Chapter 1

Introduction to Machine Learning [5 hours]

1. Definition and Evolution of Machine Learning

Machine Learning:

We are drowning in information and starving for knowledge. — John Naisbitt

We are entering the era of big data. For example, there are about 1 trillion web pages; one hour of video is uploaded to YouTube every second, amounting to 10 years of content every day; the genomes of 1000s of people, each of which has a length of 3.8×10^9 base pairs, have been sequenced by various labs; Walmart handles more than 1M transactions per hour and has databases containing more than 2.5 petabytes (2.5 × 1015) of information; and so on.

It is more difficult to design algorithms for such tasks (compared to, say, sorting an array or calculating a payroll). Such algorithms need data.

Machine learning is the branch of AI, based on the concept that machines and systems can analyze and understand data, and learn from it and make decisions with minimal to zero human intervention.

Machine Learning (ML) is a branch of **Artificial Intelligence (AI)** that enables computers to **learn from data and improve their performance** on a task **without being explicitly programmed**.

For example: If you give a machine learning model thousands of pictures of cats and dogs (with labels), it can learn to **identify whether a new image is a cat or a dog**.

Most industries and businesses working with massive amounts of data have recognized the value of machine learning technology. By culling insights from this data, businesses are able to work more efficiently and gain an advantage over others.

Machine learning is an umbrella term for a set of techniques and tools that help computers learn and adapt on their own.

Machine Learning Vs Classic Programming:

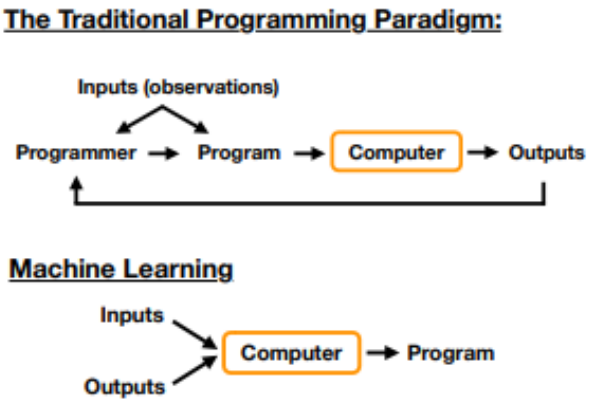

Figure: Machin eLearning vs Classic Programming

Features and Labels:

In machine learning, **features** and **labels** are fundamental concepts, especially in supervised learning.

- **Features** are the input variables that provide information to the model. They are measurable characteristics or attributes of the data used to make predictions. Features can be numeric, categorical, or even text-based, depending on the data and the machine learning algorithm being used.
- **Labels** are the output variables that the model aims to predict. They represent the desired outcomes or predictions we want to make. In supervised learning, labels are provided as part of the training data.

Here's a table illustrating the concept of **features** and **labels** in the context of a supervised learning problem.

Feature 1: Age	Feature 2: Gender	Feature 3: Salary	Label: Loan Approval
25	Male	$50,000	Approved
40	Female	$30,000	Denied
35	Male	$70,000	Approved
28	Female	$45,000	Denied

- **Features:**
    - **Age, Gender, and Salary** are the input variables (or predictors) that the model uses to make predictions.
    - These features describe each applicant and provide information relevant to predicting whether their loan will be approved or denied.
- **Label:**
    - **Loan Approval** is the output variable or target. It indicates the outcome the model is trying to predict (e.g., whether the loan application is "Approved" or "Denied").
    - Labels are only available in the training dataset for supervised learning tasks.

Evolution of Machine Learning:

The evolution of **Machine Learning (ML)** is closely tied to the growth of computer science, statistics, and data availability. Here's a **brief historical timeline** showing how ML has developed over the years:

1. Pre-1950s: Foundations

- **Mathematical groundwork** laid by **Alan Turing**, **Bayes**, and **statistical inference**.
- Turing's 1950 paper *"Computing Machinery and Intelligence"* introduced the **Turing Test**, asking *"Can machines think?"*

2. 1950s–1970s: Early AI and ML

Year	Milestone
1952	Arthur Samuel created the first **self-learning program** (checkers game).
1957	Frank Rosenblatt developed the **Perceptron** (early neural network model).
1967	First **nearest neighbor algorithm** was used for pattern recognition.

- Focused on **symbolic AI**, **rule-based systems**, and **simple learning algorithms**.

