CS272 - Theoretical Foundations of Learning, Decisions, and Games

Lecture 5: Agnostic Learning and Intro to Online Learning

09/11

Lecturer: Nika Haghtalab *Readings: UML Chapter 21*
Scribe: Devon Ding

1 Recap from Last Lecture

In the previous lecture, we wrapped up our discussion of the statistical (PAC) learning framework by exploring how the combinatorial complexity of a concept class controls its sample complexity. In particular, we established the fundamental relationship between the *VC dimension* of a concept class $\mathcal{C}$ and its *growth function*, which counts the number of distinct labelings the class can induce on a set of size m.

This connection allowed us to sharpen our generalization bounds. Instead of the crude $\log |\mathcal{C}|$ dependence that applies only to finite classes, we showed how classes of potentially infinite size can still be PAC-learnable as long as their VC dimension is finite. The growth function enabled a uniform convergence argument that yielded significantly *improved sample complexity bounds* compared to the naive finite-class analysis.

2 From Consistency to PAC Learning

Building on this foundation, we also saw a powerful general theorem that bridges the consistency model and the PAC model. Recall that in the consistency model, the learner is only required to output *some* hypothesis from $\mathcal{C}$ that fits all observed samples perfectly (if one exists). At first glance, this is a very weak guarantee, as it only concerns performance on the training set and says nothing about generalization.

Theorem 2.1 (Consistency $\Rightarrow$ PAC for finite VC dimension)**.** *Let $\mathcal{C}$ be any concept class with $\mathrm{VCdim}(\mathcal{C}) = d$. Any algorithm A that always returns a hypothesis $h \in \mathcal{C}$ consistent with any realizable sample set S also PAC-learns $\mathcal{C}$ with sample complexity*

$$m_{\mathcal{C}}(\epsilon, \delta) = O\left(\frac{1}{\epsilon}\left(d \log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)\right).$$

This theorem shows that *any* consistent learner must also be a PAC learner, provided we see a sufficiently large i.i.d. sample. It unifies two notions of learnability that initially seemed unrelated: consistency is enough to guarantee generalization, if the class has a finite VC dimension and the sample size is large enough.

Remark 2.2. There also exist algorithms (not necessarily consistent) that achieve $O(\frac{1}{\epsilon}(d + \log \frac{1}{\delta}))$ sample complexity, and this bound is tight up to constant factors. However, such algorithms may deliberately trade off small training error for better generalization, and so do not necessarily return a hypothesis h with $\mathrm{err}_S(h) = 0$.

2.1 Lower Bounds on Sample Complexity

We also have matching lower bounds showing that this dependence on d and $\frac{1}{\epsilon}$ is essentially unavoidable:

Theorem 2.3. *Let $\mathcal{C}$ be any concept class of VC dimension d. Then there exists a distribution $\mathcal{D}$ such that any (ϵ, δ)-PAC learner for $\mathcal{C}$ requires at least*

$$m_{\mathcal{C}}(\epsilon, \delta) \;\geq\; \Omega\Big(\frac{1}{\epsilon}\big(d + \log \frac{1}{\delta}\big)\Big)$$

samples.

This shows that our earlier upper bounds are tight up to constant factors, and motivates exploring what happens when we relax the assumptions that make these bounds possible.

3 Agnostic Learning

So far, our discussion has assumed the *realizability* or *consistency* assumption: that there exists some true concept $c^\star \in \mathcal{C}$ such that all examples $(x, y) \sim \mathcal{D}$ satisfy $y = c^\star(x)$. But in many realistic settings, no hypothesis in $\mathcal{C}$ is perfectly consistent with $\mathcal{D}$. This can happen for several reasons:

- **Noisy or imperfect labels.** Even if there is an underlying ground-truth labeling function, the labels we observe may be corrupted. For example, if the data is labeled by crowdsourced workers, they may make errors or disagree on borderline cases.
- **Insufficient features.** Our representation of the input space X might be too simplistic to capture the distinctions needed to separate the classes.
- **Limited expressiveness of the concept class.** Even if the features are expressive, the chosen hypothesis class $\mathcal{C}$ might be too restricted to represent the true decision boundary.

