# Formulation of the learning problem

Maxim Raginsky

September 4, 2013

Now that we have seen an informal statement of the learning problem, as well as acquired some technical tools in the form of concentration inequalities, we can proceed to define the learning problem formally. Recall that the basic goal is to be able to predict some random variable $Y$ of interest from a correlated random observation $X$, where the predictor is to be constructed on the basis of $n$ i.i.d. training samples $(X_1, Y_1), \dots, (X_n, Y_n)$ from the joint distribution of $(X, Y)$. We will start by looking at an idealized scenario (often called the *realizable case* in the literature), in which $Y$ is a *deterministic* function of $X$, and we happen to know the function class to which it belongs. This simple set-up will let us pose, in a clean form, the basic requirements a learning algorithm should satisfy. Once we are done with the realizable case, we can move on to the general setting, in which the relationship between $X$ and $Y$ is probabilistic and not known precisely. This is often referred to as the *model-free* or *agnostic* case.

This order of presentation is, essentially, historical. The first statement of the learning problem is hard to trace precisely, but the "modern" algorithmic formalization seems to originate with the 1984 work of Valiant [Val84] on learning Boolean formulae. Valiant has focused on *computationally efficient* learning algorithms. The agnostic (or model-free) formulation was first proposed and studied by Haussler [Hau92] in 1992.

In this lecture, I will be closely following the excellent exposition of Vidyasagar [Vid03, Ch. 3].

## 1 The realizable case

We start with an idealized scenario, now often referred to in the literature as the *realizable case*. The basic set-up is as follows. We have a set $\mathsf{X}$ (often called the *feature space* or *input space*) and a family $\mathscr{P}$ of probability distributions on $\mathsf{X}$. We obtain an i.i.d. sample $X^n = (X_1, \dots, X_n)$ drawn according to some $P \in \mathscr{P}$, which we do not know (although it may very well be the case that $\mathscr{P}$ is a singleton, $|\mathscr{P}| = 1$, in which case we, of course, *do* know $P$). We will look at two basic problems:

1. *Concept learning:* There is a class $\mathscr{C}$ of subsets of $\mathsf{X}$, called the *concept class*, and an unknown *target concept* $C^* \in \mathscr{C}$ is picked by Nature. For each feature $X_i$ in our sample $X^n$, we receive a binary *label* $Y_i = \mathbf{1}_{\{X_i \in C^*\}}$. The $n$ feature-label pairs form the *training set*

$$(X_1, Y_1) = (X_1, \mathbf{1}_{\{X_1 \in C^*\}}), \dots, (X_n, Y_n) = (X_n, \mathbf{1}_{\{X_n \in C^*\}}). \tag{1}$$

The objective is to approximate the target concept $C^*$ as accurately as possible.

2. *Function learning:* There is a class $\mathscr{F}$ of functions $f : \mathsf{X} \to [0,1]$, and an unknown *target function* $f^* \in \mathscr{F}$ is picked by nature. For each input point $X_i$ in the sample $X^n$, we receive a real-valued *output* $Y_i = f^*(X_i)$. The $n$ input-output pairs

$$(X_1, Y_1) = (X_1, f^*(X_1)), \dots, (X_n, f^*(X_n)). \tag{2}$$

The objective is to approximate the target function $f^*$ as accurately as possible. (**Note:** the requirement that $f$ map $\mathsf{X}$ into $[0,1]$ is imposed primarily for technical convenience; using appropriate moment and/or tail behavior assumptions on $P$, it is possible to remove this requirement, but the resulting proofs will be somewhat laborious.)

We will now consider these two problems separately.

## 1.1 Concept learning

As we already stated, the goal of concept learning is to approximate the target concept $C^*$ as accurately as possible on the basis of the training data (1). This is done by means of a *learning algorithm.* An algorithm of this sort should be capable of producing an approximation to $C^*$ given the training set of the form (1) of any size $n$. More precisely:

**Definition 1.** *A* concept learning problem *is specified by a triple* $(\mathsf{X}, \mathscr{P}, \mathscr{C})$*, where* $\mathsf{X}$ *is the feature space,* $\mathscr{P}$ *is a family of probability distributions on* $\mathsf{X}$*, and* $\mathscr{C}$ *is a concept class. A* learning algorithm *for* $(\mathsf{X}, \mathscr{P}, \mathscr{C})$ *is a sequence* $\mathscr{A} = \{A_n\}_{n=1}^\infty$ *of mappings*

$$A_n : (\mathsf{X} \times \{0,1\})^n \to \mathscr{C}.$$

If $\mathscr{P}$ consists of only one distribution $P$, then the mappings $A_n$ may depend on $P$; otherwise, they may only depend on $\mathscr{P}$ as a whole. The idea behind the above definition is that for each training set size $n$ we have a definite procedure for forming an approximation to the unknown target concept $C^*$ on the basis of the training set of that size.

For brevity, let us denote by $Z_i$ the $i$th training pair $(X_i, Y_i) = (X_i, \mathbf{1}_{\{X_i \in C^*\}})$, and let us denote by $\mathsf{Z}$ the set $\mathsf{X} \times \{0,1\}$. Given a training set $Z^n = (Z_1, \dots, Z_n) \in \mathsf{Z}^n$ and a learning algorithm $\mathscr{A}$, the approximation to $C^*$ is

$$\widehat{C}_n = A_n(Z^n) = A_n(Z_1, \dots, Z_n) = A_n\big((X_1, \mathbf{1}_{\{X_1 \in C^*\}}, \dots, (X_n, \mathbf{1}_{\{X_n \in C^*\}})\big).$$

Note that $\widehat{C}_n$ is an element of the concept class $\mathscr{C}$ (by definition), and that it is a random variable since it depends on the random sample $Z^n$. It is often referred to as a *hypothesis* output by the learning algorithm $\mathscr{A}$.

How shall we measure the goodness of this approximation $\widehat{C}_n$? A natural thing to do is the following. Suppose now we draw a fresh feature $X$ from the same distribution $P \in \mathscr{P}$ as the one that has generated the training feature set $X^n$ and venture a *hypothesis* that $X$ belongs to the target concept $C^*$ if $X \in \widehat{C}_n$, i.e., if $\mathbf{1}_{\{X \in \widehat{C}_n\}} = 1$. When would we make a mistake, i.e., *misclassify* $X$? There are two mutually exclusive cases:

1. $X$ is in $C^*$, but not in $\widehat{C}_n$, i.e., $X \in C^* \cap \widehat{C}_n^c$, where $\widehat{C}_n^c = \mathsf{X} \backslash \widehat{C}_n$ is the complement of $\widehat{C}_n$ in $\mathsf{X}$.

