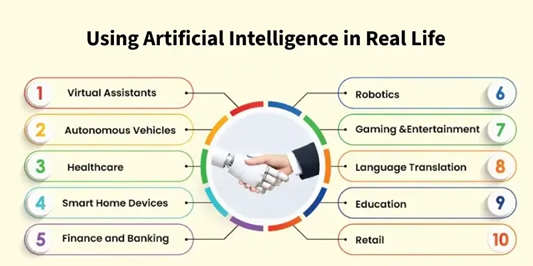
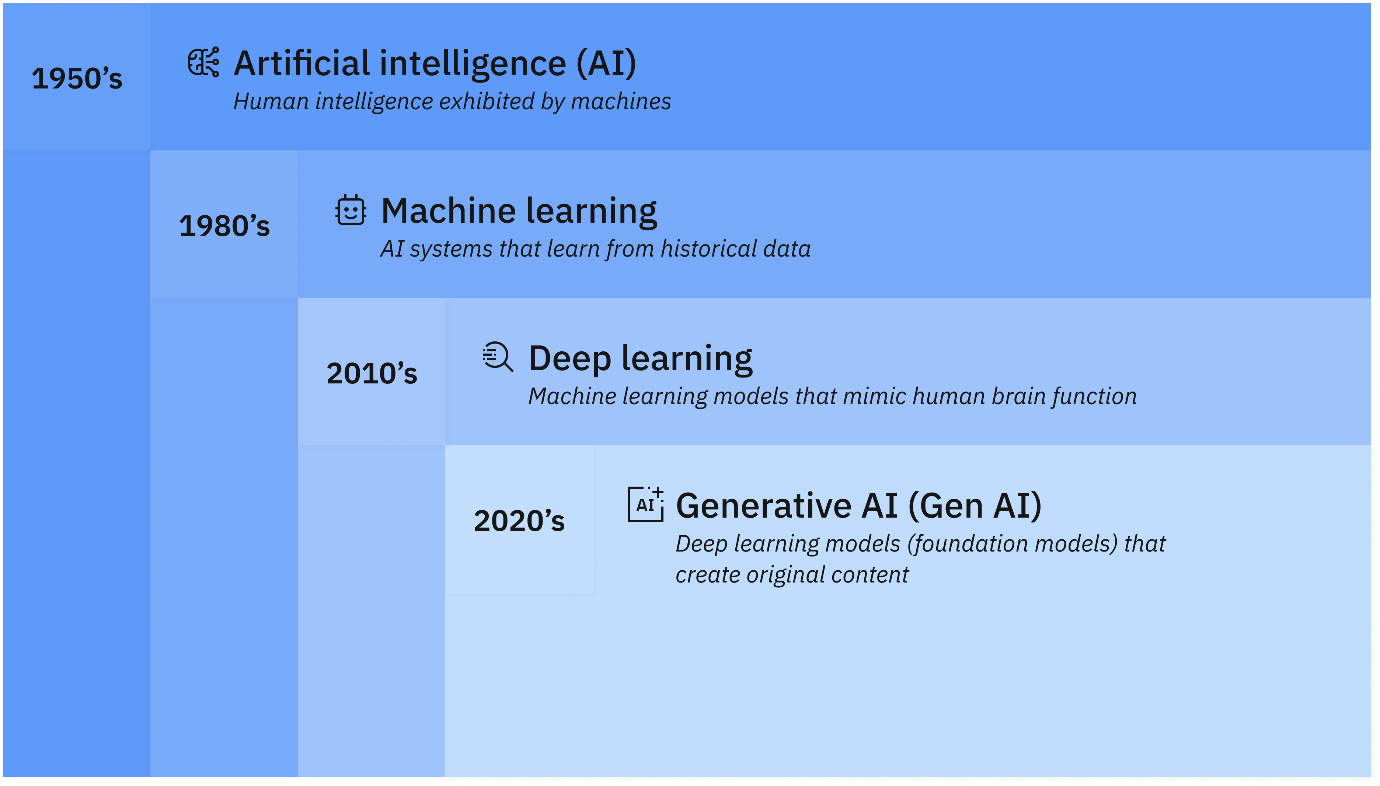
**Lab 01**

Artificial Intelligence (AI) is a type of technology that allows computers and machines to think and act like humans. It helps them learn from experience, understand information, solve problems, make decisions, be creative, and work on their own without constant human instructions.



**Machine Learning (ML)** is a branch of Artificial Intelligence (AI) that focuses on teaching computers to learn from data and make decisions or predictions without being directly programmed for each specific task.

Machine learning is when a computer learns from experience just like humans do.



|  |  |
| --- | --- |
| **Traditional AI** | **Machine Learning** |
| Rule-based, logic-driven | Data-driven, statistical |
| Expert systems, planning | Neural networks, decision trees |
| Symbolic reasoning | Pattern recognition from data |

**AI refers to the goal**: building machines that behave intelligently.

**ML refers to the approach**: teaching machines to learn from data

**For Data Preprocessing**

**Data:** Data refers to raw facts and figures that are collected for analysis. It can be in the form of numbers, text, images, videos, or audio.

