PAC Learnability

Karl Stratos

1 PAC Learning

How can we formalize the “inherent difficulty” of a learning problem? To answer this question, we consider a simple setting. Let C be the **concept class** we're interested in learning, where each concept $c : \mathcal{X} \to \{\pm 1\}$ is a binary classifier of inputs $x \in \mathcal{X}$. Now, there is an **input distribution** D over $\mathcal{X}$. This enables us to sample input $x \sim \mathcal{X}$ according to D.

Definition 1.1 (Learning problem)**.** *We know the concept class C. There is some target concept $c \in C$ we wish to learn. We don't know c, but we can request m iid examples $x^{(1)} \ldots x^{(m)} \sim D$ labeled by c (aka. training data). Denote the set of m labeled examples by $S = \left\{ \left(x^{(i)}, c\left(x^{(i)} \right) \right) \right\}_{i=1}^{m}$. After observing S, we pick a* ***hypothesis*** *$h_S \in C$ that we think is c. The problem is to find a hypothesis with small* ***generalization error****, defined as:*

$$P_{x \sim D} \left(h_S(x) \neq c(x) \right)$$

The relation between generalization error and m tells us how hard it is to learn c. If the target concept is hard, we will need a lot of labeled examples S before h_S has small generalization error. This suggests that we might want to define the learnability of a concept class C as something like:

- C is learnable if for any $c \in C$ and D we can achieve generalization error $P_{x \sim D} \left(h_S(x) \neq c(x) \right) \leq \epsilon$ with $m = |S|$ polynomial in $1/\epsilon$.

But we can't make such a deterministic statement on $P_{x \sim D} \left(h_S(x) \neq c(x) \right)$ because S *is a random variable*! This naturally leads to a soft statement on the error known as PAC (Probably Approximately Correct) learnability.

Definition 1.2 (PAC learnablility)**.** *A concept class C is* ***PAC learnable*** *if there exist some algorithm $\mathcal{A}$ and a polynomial function poly$(\cdot)$ such that the following holds. Pick any target concept $c \in C$. Pick any input distribution D over $\mathcal{X}$. Pick any $\epsilon, \delta \in [0, 1]$. Define $S := \left\{ \left(x^{(i)}, c\left(x^{(i)} \right) \right) \right\}_{i=1}^{m}$ where $x^{(i)} \sim D$ are iid samples. Given $m \geq poly(1/\epsilon, 1/\delta, dim(\mathcal{X}), size(c))$, where $dim(\mathcal{X}), size(c)$ denote the computational costs of representing inputs $x \in \mathcal{X}$ and target c, the generalization error of $h_S \leftarrow \mathcal{A}(S)$ is bounded as*

$$P_{x \sim D} \left(h_S(x) \neq c(x) \right) \leq \epsilon$$

with probability at least $1 - \delta$ (wrt the randomness in S).

This is an extremely robust statement on learnability. No matter how adversarial the setting is (i.e., we can freely choose a target $c \in C$ and an input distribution D to hamper the learner), the algorithm $\mathcal{A}$ must be able to reduce the generalization error arbitrarily small with a number of labeled examples only polynomial in the quantities related to the desired accuracy (ϵ) and confidence ($1-\delta$). Note that this is a nested probability statement. To make it explicit, we can equivalently write

$$P_S\left(P_{x\sim D}\left(h_S(x)\neq c(x)\right)\leq\epsilon\right)\geq 1-\delta \tag{1}$$

or $P_S\left(P_{x\sim D}\left(h_S(x)\neq c(x)\right)>\epsilon\right)<\delta$.

2 Example: Rectangles in $\mathbb{R}^2$

The only way to understand PAC learnability is through an example. A classical example is the concept class of *rectangles*, where each rectangle maps a point on the plane $x\in\mathbb{R}^2$ to $+1$ if it's in the rectangle and -1 otherwise. Unfortunately, showing that this class is PAC learnable involves some rather subtle steps that I find are not completely fleshed out in many textbooks (e.g., Mohri et al.), hence this note.

