UNIT – 1 Introduction to Data Science

1. What is Data Science

Data science is a field that combines statistical analysis, machine learning, data visualization, and domain expertise to understand and interpret complex data. It involves various processes, including data collection, cleaning, analysis, modelling, and communication of insights. The goal is to extract valuable information from data to inform decision-making and strategy.

Key components of data science include:

1. **Data Collection**: Gathering data from various sources.
2. **Data Cleaning**: Preparing the data for analysis by removing errors and inconsistencies.
3. **Exploratory Data Analysis (EDA)**: Using statistical methods and visualization to explore the data.
4. **Machine Learning**: Applying algorithms to create models that can predict outcomes or classify data.
5. **Data Visualization**: Creating visual representations of data to make insights clear and understandable.
6. **Domain Knowledge**: Understanding the specific area in which the analysis is applied to provide context.

Multiple-Choice Questions

1. What is the primary goal of data science?
   - A) To collect data
   - B) To analyze data
   - C) To extract insights from data
   - D) To visualize data
2. Which of the following is NOT a step in the data science process?
   - A) Data Cleaning
   - B) Model Deployment
   - C) Data Ignoring
   - D) Data Visualization
3. Which of the following techniques is commonly used in machine learning?
   - A) Linear Regression
   - B) Statistical Testing
   - C) Descriptive Statistics
   - D) Data Collection
4. What does EDA stand for?
   - A) Exploratory Data Analysis
   - B) Enhanced Data Assessment
   - C) Effective Data Application
   - D) Evaluative Data Analysis
5. Which programming language is most commonly associated with data science?
   - A) Java
   - B) C++
   - C) Python
   - D) HTML
6. What is the purpose of data visualization?
   - A) To collect data efficiently
   - B) To present data in a clear and understandable way
   - C) To clean the data automatically
   - D) To store data securely
7. Which of the following is a type of supervised learning?
   - A) Clustering
   - B) Classification
   - C) Association
   - D) Dimensionality Reduction

2. Need for Data Science

Data science has become essential in today's data-driven world for several reasons:

1. **Data-Driven Decision Making** : Organizations use data science to make informed decisions based on empirical evidence rather than intuition or guesswork.

2. **Understanding Customer Behaviour**: Data science helps businesses analyse customer data to understand preferences and behaviours, leading to personalized marketing and improved customer experiences.

3. **Predictive Analytics** : Companies leverage data science to forecast trends and outcomes, allowing them to be proactive rather than reactive in their strategies.

4. **Operational Efficiency** : Data science can identify inefficiencies in processes and suggest improvements, leading to cost savings and better resource allocation.

5. **Competitive Advantage** : Organizations that utilize data science effectively can gain insights that provide a competitive edge in their industry.

6. **Innovation** : Data science enables organizations to explore new business models, products, and services based on data insights.

- **Multiple-Choice Questions (MCQs) on the Need for Data Science**

1. What is a primary benefit of data-driven decision making?
   - A) Reducing the number of employees
   - B) Making decisions based on empirical evidence
   - C) Increasing the complexity of decisions
   - D) Relying on historical trends alone
2. How does data science help in understanding customer behaviour?
   - A) By ignoring data
   - B) By analysing customer data for insights
   - C) By increasing product prices
   - D) By limiting customer interactions
3. What is predictive analytics primarily used for?
   - A) To analyse past data only
   - B) To forecast future trends and outcomes
   - C) To gather data from social media
   - D) To clean data sets
4. Which of the following is a way data science can improve operational efficiency?
   - A) By complicating processes
   - B) By identifying inefficiencies and suggesting improvements
   - C) By eliminating all data analysis
   - D) By increasing the number of reports generated
5. What does gaining a competitive advantage through data science involve?
   - A) Implementing outdated practices
   - B) Using insights that competitors do not have
   - C) Reducing the data science team
   - D) Ignoring customer feedback

6. What role does data science play in innovation?
   - A) It restricts creativity by focusing on data
   - B) It enables exploration of new business models based on data insights
   - C) It only supports existing business models
   - D) It eliminates the need for new products
7. Which of the following industries has benefited from data science?
   - A) Healthcare
   - B) Finance
   - C) Retail
   - D) All of the above

3. Business Intelligence (BI) VS Data Science

1. **Definition**: Focuses on analyzing historical data to provide actionable insights and improve decision-making.

2. **Key Goal**: Reporting, visualization, and descriptive analytics.

3. **Techniques**: Dashboards, SQL queries, data visualization tools (e.g., Power BI, Tableau).

4. **Usage**: Monitoring KPIs, trend analysis, performance management.

5. **Output**: Predefined reports and dashboards for business users.

Data Science (DS)

1. **Definition**: Extracts insights and predictions from large, complex datasets using advanced algorithms.
2. **Key Goal**: Predictive and prescriptive analytics.
3. **Techniques**: Machine learning, AI, statistical modelling, coding (e.g., Python, R).
4. **Usage**: Forecasting trends, customer segmentation, anomaly detection.
5. **Output**: Predictive models, recommendations, insights for innovation.

MCQs

1. What is the primary goal of Business Intelligence?
   a) Predicting future trends
   b) Monitoring and reporting historical data
   c) Developing machine learning algorithms
   d) Building predictive models
2. Which tool is commonly used in Data Science?
   a) Tableau
   b) Power BI
   c) Python
   d) SAP
3. Business Intelligence focuses primarily on:
   a) Unstructured data analysis
   b) Prescriptive analytics
   c) Data visualization and dashboards
   d) Deep learning algorithms
4. What is the key difference between BI and Data Science?
   a) BI is for the future, and Data Science focuses on the past
   b) BI uses AI, while Data Science uses SQL
   c) BI provides descriptive analytics, while Data Science offers predictive insights
   d) Both are identical in purpose

5. Which of the following is a predictive analytics technique?
   a) KPI dashboards
   b) Time-series forecasting
   c) SQL querying
   d) Data visualization

4. Components of Data Science

Data Science is a multidisciplinary field that combines various components to extract meaningful insights from data. The key components include:

1. **Data Collection**
Definition: The process of gathering raw data from various sources such as databases, APIs, web scraping, or manual entry.
Tools: SQL, Python (libraries like requests, Beautiful Soup), APIs.

2. Data Preparation (Data Cleaning and Pre-processing)
Definition: Cleaning and transforming raw data into a structured and usable format.
Techniques: Handling missing values, normalization, outlier detection.
Tools: Python (Pandas, NumPy), Excel, ETL tools.

3. Data Exploration (Exploratory Data Analysis - EDA)
Definition: Analyzing data patterns, trends, and anomalies to understand its structure and potential.
Techniques: Summary statistics, visualizations.
Tools: Matplotlib, Seaborn, Tableau, Power BI.
4. Data Modelling
Definition: Building models using statistical and machine learning algorithms to make predictions or classify data.
Techniques: Regression, classification, clustering, time-series analysis.
Tools: Python (Scikit-learn, Tens or Flow), R, SAS.

5. Model Evaluation
Definition: Assessing the performance of a model using metrics to ensure accuracy and reliability.
Metrics: Accuracy, precision, recall, F1 score, RMSE.
Tools: Python (Scikit-learn), R, cross-validation techniques.

6. Data Visualization
Definition: Representing data insights through charts, graphs, and dashboards.
Purpose: Communicating results effectively to stakeholders.
Tools: Matplotlib, Seaborn, Power BI, Tableau.

7. Deployment and Communication
Definition: Integrating the model into production and effectively communicating results to stakeholders.
Tools: Flask, Docker, APIs, PowerPoint for presentations.

