1

The framework of learning

1.1 Introduction

We begin by describing some cases of learning, simplified to the extreme, to convey an intuition of what learning is.

Case 1.1 Among the most used instances of learning (although not necessarily with this name) is linear regression. This amounts to finding a straight line that best approximates a functional relationship presumed to be implicit in a set of data points in $\mathbb{R}^2$, $\{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$ (Figure 1.1). The yardstick used to measure how good an approximation a given line $Y = aX + b$ is, is called *least squares*. The best line is the one that minimizes

$$Q(a, b) = \sum_{i=1}^{m} (y_i - ax_i - b)^2.$$

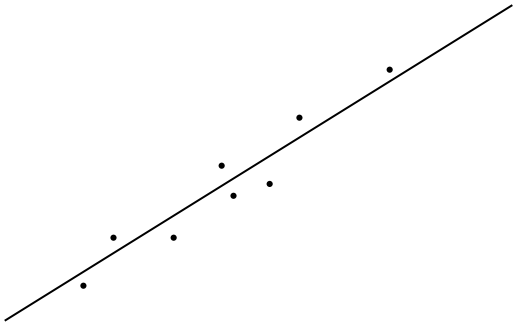

Figure 1.1

Case 1.2 Case 1.1 readily extends to a classical situation in science, namely, that of learning a physical law by curve fitting to data. Assume that the law at hand, an unknown function $f : \mathbb{R} \to \mathbb{R}$, has a specific form and that the space of all functions with this form can be parameterized by N real numbers. For instance, if f is assumed to be a polynomial of degree d, then $N = d+1$ and the parameters are the unknown coefficients $w_0, \ldots, w_d$ of f. In this case, finding the *best fit* by the *least squares method* estimates the unknown f from a set of pairs $\{(x_1, y_1), \ldots, (x_m, y_m)\}$. If the measurements generating this set were exact, then y_i would be equal to $f(x_i)$. However, in general one expects the values y_i to be affected by noise. That is, $y_i = f(x_i) + \varepsilon$, where ε is a random variable (which may depend on x_i) with mean zero. One then computes the vector of coefficients w such that the value

$$\sum_{i=1}^{m} (f_w(x_i) - y_i)^2, \qquad \text{with } f_w(x) = \sum_{j=0}^{d} w_j x^j$$

is minimized, where, typically, $m > N$. In general, the minimum value above is not 0. To solve this minimization problem, one uses the least squares technique, a method going back to Gauss and Legendre that is computationally efficient and relies on numerical linear algebra.

Since the values y_i are affected by noise, one might take as starting point, instead of the unknown f, a family of probability measures ε_x on $\mathbb{R}$ varying with $x \in \mathbb{R}$. The only requirement on these measures is that for all $x \in \mathbb{R}$, the mean of ε_x is $f(x)$. Then y_i is randomly drawn from ε_{x_i}. In some contexts the x_i, rather than being chosen, are also generated by a probability measure ρ_X on $\mathbb{R}$. Thus, the starting point could even be a single measure ρ on $\mathbb{R} \times \mathbb{R}$ – capturing both the measure ρ_X and the measures ε_x for $x \in \mathbb{R}$ – from which the pairs (x_i, y_i) are randomly drawn.

A more general form of the functions in our approximating class could be given by

$$f_w(x) = \sum_{i=1}^{N} w_i \phi_i(x),$$

where the ϕ_i are the elements of a basis of a specific function space, not necessarily of polynomials.

Case 1.3 The training of neural networks is an extension of Case 1.2. Roughly speaking, a neural network is a directed graph containing some input nodes, some output nodes, and some intermediate nodes where certain functions are

computed. If X denotes the input space (whose elements are fed to the input nodes) and Y the output space (of possible elements returned by the output nodes), a neural network computes a function from X to Y. The literature on neural networks shows a variety of choices for X and Y, which can be continuous or discrete, as well as for the functions computed at the intermediate nodes. A common feature of all neural nets, though, is the dependence of these functions on a set of parameters, usually called *weights*, $w = \{w_j\}_{j\in J}$. This set determines the function $f_w : X \to Y$ computed by the network.

Neural networks are *trained* to learn functions. As in Case 1.2, there is a target function $f : X \to Y$, and the network is given a set of randomly chosen pairs $(x_1, y_1), \ldots, (x_m, y_m)$ in $X \times Y$. Then, training algorithms select a set of weights w attempting to minimize some distance from f_w to the target function $f : X \to Y$.