2. $X$ is not in $C^*$, but it is in $\widehat{C}_n$, i.e., $X \in (C^*)^c \cap \widehat{C}_n$.

Thus, we will misclassify $X$ precisely when it happens to lie in the *symmetric difference*

$$C^* \triangle \widehat{C}_n \triangleq (C^* \cap \widehat{C}_n^c) \cup ((C^*)^c \cap \widehat{C}_n).$$

This will happen with probability $P(C^* \triangle \widehat{C}_n)$ — note, by the way, that this is a random number since $\widehat{C}_n$ depends on the training data $Z^n$. At any rate, we take the $P$-probability of the symmetric difference $C^* \triangle \widehat{C}_n$ as our measure of performance of $\mathscr{A}$. In order to streamline the notation, let us define the *risk* (or *loss*) of any $C \in \mathscr{C}$ w.r.t. $C^*$ and $P$ as

$$L_P(C, C^*) \triangleq P(C \triangle C^*) = P(X \in C \triangle C^*).$$

**Exercise 1.** *Prove that*

$$L_P(C, C^*) = \int_{\mathsf{X}} \left| \mathbf{1}_{\{x \in C\}} - \mathbf{1}_{\{x \in C^*\}} \right|^2 P(dx).$$

*In other words, $L_P(C, C^*)$ is the squared $L^2(P)$ norm of the difference of the indicator functions $I_C(\cdot) = \mathbf{1}_{\{\cdot \in C\}}$ and $I_{C^*}(\cdot) = \mathbf{1}_{\{\cdot \in C^*\}}$, $L_P(C, C^*) = \| I_C - I_{C^*} \|^2_{L^2(P)}$.*

Roughly speaking, we will say that $\mathscr{A}$ is a good algorithm if

$$L_P(\widehat{C}_n, C^*) \to 0 \qquad \text{as } n \to \infty \tag{3}$$

for any $P \in \mathscr{P}$ and any $C^* \in \mathscr{C}$. Since $\widehat{C}_n$ is a random element of $\mathscr{C}$, the convergence in (3) can only be in some probabilistic sense. In order to make things precise, let us define the following two quantities:

$$\begin{aligned} r(n, \varepsilon, P) &\triangleq \sup_{C \in \mathscr{C}} P^n \left( X^n \in \mathsf{X}^n : L_P(\widehat{C}_n, C) \ge \varepsilon \right) \\ \bar{r}(n, \varepsilon, \mathscr{P}) &\triangleq \sup_{P \in \mathscr{P}} r(n, \varepsilon, P) \end{aligned}$$

where $P^n$ denotes the $n$-fold product of $P$. For a fixed $P$ (which amounts to assuming that the features $X^n$ were drawn i.i.d. from $P$), $r(n, \varepsilon, P)$ quantifies the "size" of the set of "bad" samples, where we say that a sample $X^n$ is bad if it causes the learning algorithm to output a hypothesis $\widehat{C}_n$ whose risk is larger than $\varepsilon$. The quantity $\bar{r}(n, \varepsilon, \mathscr{P})$ accounts for the fact that we do not know which $P \in \mathscr{P}$ has generated the training feature points.

With all these things defined, we can now state the following:

**Definition 2.** *A learning algorithm $\mathscr{A} = \{A_n\}$ is* probably approximately correct *(or* PAC*)* to accurary $\varepsilon$ *if*

$$\lim_{n \to \infty} \bar{r}(n, \varepsilon, \mathscr{P}) = 0. \tag{4}$$

*We say that $\mathscr{A}$ is PAC if it is PAC to accurary $\varepsilon$ for every $\varepsilon > 0$. The concept class $\mathscr{C}$ is called* PAC learnable to accuracy $\varepsilon$ w.r.t. $\mathscr{P}$ *if there exists an algorithm that is PAC to accuracy $\varepsilon$. Finally, we say that $\mathscr{C}$ is* PAC learnable *if there exists an algorithm that is PAC.*

The term "probably approximately correct," which seems to have first been introduced by Angluin [Ang88], is motivated by the following observations. First, the hypothesis $\widehat{C}_n$ output by $\mathscr{A}$ for some $n$ is only an *approximation* to the target concept $C^*$. Thus, $L_P(\widehat{C}_n, C^*)$ will be, in general, nonzero. But if it is small, then we are justified in claiming that $\widehat{C}_n$ is *approximately correct.* Secondly, we may always encounter a bad sample, so $L_P(\widehat{C}_n, C^*)$ can be made small only *with high probability.* Thus, informally speaking, a PAC algorithm is one that "works reasonably well most of the time."

An equivalent way of phrasing the statement that a learning algorithm is PAC is as follows: For any $\varepsilon > 0$ and $\delta > 0$, there exists some $n(\varepsilon, \delta) \in \mathbb{N}$, such that

$$P^n\left(X^n \in \mathsf{X}^n : L_P(\widehat{C}_n, C) \ge \varepsilon\right) \le \delta, \qquad \forall n \ge n(\varepsilon,\delta), \forall C \in \mathscr{C}, \forall P \in \mathscr{P}. \tag{5}$$

In this context, $\varepsilon$ is called the *accuracy parameter*, while $\delta$ is called the *confidence parameter*. The meaning of this alternative characterization is as follows. If the sample size $n$ is at least $n(\varepsilon,\delta)$, then we can state with confidence at least $1-\delta$ that the hypothesis $\widehat{C}_n$ will correctly classify a fresh random point $X \in \mathsf{X}$ with probability at least $1-\varepsilon$.