3. 1980s: Rise of Algorithms

Highlight	Description
Backpropagation algorithm	Enabled training of multi-layer neural networks (revived interest in neural nets).

Highlight	Description
Decision Trees, SVMs	Emerged as powerful supervised learning models.

- Shift from logic-based AI to **statistical learning**.
- Increase in computational power and data helped algorithm development.

4. 1990s: Statistical Learning Era

- **Machine Learning split from AI** as a distinct field.
- Strong use of **probability theory** and **statistics** (Bayesian networks, HMMs).
- **Support Vector Machines (SVM)** and **Ensemble Methods** (e.g., Bagging, Boosting) gained popularity.

5. 2000s: Big Data & Real-World Applications

- Explosion of **data availability** due to the internet.
- Rise of **open-source ML libraries** (e.g., Weka, Scikit-learn).
- Applied in **search engines, bioinformatics, recommender systems**.

6. 2010s: Deep Learning Revolution

Year	Milestone
2012	**AlexNet** wins ImageNet competition → Popularized deep learning.
2014	**GANs (Generative Adversarial Networks)** introduced by Ian Goodfellow.

- Rise of **deep neural networks**, **convolutional nets**, **recurrent nets**, **transfer learning**.
- Used in **speech recognition, image processing, NLP (BERT, GPT)**.

7. 2020s–Present: AI Everywhere

- **Large Language Models (LLMs)** (like GPT, BERT, Claude).
- ML used in **self-driving cars, healthcare, finance, edge devices**.
- Emphasis on **Explainable AI**, **Ethical AI**, **TinyML**, **AutoML**.

2. Types of Machine Learning

a) Supervised Learning

Supervised learning is the subcategory of machine learning that focuses on learning a classification, or regression model, that is, learning from labeled training data (i.e., inputs that also contain the desired outputs or targets; basically, "examples" of what we want to predict). The model learns from a **labeled dataset**, where each input has a corresponding correct output. Supervised models are trained using labeled datasets, where input-output pairs are explicitly provided.

Goal: Learn a mapping from input (X) to output (Y).

Example: Consider the following data regarding patients entering a clinic. The data consists of the gender and age of the patients and each patient is labeled as "healthy" or "sick".

gender	age	label
M	48	sick
M	67	sick
F	53	healthy
M	49	healthy
F	34	sick
M	21	healthy

