# UNIT-1- INTRODUCTION TO MACHINE LEARNING

## MACHINE LEARNING

Machine learning (ML) is defined as a discipline of artificial intelligence (AI) that provides machines the ability to automatically learn from data and past experiences to identify patterns and make predictions with minimal human intervention.

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy. IBM has a rich history with machine learning.

### Father of Machine Learning

The term machine learning was coined in 1959 by Arthur Samuel, an IBM employee and pioneer in the field of computer gaming and artificial intelligence. The synonym self-teaching computers was also used in this time period.

### History of Machine Learning

1940s-1950s: Birth of AI; Alan Turing's work lays foundations for machine learning.

1950s-1960s: Introduction of the perceptron and early neural networks, but limited success.

1960s-1970s: Focus on rule-based systems and expert systems.

1980s-1990s: Rediscovery of back propagation sparks interest in neural networks; SVMs and ensemble learning gain popularity.

2000s-2010s: Big data and increased computing power lead to the resurgence of deep learning; applications in image, speech, and natural language processing.

2010s-Present: Widening adoption of machine learning across industries; development of open-source frameworks like TensorFlow and PyTorch.

2020s: On-going advancements in reinforcement learning, generative models, and ethical considerations; machine learning's increasing impact on various sectors.

## TYPES OF MACHINE LEARNING

Machine learning can be broadly categorized into three main types based on the learning style and approach:

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning

### Supervised learning

Supervised learning is a type of machine learning where the algorithm is trained on a labeled dataset, meaning that each input in the training data is associated with a corresponding output. The goal is for the algorithm to learn a mapping from inputs to outputs, allowing it to make predictions on new, unseen data. Here are key characteristics and components of supervised learning:

### Labelled Training Data:

In supervised learning, the training dataset consists of pairs of input-output examples. The input represents the features or attributes of the data, and the output is the corresponding label or target value that the algorithm aims to predict.

Objective:

The primary objective of supervised learning is to learn a function or mapping that can accurately predict the output for new, unseen inputs. This involves capturing the underlying patterns and relationships within the data during the training phase.

Types of Supervised Learning Tasks:

Classification: The algorithm predicts the category or class label of the input. Examples include spam detection, image recognition, and sentiment analysis.

Regression: The algorithm predicts a continuous output or a numerical value. Examples include predicting house prices, stock prices, or temperature.

Model Evaluation:

The performance of a supervised learning model is evaluated based on its ability to make accurate predictions on new, unseen data. Common evaluation metrics include accuracy, precision, recall, F1 score (for classification), and mean squared error (for regression).

Common Algorithms:

Supervised learning encompasses a variety of algorithms, including:

Linear Regression: For regression tasks where the relationship between variables is assumed to be linear.

Logistic Regression: For binary classification problems.

Support Vector Machines (SVM): Effective for both classification and regression tasks.

Decision Trees and Random Forests: Versatile for various tasks, including classification and regression.

Challenges:

Challenges in supervised learning include the need for labeled data, potential bias in the training data, and overfitting, where the model performs well on the training data but poorly on new data.

Applications:

Supervised learning finds applications in a wide range of fields, such as healthcare (diagnosis of diseases), finance (credit scoring), natural language processing (language translation), and image recognition (object detection).

Supervised learning is a foundational approach in machine learning, providing a framework for training models to make predictions based on labeled examples, making it a valuable tool in various real-world applications.

**Unsupervised learning**

Unsupervised learning is a category of machine learning where the algorithm is trained on unlabeled data, and its objective is to discover patterns, structures, or relationships within the data without explicit guidance or labeled outputs. The algorithm explores the inherent structure of the data to uncover hidden insights. Key aspects of unsupervised learning include:

Unlabeled Data:

In unsupervised learning, the training dataset consists of input data without corresponding output labels. The algorithm explores the data to identify inherent patterns or groupings.

Objective:

The primary goal of unsupervised learning is to reveal the underlying structure of the data. This can involve tasks such as clustering, where similar data points are grouped together, or dimensionality reduction, where the dataset is represented in a lower-dimensional space.

Types of Unsupervised Learning Tasks:

Clustering: Grouping similar data points together based on certain criteria. Examples include k-means clustering and hierarchical clustering.

Dimensionality Reduction: Reducing the number of features or variables in the data while preserving its essential characteristics. Principal Component Analysis (PCA) is a common technique for dimensionality reduction.

Density Estimation: Estimating the probability density function of the data. Gaussian Mixture Models (GMMs) are an example of a density estimation method.

Model Evaluation:

Evaluating the performance of unsupervised learning models is often more subjective than in supervised learning. Metrics may depend on the specific task, such as silhouette score for clustering or explained variance for dimensionality reduction.

