Subject: U24CST362 – Machine Learning Fundamentals

Course: B. Sc Computer Science with specialization in Artificial Intelligence and Machine Language

Prepared By: SAKSHI PANDEY

UNIT III

Binary Classification vs Multiclass Classification

Logistic Regression for Classification

Decision Tree for Classification and Regression

Random Forest Classifier and Regressor

Support Vector machine (SVM) - Linear and Kernel

Hyperparameter Tuning in Supervised Models

Confusion Matrix and Performance Metrics

Precision, Recall, F1-Score

ROC Curve and AUC Interpretation

Model Selection for Classification Tasks

CLASSIFICATION

Definition:

Classification is a supervised learning technique used to categorize data into predefined classes or labels based on input features.

How It Works:

- **Data Collection:** Gather labeled data (e.g., spam/not spam).
- **Feature Extraction:** Identify useful characteristics (color, texture, shape, etc.).
- **Model Training:** The model learns patterns between features and labels.
- **Evaluation:** Test performance on unseen data.
- **Prediction:** Predict class labels for new inputs.

Types of Classification:

- **Binary Classification:** Two categories (e.g., spam vs not spam).
- **Multiclass Classification:** More than two classes (e.g., cat, dog, bird).
- **Multi-label Classification:** One data point can belong to multiple classes (e.g., movie = action + comedy).

IN DETAIL

Common Algorithms:

- **Linear Models:** Logistic Regression, Linear SVM, Perceptron, SGD Classifier
- **Non-linear Models:** KNN, Kernel SVM, Naive Bayes, Decision Tree
- **Ensemble Models:** Random Forest, AdaBoost, Bagging, Voting, Extra Trees
- **Neural Networks:** Multi-layer Perceptrons

Key Concepts:

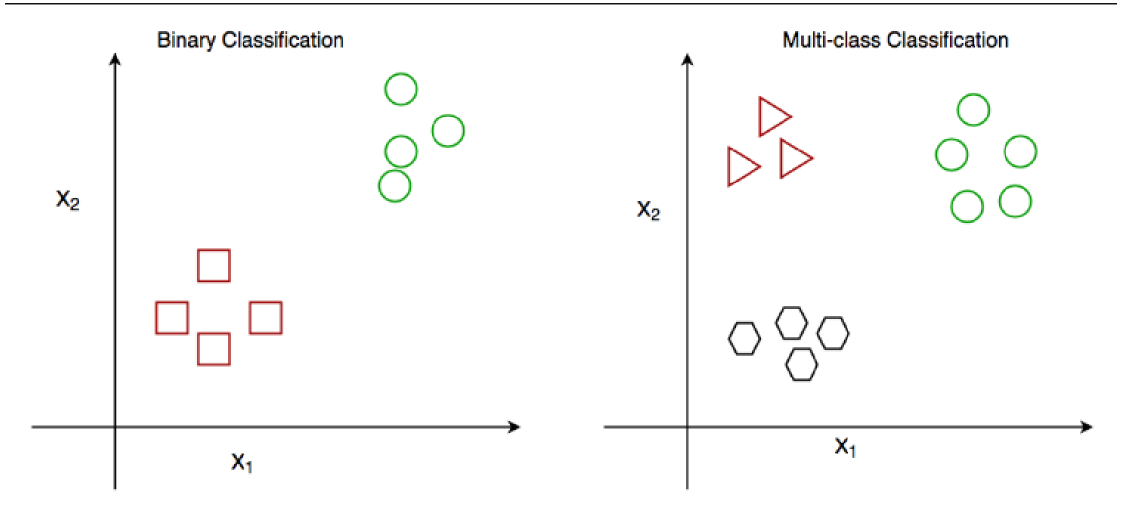
- **Decision Boundary:** Line/region separating classes.
- **Handling Imbalanced Data:** Use resampling or class weighting.
- **Interpretability:** Models like Decision Trees are easier to explain.

Real-life Applications:

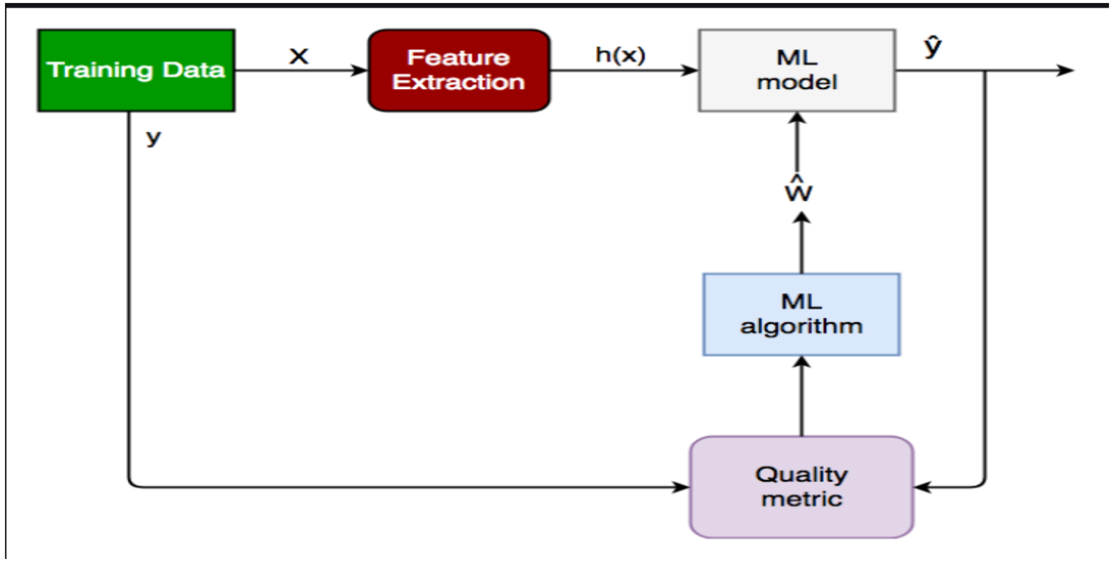
- Email Spam Filtering
- Credit Risk Assessment
- Medical Diagnosis
- Image Classification
- Sentiment Analysis
- Fraud Detection
- Recommendation Systems