Figure: A Labeled dataset

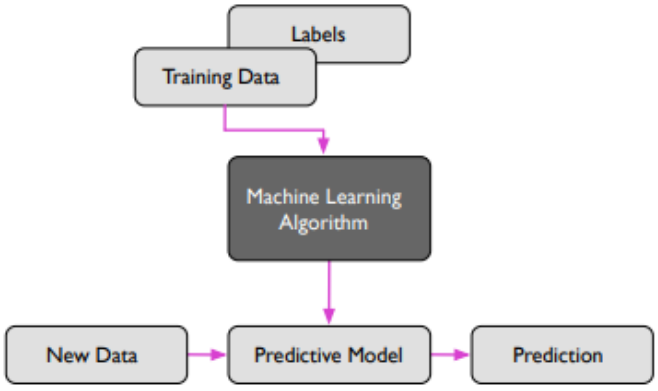

Figure: Rough overview of the supervised learning process

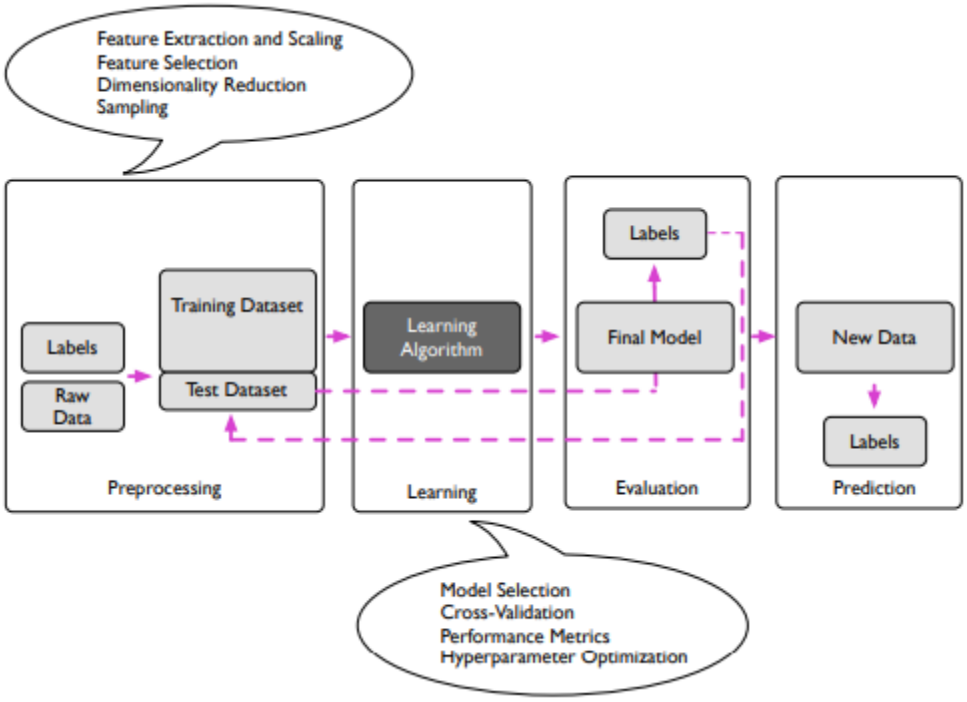

Figure: More detailed illustration of the supervised learning process.

In supervised learning, **we are given a labeled training dataset** from which a machine learning algorithm can learn a model. The learned (or trained) **model can be used to predict labels of unlabeled data points.** These unlabeled data points could be either test data points (for which we actually have labels but we withheld them for testing purposes) or unlabeled data that we already collected or will collect in the future. For example, given a corpus of spam and non-spam email, a supervised learning task would be to learn a model that predicts to which class (spam or non-spam) new emails belong.

 Examples:

- Email spam detection
- Disease prediction
- House price prediction

 Common Algorithms:

- Linear Regression
- Decision Trees
- Support Vector Machines (SVM)
- Neural Networks

Types of supervised Learning:

1. **Classification:** The task of assigning inputs into discrete categories. Trains models on labeled data to predict or classify new, unseen data.

Example:

Predicting whether an email is spam or not.

Feature 1: Email Content	Feature 2: Sender Address	Label: Spam/Not Spam
Contains "Win a Prize"	unknown@spam.com	Spam
Contains "Meeting Update"	colleague@work.com	Not Spam

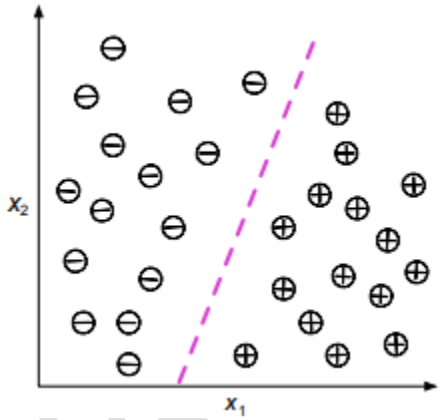

Figure: Illustration of a binary classification problem (plus and minus signs denote class labels) and two feature variables, (x1 and x2).

2. **Regression:** Predicting continuous numerical outputs

Example:

Predicting house prices.

Feature 1: Size (sq ft)	Feature 2: Bedrooms	Feature 3: Location	Label: Price ($)
1500	3	Suburban	300,000
2000	4	Urban	450,000

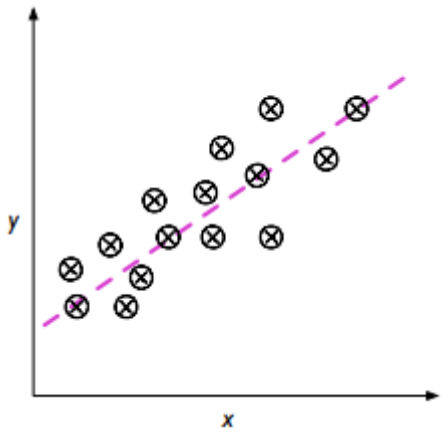

Figure: Illustration of a linear regression model with one feature variable (x) and the target variable (y).

b) Unsupervised Learning

In contrast to supervised learning, unsupervised learning is a branch of machine learning that is **concerned with unlabeled data**. Common tasks in unsupervised learning are clustering analysis (assigning group memberships) and dimensionality reduction (compressing data onto a lower-dimensional subspace or manifold).

The model is given **unlabeled data** and must find **hidden patterns or groupings**.

Goal: Discover structure in the data (clustering or association).

