Lecture 3: Agnostic Learning and Bias-Complexity tradeoffs

Akshay Krishnamurthy
akshay@cs.umass.edu

September 21, 2017

1 Recap

Last time we saw Hoeffding's inequality, which quantifies concentration of measure for bounded random variables.

Theorem 1 (Hoeffding)**.** *Let $X_1, \ldots, X_n$ be iid random variables with mean μ and with $X_i \in [a, b]$. Then*

$$\mathbb{P}\left(\left|\frac{1}{n}\sum_{i=1}^{n} X_i - \mu\right| > \epsilon\right) \leq 2\exp\left(\frac{-2n\epsilon^2}{(b-a)^2}\right).$$

In earlier lectures we saw the PAC-learning bound for finite hypothesis classes under the realizability assumption. Specifically we saw that $n_{\mathcal{H}}(\epsilon, \delta) \leq \lceil \log(|\mathcal{H}|/\delta)/\epsilon \rceil$ is an upper bound on the sample complexity for learning a finite hypothesis class under realizability. Today, we are going to move beyond realizability and use Hoeffding's inequality in a crucial way to produce an analogous result.

2 Agnostic PAC-learning

In the original PAC learning model, we assumed a data generation model where $\mathcal{D}$ is a distribution over $\mathcal{X}$ and the labels were chosen by some hypothesis $h^\star \in \mathcal{H}$. This assumption is quite strong, since it does not allow for any inherent noise in the label (i.e. $P(y|x) \in \{0, 1\}$ always). This is not likely in practice, and the Agnostic PAC-learning framework is a way to relax this.

Remark 2. *A simpler model called the* random classification noise *model, also accounts for inherent randomness in the labels. Here the labeling function is still $h^\star \in \mathcal{H}$, but now we have $\mathcal{D}(Y = h^\star(X)|X) = 1 - \eta$ for some parameter η. This model does not account for effects that are not captured by $\mathcal{H}$, so is less general than agnostic learning.*

Formally, in Agnostic PAC-learning, we instead have an arbitrary distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$ where $\mathcal{X}$ is the example space and $\mathcal{Y} = \{0, 1\}$ is the label space. Clearly this is more general, but now there can be inherent label noise, i.e. $\mathcal{D}(y|x) \in [0, 1]$, but also there may be some aspects of the labeling rule that we cannot model using $\mathcal{H}$. However all other definitions are preserved, so we may still write the risk of a hypothesis, under 0/1 loss as

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}\ell(h(x), y) = \mathbb{P}_{(x,y)\sim\mathcal{D}}[h(x) \neq y].$$

And the empirical risk, given samples $\{(X_i, Y_i)\}_{i=1}^n$ iid from $\mathcal{D}$ is

$$\frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\{h(X_i) \neq Y_i\}.$$

The optimal predictor and excess risk. Before turning to the learning problem, let's recall what the optimal predictor is for this setting. We wish to minimize $R(h)$ over all functions $h : \mathcal{X} \to \mathcal{Y}$. This decomposes nicely

$$\begin{aligned}\min_{h:\mathcal{X}\to\mathcal{Y}} \mathbb{P}_{(x,y)\sim\mathcal{D}}[h(x) \neq y] &= \sum_{x\in\mathcal{X}} \mathcal{D}(x)\left(\min_{h(x)\in\{0,1\}} \mathcal{D}(Y = 1|X = x)\mathbf{1}\{h(x) = 0\} + \mathcal{D}(Y = 0|X = x)\mathbf{1}\{h(x) = 1\}\right) \\ &= \sum_{x\in\mathcal{X}} \mathcal{D}(x)\min\{\mathcal{D}(Y = 1|X = x), \mathcal{D}(Y = 0|X = x)\}.\end{aligned}$$