The two problems of interest to us are:

1. Determine conditions under which a given concept class $\mathscr{C}$ is PAC learnable.

2. Obtain upper and lower bounds on $n(\varepsilon,\delta)$ as a function of $\varepsilon,\delta$. The following terminology is often used: the smallest number $n(\varepsilon,\delta)$ such that (5) holds is called the *sample complexity.*

## 1.2 Function learning

The goal of function learning is to construct an accurate approximation to an unknown target function $f^* \in \mathscr{F}$ on the basis of training data of the form (2). Analogously to the concept learning scenario, we have:

**Definition 3.** *A* function learning problem *is specified by a triple* $(\mathsf{X}, \mathscr{P}, \mathscr{F})$, *where* $\mathsf{X}$ *is the input space,* $\mathscr{P}$ *is a family of probability distributions on* $\mathsf{X}$, *and* $\mathscr{F}$ *is a class of functions* $f : \mathsf{X} \to [0,1]$. *A* learning algorithm *for* $(\mathsf{X}, \mathscr{P}, \mathscr{F})$ *is a sequence* $\mathscr{A} = \{A_n\}_{n=1}^{\infty}$ *of mappings*

$$A_n : (\mathsf{X} \times [0,1])^n \to \mathscr{F}.$$

As before, let us denote by $Z_i$ the input-output pair $(X_i, Y_i) = (X_i, f^*(X_i))$ and by $\mathsf{Z}$ the product set $\mathsf{X} \times [0,1]$. Given a training set $Z^n = (Z_1, \dots, Z_n) \in \mathsf{Z}^n$ and a learning algorithm $\mathscr{A}$, the approximation to $f^*$ is

$$\widehat{f}_n = A_n(Z^n) = A_n\left((X_1, f^*(X_1)), \dots, (X_n, f^*(X_n))\right).$$

As in the concept learning setting, $\widehat{f}_n$ is a *random element* of the function class $\mathscr{F}$.

In order to measure the performance of $\mathscr{A}$, we again imagine drawing a fresh input point $X \in \mathsf{X}$ from the same distribution $P \in \mathscr{P}$ that has generated the training inputs $X^n$. A natural error metric is the squared loss $|\widehat{f}_n(X)-f^*(X)|^2$. As before, we can define the *risk* (or *loss*) of any $f \in \mathscr{F}$ w.r.t. $f^*$ and $P$ as

$$L_P(f,f^*) \triangleq \mathbb{E}_P|f(X)-f^*(X)|^2 = \|f-f^*\|^2_{L^2(P)} = \int_{\mathsf{X}} |f(x)-f^*(x)|^2 P(dx). \tag{6}$$

Thus, the quantity of interest is the risk of $\widehat{f}_n$:

$$L_P(\widehat{f}_n,f^*) = \int_{\mathsf{X}} |\widehat{f}_n(x)-f^*(x)|^2 P(dx).$$

Keep in mind that $L_P(\widehat{f}_n,f^*)$ is a random variable, as it depends on $\widehat{f}_n$, which in turn depends on the random sample $X^n \in \mathsf{X}^n$.

**Remark 1.** The concept learning problem is, in fact, a special case of the function learning problem. Indeed, fix a concept class $\mathscr{C}$ and consider the function class $\mathscr{F}$ consisting of the indicator functions of the sets in $\mathscr{C}$:

$$\mathscr{F} = \{I_C : C \in \mathscr{C}\}.$$

Then for any $f = I_C$ and $f^* = I_{C^*}$ we will have

$$L_P(f,f^*) = \|I_C - I_{C^*}\|^2_{L^2(P)} = P(C\triangle C^*),$$

which is the error metric we have defined for concept learning.

As before, given a function learning problem $(\mathsf{X},\mathscr{P},\mathscr{F})$ and an algorithm $\mathscr{A}$, we can define

$$\begin{aligned} r(n,\varepsilon,P) &\triangleq \sup_{f\in\mathscr{F}} P^n\left(X^n \in \mathsf{X}^n : L_P(\widehat{f}_n,f) \ge \varepsilon\right) \\ \bar{r}(n,\varepsilon,\mathscr{P}) &\triangleq \sup_{P\in\mathscr{P}} r(n,\varepsilon,P) \end{aligned}$$

for every $n \in \mathbb{N}$ and $\varepsilon > 0$.The meaning of these quantities is exactly parallel to the corresponding quantities in concept learning, and leads to the following definition:

**Definition 4.** *A learning algorithm $\mathscr{A} = \{A_n\}$ is* PAC to accuracy $\varepsilon$ *if*

$$\lim_{n\to\infty} \bar{r}(n,\varepsilon,\mathscr{P}) = 0,$$

*and PAC if it is PAC to accuracy $\varepsilon$ for all $\varepsilon > 0$. A function class $\mathscr{F} = \{f : \mathsf{X} \to [0,1]\}$ is* PAC-learnable (to accuracy $\varepsilon$) *w.r.t. $\mathscr{P}$ if there exists an algorithm $\mathscr{A}$ that is PAC for $(\mathsf{X},\mathscr{P},\mathscr{F})$ (to accuracy $\varepsilon$).*

An equivalent way of stating that $\mathscr{A}$ is PAC is that, for any $\varepsilon,\delta > 0$ there exists some $n(\varepsilon,\delta) \in \mathbb{N}$ such that

$$P^n\left(X^n \in \mathsf{X}^n : L_P(\widehat{f}_n,f) \ge \varepsilon\right) \le \delta, \qquad \forall n \ge n(\varepsilon,\delta), \forall f \in \mathscr{F}, \forall P \in \mathscr{P}.$$

The smallest $n(\varepsilon,\delta) \in \mathbb{N}$ for which the above inequality holds is termed the *sample complexity*.

# 2 The model-free case

The realizable setting we have focused on in the preceding section rests on certain assumptions, which are not always warranted:

- The assumption that the target concept $C^*$ belongs to $\mathscr{C}$ (or that the target function $f^*$ belongs to $\mathscr{F}$) means that we are trying to fit a hypothesis to data, which are *a priori* known to have been generated by some member of the model class defined by $\mathscr{C}$ (or by $\mathscr{F}$). However, in general we may not want to (or be able to) assume much about the data generation process, and instead would like to find the best fit to the data at hand using an element of some model class of our choice.

- The assumption that the training features (or inputs) are labelled noiselessly by $\mathbf{1}_{\{x \in C^*\}}$ (or by $f(x)$) rules out the possibility of noisy measurements or observations.

