



# TOPICS

# Python and AI Library

## NumPy

- Stands for **Numerical Python**.
- Used for **fast mathematical operations** on large datasets.
- Provides powerful **array (matrix)** support and tools for numerical computation.
- Foundation for other AI libraries like Pandas, TensorFlow, and Scikit-learn.

*Example:* Doing quick calculations on arrays of numbers.

# Python and AI Library

- ◆ **Pandas**

- Used for **data analysis and manipulation**.
- Works with **DataFrames** — like tables in Excel (rows & columns).
- Makes it easy to **clean**, **filter**, and **analyze** data.
- Can read and write data from CSV, Excel, or databases.
-  *Example:* Analyzing student marks stored in an Excel sheet.

# Python and AI Library

- ◆ **Scikit-learn**
  - A **machine learning library** built on top of NumPy and Pandas.
  - Contains ready-made algorithms for:
    - **Regression** (predicting values)
    - **Classification** (categorizing data)
    - **Clustering** (grouping similar data)
  - Also supports **data preprocessing** and **model evaluation**.
  -  *Example:* Predicting house prices or detecting spam emails.

# Python and AI Library

## TensorFlow / PyTorch

- Libraries for **deep learning** and **AI model development**.
- Help build **neural networks** that learn from large data sets.
- **TensorFlow** (by Google): Best for production and large-scale systems.
- **PyTorch** (by Meta): Easier to use and great for research.
-  Example: Training AI to recognize images or understand text.

# Data Handling

## Cleaning

- Removing errors, duplicates, or missing data from datasets.
  - 💡 Example: Deleting blank rows from a CSV file.

### ◆ Preprocessing

- Converting raw data into a usable format (e.g., converting text to numbers).
  - 💡 Example: Scaling numeric data or encoding categories.

### ◆ Feature Engineering

- Selecting or creating important data columns (features) that improve model performance.
  - 💡 Example: Creating an “age group” column from a person’s birth year.

# Machine Learning Basics

## Regression

- Predicting continuous values.
  - 💡 Example: Predicting house prices or marks.

### ♦ Classification

- Categorizing data into classes.
  - 💡 Example: Spam or Not Spam, Pass or Fail.

### ♦ Clustering

- Grouping similar data without predefined labels.
  - 💡 Example: Grouping customers by buying behavior.

### ♦ Model Evaluation

- Checking how accurate your trained model is using test data.
  - 💡 Example: Measuring prediction accuracy or error rate.

# GenAI and Prompt Engineering

- ◆ **OpenAI**

- The company behind models like **GPT-4** and **ChatGPT**.
- Provides APIs for text generation, summarization, and more.

- ◆ **LangChain**

- A framework to build apps powered by LLMs.
- Helps connect models with external data and tools.

- ◆ **RAG (Retrieval-Augmented Generation)**

- Technique that combines **stored data** + **LLM output** for more accurate answers.
  - 💡 Example: Chatbot that searches your documents before replying.

- ◆ **LLM Integration**

- Connecting AI models (like GPT) into your own apps or systems using APIs.

# Deployment Tools

- ◆ **Flask / FastAPI**
  - Lightweight Python frameworks used to turn ML models into web APIs.
- ◆ **Docker**
  - Used to package your application and all dependencies into a container.
    - 💡 Makes it easy to deploy anywhere.
- ◆ **Git**
  - Version control tool to track code changes and collaborate with teams.
- ◆ **Cloud Hosting (AWS / Azure)**
  - Platforms to host and run your applications online.
    - 💡 Example: Deploying an AI model on AWS EC2 or Azure App Service.