Case 1.4 A standard example of pattern recognition involves handwritten characters. Consider the problem of classifying handwritten letters of the English alphabet. Here, elements in our space X could be matrices with entries in the interval $[0, 1]$ – each entry representing a pixel in a certain gray scale of a digitized photograph of the handwritten letter or some features extracted from the letter. We may take Y to be

$$Y = \left\{ y \in \mathbb{R}^{26} \mid y = \sum_{i=1}^{26} \lambda_i e_i \text{ such that } \sum_{i=1}^{26} \lambda_i = 1 \right\}.$$

Here e_i is the ith coordinate vector in $\mathbb{R}^{26}$, each coordinate corresponding to a letter. If $\Delta \subset Y$ is the set of points y as above such that $0 \le \lambda_i \le 1$, for $i = 1, \ldots, 26$, one can interpret a point in Δ as a probability measure on the set $\{\mathtt{A}, \mathtt{B}, \mathtt{C}, \ldots, \mathtt{X}, \mathtt{Y}, \mathtt{Z}\}$. The problem is to learn the ideal function $f : X \to Y$ that associates, to a given handwritten letter x, a linear combination of the e_i with coefficients $\{\text{Prob}\{x = \mathtt{A}\}, \text{Prob}\{x = \mathtt{B}\}, \ldots, \text{Prob}\{x = \mathtt{Z}\}\}$. Unambiguous letters are mapped into a coordinate vector, and in the (pure) classification problem f takes values on these e_i. "Learning f" means finding a sufficiently good approximation of f within a given prescribed class.

The approximation of f is constructed from a set of samples of handwritten letters, each of them with a label in Y. The set $\{(x_1, y_1), \ldots, (x_m, y_m)\}$ of these m samples is randomly drawn from $X \times Y$ according to a measure ρ on $X \times Y$. This measure satisfies $\rho(X \times \Delta) = 1$. In addition, in practice, it is concentrated around the set of pairs (x, y) with $y = e_i$ for some $1 \le i \le 26$. That is, the occurring elements $x \in X$ are handwritten letters and not, say, a digitized image of the *Mona Lisa*. The function f to be learned is the regression function f_ρ of ρ.

That is, $f_\rho(x)$ is the average of the y values of $\{x\} \times Y$ (we are more precise about ρ and the regression function in Section 1.2).

Case 1.5 A standard approach for approximating characteristic (or indicator) functions of sets is known as *PAC learning* (from "probably approximately correct"). Let T (the *target concept*) be a subset of $\mathbb{R}^n$ and ρ_X be a probability measure on $\mathbb{R}^n$ that we assume is not known in advance. Intuitively, a set $S \subset \mathbb{R}^n$ approximates T when the symmetric difference $S \Delta T = (S \setminus T) \cup (T \setminus S)$ is small, that is, has a small measure. Note that if f_S and f_T denote the characteristic functions of S and T, respectively, this measure, called the *error of* S, is $\int_{\mathbb{R}^n} |f_S - f_T| \, d\rho_X$. Note that since the functions take values in $\{0, 1\}$, only this integral coincides with $\int_{\mathbb{R}^n} (f_S - f_T)^2 \, d\rho_X$.

Let $\mathcal{C}$ be a class of subsets of $\mathbb{R}^n$ and assume that $T \in \mathcal{C}$. One strategy for constructing an approximation of T in $\mathcal{C}$ is the following. First, draw points $x_1, \ldots, x_m \in \mathbb{R}^n$ according to ρ_X and label each of them with 1 or 0 according to whether they belong to T. Second, compute any function $f_S : \mathbb{R}^n \to \{0, 1\}$, $f_S \in \mathcal{C}$, that coincides with this labeling over $\{x_1, \ldots, x_m\}$. Such a function will provide a good approximation S of T (small error with respect to ρ_X) as long as m is large enough and $\mathcal{C}$ is not too wild. Thus the measure ρ_X is used in both capacities, governing the sample drawing and measuring the error set $S \Delta T$.

A major goal in PAC learning is to estimate how large m needs to be to obtain an ε approximation of T with probability at least $1 - \delta$ as a function of ε and δ.

The situation described above is noise free since each randomly drawn point $x_i \in \mathbb{R}^n$ is correctly labeled. Extensions of PAC learning allowing for labeling mistakes with small probability exist.