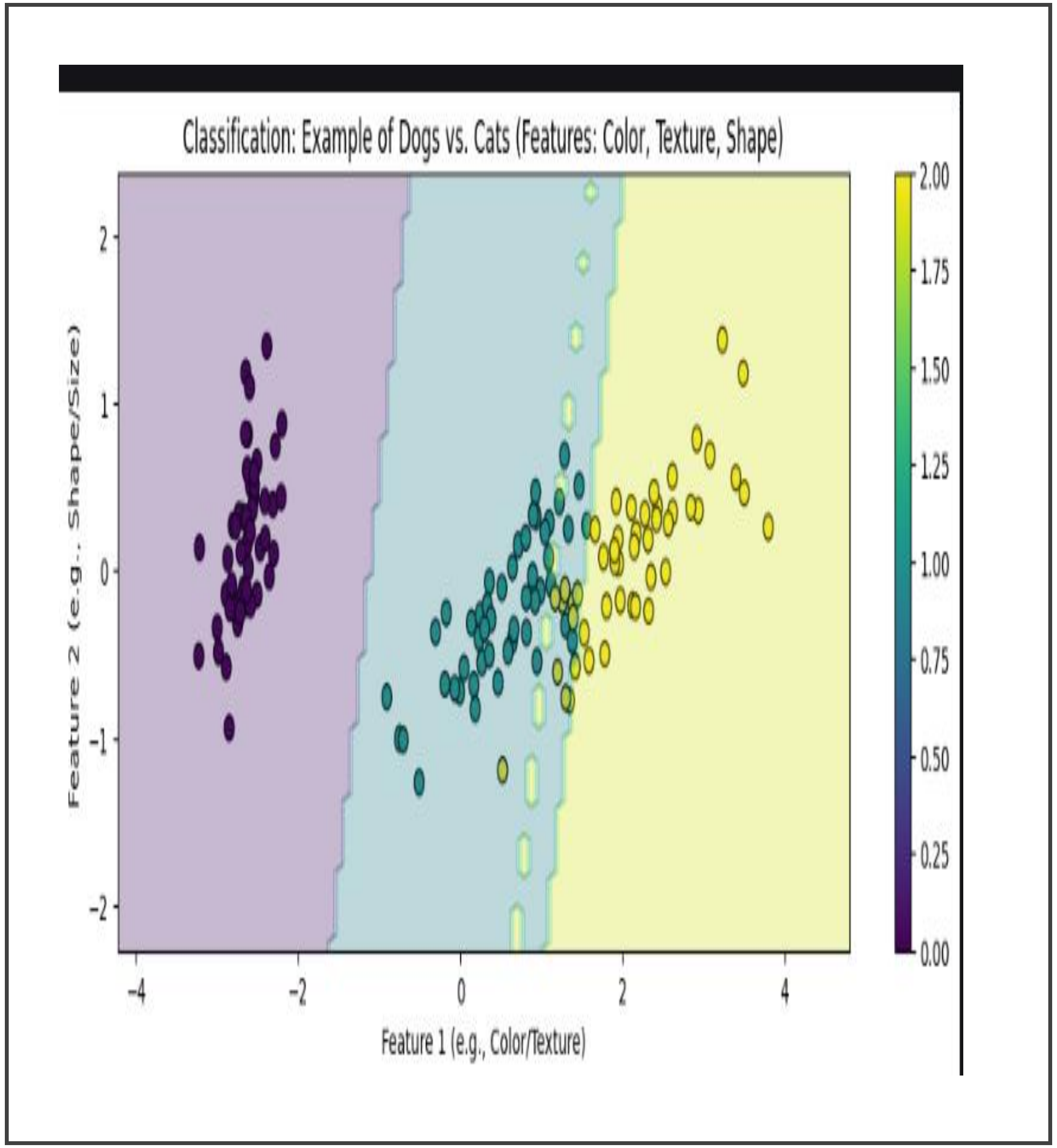
TYPES OF CLASSIFICATION

Type	Description	Example	Key Point
Binary Classification	Classifies data into **two categories** only.	Spam vs. Not Spam, Fraud vs. Legit	Simple Yes/No or True/False decision.
Multiclass Classification	Classifies data into **more than two classes**, but **only one label per input**.	Classifying animals as Cat, Dog, or Bird	Each data point belongs to **one** class only.
Multi-label Classification	Each data point can belong to **multiple classes simultaneously**.	A movie labeled as both *Action* and *Comedy*	Allows **multiple labels** for a single instance.



