Study Guide: PAC Learning Framework

Student Name

October 9, 2024

1 Introduction to PAC Learning (Section 3.1)

The **Probably Approximately Correct (PAC) Learning** framework formalizes the goal of machine learning: to find a hypothesis h that generalizes well from a limited number of training examples. A hypothesis $h \in H$ is chosen from a hypothesis class H, and the objective is to minimize the risk $L_D(h) = \mathbb{P}_{x \sim D}[h(x) \neq f(x)]$, where f is the target function and D is the distribution over the domain $\mathcal{X}$.

1.1 PAC Learnability Definition

A hypothesis class H is PAC learnable if there exists an algorithm $\mathcal{A}$, and for every distribution D, every target function f, and for every $\epsilon, \delta > 0$, the algorithm produces a hypothesis $h \in H$ such that, with probability at least $1 - \delta$,

$$L_D(h) \leq \epsilon.$$

The algorithm must run in time polynomial in $1/\epsilon$, $1/\delta$, and the size of the input.

1.2 Sample Complexity

The number of training examples required to guarantee PAC learnability is called the **sample complexity**. For a hypothesis class H, the sample complexity is a function of ϵ, δ, and the complexity of H. Generally, the sample complexity grows with the size of H, as larger hypothesis classes have more potential to overfit the data.

2 Uniform Convergence (Section 3.2)

A key concept in PAC learning is **uniform convergence**, which guarantees that the empirical risk $L_S(h)$ is close to the true risk $L_D(h)$ for all hypotheses $h \in H$. This ensures that minimizing the empirical risk leads to a hypothesis with low true risk.

2.1 Empirical Risk Minimization (ERM)

The Empirical Risk Minimization (ERM) principle selects the hypothesis $h \in H$ that minimizes the empirical risk over the sample S:

$$L_S(h) = \frac{1}{|S|} \sum_{i=1}^{|S|} \mathbb{I}[h(x_i) \neq f(x_i)].$$

Uniform convergence ensures that, with high probability, $L_D(h) \approx L_S(h)$.

2.2 VC Dimension and Learnability (Section 3.3)

The **VC (Vapnik-Chervonenkis) dimension** is a measure of the capacity or complexity of a hypothesis class. A class H can shatter a set of m points if, for every possible labeling of the points, there exists a hypothesis in H that perfectly classifies them. The VC dimension of H, denoted as $\mathrm{VC}(H)$, is the maximum number of points that H can shatter.

Theorem (PAC Learnability and VC Dimension): A hypothesis class H is PAC learnable if and only if $\mathrm{VC}(H)$ is finite.

3 Agnostic PAC Learning (Section 3.4)

In the **agnostic setting**, we relax the assumption that the target function f belongs to the hypothesis class H. Here, the goal is to find the hypothesis $h \in H$ that minimizes the *true risk*:

$$h^* = \arg\min_{h \in H} L_D(h).$$

The PAC framework can be extended to the agnostic setting, and the sample complexity depends on both ϵ and the approximation error of the best hypothesis in H.

4 Summary (Section 3.5)

- The PAC learning framework provides a formal method for understanding learnability, based on the probability of producing a hypothesis with low error.
- Uniform convergence and the VC dimension are central concepts in determining whether a hypothesis class is PAC learnable.
- In agnostic PAC learning, the goal is to minimize the true risk even when the target function may not belong to the hypothesis class.

Brief Study Guide on PAC Learning on a Finite Hypothesis Class

Study Notes

1 Overview of PAC Learning

PAC (Probably Approximately Correct) learning is a framework in machine learning introduced by Leslie Valiant in 1984. It formalizes the concept of learning a function from a set of examples, providing theoretical guarantees on how efficiently a learning algorithm can approximate an unknown target function.

The key objective in PAC learning is to learn a hypothesis h from a finite hypothesis class H, based on random samples drawn from a distribution D, such that the hypothesis closely approximates the unknown target function f.

In PAC learning, the goal is to find a hypothesis $h \in H$ that has low error with high probability. More formally, the learner must find h such that with probability at least $1 - \delta$, the hypothesis h has error less than ϵ, where ϵ is the allowable error and δ is the failure probability.

