## UNIT–I: Syllabus

**Introduction-**Artificial Intelligence, Machine Learning, Deep learning and Data Science, Types of Machine Learning Systems: supervised, unsupervised, semi-supervised, Reinforcement, why machine learning, Problems machine learning can solve, Main Challenges of Machine Learning, Essential libraries and tools for machine learning-Jupyter Notebook, NumPy, Pandas, Matplotlib, Scikit Learn.

Artificial Intelligence, Machine Learning, Deep learning, and Data Science:

These four terms are often used interchangeably but have distinct meanings and applications.

Data Science:

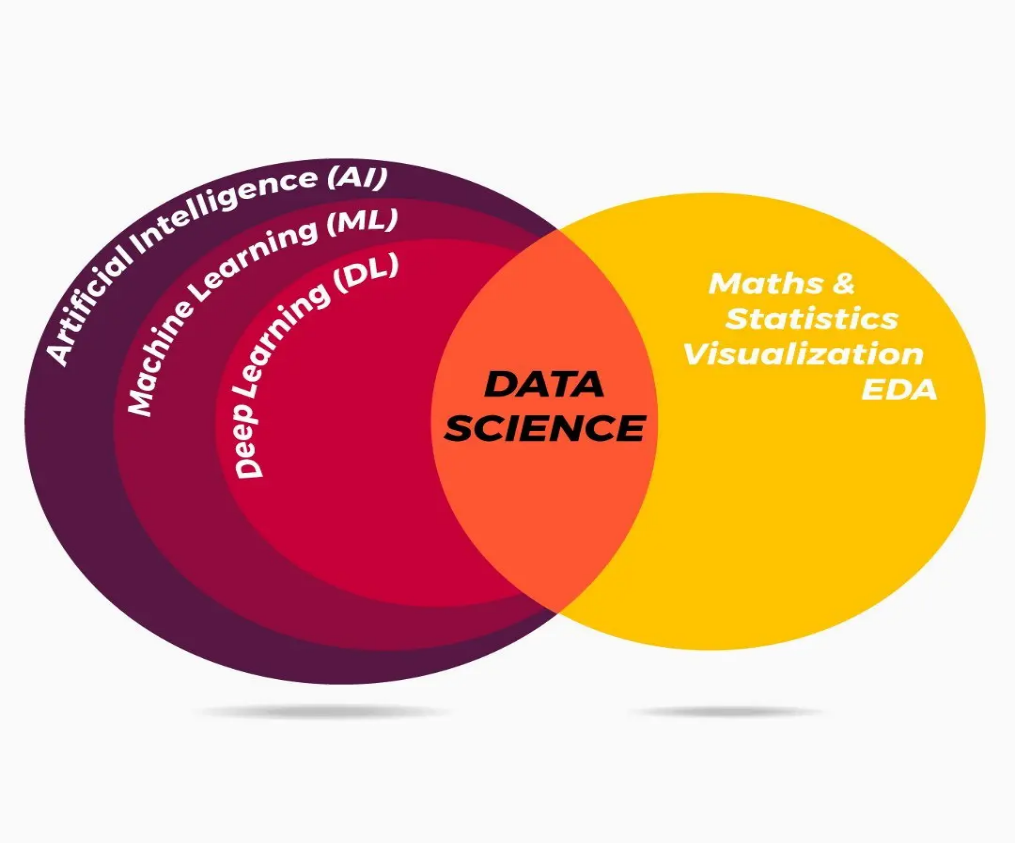
Data science is a field of study that combines domain expertise, programming skills, and knowledge of mathematics and statistics to extract meaningful insights from data.

Data science applies Machine Learning and Deep learning algorithms to numbers, text, images, audio and video etc to produce Artificial Intelligence Systems. These systems generate insights for Business users that can be translated into tangible business value.

Artificial Intelligence is the concept of creating smart intelligent machines.

Machine Learning is a subset of artificial intelligence that helps you build [AI-driven applications.](https://www.simplilearn.com/tutorials/artificial-intelligence-tutorial/artificial-intelligence-applications)

Deep Learning is a subset of machine learning that uses vast volumes of data and complex algorithms to train a model.



Artificial Intelligence:

* Ability of a machine to imitate intelligent human behavior.
* Artificial Intelligence is the concept of creating smart intelligent machines.
* Artificial intelligence, commonly referred to as AI, is the process of imparting data, information, and human intelligence to machines.
* The main goal of Artificial Intelligence is to develop self-reliant machines that can think and act like humans.
* These machines can mimic human behavior and perform tasks by learning and problem-solving. Most of the AI systems simulate natural intelligence to solve complex problems.

| **Aspect** | **Data Science** | **Artificial Intelligence (AI)** | **Machine Learning (ML)** | **Deep Learning (DL)** |
| --- | --- | --- | --- | --- |
| **Definition** | A multidisciplinary field that extracts insights from data using statistics, machine learning, and domain expertise. | The broader concept of creating machines that simulate human intelligence. | A subset of AI where algorithms learn from data to make predictions or decisions. | A specialized subset of ML that uses neural networks to learn from vast amounts of data. |
| **Scope** | Broad (includes AI, ML, statistics, big data, etc.) | Broad (includes ML, DL, expert systems, robotics, etc.) | Narrower (focused on learning from data) | Very specific (focuses on deep neural networks) |
| **Techniques Used** | Statistics, data mining, visualization, ML, big data processing | Rule-based systems, ML, DL, robotics, NLP, expert systems | Supervised, unsupervised, reinforcement learning | Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Transformers, GANs |
| **Purpose** | Extract insights and solve data-driven problems | Enable machines to mimic human cognitive functions | Enable computers to learn from experience | Process complex patterns and make sophisticated predictions |
| **Example Applications** | Business intelligence, fraud detection, healthcare analytics | Chatbots, self-driving cars, robotics, expert systems | Spam detection, recommendation systems, stock price prediction | Image recognition, speech recognition, autonomous vehicles |
| **Dependency on Data** | High | Moderate to High | High | Very High (needs large datasets) |

**Types of Machine Learning Systems: supervised, unsupervised, semi-supervised, Reinforcement**

Machine Learning Definition:

Machine Learning is the science (and art) of programming computers so they can learn from data.