Example. Imagine a binary classification task where X is the set of social media posts and $Y = \{\text{appropriate}, \text{inappropriate}\}$. The labels are obtained from a pool of human annotators. There may be mistakes in their labels (noise), the features may not fully capture the semantic content, and even our hypothesis space (say, all linear classifiers) may not contain a perfect decision rule. In such a scenario, it is unrealistic to hope for a hypothesis h with $\mathrm{err}_{\mathcal{D}}(h) = 0$.

This motivates moving beyond the realizable PAC model to a more general and robust framework, called *agnostic learning* or sometimes referred to as *agnostic PAC model*.

Definition 3.1 (Agnostic Learning)**.** An algorithm A is said to *agnostically learn* a concept class $\mathcal{C}$ if there exists a function $m_{\mathcal{C}}(\epsilon, \delta)$ such that for every distribution $\mathcal{D}$ on $X \times Y$ and every i.i.d. sample set S of size $m \geq m_{\mathcal{C}}(\epsilon, \delta)$, with probability at least $1 - \delta$, the hypothesis $h_S = A(S)$ satisfies

$$\mathrm{err}_{\mathcal{D}}(h_S) \;\leq\; \inf_{h \in \mathcal{C}} \mathrm{err}_{\mathcal{D}}(h) + \epsilon.$$

In words, this means that the learner's output is guaranteed to be nearly as good as the best possible hypothesis in $\mathcal{C}$, even if the true labeling function does not lie in $\mathcal{C}$ at all. The learner is not expected to find the "truth," but simply the best approximation available within $\mathcal{C}$.

3.1 Agnostic Learning via ERM

We study binary classification with 0–1 loss. For a hypothesis h and distribution $\mathcal{D}$, $\mathrm{err}_{\mathcal{D}}(h) := \Pr_{(x,y)\sim\mathcal{D}}[h(x) \neq y]$ and for a sample $S = \{(x_i, y_i)\}_{i=1}^m$, $\mathrm{err}_S(h) := \frac{1}{m}\sum_{i=1}^m \mathbf{1}[h(x_i) \neq y_i]$. An *ERM* algorithm returns $h_S \in \arg\min_{h\in\mathcal{C}} \mathrm{err}_S(h)$.

Theorem 3.2 (ERM is an agnostic learner)**.** *Let $\mathcal{C}$ be a hypothesis class with finite VC dimension d. There is a constant c, such that for any $\epsilon, \delta \in (0,1)$, if*

$$m \;\geq\; c \cdot \Big(\frac{1}{\epsilon^2}\big(d + \log\frac{1}{\delta}\big)\Big),$$

then with probability at least $1 - \delta$ over an i.i.d. sample S of size m, the ERM h_S satisfies

$$\mathrm{err}_{\mathcal{D}}(h_S) \;\leq\; \inf_{h\in\mathcal{C}} \mathrm{err}_{\mathcal{D}}(h) \;+\; \epsilon.$$

Proof sketch. We will prove an $O(\frac{1}{\epsilon^2}(\log|C| + \log\frac{1}{\delta}))$ bound in the finite case; the general theorem will be left as a homework problem as it closely follows the same approach as PAC learning for classes with finite VC dimension.

Step 1: Uniform convergence. Fix $h \in \mathcal{C}$ and define i.i.d. variables $Z_i = \mathbf{1}[h(x_i) \neq y_i] \in [0,1]$ with $\mathbb{E}[Z_i] = \mathrm{err}_{\mathcal{D}}(h)$. Hoeffding's inequality gives

$$\Pr\Big[\big|\mathrm{err}_S(h) - \mathrm{err}_{\mathcal{D}}(h)\big| \geq t\Big] \leq 2e^{-2mt^2}.$$

Taking a union bound over all $h \in \mathcal{C}$ yields

$$\Pr\Big[\sup_{h\in\mathcal{C}} \big|\mathrm{err}_S(h) - \mathrm{err}_{\mathcal{D}}(h)\big| > \tfrac{\epsilon}{2}\Big] \;\leq\; 2|\mathcal{C}|e^{-m\epsilon^2/2}.$$