Common Algorithms:

Unsupervised learning algorithms include:

K-Means: A popular clustering algorithm that partitions data into k clusters.

Hierarchical Clustering: Builds a hierarchy of clusters by either bottom-up or top-down approaches.

PCA (Principal Component Analysis): Reduces the dimensionality of the data by transforming it into a set of orthogonal components.

Challenges:

Challenges in unsupervised learning include determining the appropriate number of clusters, dealing with high-dimensional data, and interpreting the discovered patterns or structures.

Applications:

Unsupervised learning has applications in various domains, including anomaly detection, customer segmentation, pattern recognition, and exploratory data analysis.

Unsupervised learning techniques are valuable for uncovering insights from data when explicit labels or outputs are unavailable, allowing for a deeper understanding of the inherent structure and patterns within the dataset.

**Reinforcement learning**

Reinforcement Learning (RL) is a type of machine learning where an agent interacts with an environment and learns to make decisions by receiving feedback in the form of rewards or penalties. The agent's goal is to learn a policy or strategy that maximizes the cumulative reward over time. Key components and characteristics of reinforcement learning include:

Agent and Environment:

In RL, there is an agent that makes decisions and interacts with an environment. The environment provides feedback to the agent in the form of rewards or punishments based on the actions taken by the agent.

Actions, States, and Rewards:

The agent takes actions within the environment, transitioning between different states. Each action leads to a change in the state, and the agent receives a reward or penalty based on the consequences of its actions.

Objective:

The primary objective of reinforcement learning is for the agent to learn a policy—a mapping from states to actions—that maximizes the cumulative reward over time. The agent explores different actions and learns from the feedback received.

Exploration and Exploitation:

RL involves a trade-off between exploration (trying new actions to discover their effects) and exploitation (choosing actions that are known to yield higher rewards based on past experiences).

Markov Decision Process (MDP):

Reinforcement learning problems are often formulated as Markov Decision Processes, where the environment is assumed to have the Markov property—future states depend only on the current state and action, not on the sequence of events that led to the current state.

Learning Algorithms:

RL algorithms include model-free methods (e.g., Q-learning and Monte Carlo methods) and model-based methods (e.g., Dynamic Programming and Temporal Difference learning). Deep Reinforcement Learning (DRL) integrates neural networks to handle high-dimensional state spaces.

Applications:

Reinforcement learning is applied in various domains, such as robotics, gaming (e.g., AlphaGo), autonomous systems (e.g., self-driving cars), and resource management (e.g., optimizing energy consumption).

**APPLICATIONS OF MACHINE LEARNING**

- Healthcare: Disease diagnosis, personalized treatment, drug discovery, medical image analysis.
- Finance: Credit scoring, fraud detection, algorithmic trading, customer segmentation.
- E-commerce: Product recommendations, demand forecasting, price optimization, customer churn prediction.
- Marketing: Sentiment analysis, ad targeting, click-through rate prediction, social media monitoring.
- Manufacturing: Predictive maintenance, quality control, supply chain optimization, energy consumption.
- Autonomous Vehicles: Self-driving cars, traffic prediction, route optimization, vehicle health monitoring.

- Natural Language Processing: Speech recognition, language translation, sentiment analysis, chatbots.
- Computer Vision: Object detection, image classification, facial recognition, video analysis.
- Environmental Science: Climate modeling, species identification, deforestation detection, pollution monitoring.
- Entertainment: Content recommendation, player behavior analysis, game AI, virtual reality experiences.

## MACHINE LEARNING PROCESS

Machine learning has given computers the ability to learn on their own without having to be explicitly programmed.
A machine learning project's life cycle is a cyclic method for developing an effective machine learning project. The life cycle's primary goal is to find a solution to the problem or project.

**1. Data Gathering:** As the name suggests in this step we gather all the data-related problems.
The steps involve Identifying various data sources, collecting data, and integrating the data obtained from different sources.

**2. Preparation of Data:** We must prepare the data for further processing after it has been collected. Data preparation entails putting our data in an appropriate location and preparing it for use in machine learning training.
In this stage, we combine all of the data and then randomize the order of the data.

This stage may be separated into two parts: data exploration, in which we learn about the data, and data pre-processing, in which the data is prepared for analysis.

**3. Data Wrangling:** It is the process of cleaning the data, selecting the variable to utilize, and changing the data into a suitable format for analysis in the following phase.

**4. Data Analysis:** The data has now been cleaned and prepped and is ready to be analyzed. This stage entails Analytical methods selection, creating models, examine the outcome.

**5. Train Model:** Datasets are used to train the model, which is then used to train the model using various machine learning techniques.
A model must be trained in order for it to comprehend the numerous patterns, rules, and characteristics.