Binary classification vs Multi class classification





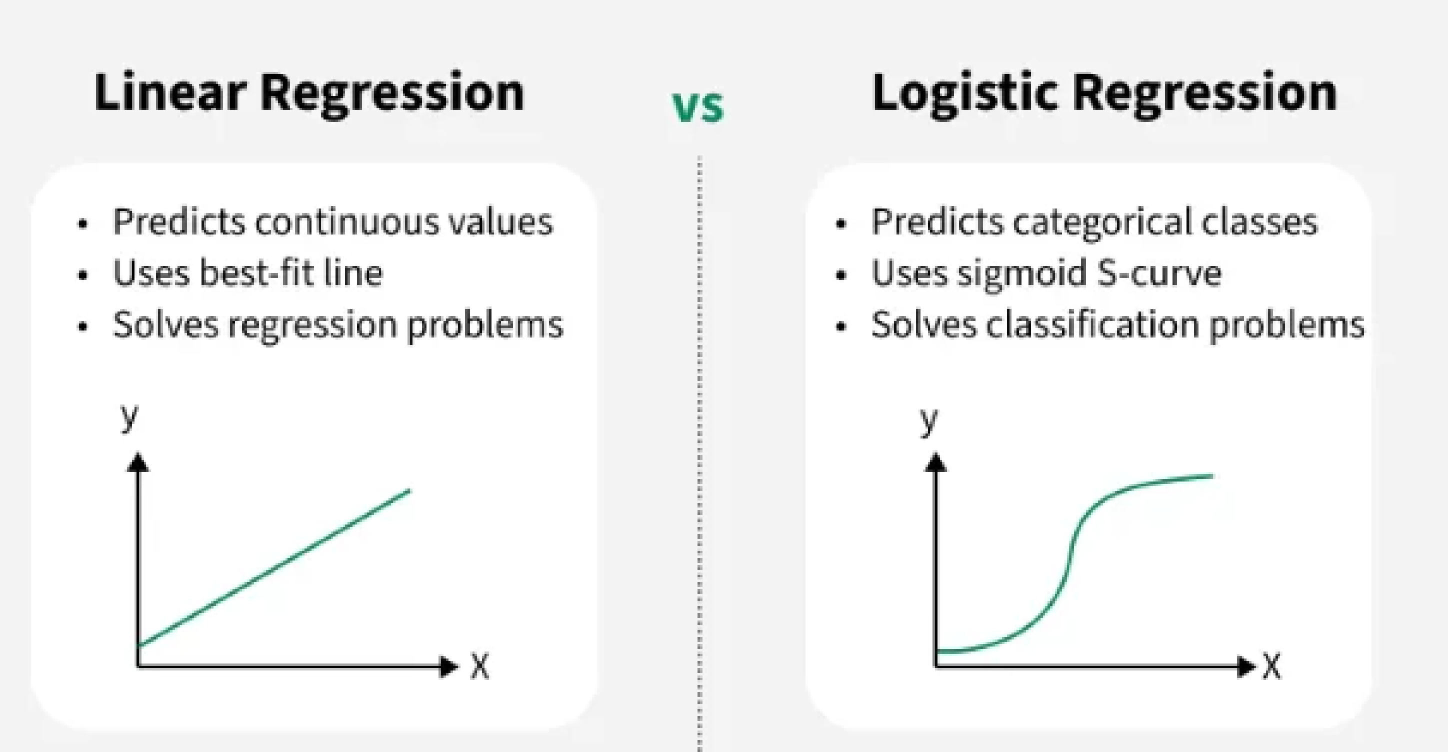
LOGISTIC REGRESSION

Introduction to Logistic Regression

Definition:

- Logistic Regression is a **supervised learning algorithm** used for **classification problems** (mainly binary).
- It predicts the **probability** that an input belongs to a specific class using the **sigmoid function**, which maps outputs between **0 and 1**.