Here is a slightly more general definition:

Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed. —Arthur Samuel, 1959

A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E. —Tom Mitchell, 1997

**An example to understand Tom Mitchell definition:**

For example, your spam filter is a Machine Learning program that can learn to flag spam given examples of spam emails (e.g., flagged by users) and examples of regular (non spam, also called “ham”) emails. The examples that the system uses to learn are called the training set. Each training example is called a training instance (or sample).

In this case, the task T is to flag spam for new emails, the experience E is the training data, and the perform

ance measure P needs to be defined; for example, you can use the ratio of correctly classified emails. This particular performance measure is called accuracy and it is often used in classification tasks.

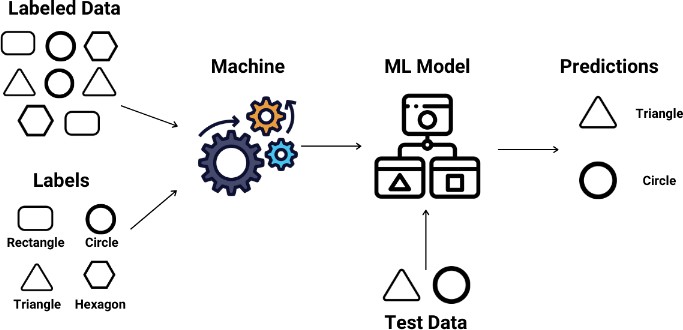
Types of Machine Learning Systems

# **Supervised learning**

**Definition**: In supervised learning, the model is trained on a labeled dataset, which means that for every input, the corresponding output is provided. The goal is to learn a mapping from inputs to outputs, so the model can make accurate predictions on unseen data.

# Key Characteristics:

* + - **Labeled Data**: The training data consists of input-output pairs (e.g., input: an image of a cat, output: "cat").
    - **Goal**: Learn a function that maps inputs to outputs by minimizing the difference between predicted outputs and actual outputs.

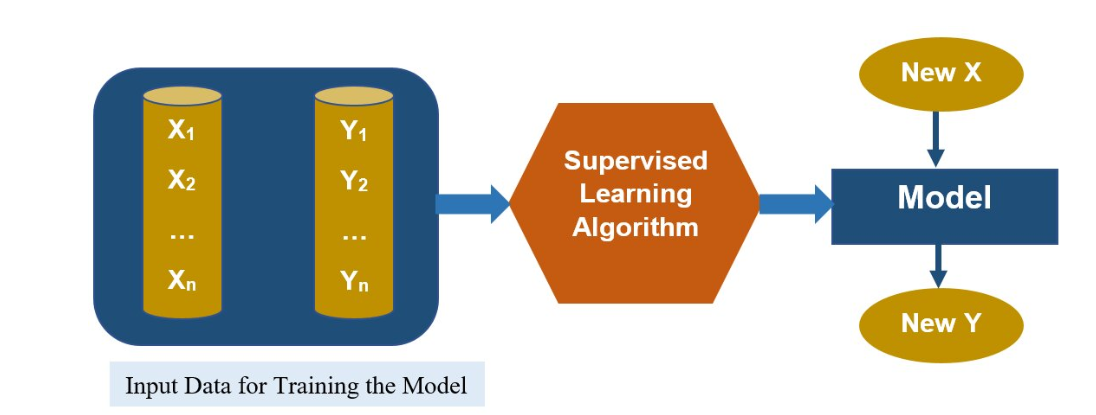


# Example:

1. Predicting house prices based on features like size, location, and number of bedrooms. You have a dataset of houses with known prices (labeled data) and the task is to predict the price of a new house.

Supervised Learning:

This learning works with labelled data. A set of X, and Y (pre-classified training examples) are given. The model is trained on this data and then for a new observation (input) X, the model tells us the predicted value of Y.



Given a set of Input features X1,X2,….Xn.

A target feature Y

A set of training examples where the value for the input features (X1,X2,….Xn) and the corresponding target features (Y1,Y2,…,Yn) are given for each example.

A new example (Test data)where only the values for the input features are given.

We must predict the values for the target feature (Y) for the new example(Test data).

This learning works with labelled data. A set of X, and Y (pre-classified training examples) are given. The model is trained on this data and then for a new observation (input) X, the model tells us the predicted value of Y.

Types of Supervised Learning:

1. Regression
2. Classification

Regression – Given a set of input features X1, X2, …, Xn, a target feature Y is predicted. Y is a continuous function. Examples of regression are predicting housing prices, predicting sales volume, predicting market demand, etc. Regression could be linear or nonlinear.

Classification – The target feature Y is discrete. i.e. It could either be binary or multiclass. An example of binary Y (output) is ‘Yes’, or ‘No’ (as in predicting whether someone would be Covid positive given his weight, age, co-morbidity, etc.). An example of multiclass Y is predicting the grade of a student (out of A,B,C,D,F), predicting the type of weather (sunny, rainy, cloudy with no rain, part sunny and part cloudy), etc.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | X1 | X2 | …….. | Xn | Target(Y) |
| I1 | A1 | A2 |  | An | Y1 |
| I2 | B1 | B2 |  | Bn | Y2 |
| I3 | C1 | C2 |  | Cn | Y3 |
|  |  |  |  |  |  |
| Test Instance | Z1 | Z2 |  | Zn | Yn |

Here are some of the most important supervised learning algorithms:

• k-Nearest Neighbors

• Linear Regression

• Logistic Regression

• Support Vector Machines (SVMs)

• Decision Trees and Random Forests

• Neural networks

# **Unsupervised Learning**

**Definition**: In unsupervised learning, the model is trained on unlabeled data, meaning the model must find patterns or structures in the data without any explicit feedback (no labels). The goal is to discover hidden patterns or groupings in the data.

