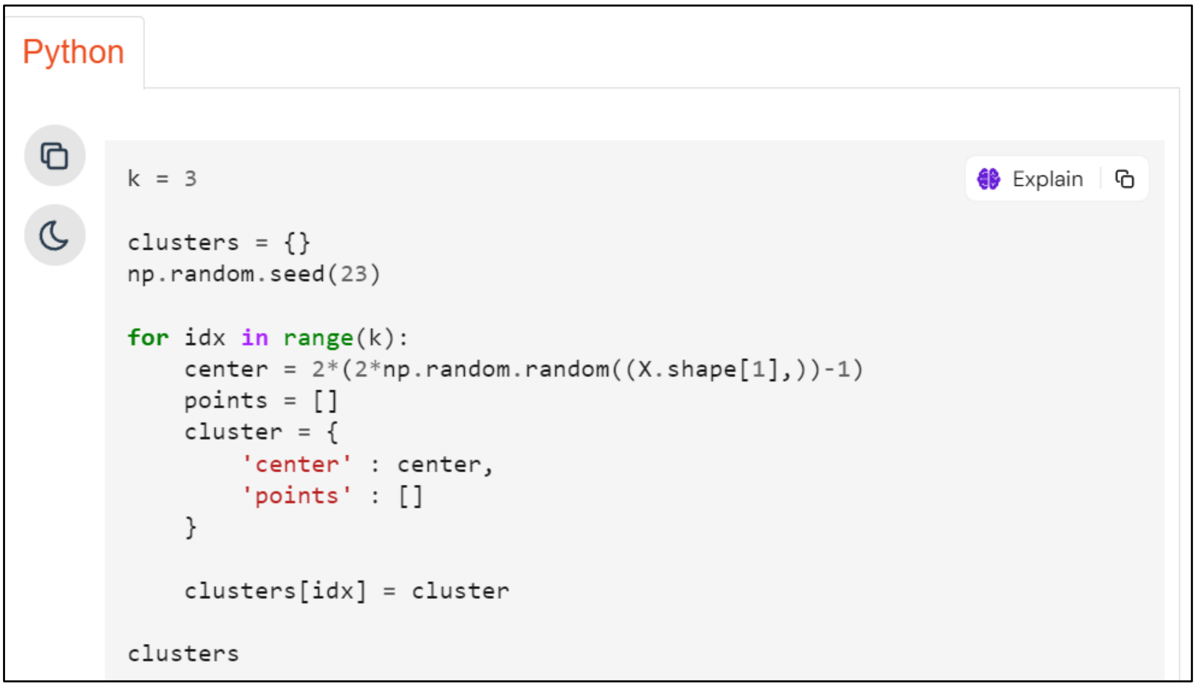


Output:

Clustering dataset

Initialize the random centroids

The code initializes three clusters for K-means clustering. It sets a random seed and generates random cluster centers within a specified range, and creates an empty list of points for each cluster.

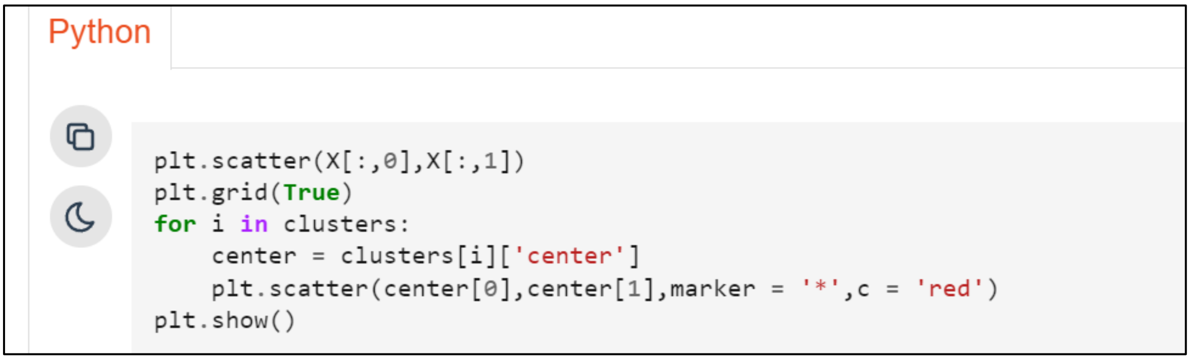


```
k = 3

clusters = {}
np.random.seed(23)

for idx in range(k):
    center = 2*(2*np.random.random((X.shape[1],))-1)
    points = []
    cluster = {
        'center' : center,
        'points' : []
    }

    clusters[idx] = cluster

clusters
```

Output:

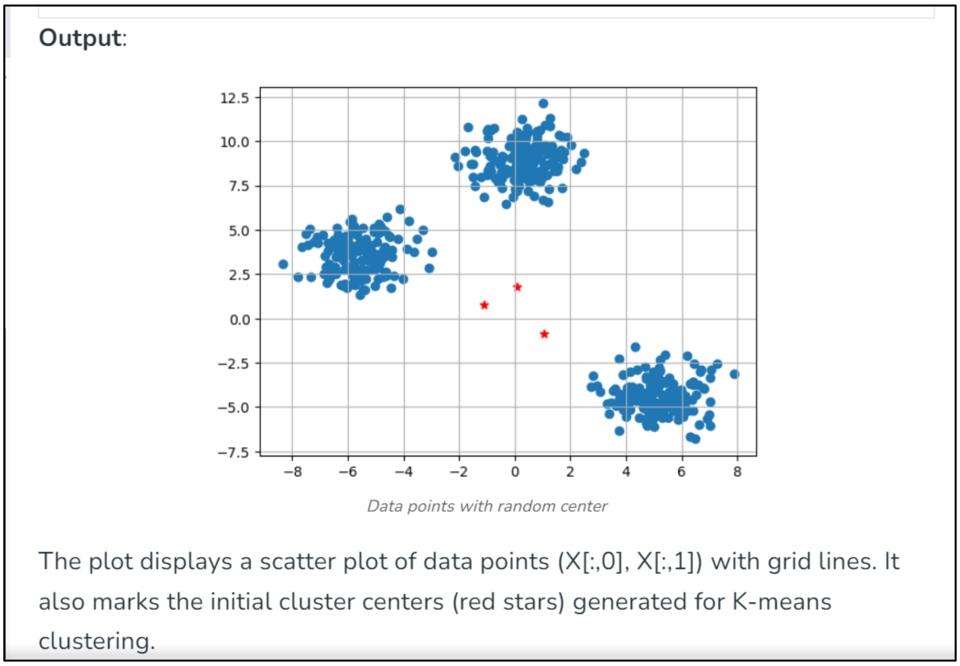
```
{0: {'center': array([0.06919154, 1.78785042]), 'points': []},

1: {'center': array([ 1.06183904, -0.87041662]), 'points': []},

2: {'center': array([-1.11581855,  0.74488834]), 'points': []}}
```

Plot the random initialize center with data points



```
plt.scatter(X[:,0],X[:,1])
plt.grid(True)
for i in clusters:
    center = clusters[i]['center']
    plt.scatter(center[0],center[1],marker = '*',c = 'red')
plt.show()
```

Output:

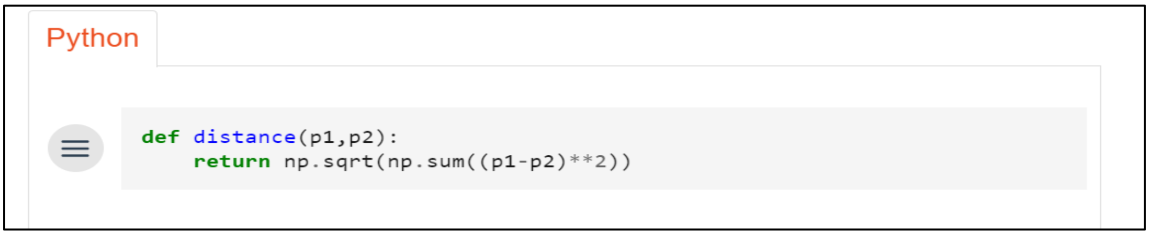


Output:

Data points with random center

The plot displays a scatter plot of data points (X[:,0], X[:,1]) with grid lines. It also marks the initial cluster centers (red stars) generated for K-means clustering.

