ELEG/CISC 867: Advanced Machine Learning **Spring 2019**

Lecture 3: PAC Learning

Lecturer: Xiugang Wu *02/21/2019, 02/26/2019 & 02/28/2019*

In the previous lecture, we have introduced a simplified learning model, and shown that for a finite hypothesis class with realizability assumption, if ERM rule is applied on a sufficiently large training sample (whose size is independent of the underlying distribution or labeling function), then the output hypothesis will be probably approximately correct. This lecture will discuss the PAC (Probably Approximately Correct) learning model in its full generality.

1 PAC Learning Model

Last lecture, we have made several assumptions on the following learning model to simplify our discussion. We now relax these assumptions and introduce the general PAC learning framework.

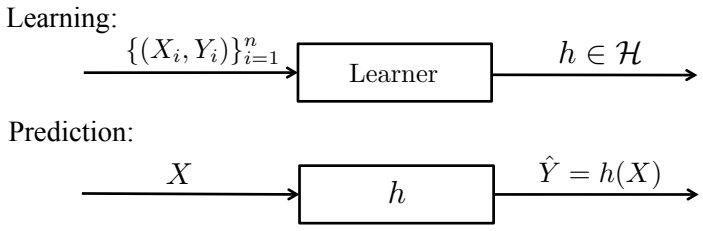

Figure 1: PAC learning model.

Label set. Last time, we assumed the label set $\mathcal{Y} = \{0, 1\}$, which corresponds to predicting a binary label to a given example X. However, many learning tasks take a different form. For example, one may wish to predict a real valued number (say, the temperature at 9:00 p.m. tomorrow), or a label picked from a finite set of labels (like the topic of the main story in tomorrow's paper). One can easily extend our model to such *regression* or *multiclass classification* problems by relaxing the label set to be the set of real vectors or the set of multiple labels. This generalized $\mathcal{Y}$ set is also often referred to as the *target* set.

Data-Generation Mechanism. Last time, we assumed that the target variable Y is fully determined by the input X, which may not be a realistic assumption in many cases, e.g. think of predicting which team will win a basketball game based on their history. From now on, we will replace the "target labeling function" with a more flexible notion, i.e. a data-labels generating distribution. In particular, we consider a joint distribution P_{XY}, or simply P, over $\mathcal{X} \times \mathcal{Y}$. One can view such a distribution as being composed of two parts: a distribution P_X over unlabeled domain points (sometimes called the marginal distribution) and a conditional distribution over labels for each domain point, $P_{Y|X}$.

Performance Measure of a Predictor. We now introduce a general framework of quantifying the performance of a predictor. Given a target set $\mathcal{Y}$ and a reconstructed target set $\hat{\mathcal{Y}}$, let ℓ be any function from $\mathcal{Y} \times \hat{\mathcal{Y}}$ to the set of nonnegative real numbers, $\ell : \mathcal{Y} \times \hat{\mathcal{Y}} \to \mathbb{R}_+$. Such functions are referred to as *loss functions* as they are used to quantify how bad we feel about our reconstruction $\hat{y}$ once we find out the ground truth y. Define the risk $L(h, P)$ associated with a predictor h under data-generating distribution P

as the expected loss when applying h to X, i.e.

$$L(h, P) \triangleq \mathbb{E}_{(X,Y)\sim P}[\ell(Y, h(X))].$$

This risk is also called the true risk as it statistically measures the true performance of predictor h on unseen data. In contrast, one can also consider the empirical risk that the predictor h incurs over the training sample,

$$\frac{1}{n}\sum_{i=1}^{n} \ell(y_i, h(x_i)).$$

It can be readily seen that the the empirical risk is simply the risk of h evaluated under the empirical distribution P_n, i.e. $L(h, P_n)$.

Several widely used loss functions are as follows:

- 0-1 loss: The 0-1 loss, widely used in classification, is defined as

  $$\ell_{0-1}(y, \hat{y}) = \mathbb{I}(y \neq \hat{y}).$$

  The risk of h under distribution P and loss function ℓ_{0-1} is simply the probability of error,

  $$\mathbb{E}_P[\ell_{0-1}(Y, h(X))] = P(Y \neq h(X)).$$

- Square loss: The square loss, also known as ℓ_2 loss or quadratic loss, is usually used in the regression problem and is defined as

  $$\ell_{\text{sq}}(y, \hat{y}) = \|y - \hat{y}\|^2.$$

  The risk of h under distribution P and loss function ℓ_{sq} is generally known as the Mean Square Error (MSE),

  $$\mathbb{E}_P[\ell_{\text{sq}}(Y, h(X))] = \mathbb{E}_P[\|Y - h(X)\|^2].$$

- Logarithmic loss: The logarithmic loss, or log loss in short, is a loss function widely used in classification when the reconstruction is "soft" and $\hat{y}$ represents a distribution over $\mathcal{Y}$,

  $$\ell_{\log}(y, \hat{y}) = \log \frac{1}{\hat{y}(y)}.$$

  The risk of h under distribution P and loss function $\ell_{\log}$ is given by

  $$\mathbb{E}_{(X,Y)\sim P}[\ell_{\log}(Y, h(X))] = \mathbb{E}_{(X,Y)\sim P}[-\log [h(X)](Y)].$$