FORMULA : $P(X)=1/1+e-(w\cdot X+b)$



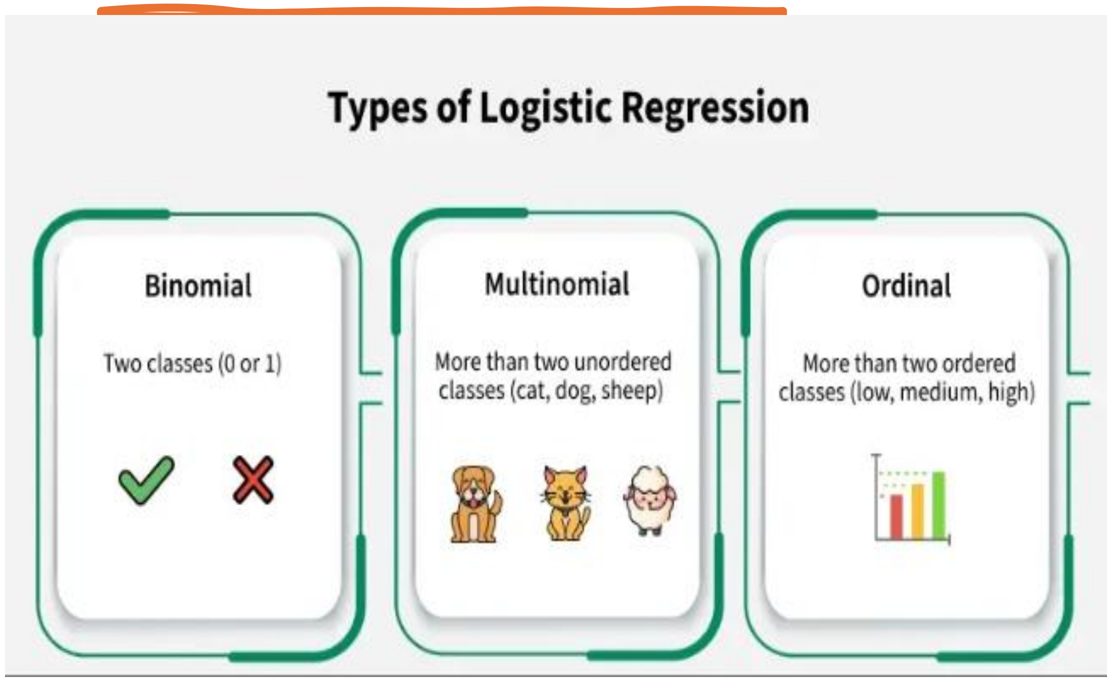
Linear Regression vs **Logistic Regression**

Linear Regression:

- Predicts continuous values
- Uses best-fit line
- Solves regression problems

Logistic Regression:

- Predicts categorical classes
- Uses sigmoid S-curve
- Solves classification problems


Types of Logistic Regression
Binomial
Two classes (0 or 1)
Multinomial
More than two unordered classes (cat, dog, sheep)
Ordinal
More than two ordered classes (low, medium, high)

Type	Description	Example
Binomial	Two possible outcomes	Yes/No, 0/1
Multinomial	Three or more unordered categories	Cat, Dog, Sheep
Ordinal	Ordered categories	Low, Medium, High

Working, Evaluation & Comparison

How It Works:

Compute linear combination:

Apply **Sigmoid Function** → Converts z into probability (0–1)

Classify using a threshold (e.g., ≥0.5 → Class 1, else Class 0)

Evaluation Metrics:

Accuracy = (TP + TN) / Total

Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

F1 Score = 2 × (Precision × Recall) / (Precision + Recall)

AUC-ROC, AUC-PR – Overall performance across thresholds

Key Assumptions:

Independent observations

Linearity between features and log-odds

No extreme outliers

Large sample size

DIFFEREENCE

Linear Regression	**Logistic Regression**
Predicts **continuous** values	Predicts **categorical** values
Used for **regression** problems	Used for **classification** problems
Finds a **best-fit line**	Finds an **S-curve (Sigmoid)**
Uses **Least Squares**	Uses **Maximum Likelihood Estimation**

CART (Classification and Regression Tree)

- **Definition:**
- **CART (Classification and Regression Trees)** is a type of **Decision Tree algorithm** used for both **classification** and **regression** tasks.
- Developed by **Breiman, Friedman, Olshen, and Stone (1984)**.
- It recursively splits the dataset into smaller subsets based on feature values to make predictions.
- **Types of CART:**
- **Key Concepts:**
- **Tree Structure:** Consists of nodes (decisions), branches (splits), and leaves (outcomes).
- **Splitting Criteria:**
- *Classification* → Gini Impurity
- *Regression* → Residual Reduction

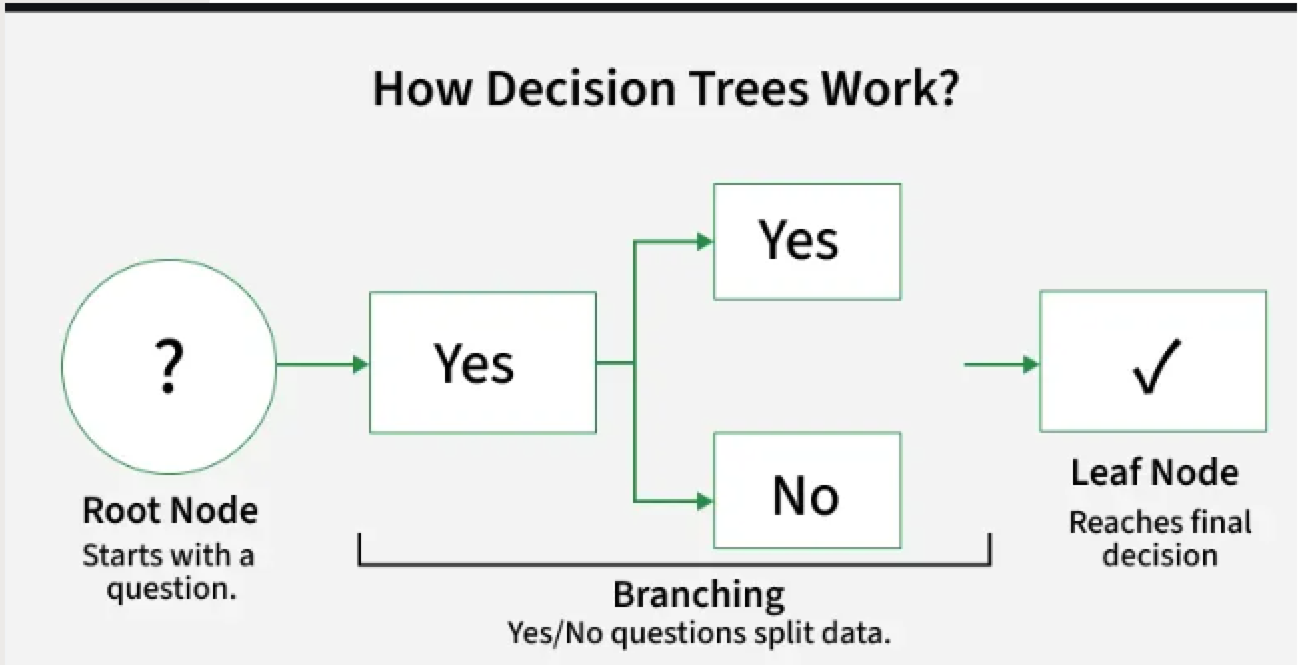
Type	Target Variable	Purpose
Classification Tree	Categorical	Predicts class labels (e.g., Spam / Not Spam)
Regression Tree	Continuous	Predicts numeric values (e.g., House Price)

