Module I: Foundation of Machine Learning

8 Periods

1 Introduction: What Is Learning?

Definition: Learning in machine learning refers to a system improving its performance on a task by extracting patterns from data without explicit programming.

Key Points:

- Learning Process: A system observes data, identifies patterns, and generalizes to predict or decide on new data.
- Components: Task (e.g., classification), data (inputs/labels), model (function mapping inputs to outputs), and performance metric (e.g., accuracy).
- Goal: Minimize errors on unseen data (generalization).
- Example: A spam email classifier learns from labeled emails to predict spam.

2 When Do We Need Machine Learning?

Definition: Machine learning is needed when rule-based programming is infeasible due to complexity, scale, or dynamic environments.

Key Points:

- Scenarios:
  - Complex patterns (e.g., image recognition).
  - Large datasets (e.g., customer behavior prediction).
  - Dynamic environments (e.g., fraud detection).
  - No explicit rules (e.g., autonomous driving).
- Contrast with Traditional Programming: Traditional: Input + Rules $\rightarrow$ Output; ML: Input + Output $\rightarrow$ Rules.
- Examples: Handwriting recognition, recommendation systems.

3 Types of Learning

Definition: Machine learning tasks are categorized based on data and feedback.

Key Points:

- Supervised Learning: Labeled data $\{(x_i, y_i)\}$, predicts y for new x.
  - Classification: Discrete outputs (e.g., spam/not spam).
  - Regression: Continuous outputs (e.g., house prices).
  - Example: Email spam classification.

- Unsupervised Learning: Unlabeled data $\{x_i\}$, finds patterns.
  - Clustering: Groups similar data (e.g., customer segmentation).
  - Dimensionality Reduction: Reduces dimensions (e.g., PCA).
  - Example: Market basket analysis.
- Reinforcement Learning: Agent learns via rewards/penalties.
  - Example: Game-playing AI (e.g., AlphaGo).
- Other Types: Semi-supervised, self-supervised learning.

4 The Statistical Learning Framework

Definition: Formalizes ML as finding a function $f : \mathcal{X} \to \mathcal{Y}$ minimizing expected risk over an unknown distribution $P(X, Y)$.

Key Points:

- Components:
  - Input space $\mathcal{X}$, output space $\mathcal{Y}$.
  - Hypothesis class $\mathcal{H}$ (e.g., linear models).
  - Loss function L (e.g., squared error).
- Expected Risk: $R(f) = \mathbb{E}_{(x,y)\sim P}[L(f(x), y)]$.
- Challenge: $P(X, Y)$ is unknown; use training data $D = \{(x_i, y_i)\}_{i=1}^n$, i.i.d. from P.
- Goal: Minimize $R(f)$.

5 Empirical Risk Minimization (ERM)

Definition: ERM minimizes the empirical risk, the average loss over training data.

Key Points:

- Empirical Risk:
$$\hat{R}_n(f) = \frac{1}{n} \sum_{i=1}^{n} L(f(x_i), y_i).$$
- Objective: $f^* = \arg\min_{f\in\mathcal{H}} \hat{R}_n(f)$.
- Assumption: For large n, $\hat{R}_n(f) \approx R(f)$.
- Limitations: Overfitting for complex $\mathcal{H}$, no preference among similar-risk functions.
- Example: Linear regression minimizing mean squared error.

6 ERM with Inductive Bias

Definition: Adds regularization or constraints to ERM to incorporate prior assumptions.

Key Points:

- Objective:
$$f^* = \arg\min_{f \in \mathcal{H}} \left[\hat{R}_n(f) + \lambda \Omega(f) \right],$$
where $\Omega(f)$ is regularization, λ balances terms.
- Inductive Bias Examples:
  - L2 regularization: $\Omega(f) = \|w\|_2^2$.
  - L1 regularization: $\Omega(f) = \|w\|_1$.
  - Structural bias: Restricting $\mathcal{H}$ (e.g., linear functions).
- Benefits: Reduces overfitting, incorporates domain knowledge.
- Trade-off: Strong bias may cause underfitting.
- Example: Ridge regression.