MCQs

1. What is the first step in the Data Science process?
   a) Data Visualization
   b) Data Collection
   c) Data Modelling
   d) Model Evaluation

2. Which of the following is NOT a part of Data Preparation?
   a) Handling missing values
   b) Normalization
   c) Model deployment
   d) Outlier detection
3. What is the primary goal of Exploratory Data Analysis (EDA)?
   a) Build predictive models
   b) Deploy machine learning models
   c) Identify patterns and trends in data
   d) Handle missing data
4. Which of these tools is widely used for Data Modelling in Data Science?
   a) Power BI
   b) Scikit-learn
   c) Tableau
   d) SQL
5. Which metric is used to evaluate the accuracy of a classification model?
   a) RMSE
   b) Precision
   c) Silhouette Score
   d) Variance
6. What is the main objective of Data Visualization?
   a) Clean the data
   b) Communicate insights effectively
   c) Train machine learning models
   d) Collect raw data

5. Data Science Life Cycle

1. Problem Definition

Understanding the business problem and defining the goals.

2. Data Collection

Gathering relevant data from various sources.

3. Data Preparation (Data Wrangling)

Cleaning, transforming, and organizing the data.

4. Exploratory Data Analysis (EDA)

Analyzing data patterns, trends, and insights using visualization and statistics.

5. Model Building

Selecting algorithms and training machine learning models.

6. Model Evaluation

Testing the model's performance using metrics (e.g., accuracy, precision).

7. Deployment

Integrating the model into a production environment.

8. Monitoring and Maintenance

Continuously monitoring and improving the model.

MCQs

1. Which is the first step in the Data Science Life Cycle?
   A. Data Collection
   B. Problem Definition
   C. Model Building
   D. Data Visualization
2. What is the main objective of Data Preparation?
   A. Building machine learning models
   B. Cleaning and organizing data for analysis
   C. Testing the model's accuracy
   D. Deploying the model
3. Which process helps identify patterns and trends in data?
   A. Model Deployment
   B. Data Collection
   C. Exploratory Data Analysis (EDA)
   D. Data Cleaning
4. What step involves splitting data into training and testing datasets?
   A. Model Deployment
   B. Model Evaluation
   C. Model Building
   D. Data Preparation
5. Which of the following is NOT a model evaluation metric?
   A. Accuracy
   B. Precision
   C. Data Cleaning
   D. Recall
6. After deploying a model, what is the next step?
   A. Model Building
   B. Data Wrangling
   C. Monitoring and Maintenance
   D. EDA
7. What is the purpose of data visualization during EDA?
   A. Build predictive models
   B. Clean raw data
   C. Identify patterns and insights
   D. Test model performance

6. Tools for Data Science

Data scientists use various tools for data collection, cleaning, analysis, visualization, and machine learning. Here are the key categories and examples:

1. Data Collection and Storage Tools

Databases: MySQL, PostgreSQL, MongoDB, Cassandra, Oracle.
Big Data Platforms: Hadoop, Apache Spark, AWS, Azure.

2. Data Preparation Tools

Programming Languages: Python (Pandas, NumPy), R.
Data Integration: Apache NiFi, Talend, Alteryx.

3. Data Analysis Tools

Statistical Tools: R, MATLAB, SAS.
Machine Learning Frameworks: Scikit-learn, TensorFlow, PyTorch.

4. Data Visualization Tools

Tableau, Power BI, Matplotlib, Seaborn, Plotly.

5. Collaboration and Version Control

Git, GitHub, Jupyter Notebooks, Google Colab.

6. Deployment and Monitoring Tools

Deployment: Docker, Kubernetes.
Monitoring: MLflow, Prometheus.

MCQs

1. Which of the following is NOT a data storage tool?
   A. MySQL
   B. PostgreSQL
   C. Hadoop
   D. Matplotlib
2. Which language is widely used for data manipulation and analysis?
   A. Java B. Python C. C++ D. PHP
3. Tableau is primarily used for:
   A. Data Storage B. Data Visualization
   C. Machine Learning D. Statistical Analysis
4. Which framework is used for building machine learning models?
   A. Tens or Flow B. Power BI
   C. Tableau D. MongoDB
5. Which of the following is a version control tool?
   A. Jupyter Notebook B. Git C. Pandas D. NumPy
6. For processing large-scale data in a distributed environment, which tool is best suited?
   A. Apache Spark B. SAS C. Seaborn D. MATLAB
7. What tool is commonly used for creating dashboards and business intelligence reports?
   A. Power BI B. Tens or Flow
   C. GitHub D. Scikit-learn
8. Which Python library is used for data visualization?
   A. NumPy B. Pandas C. Matplotlib D. Tens or Flow
9. Docker is used for:
   A. Data Cleaning
   B. Model Deployment
   C. Statistical Analysis
   D. Data Visualization
10. Jupyter Notebook is mainly used for:
   A. Writing code and documenting analysis
   B. Storing large datasets
   C. Version control
   D. Creating production-ready applications

UNIT – 2 Introduction of Big Data

➢ Introduction to Big Data

Big Data refers to extremely large datasets that cannot be effectively managed, processed, or analyzed using traditional data processing techniques due to their volume, velocity, and variety. It represents a transformative approach to managing and analyzing data that helps organizations make data-driven decisions, uncover patterns, and gain valuable insights.

Characteristics of Big Data (The 5 V's)

1. Volume

The sheer size of data generated daily, often measured in terabytes or petabytes. Examples include social media posts, transaction records, and sensor data.

2. Velocity

The speed at which data is generated and processed. For instance, real-time data streams from IoT devices or financial transactions.

3. Variety

The diversity of data types, such as structured (databases), semi-structured (XML, JSON), and unstructured (videos, images, text).

4. Veracity

The quality and accuracy of data, which can often be incomplete, noisy, or misleading.

5. Value

The actionable insights and benefits derived from analyzing Big Data.

➢ Sources of Big Data

1. Social Media: Platforms like Facebook, Twitter, and Instagram generate vast amounts of user-generated content.

2. E-commerce: Transaction records, customer preferences, and behavior.

3. Healthcare: Patient records, medical imaging, and genomic data.

4. IoT (Internet of Things): Data from connected devices like sensors, cameras, and wearables.

5. Government & Public Services: Census data, transportation data, and utility records.

➢ Technologies Used in Big Data

1. Storage

Hadoop Distributed File System (HDFS)

Cloud storage (e.g., AWS S3, Google Cloud Storage)

2. Processing

Apache Hadoop

Apache Spark

3. Database Systems

NoSQL databases like MongoDB, Cassandra, and HBase.

Distributed SQL systems like Google BigQuery.

4. Analytics

Tools like Tableau, Power BI, and Apache Kafka.

Machine learning frameworks like TensorFlow and Scikit-learn.

- **Applications of Big Data**

1. Business Intelligence: Optimizing customer experience and marketing strategies.

2. Healthcare: Personalized medicine and predictive analytics.

3. Finance: Fraud detection and risk management.

4. Smart Cities: Traffic management, energy optimization, and public safety.

5. Entertainment: Recommendation engines (e.g., Netflix, Spotify).

- **Challenges in Big Data**

1. Data Security and Privacy: Protecting sensitive data from breaches.

2. Storage and Management: Handling the volume and diversity of data.

3. Scalability: Ensuring systems grow with the data.

4. Skilled Workforce: Availability of data scientists and analysts.

- **Big Data can be classified into different types based on its nature, source, and structure:**

1. Based on Data Type

Structured Data: Organized in rows and columns, stored in relational databases (e.g., SQL).

Example: Customer records, transaction data.

Unstructured Data: Lacks a predefined format, difficult to process (e.g., videos, images, social media posts).

Example: Emails, audio files.

Semi-structured Data: Does not follow strict schema but has some organizational properties.

Example: XML, JSON files.

2. Based on Source

Human-generated Data: Data created by human activities.

Example: Social media updates, online purchases.

Machine-generated Data: Automatically created by machines or devices.

Example: IoT sensor data, server logs.

3. Based on Processing Requirements

Batch Data: Processed in chunks over time (e.g., transaction data at the end of the day).

Stream Data: Real-time data processed as it is generated (e.g., stock market feeds).