2 How PAC Learning Works

PAC learning operates with the following key components:

- **Target Function f:** An unknown function mapping inputs to outputs.
- **Hypothesis Class H:** A finite set of candidate hypotheses, one of which is to be selected based on training data.
- **Distribution D:** A fixed but unknown distribution from which the training examples are drawn.
- **Error ϵ:** A small parameter representing the allowable error. The goal is to find a hypothesis $h \in H$ whose error is less than ϵ with high probability.
- **Confidence $1 - \delta$:** A parameter specifying the probability that the learning algorithm successfully finds a good hypothesis. The algorithm is allowed to fail with probability at most δ.

The hypothesis h is considered PAC-learned if:

$$\Pr_{x \sim D}\left(h(x) \neq f(x)\right) < \epsilon$$

with probability at least $1 - \delta$.

3 Sample Complexity for a Finite Hypothesis Class

When the hypothesis class H is finite, the sample complexity, or the number of training examples m required to PAC-learn the target function, can be derived as follows:

$$m \geq \frac{\log\left(\frac{|H|}{\delta}\right)}{\epsilon}$$

This formula gives the minimum number of samples needed to ensure that, with high probability, the hypothesis h will have an error of at most ϵ. The components of the formula are:

- $|H|$: The number of hypotheses in the hypothesis class. The larger the hypothesis class, the more samples we need to distinguish between the hypotheses.
- δ: The confidence parameter. A smaller δ (higher confidence) requires more samples.
- ϵ: The accuracy parameter. A smaller ϵ (lower error) requires more samples.

This formula shows that the number of samples depends logarithmically on the size of the hypothesis class, meaning that even a modest increase in the number of hypotheses can lead to a significant increase in the required sample size.

4 How to Tell if a Function is PAC Learnable (Finite Case)

In the finite hypothesis case, the learnability of a function class H depends on the following conditions:

- **Finite Hypothesis Class:** The hypothesis class H must be finite. If $|H|$ is finite, the formula for sample complexity applies directly.
- **Efficient Algorithm:** There must exist a learning algorithm that can output a hypothesis that meets the PAC learning criteria (i.e., within ϵ error and $1-\delta$ confidence).

The sample complexity formula $m \geq \frac{\log\left(\frac{|H|}{\delta}\right)}{\epsilon}$ shows that if we can get enough samples, the hypothesis class is PAC-learnable.

5 Solving a Simple PAC Learning Problem

Consider a finite hypothesis class H consisting of 100 different hypotheses. Suppose we want to PAC-learn the target function with error $\epsilon = 0.05$ and confidence $1-\delta = 0.99$ (i.e., $\delta = 0.01$).

5.1 Problem:

How many samples m are required to PAC-learn the target function with these parameters?

5.2 Solution Outline

We use the sample complexity formula for a finite hypothesis class:

$$m \geq \frac{\log\left(\frac{|H|}{\delta}\right)}{\epsilon}$$

Substitute the values:

$$\begin{aligned} |H| &= 100 \\ \delta &= 0.01 \\ \epsilon &= 0.05 \end{aligned}$$

Calculate the logarithmic term:

$$\log\left(\frac{|H|}{\delta}\right) = \log\left(\frac{100}{0.01}\right) = \log(10,000) = 4\log(10) = 4 \times 2.3026 = 9.2103$$

Plug in the values:

$$m \geq \frac{9.2103}{0.05} = 184.21$$

Thus, at least 185 samples are required to PAC-learn the target function with error $\epsilon = 0.05$ and confidence $1-\delta = 0.99$.

6 Summary

- In PAC Learning with a finite hypothesis class, the goal is to find a hypothesis $h \in H$ that approximates the unknown target function f with low error and high probability.
- The sample complexity for PAC learning depends on the size of the hypothesis class $|H|$, the error ϵ, and the confidence $1 - \delta$.
- The formula for the number of samples needed to PAC-learn a finite hypothesis class is:

$$m \geq \frac{\log\left(\frac{|H|}{\delta}\right)}{\epsilon}$$

- For a finite hypothesis class, the number of samples grows logarithmically with the size of the class, meaning that even a modest increase in the number of hypotheses can significantly increase the sample size required for learning.