WORKING

- **Working, Advantages & Applications**
- **Working Steps:**
- Find the **best split** using Gini Index (for classification).
- Split the data into **subsets** to increase purity.
- Repeat recursively until stopping criteria are met.
- **Prune** the tree to improve generalization.
- **Formula – Gini Impurity:**
- Gini=1−∑(pi)2
- where pip_ipi = probability of a class.
- 0 → Pure node, 1 → High impurity

- Advantages:
  - Handles both classification & regression
  - Simple, interpretable, and non-parametric
  - Performs implicit feature selection
  - Robust to outliers
- Limitations:
  - Can overfit on noisy data
  - Sensitive to small data changes
- Applications:
  - Credit risk analysis
  - Medical diagnosis
  - Environmental studies
  - Financial predictions

Decision Tree



Definition:
A **Decision Tree** is a tree-like model used for **classification and prediction**, representing decisions and their possible outcomes through nodes and branches.

Main Components:

Root Node: Represents the entire dataset.

Branches: Show decision paths.

Internal Nodes: Represent tests or decisions on features.

Leaf Nodes: Represent final outcomes or predictions.

Types of Decision Trees:

Type	**Purpose**	**Example**
Classification Tree	Predicts **categorical** outcomes	Spam / Not Spam
Regression Tree	Predicts **continuous** outcomes	House Price Prediction

How It Works:

Start at root → ask feature-based questions.

Split data into subsets (Yes/No).

Continue splitting until reaching leaf nodes (final prediction).

Splitting Criteria:

Gini Impurity – measures purity of node.

Entropy – measures disorder or uncertainty.

Pruning:
Removes unnecessary branches to **prevent overfitting** and improve generalization.

Random Forest

Definition:

Random Forest is an ensemble learning algorithm that builds **multiple Decision Trees** and combines their results for **better accuracy** and **reduced overfitting**.

Classification → Uses **majority voting**

Regression → Uses **average prediction**

Working Principle:

Create many decision trees using **random subsets** of data.

Each tree uses **random features** for splitting.

Each tree makes its own prediction.

Combine predictions → **Majority vote (classification)** or **Average (regression)**.