- Finally, even if the above assumption were true, we would not necessarily have *a priori* knowledge of the concept class $\mathscr{C}$ (or the function class $\mathscr{F}$) containing the target concept (or function). In that case, the best we could hope for is to pick our own model class and seek the best *approximation* to the unknown target concept (or function) among the elements of that class.

The *model-free learning problem* (also referred to as the *agnostic case*), introduced by Haussler [Hau92], takes a more general decision-theoretic approach and removes the above restrictions. It has the following ingredients:

- Sets $\mathsf{X}$, $\mathsf{Y}$, and $\mathsf{U}$

- A class $\mathscr{P}$ of probability distributions on $\mathsf{Z} \triangleq \mathsf{X} \times \mathsf{Y}$

- A class $\mathscr{F}$ of functions $f : \mathsf{X} \to \mathsf{U}$ (the *hypothesis space*)

- A *loss function* $\ell : \mathsf{Y} \times \mathsf{U} \to [0,1]$

The learning process takes place as follows. We obtain an i.i.d. sample $Z^n = (Z_1, \ldots, Z_n)$, where each $Z_i = (X_i, Y_i)$ is drawn from the same fixed but unknown $P \in \mathscr{P}$. A *learning algorithm* is a sequence $\mathscr{A} = \{A_n\}_{n=1}^\infty$ of mappings

$$A_n : \mathsf{Z}^n \to \mathscr{F}.$$

As before, let

$$\widehat{f}_n = A_n(Z^n) = A_n(Z_1, \ldots, Z_n) = A_n((X_1, Y_1), \ldots, (X_n, Y_n)).$$

This is the hypothesis emitted by the learning algorithm based on the *training data* $Z^n$. Note that, by definition, $\widehat{f}_n$ is a *random element* of the hypothesis space $\mathscr{F}$, and that it maps each point $x \in \mathsf{X}$ to a point $u = \widehat{f}_n(x) \in \mathsf{U}$. Following the same steps as in the realizable case, we evaluate the goodness of $\widehat{f}_n$ by its expected loss

$$L_P(\widehat{f}_n) \triangleq \mathbb{E}_P\left[\ell(Y, \widehat{f}_n(X)) \middle| Z^n\right] = \int_{\mathsf{X} \times \mathsf{Y}} \ell(y, \widehat{f}_n(x)) P(dx, dy),$$

where the expectation is w.r.t. a random couple $(X,Y) \in \mathsf{Z}$ drawn according to the same $P$ but independently of $Z^n$. Note that $L_P(\widehat{f}_n)$ is a random variable since so is $\widehat{f}_n$. In general, we can define the expected risk w.r.t. $P$ for every $f$ in our hypothesis space by

$$L_P(f) \triangleq \mathbb{E}_P[\ell(Y, f(X))] = \int_{\mathsf{X}\times\mathsf{Y}} \ell(y, f(x)) P(dx, dy)$$

as well as the *minimum risk*

$$L_P^*(\mathscr{F}) \triangleq \inf_{f \in \mathscr{F}} L_P(f).$$

Conceptually, $L_P^*(\mathscr{F})$ is the best *possible* performance of any hypothesis in $\mathscr{F}$ when the samples are drawn from $P$; similarly, $L_P(\widehat{f}_n)$ is the *actual* performance of the algorithm with access to a training sample of size $n$. It is clear from definitions that

$$0 \le L_P^*(\mathscr{F}) \le L_P(\widehat{f}_n) \le 1.$$

The goal of learning is to guarantee that $L_P(\widehat{f}_n)$ is as close as possible to $L_P^*(\mathscr{F})$, whatever the true $P \in \mathscr{P}$ happens to be. In order to speak about this quantitatively, we need to assess the probability of getting a "bad" sample. To that end, we define, similarly to what we have done earlier, the quantity

$$r(n,\varepsilon) \triangleq \sup_{P \in \mathscr{P}} P^n\left(Z^n \in \mathsf{Z}^n : L_P(\widehat{f}_n) \ge L_P^*(\mathscr{F}) + \varepsilon\right) \tag{7}$$

for every $\varepsilon > 0$. Thus, a sample $Z^n \sim P^n$ is declared to be "bad" if it leads to a hypothesis whose expected risk on an independent test point $(X,Y) \sim P$ is greater than the smallest possible loss $L_P^*(\mathscr{F})$ by at least $\varepsilon$. We have the following:

**Definition 5.** *We say that a learning algorithm for a problem* $(\mathsf{X},\mathsf{Y},\mathsf{U},\mathscr{P},\mathscr{F},\ell)$ *is* PAC to accuracy *$\varepsilon$ if*

$$\lim_{n\to\infty} r(n,\varepsilon) = 0.$$

*An algorithm that is PAC to accuracy $\varepsilon$ for every $\varepsilon > 0$ is said to be PAC. A learning problem specified by a tuple* $(\mathsf{X},\mathsf{Y},\mathsf{U},\mathscr{P},\mathscr{F},\ell)$ *is* model-free *(or* agnostically*)* learnable *(to accuracy $\varepsilon$) if there exists an algorithm for it which is PAC (to accuracy $\varepsilon$).*

Let us look at some examples.

## 2.1 Function learning in the realizable case

First we show that the model-free framework contains the realizable set-up as a special case. To see this, let $\mathsf{X}$ be an arbitrary space and let $\mathsf{Y} = \mathsf{U} = [0,1]$. Let $\mathscr{F}$ be a class of functions $f : \mathsf{X} \to [0,1]$. Let $\mathscr{P}_{\mathsf{X}}$ be a family of probability distributions $P_X$ on $\mathsf{X}$. To each $P_X$ and each $f \in \mathscr{F}$ associate a probability distribution $P_f$ on $\mathsf{X} \times \mathsf{Y}$ as follows: let $X \sim P_X$, and let the conditional distribution of $Y$ given $X = x$ be given by

$$P_{Y|X,f}(B|X=x) = \mathbf{1}_{\{f(x)\in B\}}$$

for all (measurable) sets $B \subseteq \mathsf{Y}$. The resulting joint distribution $P_{X,f}$ is then uniquely defined by its action on the "rectangles" $A \times B$, $A \subseteq \mathsf{X}$ and $B \subseteq \mathsf{Y}$:

$$P_{X,f}(A \times B) \triangleq \int_A P_{Y|X,f}(B|x) P_X(dx) = \int_A \mathbf{1}_{\{f(x)\in B\}} P_X(dx)$$

Finally, let $\mathscr{P} = \{P_{X,f} : f \in \mathscr{F}, P_X \in \mathscr{P}_\mathsf{X}\}$. Finally, let $\ell(y,u) \triangleq |y-u|^2$.

