COS-511: Learning Theory **Spring 2017**

Lecture 2: ERM, Finite Classes and the Uniform Convergence Property

Lecturer: Roi Livni

2.1 PAC Model

In previous lecture, we defined a learning problem that is defined by a triple $(\chi, \mathcal{Y}, \mathcal{C})$ of a domain χ a labeling $\mathcal{Y} = \{-1, 1\}$ and a concept class $\mathcal{C}$.

We've described what is a learning algorithm. A learning algorithm A, receives as input a sample $S = \{(\mathbf{x}_1, y_1), \ldots, (x_m, y_m)\}$ that is generated IID acording to some arbitrary distribution D and returns a hypothesis $h \in \mathcal{H}$ from some hypothesis class.

We said that the objective of the learner A is to return an hypothesis h_S^A such that: if the true labeling is from $\mathcal{C}$ (meaning, for some $h \in \mathcal{C}$ we have $y = h(x)$ a.s.) then: with high probability (over the sample S) the generalization error of the hypothesis h_S^A is small. This is rigorously defined in Def. 1.1

We next describe a few variations of the PAC model:

2.1.1 Variations on the PAC Model

proper/improper: We do not assume that the learner returns a target function $h \in \mathcal{C}$, this is sometimes referred to as *improper* learning. When the algorithm is guaranteed to return a target function $h \in \mathcal{C}$ we will say it performs *proper* learning.

We will later observe that, putting computational issues aside, if a concept class is learnable it will be learnable in the proper model. However, in later lectures, we will discuss computational issues and we'll observe that the improper settings allow us to consider a much richer class of learning algorithms.

Agnostic vs. Realizable In the realizable setting, we made the assumption that some $h \in \mathcal{C}$ achieves zero error. In the agnostic setting, we do not make such an assumption. The objective is then to return some target function h_S^A such that

$$\mathrm{err}(h_S^A) < \min_{h^* \in \mathcal{C}} \mathrm{err}(h^*) + \epsilon.$$

Definition 2.1. *[(agnostic) PAC Learning] A concept class $\mathcal{C}$ of target functions is PAC learnable (w.r.t to $\mathcal{H}$) if there exists an algorithm A and function $m_{\mathcal{C}}^A : (0,1)^2 \to \mathbb{N}$ with the following property:*

Assume $S = ((x_1, y_1), \ldots, (x_m, y_m))$ is a sample of IID examples generated by some arbitrary distribution D. If S is the input of A and $m > m_{\mathcal{C}}^A(\epsilon, \delta)$ then the algorithm returns a hypothesis $h_S^A \in \mathcal{H}$ such that, with

probability $1-\delta$ *(over the choice of the* m *training examples):*

$$err(h_S^A) < \min_{h\in\mathcal{C}} err(h) + \epsilon$$

The function $m_{\mathcal{C}}^A(\epsilon,\delta)$ *is referred to as* the sample complexity *of algorithm* A*.*

Two sources of noise: In the agnostic model, the suboptimality of the class $\mathcal{C}$ may come from two sources of noise.

First, the model does not assume deterministic labeling. i.e. $D(h(x)=y)$ may not necessarily be 0 or 1.

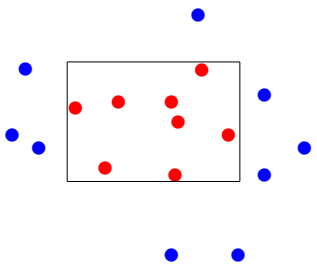
Given non-deterministic labeling, if one has access to the distribution D, the optimal classifier (called Bayes optimal) is always $h_{\text{bayes}}(x) = \max_{0,1} D(y|x)$. In the agnostic model we do not necessarily assume $h_{\text{bayes}} \in \mathcal{C}$.