# Key Characteristics:

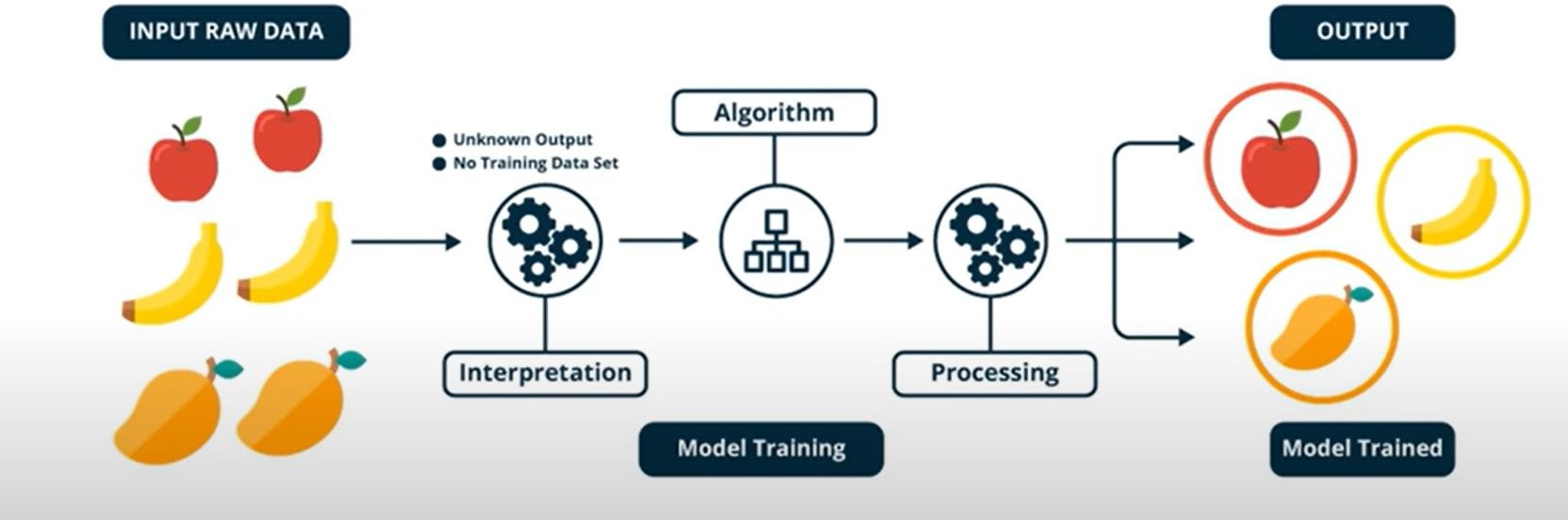
* + - **Unlabeled Data**: The data consists of inputs only, with no corresponding output labels (e.g., a dataset of images without any information on what they contain).
    - **Goal**: Identify patterns, clusters, or associations in the data.

**Example1**: Customer segmentation in marketing. Given a dataset of customers' purchase behavior, unsupervised learning could group customers into segments based on similar behaviors, without knowing ahead of time which customer belongs to which segment.

**Example2**: The machine looks at a dataset of fruit images and organizes them into three clusters based on similarities, such as color and shape. One cluster might consist of all red round objects (apples), another of long yellow objects (bananas), and another of oval yellow- orange objects (mangoes).

# How It Works:

1. **Unlabeled Data**: There are no labels saying "apple," "banana," or "mango."
2. **Clustering**: The machine clusters the data based on the inherent similarities, forming groups of similar fruits without knowing their actual names.



Here are some of the most important unsupervised learning algorithms.

• Clustering

—K-Means

—DBSCAN

—Hierarchical Cluster Analysis (HCA)

• Anomaly detection and novelty detection

—One-class SVM

—Isolation Forest

• Visualization and dimensionality reduction

—Principal Component Analysis (PCA)

—Kernel PCA

—Locally-Linear Embedding (LLE)

—t-distributed Stochastic Neighbor Embedding (t-SNE)

• Association rule learning

—Apriori

—Eclat

**3.Semisupervised learning:**

Some algorithms can deal with partially labeled training data, usually a lot of unlabeled data and a little bit of labeled data. This is called semisupervised learning.

# 

Example: Some photo-hosting services, such as Google Photos, are good examples of this. Once you upload all your family photos to the service, it automatically recognizes that the same person A shows up in photos 1, 5, and 11, while another person B shows up in photos 2, 5, and 7. This is the unsupervised part of the algorithm (clustering). Now all the system needs is for you to tell it who these people are. Just one label per person, and it is able to name everyone in every photo, which is useful for searching photos.

# **Reinforcement Learning (RL)**

**Definition**:

* In reinforcement learning, the model learns by interacting with an environment. The model makes decisions, observes the results (rewards or penalties), and adjusts its actions

to maximize cumulative rewards over time.

* Unlike supervised learning, RL does not require labeled input/output pairs and instead relies on a system of rewards and punishments.

**Key Concepts of Reinforcement Learning:**

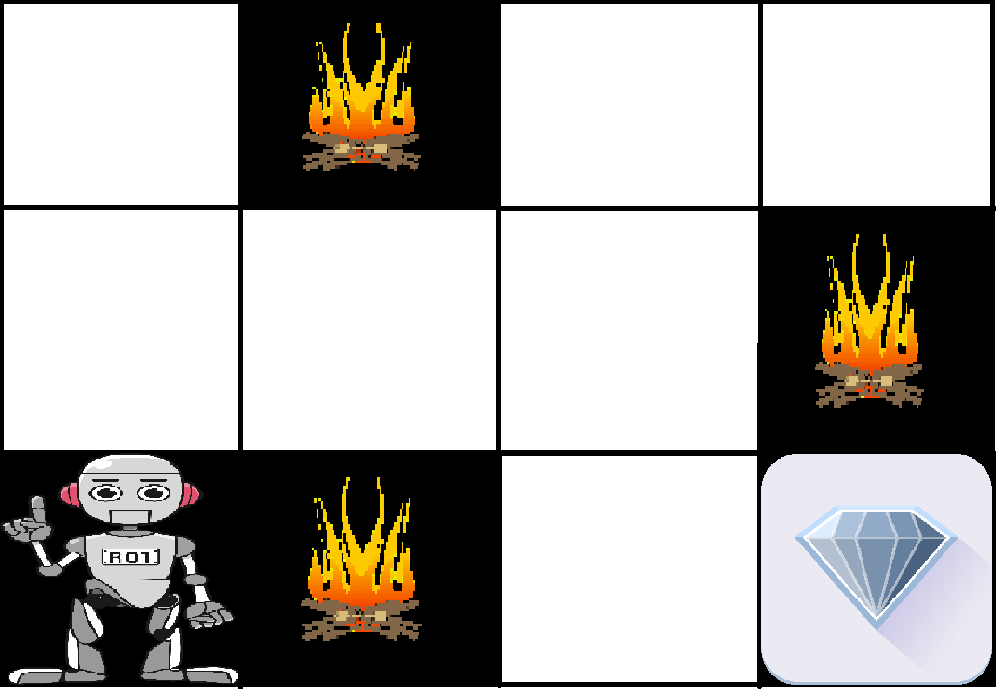
* Agent: The learner or decision-maker.
* Environment: Everything the agent interacts with.
* State: A specific situation in which the agent finds itself.
* Action: All possible moves the agent can make.
* Reward: Feedback from the environment based on the action taken.