Define Euclidean distance

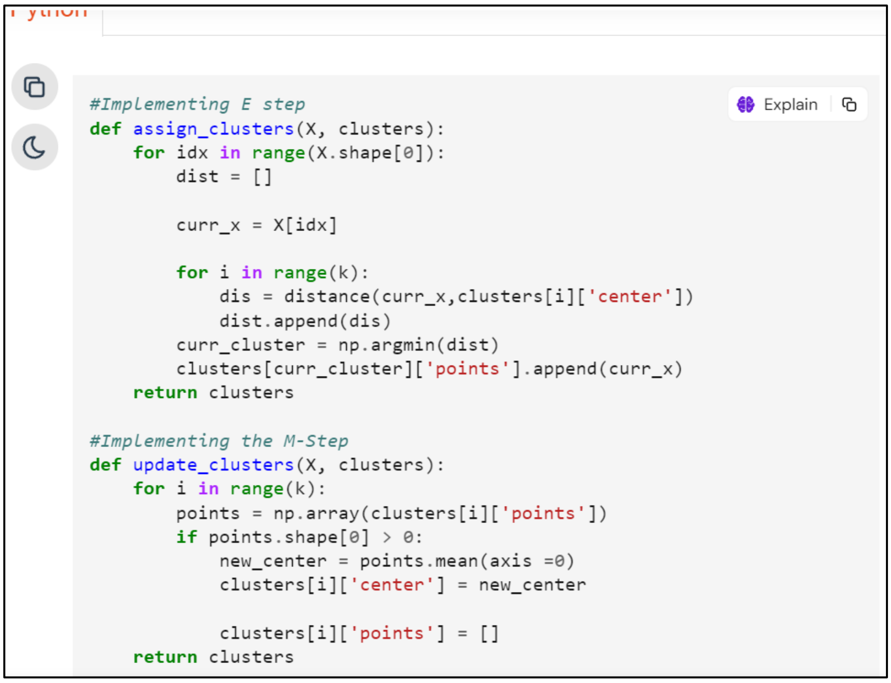


Python

```python
def distance(p1,p2):
    return np.sqrt(np.sum((p1-p2)**2))
```

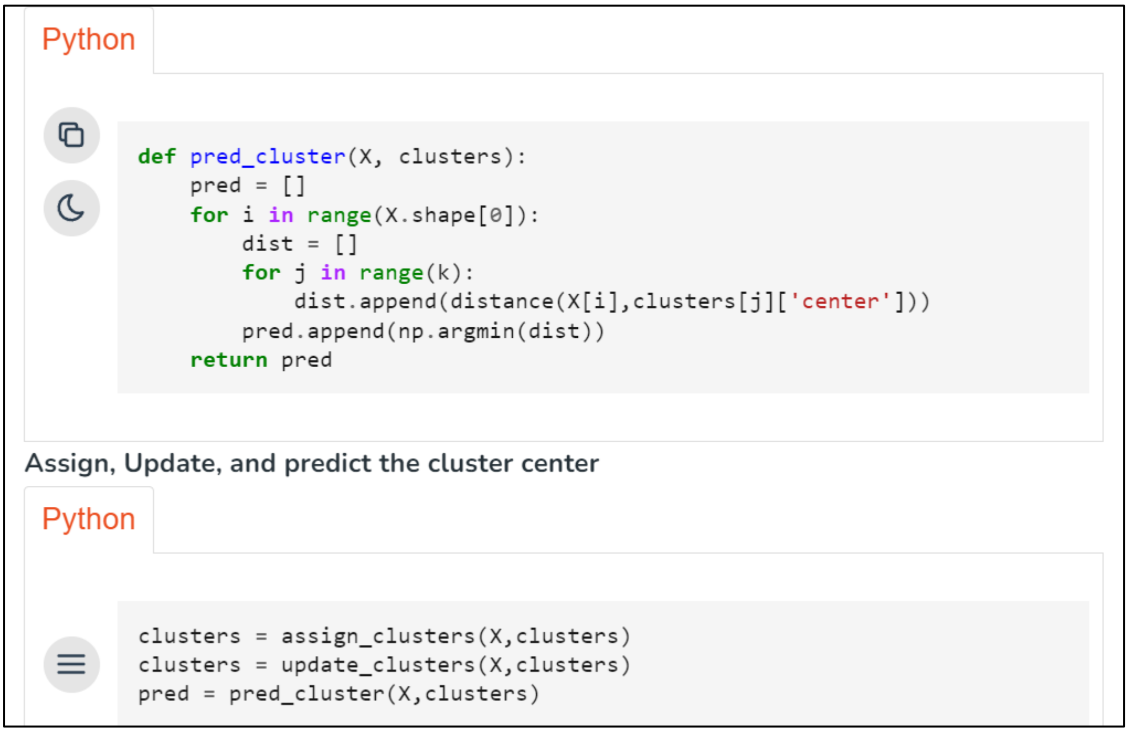
Create the function to Assign and Update the cluster center

The E-step assigns data points to the nearest cluster center, and the M-step updates cluster centers based on the mean of assigned points in K-means clustering.



```python
#Implementing E step
def assign_clusters(X, clusters):
    for idx in range(X.shape[0]):
        dist = []

        curr_x = X[idx]

        for i in range(k):
            dis = distance(curr_x,clusters[i]['center'])
            dist.append(dis)
        curr_cluster = np.argmin(dist)
        clusters[curr_cluster]['points'].append(curr_x)
    return clusters

#Implementing the M-Step
def update_clusters(X, clusters):
    for i in range(k):
        points = np.array(clusters[i]['points'])
        if points.shape[0] > 0:
            new_center = points.mean(axis =0)
            clusters[i]['center'] = new_center

            clusters[i]['points'] = []
    return clusters
```

Step 7: Create the function to Predict the cluster for the datapoints

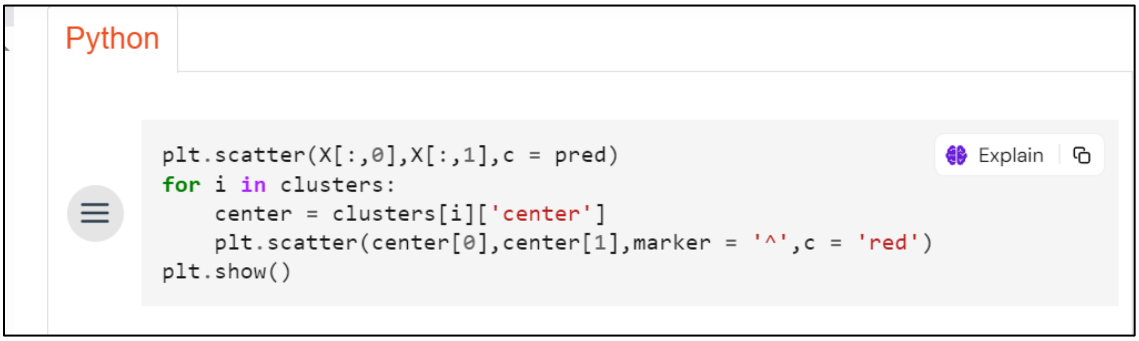


```python
def pred_cluster(X, clusters):
    pred = []
    for i in range(X.shape[0]):
        dist = []
        for j in range(k):
            dist.append(distance(X[i],clusters[j]['center']))
        pred.append(np.argmin(dist))
    return pred
```