Now, fixing a probability distribution $P \in \mathscr{P}$ is equivalent to fixing some $P_X \in \mathscr{P}_\mathsf{X}$ and some $f \in \mathscr{F}$. A random element of $\mathsf{Z} = \mathsf{X} \times \mathsf{Y}$ drawn according to such a $P$ has the form $(X, f(X))$, where $X \sim P_X$. An i.i.d. sequence $(X_1, Y_1), \ldots, (X_n, Y_n)$ drawn according to $P$ therefore has the form

$$(X_1, f(X_1)), \ldots, (X_n, f(X_n)),$$

which is precisely what we had in our discussion of function learning in the realizable case. Next, for any $P = P_{X,f} \in \mathscr{P}$ and any other $g \in \mathscr{F}$, we have

$$\begin{aligned} L_{P_{X,f}}(g) &= \int_{\mathsf{X}\times\mathsf{Y}} |y - g(x)|^2 P_{X,f}(dx, dy) \\ &= \int_{\mathsf{X}\times\mathsf{Y}} \mathbf{1}_{\{y=f(x)\}} |y - g(x)|^2 P_X(dx) \\ &= \int_{\mathsf{X}} |f(x) - g(x)|^2 P_X(dx) \\ &= \|f-g\|^2_{L^2(P_X)}, \end{aligned}$$

which is precisely the risk $L_{P_X}(g,f)$ defined in (6). Moreover,

$$L^*_{P_{X,f}} = \inf_{g\in\mathscr{F}} L_{P_{X,f}}(g) = \inf_{g\in\mathscr{F}} \|f-g\|^2_{L^2(P_X)} \equiv 0.$$

Therefore,

$$\begin{aligned} r(n,\varepsilon) &= \sup_{P_{X,f}\in\mathscr{P}} P^n_{X,f}\left(Z^n \in \mathsf{Z}^n : L_{P_{X,f}}(\widehat{f}_n) \ge L^*_{P_{X,f}} + \varepsilon\right) \\ &= \sup_{P_X \in \mathscr{P}_\mathsf{X}} \sup_{f\in\mathscr{F}} P^n_X\left(X^n \in \mathsf{X}^n : L_P(\widehat{f}_n, f) \ge \varepsilon\right) \\ &\equiv \bar{r}(n,\varepsilon,\mathscr{P}_\mathsf{X}). \end{aligned}$$

Thus, the function learning problem in the realizable case can be covered under the model-free framework as well.

## 2.2 Learning to classify with noisy labels

Consider the concept learning problem in the realizable case, except that now the labels $Y_i$, which in the original problem had the form $\mathbf{1}_{\{X_i \in C^*\}}$ for some target concept $C^*$, are *noisy*. That is, if $X_i$ is a training feature point, then the label $Y_i$ may be "flipped" due to chance, independently of all other $X_j$'s, $j \neq i$.

The precise formulation of this problem is as follows. Let $\mathsf{X}$ be a given feature space, let $\mathscr{C}$ be a concept class on it, and let $\mathscr{P}_{\mathsf{X}}$ be a class of probability distributions on $\mathsf{X}$. Suppose that Nature picks some distribution $P_X \in \mathscr{P}_{\mathsf{X}}$ of the features and some target concept $C^* \in \mathscr{C}$. The training data are generated as follows. First, an i.i.d. sample $X^n = (X_1, \ldots, X_n)$ is drawn according to some $P_X \in \mathscr{P}_{\mathsf{X}}$. Then the corresponding labels $Y_1, \ldots, Y_n \in \{0,1\}$ are generated as follows:

$$Y_i = \begin{cases} \mathbf{1}_{\{X_i \in C^*\}}, & \text{with probability } 1-\eta \\ 1 - \mathbf{1}_{\{X_i \in C^*\}}, & \text{with probability } \eta \end{cases} \qquad \text{independently of } X^n, \{Y_j\}_{j \neq i}$$

where $\eta < 1/2$ is the *classification noise rate*.

To cast this problem into the model-free framework, let $\mathsf{Y} = \mathsf{U} = \{0,1\}$, let $\mathscr{F} = \{I_C : C \in \mathscr{C}\}$, and let $\ell(y,u) = |y-u|^2$. Define a class $\mathscr{P}$ of probability distributions $\{P_{X,C} : P_X \in \mathscr{P}_{\mathsf{X}}, C \in \mathscr{C}\}$ on $\mathsf{X} \times \mathsf{Y} = \mathsf{X} \times \{0,1\}$ as follows. Let $X \sim P_X$, and for a given $C \in \mathscr{C}$ consider the conditional probability of $Y = 1$ given $X = x$. If $x \in C$, then $Y = 1$ if and only if there was no error in the label; on the other hand, if $x \not\in C$, then $Y = 1$ if and only if there was an error. That is,

$$\begin{aligned} P_{Y|X,C}(1|X=x) &= (1-\eta)\mathbf{1}_{\{x \in C\}} + \eta \mathbf{1}_{\{x \in C^c\}} \\ &= (1-\eta)\mathbf{1}_{\{x \in C\}} + \eta(1 - \mathbf{1}_{\{x \in C\}}); \\ P_{Y|X,C}(0|X=x) &= 1 - P_{Y|X,C}(1|X=x) \\ &= \eta \mathbf{1}_{\{x \in C\}} + (1-\eta)(1 - \mathbf{1}_{\{x \in C\}}). \end{aligned}$$