It is useful to introduce the regression function $\eta : \mathcal{X} \to [0,1]$ given by $\eta(x) = \mathcal{D}(Y = 1|X = x)$. Then the optimal rule is $h^\star(x) = \mathbf{1}\{\eta(x) \geq 1/2\}$ and the minimal risk is

$$\min_h R(h) = R(h^\star) = \mathbb{E}_{x\sim\mathcal{D}(x)}[\min\{\eta(x), 1-\eta(x)\}]$$

Exercise 1. *In the random classification noise model, what is the global risk minimizer and what is its risk?*

Agnostic PAC Sample Complexity. As before, we will operate over a restricted hypothesis class $\mathcal{H}$, and since we no longer have the realizability guarantee, it is unreasonable to ask for a hypothesis with risk at most ϵ. First of all, using the expression for the minimal risk, this simply might not be possible for the learning problem, regardless of what $\mathcal{H}$ we use. Moreover, working with $\mathcal{H}$, we may not even be able to get to the minimal risk $R(h^\star)$. Thus, the Agnostic-PAC setting requires a new definition of learnability and sample complexity, which is similar to before, but instead compares with the best risk achievable using hypotheses in $\mathcal{H}$.

Definition 3 (Agnostic PAC Learnability)**.** *A hypothesis $\mathcal{H}$ is agnostic PAC learnable if there exists a function $n_\mathcal{H} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm such that for every $\epsilon, \delta \in (0,1)$ and every distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, when running the learning algorithm on $n \geq n_\mathcal{H}(\epsilon, \delta)$ samples generated iid from $\mathcal{D}$ the algorithm returns a hypothesis $\hat{h}$ with $R(\hat{h}) \leq \min_{h\in\mathcal{H}} R(h) + \epsilon$, except with probability δ.*

The quantification here is the same as for the (realizable) PAC learning definition. The only difference is what we ask of the learning algorithm and really this definition subsumes the previous one that we saw. Indeed the realizability assumptions ensures that $\min_{h\in\mathcal{H}} R(h) = 0$ and so this definition is more general. The important thing is that this definition is something we can actually achieve in this more challenging agnostic setting.

Theorem 4 (Agnostic PAC for finite classes)**.** *Let $\mathcal{H}$ be a class with $|\mathcal{H}| < \infty$. Then $\mathcal{H}$ is agnostically-PAC learnable with*

$$n_\mathcal{H}(\epsilon, \delta) \leq \left\lceil \frac{2\log(2|\mathcal{H}|/\delta)}{\epsilon^2} \right\rceil.$$

Before turning to the proof, some remarks are in order.

1. **Discretization.** We typically don't have a finite hypothesis space (e.g., linear separators). However, typical implementation on a digital system always use finite precision, so in practice we actually do end up working with a large but finite class. For example, if we represent a hypothesis mathematically with d real parameters, our implementation will likely represent hypotheses with d 64-bit floating point numbers, in which case we really only have 2^{64d} possible hypothesis. Plugging this in here gives $O(\frac{d+\log(1/\delta)}{\epsilon^2})$-type sample complexity. But we'll see a more rigorous way to analyze infinite classes in the next lecture.

2. **Alternative presentation.** An equivalent statement is: With probability at least $1-\delta$

$$R(\hat{h}_{S,ERM}) - \min_{h\in\mathcal{H}} R(h) \leq \sqrt{\frac{2\log(2|\mathcal{H}|/\delta)}{n}}.$$

   It is worth becoming familiar with both types or presentation.

3. **Dependence on ϵ.** In the realizable PAC setting, we saw that the sample complexity scaled with $1/\epsilon$, but here it scales with $1/\epsilon^2$. This is much worse since $1/\epsilon^2 \gg 1/\epsilon$ when ϵ is small. This is the price we pay for this harder learning problem, but as you'll see on Homework 1, sometimes better sample complexity bounds are possible under weaker distributional assumptions than realizability.

Proof of Theorem 4. The idea of the proof is to first show that with high probability, the sample S is such that the empirical risk of every hypothesis $h \in \mathcal{H}$ is close to the true risk. If that is the case, then when we minimize the empirical risk (the ERM learning algorithm), then we should approximately minimize the true risk.