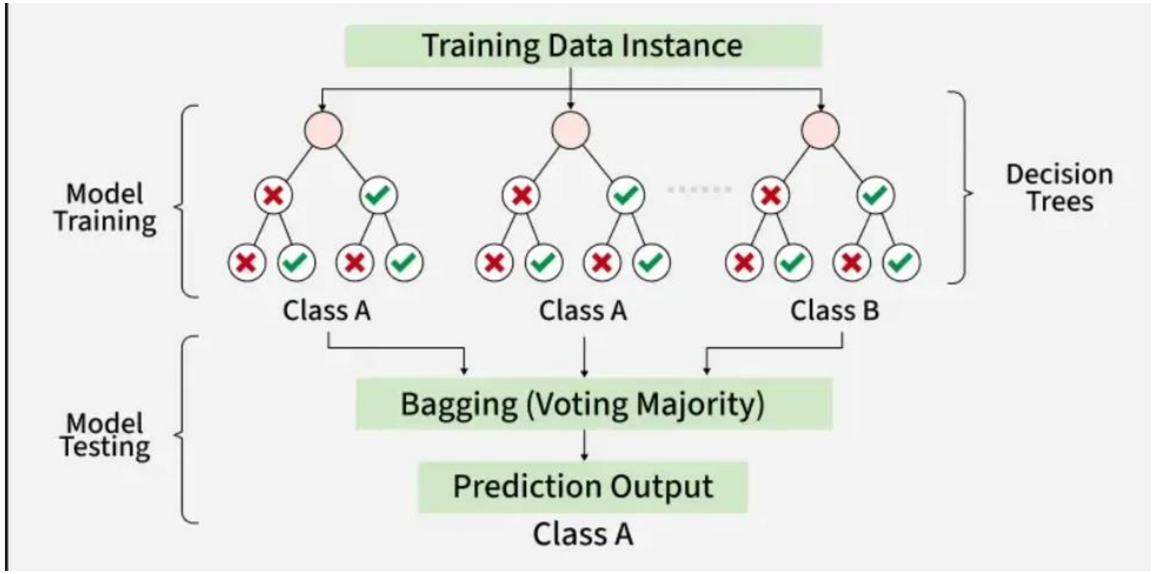
⍰ **Key Features:**

Handles **missing data**

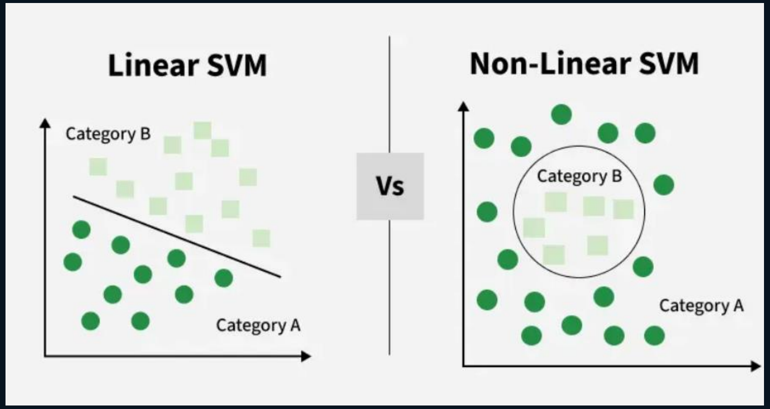
Shows **feature importance**

Works well with **large & complex datasets**

Suitable for both **classification & regression**



Support Vector Machines



Definition:

Support Vector Machine (SVM) is a supervised learning algorithm used for **classification and regression**. It finds the **best boundary (hyperplane)** that separates classes with the **maximum margin**.

Key Concepts:

- **Hyperplane:** Decision boundary separating classes.
- **Support Vectors:** Data points closest to the hyperplane that influence its position.
- **Margin:** Distance between hyperplane and support vectors (maximize this).
- **Kernel:** Maps non-linear data to higher dimensions for separation.
  - Linear, Polynomial, RBF (Gaussian).
- **Hard Margin:** No misclassification allowed.
- **Soft Margin:** Allows some errors for better generalization.

Working:

- Identify a hyperplane that best separates classes.
- Maximize margin between support vectors.
- Use kernel trick for non-linear data.
- Classify new points based on which side of hyperplane they fall.

Hyperparameter Tuning

- **Definition:**
Hyperparameter tuning is the process of finding the **best set of hyperparameters** that control the learning process of a machine learning model to achieve **optimal performance**.
- These are set **before training** (e.g., learning rate, number of trees, kernel type).
- Goal → **Improve accuracy**, **reduce overfitting**, and **enhance generalization**.

Key Techniques:

Advantages:
Boosts model performance

Prevents overfitting/underfitting

Improves generalization

Optimizes resource usage

□ **Challenges:**
High computation cost
Large hyperparameter space
Requires domain knowledge

Applications:
Used in **tuning ML models** like SVM, Random Forest, Neural Networks, Logistic Regression for better results.

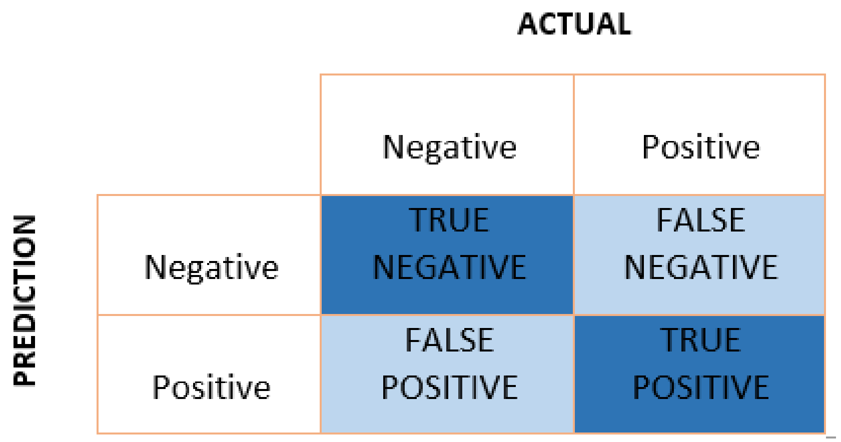
Method	**Description**	**Pros / Cons**
GridSearchCV	Tries **all combinations** of hyperparameters using cross-validation.	Accurate Slow & expensive
RandomizedSearchCV	Tries **random combinations** from a parameter range.	Faster May miss best combo
Bayesian Optimization	Learns from past results to **intelligently choose** next hyperparameters.	Efficient Complex to implement

Confusion Matrix

- **1. Confusion Matrix (2×2 for binary classification):**

- **TP:** Correct positive predictions
- **TN:** Correct negative predictions
- **FP (Type I Error):** Incorrect positive predictions
- **FN (Type II Error):** Incorrect negative predictions
- **2. Key Metrics:**
- **Precision:** TP / (TP + FP) → How many predicted positives are actually positive
- **Recall (Sensitivity / TPR):** TP / (TP + FN) → How many actual positives are correctly predicted
- **F1 Score:** Harmonic mean of Precision & Recall → Balances both, useful for imbalanced data
- **3. Metric Importance Examples:**
- **High Recall:** Medical diagnosis, fraud detection → minimize false negatives
- **High Precision:** Spam detection → minimize false positives
- **Takeaway:**

Confusion matrix & derived metrics give **better insight than accuracy**, especially for **imbalanced datasets** and critical applications.