7 PAC Learning

Definition: Probably Approximately Correct (PAC) learning ensures low error with high probability.

Key Points:

- PAC Learnability: $\mathcal{H}$ is PAC-learnable if, for any $\epsilon, \delta > 0$, there exists $m(\epsilon, \delta)$ such that, with probability $1 - \delta$, the algorithm outputs $h \in \mathcal{H}$ with $R(h) \leq \epsilon$.
- Components: Error ϵ, confidence $1 - \delta$, sample complexity.
- Assumptions: i.i.d. data, finite complexity (e.g., VC dimension).
- Example: Linear classifier with finite samples.

8 Agnostic PAC Learning

Definition: Extends PAC learning to non-realizable cases, finding the best hypothesis in $\mathcal{H}$.

Key Points:

- Objective: $R(h) \leq \min_{f \in \mathcal{H}} R(f) + \epsilon$, with probability $1 - \delta$.
- Difference from PAC: True function may not be in $\mathcal{H}$.
- Key Insight: Uses generalization bounds (e.g., VC dimension).
- Example: Neural network in complex, non-realizable cases.

9 Learning via Uniform Convergence

Definition: Ensures empirical risk converges uniformly to expected risk for all $f \in \mathcal{H}$.

Key Points:

- Formal Definition: $\sup_{f \in \mathcal{H}} |\hat{R}_n(f) - R(f)| \leq \epsilon$, with probability $1 - \delta$.
- Implication: Guarantees ERM finds near-optimal hypothesis.
- Conditions: Finite $\mathcal{H}$ or bounded complexity (e.g., VC dimension).
- Example: Finite set of decision trees.

10 Finite Classes Are Agnostic PAC Learnable

Definition: Finite hypothesis classes are agnostic PAC learnable due to uniform convergence.

Key Points:

- Why Finite?: Bounded by Hoeffding's inequality, sample complexity depends on $\log |\mathcal{H}|$.
- Sample Complexity:
$$m \geq \frac{1}{\epsilon^2} \left(\log |\mathcal{H}| + \log \left(\frac{1}{\delta} \right) \right).$$
- Implication: ERM suffices for finite $\mathcal{H}$.
- Example: 100 linear classifiers.

11 The Bias-Complexity Trade-off: No-Free-Lunch Theorem

Definition: No-Free-Lunch Theorem states no algorithm outperforms others across all problems.

Key Points:

- Core Idea: Performance depends on data distribution; inductive bias is necessary.
- Bias-Complexity Trade-off:
  - High bias, low complexity: Simple models underfit.
  - Low bias, high complexity: Complex models overfit.
- Example: Linear model vs. neural network.

12 Error Decomposition

Definition: Breaks down expected error into bias, variance, and irreducible error.

Key Points:

- Components:

  - Bias: Error due to simplistic assumptions.
  - Variance: Error due to sensitivity to data variations.
  - Irreducible Error: Noise inherent to the problem.
- Formula (squared loss):

$$\mathbb{E}[(y - f(x))^2] = \text{Bias}^2(f) + \text{Variance}(f) + \sigma^2.$$

- Trade-off: High bias → low variance; high variance → low bias.
- Example: Polynomial regression (degree 1 vs. degree 10).

13 Exam Pattern Summary

Key Question Types:

1. Definitions (2–3 marks): Define supervised learning, PAC learning, ERM.
2. Short Answers (5–6 marks): Explain ML need, ERM vs. ERM with bias, No-Free-Lunch Theorem.
3. Long Answers (8–10 marks): Describe Statistical Learning Framework, uniform convergence, error decomposition.
4. Mathematical Problems (5–10 marks): Compute sample complexity, derive regularized ERM.

Study Tips:

- Memorize formulas: Expected risk, empirical risk, sample complexity, error decomposition.
- Practice examples: Linear regression (ERM), neural networks (inductive bias).
- Understand relationships: ERM → PAC → Uniform Convergence → Learnability.