machine learning core 3 types

- **supervised learning:** Trains models on labeled data to predict or classify new, unseen data.
- **unsupervised learning:** Finds patterns or groups in unlabeled data, like clustering or `dimensionality` reduction.
- **reinforcement learning:** Learns through trial and error to maximize rewards, ideal for decision-making tasks.

Module 1: Machine Learning Pipeline

In order to make predictions there are some steps through which data passes in order to produce a machine learning model that can make predictions.

1. ML workflow
2. Data Cleaning
3. Feature Scaling
4. Data Preprocessing in Python

Module 2: Supervised Learning

Supervised learning algorithms are generally categorized into ***two main types:****

- ****Classification**** - where the goal is to predict discrete labels or categories
- ****Regression**** - where the aim is to predict continuous numerical values.

1. Linear Regression

This is one of the simplest ways to predict numbers using a straight line. It helps find the relationship between input and output.

- Introduction to Linear Regression
- Gradient Descent in Linear Regression
- Multiple Linear Regression
- Ridge Regression
- Lasso regression
- Elastic net Regression

2. Logistic Regression

Used when the output is a "yes or no" type answer. It helps in predicting categories like pass/fail or spam/not spam.

- Understanding Logistic Regression

- Cost function in Logistic Regression

3. **Decision Trees**

A model that makes decisions by asking a series of simple questions, like a flowchart. Easy to understand and use.

- Decision Tree in Machine Learning
- Types of Decision tree algorithms
- Decision Tree - Regression (Implementation)
- Decision tree - Classification (Implementation)

4. **Support Vector Machines (SVM)**

A bit more advanced—it tries to draw the best line (or boundary) to separate different categories of data.

- Understanding SVMs
- SVM Hyperparameter Tuning - GridSearchCV
- Non-Linear SVM

5. k-Nearest Neighbors (k-NN)

This model looks at the closest data points (neighbors) to make predictions. Super simple and based on similarity.

- Introduction to KNN
- Decision Boundaries in K-Nearest Neighbors (KNN)

6. Naïve Bayes

A quick and smart way to classify things based on probability. It works well for text and spam detection.

- Introduction to Naive Bayes
- Gaussian Naive Bayes
- Multinomial Naive Bayes
- Bernoulli Naive Bayes
- Complement Naive Bayes

7. Random Forest (Bagging Algorithm)

A powerful model that builds lots of decision trees and combines them for better accuracy and stability.

- Introduction to Random forest

- Random Forest Classifier
- Random Forest Regression
- Hyperparameter Tuning in Random Forest

Introduction to Ensemble Learning

Ensemble learning combines multiple simple models to create a stronger, smarter model. There are mainly two types of ensemble learning:

- **Bagging** that combines multiple models trained independently.
- **Boosting** that builds models sequentially each correcting the errors of the previous one.

Module 3: Unsupervised learning

Unsupervised learning are again divided into **three main categories** based on their purpose:

- Clustering
- Association Rule Mining
- Dimensionality Reduction.

1. Clustering

Clustering algorithms group data points into clusters based on their similarities or differences. Types of clustering algorithms are:

Centroid-based Methods:

- K-Means clustering
- Elbow Method for optimal value of k in KMeans
- K-Means++ clustering
- K-Mode clustering
- Fuzzy C-Means (FCM) Clustering

Distribution-based Methods:

- Gaussian mixture models
- Expectation-Maximization Algorithm
- Dirichlet process mixture models (DPMMs)

Connectivity based methods:

- Hierarchical clustering
- Agglomerative Clustering
- Divisive clustering

- Affinity propagation

Density Based methods:

- DBSCAN (Density-Based Spatial Clustering of Applications with Noise)
- OPTICS (Ordering Points To Identify the Clustering Structure)

2. Dimensionality Reduction

Dimensionality reduction is used to simplify datasets by reducing the number of features while retaining the most important information.

- Principal Component Analysis (PCA)
- t-distributed Stochastic Neighbor Embedding (t-SNE)
- Non-negative Matrix Factorization (NMF)
- Independent Component Analysis (ICA)
- Isomap
- Locally Linear Embedding (LLE)

3. Association Rule

Find patterns between items in large datasets typically in market basket analysis.

- Apriori algorithm
- Implementing apriori algorithm
- FP-Growth (Frequent Pattern-Growth)
- ECLAT (Equivalence Class Clustering and bottom-up Lattice Traversal)

Module 4: Reinforcement Learning

Reinforcement learning interacts with environment and learn from them based on rewards.

1. **Model-Based Methods**

These methods use a model of the environment to predict outcomes and help the agent plan actions by simulating potential results.

- Markov decision processes (MDPs)
- Bellman equation
- Value iteration algorithm
- Monte Carlo Tree Search

2. Model-Free Methods

The agent learns directly from experience by interacting with the environment and adjusting its actions based on feedback.

- Q-Learning
- SARSA
- Monte Carlo Methods
- Reinforce Algorithm
- Actor-Critic Algorithm
- Asynchronous Advantage Actor-Critic (A3C)

Module 5: Semi Supervised Learning

It uses a mix of labeled and unlabeled data making it helpful when labeling data is costly or it is very limited.

- Semi Supervised Classification
- Self-Training in Semi-Supervised Learning
- Few-shot learning in Machine Learning

Module 6: Deployment of ML Models

The trained ML model must be integrated into an application or service to make its predictions accessible.

- Machine learning deployement
- Deploy ML Model using Streamlit Library
- Deploy ML web app on Heroku
- Create UIs for prototyping Machine Learning model with Gradio

APIs allow other applications or systems to access the ML model's functionality and integrate them into larger workflows.

- Deploy Machine Learning Model using Flask
- Deploying ML Models as API using FastAPI

MLOps ensure they are deployed, monitored and maintained efficiently in real-world production systems.

- MLOps
- Continuous Integration and Continuous Deployment (CI/CD) in MLOps
- End-to-End MLOps