		ACTUAL	
		Negative	Positive
PREDICTION	Negative	TRUE NEGATIVE	FALSE NEGATIVE
	Positive	FALSE POSITIVE	TRUE POSITIVE

Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

EVALUATION METRICS

Evaluation Metrics in Machine Learning

Purpose:
Measure how well a model performs in classification, regression, or clustering tasks to ensure optimal predictions.

Classification Metrics

- **Accuracy:** % of correct predictions; may mislead on imbalanced data.
- **Precision:** Correct positive predictions / All positive predictions; important when false positives are costly.
- **Recall (Sensitivity):** Correctly identified positives / All actual positives; crucial when missing positives is costly.
- **F1 Score:** Harmonic mean of Precision & Recall; balances both metrics.
- **Log Loss:** Penalizes low probability for true classes; lower is better.
- **AUC-ROC:** Measures model's ability to distinguish classes; higher AUC = better performance.
- **Confusion Matrix:** Shows TP, TN, FP, FN for detailed error analysis.

Regression Metrics

- **MAE (Mean Absolute Error):** Avg. absolute difference between predicted & actual.
- **MSE (Mean Squared Error):** Avg. squared difference; penalizes large errors.
- **RMSE:** Square root of MSE; same units as target.
- **RMSLE:** Penalizes underestimation; useful for wide-ranging targets.
- **R^2 (R-squared):** Proportion of variance explained by model; closer to 1 = better fit.

Clustering Metrics

- **Silhouette Score:** Measures cohesion & separation; closer to +1 = well-clustered.
- **Davies-Bouldin Index:** Measures cluster similarity; lower = better separation.

Key Takeaway:
Selecting the right metric for the problem type is essential to accurately evaluate and improve model performance.

ROC Curve and AUC

AUC-ROC Curve in Machine Learning

Purpose:

Evaluate how well a binary or multiclass classification model distinguishes between positive and negative classes across thresholds.

Key Components:

- **TPR (Recall/Sensitivity):** Correctly predicted positives / All actual positives
- **FPR:** Incorrectly predicted positives / All actual negatives
- **Specificity:** 1 – FPR
- **AUC (Area Under Curve):** Higher AUC → better class separat

ROC Curve:

- Plots TPR vs FPR at different thresholds
- Visualizes trade-off between sensitivity and specificity

Interpretation:

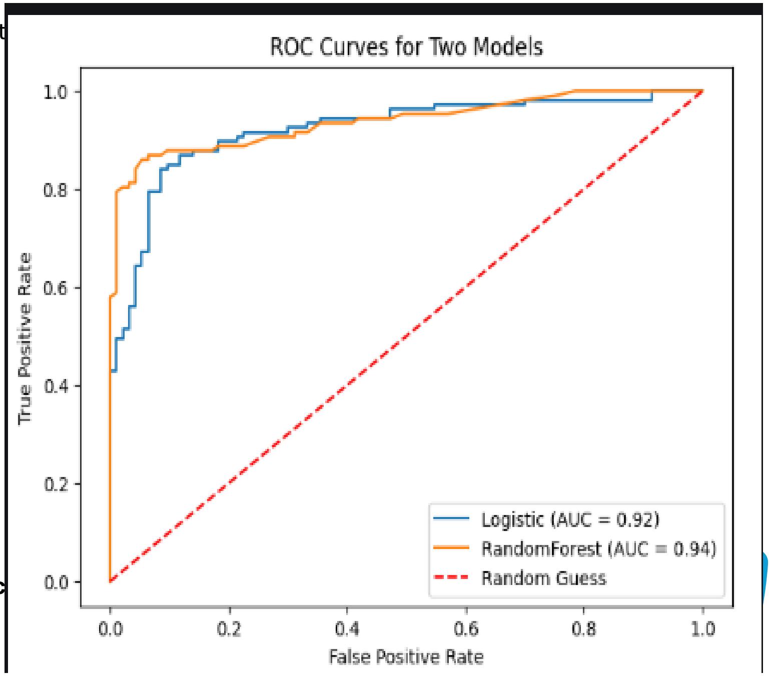
- **AUC = 1:** Perfect model
- **AUC ≈ 0.5:** Random guessing
- **AUC < 0.5:** Worse than random

Usage Tips:

- Effective for balanced datasets
- Less reliable for highly imbalanced datasets → consider **Prec**

Multiclass Extension:

- **One-vs-All approach:** Compute ROC & AUC for each class
- Combine curves to evaluate model's discrimination performance



Takeaway:

AUC-ROC provides a threshold-independent measure of model performance and its ability to rank positive cases higher than negative ones.

MODEL SELECTION

Model Selection in Machine Learning

Definition:
Choosing the best model for a task to ensure high accuracy, efficiency, and reliability.