2.1.2 Examples

Example 2.1 (Axis Aligned Rectangles)**.** *The first example of a hypothesis class will be of rectangles aligned to the axis. Here we take the domain* $\chi = \mathbb{R}^2$ *and we let* $\mathcal{C}$ *include be defined by all rectangles that are aligned to the axis. Namely for every* (z_1,z_2,z_3,z_4) *consider the following function over the plane*

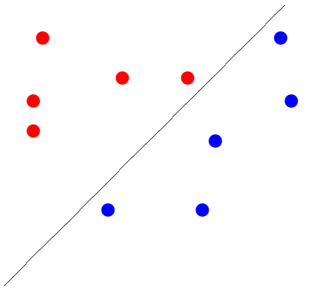
$$f_{z_1,z_2,z_3,z_4}(x_1,x_2) = \begin{cases} 1 & z_1 \le x_1 \le z_2,\ z_3 \le x_2 \le z_4 \\ 0 & \text{else} \end{cases}$$

Then $\mathcal{C} = \{f_{z_1,z_2,z_3,z_4} : (z_1,z_2,z_3,z_4) \in \mathbb{R}^4\}$.



Example 2.2 (Half-spaces)**.** *A second example that is of some importance is defined by hyperplane. Here we let the domain be* $\chi = \mathbb{R}^d$ *for some integer* d*. For every* $\mathbf{w} \in \mathbb{R}^d$*, induces a half space by consider all elements* $\mathbf{x}$ *such that* $\mathbf{w}\cdot\mathbf{x} \ge 0$*. Thus, we may consider the class of target functions described as follows*

$$\mathcal{C} = \{f_{\mathbf{w}} : \mathbf{w} \in \mathbb{R}^d, f_{\mathbf{w}}(x) = \text{sign}(\mathbf{w}\cdot\mathbf{x})\}$$



2.2 Empirical Risk Minimization

Perhaps the simplest strategy for a learner to achieve its objective is the *Empirical Risk Minimization* approach (ERM). Recall that the learner has access to a sample set $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ and its objective is to return a target function h_S^A with small generalization error. The learner does not have access to the true distribution, hence one simple approach is to evaluate its error on the sample. Thus, given a sample S we define the *empirical error* or *training error* of the hypothesis h:

$$\mathrm{err}_S(h) = \frac{1}{m}\sum_{i=1}^{m} \ell_{0,1}(h(x_i), y_i). \tag{2.1}$$

A simple approach to learning is to apply the ERM rule to the restricted class of target functions we wish to learn, $\mathcal{C}$. An algorithm A is an ERM learner if it maintains as a hypothesis class $\mathcal{H} = \mathcal{C}$ and for input S returns a hypothesis h_S^A such that:

$$\mathrm{err}_S(h_S^A) = \min_{h \in \mathcal{H}} \mathrm{err}_S(h).$$

2.3 Finite Classes are Learnable – Through an ERM Algorithm

To motivate the ERM strategy, we will show that finite hypothesis classes are learnable and that an ERM algorithm can learn them. The result follows from two well known principles: Hoeffding inequality (which you will prove, a variant, in the home work assignment) and the union bound principle, which is a simple to prove yet powerfull tool that will play a major role throughout the course:

Claim 2.2 (Union Bound)**.** *Let $A_1, \ldots, A_t$ be some events then*

$$\mathbb{P}(\cup_{i=1}^{t} A_i) \leq \sum_{i=1}^{t} \mathbb{P}(A_i)$$

Theorem 2.3 (Hoeffding's inequality)**.** *Let $X_1, \ldots X_m$ be IID random variables such that $0 \leq X \leq 1$. Set $\bar{X} = \frac{1}{m}\sum_{i=1}^{m} X_i$ then*

$$\mathbb{P}(|\bar{X} - \mathbb{E}(X)| \geq t) \leq 2e^{-2nt^2}$$

Claim 2.4. *Consider a finite class of target functions $\mathcal{H} = h_1, \ldots, h_t$ over a domain χ. Then if $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ is a sample drawn IID from some arbitrary distribution, and if $m > \frac{2}{\epsilon^2} \log \frac{2|H|}{\delta}$ then with probability $1 - \delta$ we have that*

$$\max_{h \in \mathcal{H}} |err_S(h) - err(h)| < \epsilon$$

Proof. For a fixed $h \in H$, consider the random variable $\text{err}_S(h) = \frac{1}{m} \sum_{i=1}^{m} \ell_{0,1}(h(x_i), y_i)$. Note that $\text{err}_S(h)$ is the mean of m IID positive random variables bounded by 1 with expectation $\text{err}(h)$ (namely the random variables $\{\ell_{0,1}(h(x_1), y_1), \ldots, \ell_{0,1}(h(x_m), y_m)\}$.