Assign, Update, and predict the cluster center

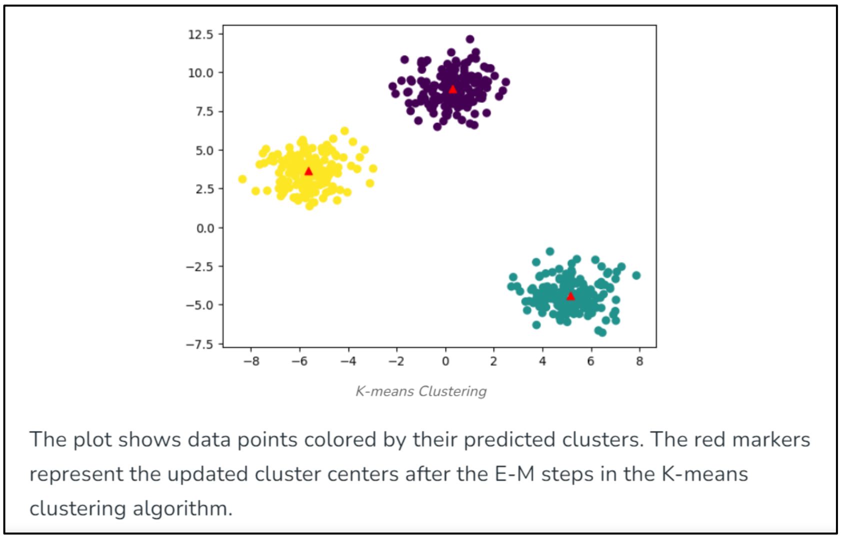
```python
clusters = assign_clusters(X,clusters)
clusters = update_clusters(X,clusters)
pred = pred_cluster(X,clusters)
```

Plot the data points with their predicted cluster center



```python
plt.scatter(X[:,0],X[:,1],c = pred)
for i in clusters:
    center = clusters[i]['center']
    plt.scatter(center[0],center[1],marker = '^',c = 'red')
plt.show()
```

Output:



K-means Clustering

The plot shows data points colored by their predicted clusters. The red markers represent the updated cluster centers after the E-M steps in the K-means clustering algorithm.

Example 2

Import the necessary libraries

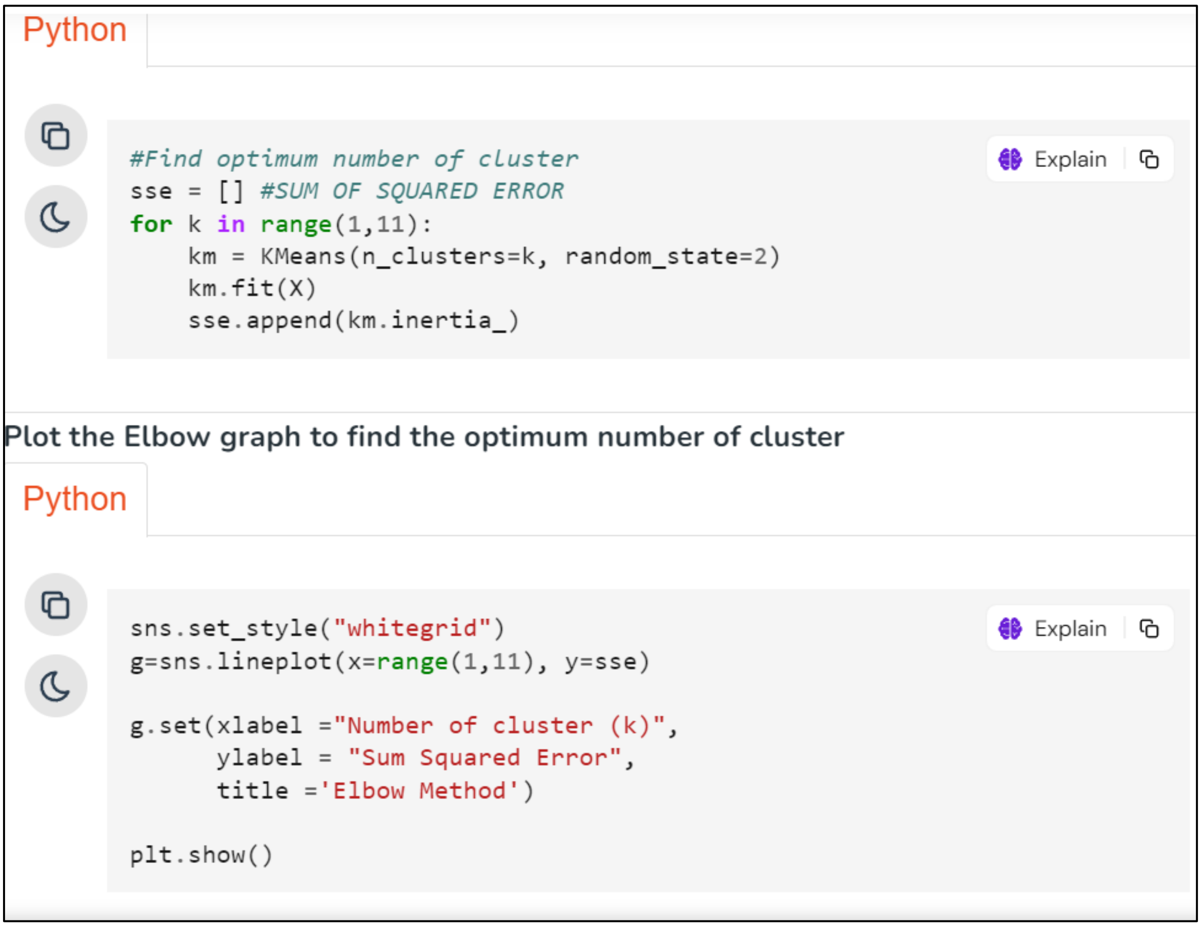
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.cm as cm
from sklearn.datasets import load_iris
from sklearn.cluster import KMeans

Load the Dataset

```
X, y = load_iris(return_X_y=True)
```

Elbow Method

Finding the ideal number of groups to divide the data into is a basic stage in any unsupervised algorithm. One of the most common techniques for figuring out this ideal value of k is the elbow approach.

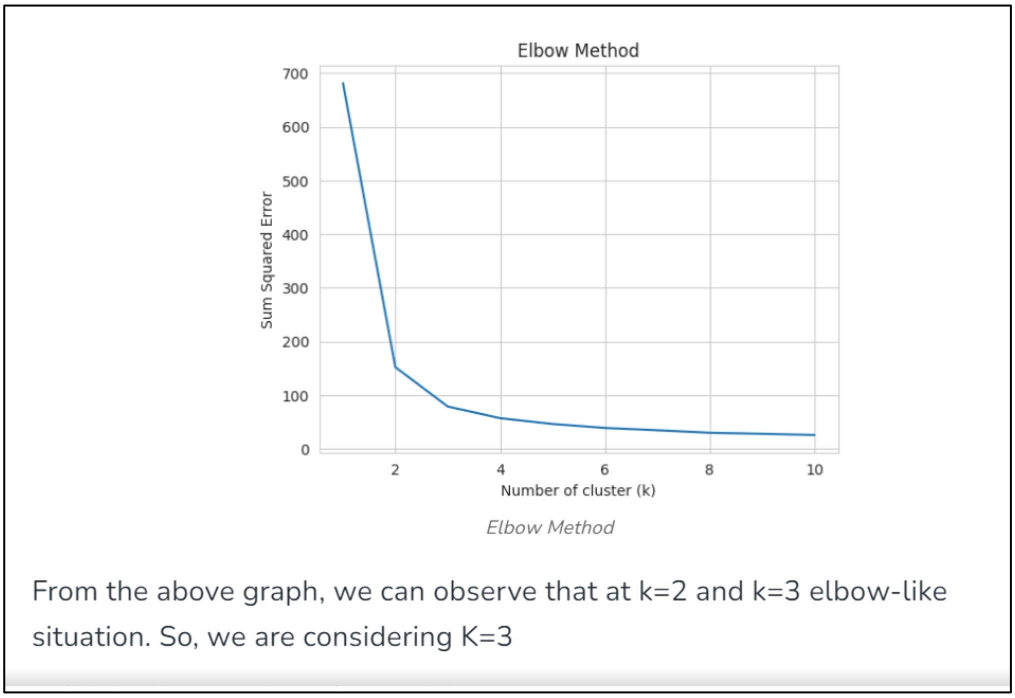


```python
#Find optimum number of cluster
sse = [] #SUM OF SQUARED ERROR
for k in range(1,11):
    km = KMeans(n_clusters=k, random_state=2)
    km.fit(X)
    sse.append(km.inertia_)
```

