Topic 6: Bounds on optimal sample complexity

6.1 Comparing sample complexity upper bounds for PAC learning

Using the VC Theorem (Theorem 5.2, combined with Proposition 5.8), we conclude that for any concept class $\mathcal{C} \subseteq \{0,1\}^{\mathcal{X}}$, ERM with $\mathcal{H} = \mathcal{C}$ has the following guarantee: for any probability distribution P over $\mathcal{X} \times \{0,1\}0,1$, if the training data is a size-n iid sample from P, then for any $\delta \in (0,1)$, with probability at least $1-\delta$, the hypothesis $\mathbf{h}$ returned by ERM satisfies

$$\mathsf{err}(\mathbf{h}; P) \leq \inf_{c \in \mathcal{C}} \mathsf{err}(c; P) + O\left(\sqrt{\frac{d \log(n/d)}{n}}\right)$$

where $d := \mathsf{VCdim}(\mathcal{C})$. Under $\mathcal{C}$-realizability, the first term on the right-hand side is zero, and the second term is at most $\epsilon \in (0,1)$ when n is at least

$$O\left(\frac{d \log(1/\epsilon)}{\epsilon^2} + \frac{\log(1/\delta)}{\epsilon^2}\right).$$

The quantity on the right-hand side is therefore an upper bound on the optimal (i.e., lowest achievable) sample complexity for (ϵ, δ)-PAC learning of $\mathcal{C}$, since it is the sample complexity achieved by a particular learning algorithm (ERM). In fact, using an improved version of the VC Theorem (analogous to the improvement of Bennett's inequality over Hoeffding's inequality), the sample complexity achieved by ERM can be improved to the following.

Theorem 6.1. *For any $\epsilon, \delta \in (0,1)$, if the number of training data n is at least*

$$O\left(\frac{d \log(1/\epsilon)}{\epsilon} + \frac{\log(1/\delta)}{\epsilon}\right) \tag{6.1}$$

where $d := \mathsf{VCdim}(\mathcal{H})$, then every hypothesis in $\mathcal{H}$ consistent with an iid sample of size-n from P has error rate on P at most ϵ. Consequently, ERM (Algorithm 4.3 with hypothesis class $\mathcal{H} = \mathcal{C}$) (ϵ, δ)-PAC learns $\mathcal{C}$ with sample complexity as in (6.1) where $d = \mathsf{VCdim}(\mathcal{C})$.

The sample complexity upper bound in Theorem 6.1 is an important improvement over the sample complexity upper bound implied by Theorem 4.2, namely

$$\frac{\ln(|\mathcal{C}|) + \ln(1/\delta)}{\epsilon},$$

which is only applicable to finite concept classes. Similar to Theorem 4.2, the sample complexity depends only polynomially on $1/\epsilon$ and $\log(1/\delta)$ (though the precise dependence on $1/\epsilon$ has gotten a little worse); it replaces the dependence on $\ln(|\mathcal{C}|)$ with a dependence on $\mathsf{VCdim}(\mathcal{C})$.

We make two final remarks on upper bounds. First, the guarantee is said to be *distribution-free* because it applies to any probability distribution P satisfying $\mathcal{C}$-realizability; no assumptions are made on the marginal distribution of P over $\mathcal{X}$. It may be possible to achieve a better sample complexity if one only has to consider specific (classes of) marginal distributions. Second, we note that sample complexity does not take into account computational complexity. In fact, it is not always computationally tractable to implement ERM. Therefore, sample complexity is only part of the story in understanding the difficulty of learning.

Example 6.1. From Example 5.2, the VC dimension of $\mathsf{LTF}_{\mathbb{R}^d}$ is $d+1$. (The same is true for LTF_d over the Boolean d-cube.) Therefore, ERM (or CH) can be used to PAC learn $\mathsf{LTF}_{\mathbb{R}^d}$ with sample complexity

$$O\left(\frac{d\log(1/\epsilon)}{\epsilon} + \frac{\log(1/\delta)}{\epsilon}\right).$$

CH has a computationally-efficient implementation via linear programming: to find a LTF consistent with training data, it suffices to solve a linear feasibility problem, which can be done efficiently via algorithms for linear programming. If the training data is $S = ((x_1, y_1), \ldots, (x_n, y_n)) \in (\mathbb{R}^d \times \{0,1\})^n$, then the system of linear inequalities (in variables $w \in \mathbb{R}^d$ and $\theta \in \mathbb{R}$) to satisfy is

$$\begin{aligned} w^\mathsf{T} x_i - \theta > 0 \quad &\text{for all } i \in [n] \text{ such that } y_i = 1, \\ w^\mathsf{T} x_i - \theta \leq 0 \quad &\text{for all } i \in [n] \text{ such that } y_i = 0. \end{aligned}$$

