Machine Learning Notes

---

1. Introduction to Machine Learning (ML)

- **Definition**:
  Machine Learning is a branch of Artificial Intelligence (AI) that focuses on creating systems that can learn from data and improve performance without being explicitly programmed.
- **How it works**:
  ML models take **input data**, extract **patterns**, and generate **predictions or decisions**.
- **Types of ML**:
  1. **Supervised Learning**
     - Learns from labeled data (input-output pairs).
     - Example: Predicting house prices from size and location.
  2. **Unsupervised Learning**
     - Works on unlabeled data to find hidden structures.
     - Example: Customer segmentation in marketing.
  3. **Reinforcement Learning (RL)**
     - Agent learns by interacting with an environment and receiving rewards/penalties.
     - Example: AlphaGo playing Go, self-driving cars.
- **Applications**:
  - Finance → fraud detection
  - Healthcare → disease diagnosis
  - E-commerce → product recommendations
  - NLP → language translation, chatbots

---

2. Bayesian Decision Theory

- **Concept**:
  A **probabilistic framework** for making optimal decisions under uncertainty.

- **Key Principle**:
  Choose the class with the highest **posterior probability** given the evidence.

P(Ci|x)=P(x|Ci)P(Ci)P(x)P(C_i | x) = \frac{P(x | C_i) P(C_i)}{P(x)}P(Ci|x)=P(x)P(x|Ci)P(Ci)

where:

  - CiC_iCi = class
  - P(Ci)P(C_i)P(Ci) = prior probability of class
  - P(x|Ci)P(x | C_i)P(x|Ci) = likelihood
  - P(Ci|x)P(C_i | x)P(Ci|x) = posterior

- **Decision Rule**:
  Assign xxx to the class with **maximum posterior probability**.
- **Example**:
  Email classification:
  - If words like "win" and "lottery" appear, probability of **Spam** > **Not Spam**.
  - Classifier assigns "Spam".

---

3. Parametric and Non-Parametric Approaches

Parametric Models

- Assume data follows a **fixed functional form**.
- Characterized by a finite set of parameters.
- **Examples**: Linear regression, Logistic regression, Naive Bayes.
- **Advantages**:
  - Simple to train.
  - Requires less data.
- **Limitations**:
  - Can be inaccurate if assumptions about distribution are wrong.

Non-Parametric Models

- Do not assume a fixed distribution.
- Model complexity grows with data.
- **Examples**: k-Nearest Neighbors (kNN), Decision Trees, Kernel methods.

- **Advantages**: Flexible, adapt to complex data.
- **Limitations**: Require large data, may overfit.
- **Example Comparison**:
    - Predicting house price:
        - **Parametric (Linear Regression)** → Fits a straight line between price and size.
        - **Non-Parametric (kNN)** → Finds houses with similar size in training data and averages their prices.

---

4. Perceptron Criteria and Discriminative Models

Perceptron

- A simple binary classifier (1958, Rosenblatt).
- Decision boundary = **linear hyperplane**.
- **Learning Rule**:
  If misclassified → update weights:

$$w_{new} = w_{old} + \eta (y - \hat{y})x$$

where η = learning rate.

- **Limitation**: Only works for **linearly separable data**.

Discriminative Models

- Focus on learning **P(y|x)**, the boundary between classes.
- **Examples**: Logistic regression, SVM, Neural networks.
- **Advantage**: Usually higher accuracy.
- **Contrast**:
    - **Generative models** (e.g., Naive Bayes, HMM) learn full data distribution $P(x, y)$.

---

5. Logistic Regression

- **Definition**:
  A linear model for classification that predicts probabilities using the **sigmoid function**:

P(y=1|x)=11+e−(wTx+b)P(y=1|x) = \frac{1}{1+e^{-(w^Tx+b)}}P(y=1|x)=1+e−(wTx+b)1

- **Decision Rule**:
  If probability > 0.5 → Class 1, else Class 0.
- **Example**:
  Predicting if a student passes:
  - Input: hours studied, attendance.
  - Output: Probability of passing.

---

6. Decision Trees

- **Definition**:
  A flowchart-like structure where nodes split data by feature values.
- **Splitting Criteria**:
  - Information Gain (Entropy)
  - Gini Index
  - Chi-square
- **Advantages**:
  - Easy to visualize.
  - Handles both numerical and categorical data.
- **Limitations**:
  - Overfitting (solved using pruning or Random Forest).
- **Example**: Loan approval
  - Root node: “Income”
  - If > 50k → Approved; If < 50k → Check “Credit Score”.

---

7. Hidden Markov Models (HMMs)

- **Definition**:
  A statistical model for sequential data where states are **hidden** and only observations are visible.
- **Components**:

    - States (hidden): e.g., Weather = {Sunny, Rainy}
    - Observations: e.g., Activities = {Walking, Shopping, Cleaning}
    - Transition probabilities: P(next state | current state).
    - Emission probabilities: P(observation | state).
- **Applications**:
    - Speech recognition: Hidden states = phonemes, Observations = audio signals.
    - NLP: Part-of-speech tagging.

---

8. Ensemble Methods

- **Concept**: Combining multiple models for better performance.

Types:

1. **Bagging**
    - Train models on random subsets of data.
    - Example: Random Forest.
    - Reduces variance (overfitting).
2. **Boosting**
    - Train models sequentially; each model fixes errors of previous.
    - Example: AdaBoost, Gradient Boosting, XGBoost.
    - Reduces bias.
3. **Stacking**
    - Combine predictions of different models using a **meta-model**.

- **Example**:
  In a medical diagnosis system, multiple models (logistic regression, decision tree, SVM) are combined using ensemble methods for more reliable prediction.

---

9. Dimensionality Problems

- **Curse of Dimensionality**:
    - In high dimensions, data becomes sparse.

    - Distance-based algorithms (e.g., kNN) lose effectiveness.
- **Problems**:
    - Overfitting (too many features).
    - Computational cost increases.
    - Harder to visualize and interpret.
- **Solutions**:
    - **Feature Selection**: Keep only important features (e.g., removing irrelevant survey questions).
    - **Dimensionality Reduction**: Transform features into fewer dimensions.
        - PCA (Principal Component Analysis).
        - t-SNE for visualization.
        - Autoencoders (neural networks for compression).
- **Example**:
  In face recognition, instead of using **10,000 pixels per image**, PCA reduces it to **100 dimensions** while preserving key facial features.