Plot the Elbow graph to find the optimum number of cluster

```python
sns.set_style("whitegrid")
g=sns.lineplot(x=range(1,11), y=sse)

g.set(xlabel ="Number of cluster (k)",
      ylabel = "Sum Squared Error",
      title ='Elbow Method')

plt.show()
```

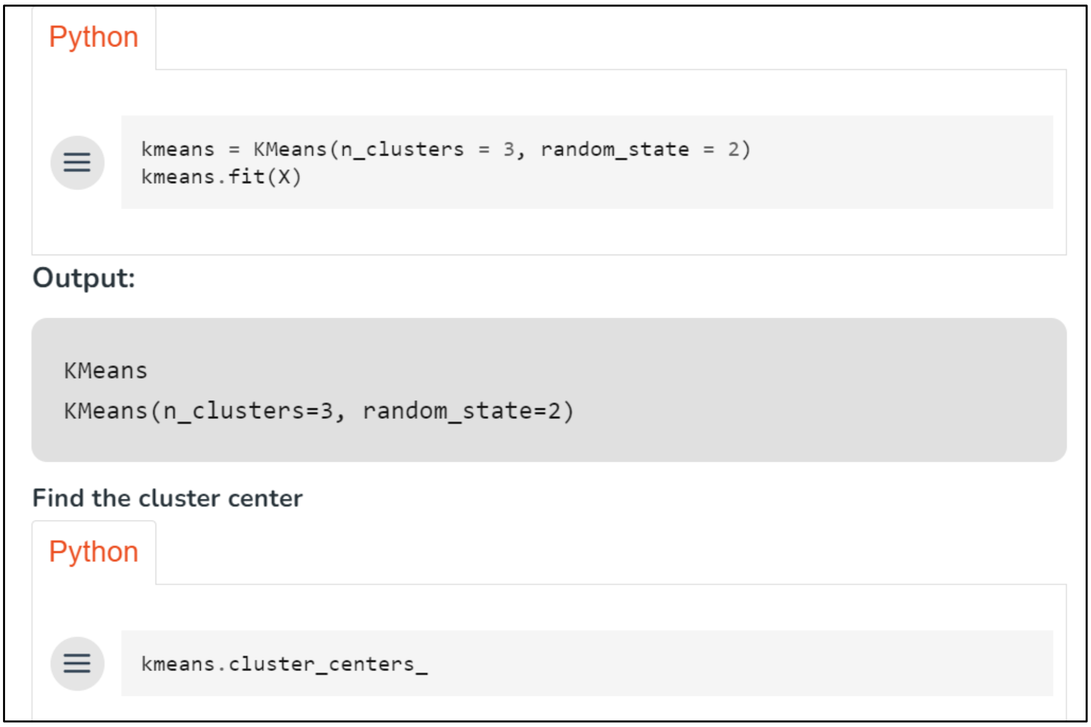
Output:



Elbow Method

From the above graph, we can observe that at k=2 and k=3 elbow-like situation. So, we are considering K=3

Build the K-means clustering model



Python

```
kmeans = KMeans(n_clusters = 3, random_state = 2)
kmeans.fit(X)
```

Output:

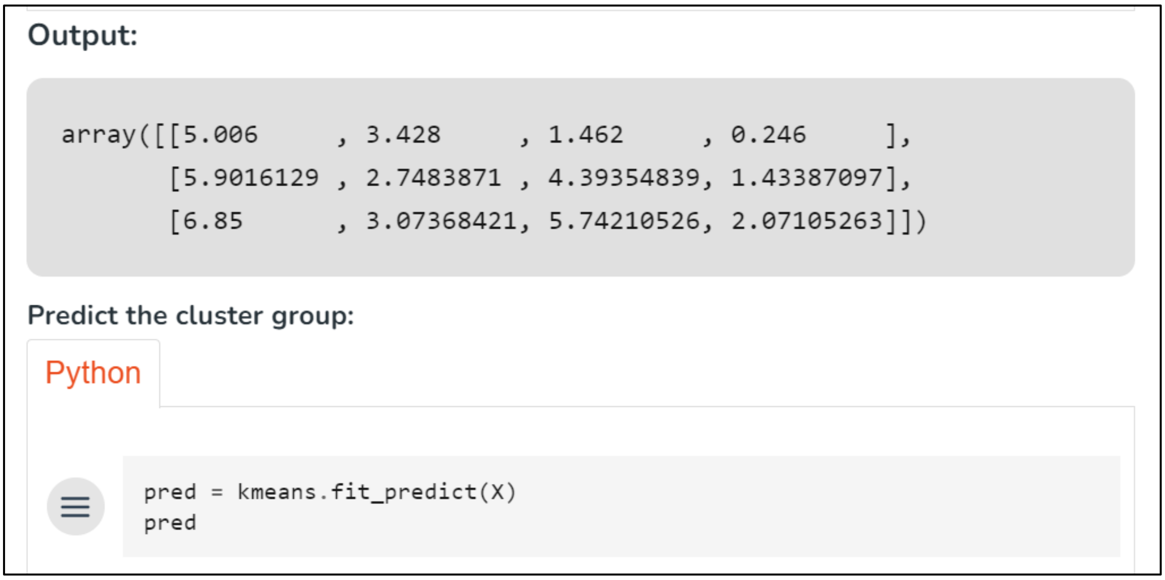
```
KMeans
KMeans(n_clusters=3, random_state=2)
```

Find the cluster center

Python

```
kmeans.cluster_centers_
```

Output:



Output:

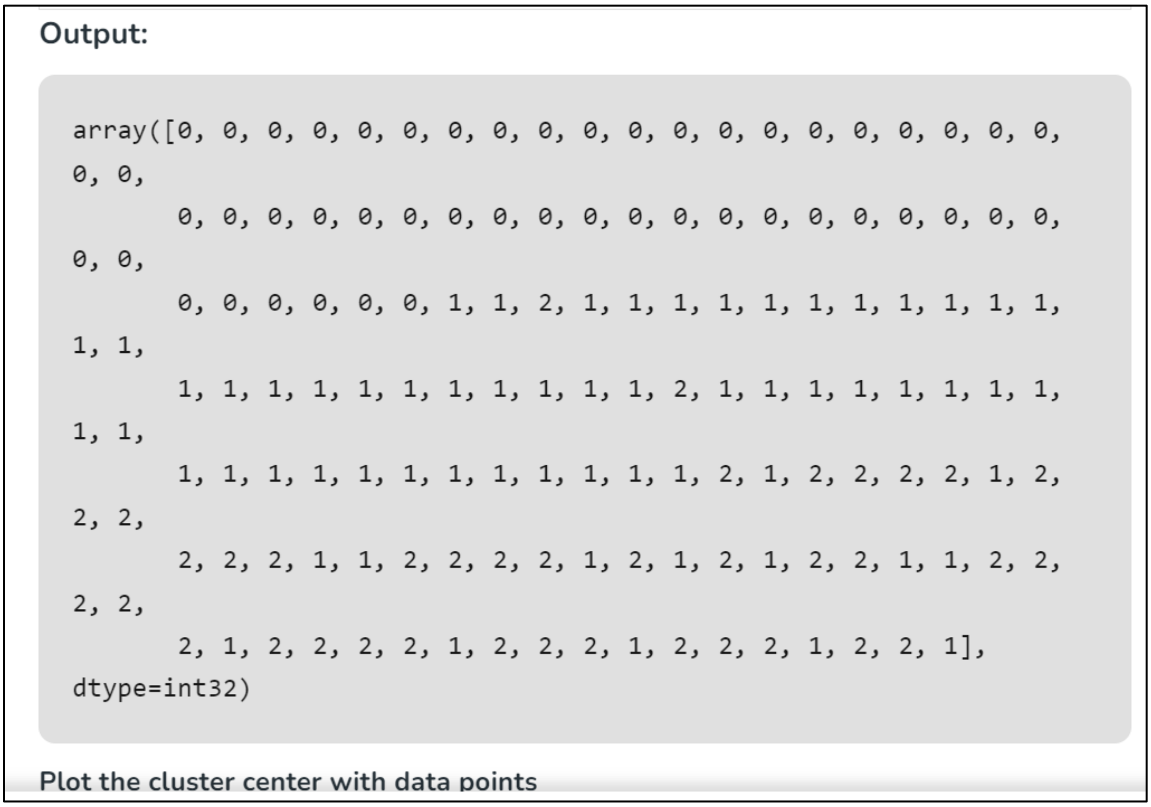
```
array([[5.006     , 3.428     , 1.462     , 0.246     ],
       [5.9016129 , 2.7483871 , 4.39354839, 1.43387097],
       [6.85      , 3.07368421, 5.74210526, 2.07105263]])
```