The system of inequalities is guaranteed to be feasible for training data that is sampled from a $\mathsf{LTF}_{\mathbb{R}^d}$-realizable distribution. $\lozenge$

6.2 Sample complexity lower bound for PAC learning

6.2.1 Statement of the lower bound

A lower bound on the optimal sample complexity of (ϵ, δ)-PAC learning a concept class $\mathcal{C}$ complements the upper bound implied by Theorem 6.1. Since upper bounds are (typically) proved by finding specific algorithms that achieve a certain sample complexity, a lower bound shows how much room for improvement there is (in terms of sample complexity) by designing better algorithms.

Theorem 6.2. *For any concept class $\mathcal{C} \subseteq \{0,1\}^\mathcal{X}$ with $d := \mathsf{VCdim}(\mathcal{C})$, any learning algorithm A, and any $\epsilon \in (0,1)$, there is a probability distribution P over $\mathcal{X} \times \{0,1\}$ that satisfies $\mathcal{C}$-realizability such that, if A receives an iid sample of size-n with*

$$n < \frac{d-1}{16\epsilon},$$

then with probability at least $1/4$, the output $\mathbf{h}$ of A satisfies $\mathsf{err}(\mathbf{h}; P) > \epsilon$.

Theorem 6.2 implies a lower bound of

$$\Omega\left(\frac{d}{\epsilon}\right)$$

on the optimal sample complexity of $(\epsilon, 1/4)$-PAC learning $\mathcal{C}$. A simple modification yields a lower bound for (ϵ, δ)-PAC learning of

$$\Omega\left(\frac{d + \log(1/\delta)}{\epsilon}\right).$$

Up to the $\log(1/\epsilon)$ factor and hidden constants, it matches the upper bound from Theorem 6.1.[1] In particular, it establishes the VC dimension as the key property of a concept class that determines if it is PAC-learnable or not.

[1] In fact, the $\log(1/\epsilon)$ can be removed from the upper bound by using a different learning algorithm (Hanneke, 2016).

Similar to the lower bounds we had given in online learning (e.g., Theorems 1.5, 2.4 and 3.6), the proof of Theorem 6.2 is based on an adversary argument. For each learning algorithm, the adversary chooses an algorithm-specific distribution P that will cause that specific algorithm to fail. The adversary has freedom to choose P in any way as long as it satisfies $\mathcal{C}$-realizability.[2]

6.2.2 Heuristic argument for constant ϵ

Before giving the proof of Theorem 6.2, we give some of ideas in a heuristic proof of a weaker claim: if $n < K \cdot d$ for some constant $K > 0$, then for any learning algorithm, there is a $\mathcal{C}$-realizable distribution P on which the learning algorithm will typically output a hypothesis $\mathbf{h}$ with $\mathsf{err}(\mathbf{h}; P) > 1/(2e^K)$. This will (heuristically) imply that ϵ-learning of $\mathcal{C}$ for $\epsilon = 1/(2e^K)$ has sample complexity more than $K \cdot d$. Here, $1/(2e^K)$ should just be thought of as another positive constant; its particular dependence on K is not important.

The proof must make use of the VC dimension of $\mathcal{C}$, and in particular, the existence of a set of cardinality d that is shattered by $\mathcal{C}$. So let $T := \{t_1, \ldots, t_d\} \subseteq \mathcal{X}$ be such a shattered set. This means that $\mathcal{C}[T] = \{0,1\}^d$, so all 2^d possible ways of labeling the points in T are achievable by concepts from $\mathcal{C}$.

We can think of the iid sample of size-n from P in two parts. The first part is the unlabeled data $\mathbf{x}_1, \ldots, \mathbf{x}_n$, which comprise an iid sample of size-n from the marginal of P over $\mathcal{X}$. Let us assume the marginal distribution of P over $\mathcal{X}$ is the uniform distribution on T. In this case, the iid sample $\mathbf{x}_1, \ldots, \mathbf{x}_n$ can be regarded as "sampling with replacement" from T. The second part is the collection of labels provided the target concept $c^\star$ on the points of $T \cap \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$. In particular, the training data does not reveal the values of $c^\star$ on $T \setminus \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$.

What is the expected number of points from T that are seen in the iid sample $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$? This is an elementary question in probability whose answer is

$$d\left(1 - \left(1 - \frac{1}{d}\right)^n\right).$$

For large d (and using the fact $n < Kd$), this quantity is less than $d(1 - 1/e^K)$.