Step 1: Concentration. We need to show that the empirical risk is close to the true risk. Let's start by just looking at one hypothesis h.

$$|\hat{R}(h) - R(h)| = \left| \frac{1}{n}\sum_{i=1}^n \mathbf{1}\{h(X_i) \neq Y_i\} - \mathbb{E}\mathbf{1}\{h(X) \neq Y\} \right|.$$

The empirical risk is a sum of iid random variables, with bounded range, and whose mean is the true risk. This is exactly what is required for Hoeffding's inequality. Specifically if we let $Z_i = \mathbf{1}\{h(X_i) \neq Y_i\}$ then $R(h) = \mathbb{E}Z_i$ and $\hat{R}(h) = \frac{1}{n}\sum_{i=1}^n Z_i$. Since we are using the 0/1 loss, we know that $Z_i \in [0,1]$ and hence

$$\mathbb{P}\left(|\hat{R}(h) - R(h)| > t\right) = \mathbb{P}\left(\left|\frac{1}{n}\sum_{i=1}^n Z_i - \mathbb{E}Z_i\right| > t\right) \leq 2\exp\left(-2nt^2\right).$$

By the union bound

$$\mathbb{P}\left(\exists h \in \mathcal{H}, |\hat{R}(h) - R(h)| > t\right) \leq \sum_{h\in\mathcal{H}} \mathbb{P}\left(|\hat{R}(h) - R(h)| > t\right) \leq 2|\mathcal{H}|\exp\left(-2nt^2\right).$$

We want a failure probability of at most δ, so setting the RHS to be at most δ and re-arranging shows that with probability at least $1-\delta$ (where the randomness is over the training sample)

$$\forall h \in \mathcal{H}, |\hat{R}(h) - R(h)| \leq \sqrt{\frac{\log(2|\mathcal{H}|/\delta)}{2n}}.$$

Step 2: ERM. Conditioning on the $1-\delta$ event, apply the inequality to the ERM and the true risk minimizer $\tilde{h} = \text{argmin}_{h\in\mathcal{H}} R(h)$.

$$\begin{aligned} R(\hat{h}_{S,ERM}) - R(\tilde{h}) &= R(\hat{h}_{S,ERM}) - \hat{R}(\hat{h}_{S,ERM}) + \hat{R}(\hat{h}_{S,ERM}) - \hat{R}(\tilde{h}) + \hat{R}(\tilde{h}) - R(\tilde{h}) \\ &\leq \sqrt{\frac{\log(2|\mathcal{H}|/\delta)}{2n}} + \hat{R}(\hat{h}_{S,ERM}) - \hat{R}(\tilde{h}) + \sqrt{\frac{\log(2|\mathcal{H}|/\delta)}{2n}} \leq \sqrt{\frac{2\log(2|\mathcal{H}|/\delta)}{n}}. \end{aligned}$$

Since we want the excess risk to be at most ϵ, if we upper bound the RHS by ϵ and solve for n, we get the result. □

Uniform Convergence In the proof, we used the fact that for every $h \in \mathcal{H}$ the empirical risk concentrates around the true risk with high probability. This is a useful concept known as *uniform convergence*.

Definition 5 (Uniform convergence)**.** *A hypothesis class $\mathcal{H}$ has the* uniform convergence property *if there exists a function $n_{\mathcal{H}}^{UC} : (0,1)^2 \to \mathbb{N}$ such that for every $\epsilon, \delta \in (0,1)$ and every distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, if S is a sample of $n \geq n_{\mathcal{H}}^{UC}(\epsilon, \delta)$ drawn iid from $\mathcal{D}$, then with probability at least $1-\delta$,*

$$\forall h \in \mathcal{H}, |\hat{R}(h) - R(h)| \leq \epsilon.$$

Uniform convergence is one of the simplest ways to show that a concept class is learnable.