4. Based on Domain

Business Data: Customer, sales, and marketing data.

Scientific Data: Research data from experiments, space exploration.

Government Data: Census data, crime statistics.

Multiple Choice Questions (MCQs)

1. What type of Big Data is an email?
   A) Structured
   B) Unstructured
   C) Semi-structured
   D) Machine-generated
2. Which of the following is an example of structured data?
   A) Social media posts
   B) Relational database records
   C) Audio files
   D) IoT sensor data
3. What type of Big Data is generated by IoT devices?
   A) Human-generated
   B) Semi-structured
   C) Machine-generated
   D) Batch data
4. Which classification applies to Big Data processed in real-time?
   A) Batch Data
   B) Stream Data
   C) Structured Data
   D) Scientific Data
5. XML and JSON are examples of which type of Big Data?
   A) Structured
   B) Unstructured
   C) Semi-structured
   D) Machine-generated

- **Definition of Big Data**

  Big Data refers to extremely large datasets that are complex, diverse, and grow rapidly, making them difficult to process using traditional data management tools. These datasets exhibit characteristics such as high volume, velocity, variety, veracity, and value.

- **Evolution of Big Data**

  1. Early Days (1960s–1980s)

  Data was primarily structured and stored in relational databases.

  Limited storage capacity and computing power restricted data processing.

  Example: Mainframe computers processing transaction records.

  2. Growth of the Internet (1990s)

  Rapid increase in data generation due to the internet boom.

  Emergence of semi-structured data like emails and web logs.

  Tools like SQL were used to manage and query relational databases.

  3. Rise of Social Media & IoT (2000s)

  Social media platforms and IoT devices started producing massive volumes of unstructured data.

  Limitations of traditional databases led to the development of new frameworks like Hadoop.

  Big Data was formally recognized as a field of study.

  4. Modern Era (2010s–Present)

  Advancements in cloud computing, distributed systems, and machine learning enhanced Big Data analytics.

  Real-time data processing technologies like Apache Spark became popular.

  Big Data became critical in industries such as finance, healthcare, and retail.

  Key Milestones in Big Data Evolution

  1. 2004: Google publishes its paper on the MapReduce programming model.

  2. 2006: Apache Hadoop, an open-source Big Data framework, is released.

  3. 2010s: NoSQL databases and real-time processing tools like Kafka and Spark emerge.

  4. 2020s: Integration of Big Data with AI, IoT, and edge computing.

- **Multiple Choice Questions (MCQs)**

  1. What is Big Data primarily defined by?

     A) High volume, low speed, and structured data

     B) High volume, velocity, and variety

     C) Low variety, high veracity, and value

     D) Limited size and fixed structure

2. When did Big Data gain prominence as a field of study?

A) 1960s

B) 1980s

C) 2000s

D) 2020s

3. Which of the following technologies played a key role in the evolution of Big Data?

A) Apache Hadoop

B) Relational databases

C) MapReduce

D) Both A and C

4. What triggered the need for new Big Data tools in the 2000s?

A) Increase in social media and IoT-generated data

B) Limited storage space

C) Decline of relational databases

D) Introduction of SQL

5. Which of the following best describes the modern era of Big Data?

A) Use of traditional databases for storage

B) Integration of Big Data with AI, IoT, and cloud computing

C) Exclusively focusing on structured data

D) Dependence on mainframe computers

- **Big Data Architecture**

Big Data architecture refers to the design and framework for collecting, processing, storing, and analyzing massive volumes of data efficiently. It involves various components and technologies that work together to manage the 5 V's of Big Data (Volume, Velocity, Variety, Veracity, and Value).

- **Components of Big Data Architecture**

1. Data Sources

The origin of the data, which can include:Structured data (databases, spreadsheets).

Unstructured data (social media, videos).

Machine-generated data (IoT sensors, logs).

2. Data Ingestion Layer

Responsible for capturing and importing data into the system.

Techniques:Batch Processing: Data is processed in chunks (e.g., Hadoop).

Stream Processing: Real-time data processing (e.g., Apache Kafka).

3. Data Storage Layer

Where the data is stored for further processing and analysis.

Distributed File Systems: Hadoop Distributed File System (HDFS).

Databases: NoSQL databases (MongoDB, Cassandra).

Cloud Storage: AWS S3, Azure Blob Storage.

4. Data Processing Layer

Processes and transforms data into a usable format.

Batch Processing: MapReduce, Apache Hive.

Real-time Processing: Apache Spark, Apache Storm.

5. Data Analysis and Query Layer

Tools for querying, analyzing, and visualizing the data.

Analytics Tools: Apache Hive, Apache Pig, SQL engines.

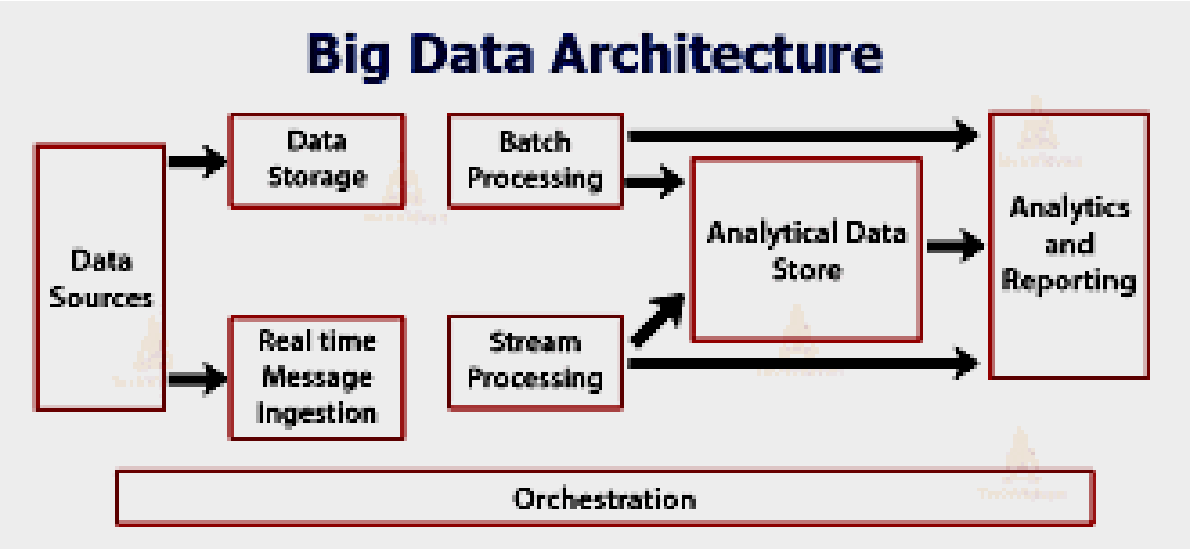
Visualization Tools: Tableau, Power BI.

6. Data Consumption Layer

Provides processed data to users or applications.

Dashboards, Reports, or APIs for business intelligence and decision-making.

- **Big Data Architecture Diagram**



Key Features of Big Data Architecture

1. Scalability: Handle growing volumes of data.

2. Fault Tolerance: Ensure system reliability.

3. Real-time Processing: Analyze data as it is generated.

4. Interoperability: Integration with various tools and technologies.

- **Multiple Choice Questions (MCQs)**

1. What is the main function of the Data Ingestion Layer in Big Data architecture?

   A) Storing data in databases

   B) Importing and capturing data from sources

   C) Visualizing processed data

   D) Querying the stored data

2. Which of the following technologies is used for real-time data processing in Big Data architecture?

   A) MapReduce

   B) Apache Spark

   C) HDFS

   D) MongoDB

3. What type of storage is HDFS primarily designed for?

   A) Centralized storage

   B) Distributed storage

   C) Local storage

   D) Cloud-based storage

4. Which layer in Big Data architecture is responsible for analyzing and visualizing the data?

   A) Data Ingestion Layer

   B) Data Processing Layer

   C) Data Analysis and Query Layer

   D) Data Storage Layer

5. What is the main purpose of NoSQL databases in Big Data architecture?

   A) Managing structured data only

   B) Handling unstructured and semi-structured data

   C) Real-time data visualization

   D) Reducing storage cost

UNIT – 3 Introduction to HADOOP

➢ Apache Hadoop

Apache Hadoop is an open-source framework developed by the Apache Software Foundation for distributed storage and processing of large-scale datasets across computer clusters. It is specifically designed for handling big data, providing scalability, reliability, and efficiency.