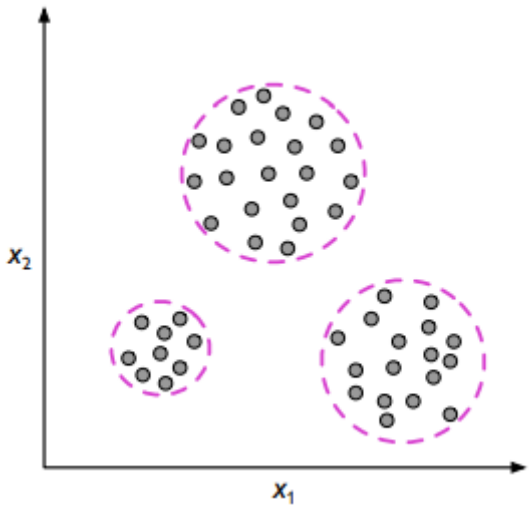

Figure: Illustration of clustering, where the dashed lines indicate potential group membership assignments of unlabeled data points.

Example: Consider the following data regarding patients entering a clinic. The data consists of the gender and age of the patients. Based on this data, can we infer anything regarding the patients entering the clinic?

gender	age
M	48
M	67
F	53
M	49
F	34
M	21

Figure: An Unlabeled data

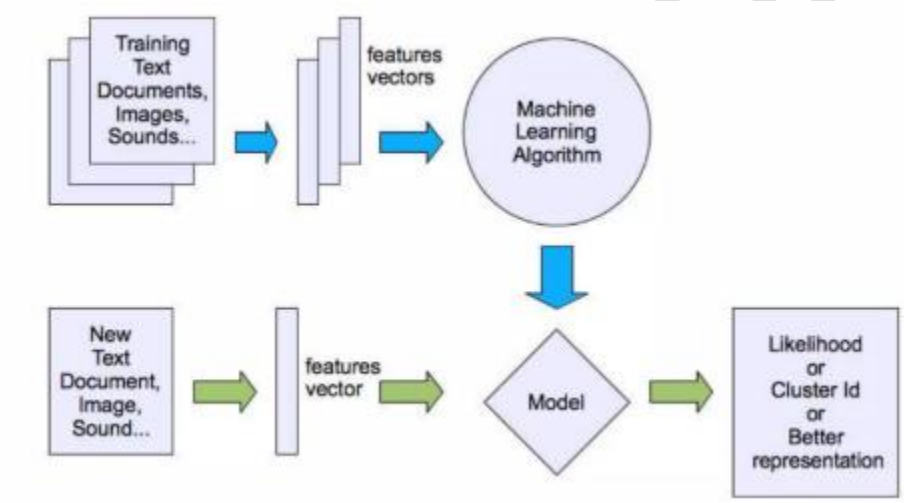

Figure: Unsupervised Model

- **Examples:**

  - Customer segmentation
  - Market basket analysis
  - Dimensionality reduction

- **Common Algorithms:**

  - K-Means Clustering
  - Hierarchical Clustering
  - Principal Component Analysis (PCA)
  - Apriori Algorithm

c) Reinforcement Learning

An agent learns by interacting with an **environment**, receiving **rewards** or **penalties** for its actions. **Goal:** Learn a policy to maximize cumulative reward over time.

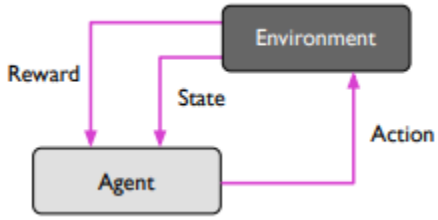

Figure: Illustration of reinforcement learning

Reinforcement is the **process of learning from rewards while performing a series of actions**. In reinforcement learning, we do not tell the learner or agent (for example, a robot), which action to take but merely assign a reward to each action and/or the overall outcome. Instead of having "correct/false" labels for each step, the learner must discover or learn a behavior that maximizes the reward for a series of actions.

Example Consider teaching a dog a new trick: we cannot tell it what to do, but we can reward/punish it if it does the right/wrong thing. It has to find out what it did that made it get the reward/punishment.