**How Reinforcement Learning Works:**

* RL operates on the principle of learning optimal behavior through trial and error.
* The agent takes actions within the environment, receives rewards or penalties, and adjusts its behavior to maximize the cumulative reward.
* This learning process is characterized by the following elements:
* Policy: A strategy used by the agent to determine the next action based on the current state.
* Reward Function: A function that provides a scalar feedback signal based on the state and action.
* Value Function: A function that estimates the expected cumulative reward from a given state.
* Model of the Environment: A representation of the environment that helps in planning by predicting future states and rewards.

**Example**: **Navigating a Maze**

The problem is as follows: We have an agent and a reward, with many hurdles in between. The agent is supposed to find the best possible path to reach the reward. The following problem explains the problem more easily.

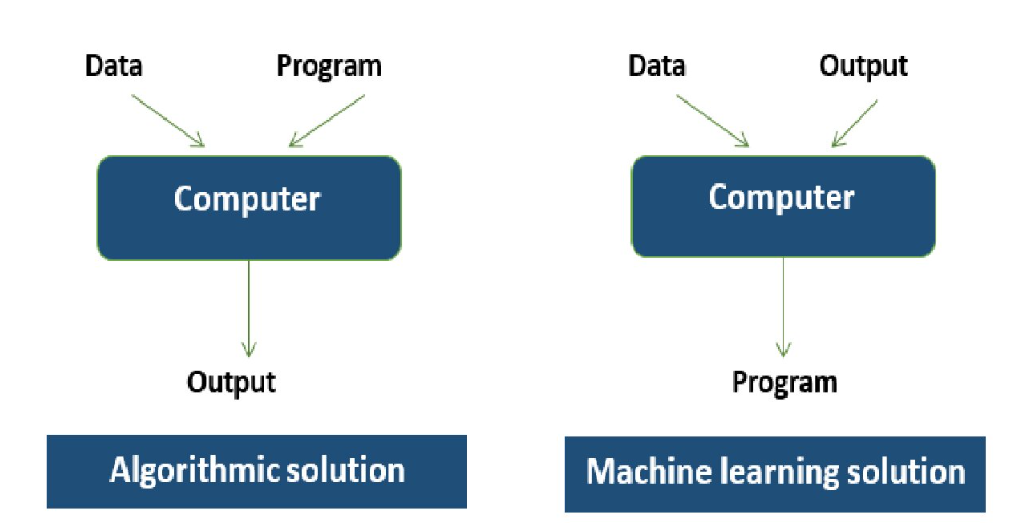


The above image shows the robot, diamond, and fire. The goal of the robot is to get the reward that is the diamond and avoid the hurdles that are fired. The robot learns by trying all the possible paths and then choosing the path which gives him the reward with the least hurdles. Each right step will give the robot a reward and each wrong step will subtract the reward of the robot. The total reward will be calculated when it reaches the final reward that is the diamond.

**Why Machine Learning:**

**Traditional Approach (Rule-Based Programming)**

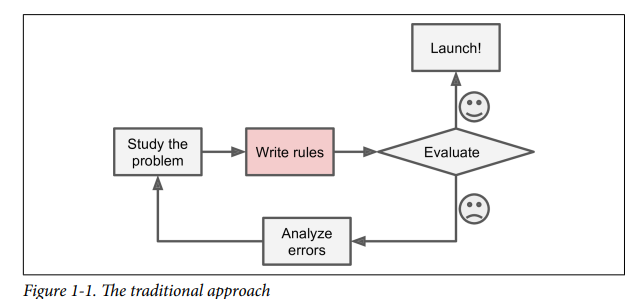
The traditional approach relies on explicitly programmed rules and logic defined by human experts. It requires developers to manually create decision trees, conditions, and algorithms to solve a problem. This method is effective for simple, structured tasks where rules are well-defined and do not change frequently.



1. First you would look at what spam typically looks like. You might notice that some words or phrases (such as “4U,” “credit card,” “free,” and “amazing”) tend to come up a lot in the subject. Perhaps you would also notice a few other patterns in the sender’s name, the email’s body, and so on.

2. You would write a detection algorithm for each of the patterns that you noticed, and your program would flag emails as spam if a number of these patterns are detected.

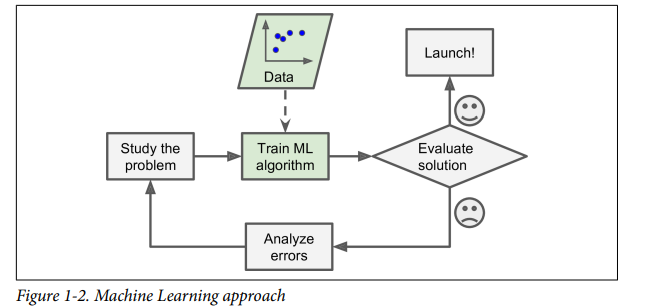
3. You would test your program, and repeat steps 1 and 2 until it is good enough.



Since the problem is not trivial, your program will likely become a long list of complex

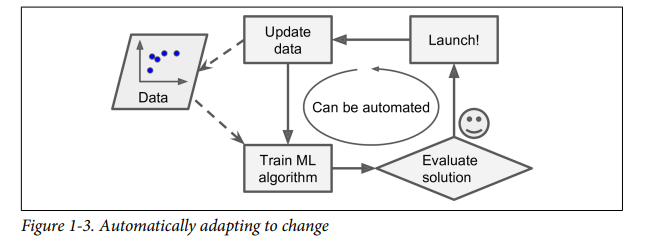
rules—pretty hard to maintain.