➢ Hadoop Ecosystem Components

The Hadoop ecosystem includes a wide range of tools to handle various aspects of big data processing:

Hive: A data warehousing tool that provides SQL-like querying capabilities.

HBase: A distributed NoSQL database for real-time data access.

Pig: A high-level scripting language for data transformation tasks.

Spark: An in-memory processing engine for faster data analytics.

Sqoop: For importing and exporting data between Hadoop and relational databases.

Flume: Designed to collect, aggregate, and move large amounts of log data.

Zookeeper: Provides distributed synchronization and configuration management.

➢ Advantages of Apache Hadoop

1. Scalability: Can scale horizontally to handle petabytes of data by adding nodes.

2. Fault Tolerance: Automatically replicates data across nodes to prevent data loss.

3. Cost-Effective: Uses commodity hardware, reducing infrastructure costs.

4. Flexible Data Handling: Processes structured, semi-structured, and unstructured data.

5. Open-Source: Freely available, with a large and active developer community.

➢ Use Cases of Apache Hadoop

Data Analytics: Social media, web traffic, and business intelligence.

Fraud Detection: Financial institutions use Hadoop for fraud prevention.

Search Engines: Helps index and retrieve large volumes of data.

Log Analysis: Real-time monitoring and anomaly detection in IT systems.

Machine Learning: For training models on massive datasets.

Apache Hadoop has become a foundational technology for big data analytics and is widely adopted in industries like finance, healthcare, retail, and technology. It continues to evolve, with additional tools and frameworks expanding its capabilities.

MCQs on Apache Hadoop

1. What is the main purpose of Hadoop?

a) To manage relational databases
b) To process and store large datasets in a distributed manner
c) To develop web applications
d) To optimize network bandwidth

2. What is HDFS in Hadoop?

a) High Distributed File System

b) Hadoop Data File System

c) Hadoop Distributed File System

d) Hadoop Disk File System

3. Which of the following is NOT a component of Hadoop?

a) HDFS b) MapReduce c) SQL Server d) YARN

4. What does YARN stand for?

a) Yet Another Resource Negotiator

b) Your Advanced Resource Network

c) Your Application Resource Network

d) Yet Another Random Network

5. What is the role of MapReduce in Hadoop?

a) To store data

b) To manage resources

c) To process data in parallel

d) To schedule tasks

6. How does Hadoop handle hardware failures?

a) By restarting the system

b) By replicating data across multiple nodes

c) By shutting down the cluster

d) By using expensive hardware

7. Which programming language is primarily used for Hadoop?

a) Python b) Java c) C++ d) PHP

8. What is the default replication factor in HDFS?

a) 1 b) 2 c) 3 d) 4

9. Which tool in the Hadoop ecosystem is used for SQL-like queries?

a) Pig b) Hive c) Flume d) Sqoop

10. What is the role of Zookeeper in the Hadoop ecosystem?

a) Storing unstructured data b) Managing and coordinating distributed applications

c) Importing data into HDFS

d) Performing ETL operations

- **Hadoop Architecture**

The Hadoop architecture is designed to process and store massive datasets efficiently in a distributed and fault-tolerant manner. It is based on the Master-Slave architecture and includes the following major components**:**

1. Hadoop Distributed File System (HDFS)

Master Component: NameNode

Manages the file system metadata and controls access to files.

Keeps track of where data blocks are stored across the cluster.

Slave Component: DataNode

Stores actual data in blocks.

Reports block status and health to the NameNode.

2. YARN (Yet Another Resource Negotiator)

ResourceManager: Allocates cluster resources and assigns tasks to nodes.

NodeManager: Manages individual nodes in the cluster and monitors resource usage.

3. MapReduce Framework

Programming model for processing large datasets in parallel.

Map Phase: Processes input data and produces intermediate key-value pairs.

Reduce Phase: Aggregates the output from the Map phase into meaningful results.

- **Hadoop Ecosystem**

The Hadoop ecosystem consists of various tools that extend Hadoop's functionality, enabling it to manage, process, and analyze diverse data types effectively.

Key Components:

1. Hive: SQL-like querying for structured data.
2. Pig: High-level scripting for data transformation.
3. HBase: A NoSQL database for real-time data access.
4. Sqoop: Import/export data between Hadoop and relational databases.
5. Flume: Collects and moves log data into HDFS.
6. Spark: An in-memory data processing engine for real-time analytics.
7. Zookeeper: Coordination and synchronization for distributed systems.
8. Mahout: Machine learning and recommendation system libraries.

MCQs

1. What is the primary function of HDFS in Hadoop?
   a) To process data
   b) To store data in a distributed manner
   c) To query data
   d) To manage resources
2. What is the responsibility of the NameNode in HDFS?
   a) Store data blocks
   b) Manage metadata and block locations
   c) Schedule jobs in the cluster
   d) Monitor resource usage
3. Which component is responsible for resource allocation in YARN?
   a) DataNode
   b) ResourceManager
   c) NameNode
   d) NodeManager
4. What are the two phases of a MapReduce job?
   a) Store and Process
   b) Map and Reduce
   c) Fetch and Aggregate
   d) Query and Execute
5. Which tool in the Hadoop ecosystem is used for real-time data processing?
   a) Sqoop
   b) Spark
   c) Pig
   d) Hive
6. What is the default block size in HDFS for storing data?
   a) 32MB
   b) 64MB
   c) 128MB
   d) 256MB
7. Which Hadoop ecosystem component is used for machine learning?
   a) Flume b) Mahout c) Hive d) HBase

8. Which of the following tools is used for importing and exporting data in Hadoop?

a) Hive

b) Pig

c) Sqoop

d) Flume

9. In the Hadoop architecture, which node performs data storage tasks?

a) NameNode

b) DataNode

c) ResourceManager

d) NodeManager

10. What is the role of Zookeeper in the Hadoop ecosystem?

a) Data storage

b) Log management

c) Coordination and synchronization

d) Machine learning

- **Hadoop Ecosystem Components**

The Hadoop ecosystem includes a variety of tools and frameworks that complement Hadoop's core functionality (HDFS, MapReduce, and YARN) to handle diverse big data tasks like storage, processing, analysis, and real-time computation.

1. Core Components

- HDFS (Hadoop Distributed File System): Distributed storage for large datasets.
- MapReduce: Programming model for batch data processing.
- YARN (Yet Another Resource Negotiator): Resource management and task scheduling.

2. Ecosystem Tools

1. Hive

- ✓ Data warehousing and SQL-like querying for large datasets.
- ✓ Suitable for structured data.
- ✓ Converts SQL queries into MapReduce jobs.

2. Pig

- ✓ High-level scripting language for data transformation.
- ✓ Converts Pig scripts into MapReduce tasks.
- ✓ Suitable for semi-structured and unstructured data.

3. HBase

- ✓ A distributed NoSQL database for real-time read/write access.
- ✓ Built on HDFS, optimized for random access.

4. Spark

- ✓ An in-memory processing engine for fast analytics.
- ✓ Supports batch processing, machine learning, graph processing, and stream processing.

5. Sqoop

- ✓ Transfers data between Hadoop and relational databases like MySQL or Oracle.
- ✓ Ideal for ETL (Extract, Transform, Load) operations.

6. Flume

- ✓ Collects, aggregates, and moves large amounts of log data into HDFS.
- ✓ Best for streaming data.

