Lecture 7: General PAC Guarantee

Lecturer: Jacob Abernethy *Scribes: Xi Liu*
Editors: Luke Brandl, Nghia Nguyen, and Shuang Qiu

7.1 Review: Learning Axis-Aligned Rectangles

$\mathcal{X} = \mathbb{R}^2, \mathcal{Y} = \{0, 1\}, \mathcal{C} = \{[a_1, a_2] \times [b_1, b_2] : a_1 \leq a_2, b_1 \leq b_2\}$
Result: As long as $m \geq \frac{4}{\epsilon} \log \frac{4}{\delta}$, where m is the number of samples, then $R(\hat{r}) \leq \epsilon$ with probability at least $1 - \delta$, where $\hat{r}$ is called *tightest rectangle containing positives*, and $r_s = \hat{r}$.

Fact 7.1. *The* `Tightest Rectangle` *algorithm is not fundamental. Indeed, any consistent hypothesis would have a similar guarantee.*

In fact, a typical PAC learning algorithm outputs any $h \in \mathcal{C}$ that is consistent with training data.

7.2 Simplest General PAC Guarantee

Theorem 7.2. *Let $|\mathcal{C}| < \infty$ and S be the sample set. Let h_s be any $h \in \mathcal{C}$ consistent with the target concept on S: $\hat{R}(h_s) = 0$. As long as $|S| = m \geq \frac{1}{\epsilon}\left(\log|\mathcal{C}| + \log\frac{1}{\delta}\right)$, we have $R(h_s) \leq \epsilon$ with probability at least $1 - \delta$*

Proof: Let $\hat{h} \in \mathcal{C}$ be a hypothesis consistent with the target concept c on S.

$$\begin{aligned}
\Pr_{S\sim D^m}(R(\hat{h}) > \epsilon) &\leq \Pr_{S\sim D^m}(\exists h : h|_s = c|_s \text{ and } R(h) > \epsilon) \\
&\leq \sum_{h\in\mathcal{C}} \Pr_{S\sim D^m}(R(h) > \epsilon \text{ and } h \text{ consistent}) && \text{(Union bound)} \\
&\leq \sum_{h\in\mathcal{C}} \Pr_{S\sim D^m}(h \text{ consistent} | R(h) > \epsilon) && \text{(Definition of conditional probability)} \\
&\leq \sum_{h\in\mathcal{C}} (1-\epsilon)^m \\
&\leq \sum_{h\in\mathcal{C}} e^{-m\epsilon} = |\mathcal{C}|e^{-m\epsilon} \leq \delta
\end{aligned}$$

To complete the proof, note that $e^{m\epsilon} \geq \frac{|\mathcal{C}|}{\delta}$ is equivalent to $m \geq \frac{\log|\mathcal{C}| + \log\frac{1}{\delta}}{\epsilon}$. ■

7.3 Revisit the Definition of PAC Learning

Often data objects require descriptions:

- Might require n bits to describe a binary string.
- Might require n real #S to describe n-dimensional vector.

Also: concepts have description length

Let n = description length of object $x \in \mathcal{X}$, $size(c)$ = description length of $c \in \mathcal{C}$

Definition 7.3 (Updated PAC Learning Definition)**.** *A concept class $\mathcal{C}$ is **PAC learnable** if there is a polynomial form $poly(\cdot,\cdot,\cdot,\cdot)$ and an algorithm $\mathcal{A}$ such that $\forall \epsilon, \delta \geq 0$, for all distributions D on $\mathcal{X}$, for all target concept $c \in \mathcal{C}$, as long as $m \geq poly(\frac{1}{\epsilon}, \frac{1}{\delta}, n, size(c))$, then $\underset{S \sim D^m}{Pr}[R(h_s) > \epsilon] < \delta$, where h_s is an output of $\mathcal{A}$ on S.*

Definition 7.4 (Efficiently PAC Learnable)**.** *If the algorithm $\mathcal{A}$ in Definition 7.3 runs in time $poly(\frac{1}{\epsilon}, \frac{1}{\delta}, n, size(c))$, then we say $\mathcal{C}$ is **efficiently PAC learnable**. When such an algorithm A exists, it is called a PAC-learning algorithm for C.*

Examples

- Let $\mathcal{C}$ be the set of "monotone disjunctions". Here $\mathcal{X} = \{0,1\}^n$ and $c(\mathbf{x}) = x(i_1) \vee x(i_2) \vee ... \vee x(i_k)$ for any subset $\{i_1, ..., i_k\} \subset [n], \forall k \in [n]$. The sample complexity is $m \geq \frac{1}{\epsilon}(\log |\mathcal{C}| + \log \frac{1}{\delta}) = \frac{n + \log \frac{1}{\delta}}{\epsilon}$, which is efficiently PAC learnable.

- Let $\mathcal{C}$ be the *universal concept class* $\mathcal{C} = \{\text{all functions} \mathcal{X} \to \{0,1\}\}$
Sample complexity is $m \geq \frac{\log(\#\text{allfunctions}) \log \frac{1}{\delta}}{\epsilon} \geq \frac{2^n}{\epsilon}$. Therefore, the universal concept class is not PAC learnable.

- Let $\mathcal{C}$ be the class of "short" boolean expressions $c(\mathbf{x}) = (x(11) \vee x(4) \wedge \neg x(3)) \vee \neg(x(1) \vee x(2))$; let's say the length of the expression is no more than s. Then clearly the number of possible hypotheses is no more than the number of possible expressions, hence

$$|\mathcal{C}| \leq (n+s)^k.$$

To PAC learn $\mathcal{C}$ requires only:

$$m = \Omega\left(\frac{k \log n + \log \frac{1}{\epsilon}}{\epsilon}\right),$$

where k is the description length of the function. However, it is *very unlikely* that this class can be efficiently PAC learned, as this problem looks very similar to solving SAT problems (it's not exactly SAT, since the inputs are chosen randomly).

7.4 Some Philosophy

William of Occam, a theologian, made a statement known as Occam's Razor: *Plurality should not be posited without necessity*, i.e. we should tend to search for simplest explanations.