Proposition 6 (UC ⇒ Agnostic PAC)**.** *If $\mathcal{H}$ has the uniform convergence property with function $n_{\mathcal{H}}^{UC}$ then $\mathcal{H}$ is Agnostic-PAC learnable with sample complexity $n_{\mathcal{H}}(\epsilon, \delta) \leq n_{\mathcal{H}}^{UC}(\epsilon/2, \delta)$. Moreover ERM is an algorithm that achieves this sample complexity.*

Proof. This claim is embedded in the proof of Theorem 4, but it can be illuminating to separate things this way.

Let S be a sample of size $n \geq n_{\mathcal{H}}^{\text{UC}}(\epsilon/2, \delta)$. By the definition, we know that with probability at least $1-\delta$, $\forall h \in \mathcal{H}, |\hat{R}(h) - R(h)| \leq \epsilon/2$. Now consider the ERM learning rule and let $\tilde{h} = \text{argmin}_{h\in\mathcal{H}} R(h)$. Use the uniform convergence property twice

$$\begin{aligned} R(\hat{h}_{S,ERM}) - R(\tilde{h}) &= R(\hat{h}_{S,ERM}) - \hat{R}(\hat{h}_{S,ERM}) + \hat{R}(\hat{h}_{S,ERM}) - \hat{R}(\tilde{h}) + \hat{R}(\tilde{h}) - R(\tilde{h}) \\ &\leq \epsilon/2 + \hat{R}(\hat{h}_{S,ERM}) - \hat{R}(\tilde{h}) + \epsilon/2 \leq \epsilon. \end{aligned}$$

□

3 Bias-Complexity Tradeoff

Recall the error decomposition from Lecture 1 and let's just focus on ERM predictors from finite hypothesis classes for now. We had two terms, the estimation error and the approximation error.

$$R(\hat{h}_n) - R(h^\star) \leq \underbrace{R(\hat{h}_n) - \min_{h\in\mathcal{H}} R(h)}_{\text{Estimation Error}} + \underbrace{\min_{h\in\mathcal{H}} R(h) - R(h^\star)}_{\text{Approximation Error}} \leq O\left(\sqrt{\frac{\log(|\mathcal{H}|/\delta)}{n}}\right) + \min_{h\in\mathcal{H}} R(h) - R(h^\star)$$

We know how to control the estimation error, but what about the approximation error? At a high level, can we make the excess risk small in general? It seems tricky, since without knowing what the distribution is, it would be incredibly fortuitous if we chose $\mathcal{H}$ so that the approximation error is small. Unfortunately it is not possible to do this in general, and this type of statement is typically formalized with a **No Free Lunch Theorem**

Theorem 7 (No free lunch theorem)**.** *Let A be a learning algorithm for binary classification (with $0/1$ loss) over $\mathcal{X}$ and let $n \leq |\mathcal{X}|/2$. Then there exists a distribution $\mathcal{D}$ over $\mathcal{X} \times \{0,1\}$ such that: (1) $R(h^\star) = 0$ but (2) with probability at least $1/7$ over the sample S of size n we have $R(A(S)) \geq 1/8$.*

Proof. The idea is that if the domain is large, the sample will not contain many of the instances, and the learning algorithm will not know what to do on those instances. Let C be a subset of $\mathcal{X}$ of size $2n$. Consider all possible labeling functions on these $2n$ examples, there are $T = 2^{2n}$ of them and we call them $f_1, \ldots, f_T$. Associated with each f_i, let $\mathcal{D}_i$ be the distribution that is uniform over C with labels provided by f_i. Clearly $R_{\mathcal{D}_i}(f_i) = 0$. It is sufficient to show that for every A

$$\max_{i \in [T]} \mathbb{E}_{S \sim \mathcal{D}_i^n}[R_{\mathcal{D}_i}(A(S))] \geq 1/4, \tag{1}$$

since for any random variable Z taking values in $[0,1]$

$$\mathbb{E}[Z] = \int_0^1 Zp(Z) = \int_0^{1/8} Zp(Z) + \int_{1/8}^1 Zp(Z) \leq 1/8(1 - \mathbb{P}[Z \geq 1/8]) + \mathbb{P}[Z \geq 1/8]$$

Taking Z to be $R_{\mathcal{D}_i}(A(S))$ and re-arranging proves what we wanted. We thus are left to show Eq. (1).

