**General Data Science & Python Basics**

1. **What is Data Science?**
   * It is a multidisciplinary field that uses scientific methods, algorithms, and systems to extract knowledge from structured and unstructured data.
2. **What is the difference between supervised and unsupervised learning?**
   * Supervised learning uses labeled data (e.g., regression, classification), whereas unsupervised learning finds patterns in unlabeled data (e.g., clustering).
3. **What are some key Python libraries used in Data Science?**
   * NumPy, Pandas, Matplotlib, Seaborn, Scikit-learn, Statsmodels, TensorFlow, PySpark.
4. **What are data frames in Pandas?**
   * A 2D labeled data structure with columns of potentially different types.
5. **How is missing data handled in Pandas?**
   * Using isnull(), dropna(), fillna() for detection and imputation.

**🔹 Data Wrangling & EDA**

1. **What is data wrangling?**
   * The process of cleaning, structuring, and enriching raw data for analysis.
2. **What are common steps in data cleaning?**
   * Handling missing values, removing duplicates, fixing data types, handling outliers.
3. **How do you detect and handle outliers?**
   * Using boxplots or statistical methods like Z-score and IQR.
4. **What is the purpose of groupby() in Pandas?**
   * To split data into groups for aggregation or transformation.
5. **How do you merge two datasets in Pandas?**
   * Using merge(), concat(), or join() functions.

**11normaliza and types**

| **Name** | **Description** | **Formula / Example** |
| --- | --- | --- |
| **Min-Max Normalization** | Scales data between a fixed range (usually 0 and 1). | X – min(x) / max(x) – min(x) |
| **Z-score Normalization (Standardization)** | Scales data to have mean = 0 and standard deviation = 1. | x−μ/​σ meu is mean and sigma is sid devia |

**🔹 Data Visualization (Matplotlib, Seaborn)**

1. **What are the main types of plots in Matplotlib?**
   * Line, bar, histogram, scatter, pie, boxplot.
2. **What does a boxplot show?**
   * It shows distribution, median, and outliers using quartiles.
3. **What is the difference between plt.plot() and plt.scatter()?**
   * plot() is for continuous lines; scatter() is for point-wise plotting.
4. **What Seaborn plot is best for correlation?**
   * Heatmap with sns.heatmap(df.corr()).
5. **How can visualizations help in feature selection?**
   * They help identify variable importance and relationships (e.g., via pairplot, heatmaps).

**🔹 Statistics for Data Science**

1. **What is a p-value?**
   * It is the probability that the observed data would occur by chance if the null hypothesis is true.
2. **What is correlation?**
   * A measure of the relationship between two variables; ranges from -1 to +1.
3. **What is the difference between population and sample?**
   * Population includes all data points; a sample is a subset used for inference.
4. **What is standard deviation?**
   * It measures the amount of variation or dispersion in a dataset.
5. **What are descriptive statistics?**
   * Mean, median, mode, range, variance, standard deviation, etc.

**🔹 Regression (Linear, Polynomial, Multiple)**

1. **What is linear regression?**
   * A technique to model the relationship between dependent and one/more independent variables.
2. **What does R² indicate in regression?**
   * The proportion of variance in the dependent variable explained by the independent variable(s).
3. **How is polynomial regression different from linear?**
   * It models the relationship using higher-degree polynomials to capture non-linear trends.
4. **What assumptions does linear regression make?**
   * Linearity, normality, no multicollinearity, homoscedasticity, independence.
5. **How to evaluate regression models?**
   * Metrics: R², MAE, MSE, RMSE.

**🔹 Classification (Logistic, KNN, Decision Tree)**

1. **What is logistic regression?**
   * A classification algorithm used for binary outcomes (e.g., yes/no).
2. **What is KNN and how does it work?**

KNN is a **lazy, instance-based learning** algorithm, meaning it doesn't learn an explicit model but memorizes the training data.  
**Working Steps:**

1. **Choose K**: the number of nearest neighbors to consider.
2. **Calculate distance**: Use metrics like **Euclidean or Manhattan distance** to compute how far each training point is from the test point.
3. **Sort and vote**: Select the K closest points and do majority voting (classification) or average (regression).
4. **Output prediction**.