Applying Hoeffding's inequality we obtain that for every S and fixed h:

$$\mathbb{P}_S(|\text{err}_S(h) - \text{err}(h)| > \epsilon) < 2e^{-\frac{2m}{\epsilon^2}}$$

Applying the union bound we obtain that

$$\mathbb{P}_S(\exists h |\text{err}_S(h) - \text{err}(h)| > \epsilon) \leq 2|\mathcal{H}|e^{-\frac{2m}{\epsilon^2}}.$$

Thus, if $m = O(\frac{2}{\epsilon^2} \log \frac{2|\mathcal{H}|}{\delta})$ we obtain that with probability at least $(1 - \delta)$

$$\max_{h \in \mathcal{H}} |\text{err}_S(h) - \text{err}(h)| < \epsilon$$

□

Corollary 2.5 (Finite Classes are learnable)**.** *Any finite hypothesis class is learnable. In particular, any ERM algorithm can learn a finite hypothesis class with sample complexity $m = O(\frac{1}{\epsilon^2} \log \frac{|H|}{\delta})$.*

Proof. We will prove the result in the agnostic model (which is stronger). Let D be some distribution over $\chi \times \mathcal{Y}$, and A an ERM algorithm. Given a sample S, let h_S^A be the hypothesis returned by algorithm A, and let h^* be the optimal hypothesis in the class. We know by definition that $\text{err}_S(h_S^A) \leq \text{err}_S(h^*)$. On the other hand, choosing m large enough, we have with probability at least $(1 - \delta)$ that

$$|\text{err}_S(h) - \text{err}(h)| \leq \frac{\epsilon}{2} \quad \text{and} \quad |\text{err}_S(h^*) - \text{err}(h^*)| \leq \frac{\epsilon}{2}$$

Thus we get that

$$\text{err}(h) \leq \text{err}_S(h) + \frac{\epsilon}{2} \leq \text{err}_S(h^*) + \frac{\epsilon}{2} \leq \text{err}(h^*) + \epsilon$$

□

2.4 VC–Dimension and Uniform Convergence

2.4.1 Unfirom Convergence (Glivenko–Cantelli)

We begin by defining the notion of uniform convergence. As we will later see uniform convergence is the main property that justifies a use of a ERM algorithm. We will further see that uniform convergence is the main property one needs for learnability.

Definition 2.6 (Uniform Convergence Property)**.** *We say that a hypothesis class $\mathcal{H}$ has the* uniform convergence property *if there exists a function $m : (0,1)^2 \to \mathbb{N}$ such that for every $\epsilon, \delta \in (0,1)$ and for every probability distribution D over χ, if $S = ((x_1, y_1), \ldots, (x_m, y_m))$ is a sample of size $m \geq m(\epsilon, \delta)$ drawn IID according to D then with probability at least $(1-\delta)$ we have that*

$$\sup_{h \in \mathcal{C}} |err_S(h) - err(h)| < \epsilon$$

The notion of uniform convergence is strongly related to the notion of *Glivenko Cantelli* class, and in fact they are equivalent:

Definition 2.7 (Glivenko Cnatelli Class)**.** *Given a distribution D and a target function $h : \chi \to \{0,1\}$ we consider the following random variable*

$$|D_n(h) - D(h)| = |\frac{1}{n}\sum_{i=1}^{n} h(x_i) - \mathop{\mathbb{E}}_{x \sim D}[h(x)]\,|,$$

where $x_1, \ldots, x_n$ are IID sample drawn by the distribution D.

An hypothesis class $\mathcal{H}$ is said to have the uniform convergence property (or a uniform Glivenko–Cantelli Class) if for any distribution D over χ we have

$$\sup_{h \in \mathcal{C}} |D_n(h) - D(h)| \to 0 \quad \textit{almost surely as } n \to \infty.$$

As an example for a class with uniform convergence property, we consider finite classes. The next claim is a direct corollary of Claim. 2.4:

Example 2.3. *A finite Hypothesis class has the uniform convergence property.*

In the next lecture we will define the VC dimension.