Let us denote a sequence of training examples by $U \in C^n$ and the same sequence, labeled by f_i, by U_i. We are taking maximum over labeling functions f_i and average over the sequence of training examples U. This is only larger than if we took the average over labeling function and the minimum over the training examples U. Formally,

$$\max_{i \in [T]} \mathbb{E}_{S \sim \mathcal{D}_i^n}[R_{\mathcal{D}_i}(A(S)) \geq \min_{U \in C^n} \frac{1}{T} \sum_{i=1}^T R_{\mathcal{D}_i}(A(U_i))$$

Fix the training set U. Since $n \leq 2|C|$, there must be $p \geq n$ other examples $v_1, \ldots, v_p \in C$ that do not appear in U. We will focus on just the errors made on these examples. Observe that

$$R_{\mathcal{D}_i}(h) = \frac{1}{2n} \sum_{x \in C} \mathbf{1}\{h(x) \neq f_i(x)\} \geq \frac{1}{2p} \sum_{r=1}^p \mathbf{1}\{h(v_r) \neq f_i(v_r)\}.$$

Thus,

$$\frac{1}{T} \sum_{i=1}^T R_{\mathcal{D}_i}(A(U_i)) \geq \frac{1}{T} \sum_{i=1}^T \frac{1}{2p} \sum_{r=1}^p \mathbf{1}\{A(U_i)(v_r) \neq f_i(v_r)\} \geq \frac{1}{2} \min_{r \in [p]} \frac{1}{T} \sum_{i=1}^T \mathbf{1}\{A(U_i)(v_r) \neq f_i(v_r)\}$$

Crucially, the set of training examples U is fixed and we're only looking at the error on a fixed unseen example v_r. Now we can partition the labeling functions f_i into $T/2$ pairs, such that each pair agrees on everything except v_r. Since a pair (f_i, f_j) produces the same labeled dataset $U_i = U_j$, it must be the case that

$$\mathbf{1}\{A(U_i)(v_r) \neq f_i(v_r)\} + \mathbf{1}\{A(U_j)(v_r) \neq f_j(v_r)\} = 1.$$

Thus we can re-write the sum over T terms as a sum over the $T/2$ pairs, and each of these must be 1. This means that the average here is at least $1/2$, which proves what we wanted. □

The theorem reveals that the estimation/approximation tradeoff is fundamental, for every learning algorithm, including ERM rules, there is a distribution where we must incur large error. By the decomposition, the error must stem from either estimation or approximation. While we can try to choose $\mathcal{H}$ to make the errors as small as possible, there are always learning problems where we will not be able to do very well.

Put another way, suppose we had a (possibly data-driven) way to choose a small hypothesis class that contained $h^\star$. This would make the approximation error zero, and since the class is small, the estimation error would also be quite small. This would contradict Theorem 7, so this is not possible.

Really our choice is to between choose $\mathcal{H}$ small or $\mathcal{H}$ big. If we choose $\mathcal{H}$ to be big, then it is more plausible that the approximation error is small, but looking at the estimation error term, it will unfortunately be larger. Deciding how to choose $\mathcal{H}$ to balance these terms is the subject of model selection and cross validation, which we'll study in detail in a few lectures.

This theorem also formalizes why inductive bias is important for learning problems, if you think about the choice of $\mathcal{H}$ as capturing your inductive bias. If you have a learning algorthm that has low total risk on a problem, it is in some sense because of the inductive bias.