**Dataset:** A dataset is a structured collection of related data used for analysis or to train machine learning models. While it is commonly organized in rows and columns like a table, datasets can also consist of other formats such as images, text, audio, video, or sensor data.

**Missing Values:** Missing values occur when no data is stored for a certain variable in an observation. These gaps can affect model performance, so they are usually handled using techniques like deletion or imputation (filling with mean, median, etc.).

**Duplicate Values:** Duplicate values refer to repeated rows or records in a dataset. These are generally removed because they can bias the analysis or model results, especially if they're unintentional.

**Categorical columns:** Categorical columns are features in a dataset that represent categories or groups rather than numerical values. Examples include Gender (Male/Female), Color (Red, Green, Blue), or Diagnosis (Benign, Malignant).

**Normalization:** Normalization is a technique used to scale numerical data into a fixed range, usually [0, 1]. It ensures that no single feature dominates due to its scale.

### **Min-Max Normalization**

**Purpose:** Scales the data between a specific range, usually **0 and 1**.

**Formula:**

Suppose we have the following data:

Original Data: [20, 40, 60, 80, 100]

Here,

Now apply the formula to each value:

| Original (x) | Min-Max Normalized |
| --- | --- |
| 20 | (20−20)/(100−20) = 0 |
| 40 | (40−20)/(100−20) = 0.25 |
| 60 | (60−20)/(100−20) = 0.5 |
| 80 | (80−20)/(100−20) = 0.75 |
| 100 | (100−20)/(100−20) = 1 |

### **Z-score Normalization (Standardization)**

**Purpose:** Scales data to have **mean = 0** and **standard deviation = 1**.

**Formula:**

Where:

* is the mean
* is the standard deviation

Using the same data:

Original Data: [20, 40, 60, 80, 100]

* Mean () = (20+40+60+80+100)/5 = 60
* Standard Deviation () = 28.28 (approx)

Now apply the formula:

| Original (x) | Z-score Normalized |
| --- | --- |
| 20 | (20−60)/28.28 ≈ -1.41 |
| 40 | (40−60)/28.28 ≈ -0.71 |
| 60 | (60−60)/28.28 = 0 |
| 80 | (80−60)/28.28 ≈ 0.71 |
| 100 | (100−60)/28.28 ≈ 1.41 |

**Libraries**

**NumPy (Numerical Python):**

NumPy is mainly used for numerical data that is, numbers in arrays or matrices. It creates arrays where all elements must be of the same type, like all integers or all floats. It is fast and efficient when we want to perform operations like adding, multiplying, or finding the average of large groups of numbers.

import numpy as np

arr = np.array([10, 20, 30])

print(arr \* 2) # Output: [20 40 60]

**Pandas:**

Pandas is a powerful Python library built on top of NumPy and is primarily used for working with structured data, such as tables from Excel or CSV files. Pandas is especially useful for data analysis tasks such as reading data from files (like CSV or Excel), handling missing values, filtering and sorting data, and working with various data types including numbers, text, and dates.

import pandas as pd

data = {'Name': ['Amit', 'Priya'], 'Marks': [85, 90]}

df = pd.DataFrame(data)

print(df)

**Label Encoder** is a technique used to convert categorical (text) data into numerical form so that machine learning algorithms can process it. Most ML models can’t work with strings, so we transform the labels (categories) into numbers.

from sklearn.preprocessing import LabelEncoder

colors = ['Red', 'Green', 'Blue', 'Red', 'Blue']

encoder = LabelEncoder()

encoded\_colors = encoder.fit\_transform(colors)

print(encoded\_colors) # Output: [2 1 0 2 0]

**Some basic things:**

**Load or Read the data:**

df = pd.read\_csv('data.csv')

**Explore the data:**

df.head() # First 5 rows

df.shape # Rows and columns

df.info() # Summary of columns and types

df.describe() # Stats for numerical columns

**Handle the missing values:**

df.isnull().sum() # Count missing values per column

df = df.dropna() # Drop rows with any missing values

df['col\_name'].fillna(0) # Fill missing with 0 or a value

df.fillna(df.mean(), inplace=True) # Fill with mean

**Check duplicates**

df.duplicated().sum() # Check number of duplicate rows

df.drop\_duplicates(inplace=True) # Remove them

**Normalizing the data**

from sklearn.preprocessing import MinMaxScaler, StandardScaler

scaler = MinMaxScaler() # 0 to 1

df[['Age', 'Salary']] = scaler.fit\_transform(df[['Age', 'Salary']])

**To sort and filter:**

df.sort\_values(by='Age', ascending=False, inplace=True)

df\_filtered = df[df['Salary'] > 50000]

**Create a new column:**

df['Age\_Category'] = df['Age'].apply(lambda x: 'Adult' if x >= 18 else 'Minor')

**Create a graph**

import pandas as pd

import matplotlib.pyplot as plt

df = pd.read\_csv('students.csv')

# Simple bar chart

plt.bar(df['Name'], df['Marks'])

plt.xlabel('Name')

plt.ylabel('Marks')

plt.title('Student Marks')

plt.show()

**Save the data:**

df.to\_csv('cleaned\_data.csv', index=False)