7. Zookeeper

- ✓ Manages and coordinates distributed systems.
- ✓ Ensures synchronization across nodes.

8. Oozie

- ✓ Workflow scheduler for managing Hadoop jobs.
- ✓ Executes workflows involving Hive, Pig, and MapReduce tasks.

9. Mahout

- ✓ Provides machine learning algorithms for clustering, classification, and recommendations.

10. Kafka

- ✓ A distributed messaging system for real-time data streams.
- ✓ Often used with Spark or Flume.

MCQs

1. Which Hadoop tool provides SQL-like querying capabilities?

a) Pig b) Hive

c) Sqoop d) Oozie

2. What is the purpose of HBase in the Hadoop ecosystem?

a) To process data in real-time

b) To provide NoSQL database storage

c) To collect log data

d) To manage workflows

3. Which component is used for data import/export between Hadoop and relational databases?

a) Flume b) Sqoop c) Spark d) Mahout

4. What is the main use of Flume in the Hadoop ecosystem?

a) To manage workflows

b) To process structured data

c) To collect and move log data

d) To provide SQL queries

5. Which of the following is a workflow management tool in Hadoop?

a) Zookeeper

b) Oozie

c) Pig

d) Hive

6. What is the primary function of Spark in the Hadoop ecosystem?

a) Data storage

b) Batch processing

c) In-memory data processing

d) Real-time log collection

7. Which tool is used for machine learning tasks in Hadoop?

a) Oozie

b) Mahout

c) Hive

d) Sqoop

8. What does Zookeeper provide in the Hadoop ecosystem?

a) Data synchronization and coordination

b) SQL-like querying

c) Real-time data streaming

d) Workflow scheduling

9. Which component handles the streaming of real-time data in Hadoop?

a) Kafka

b) Pig

c) Hive

d) Spark

10. Which tool in the Hadoop ecosystem is ideal for large-scale graph processing?

a) Hive

b) Spark

c) HBase

d) Flume

- **MapReduce Overview**

MapReduce is a programming model and processing framework in Hadoop for processing large datasets in a distributed and parallel manner. It consists of two key phases:

1. Map Phase:

- ✓ Splits the input data into smaller chunks and processes them independently.
- ✓ Converts data into key-value pairs.

2. Reduce Phase:

- ✓ Aggregates, filters, or summarizes the intermediate key-value pairs produced by the Map phase.
- ✓ Produces the final output.

Key Features:

- ✓ Works with HDFS for fault tolerance and distributed processing.
- ✓ Suitable for batch processing of large-scale datasets.

- **Workflow of MapReduce**

1. Input data is split into chunks.

2. The Mapper processes each chunk to produce intermediate key-value pairs.

3. The Shuffle and Sort phase organizes these key-value pairs by key.

4. The Reducer processes grouped data to generate the final output.

MCQs

1. What is the main purpose of MapReduce?

a) To store large datasets

b) To process large datasets in a distributed manner

c) To query relational databases

d) To synchronize distributed systems

2. What does the Mapper in MapReduce do?

a) Aggregates intermediate results

b) Processes input data to produce key-value pairs

c) Sorts key-value pairs

d) Stores final output

3. What is the role of the Reducer in MapReduce?

a) To split input data

b) To store intermediate data

c) To aggregate and summarize key-value pairs

d) To manage resources

4. What phase in MapReduce organizes key-value pairs by key?

a) Map phase

b) Shuffle and Sort phase

c) Reduce phase

d) Input phase

5. Which of the following is NOT a part of the MapReduce framework?

a) Mapper

b) Reducer

c) NameNode

d) Combiner

6. What is the Combiner used for in MapReduce?

a) To split the input data b) To reduce the amount of data transferred to the Reducer

c) To replicate data across nodes d) To sort intermediate data

7. Which of the following describes the output of the Map phase?

a) Key-value pairs organized by key

b) Final aggregated results

c) Raw input data

d) Intermediate key-value pairs

8. In MapReduce, what is the default number of Reducers?

a) 1

b) 2

c) 4

d) Depends on the cluster size

9. What type of tasks can MapReduce efficiently handle?

a) Real-time data processing

b) Small-scale dataset processing

c) Batch processing of large-scale datasets

d) Graphical user interface tasks

10. Which Hadoop component is responsible for scheduling MapReduce tasks?

a) YARN b) HDFS

c) NameNode d) Hive

Key Concepts in MapReduce

- ✓ InputSplit: Logical representation of input data processed by Mapper.
- ✓ Partitioner: Decides how intermediate key-value pairs are distributed to Reducers.
- ✓ Hadoop Counters: Used to track metrics like processed records or errors.

- **HDFS (Hadoop Distributed File System)**

HDFS is the storage layer of the Hadoop ecosystem, designed for storing and managing large datasets in a distributed and fault-tolerant manner. It splits data into smaller blocks and distributes them across nodes in a cluster.

Key Components of HDFS

1. NameNode (Master Node)

- ✓ Manages metadata (e.g., file structure, locations of data blocks).
- ✓ Coordinates data storage but doesn't store actual data.

2. DataNode (Slave Node)

- ✓ Stores actual data in the form of blocks.
- ✓ Sends periodic heartbeats to the NameNode to indicate availability.

3. Secondary NameNode

- ✓ Acts as a checkpointing mechanism for the NameNode.
- ✓ Helps in merging and backing up metadata but does not replace the NameNode in case of failure.

4. Block Storage

- ✓ Data is split into blocks (default: 128MB) for storage.
- ✓ Blocks are replicated (default replication factor: 3) for fault tolerance.

Features of HDFS

1. Fault Tolerance: Automatic data replication across multiple nodes ensures reliability.

2. Scalability: Easily scales out by adding more nodes to the cluster.

3. High Throughput: Optimized for batch processing of large files.

4. Write Once, Read Many: Data is written once and read multiple times, making it ideal for analytics.

5. Streaming Data Access: Designed for high-speed data processing.

HDFS Workflow

1. Data Input: Data is divided into blocks and sent to DataNodes for storage.

2. Metadata Management: NameNode tracks the locations of all blocks.

3. Fault Tolerance: If a DataNode fails, the system retrieves the data from replicas stored on other nodes.

MCQs

1. What is the primary function of HDFS?

a) To process data in real-time

b) To store large datasets in a distributed manner

c) To manage relational databases

d) To schedule tasks in Hadoop

2. Which component of HDFS manages metadata?

a) DataNode b) NameNode

c) Secondary NameNode d) ResourceManager

3. What is the default block size in HDFS?

a) 64MB b) 128MB c) 256MB d) 512MB

4. What does the Secondary NameNode do?

a) Replaces the NameNode in case of failure

b) Stores actual data blocks

c) Periodically merges and checkpoints metadata

d) Manages task scheduling

5. How does HDFS ensure fault tolerance?

a) By using expensive hardware

b) By replicating data blocks across multiple nodes

c) By compressing data

d) By storing all data on a single node

6. What is the default replication factor in HDFS?

a) 1 b) 2 c) 3 d) 4

7. Which of the following is NOT a characteristic of HDFS?

a) Write Once, Read Many

b) Real-time data updates

c) Fault tolerance

d) Distributed storage

8. What happens if a DataNode fails in HDFS?

a) The system shuts down.

b) Data is lost permanently.

c) Data is retrieved from replicated blocks on other nodes.

d) The NameNode fails.

9. Which node in HDFS stores actual data?

a) NameNode b) DataNode

c) Secondary NameNode d) ResourceManager

- **YARN (Yet Another Resource Negotiator)**

YARN is a core component of Hadoop responsible for cluster resource management and task scheduling. It enables multiple data processing engines like MapReduce, Spark, and others to run on Hadoop, making the system more efficient and versatile.

Key Components of YARN

1. ResourceManager (Master Node):

- ✓ Global resource management and scheduling.
- ✓ Allocates resources to applications running on the cluster.