Case 1.6 (Monte Carlo integration) An early instance of randomization in algorithmics appeared in numerical integration. Let $f : [0, 1]^n \to \mathbb{R}$. One way of approximating the integral $\int_{x \in [0,1]^n} f(x) \, dx$ consists of randomly drawing points $x_1, \ldots, x_m \in [0, 1]^n$ and computing

$$I_m(f) = \frac{1}{m} \sum_{i=1}^{m} f(x_i).$$

Under mild conditions on the regularity of f, $I_m(f) \to \int f$ with probability 1; that is, for all $\varepsilon > 0$,

$$\lim_{m \to \infty} \operatorname*{Prob}_{x_1, \ldots, x_m} \left\{ \left| I_m(f) - \int_{x \in [0,1]^n} f(x) \, dx \right| > \varepsilon \right\} \to 0.$$

Again we find the theme of learning an object (here a single real number, although defined in a nontrivial way through f) from a sample. In this case

the measure governing the sample is known (the measure in $[0,1]^n$ inherited from the standard Lebesgue measure on $\mathbb{R}^n$), but the same idea can be used for an unknown measure. If ρ_X is a probability measure on $X \subset \mathbb{R}^n$, a domain or manifold, $I_m(f)$ will approximate $\int_{x\in X} f(x)\, d\rho_X$ for large m with high probability as long as the points $x_1, \ldots, x_m$ are drawn from X according to the measure ρ_X. Note that no noise is involved here. An extension of this idea to include noise is, however, possible.

A common characteristic of Cases 1.2–1.5 is the existence of both an "unknown" function $f : X \to Y$ and a probability measure allowing one to randomly draw points in $X \times Y$. That measure can be on X (Case 1.5), on Y varying with $x \in X$ (Cases 1.2 and 1.3), or on the product $X \times Y$ (Case 1.4). The only requirement it satisfies is that, if for $x \in X$ a point $y \in Y$ can be randomly drawn, then the expected value of y is $f(x)$. That is, the noise is centered at zero. Case 1.6 does not follow this pattern. However, we have included it since it is a well-known algorithm and shares the flavor of learning an unknown object from random data.

The development in this book, for reasons of unity and generality, is based on a single measure on $X \times Y$. However, one should keep in mind the distinction between "inputs" $x \in X$ and "outputs" $y \in Y$.

1.2 A formal setting

Since we want to study learning from random sampling, the primary object in our development is a probability measure ρ governing the sampling that is not known in advance.

Let X be a compact metric space (e.g., a domain or a manifold in Euclidean space) and $Y = \mathbb{R}^k$. For convenience we will take $k = 1$ for the time being. Let ρ be a Borel probability measure on $Z = X \times Y$ whose regularity properties will be assumed as required. In the following we try to utilize concepts formed naturally and solely from X, Y, and ρ.

Throughout this book, if ξ is a random variable (i.e., a real-valued function on a probability space Z), we will use $\mathbf{E}(\xi)$ to denote the *expected value* (or average, or mean) of ξ and $\sigma^2(\xi)$ to denote its *variance*. Thus

$$\mathbf{E}(\xi) = \int_{z\in Z} \xi(z)\, dp \quad \text{and} \quad \sigma^2(\xi) = \mathbf{E}((\xi - \mathbf{E}(\xi))^2) = \mathbf{E}(\xi^2) - (\mathbf{E}(\xi))^2.$$

A central concept in the next few chapters is the *generalization error* (or *least squares error* or, if there is no risk of ambiguity, simply *error*) of f, for

$f : X \to Y$, defined by

$$\mathcal{E}(f) = \mathcal{E}_\rho(f) = \int_Z (f(x) - y)^2 \, d\rho.$$

For each input $x \in X$ and output $y \in Y$, $(f(x) - y)^2$ is the error incurred through the use of f as a model for the process producing y from x. This is a local error. By integrating over $X \times Y$ (w.r.t. ρ, of course) we average out this local error over all pairs (x, y). Hence the word "error" for $\mathcal{E}(f)$.