A rectangle on the plane can be expressed as a function $c_{l,r,b,t}:\mathbb{R}^2\to\{\pm1\}$ for some $l\leq r$ and $b\leq t$ defined as

$$c_{l,r,b,t}(x):=\begin{cases}1 & \text{if } x_1\in[l,r] \text{ and } x_2\in[b,t]\\ -1 & \text{otherwise}\end{cases}$$

Proposition 2.1. *The rectangle concept class*

$$C:=\{c_{l,r,b,t}:l\leq r,\ b\leq t\}$$

is PAC learnable.

How can we prove Proposition 2.1? We must provide *some* algorithm that, given limited supervision, produces a hypothesis with small generalization error with high probability. Here is one such algorithm, which simply picks the smallest rectangle consistent with the given labeled data.

LearnRectangle
Input: m points $\left(x^{(1)},y^{(1)}\right)\ldots\left(x^{(m)},y^{(m)}\right)\in\mathbb{R}^d\times\{\pm1\}$ labeled by some $c\in C$

- Assign

$$l'\leftarrow\max_{i:\,y^{(i)}=1}x_1^{(i)} \qquad r'\leftarrow\min_{i:\,y^{(i)}=1}x_1^{(i)}$$

$$b'\leftarrow\max_{i:\,y^{(i)}=1}x_2^{(i)} \qquad t'\leftarrow\min_{i:\,y^{(i)}=1}x_2^{(i)}$$

Output: $c_{l',r',b',t'}\in C$

Remark This is a conservative algorithm: the output hypothesis can never make false positives. False negatives may only occur in the region

$$R_E:=R_T\backslash R_S$$

where $R_T=[l,r]\times[b,t]$ is the area of the target rectangle and $R_S=[l',r']\times[b',t']$ is the area of the hypothesis rectangle. See Figure 1 for an illustration.

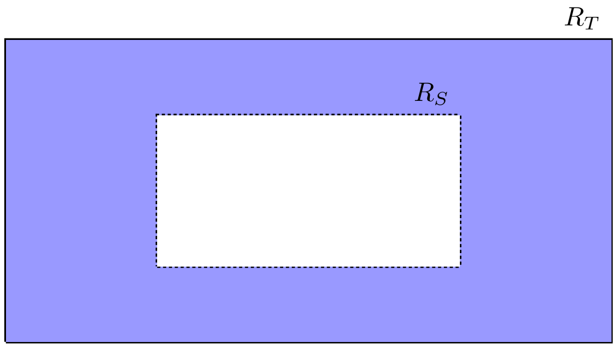

Figure 1: The shaded area is the error region $R_E := R_T \backslash R_S$ (mind the dotted lines). For any distribution D over $\mathbb{R}^2$, the probability mass of R_E is exactly the generalization error $P_{x\sim D}\left(h_S(x) \neq c(x)\right)$ of $h_S \leftarrow$ **LearnRectangle**(S).

Lemma 2.1. *Pick any target rectangle $c \in C$. Pick any input distribution D over $\mathbb{R}^2$. Pick any $\epsilon, \delta \in [0,1]$. Define $S := \left\{\left(x^{(i)}, c\left(x^{(i)}\right)\right)\right\}_{i=1}^m$ where $x^{(i)} \sim D$ are iid samples. Given $m \geq (4/\epsilon)\ln(4/(1-\delta))$, the generalization error of $h_S \leftarrow$* ***LearnRectangle***(S) *is bounded as*

$$P_{x\sim D}\left(h_S(x) \neq c(x)\right) \leq \epsilon$$

with probability at least $1-\delta$ (wrt the randomness in S).