Predict the cluster group:

Python

```
pred = kmeans.fit_predict(X)
pred
```

Output:

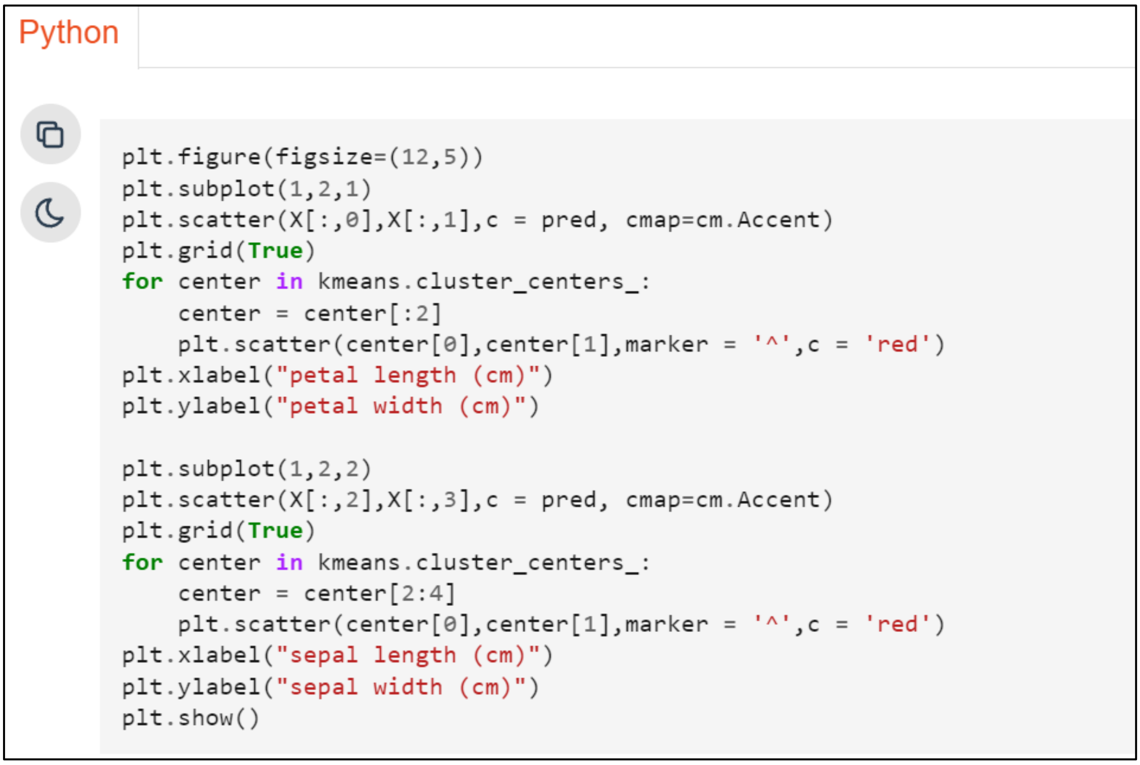


Output:

```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0,
       0, 0, 0, 0, 0, 0, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 2, 2, 2, 2, 1, 2,
2, 2,
       2, 2, 2, 1, 1, 2, 2, 2, 2, 1, 2, 1, 2, 1, 2, 2, 1, 1, 2, 2,
2, 2,
       2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 1, 2, 2, 2, 1, 2, 2, 1],
dtype=int32)
```

Plot the cluster center with data points

Plot the cluster center with data points

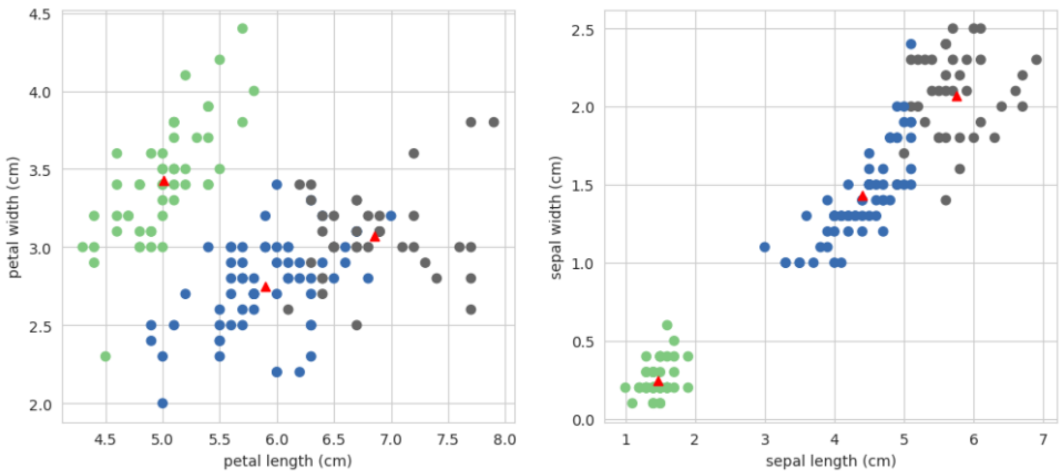


Python

```python
plt.figure(figsize=(12,5))
plt.subplot(1,2,1)
plt.scatter(X[:,0],X[:,1],c = pred, cmap=cm.Accent)
plt.grid(True)
for center in kmeans.cluster_centers_:
    center = center[:2]
    plt.scatter(center[0],center[1],marker = '^',c = 'red')
plt.xlabel("petal length (cm)")
plt.ylabel("petal width (cm)")

plt.subplot(1,2,2)
plt.scatter(X[:,2],X[:,3],c = pred, cmap=cm.Accent)
plt.grid(True)
for center in kmeans.cluster_centers_:
    center = center[2:4]
    plt.scatter(center[0],center[1],marker = '^',c = 'red')
plt.xlabel("sepal length (cm)")
plt.ylabel("sepal width (cm)")
plt.show()
```

Output:

Output:



K-means clustering

The subplot on the left display petal length vs. petal width with data points colored by clusters, and red markers indicate K-means cluster centers. The subplot on the right show sepal length vs. sepal width similarly.

Conclusion

In conclusion, K-means clustering is a powerful unsupervised machine learning algorithm for grouping unlabeled datasets. Its objective is to divide data into clusters, making similar data points part of the same group. The algorithm initializes cluster centroids and iteratively assigns data points to the nearest centroid, updating centroids based on the mean of points in each cluster.

Frequently Asked Questions (FAQs)

1. **What is k-means clustering for data analysis?**

K-means is a partitioning method that divides a dataset into ‘k’ distinct, non-overlapping subsets (clusters) based on similarity, aiming to minimize the variance within each cluster.

2. **What is an example of k-means in real life?**

Customer segmentation in marketing, where k-means groups customers based on purchasing behavior, allows businesses to tailor marketing strategies for different segments.

3. **What type of data is the k-means clustering model?**

K-means works well with numerical data, where the concept of distance between data points is meaningful. It’s commonly applied to continuous variables.

4. **Is K-means used for prediction?**

K-means is primarily used for clustering and grouping similar data points. It does not predict labels for new data; it assigns them to existing clusters based on similarity.

5**. What is the objective of k-means clustering?**

The objective is to partition data into ‘k’ clusters, minimizing the intra-cluster variance. It seeks to form groups where data points within each cluster are more similar to each other than to those in other clusters.