Then for any measurable set $A \subseteq \mathsf{X}$ we will have

$$\begin{aligned} P_{X,C}(A \times \{1\}) &= \int_A P_{Y|X,C}(1|X=x) P_X(dx) \\ &= \int_A \left[(1-\eta)\mathbf{1}_{\{x \in C\}} + \eta(1 - \mathbf{1}_{\{x \in C\}})\right] P_X(dx) \\ &= (1-\eta)\int_A \mathbf{1}_{\{x \in C\}} P_X(dx) + \eta \int_A P_X(dx) - \eta \int_A \mathbf{1}_{\{x \in C\}} P_X(dx) \\ &= \eta P_X(A) + (1-2\eta) P_X(A \cap C) \end{aligned} \tag{8}$$

and similarly

$$P_{X,C}(A \times \{0\}) = (1-\eta)P_X(A) - (1-2\eta)P_X(A \cap C). \tag{9}$$

Given a hypothesis $f = I_{C'} \in \mathscr{F}$, we have

$$L_{P_{X,C}}(I_{C'}) = \int_{\mathsf{X} \times \mathsf{Y}} |y - I_{C'}(x)|^2 P_{X,C}(dx, dy).$$

Computing this integral is straightforward but tedious. We start by expanding it as follows:

$$\begin{aligned} &\int_{\mathsf{X} \times \mathsf{Y}} |y - I_{C'}(x)|^2 P_{X,C}(dx, dy) \\ &= \int_{\mathsf{X}} |0 - I_{C'}(x)|^2 P_{X,C}(dx \times \{0\}) + \int_{\mathsf{X}} |1 - I_{C'}(x)|^2 P_{X,C}(dx \times \{1\}) \\ &= \int_{\mathsf{X}} \mathbf{1}_{\{x \in C'\}} P_{X,C}(dx \times \{0\}) + \int_{\mathsf{X}} \mathbf{1}_{\{x \in (C')^c\}} P_{X,C}(dx \times \{1\}) \\ &= P_{X,C}(C' \times \{0\}) + P_{X,C}((C')^c \times \{1\}). \end{aligned}$$

Substituting the expressions (8) and (9) into the above, we get

$$\begin{aligned}
L_{P_{X,C}}(I_{C'}) &= (1-\eta)P_X(C') - (1-2\eta)P_X(C\cap C') + \eta P_X((C')^c) + (1-2\eta)P_X(C\cap (C')^c) \\
&= (1-\eta)(\underbrace{P_X(C\cap C') + P_X(C^c\cap C')}_{P_X(C')}) - (1-2\eta)P_X(C\cap C') \\
&\qquad + \eta(\underbrace{P_X(C\cap (C')^c) + P_X(C^c\cap (C')^c)}_{P_X((C')^c)}) + (1-2\eta)P_X(C\cap (C')^c) \\
&= (1-\eta)(\underbrace{P_X(C\cap (C')^c) + P_X(C^c\cap C')}_{P_X(C\triangle C')}) + \eta P_X(C\cap C') + \eta\underbrace{P(C^c\cap (C')^c)}_{P_X((C\cup C')^c)} \\
&= (1-\eta)P_X(C\triangle C') + \eta P_X(C\cap C') + \eta(1-P_X(C\cup C')) \\
&= (1-\eta)P_X(C\triangle C') + \eta - \eta(\underbrace{P_X(C\cup C') - P_X(C\cap C')}_{P_X(C\triangle C')}) \\
&= \eta + (1-2\eta)P_X(C\triangle C') \\
&\equiv \eta + (1-2\eta)L_{P_X}(C',C).
\end{aligned}$$

From this, we have

$$\begin{aligned}
L^*_{P_{X,C}}(\mathscr{F}) &= \inf_{C'\in\mathscr{C}} L_{P_{X,C}}(I_{C'}) \\
&= \eta + (1-2\eta)\inf_{C'\in\mathscr{C}} P_X(C\triangle C') \\
&= \eta,
\end{aligned}$$

where the infimum is achieved by letting $C' = C$. From this it follows that

$$L_{P_{X,C}}(C') \ge L^*_{P_{X,C}} + \varepsilon \qquad \Longleftrightarrow \qquad P_{X,C}(C\triangle C') \ge \frac{\varepsilon}{1-2\eta}$$

In other words, learning a concept to accuracy $\varepsilon$ with noise rate $\eta$ is equivalent to learning a concept to accuracy $\varepsilon/(1-2\eta)$ in the noise-free case:

$$r(n,\varepsilon) = \bar{r}\left(n, \frac{\varepsilon}{1-2\eta}, \mathscr{P}_{\mathsf{X}}\right).$$

# 3 Empirical risk minimization

Having formulated the model-free learning problem, we must now turn to the question of how to construct PAC learning algorithms (and the related question of when a hypothesis class is PAC-learnable in the model-free setting).

We will first start with a heuristic argument and then make it rigorous. Suppose we are faced with the learning problem specified by $(\mathsf{X},\mathsf{Y},\mathsf{U},\mathscr{P},\mathscr{F},\ell)$. Given a training set $Z^n = (Z_1,\dots,Z_n)$, where each $Z_i = (X_i,Y_i)$ is independently drawn according to some unknown $P\in\mathscr{P}$, what should we do? The first thing to note is that, for any hypothesis $f \in \mathscr{F}$, we can approximate its risk $L_P(f)$ by the *empirical risk*

$$\frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, f(X_i)), \tag{10}$$

whose expectation w.r.t. the distribution of $Z^n$ is clearly equal to $L_P(f)$. In fact, since $\ell$ is bounded between 0 and 1, Hoeffding's inequality tells us that

$$\left|\frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, f(X_i)) - L_P(f)\right| < \varepsilon \qquad \text{with probability at least } 1 - 2e^{-2n\varepsilon^2}.$$

We can express these statements more succinctly if we define, for each $f \in \mathscr{F}$, the function $\ell_f : \mathsf{Z} \to [0,1]$ by

$$\ell_f(z) \equiv \ell_f(x, y) \triangleq \ell(y, f(x)). \tag{11}$$

Then the empirical risk (10) is just the expectation of $\ell_f$ w.r.t. the empirical distribution $\widehat{P}_{Z^n}$:

$$\widehat{P}_{Z^n}(\ell_f) = \frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, f(X_i)),$$

and, since $L_P(f) = \mathbb{E}_P[\ell(Y, f(X))] = P(\ell_f)$, we will have

$$\left|\widehat{P}_{Z^n}(\ell_f) - P(\ell_f)\right| < \varepsilon \qquad \text{with probability at least } 1 - 2e^{-2n\varepsilon^2}. \tag{12}$$