**6. Test Model:** The % correctness of the model is determined by testing it against the project or problem's requirements.

**7. Deployment model:** We deploy the model in the real system if the above-prepared model produces an accurate output that meets our requirements at a reasonable pace.

**WELL POSED LEARNING PROBLEM:**

The formal definition of Well posed learning problem is, "A computer program is said to learn from Experience E when given a task T, and some performance measure P. If it performs on T with a performance measure P, then it upgrades with experience E.
To break it down, the three important components of a well-posed learning problem are,

- Task
- Performance Measure
- Experience

To understand the topic better let's have a look at a few classical examples,

- **Learning to play Checkers:**
  A computer might improve its performance as an ability to win at the class of tasks that are about playing checkers. The performance keeps improving through experience by playing against itself.
  To simplify,
  T -> Play the checkers game.
  P -> Percentage of games won against the opponent.
  E -> Playing practice games against itself.

- **Handwriting Recognition:**
  Handwriting recognition (HWR) is a technology that converts a user's handwritten letters or words into a computer-readable format (e.g., Unicode text).
  Its applications are numerous, it is used in reading postal addresses, bank forms, etc.
  T -> recognizing and classifying handwritten words from images.
  P -> Percentage of correctly identified words.
  E -> set of handwritten words with their classifications in a database.

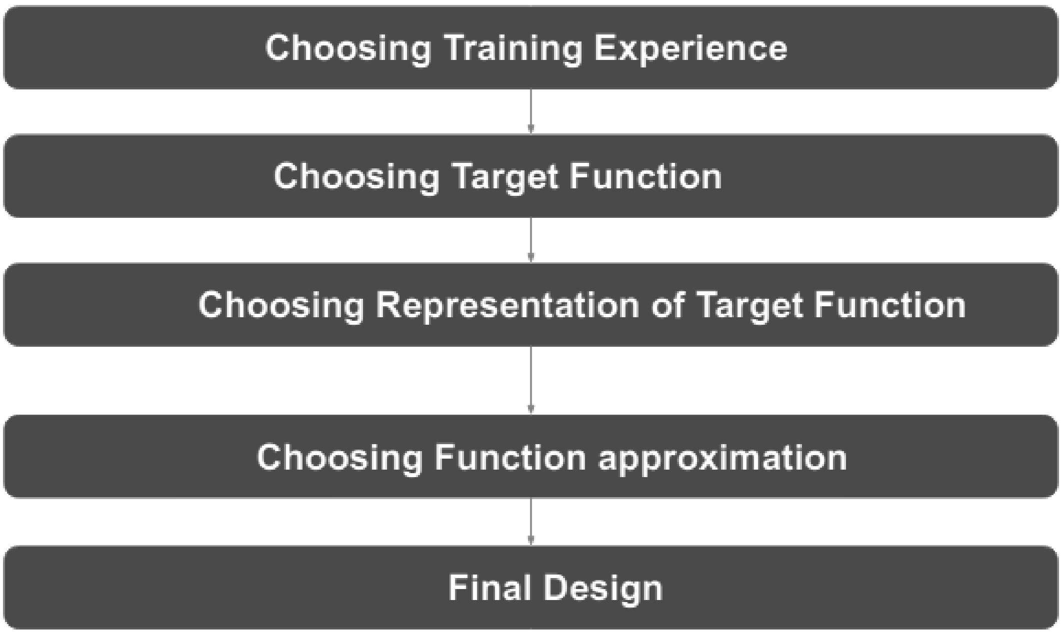
- **A Robot Driving Learning Problem:**
  For a robot to drive on a four-lane highway it needs a human-like understanding of all the possibilities it might encounter.

With the use of sight scanners and advanced machine learning algorithms, it can be made possible.
T –> To drive on public four-lane highways using sight scanners.
P -> the average distance progressed before an error.
E -> the order of images and steering instructions noted down while observing a human driver.

- **A spam filtering for emails learning problem:**
  A spam filter is software that detects unsolicited and undesired email and prevents it from reaching the inbox of a user.
  T -> Identifying whether or not an email is spam.
  P -> The percentage of emails correctly categorized as spam or non-spam.
  E -> Observing how you categorize emails as spam or non-spam.

- **Face Recognition Problem:**
  A facial recognition system device is capable of matching a human face from a digital image or a video frame against a database of faces.
  It works by locating and measuring facial characteristics from a given image and is often used to verify users through ID verification services.
  T -> Predicting distinct sorts of faces.
  P -> Ability to anticipate the largest number of different sorts of faces.
  E -> train the system with as many datasets of varied facial photos as possible.