KNN works under the assumption that **similar instances have similar outputs**. It performs well on well-separated data but is computationally expensive at prediction time.

--------------------------------------------

1. **What is a confusion matrix?**
   * A table showing TP, TN, FP, FN used to evaluate classification performance.

**29.ROC curve** is a **graph** that shows the performance of a classification model at different thresholds.

**📈 ROC Plot:**

* **X-axis**: False Positive Rate (FPR)
* **Y-axis**: True Positive Rate (TPR or Recall)

Each point on the curve represents a **(FPR, TPR)** pair for a particular threshold

 AUC is the **area under the ROC curve**.

 It ranges from **0 to 1**.

Higher it is best for us

1. **What are precision, recall, and F1 score?**
   * Precision = TP / (TP + FP), Recall = TP / (TP + FN), F1 = 2\*(Precision\*Recall)/(Precision+Recall)
2. **What is entropy in decision trees?**
   * A measure of impurity; lower entropy means more homogeneity.

31.**decision tree work\_**

A Decision Tree is built by recursively splitting the dataset based on a feature that best separates the classes.  
It uses **measures like Gini Index or Information Gain** (based on Entropy) to decide the best feature.

**Example Working:**

1. Start at the root node and evaluate all features.
2. Calculate **Information Gain** for each feature:

IG=Entropy(parent)−∑ (instances in child/instances in parent×Entropy(child))

1. Select the feature with the highest gain.
2. Repeat recursively on subnodes until all nodes are pure (same class) or max depth is reached.

It works like a flowchart: starting from a root node, data is split until the output class is clear.

**32.naive bayes-**

**Base on conditional probab, prop of a given that b already happen p(a I b) = p(A n B)/p(B)**

**Bayes is use to update our current belief if new extra info avai-**

* **p(a I b) = p(b I a)\*p(a)/p(B) it need 2 pieces of info**
* **eg-** P(Rain)=0.3.
* P(Cloudy)=0.4→ 40% of days are cloudy.
* P(Cloudy∣Rain)=0.8→ If it rains, there’s an 80% chance it was cloudy.
* Find p(r/c)

**🔹 Clustering & Dimensionality Reduction**

1. **What is K-means clustering?**
   * An unsupervised algorithm that partitions data into K clusters by minimizing intra-cluster variance.

How K-Means Works (Step-by-step):

Choose the number of clusters (K).

Initialize K centroids randomly from the dataset.

Assign each point to the nearest centroid (using Euclidean distance).  
→ This forms K clusters.

Update centroids: For each cluster, compute the mean of all points in the cluster and move the centroid to this mean.

Repeat steps 3 and 4 until:

* + Centroids don’t change (convergence), or
  + A max number of iterations is reached.

1. **What is the elbow method?**
   * A method to choose optimal K in K-means by plotting inertia vs. K.
2. **What is PCA (Principal Component Analysis)?**
   * A technique to reduce the dimensionality of data while preserving variance.
3. **How do you interpret principal components?**
   * As new axes (linear combinations of original features) capturing maximum variance.
4. **Why is feature scaling important before PCA?**
   * Because PCA is sensitive to variances and scales of variables.

**🔹 Text Analytics**

1. **What is tokenization?**
   * Splitting text into individual words or tokens.
2. **What are stopwords?**
   * Common words (like "the", "is") removed during preprocessing.
3. **What is TF-IDF?**
   * Term Frequency–Inverse Document Frequency; measures importance of a word in a document. Formula = no. time term appear / total no. terms
4. **What is stemming and lemmatization?**
   * Techniques to reduce words to their root form.
5. **How is text converted to numeric for ML models?**

Using techniques like Bag-of-Words, TF-IDF, or word embeddings.

Bow - **Bag of Words** is a fundamental technique in **Natural Language Processing (NLP)** used to convert **text data into numerical form** so that it can be used in machine learning models. Basically lik eonehotencoding

-----------------------------------------------------------------------------------------------------

Data – collection of raw, unorganized facts , INFO – Processed and organized data

Data/ 2 type=

1.categorial(qualitative)- has labels or names eg ,gender

and is / into nominal- has no order

and ordinal has order.