Now, given the data $Z^n$ we can compute the empirical risks $\widehat{P}_{Z^n}(\ell_f)$ for every $f$ in our hypothesis class $\mathscr{F}$. Since (12) holds for each $f \in \mathscr{F}$ individually, we may intuitively claim that the empirical risk for each $f$ is a sufficiently accurate estimator of the corresponding true risk $L_P(f) \equiv P(\ell_f)$. Thus, a reasonable learning strategy would be to find any $\widehat{f}_n \in \mathscr{F}$ that would *minimize* the empirical risk, i.e., take

$$\widehat{f}_n = \operatorname*{arg\,min}_{f \in \mathscr{F}} \widehat{P}_{Z^n}(\ell_f) = \operatorname*{arg\,min}_{f \in \mathscr{F}} \frac{1}{n}\sum_{i=1}^{n} \ell(Y_i, f(X_i)). \tag{13}$$

The reason why we would expect something like (13) to work is as follows: if a given $f^*$ is a minimizer of $L_P(f) = P(\ell_f)$ over $\mathscr{F}$,

$$f^* = \operatorname*{arg\,min}_{f \in \mathscr{F}} P(\ell_f),$$

then its empirical risk, $\widehat{P}_{Z^n}(f^*)$, will be close to $L_P(f^*) = P(\ell_{f^*}) = L_P^*(\mathscr{F})$ with high probability. Moreover, it makes sense to expect that, in some sense, $\widehat{f}_n$ defined in (13) would be "close" to $f^*$, resulting in something like

$$P(\widehat{f}_n) \approx \widehat{P}_{Z^n}(\widehat{f}_n) \approx \widehat{P}_{Z^n}(f^*) \approx P(f^*)$$

with high probability.

Unfortunately, this is not true in general. However, as we will now see, it is true under certain regularity conditions on the objects $\mathscr{P}$, $\mathscr{F}$, and $\ell$. In order to state these regularity conditions precisely, let us define the *induced loss function class*

$$\mathscr{L}_{\mathscr{F}} \triangleq \{\ell_f : f \in \mathscr{F}\}.$$

Each $\ell_f \in \mathscr{L}_{\mathscr{F}}$ corresponds to the hypothesis $f \in \mathscr{F}$ via (11). Now, for any $n \in \mathbb{N}$ and any $\varepsilon > 0$ let us define

$$q(n,\varepsilon) \triangleq \sup_{P \in \mathscr{P}} P^n\left(Z^n \in \mathsf{Z}^n : \sup_{f \in \mathscr{F}} \left|\widehat{P}_{Z^n}(\ell_f) - P(\ell_f)\right| \geq \varepsilon\right). \tag{14}$$

For a fixed $P \in \mathscr{P}$, quantity $\sup_{f \in \mathscr{F}} \left|\widehat{P}_{Z^n}(\ell_f) - P(\ell_f)\right|$ is the *worst-case deviation* between the empirical means $\widehat{P}_{Z^n}(\ell_f)$ and their expectations $P(\ell_f)$ over the entire hypothesis class $\mathscr{F}$. Given $P$, we say that an i.i.d. sample $Z^n \in \mathsf{Z}^n$ is "bad" if there exists at least one $f \in \mathscr{F}$, for which

$$\left|\widehat{P}_{Z^n}(\ell_f) - P(\ell_f)\right| \geq \varepsilon.$$

Equivalently, a sample is bad if

$$\sup_{f \in \mathscr{F}} \left|\widehat{P}_{Z^n}(\ell_f) - P(\ell_f)\right| \geq \varepsilon.$$

The quantity $q(n,\varepsilon)$ then compensates for the fact that $P$ is unknown by considering the *worst case* over the entire class $\mathscr{P}$. With this in mind, we make the following definition:

**Definition 6.** *We say that the induced class $\mathscr{L}_{\mathscr{F}}$ has the* uniform convergence of empirical means (UCEM) *property w.r.t. $\mathscr{P}$ if*

$$\lim_{n \to \infty} q(n,\varepsilon) = 0$$

*for every $\varepsilon > 0$.*

**Theorem 1.** *If the induced class $\mathscr{L}_{\mathscr{F}}$ has the UCEM property, then the* empirical risk minimization (ERM) *algorithm of* (13) *is PAC.*

*Proof.* Fix $\varepsilon, \delta > 0$. We will now show that we can find a sufficiently large $n(\varepsilon,\delta)$, such that $r(n,\varepsilon) < \delta$ for all $n \geq n(\varepsilon,\delta)$, where $r(n,\varepsilon)$ is defined in (7).

Let $f^* \in \mathscr{F}$ minimize the true risk w.r.t. $P$, i.e., $P(f^*) = L^*_P(\mathscr{F})$. For any $n$, we have

$$\begin{aligned} L_P(\widehat{f}_n) - L^*_P &= P\big(\ell_{\widehat{f}_n}\big) - P\big(f^*\big) \\ &= \underbrace{P\big(\ell_{\widehat{f}_n}\big) - \widehat{P}_{Z^n}\big(\ell_{\widehat{f}_n}\big)}_{T_1} + \underbrace{\widehat{P}_{Z^n}\big(\ell_{\widehat{f}_n}\big) - \widehat{P}_{Z^n}\big(\ell_{f^*}\big)}_{T_2} + \underbrace{\widehat{P}_{Z^n}\big(\ell_{f^*}\big) - P\big(\ell_{f^*}\big)}_{T_3}, \end{aligned}$$

where in the second line we have added and subtracted $\widehat{P}_{Z^n}(\ell_{\widehat{f}_n})$ and $\widehat{P}_{Z^n}(\ell_{f^*})$. We will now analyze the behavior of the three terms, $T_1$, $T_2$, and $T_3$. Since $\widehat{f}_n$ minimizes the empirical risk $\widehat{P}_{Z^n}(\ell_f)$ over all $f \in \mathscr{F}$, we will have

$$T_2 = \widehat{P}_{Z^n}(\ell_{\widehat{f}_n}) - \widehat{P}_{Z^n}(\ell_{f^*}) \le 0.$$