**DESIGNING A LEARNING SYSTEM**




Step 1:

Choosing the Training Experience: The very important and first task is to choose the training data or training experience which will be fed to the Machine Learning Algorithm. It is important to note that the data or experience that we fed to the algorithm must have a significant impact on the Success or Failure of the Model. So Training data or experience should be chosen wisely. Below are the attributes which will impact on Success and Failure of Data:

The training experience will be able to provide direct or indirect feedback regarding choices. For example: While Playing chess the training data will provide feedback to itself like instead of this move if this is chosen the chances of success increases.
Second important attribute is the degree to which the learner will control the sequences of training examples. For example: when training data is fed to the machine then at that time accuracy is very less but when it gains experience while playing again and again with itself or opponent the machine algorithm will get feedback and control the chess game accordingly.

Third important attribute is how it will represent the distribution of examples over which performance will be measured. For example, a Machine learning algorithm will get experience while going through a number of different cases and different examples. Thus, Machine Learning Algorithm will get more and more experience by passing through more and more examples and hence its performance will increase.

Step 2:

Choosing target function: The next important step is choosing the target function. It means according to the knowledge fed to the algorithm the machine learning will choose NextMove function which will describe what type of legal moves should be taken. For example : While playing chess with the opponent, when opponent will play then the machine learning algorithm will decide what be the number of possible legal moves taken in order to get success.

Step 3:

Choosing Representation for Target function: When the machine algorithm will know all the possible legal moves the next step is to choose the optimized move using any representation i.e. using linear Equations, Hierarchical Graph Representation, Tabular form etc. The NextMove function will move the Target move like out of these move which will provide more success rate. For Example : while playing chess machine have 4 possible moves, so the machine will choose that optimized move which will provide success to it.

Step 4:

Choosing Function Approximation Algorithm: An optimized move cannot be chosen just with the training data. The training data had to go through with set of example and through these examples the training data will approximates which steps are chosen and after that machine will provide feedback on it. For Example : When a training data of Playing chess is fed  to algorithm so at that time it is not machine algorithm will fail or get success and again from that failure or success it will measure while next move what step should be chosen and what is its success rate.

Step 5:

Final Design: The final design is created at last when system goes from number of examples , failures and success , correct and incorrect decision and what will be the next step etc. Example: DeepBlue is an Intelligent  computer which is ML-based won chess game against the chess expert Garry Kasparov, and it became the first computer which had beaten a human chess expert.

## PERSPECTIVES AND ISSUES IN MACHINE LEARNING

### Perspectives in Machine Learning

Involves searching very large hypothesis to determine one that best fits the observed data and any prior knowledge held by.

### Issues in Machine Learning

Inadequate Training Data:

Insufficient quantity or poor quality of data can hinder the performance of machine learning algorithms, leading to inaccurate predictions and unreliable results.

Poor Quality of Data:

Data that is noisy, incomplete, inaccurate, or unclean can result in lower accuracy and quality of machine learning models.

Non-representative Training Data:

Training data that does not adequately represent the target population can lead to biased models and inaccurate predictions, impacting the generalization ability of the model.

Overfitting and Underfitting:

Overfitting occurs when a model captures noise in the training data, while underfitting occurs when a model is too simplistic to capture the underlying patterns. Both issues can degrade the performance of machine learning models.

Monitoring and Maintenance:

Continuous monitoring and maintenance of machine learning models are necessary to ensure their effectiveness over time and to detect issues such as concept drift or degradation in performance.

Getting Bad Recommendations:

Machine learning models may provide inaccurate or outdated recommendations if they are not updated or monitored regularly, leading to poor user experiences and decreased effectiveness.

Lack of Skilled Resources:

The shortage of skilled professionals with expertise in machine learning and data science poses challenges for organizations seeking to develop and deploy machine learning solutions.

Customer Segmentation:

Identifying relevant customer segments and delivering personalized recommendations requires sophisticated algorithms and accurate data, which can be challenging to achieve.

Process Complexity of Machine Learning:

The complexity of the machine learning process, including data analysis, model training, and evaluation, can make it challenging to develop and deploy machine learning solutions effectively.

Data Bias:

Bias in training data can lead to skewed outcomes and inaccurate predictions, affecting the fairness and reliability of machine learning models.

Lack of Explainability:

Machine learning models that lack explainability may be difficult to interpret and trust, limiting their usability in certain applications.

Slow Implementations and Results:

Slow execution and processing times can impact the practicality and efficiency of machine learning models, especially in real-time or resource-constrained environments.

Irrelevant Features:

Including irrelevant features in the training data can introduce noise and complexity into machine learning models, reducing their effectiveness and efficiency.