2.numerical(quantitative)- has numbers eg 20,1

and is / in discrete- include countable no. that can be counted one by one eg- age ,

and continuous that has measurable no. eg height

----------------------------------------------------------------------------------------------------------------

1. **Q: What is Impala?**  
   **A:** Impala is an open-source massively parallel processing (MPP) SQL query engine for processing large volumes of data stored in Hadoop Distributed File System (HDFS) using SQL-like syntax. It enables real-time, low-latency queries.
2. **Q: How is Impala different from Hive?**  
   **A:** Hive converts SQL queries to MapReduce jobs, which have higher latency. Impala executes queries directly using its own execution engine, making it faster for real-time analytics.
3. **Q: What are the main components of Impala?**  
   **A:**
   * impalad: Query execution engine.
   * impala-catalog: Manages metadata caching.
   * impala-statestore: Tracks health of nodes.
   * impala-shell: CLI to interact with Impala.
4. **Q: What file formats does Impala support?**  
   **A:** TEXT, Parquet, ORC, Avro, RCFile, etc.
5. **Q: Where does Impala store metadata?**  
   **A:** It shares metadata with Hive using the Hive Metastore.
6. **Q: How do you launch Impala shell?**  
   **A:** Run impala-shell on the terminal after starting all Impala services.
7. **Q: Can Impala handle unstructured data?**  
   **A:** No, Impala is best suited for structured/tabular data.
8. **Q: How is data loaded into Impala tables?**  
   **A:** Impala can load data from HDFS using LOAD DATA, or use INSERT statements for manual insertion.

## 🔹 **Experiment 12: Scala with Apache Spark**

### 🔸 Conceptual Questions:

1. **Q: What is Apache Spark?**  
   **A:** A distributed computing framework for big data processing, offering APIs in Scala, Java, Python, and R. It supports batch and real-time processing.
2. **Q: Why is Scala used in Spark?**  
   **A:** Spark is written in Scala. It is fast, concise, functional, and has seamless integration with Spark APIs.
3. **Q: What is the difference between RDD and DataFrame?**  
   **A:**
   * RDD: Low-level, unstructured, immutable distributed collection of objects.
   * DataFrame: Structured, optimized for performance, similar to SQL tables.
4. **Q: What is transformation and action in Spark?**  
   **A:**
   * **Transformation:** Lazy operations (e.g., map, filter) that define a new RDD.
   * **Action:** Triggers execution (e.g., count, collect, saveAsTextFile).
5. **Q: How does Spark achieve fault tolerance?**  
   **A:** Through RDD lineage — it tracks how data is derived, so lost partitions can be recomputed.
6. **Q: What is lazy evaluation in Spark?**  
   **A:** Spark doesn’t compute data until an action is called, allowing for optimization.
7. **Q: What is .reduceByKey(\_ + \_)?**  
   **A:** It adds values for each key. In word count, it sums the occurrences of each word.
8. **Q: Why do we use saveAsTextFile()?**  
   **A:** It writes the output of an RDD to a directory in the file system.
9. **Q: What happens if the output folder already exists?**  
   **A:** Spark will throw an error. The directory must not exist prior.
10. **Q: Can you run Spark on Windows?**  
    **A:** Yes, using WSL (Windows Subsystem for Linux) or Docker, but Spark is optimized for Linux systems.

**1. Label Encoding**

* **What it does:** Assigns a unique integer (e.g., 0, 1, 2...) to each category.
* **Example:**

Categories: [Red, Green, Blue]

Label Encoded: [0, 1, 2]

**2. One-Hot Encoding**

* **What it does:** Converts each category into a binary vector, where only one bit is "1" (hot) and the rest are "0".
* **Example:**

Categories: [Red, Green, Blue]

One-Hot Encoded:

Red -> [1, 0, 0]

Green -> [0, 1, 0]

Blue -> [0, 0, 1]

**Measu cent ten** – mean mode median(measure of where center of data lie)

**Measu pf dispersion** – how data is spread , **vari** = how far data from mean sum (x – mean)^2 / N ,**STD DEV** IS DIST FROM MEAN