Thus, the labels assigned by $c^\star$ on more than d/e^K points in T are never seen by the learner. Since all ways to label T are possible under $\mathcal{C}$-realizability, the best that the learner can do is to randomly guess these unseen labels, getting only about half of them correct. So $\mathbf{h}$ errs on more than $d/(2e^K)$ points, i.e., $\mathsf{err}(\mathbf{h}; P) > 1/(2e^K)$.

6.2.3 Heuristic argument for all $\epsilon > 0$

The heuristic argument above forced the learner's hypothesis to have a constant error rate. To prove Theorem 6.2, we only need to force the learner's hypothesis to have error rate more than ϵ, which—in the asymptotic $\epsilon \to 0$—will eventually be smaller than any given constant. We can therefore afford to realize the scenario from the previous heuristic argument on just an $O(\epsilon)$ proportion of the distribution. To achieve this, we have the marginal distribution of P spread out $O(\epsilon)$ probability mass uniformly over the first $d-1$ points $t_1, \ldots, t_{d-1}$ in T. The remaining $1 - O(\epsilon)$ mass is left to t_d, whose only role is to make most of the training data "useless".

[2] PAC learning requires a learning algorithm to succeed under all distributions P satisfying $\mathcal{C}$-realizability, so the adversary has freedom to any marginal distribution of P over $\mathcal{X}$.

In a size-n iid sample from this marginal distribution, only every one in about $\Omega(1/\epsilon)$ of the draws will be among $t_1, \ldots, t_{d-1}$; most of the sample is comprised of the "useless" point t_d. Hence, following the previous heuristic argument, if n is less than $\Omega((d-1)/\epsilon)$, then the learner never sees (the labels of) a constant fraction of $t_1, \ldots, t_{d-1}$, and hence can at best guess their labels. Ultimately, the learner's returned hypothesis is likely to have error rate

$$\mathsf{err}(\mathbf{h}; P) > \Omega(1) \times (\#\text{ missed points among } t_1, \ldots, t_{d-1}) \times (\text{probability mass of each } t_i)$$

which is at least ϵ when all of the constants are set just right.

6.2.4 Rigorous proof for all $\epsilon > 0$

We now turn the heuristic arguments into a rigorous proof of Theorem 6.2.

Proof of Theorem 6.2. The proof uses the probabilistic method to show the existence of the $\mathcal{C}$-realizable P that causes the given learning algorithm to fail. Here, failure means that the hypothesis $\mathbf{h}$ returned by the learning algorithm A has $\mathsf{err}(\mathbf{h}; P) > \epsilon$ with probability at least $1/4$.

To use the probabilistic method, we define a random probability distribution $\mathbf{P}$, specifically one that is uniformly distributed over the set $\{P_b \mid b \in \{0,1\}^d\}$. The members of this set of distributions are defined as follows.

- Each of the distributions P_b for $b \in \{0,1\}^d$ has the same marginal distribution Q over $\mathcal{X}$, where
$$Q(t_1) = \cdots = Q(t_{d-1}) := \frac{4\epsilon}{d-1} \qquad \text{and} \qquad Q(t_d) := 1 - 4\epsilon.$$
Here, $T := \{t_1, \ldots, t_d\}$ is a set of d points shattered by $\mathcal{C}$. Call $t_1, \ldots, t_{d-1}$ the "informative points".

- For $(\mathbf{x}, \mathbf{y}) \sim P_b$, we have
$$\mathbf{x} = t_i \implies \mathbf{y} = b_i, \quad \forall i \in [d].$$
That is, under P_b, the labels are given by the concept $c \in \mathcal{C}$ with the behavior
$$(c(t_1), \ldots, c(t_d)) = (b_1, \ldots, b_d).$$

The random distribution $\mathbf{P}$ can be generated as follows. Let $\mathbf{b} = (\mathbf{b}_1, \ldots, \mathbf{b}_d)$ be uniformly distributed on $\{0,1\}^d$, so $\mathbf{b}_1, \ldots, \mathbf{b}_d$ are iid Bernoulli(1/2) random variables. Then $\mathbf{P} := P_{\mathbf{b}}$.

Let $\mathbf{x}_1, \ldots, \mathbf{x}_n$ be a size-n iid sample from Q, with $n < (d-1)/(16\epsilon)$. The learning algorithm A receives the labels $\mathbf{b}_i$ corresponding to the $t_i \in \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$. It may also receive the label $\mathbf{b}_d$ if $t_d \in \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, but the subsequent analysis will completely ignore t_d and $\mathbf{b}_d$. What is important is that A does not receive the labels $\mathbf{b}_i$ for $t_i \notin \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$.