In contrast, a spam filter based on Machine Learning techniques automatically learns which words and phrases are good predictors of spam by detecting unusually frequent patterns of words in the spam examples compared to the ham examples. The program is much shorter, easier to maintain, and most likely more accurate.



Moreover, if spammers notice that all their emails containing “4U” are blocked, they might start writing “For U” instead. A spam filter using traditional programming techniques would need to be updated to flag “For U” emails. If spammers keep work‐ ing around your spam filter, you will need to keep writing new rules forever.

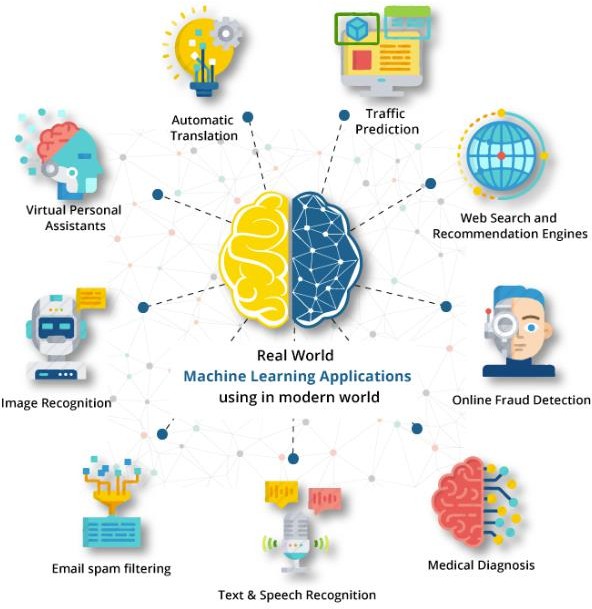
In contrast, a spam filter based on Machine Learning techniques automatically noti‐ ces that “For U” has become unusually frequent in spam flagged by users, and it starts flagging them without your intervention.



Machine Learning excels at solving complex problems where traditional approaches fail or no known algorithm exists. For example, speech recognition cannot rely on simple rule-based methods due to variations in pronunciation, noise, and multiple languages. Instead, ML models learn patterns from vast amounts of data, enabling accurate predictions.

**Problems machine learning can solve:**

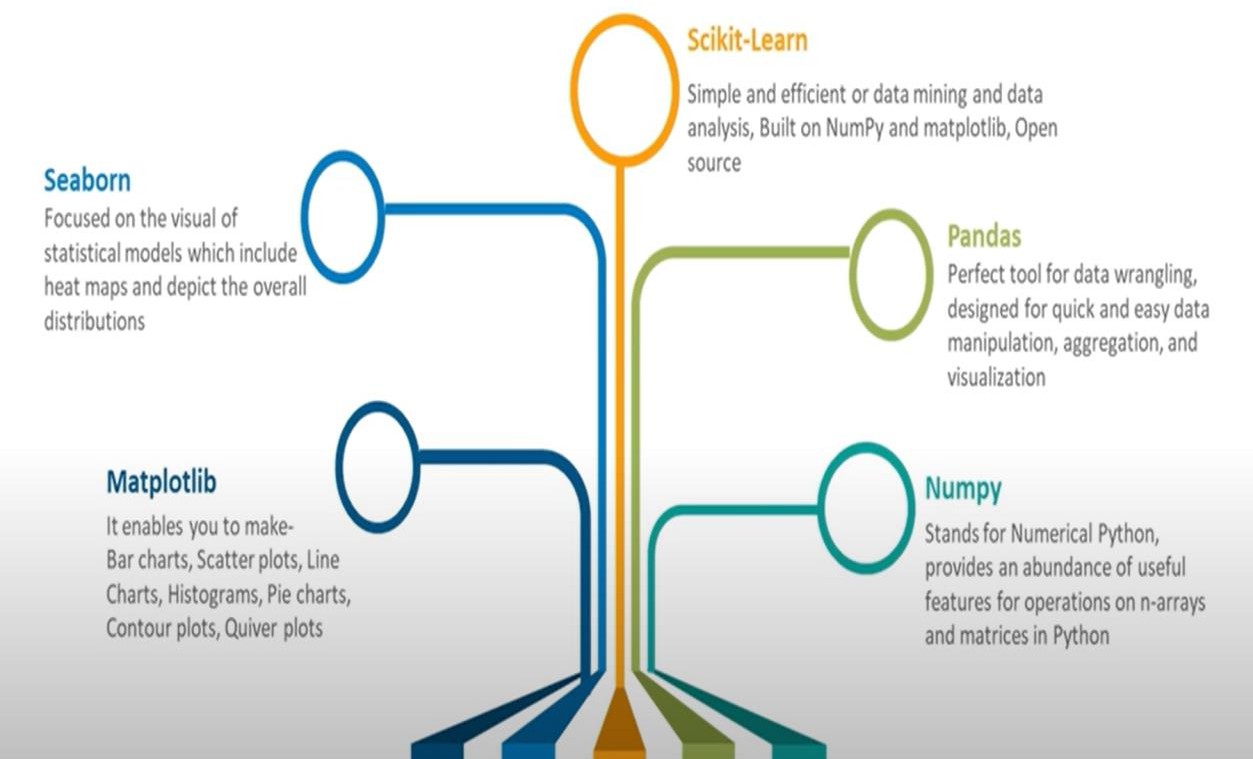
**Where Machine learning is used?**



Machine Learning (ML) is applied across numerous industries, solving complex problems and automating tasks. Here are some key applications of ML across various domains:

1. **Healthcare**
   * **Disease Prediction and Diagnosis**: ML algorithms are used to analyze medical data (e.g., scans, patient records) to predict diseases like cancer, diabetes, and cardiovascular diseases early on.
   * **Personalized Medicine**: Machine learning models help in tailoring treatment plans based on an individual's genetic makeup, lifestyle, and response to previous treatments.
   * **Drug Discovery**: ML accelerates the drug discovery process by predicting how different compounds might behave in biological systems.
   * **Medical Imaging**: Algorithms assist in detecting anomalies in medical scans, such as X- rays, MRIs, and CT scans, improving diagnostic accuracy.
2. **Finance**
   * **Fraud Detection**: ML models analyze transactions to detect fraudulent behavior in real-time by identifying patterns that deviate from the norm.
   * **Algorithmic Trading**: Financial institutions use machine learning to predict market movements and execute trades at optimal times for maximum profit.
   * **Credit Scoring**: ML is applied to assess the creditworthiness of individuals by analyzing financial history, spending patterns, and other variables.
   * **Risk Management**: Financial firms use ML models to predict risks associated with lending, investments, and other financial activities.
3. **Retail and E-commerce**
   * **Recommendation Systems**: ML algorithms suggest products to customers based on their browsing history, past purchases, and similar customer preferences (e.g., Amazon, Netflix).
   * **Price Optimization**: Retailers use ML to dynamically adjust prices based on demand, competition, and customer behavior.
   * **Inventory Management**: Machine learning models predict demand for products, helping businesses optimize inventory levels and reduce overstock or stockouts.
   * **Customer Segmentation**: ML helps businesses segment customers based on their purchasing behavior, enabling personalized marketing strategies.
4. **Transportation**
   * **Autonomous Vehicles**: Machine learning is critical in enabling self-driving cars by processing real-time data from sensors (e.g., cameras, radar) and making decisions about driving behavior.
   * **Traffic Prediction**: ML models predict traffic conditions by analyzing data from road sensors, historical traffic patterns, and current events, helping navigation apps like Google Maps or Waze suggest optimal routes.
   * **Predictive Maintenance**: In logistics and transportation industries, machine learning helps predict when vehicles or machinery will require maintenance, preventing breakdowns and optimizing fleet management.
5. **Marketing and Advertising**
   * **Targeted Advertising**: ML algorithms analyze consumer data to deliver personalized ads to the right audience at the right time, optimizing ad spend and increasing conversion rates.
   * **Sentiment Analysis**: Machine learning analyzes social media posts, reviews, and other content to gauge public sentiment toward brands or products.
   * **Customer Lifetime Value Prediction**: ML models estimate how much revenue a business can expect from a customer over time, enabling better resource allocation and retention strategies.
6. **Manufacturing**
   * **Quality Control**: ML models analyze images and sensor data to detect defects in products during manufacturing processes, improving quality assurance.
   * **Predictive Maintenance**: Machine learning predicts machinery failures before they occur, enabling timely maintenance and reducing downtime.
   * **Supply Chain Optimization**: ML models forecast demand, optimize inventory management, and streamline supply chain operations, ensuring that resources are used efficiently.
7. **Education**
   * **Personalized Learning**: ML algorithms create adaptive learning systems that tailor educational content to a student’s pace, learning style, and performance.
   * **Automated Grading**: Machine learning models assist in grading assignments and exams by analyzing written content, multiple-choice responses, or even programming code.
   * **Student Retention**: Machine learning models predict which students are at risk of dropping out by analyzing behavioral patterns and academic performance, enabling early intervention.
8. **Entertainment**
   * **Content Recommendations**: Platforms like Netflix, YouTube, and Spotify use machine learning to suggest videos, songs, and shows based on user preferences and viewing/listening habits.
   * **Content Creation**: Machine learning models assist in creating music, artwork, or writing scripts, blending AI creativity with human creativity.
   * **Personalized User Experience**: Machine learning helps platforms optimize their interfaces, making them more user-friendly and personalized to individual preferences.
9. **Agriculture**
   * **Crop Yield Prediction**: ML models predict crop yields based on data from satellite imagery, weather patterns, and soil conditions.
   * **Precision Agriculture**: Machine learning enables precision farming by analyzing data from sensors and drones to monitor crop health, soil quality, and water use.
   * **Pest and Disease Detection**: ML algorithms analyze images of crops to detect diseases or pest infestations early, helping farmers take corrective action before major damage occurs.

**Essential libraries and Tools**



# NumPy

NumPy – Numerical python is a very popular python library for array and matrix processing, with the help of a large collection of high-level mathematical functions.

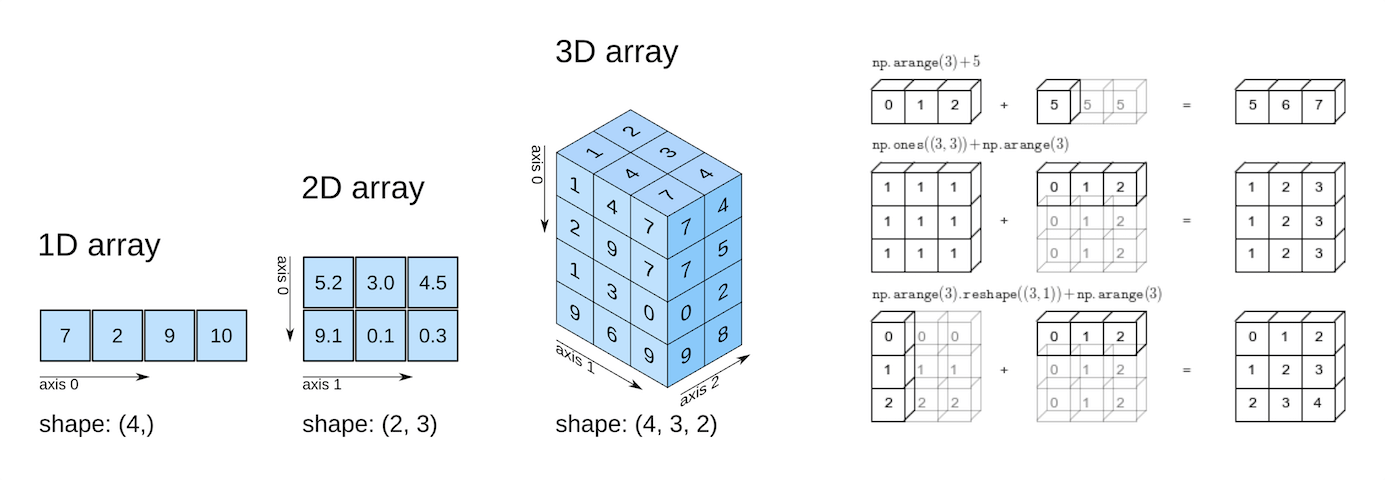
* + It is very useful for fundamental scientific computations in Machine Learning.

Purpose: Provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays.