  The empirical risk of h under training sample z^n and loss function $\ell_{\log}$ is given by

  $$\frac{1}{n}\sum_{i=1}^{n} \log \frac{1}{[h(x_i)](y_i)},$$

  which can be also written as

  $$\frac{1}{n}\sum_{i=1}^{n} H(p_i, q_i),$$

where $H(p_i, q_i)$ is the cross entropy between the distributions p_i and q_i over $\mathcal{Y}$, defined as

$$H(p_i, q_i) \triangleq \sum_{y \in \mathcal{Y}} p_i(y) \log \frac{1}{q_i(y)},$$

and p_i and q_i are respectively the distribution induced by seeing y_i and the distribution the predictor outputs, i.e.

$$p_i(y) = \mathbb{I}(y = y_i),$$
$$q_i(y) = [h(x_i)](y).$$

For this reason, log loss is also widely known as the *cross-entropy* loss.

Bayes Predictor. Suppose that one knows the underlying distribution P. Then the predictor

$$f = \operatorname*{argmin}_h L(h, P)$$

that minimizes the true risk is called the *Bayes predictor*, or *Bayes estimator*, or *Bayes decision rule*, and its resultant risk

$$\min_h L(h, P)$$

is called the *Bayes risk*. Under different loss functions, the Bayes predictor takes different forms:

- 0-1 loss: Under the 0-1 loss, the Bayes predictor f is given by the well-known maximum a posteriori (MAP) rule, i.e.,

  $$f(x) = \operatorname*{argmax}_{y \in \mathcal{Y}} p_{Y|X}(y|x),$$

  with the Bayes risk

  $$L(f, P) = 1 - \sum_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} p_{X,Y}(x, y).$$

- Square loss: Under the square loss, the Bayes predictor f is given by the conditional expectation of Y given $X = x$, i.e.,

  $$f(x) = \mathbb{E}_P[Y|X = x],$$

  with the Bayes risk

  $$L(f, P) = \mathbb{E}_P[\mathrm{Var}(Y|X)].$$

- Log loss: Under the log loss, the Bayes predictor f is given by the conditional distribution of Y given $X = x$, i.e.,

  $$[f(x)](y) = p_{Y|X}(y|x),$$

  with the Bayes risk being the conditional entropy of Y given X:

  $$L(f, P) = \mathbb{E}_{(X,Y)\sim P}[-\log p_{Y|X}(Y|X)] = H_P(Y|X).$$

Unfortunately, since we do not know P, we cannot utilize the above Bayes predictors to achieve the minimal possible error. Instead, what the learner does have access to is the training sample. So we will choose some hypothesis class, and require that the learner will, based on the training sample, find a predictor whose error is not much larger than the best possible error achievable by any hypothesis within the class. This idea is formalized in the following definition of PAC learnability.

1.1 PAC Learnability

Given the above general formulation of a learning model, we are now ready to define PAC learnability.

Definition 1.1 *(PAC Learnability) A hypothesis class $\mathcal{H}$ is PAC (Probably Approximately Correct) learnable if there exist a function $n_{\mathcal{H}} : (0,1) \times (0,1) \to \mathbb{N}$ and a learning algorithm with the following property: For every $\epsilon, \delta \in (0,1)$ and every distribution P, if $n \geq n_{\mathcal{H}}$, then*

$$\mathbb{P}_{Z^n \sim P^n} \left(L(h_{Z^n}, P) \leq \min_{h \in \mathcal{H}} L(h, P) + \epsilon \right) \geq 1 - \delta.$$

Several remarks about the above definition of PAC learnability are as follows:

- Accuracy and confidence parameters: The definition of PAC learnability contains two approximation parameters mentioned before. The accuracy parameter ϵ determines how far the output predictor can be from the optimal one within the class (this corresponds to the "approximately correct"), and the confidence parameter δ indicates how likely the output predictor is to meet that accuracy requirement (corresponds to the "probably" part of "PAC").
- Sample complexity: The function $n_{\mathcal{H}}$ determines the sample complexity of learning $\mathcal{H}$, that is, how many examples at least are required to guarantee a probably approximately correct solution. The sample complexity is a function of the accuracy and confidence parameters. It also depends on properties of the hypothesis class $\mathcal{H}$ — for example, in last lecture we showed that for a finite class satisfying the realizability assumption the sample complexity depends on log of the size of $\mathcal{H}$. In fact, using the above definition of PAC learnability, one can rephrase the result we showed in the last lecture as the following: Every finite hypothesis class $\mathcal{H}$ satisfying the realizability assumption is PAC learnable with sample complexity $n_{\mathcal{H}}(\epsilon, \delta) \leq \frac{\log(|\mathcal{H}|/\delta)}{\epsilon}$.
- Agnostic PAC learning: This framework is also generally known as agnostic PAC learning as it doesn't assume realizability. We will often omit the prefix "agnostic" in this course, in which case PAC learning refers to this general agnostic case.