Proof. Let $[l,r] \times [b,t]$ denote the area of the target rectangle R_T and define[1]

$$\begin{aligned}
z_1 &:= \sup\left\{z :\ P\left([l,r] \times [z,t]\right) \geq \epsilon/4\right\} \\
z_2 &:= \sup\left\{z :\ P\left([z,r] \times [b,t]\right) \geq \epsilon/4\right\} \\
z_3 &:= \inf\left\{z :\ P\left([l,r] \times [b,z]\right) \geq \epsilon/4\right\} \\
z_4 &:= \inf\left\{z :\ P\left([l,z] \times [b,t]\right) \geq \epsilon/4\right\}
\end{aligned}$$

They define four subrectangles of R_T each with probability mass **at least** $\epsilon/4$:

$$\begin{aligned}
R_{T,1} &:= [l,r] \times [z_1,t] \\
R_{T,2} &:= [z_2,r] \times [b,t] \\
R_{T,3} &:= [l,r] \times [b,z_3] \\
R_{T,4} &:= [l,z_4] \times [b,t]
\end{aligned}$$

The following are then subrectangles of R_T each with probability mass **at most** $\epsilon/4$:

$$\begin{aligned}
\overline{R}_{T,1} &:= [l,r] \times (z_1,t] \\
\overline{R}_{T,2} &:= (z_2,r] \times [b,t] \\
\overline{R}_{T,3} &:= [l,r] \times [b,z_3) \\
\overline{R}_{T,4} &:= [l,z_4) \times [b,t]
\end{aligned}$$

[1]We assume $P(R_T) \geq \epsilon$: otherwise the proposition is already true.

Now we are ready to give the main argument. Suppose the hypothesis rectangle R_S intersects $R_{T,i}$ for all $i = 1 \ldots 4$. Then the error region $R_E := R_T \backslash R_S$ must be *inside* $\bigcup_{i=1}^4 \overline{R}_{T,i}$ (mind the bar—see Figure 1). Thus in this case,

$$P(R_E) \leq P\left(\bigcup_{i=1}^{4} \overline{R}_{T,i}\right) \leq \sum_{i=1}^{4} P\left(\overline{R}_{T,i}\right) < \epsilon$$

where the last inequality uses the **upper bound** $P\left(\overline{R}_{T,i}\right) < \epsilon/4$. Contrapositively, if $P(R_E) \geq \epsilon$, then R_S does not intersect $R_{T,i}$ for *some* $i \in \{1 \ldots 4\}$. Thus the probability of the event $P(R_E) \geq \epsilon$ (wrt S) can be bounded as

$$\begin{aligned}
P_S\left(P\left(R_E\right) \geq \epsilon\right) &\leq P_S\left(\exists i: \; R_S \cap R_{T,i} = \varnothing\right) \\
&\leq \sum_{i=1}^{4} P_S\left(R_S \cap R_{T,i} = \varnothing\right) \\
&\leq 4\left(1 - \frac{\epsilon}{4}\right)^m
\end{aligned}$$

where the last inequality uses the **lower bound** $P(R_{T,i}) \geq \epsilon/4$. The last term can be further bounded by $4\exp(-m\epsilon/4)$ using the inequality $1 - x \leq \exp(-x)$, and we want this to be at most $1 - \delta$. Solving for m, we have that

$$m \geq \frac{4}{\epsilon} \ln\left(\frac{4}{1-\delta}\right) \qquad \Longrightarrow \qquad P_S\left(P\left(R_E\right) \geq \epsilon\right) \leq 1 - \delta$$

Thus the claim follows. □

3 General Case

Fortunately, we don't need to show the PAC learnability of each and every different concept class. There are general results that allow us to avoid doing the hard work. A high-level picture is the following:

- If the concept class is *finite*, m needed to obtain a PAC hypothesis is polynomially bounded in $1/\delta$, $1/\epsilon$, and $\log|C|$. So if C is not extremely large, it is PAC learnable. For instance, if C is all conjunctions of n Boolean variables, then $\log|C| = \log 3^n = O(n)$ so it is PAC learnable.

- If the concept class is *infinite*, m needed to obtain a PAC hypothesis is polynomially bounded in $1/\delta$, $1/\epsilon$, and a quantity α describing the complexity of C. The quantity α is typically the Rademacher complexity or the VC dimension of C in the literature.[2] For instance, if C is all rectangles on the plane (the example above), then $\text{VCdim}(C) = 4$ so it is PAC learnable. If C is all linear classifiers in $\mathbb{R}^d$, then $\text{VCdim}(C) = d$ so it is PAC learnable.

For more details, see Morhi et al.

[2] This also handles finite concept classes C in the sense that $\text{VCdim}(C) \leq \log_2|C|$.