Features:

* + - Array Operations: Efficiently performs element-wise operations on arrays.
    - Mathematical Functions: Includes functions for linear algebra, statistical operations, and mathematical operations.
    - Broadcasting: Allows operations on arrays of different shapes and sizes.

Use Case: Often used for handling data arrays and performing numerical computations.



# Pandas

Pandas-Panel data is a popular Python library for data analysis.

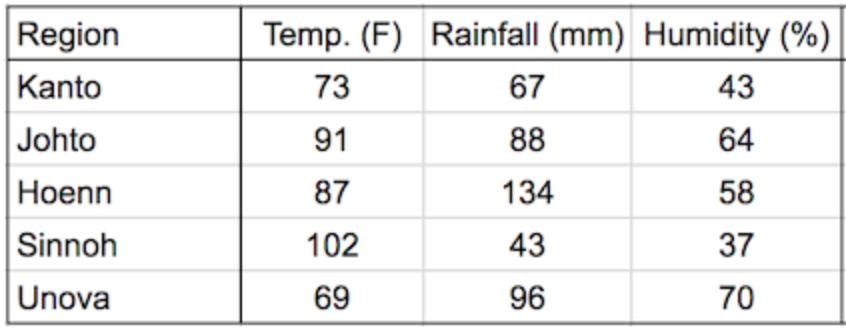
* + It is not directly related to Machine Learning but the dataset must be prepared before training for which Pandas are useful as it is developed specifically for data extraction and preparation.
  + It provides data structures and wide variety tools for data analysis. It provides many inbuilt methods for filtering, combining and grouping data.

Purpose: Provides data structures and data analysis tools. It is particularly useful for working with structured data and performing data manipulation.

Features:

* + - DataFrames: A two-dimensional, size-mutable, and heterogeneous tabular data structure.
    - Data Manipulation: Includes functionality for data cleaning, filtering, merging, and aggregation.
    - Time Series: Specialized tools for time series analysis.

Use Case: Ideal for data preprocessing, cleaning, and exploratory data analysis.



# Matplotlib

Matplotlib is a Python library for data visualization. Like Pandas, it is not directly related to Machine Learning. It is needed when a programmer wants to visualize the patterns in the data.

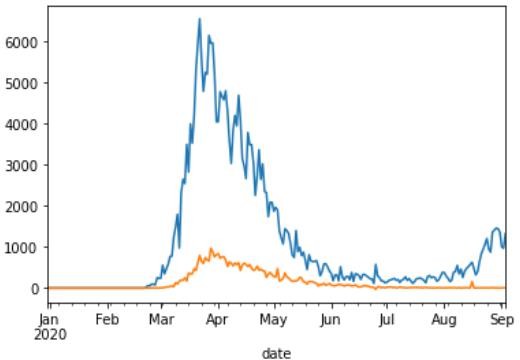
A module named pyplot makes it easy for programmers for plotting .

Purpose: A plotting library for creating static, animated, and interactive visualizations in Python.

Features:

* + Plot Types: Includes a variety of plot types such as line plots, bar charts, histograms, scatter plots, and more.
  + Customization: Provides extensive options for customizing plots, including labels, colors, and styles.
  + Integration: Works well with NumPy and Pandas for visualizing data.

Use Case: Essential for visualizing data and results of machine learning models



# Scikit-Learn

Scikit-learn is one of the most popular ML libraries for classical ML algorithms.

Scikit-learn supports most of the supervised and unsupervised learning algorithms

Purpose: Provides simple and efficient tools for data mining and data analysis, including a range of machine learning algorithms and utilities.

Features:

* + Algorithms: Implements various supervised and unsupervised learning algorithms (e.g., regression, classification, clustering).
  + Model Selection: Tools for model evaluation, cross-validation, and hyperparameter tuning.
  + Preprocessing: Functions for data preprocessing, feature extraction, and normalization.

Use Case: Useful for building, evaluating, and deploying machine learning models.

# 4. Seaborne

Seaborn is a Python visualization library based on Matplotlib that provides a high-level interface for drawing attractive and informative statistical graphics.

* It is particularly useful for making complex statistical plots with minimal code and for visualizing data distributions and relationships.

Key Features of Seaborn

* Statistical Plots:

Distribution Plots: Visualize the distribution of a dataset with histograms, KDE plots, or rug plots.

Regression Plots: Plot data and fit regression lines, along with confidence intervals.

Categorical Plots: Include bar plots, box plots, violin plots, and swarm plots to display categorical data.

* Data Visualization:

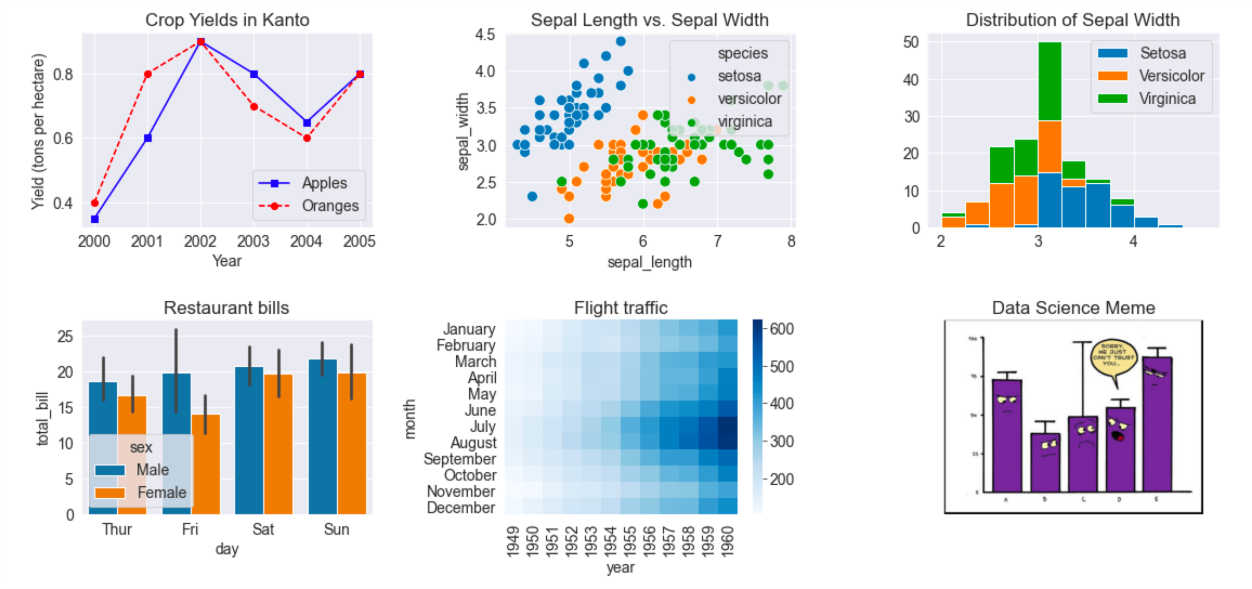
Pair Plots: Show pairwise relationships in a dataset and can include histograms or KDE plots on the diagonal.