Supervised vs. Unsupervised Machine Learning

Parameters	Supervised machine learning	Unsupervised machine learning
Input Data	Algorithms are trained using labeled data.	Algorithms are used against data that is not labeled
Computational Complexity	Simpler method	Computationally complex
Accuracy	Highly accurate	Less accurate
No. of classes	No. of classes is known	No. of classes is not known
Data Analysis	Uses offline analysis	Uses real-time analysis of data
Algorithms used	Linear and Logistics regression,KNN Random forest, multi-class classification, decision tree, Support Vector Machine, Neural Network, etc.	K-Means clustering, Hierarchical clustering, Apriori algorithm, etc.
Output	Desired output is given.	Desired output is not given.
Training data	Use training data to infer model.	No training data is used.

Parameters	Supervised machine learning	Unsupervised machine learning
Complex model	It is not possible to learn larger and more complex models than with supervised learning.	It is possible to learn larger and more complex models with unsupervised learning.
Model	We can test our model.	We can not test our model.
Called as	Supervised learning is also called classification.	Unsupervised learning is also called clustering.
Example	Example: Optical character recognition.	Example: Find a face in an image.
Supervision	supervised learning needs supervision to train the model.	Unsupervised learning does not need any supervision to train the model.

Hierarchical clustering

Hierarchical clustering is a connectivity-based clustering model that groups the data points together that are close to each other based on the measure of similarity or distance. The assumption is that data points that are close to each other are more similar or related than data points that are farther apart.

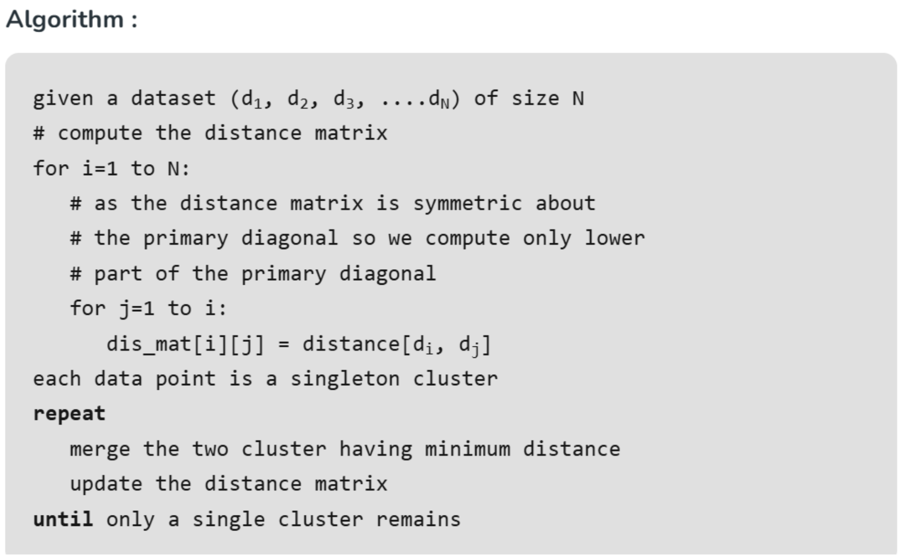
A dendrogram, a tree-like figure produced by hierarchical clustering, depicts the hierarchical relationships between groups. Individual data points are located at the bottom of the dendrogram, while the largest clusters, which include all the data points, are located at the top. In order to generate different numbers of clusters, the dendrogram can be sliced at various heights.

The dendrogram is created by iteratively merging or splitting clusters based on a measure of similarity or distance between data points. Clusters are divided or merged repeatedly until all data points are contained within a single cluster, or until the predetermined number of clusters is attained.

We can look at the dendrogram and measure the height at which the branches of the dendrogram form distinct clusters to calculate the ideal number of clusters. The dendrogram can be sliced at this height to determine the number of clusters.

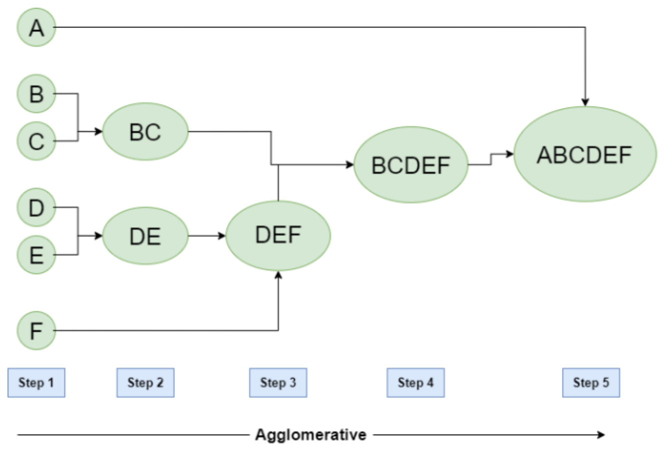
Hierarchical Agglomerative Clustering

It is also known as the bottom-up approach or hierarchical agglomerative clustering (HAC). A structure that is more informative than the unstructured set of clusters returned by flat clustering. This clustering algorithm does not require us to prespecify the number of clusters. Bottom-up algorithms treat each data as a singleton cluster at the outset and then successively agglomerate pairs of clusters until all clusters have been merged into a single cluster that contains all data.



Algorithm :

```
given a dataset ($d_1$, $d_2$, $d_3$, ....$d_N$) of size N
# compute the distance matrix
for i=1 to N:
   # as the distance matrix is symmetric about
   # the primary diagonal so we compute only lower
   # part of the primary diagonal
   for j=1 to i:
      dis_mat[i][j] = distance[$d_i$, $d_j$]
each data point is a singleton cluster
repeat
   merge the two cluster having minimum distance
   update the distance matrix
until only a single cluster remains
```

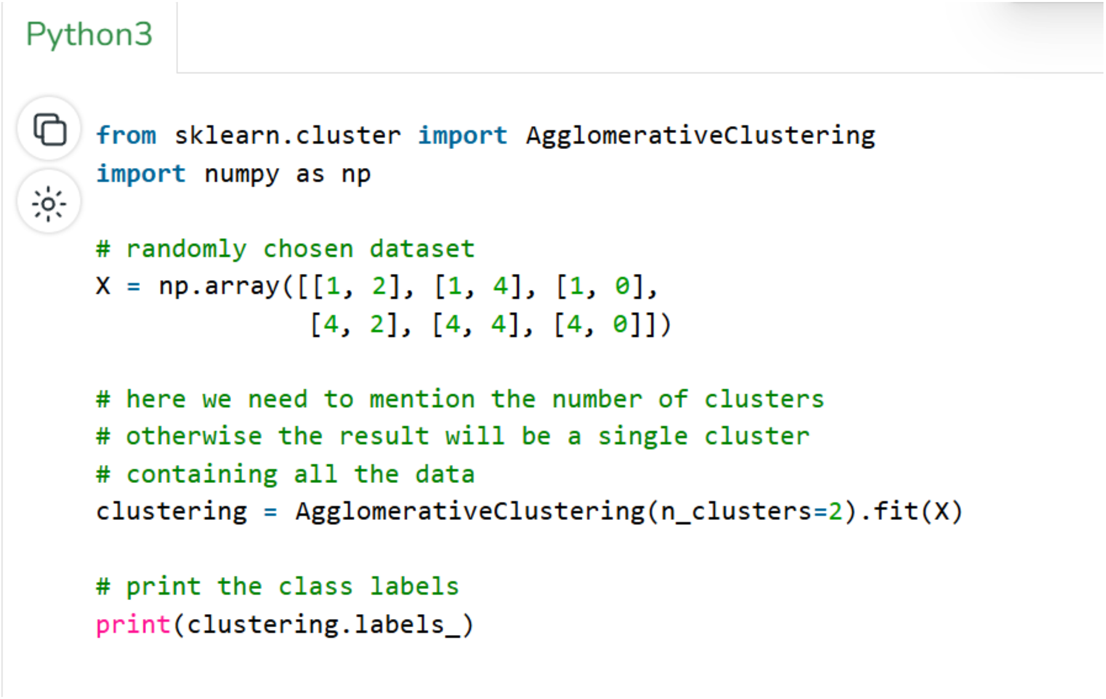


Hierarchical Agglomerative Clustering

Steps:

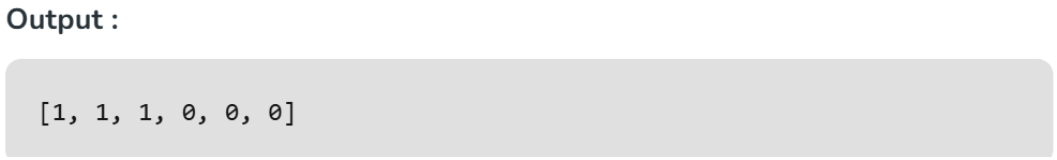
- Consider each alphabet as a single cluster and calculate the distance of one cluster from all the other clusters.
- In the second step, comparable clusters are merged together to form a single cluster. Let's say cluster (B) and cluster (C) are very similar to each other therefore we merge them in the 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)].