2. NodeManager (Slave Node):

- ✓ Manages resources on individual nodes.
- ✓ Monitors resource usage and reports to the ResourceManager.

3. ApplicationMaster:

- ✓ Manages the lifecycle of an application.
- ✓ Negotiates resources with the ResourceManager.

4. Container:

- ✓ A unit of resource allocation, including memory and CPU.
- ✓ Hosts tasks and applications on the nodes.

Features of YARN

1. Scalability: Efficiently handles large clusters.

2. Multi-tenancy: Supports multiple applications simultaneously.

3. Fault Tolerance: Re-allocates resources when failures occur.

4. Flexibility: Supports various processing frameworks like MapReduce, Spark, and Tez.

5. Improved Utilization: Dynamically allocates resources based on need.

YARN Workflow

1. The client submits an application to the ResourceManager.

2. The ResourceManager allocates a container for the ApplicationMaster.

3. The ApplicationMaster requests containers from the ResourceManager for task execution.

4. NodeManagers launch containers and execute tasks.

MCQs

1. What is the primary function of YARN in Hadoop?

a) Data storage

b) Resource management and task scheduling

c) Query execution

d) Metadata management

2. Which component of YARN manages resources on individual nodes?

a) ResourceManager

b) NodeManager

c) ApplicationMaster

d) NameNode

3. What is a container in YARN?

a) A storage unit for data blocks

b) A unit of resource allocation for tasks

c) A metadata storage component

d) A backup for the ResourceManager

4. What is the role of the ResourceManager in YARN?

a) To execute MapReduce jobs b) To manage resources across the cluster

c) To store data blocks d) To monitor tasks on individual nodes

5. Which YARN component manages the lifecycle of an application?

a) NodeManager

b) ApplicationMaster

c) ResourceManager

d) TaskTracker

6. What happens if the NodeManager fails?

a) The cluster shuts down.

b) Tasks on the node are rescheduled on other nodes.

c) The ResourceManager fails.

d) All running applications stop.

7. Which of the following is NOT a feature of YARN?

a) Fault tolerance

b) Data replication

c) Multi-tenancy

d) Scalability

8. What does YARN allocate to applications for execution?

a) Data blocks

b) Metadata

c) Containers

d) Files

9. How does YARN improve resource utilization?

a) By pre-allocating fixed resources to tasks

b) By dynamically allocating resources based on demand

c) By shutting down idle nodes

d) By duplicating tasks on multiple nodes

10. Which processing frameworks can run on YARN?

a) Only MapReduce

b) Only Spark

c) Any framework designed for distributed processing

d) Only Hive and Pig

UNIT – 4 Advance Database System

➢ Types of Databases

1. Relational Databases

- ✓ Organize data in tables with rows and columns.
- ✓ Examples: MySQL, PostgreSQL, Oracle, SQL Server.

2. NoSQL Databases

- ✓ Designed for unstructured or semi-structured data.
- ✓ Types: Document, Key-Value, Column-Family, Graph Databases.
- ✓ Examples: MongoDB, Cassandra, Redis, Neo4j.

3. Distributed Databases

- ✓ Data is stored across multiple locations or systems.
- ✓ Examples: Google Spanner, Amazon DynamoDB.

4. Cloud Databases

- ✓ Hosted and accessed over the cloud.
- ✓ Examples: Google Cloud SQL, AWS RDS.

5. Object-Oriented Databases

- ✓ Store objects rather than data like rows and columns.
- ✓ Examples: ObjectDB, db4o.

6. Hierarchical Databases

- ✓ Data is organized in a tree-like structure.
- ✓ Examples: IBM Information Management System (IMS).

7. Network Databases

- ✓ Use a graph structure to represent relationships.
- ✓ Examples: Integrated Data Store (IDS).

MCQs

1. Which database is based on a table structure of rows and columns?

a) NoSQL Database

b) Relational Database

c) Hierarchical Database

d) Object-Oriented Database

2. MongoDB is an example of which type of database?

a) Relational Database

b) Hierarchical Database

c) NoSQL Database

d) Network Database

3. Which type of database is best suited for handling relationships in social networks?

a) Hierarchical Database

b) Graph Database

c) Relational Database

d) Object-Oriented Database

4. What is the primary feature of a distributed database?

a) Data stored in a tree structure

b) Data stored in a centralized location

c) Data distributed across multiple systems

d) Data stored as objects

5. AWS RDS is an example of a:

a) Relational Database

b) Cloud Database

c) Hierarchical Database

d) Network Database

- **Introduction to NoSQL Databases**

NoSQL (Not Only SQL) databases are a class of databases designed for managing unstructured, semi-structured, and schema-less data. They offer flexibility, scalability, and performance advantages over traditional relational databases, particularly for modern applications such as social networks, IoT, and big data analytics.

Key Characteristics of NoSQL:

1. **Schema**-less: No fixed schema, allowing dynamic changes in the data model.

2. **Scalability**: Supports horizontal scaling for large datasets.

3. **High Performance**: Optimized for high-speed reads/writes.

4. **Distributed Architecture**: Often built for distributed systems and fault tolerance.

5. **Diverse Data Models**: Supports different types of data models.

- **Types of NoSQL Databases:**

1. Key-Value Stores:

- ✓ Data is stored as key-value pairs.
- ✓ Examples: Redis, DynamoDB.

2. Document Stores:

- ✓ Stores semi-structured data as JSON or BSON.
- ✓ Examples: MongoDB, CouchDB.

3. Column-Family Stores:

- ✓ Optimized for large-scale tabular data.
- ✓ Examples: Cassandra, HBase.

4. Graph Databases:

- ✓ Designed for managing relationships between data.
- ✓ Examples: Neo4j, ArangoDB.

- **Advantages of NoSQL:**
  1. High flexibility for evolving applications.
  2. Supports large-scale data processing.
  3. Handles diverse and complex data types.
  4. Often open-source and cost-effective.
- **Disadvantages of NoSQL:**
  1. Limited support for complex queries (compared to SQL).
  2. Lack of standardization across NoSQL databases.
  3. May require additional development effort for certain use cases.

MCQs

1. Which of the following best describes NoSQL databases?

a) They store data only in relational tables.

b) They provide support for unstructured or semi-structured data.

c) They are always slower than relational databases.

d) They cannot scale horizontally.

2. Which type of NoSQL database is optimized for managing relationships between data?

a) Document Store b) Key-Value Store

c) Graph Database d) Column-Family Store

3. MongoDB is an example of which type of NoSQL database?

a) Key-Value Store b) Document Store

c) Column-Family Store d) Graph Database

4. Which of the following is a feature of NoSQL databases?

a) Fixed schema b) Vertical scaling

c) Support for distributed data d) Mandatory use of SQL

5. Cassandra is an example of which type of NoSQL database?

a) Document Store b) Key-Value Store

c) Graph Database d) Column-Family Store

- **Need for NoSQL Databases**

The need for NoSQL databases arises due to the limitations of traditional relational databases in handling modern data and application requirements. Here are the main reasons:

Why NoSQL is Needed:

1. Handling Big Data:

Relational databases struggle with the massive data generated by IoT, social media, and web applications. NoSQL databases can efficiently process and store large volumes of data.

2. Scalability:

Traditional databases rely on vertical scaling (adding more resources to a single server), which can be expensive.

NoSQL databases support horizontal scaling (adding more servers to a cluster), providing cost-effective scalability.

3. Flexibility:

Relational databases require a fixed schema. In contrast, NoSQL databases allow schema-less designs, which are more adaptable for dynamic and evolving data structures.

4. Unstructured Data Support:

With the rise of unstructured and semi-structured data (e.g., images, videos, JSON), NoSQL databases are better equipped to handle such data types.

5. High Performance:

NoSQL databases are optimized for high-speed read and write operations, making them ideal for applications requiring real-time data processing.