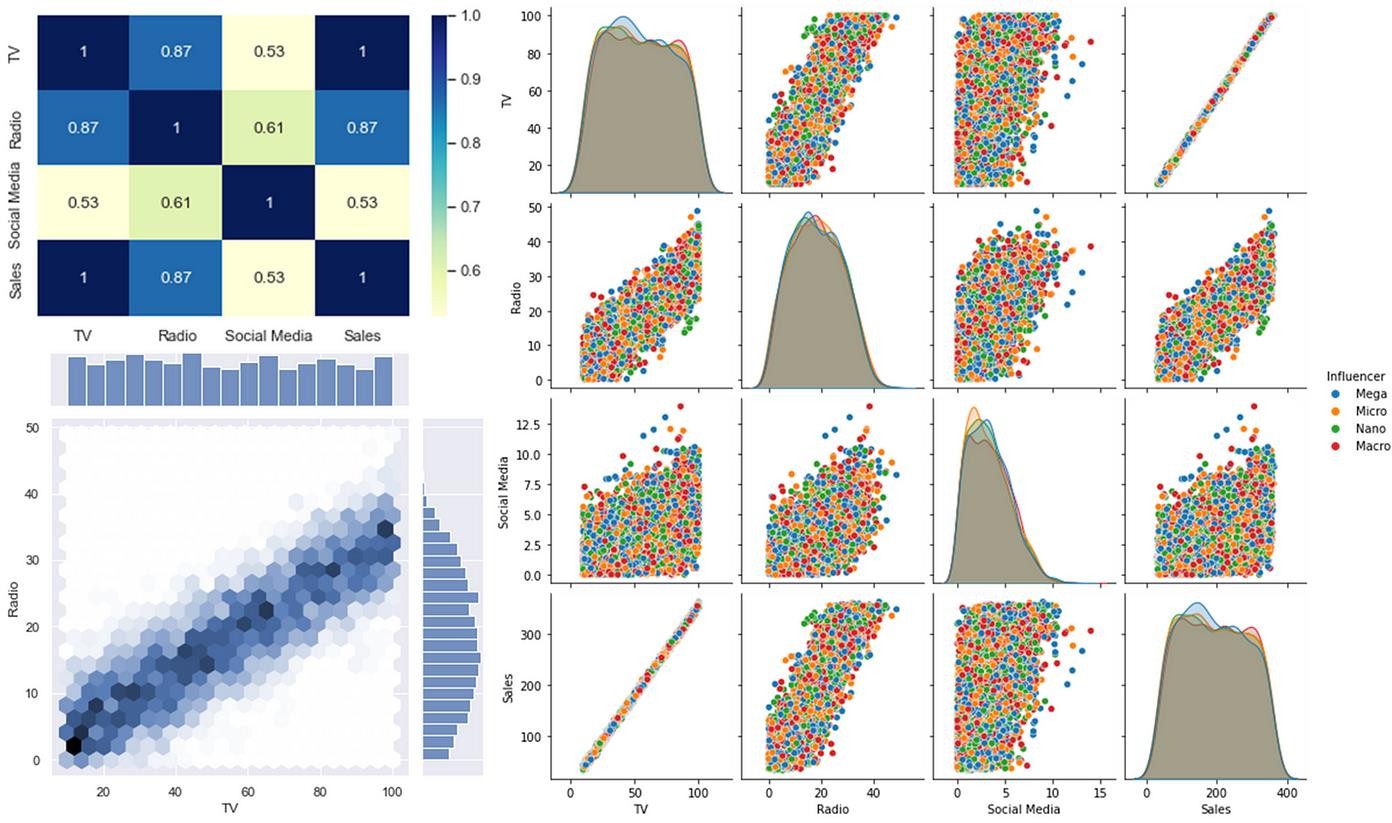
Heatmaps: Display matrix-like data with color-coding, often used for correlation matrices.

* Themes and Color Palettes:

Themes: Built-in themes to improve the aesthetics of plots (e.g., darkgrid, whitegrid, dark, white, and ticks).

Color Palettes: Predefined color palettes and the ability to create custom color schemes for visualizations.





# Jupyter Notebook

Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. It's widely used in data science, machine learning, scientific computing, and academic research due to its versatility and ease of use.

# Key Features:

1. **Interactive Code Execution:**
   1. You can write and run code in various programming languages, such as Python, R, and Julia. The most common use is with Python, leveraging powerful libraries like NumPy, pandas, and matplotlib.
   2. Code cells allow you to run code interactively, with the output displayed directly below the cell.

# Rich Text:

* 1. You can use Markdown to write narrative text, including formatted text, images, links, and LaTeX for mathematical equations. This is useful for documenting your code, explaining concepts, and presenting results.

# Data Visualization:

* 1. Jupyter Notebooks support various libraries for data visualization, such as matplotlib, seaborn, and Plotly. You can create graphs, plots, and interactive visualizations within the notebook.

# Starting Jupyter Notebook:

1. **Installation:**
   * Install Jupyter Notebook via Anaconda or pip (pip install notebook).

# Running a Notebook:

* + Launch Jupyter Notebook from the terminal with the command jupyter notebook.
  + The browser will open with the Jupyter interface, where you can create new notebooks or open existing ones.

# Basic Navigation:

* + **Cells:** Cells are the building blocks of a notebook. They can contain code or text
  + **Kernel:** The kernel executes the code contained in the notebook. You can start, stop, and restart the kernel as needed.