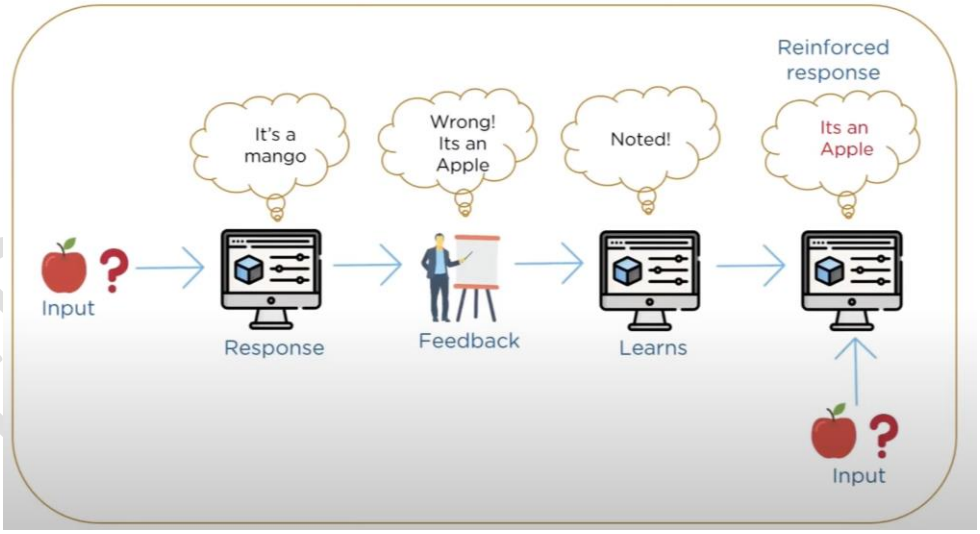

Figure: Reinforcement Learning Process

We can use a similar method to train computers to do many tasks, such as playing backgammon or chess, scheduling jobs, and controlling robot limbs.

 Examples:

- Game playing (Chess, Go, Atari)
- Robotics and control systems
- Self-driving cars

 Key Concepts:

- Agent, Environment
- Actions, Rewards
- States, Policy, Value Function

 Common Algorithms:

- Q-Learning
- Deep Q-Networks (DQN)
- Policy Gradient Methods

d) Active Learning

Active Learning is a special kind of **Supervised Machine Learning** where the model is **allowed to ask questions** — specifically, it **selects which data points should be labeled** next, in order to learn more efficiently.

In other words, instead of labeling a huge dataset blindly, the model **actively chooses** the **most informative samples** for labeling, which saves time and cost.

Why Use Active Learning?

- Labeling data (like medical images or legal documents) can be **expensive** or require **expert effort**.
- Active Learning helps you build a **high-performance model** with **fewer labeled examples**.

How It Works (Basic Workflow)

1. **Start with a small labeled dataset** and a large pool of unlabeled data.
2. Train a model on the labeled data.
3. The model evaluates which **unlabeled examples** it is **least certain** about.
4. It **asks a human expert** (oracle) to label those examples.
5. Add the new labels to the training set and **repeat**.

Example Scenario

You're building a spam filter:

- Instead of labeling 10,000 emails, start with 500.
- Train your model.
- Let the model **choose the 100 emails it's most confused about**.
- Get labels for those and retrain.
- Repeat until the model is accurate enough!

e) Semi-supervised Learning

A **semi-supervised model** is a type of machine learning model that is trained using a combination of **labeled** and **unlabeled** data.

Imagine you want to build an image classifier for cats and dogs:

- You have 100 labeled images (50 cats, 50 dogs).
- You also have 10,000 unlabeled images (just raw images without labels).

A **semi-supervised model** would use both:

- The labeled data to learn initial distinctions.
- The unlabeled data to **discover the data structure or clusters**, and refine the decision boundaries.

When to Use:

- Labeled data is scarce or expensive.
- You have access to a large pool of unlabeled data.

f) Deep Learning

Deep learning is a subset of machine learning using artificial neural networks to model complex patterns.

Example:

- **Convolutional Neural Networks (CNNs):** Used for image recognition tasks like identifying cats in pictures.

- **Recurrent Neural Networks (RNNs):** Used for sequential data like predicting the next word in a sentence.
- **Generative Adversarial Networks (GANs):** Used for generating realistic images.
- **Autoencoders:** Used for dimensionality reduction and feature learning.
- **Transformer Models:** Used for sequence-to-sequence tasks like machine translation.
- **Artificial Neural Networks (ANNs):** Used for classification and regression tasks.