Let
$$\mathbf{R} := \{i \in [d-1] \mid t_i \in \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}\}$$
be (the indices of) the informative points seen in the sample. By Bernoulli's inequality (i.e., $(1-x)^n \geq 1 - nx$ for all $x \in [0,1]$ and $n \in \mathbb{N}$),

$$\mathrm{E}[|\mathbf{R}|] = (d-1)\left(1 - \left(1 - \frac{4\epsilon}{d-1}\right)^n\right) \leq 4\epsilon n.$$

By Markov's inequality (Proposition 4.1) and the assumption $n < (d-1)/(16\epsilon)$,

$$\Pr\left(|\mathbf{R}| > \frac{d-1}{2}\right) \leq \Pr(|\mathbf{R}| \geq 8\epsilon n) \leq \frac{\mathrm{E}[|\mathbf{R}|]}{8\epsilon n} \leq \frac{1}{2}.$$

So, letting $\mathbf{U} := \{i \in [d-1] \mid t_i \notin \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}\} = [d-1] \setminus \mathbf{R}$ be the (indices of the) informative points not seen among $\mathbf{x}_1, \ldots, \mathbf{x}_n$, we have

$$\Pr\left(|\mathbf{U}| > \frac{d-1}{2}\right) \geq \frac{1}{2}.$$

Define

$$\mathbf{W} := |\{i \in \mathbf{U} \mid \mathbf{h}(t_i) \neq \mathbf{b}_i\}|$$

to be the number of informative points not seen by A and misclassified by A's hypothesis $\mathbf{h}$. The output $\mathbf{h}$ of A is independent of the labels $\{\mathbf{b}_i \mid i \in \mathbf{U}\}$, so

$$\mathbf{W} \mid (\mathbf{h}, \mathbf{x}_{1:n}) \sim \mathrm{Binomial}(|\mathbf{U}|, 1/2).$$

The median of $\mathrm{Binomial}(|\mathbf{U}|, 1/2)$ is $|\mathbf{U}|/2$, so

$$\Pr\left(\mathbf{W} \geq \frac{|\mathbf{U}|}{2} \,\middle|\, (\mathbf{h}, \mathbf{x}_{1:n})\right) \geq \frac{1}{2}.$$

In the event that $|\mathbf{U}| > (d-1)/2$ and $\mathbf{W} \geq |\mathbf{U}|/2$, it follows that

$$\begin{aligned}
\mathsf{err}(\mathbf{h}; P_{\mathbf{b}}) &\geq \sum_{i \in \mathbf{U}} \frac{4\epsilon}{d-1} \cdot \mathbb{1}\{\mathbf{h}(t_i) \neq \mathbf{b}_i\} \\
&= \frac{4\epsilon}{d-1} \cdot \mathbf{W} \\
&> \frac{4\epsilon}{d-1} \cdot \frac{d-1}{4} = \epsilon.
\end{aligned}$$

Therefore, we have shown that

$$\Pr(\mathsf{err}(\mathbf{h}; P_{\mathbf{b}}) > \epsilon) \geq \Pr\left(|\mathbf{U}| > \frac{d-1}{2} \wedge \mathbf{W} \geq \frac{|\mathbf{U}|}{2}\right) \geq \frac{1}{4}.$$

By the tower property of conditional expectations,

$$\Pr(\mathsf{err}(\mathbf{h}; P_{\mathbf{b}}) > \epsilon) = \mathrm{E}\Big[\Pr(\mathsf{err}(\mathbf{h}; P_{\mathbf{b}}) > \epsilon \mid \mathbf{b})\Big] \geq \frac{1}{4}.$$

Therefore, there exists a choice of $b \in \{0,1\}^d$ such that

$$\Pr(\mathsf{err}(\mathbf{h}; P_b) > \epsilon) \geq \mathrm{E}\Big[\Pr(\mathsf{err}(\mathbf{h}; P_{\mathbf{b}}) > \epsilon \mid \mathbf{b})\Big] \geq \frac{1}{4}. \qquad \square$$

The proof of Theorem 6.2 actually considers a so-called Bayesian setting in which the data generating mechanism is, itself, random. Let $L(A, b) := \Pr(\mathsf{err}(A(\mathbf{S}); P_b) > \epsilon)$ where $\mathbf{S} \sim P_b^n$ and $A(S)$ is the hypothesis returned by learning algorithm A on input $S \in (\mathcal{X} \times \{0,1\})^n$. Then, the proof shows that if $n < (d-1)/(16\epsilon)$,

$$\min_A \mathrm{E}[L(A, \mathbf{b})] \geq \frac{1}{4},$$

where $\mathbf{b} \sim \mathrm{Uniform}(\{0,1\}^d)$, and the minimization is taken over all learning algorithms A.