Python implementation of the above algorithm using the scikit-learn library:



Python3

```python
from sklearn.cluster import AgglomerativeClustering
import numpy as np

# randomly chosen dataset
X = np.array([[1, 2], [1, 4], [1, 0],
              [4, 2], [4, 4], [4, 0]])

# here we need to mention the number of clusters
# otherwise the result will be a single cluster
# containing all the data
clustering = AgglomerativeClustering(n_clusters=2).fit(X)

# print the class labels
print(clustering.labels_)
```

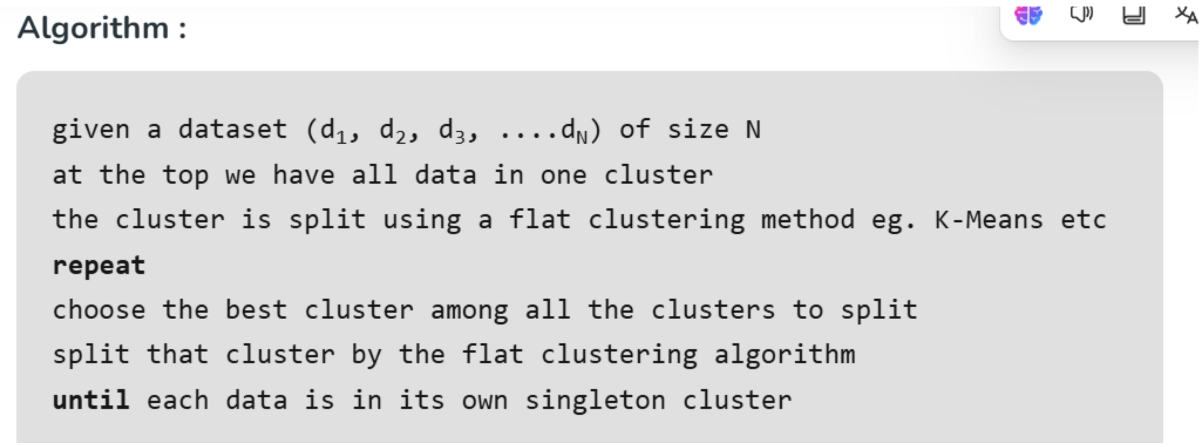


Output :

```
[1, 1, 1, 0, 0, 0]
```

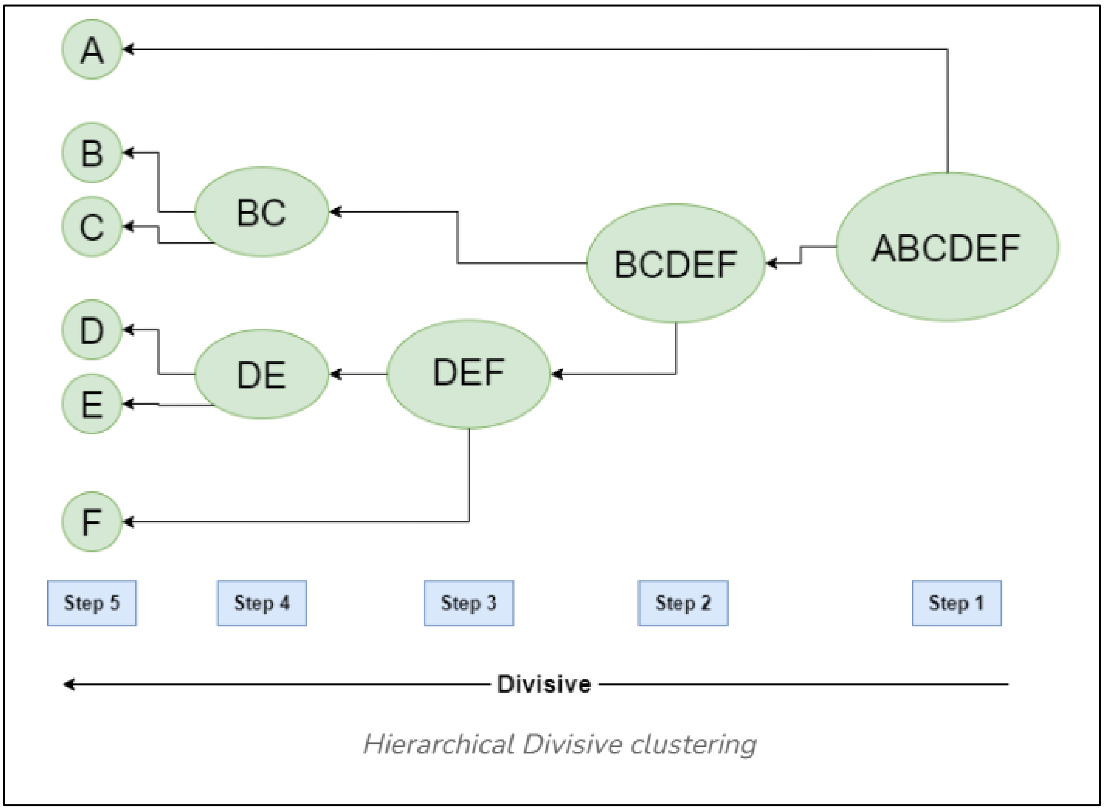
Hierarchical Divisive clustering

It is also known as a top-down approach. This algorithm also does not require to prespecify the number of clusters. Top-down clustering requires a method for splitting a cluster that contains the whole data and proceeds by splitting clusters recursively until individual data have been split into singleton clusters.



Algorithm :

```
given a dataset ($d_1$, $d_2$, $d_3$, ....$d_N$) of size N
at the top we have all data in one cluster
the cluster is split using a flat clustering method eg. K-Means etc
repeat
choose the best cluster among all the clusters to split
split that cluster by the flat clustering algorithm
until each data is in its own singleton cluster
```



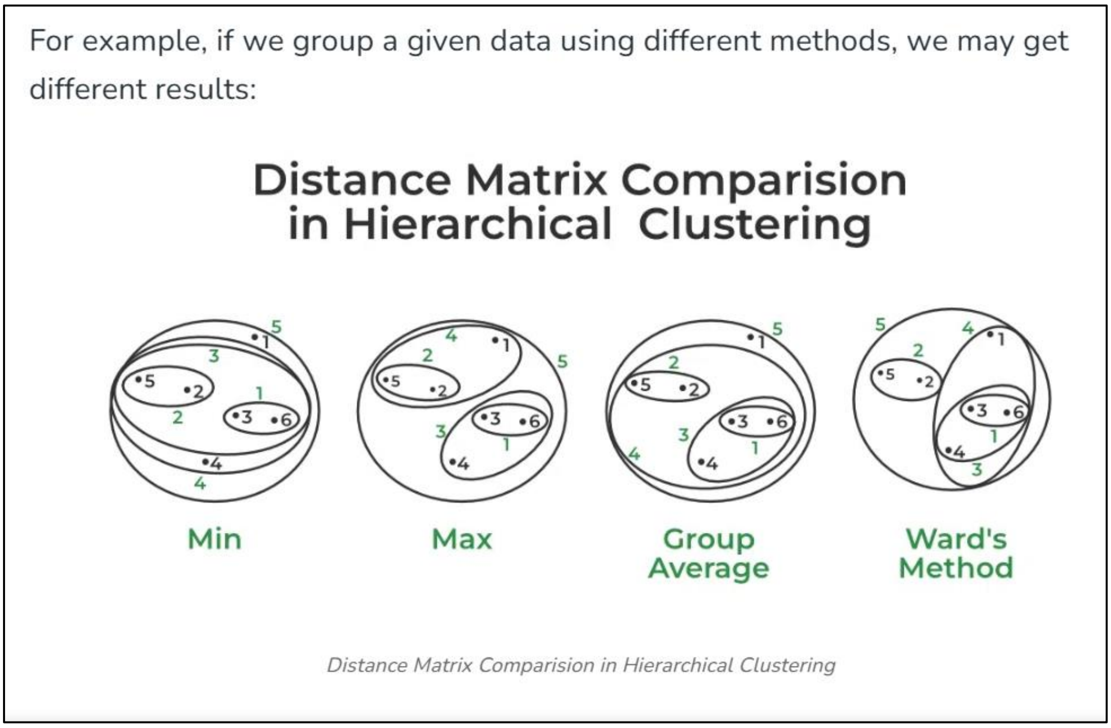
Hierarchical Divisive clustering

Computing Distance Matrix

While merging two clusters we check the distance between two every pair of clusters and merge the pair with the least distance/most similarity. But the question is how is that distance determined. There are different ways of defining Inter Cluster distance/similarity. Some of them are:

1. Min Distance: Find the minimum distance between any two points of the cluster.
2. Max Distance: Find the maximum distance between any two points of the cluster.
3. Group Average: Find the average distance between every two points of the clusters.
4. Ward's Method: The similarity of two clusters is based on the increase in squared error when two clusters are merged.

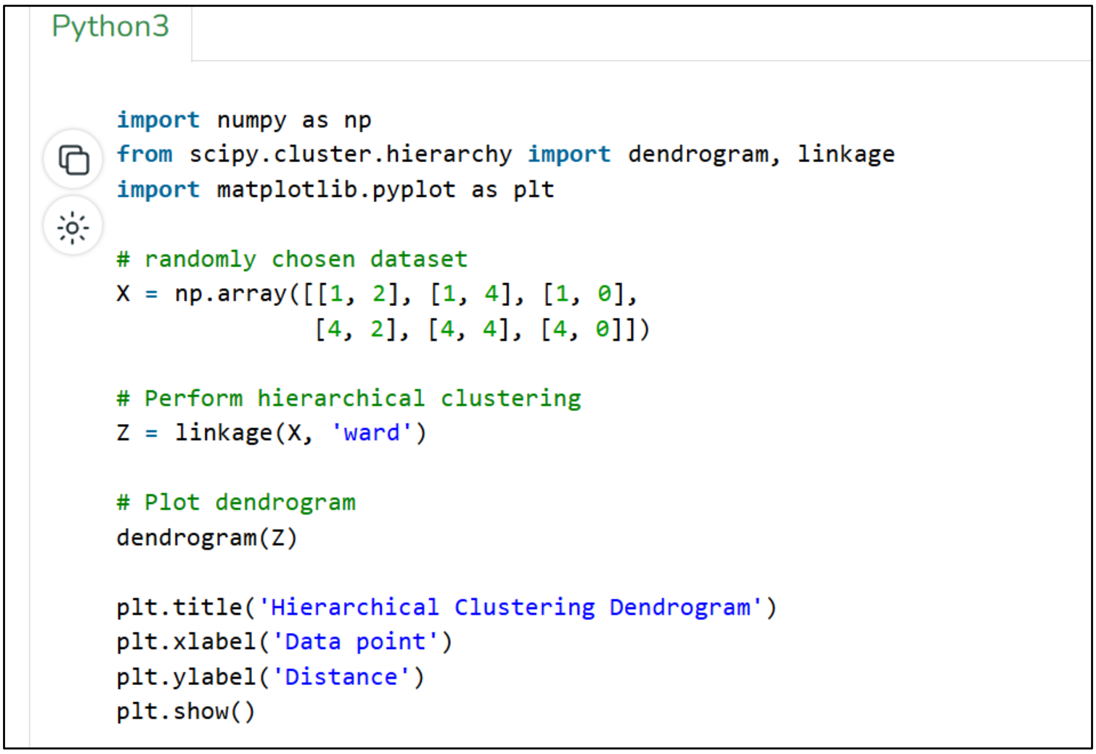
For example, if we group a given data using different methods, we may get different results:



For example, if we group a given data using different methods, we may get different results:

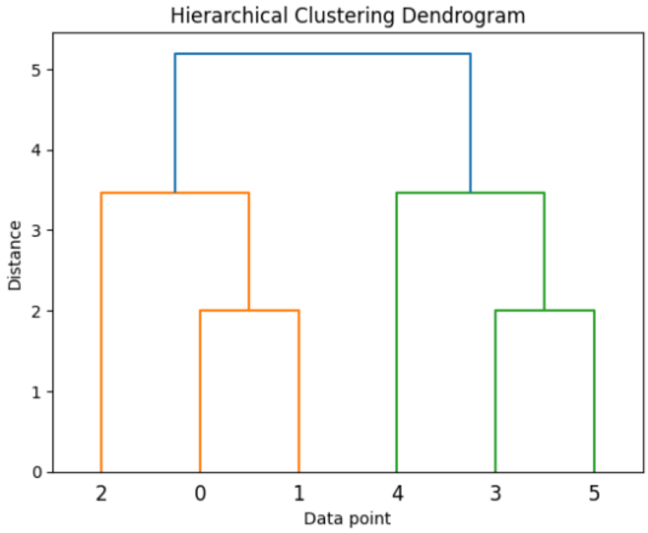
Distance Matrix Comparision in Hierarchical Clustering

Implementation Code



```python
import numpy as np
from scipy.cluster.hierarchy import dendrogram, linkage
import matplotlib.pyplot as plt

# randomly chosen dataset
X = np.array([[1, 2], [1, 4], [1, 0],
              [4, 2], [4, 4], [4, 0]])

# Perform hierarchical clustering
Z = linkage(X, 'ward')

# Plot dendrogram
dendrogram(Z)

plt.title('Hierarchical Clustering Dendrogram')
plt.xlabel('Data point')
plt.ylabel('Distance')
plt.show()
```

Output:



Hierarchical Clustering Dendrogram

Hierarchical Agglomerative *vs* Divisive Clustering

- Divisive clustering is more *complex* as compared to agglomerative clustering, as in the case of divisive clustering we need a flat clustering method as "subroutine" to split each cluster until we have each data having its own singleton cluster.

- Divisive clustering is more *efficient* if we do not generate a complete hierarchy all the way down to individual data leaves. The time complexity of a naive agglomerative clustering is **$O(n^3)$** because we exhaustively scan the N x N matrix dist_mat for the lowest distance in each of N-1 iterations. Using priority queue data structure we can reduce this complexity to **$O(n^2logn)$**. By using some more optimizations it can be brought down to **$O(n^2)$**. Whereas for divisive clustering given a fixed number of top levels, using an efficient flat algorithm like K-Means, divisive algorithms are linear in the number of patterns and clusters.

- A divisive algorithm is also more *accurate*. Agglomerative clustering makes decisions by considering the local patterns or neighbor points without initially taking into account the global distribution of data. These early decisions cannot be undone. whereas divisive clustering takes into consideration the global distribution of data when making top-level partitioning decisions.

Business Analytics can be applied across various domains:

1. Retail Analytics

- **Applications:** Sales prediction, inventory optimization, customer segmentation.
- **Example:** Amazon recommending products based on past purchases.

2. Marketing Analytics

- **Applications:** Customer churn prediction, campaign effectiveness, sentiment analysis.
- **Example:** Myntra using social media trends to optimize product launch.

3. Financial Analytics

- **Applications:** Credit risk modeling, fraud detection, portfolio management.
- **Example:** Banks using machine learning to detect fraudulent transactions in real-time.

4. Healthcare Analytics

- **Applications:** Patient outcome prediction, operational efficiency, resource allocation.
- **Example:** Hospitals predicting patient admission rates to optimize staffing.

5. Supply Chain Analytics

- **Applications:** Demand forecasting, route optimization, inventory management.
- **Example:** Domino’s using predictive analytics for delivery route optimization in India.