3. Machine Learning Workflow

Figure: Machine Learning Workflow

1. Problem Definition

This is the **foundation of any ML project**. You need to clearly understand:

- **What is the objective?**
- **What kind of problem is it?**
  - **Classification** (e.g., spam detection)
  - **Regression** (e.g., predicting house prices)
  - **Clustering** (e.g., grouping customers)

Example: If you're building a model to **predict student grades**, you're solving a **regression** problem.

2. Data Collection and Preprocessing

Data Collection

- Gather relevant data from:
  - Databases, sensors, web scraping, APIs, surveys, etc.
- Ensure the data is **relevant, recent, and accurate**.

Data Preprocessing

- Goal: Convert raw data into a clean and usable format for the model.
- Clean the data:
  - Remove **missing**, **incomplete**, or **duplicate** entries.
- Transform the data:
  - **Normalize** numeric features.
  - **Encode** categorical variables (e.g., gender: male → 0, female → 1).
- Feature Engineering:
  - Create meaningful **new features** from existing data.

3. Model Selection

Choose a suitable algorithm based on:

- Problem type (classification, regression, etc.)
- Size and type of dataset
- Accuracy vs. interpretability trade-offs

Common Models:

Task	Model Examples
Classification	Logistic Regression, SVM, Decision Trees, Random Forest
Regression	Linear Regression, Decision Trees, Gradient Boosting
Clustering	K-Means, Hierarchical Clustering

4. Model Evaluation and Validation

After training, test the model's performance on **unseen data**. Goal: Find the **best-performing model** that generalizes well.

Techniques:

- **Train/Validation/Test Split** or **Cross-Validation**
- Evaluation Metrics:
  - Classification: Accuracy, Precision, Recall, F1 Score, ROC-AUC
  - Regression: MSE, RMSE, MAE, R^2

Hyperparameter Tuning:

- Adjust model settings (like depth of trees, learning rate) using:
  - Grid Search
  - Random Search
  - Bayesian Optimization

5. Model Deployment

Once validated, the model is **deployed into production** so it can make predictions on new, real-time data. Deployment turns your ML solution into a real-world application.

Deployment Options:

- REST API (Flask, FastAPI)
- Cloud platforms (AWS, GCP, Azure)
- Mobile apps, Web apps, Dashboards

Post-deployment:

- **Monitor** the model's performance
- **Retrain** periodically to handle **data drift**

4. Challenges in Machine Learning

1. Data Quality Issues

"Garbage in, garbage out." Machine Learning heavily relies on the quality of data. Poor data leads to poor models. **Impact**: Leads to unreliable models and poor generalization.

Common Data Issues:

- **Missing values**: Incomplete entries reduce model accuracy.
- **Noisy data**: Irrelevant or random errors can confuse learning.
- **Imbalanced datasets**: One class dominates others (e.g., 95% non-spam, 5% spam).
- **Bias in data**: Skewed or unfair representations can lead to biased predictions.

2. Computational Complexity

Machine Learning often involves **large datasets** and **complex algorithms**. **Impact**: Makes ML expensive, time-consuming, and difficult to scale.

Challenges:

- **Training Time**: Deep learning models can take hours/days to train.
- **Hardware Requirements**: High-performance GPUs or cloud computing often needed.
- **Big Data Bottlenecks**: Memory limitations and slow I/O operations.

3. Interpretability and Explainability

Many ML models, especially deep learning, are **black boxes**—they give predictions, but it's hard to explain **why**.

Impact: Without explainability, it's hard to trust or adopt ML in critical domains.

Why This Matters:

- In **healthcare**, **finance**, or **law**, decisions must be **transparent and justifiable**.
- Helps **debug and improve** the model.

Solutions:

- Use simpler models (e.g., decision trees) when possible.
- Use **explainable AI (XAI)** tools like:
    - SHAP (SHapley Additive exPlanations)
    - LIME (Local Interpretable Model-agnostic Explanations)

4. Ethical Considerations

ML can unintentionally cause **harm or unfairness** if not designed carefully.

Impact: Ethical failures can lead to legal consequences, loss of public trust, or harm to individuals and society.

Key Ethical Challenges:

- **Bias and Fairness**: Biased training data can lead to discriminatory outcomes.
- **Privacy Concerns**: Using personal data without consent violates privacy laws (e.g., GDPR).
- **Job Displacement**: Automation may lead to unemployment in certain sectors.
- **Misuse**: Models can be exploited (e.g., deepfakes, surveillance, fake news).