Step 2: ERM implies near-optimality. Assume the event $\mathcal{E}$ that the LHS above is $\leq \frac{\epsilon}{2}$. Let $h^\star \in \arg\min_{h\in\mathcal{C}} \mathrm{err}_{\mathcal{D}}(h)$. Then

$$\mathrm{err}_{\mathcal{D}}(h_S) \;\leq\; \mathrm{err}_S(h_S) + \tfrac{\epsilon}{2} \;\leq\; \mathrm{err}_S(h^\star) + \tfrac{\epsilon}{2} \;\leq\; \mathrm{err}_{\mathcal{D}}(h^\star) + \epsilon,$$

where the last inequality follows from the fact that h_S is the ERM on set S, so its empirical error is better than $h^\star$. □

4 The Mistake-Bound Model of Online Learning

We now move from the statistical PAC setting, which assumes i.i.d. samples from a fixed distribution, to an *online* setting that makes no stochastic assumptions. Instead of asking how many samples are needed for a small generalization error, we will ask: how many mistakes must a learner make before it converges to correct predictions?

This perspective is especially natural in adversarial environments or interactive tasks where examples arrive sequentially and feedback is immediate.

Protocol. At each round $t = 1, 2, \ldots, T$:

1. The learner receives an instance $x_t \in X$.
2. It predicts a label $\hat{y}_t \in Y$.
3. The true label y_t is then revealed.
4. If $\hat{y}_t \neq y_t$, the learner incurs a *mistake*.

Importantly, we assume *realizability* in this model: there exists some unknown target concept $c^\star \in \mathcal{C}$ such that $y_t = c^\star(x_t)$ for all t. Thus, all mistakes come purely from the learner's initial uncertainty about $c^\star$, rather than from noise or model mismatch.

Goal. We want algorithms that make only a small number of mistakes in total — ideally, bounded by a constant that depends only on $\mathcal{C}$, not on the number of rounds T.

Definition 4.1 (Mistake-Bound Model)**.** An algorithm A is said to *learn* $\mathcal{C}$ in the *mistake-bound model* with mistake bound M if the following holds: for any sequence $(x_1, y_1), \ldots, (x_T, y_T)$ that is consistent with some $c^\star \in \mathcal{C}$ (i.e. $y_t = c^\star(x_t)$ for all t), algorithm A makes at most M prediction mistakes (rounds where $\hat{y}_t \neq y_t$). No i.i.d. or distributional assumption is required on the sequence.

This model shifts our performance measure from *error rate* to *total mistakes*, which is a much stronger and more adversarial guarantee.

Example 4.2 (One-Dimensional Thresholds)**.** *Let us warm up with a simple class:*

$$\mathcal{C}_N = \{c_a : x \mapsto \mathbf{1}[x \geq a] \;\mid\; a \in \{0, 1, \ldots, N\}\}.$$

Each concept c_a predicts 1 if $x \geq a$ and 0 otherwise. Assume the true concept is $c_{a^\star}$. Suppose the learner predicts using the smallest threshold still consistent with all past labels. Each time the learner errs on some x_t, it can eliminate at least one possible threshold value. Since there are only $N + 1$ possible thresholds, it will make at most N mistakes in total. This implies that the mistake bound for this class is at most N. But can it be less?

Let i^t be the rightmost negatively labeled point observed before time t, and j^t the leftmost positively labeled point observed by time t. Then it's clear that we need not make any mistakes on

future x_t if $x_t \leq i^t$ and $x_t \geq j^t$. So, the only additional mistakes will be made when $x_t \in (i^t, j^t)$. Now, take the prediction policy that labels x_t as negative if $x_t \in (i^t, (i^t + j^t)/2]$ and positive if $x_t \in [(i^t + j^t)/2, j^t)$. Note that if we make a mistake at time t, then

$$|j^{t+1} - i^{t+1}| \leq \frac{1}{2}|j^t - i^t|.$$

This implies that every mistake reduces our "confusion interval" — areas where we aren't sure about the label — by half. So overall, this strategy makes at most $\log(N)$ mistakes.

In the next lecture, we show that the strategy used in this example is an instances of a more general algorithm called *the Majority* or *the Halving* algorithm, that's known to incur a mistake bound of at most $\log_2(|\mathcal{C}|)$.