2 Finite Hypothesis Class: Agnostic Case

We now revisit the finite hypothesis class and show that any finite hypothesis class is learnable with ERM in the agnostic case, i.e. even without the realizability assumption.

2.1 Uniform Convergence Is Sufficient For Learnability

For ERM to work, it suffices to ensure that the empirical risk of all hypothesis in $\mathcal{H}$ are good approximations of their true risk. In other words, we need that uniformly over all hypothesis in $\mathcal{H}$, the empirical risk is close to the true risk. This is formalized in the following.

Definition 2.1 (ϵ-representative sample) *A training sequence Z^n is called ϵ-representative if*

$$\forall h \in \mathcal{H}, |L(h, P_n) - L(h, P)| \leq \epsilon.$$

Lemma 2.1 *If Z^n is $\epsilon/2$-representative, then the output h_{Z^n} of $ERM_{\mathcal{H}}(Z^n)$ satisfies*

$$L(h_{Z^n}, P) \leq L(h^*, P) + \epsilon,$$

where we assume that h^* achieves the minimum risk within the class $\mathcal{H}$.

Proof:

$$L(h_{Z^n}, P) \leq L(h_{Z^n}, P_n) + \epsilon/2 \leq L(h^*, P_n) + \epsilon/2 \leq L(h^*, P) + \epsilon/2 + \epsilon/2 = L(h^*, P) + \epsilon.$$

□

The above lemma implies that to ensure that ERM is a PAC learner, it suffices to show that with probability of at least $1 - \delta$, Z^n is ϵ-representative. The uniform convergence condition formalizes this requirement.

Definition 2.2 (Uniform convergence) *We say $\mathcal{H}$ 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 P, if $Z^n \sim P$ with $n \geq n_{\mathcal{H}}^{UC}$ then with probability of at least $1 - \delta$, Z^n is ϵ-representative.*

Corollary 2.1 *If $\mathcal{H}$ has the uniform convergence property with a function $n_{\mathcal{H}}^{UC}$, then the class is PAC learnable with sample complexity $n_{\mathcal{H}}(\epsilon, \delta) \leq n_{\mathcal{H}}^{UC}(\epsilon/2, \delta)$, and in this case ERM is a successful PAC learner for $\mathcal{H}$.*

2.2 Finite Classes Are Agnostic PAC Learnable

We now show that finite classes are agnostic PAC learnable by showing that uniform convergence holds for any finite hypothesis class. For this, we first introduce a measure concentration inequality due to Hoeffding, which quantifies the gap between empirical averages and their expected value.

Lemma 2.2 (Hoeffding's Inequality) *Let $X_1, X_2, \ldots, X_n$ be a sequence of independent random variables and assume that for all i, $\mathbb{E}[X_i] = \mu$ and $\mathbb{P}(X_i \in [a, b]) = 1$. Then, for any $\epsilon > 0$,*

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

Now consider the empirical risk of any $h \in \mathcal{H}$ under training sample Z^n,

$$L(h, P_n) = \frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, h(X_i)),$$

where $\ell(Y_i, h(X_i)), i \in [1:n]$ are i.i.d. with mean $L(h, P)$ for any $i \in [1:n]$. Let us further assume that the range of ℓ is $[0, 1]$. Then applying Hoeffding's inequality to the sequence of $\ell(Y_i, h(X_i))$, we obtain

$$P^n\left(|L(h, P_n) - L(h, P)| \geq \epsilon\right) \leq 2e^{-2n\epsilon^2},$$

and therefore

$$P^n\left(\exists h \in \mathcal{H}, \text{ s.t. } |L(h, P_n) - L(h, P)| \geq \epsilon\right) \leq 2|\mathcal{H}|e^{-2n\epsilon^2}.$$

This shows that if

$$n \geq \frac{\log(2|\mathcal{H}|/\delta)}{2\epsilon^2},$$

then

$$P^n\left(|L(h, P_n) - L(h, P)| \leq \epsilon, \forall h \in \mathcal{H}\right) \geq 1 - \delta,$$

and the corollary below follows immediately.

Corollary 2.2 *Let $\mathcal{H}$ be a finite hypothesis class and ℓ be a loss function with range $[0,1]$. Then $\mathcal{H}$ enjoys the uniform convergence property with sample complexity*

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

Furthermore, the class is agnostically PAC learnable using the ERM algorithm with sample complexity

$$n_{\mathcal{H}}(\epsilon,\delta) \leq n_{\mathcal{H}}^{UC}(\epsilon/2,\delta) = \frac{2\log(2|\mathcal{H}|/\delta)}{\epsilon^2}.$$