Next,

$$T_1 = P(\ell_{\widehat{f}_n}) - \widehat{P}_{Z^n}(\ell_{\widehat{f}_n}) \le \sup_{f\in\mathscr{F}} \left[\widehat{P}_{Z^n}(\ell_f) - P(\ell_f)\right] \le \sup_{f\in\mathscr{F}} \left|\widehat{P}_{Z^n}(\ell_f) - P(\ell_f)\right|,$$

and the same upper bound holds for $T_3$. Hence,

$$L_P(\widehat{f}_n) - L^*_P(\mathscr{F}) \le 2 \sup_{f\in\mathscr{F}} \left|\widehat{P}_{Z^n}(\ell_f) - P(\ell_f)\right|. \tag{15}$$

Now, since $\mathscr{L}_{\mathscr{F}}$ has the UCEM property, we can find some sufficiently large $n_0(\varepsilon,\delta)$, such that

$$q(n,\varepsilon/2) = \sup_{P\in\mathscr{P}} P^n\left(Z^n \in \mathsf{Z}^n : \sup_{f\in\mathscr{F}} \left|\widehat{P}_{Z^n}(\ell_f) - P(\ell_f)\right| \ge \varepsilon/2\right) < \delta, \qquad \forall n \ge n_0(\varepsilon,\delta).$$

From this it follows that, for all $n \ge n_0(\varepsilon,\delta)$, we will have

$$P^n\left(Z^n : \sup_{f\in\mathscr{F}} \left|\widehat{P}_{Z^n}(\ell_f) - P(\ell_f)\right| \ge \varepsilon/2\right) < \delta, \qquad \forall P \in \mathscr{P}.$$

From (15), we see that

$$L_P(\widehat{f}_n) \ge L^*_P(\mathscr{F}) + \varepsilon \qquad \Longrightarrow \qquad \sup_{f\in\mathscr{F}} \left|\widehat{P}_{Z^n}(\ell_f) - P(\ell_f)\right| \ge \varepsilon/2$$

for all $n$. However, for all $n \ge n_0(\varepsilon,\delta)$ the latter event will occur with probability at most $\delta$, no matter which $P$ is in effect. Therefore, for all $n \ge n_0(\varepsilon,\delta)$ we will have

$$\begin{aligned} r(n,\varepsilon) &= \sup_{P\in\mathscr{P}} P^n\left(Z^n : L_P(\widehat{f}_n) \ge L^*_P(\mathscr{F}) + \varepsilon\right) \\ &\le \sup_{P\in\mathscr{P}} P^n\left(Z^n : \sup_{f\in\mathscr{F}} \left|\widehat{P}_{Z^n}(\ell_f) - P(\ell_f)\right| \ge \varepsilon/2\right) \\ &\equiv q(n,\varepsilon/2) \\ &< \delta, \end{aligned}$$

which is precisely what we wanted to show. Thus, $r(n,\varepsilon) \to 0$ as $n \to \infty$ for every $\varepsilon > 0$, which means that the ERM algorithm is PAC. □

This theorem shows that the UCEM property of the induced class $\mathscr{L}_{\mathscr{F}}$ is a sufficient condition for the ERM algorithm to be PAC. Now the whole affair rests on us being able to establish the UCEM property for various "interesting" and "useful" problem specifications. This will be our concern in the lectures ahead. However, let me give you a hint of what to expect. In many cases, we will be able to show that the induced class $\mathscr{L}_{\mathscr{F}}$ is so well-behaved that the bound

$$\mathbb{E}_{P^n}\left[\sup_{f\in\mathscr{F}}\left|\widehat{P}_{Z^n}(\ell_f)-P(\ell_f)\right|\right]\le\frac{C_{\mathscr{F},\ell}}{\sqrt{n}} \tag{16}$$

holds for *every* $P$, where $C_{\mathscr{F},\ell}>0$ is some constant that depends only on the characteristics of the hypothesis class $\mathscr{F}$ and the loss function $\ell$. Since $\ell_f$ is bounded between 0 and 1, the function

$$g(Z^n)\triangleq\sup_{f\in\mathscr{F}}\left|\widehat{P}_{Z^n}(\ell_f)-P(\ell_f)\right|$$

has bounded differences with constants $c_1=\ldots=c_n=1/n$. McDiarmid's inequality then tells us that, for any $t>0$,

$$P^n\Big(g(Z^n)-\mathbb{E}g(Z^n)\ge t\Big)\le e^{-2nt^2}. \tag{17}$$

Let

$$n_0(\varepsilon,\delta)\triangleq\max\left\{\frac{4C^2_{\mathscr{F},\ell}}{\varepsilon^2},\frac{2}{\varepsilon^2}\log\left(\frac{1}{\delta}\right)\right\}+1. \tag{18}$$

Then for any $n\ge n_0(\varepsilon,\delta)$

$$\begin{aligned}
P^n\Big(g(Z^n)\ge\varepsilon\Big)&=P^n\Big(g(Z^n)-\mathbb{E}g(Z^n)\ge\varepsilon-\mathbb{E}g(Z^n)\Big)\\
&\le P^n\left(g(Z^n)-\mathbb{E}g(Z^n)\ge\varepsilon-\frac{C_{\mathscr{F},\ell}}{\sqrt{n}}\right) && \text{because of (16)}\\
&\le P^n\left(g(Z^n)-\mathbb{E}g(Z^n)\ge\frac{\varepsilon}{2}\right) && \text{because } n>\frac{4C^2_{\mathscr{F},\ell}}{\varepsilon^2}\\
&\le e^{-n\varepsilon^2/2} && \text{using (17) with } t=\varepsilon/2\\
&<\delta && \text{because } n>\frac{2}{\varepsilon^2}\log\left(\frac{1}{\delta}\right)
\end{aligned}$$

for *any* probability distribution $P$ over $\mathsf{Z}=\mathsf{X}\times\mathsf{Y}$. Thus, we have derived a very important fact: If the induced loss class $\mathscr{L}_{\mathscr{F}}$ satisfies (16), then (a) it has the UCEM property, and consequently is model-free learnable using the ERM algorithm, and (b) the sample complexity is polynomial in $1/\varepsilon$ and logarithmic in $1/\delta$. Our next order of business will be to derive sufficient conditions on $\mathscr{F}$ and $\ell$ for something like (16) to hold.

# References

[Ang88] D. Angluin. Queries and concept learning. *Machine Learning*, 2:319–342, 1988.

[Hau92] D. Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Information and Computation*, 95:129–161, 1992.

[Val84] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[Vid03] M. Vidyasagar. *Learning and Generalization*. Springer, 2 edition, 2003.