Importance:

- Avoids **underfitting** (too simple) & **overfitting** (too complex)
- Optimizes **performance** and **computational efficiency**
- Ensures AI systems work well in **real-world scenarios**

Steps in Model Selection:

- **Understand Problem & Data:** Regression, Classification, Clustering; analyze dataset features.
- **Select Suitable Models:**
  - Regression → Linear Regression, Random Forest, Neural Networks
  - Classification → Logistic Regression, SVM, k-NN, Neural Networks
  - Clustering → k-Means, Hierarchical, DBSCAN
- **Model Evaluation:**
  - Split data: Training & Testing sets
  - Use **k-fold cross-validation**
  - Metrics: Accuracy, Precision, Recall, F1-score (Classification); MAE, MSE, R^2 (Regression)

Model Selection Techniques:

- **Grid Search:** Exhaustive hyperparameter search (computationally intensive)
- **Random Search:** Random subset of hyperparameters (faster)
- **Bayesian Optimization:** Uses probability models to predict best hyperparameters
- **Cross-Validation Based Selection:** Averages performance over multiple splits to avoid overfitting

Takeaway:
Proper model selection ensures accurate predictions, **efficient computation**, and **robust real-world performance**.

Unit IV

Concept of Clustering in ML

K- Menas Clustering Algorithm

DBSCAN Algorithm

Hierarchical Clustering

Evaluation of Clustering Models

Introduction to Dimensionality Reduction

Principal Component Analysis

Linear Discriminant Analysis

Feature Selection vs Feature Extraction

Real- World Applications of Unsupervised Learning

Unit V

Steps to Build a Machine Learning Model

Model Training and Testing Process

Improving Model Accuracy and Generalization

Random Forest Classifier

Implementing K- Means and DBSCAN from Scratch

Performance Metrics – Accuracy, Precision, Recall

Senstivity and Specificity

ROC Curve and AUC - Visualizattion

Bias- Variance Tradeoff

Model Comparison and Selection for Deployment

PRACTICE QUESTIONS

Write difference between Classification and regression. (ii) Write five real life application of classification.

Explain Logistic Regression with example. (ii) Explain Support Vector machine with example.

Explain Decision Tree with example. (ii) Explain Random Forest with example.

Explain terms – Classifier, Classification Model, Labels, Features, KNN.

- Explain the complete pipeline for building a machine learning model from scratch. Include data preprocessing, model selection, training, evaluation, and deployment.
- Discuss the importance of splitting data into training, validation, and testing sets. How does each contribute to model development and evaluation?
- Describe various techniques to improve model accuracy and generalization. Include examples like regularization, cross-validation, and feature engineering.
- Explain the working of a Random Forest Classifier. How does it overcome the limitations of a single decision tree?
- Discuss the role of feature importance in Random Forest. How can it be used for feature selection and model interpretation?

NUMERICAL QUESTION1

Datasets have total 20 cats and dog's images. Given that machine learning model trained on Binary classification gives following results. [1] Predicted cat and actual cat are same =10, [2] Predicted not cat and actual is dog = 6 [3] Predicted cat and actual is dog = 3. [4] Predicated not cat and actual is cat =1.

(i) Draw Confusion matrix (ii) Define terms and find value of True Positives, False Positives, True negative, False Negatives (iii) Define terms and find value of Accuracy, Precision, Recall and F1-Score for model.

NUMERICAL QUESTION

Datasets have total 35 cats and dog's images. Given that machine learning model trained on Binary classification gives following results. [1] Predicted cat and actual cat are same =10, [2] Predicted not cat and actual is dog = 11 [3] Predicted cat and actual is dog = 5. [4] Predicated not cat and actual is cat =9.

(i) Draw Confusion matrix (ii) Define terms and find value of True Positives, False Positives, True negative, False Negatives (iii) Define terms and find value of Accuracy, Precision, Recall and F1-Score for model.

CASE STUDIES QUESTIONS

Case Study:

A healthcare startup is building a model to predict whether a patient has diabetes (Yes/No). Later, they expand the system to classify patients into one of four disease categories. Explain the difference between binary and multiclass classification and how model design changes accordingly.

Answer:

- **Binary classification** involves two classes (e.g., diabetic vs non-diabetic). Algorithms like logistic regression or SVM are commonly used.
- **Multiclass classification** involves more than two classes (e.g., diabetes, hypertension, asthma, none).
- Binary models use a single decision boundary, while multiclass models often use one-vs-rest or softmax-based approaches.
- Evaluation metrics shift from binary (accuracy, precision, recall) to macro/micro-averaged scores in multiclass settings.
- Model complexity increases with multiclass tasks, requiring more data and careful handling of class imbalance.

CASE STUDIES QUESTIONS

Case Study:

A bank uses decision trees to classify loan applications as approved or rejected, and to predict loan amounts. Explain how decision trees handle both classification and regression tasks.

Answer:

- For **classification**, decision trees split data based on criteria like Gini impurity or entropy to separate classes.
- For **regression**, they minimize variance within splits to predict continuous values.
- Trees are built top-down, selecting the best feature at each node.
- They are easy to interpret and visualize, but prone to overfitting.
- Pruning techniques and depth control help improve generalization.