6. Distributed Systems:

Modern applications are often distributed globally, and NoSQL databases are designed for distributed architectures, ensuring reliability and fault tolerance.

7. Cost-Effectiveness:

Many NoSQL solutions are open-source and can run on commodity hardware, reducing costs.

MCQs

1. Why are NoSQL databases preferred for handling big data?

a) They use fixed schemas.

b) They support structured data only.

c) They can process large volumes of unstructured data.

d) They cannot scale horizontally.

2. What makes NoSQL databases more flexible compared to relational databases?

a) Fixed table structures

b) Schema-less data modeling

c) Use of SQL for queries

d) Vertical scalability

3. What type of scalability is a key feature of NoSQL databases?

a) Vertical scalability

b) Horizontal scalability

c) Static scalability

d) Single-node scalability

4. Which of the following is NOT a reason for using NoSQL databases?

a) Handling unstructured data b) Cost-effective scalability

c) Fixed schema requirement d) Support for distributed systems

5. For real-time applications requiring high-speed data access, which type of database is suitable?

a) Relational databases b) NoSQL databases

c) File-based systems d) Hierarchical databases

- **Advantages of NoSQL Databases**

NoSQL databases offer several benefits that make them ideal for modern applications requiring flexibility, scalability, and performance.

Key Advantages of NoSQL Databases

1. Scalability:

NoSQL databases are designed for horizontal scaling, allowing the addition of more servers to handle increasing data and traffic.

2. Flexibility:

Schema-less design enables dynamic changes to data structures without disrupting operations.

3. Handling Diverse Data Types:

Supports structured, semi-structured, and unstructured data such as JSON, XML, videos, and images.

4. High Performance:

Optimized for fast read and write operations, suitable for real-time applications.

5. Distributed Systems Support:

Built for distributed architecture, ensuring fault tolerance and high availability.

6. Cost-Effectiveness:

Many NoSQL solutions are open-source and can run on commodity hardware, reducing costs.

7. Big Data and Real-Time Analytics:

Efficiently processes large volumes of data and provides insights in real time.

8. Easy Integration:

Supports integration with various tools and programming languages.

9. No Complex Joins:

Simplifies data retrieval by avoiding complex joins, common in relational databases.

10. Cloud-Friendly:

Works well with cloud-based architectures for global distribution.

MCQs

1. What is the main advantage of NoSQL databases in terms of scalability?

a) Vertical scaling b) Horizontal scaling

c) Limited scaling d) No scaling

2. Which of the following data types can NoSQL databases handle?

a) Structured data only b) Unstructured data only

c) Structured, semi-structured, and unstructured data d) Only tabular data

3. Why are NoSQL databases suitable for real-time applications?

a) They have a fixed schema.

b) They are optimized for high-speed read and write operations.

c) They lack support for distributed systems.

d) They are always slower than relational databases.

4. Which feature of NoSQL databases reduces the complexity of data retrieval?

a) Support for distributed systems

b) No need for complex joins

c) Use of fixed schemas

d) Support for SQL queries only

5. What makes NoSQL databases cost-effective?

a) Only works with structured data

b) Requires expensive hardware

c) Many are open-source and use commodity hardware

d) Requires complex schema design

6. How do NoSQL databases handle schema changes?

a) Requires downtime to update the schema

b) Schema changes are not supported

c) Automatically adapts to schema changes

d) Requires fixed schema upfront

- **SQL vs NoSQL Databases**

SQL (Relational) Databases

Structure: Organized in tables with rows and columns.

Schema: Predefined, fixed schema.

Scalability: Vertically scalable (adding more resources to the same server).

Data Relationships: Strong support for complex relationships using joins.

Query Language: Uses SQL (Structured Query Language).

Examples: MySQL, PostgreSQL, Oracle, SQL Server.

NoSQL (Non-Relational) Databases

Structure: Flexible data models (key-value, document, column-family, graph).

Schema: Schema-less or dynamic schema.

Scalability: Horizontally scalable (adding more servers to the cluster).

Data Relationships: Limited or no support for complex joins.

Query Language: Varies; may use APIs or custom query languages.

Examples: MongoDB, Cassandra, Redis, Neo4j.

MCQs

1. Which type of database uses a predefined schema?

a) SQL

b) NoSQL

c) Both SQL and NoSQL

d) Neither SQL nor NoSQL

2. What is a major advantage of NoSQL databases over SQL databases?

a) Strong support for joins

b) Fixed schema

c) Horizontal scalability

d) Limited support for unstructured data

3. Which type of database is better suited for handling structured data?

a) SQL

b) NoSQL

c) Both SQL and NoSQL

d) None of the above

4. Which of the following is an example of a NoSQL database?

a) MySQL

b) PostgreSQL

c) MongoDB

d) Oracle

5. What makes NoSQL databases more flexible than SQL databases?

a) Predefined schema

b) Schema-less design

c) Complex relationships

d) Dependence on SQL

6. SQL databases are typically scaled by:

a) Adding more servers (horizontal scaling)

b) Adding more resources to the same server (vertical scaling)

c) Both horizontal and vertical scaling equally

d) Neither horizontal nor vertical scaling

7. Which type of database is ideal for applications with rapidly changing data models?

a) SQL

b) NoSQL

c) Both SQL and NoSQL

d) None of the above

8. Which query language is used by SQL databases?

a) Structured Query Language (SQL)

b) JSON Query Language (JQL)

c) NoSQL Query Language

d) Custom APIs

- **Introduction to Different Types of NoSQL Databases**

NoSQL databases are categorized based on their data models and use cases. Here's an overview of the different types:

1. Key-Value Stores

Description:

Data is stored as key-value pairs, similar to a dictionary or hashmap.

Use Cases:

Caching, session management, and real-time analytics.

Examples:

Redis, DynamoDB, Riak.

2. Document Stores

Description:

Stores data as documents, usually in JSON, BSON, or XML format.

Each document contains key-value pairs and can have nested structures.

Use Cases:

Content management systems, catalog data, and event logging.

Examples:

MongoDB, CouchDB, RavenDB.

3. Column-Family Stores

Description:

Data is stored in columns rather than rows, grouped into families.

It is optimized for large-scale data retrieval and analytics.

Use Cases:

Data warehousing, analytics, and time-series data.

Examples: Cassandra, HBase, ScyllaDB.

4. Graph Databases

Description:

Designed to store and navigate relationships between entities using nodes, edges, and properties.

Use Cases:

Social networks, recommendation engines, and fraud detection.

Examples:

Neo4j, ArangoDB, Amazon Neptune.

MCQs

1. Which type of NoSQL database stores data in key-value pairs?

a) Document Store b) Column-Family Store

c) Key-Value Store d) Graph Database

2. MongoDB is an example of which type of NoSQL database?

a) Document Store b) Key-Value Store

c) Column-Family Store d) Graph Database

3. Which type of NoSQL database is optimized for managing relationships between entities?

a) Document Store b) Graph Database

c) Column-Family Store d) Key-Value Store

4. What type of NoSQL database is best suited for analytics and time-series data?

a) Key-Value Store

b) Column-Family Store

c) Graph Database

d) Document Store

5. Redis is an example of which type of NoSQL database?

a) Document Store

b) Key-Value Store

c) Column-Family Store

d) Graph Database

6. Which type of NoSQL database is most suitable for storing hierarchical data in JSON format?

a) Document Store

b) Column-Family Store

c) Key-Value Store

d) Graph Database

7. Which of the following NoSQL databases is a graph database?

a) Cassandra

b) Neo4j

c) MongoDB

d) Redis

8. Which NoSQL database type is best for storing large-scale tabular data?

a) Document Store

b) Column-Family Store

c) Key-Value Store

d) Graph Database

UNIT – 5 Data Analytics

Data analytics refers to the process of examining raw data to uncover patterns, draw conclusions, and support decision-making. It involves techniques and tools to clean, transform, and analyze data in order to gain insights and make data-driven decisions. The process may include statistical analysis, data mining, machine learning, and visualization to interpret trends and relationships within the data.

