Learning Theory: A Simplified Overview

1. Introduction

Learning Theory helps us understand how machine learning models can generalize from examples. It asks questions like: How many training samples do we need? How do we know our model will perform well on unseen data? These questions are answered through the mathematical framework of **Probably Approximately Correct (PAC) Learning**.

2. Key Questions in Learning Theory

- How can we ensure the model performs well on unseen data?
- How close is our learned function (hypothesis h) to the true function (f)?
- How many examples do we need to train a good model?
- How complex should our hypothesis space be?
- How can we avoid overfitting while still capturing patterns?

3. Intuition Behind PAC Learning

The idea is simple: if a hypothesis makes too many wrong predictions, we'll quickly notice it after seeing enough examples. Hence, a hypothesis that fits all observed data is probably a good one — not perfect, but **probably approximately correct**.

4. Definition of PAC Learning

A **PAC Learning Algorithm** is one that can find a hypothesis that is approximately correct with high probability. Formally, it finds an h such that:

error(h) ≤ ■ with probability ≥ 1 - ■

Here, - ■ (epsilon) = allowed error (approximation)
- ■ (delta) = probability of failure (confidence)

5. PAC Learning Assumptions

1. **Stationarity:** Future and past examples come from the same distribution P(X, Y).
2. **Noiseless Data:** The target function f is deterministic.
3. **Realizability:** The true function f is part of the hypothesis space H being considered.

6. Understanding Errors and Loss

The **0/1 loss function** measures whether the prediction is correct or not:

L■■(y, ■) = 0 if y = ■, else 1

The **expected generalization loss** (or expected error) over all data is:

GenLoss(h) = Σ L(y, h(x)) P(x, y)

7. Approximate Correctness

A hypothesis h is approximately correct if its error rate is less than or equal to ε:

$\text{error}(h) \leq \blacksquare$

8. Deriving Sample Complexity

Let’s define δ as a small probability that our hypothesis is bad (wrong). We want this probability to be very small:

$P(H\_bad \text{ contains a consistent hypothesis}) \leq |H|(1 - \blacksquare)^{\blacksquare} \leq \delta$

From this, we derive the **sample complexity** (number of training examples N required):

$N \geq (1/\blacksquare) \times (\ln(1/\blacksquare) + \ln|H|)$

9. Practical Implications

If the hypothesis space is too large, the number of examples required grows significantly. To balance learning and generalization, we must restrict the hypothesis space — but not too much, or we risk missing the true function. We use principles like **Ockham’s Razor** (prefer simpler models) and **regularization** to find the right balance.

10. Summary

PAC Learning provides a formal foundation for understanding generalization in machine learning. It shows that with enough data and reasonable assumptions, we can learn hypotheses that are probably (with confidence 1 - ■) and approximately (within error ■) correct.

End of Document