MCQs

Question: What is the primary goal of data analytics?

A) To store large amounts of data

B) To create complex algorithms

C) To extract valuable insights from data

D) To visualize data in graphs and charts

1. Which of the following is a key step in the data analytics process?

A) Data collection

B) Data visualization

C) Data cleaning

D) All of the above

2. What type of analysis is used to make predictions based on historical data?

A) Descriptive analysis

B) Diagnostic analysis

C) Predictive analysis

D) Prescriptive analysis

3. Which of the following tools is commonly used for data visualization?

A) Excel B) Power BI C) Tableau D) All of the above

4. Which of these is a type of unstructured data?

A) A database table

B) A text document

C) A CSV file

D) A spreadsheet

5. What is the purpose of data cleaning in the analytics process?

A) To ensure data is accurate and consistent

B) To increase the volume of data

C) To analyze the data visually

D) To create new data

6. How can data analytics be used in sports?

A) To improve team performance and player health

B) To reduce stadium maintenance costs

C) To increase ticket sales

D) To decide on player salaries

The use of data analytics spans across various fields and industries, enabling organizations to gain valuable insights, make data-driven decisions, and optimize processes.

key uses of data analytics:

1. Business Decision-Making

Use: Data analytics helps businesses to analyze performance, identify trends, and make better decisions regarding marketing, operations, and overall strategy.

Example: Retailers use analytics to predict customer preferences and optimize inventory management.

2. Customer Insights and Personalization

Use: By analyzing customer behavior and preferences, businesses can personalize offerings, improving customer experiences and satisfaction.

Example: E-commerce platforms like Amazon recommend products based on user browsing history and past purchases.

3. Fraud Detection and Risk Management

Use: Data analytics is widely used in financial institutions to detect fraudulent activities and mitigate risks by analyzing transaction patterns and anomalies.

Example: Credit card companies use data analytics to identify unusual spending patterns, flagging potential fraud.

4. Predictive Analytics

Use: Predictive models analyze historical data to forecast future events and trends, aiding in proactive decision-making.

Example: Airlines use predictive analytics to forecast flight demand, optimize pricing, and manage seat availability.

5. Healthcare Improvement

Use: In healthcare, data analytics is used to predict patient outcomes, improve diagnoses, and enhance treatment plans by analyzing patient data.

Example: Healthcare providers use analytics to identify patterns in patient data, helping doctors predict and prevent chronic conditions like diabetes.

6. Supply Chain Optimization

Use: Companies utilize data analytics to optimize their supply chain, reduce costs, and improve logistics by predicting demand and improving inventory management.

Example: Manufacturing companies use analytics to streamline production schedules and minimize downtime.

7. Sports Performance

Use: In sports, data analytics helps coaches and teams improve player performance, design effective strategies, and monitor physical conditions.

Example: Football teams use data analytics to track player movements and performance, making tactical decisions during games.

8. Marketing and Advertising

Use: Data analytics aids marketers in measuring campaign effectiveness, segmenting audiences, and optimizing advertising strategies.

Example: Social media platforms analyze user engagement data to target ads effectively, leading to better conversion rates.

9. Human Resources Management

Use: Data analytics is used in HR for employee performance analysis, recruitment, and retention strategies.

Example: Companies use employee data to predict turnover rates and implement strategies for retention.

10. Energy and Utility Management

Use: Data analytics helps monitor energy usage, optimize supply distribution, and improve sustainability practices.

Example: Smart meters in homes use analytics to track energy consumption, helping users optimize usage and reduce costs.

➢ Stages of the Data Analytics Life Cycle:

1. Problem Definition

- ✓ Understand the problem, define objectives, and determine what questions need answering.

2. Data Collection

- ✓ Gather the required data from various sources, which could include databases, sensors, or external datasets.

3. Data Cleaning and Preprocessing

- ✓ Clean and preprocess the data by removing inconsistencies, handling missing values, and transforming data into a usable format.

4. Exploratory Data Analysis (EDA)

- ✓ Analyze the data through visualization and statistical methods to identify patterns, relationships, and insights.

5. Model Building

- ✓ Select and apply appropriate algorithms and models to analyze the data and make predictions or classifications.

6. Model Evaluation

- ✓ Assess the performance of the model using evaluation metrics (accuracy, precision, recall, etc.).

7. Deployment and Monitoring

- ✓ Deploy the model into the production environment and monitor its performance over time to ensure it continues to meet objectives.

MCQs

1. What is the first step in the Data Analytics Life Cycle?

A) Data Collection B) Problem Definition

C) Model Building D) Data Cleaning and Preprocessing

2. In which stage of the Data Analytics Life Cycle is data cleaned and transformed for analysis?

A) Model Evaluation B) Exploratory Data Analysis

C) Data Collection D) Data Cleaning and Preprocessing

3. Which of the following is the primary goal of Exploratory Data Analysis (EDA)?

A) To collect data from external sources B) To build a predictive model

C) To visualize and summarize the data to find patterns and insights

D) To deploy the model into production

4. At which stage of the Data Analytics Life Cycle is the model's performance evaluated?

A) Data Collection B) Model Building C) Model Evaluation D) Problem Definition

5. Which of the following describes the deployment and monitoring stage of the Data Analytics Life Cycle?

A) Applying models to historical data B) Collecting and cleaning data

C) Putting the model into use and tracking its performance D) Visualizing the data

6. What is the purpose of the "Problem Definition" stage in the Data Analytics Life Cycle?

A) To collect data from different sources B) To identify the specific questions to answer and goals to achieve

C) To build a predictive model D) To evaluate the model's accuracy

- **Types of Analytics:**

1. Descriptive Analytics

- ✓ Focuses on understanding historical data and summarizing what has happened. It answers questions like "What happened?" through methods like data aggregation and visualization.

2. Diagnostic Analytics

- ✓ Goes a step further than descriptive analytics by investigating the reasons behind past outcomes. It answers "Why did it happen?" through techniques like correlation analysis and root cause analysis.

3. Predictive Analytics

- ✓ Uses statistical models and machine learning techniques to forecast future outcomes. It answers "What could happen?" by analyzing patterns in historical data to make predictions.

4. Prescriptive Analytics

- ✓ Recommends actions to achieve desired outcomes by using optimization and simulation techniques. It answers "What should we do?" to optimize decisions and strategies.

5. Cognitive Analytics

- ✓ Involves advanced AI techniques that simulate human thought processes. It focuses on improving decision-making by learning from experience, often using natural language processing (NLP) and machine learning.

MCQs

1. Which type of analytics answers the question, "What happened?"

A) Predictive Analytics

B) Prescriptive Analytics

C) Descriptive Analytics

D) Cognitive Analytics

2. What is the primary goal of diagnostic analytics?

A) To summarize past events B) To predict future outcomes

C) To understand why something happened D) To recommend the best course of action

3. Which type of analytics is used to predict future outcomes based on historical data?

A) Prescriptive Analytics B) Cognitive Analytics

C) Predictive Analytics D) Descriptive Analytics

4. Which of the following is a feature of prescriptive analytics?

A) It predicts future trends based on past data. B) It answers why something occurred in the past.

C) It recommends the best course of action to optimize outcomes.

D) It visualizes and summarizes historical data.

5. Which type of analytics involves using AI to simulate human thought processes?

A) Predictive Analytics B) Descriptive Analytics C) Cognitive Analytics D) Diagnostic Analytics

6. What is the key difference between descriptive and diagnostic analytics?

A) Descriptive analytics focuses on predicting future events, while diagnostic analytics focuses on past events.

B) Descriptive analytics summarizes historical data, while diagnostic analytics explores the reasons behind past events.

C) Descriptive analytics makes recommendations, while diagnostic analytics predicts outcomes.

D) Descriptive analytics uses AI, while diagnostic analytics does not.

Thank You…