The problem posed is: *What is the f that minimizes the error $\mathcal{E}(f)$?* To answer this question we note that the error $\mathcal{E}(f)$ naturally decomposes as a sum. For every $x \in X$, let $\rho(y|x)$ be the conditional (w.r.t. x) probability measure on Y. Let also ρ_X be the marginal probability measure of ρ on X, that is, the measure on X defined by $\rho_X(S) = \rho(\pi^{-1}(S))$, where $\pi : X \times Y \to X$ is the projection. For every integrable function $\varphi : X \times Y \to \mathbb{R}$ a version of Fubini's theorem relates ρ, $\rho(y|x)$, and ρ_X as follows:

$$\int_{X \times Y} \varphi(x, y) \, d\rho = \int_X \left(\int_Y \varphi(x, y) \, d\rho(y|x) \right) d\rho_X.$$

This "breaking" of ρ into the measures $\rho(y|x)$ and ρ_X corresponds to looking at Z as a product of an input domain X and an output set Y. In what follows, unless otherwise specified, integrals are to be understood as being over ρ, $\rho(y|x)$ or ρ_X.

Define $f_\rho : X \to Y$ by

$$f_\rho(x) = \int_Y y \, d\rho(y|x).$$

The function f_ρ is called the *regression function* of ρ. For each $x \in X$, $f_\rho(x)$ is the average of the y coordinate of $\{x\} \times Y$ (in topological terms, the average of y on the fiber of x). Regularity hypotheses on ρ will induce regularity properties on f_ρ.

We will assume throughout this book that f_ρ is bounded.

Fix $x \in X$ and consider the function from Y to $\mathbb{R}$ mapping y into $(y - f_\rho(x))$. Since the expected value of this function is 0, its variance is

$$\sigma^2(x) = \int_Y (y - f_\rho(x))^2 \, d\rho(y|x).$$

Now average over X, to obtain

$$\sigma_\rho^2 = \int_X \sigma^2(x) \, d\rho_X = \mathcal{E}(f_\rho).$$

The number σ_ρ^2 is a measure of how well conditioned ρ is, analogous to the notion of condition number in numerical linear algebra.

Remark 1.7

(i) It is important to note that whereas ρ and f_ρ are generally "unknown," ρ_X is known in some situations and can even be the Lebesgue measure on X inherited from Euclidean space (as in Cases 1.2 and 1.6).

(ii) In the remainder of this book, if formulas do not make sense or ∞ appears, then the assertions where these formulas occur should be considered vacuous.

Proposition 1.8 *For every $f : X \to Y$,*

$$\mathcal{E}(f) = \int_X (f(x) - f_\rho(x))^2 \, d\rho_X + \sigma_\rho^2.$$

Proof From the definition of $f_\rho(x)$ for each $x \in X$, $\int_Y (f_\rho(x) - y) = 0$. Therefore,

$$\begin{aligned}
\mathcal{E}(f) &= \int_Z (f(x) - f_\rho(x) + f_\rho(x) - y)^2 \\
&= \int_X (f(x) - f_\rho(x))^2 + \int_X \int_Y (f_\rho(x) - y)^2 \\
&\quad + 2 \int_X \int_Y (f(x) - f_\rho(x))(f_\rho(x) - y) \\
&= \int_X (f(x) - f_\rho(x))^2 + \sigma_\rho^2 + 2 \int_X (f(x) - f_\rho(x)) \int_Y (f_\rho(x) - y) \\
&= \int_X (f(x) - f_\rho(x))^2 + \sigma_\rho^2.
\end{aligned}$$

■[1]

The first term on the right-hand side of Proposition 1.8 provides an average (over X) of the error suffered from the use of f as a model for f_ρ. In addition, since σ_ρ^2 is independent of f, Proposition 1.8 implies that f_ρ has the smallest possible error among all functions $f : X \to Y$. Thus σ_ρ^2 represents a lower bound on the error $\mathcal{E}$ and it is due solely to our primary object, the measure ρ. Thus, Proposition 1.8 supports the following statement:

> *The goal is to "learn" (i.e., to find a good approximation of) f_ρ from random samples on Z.*

[1] Throughout this book, the square ■ denotes the end of a proof or the fact that no proof is given.

We now consider sampling. Let

$$\mathbf{z} \in Z^m, \quad \mathbf{z} = ((x_1, y_1), \ldots, (x_m, y_m))$$

be a *sample* in Z^m, that is, m examples independently drawn according to ρ. Here Z^m denotes the m-fold Cartesian product of Z. We define the *empirical error* of f (w.r.t. $\mathbf{z}$) to be

$$\mathcal{E}_{\mathbf{z}}(f) = \frac{1}{m} \sum_{i=1}^{m} (f(x_i) - y_i)^2.$$

If ξ is a random variable on Z, we denote the *empirical mean* of ξ (w.r.t. $\mathbf{z}$) by $\mathbf{E}_{\mathbf{z}}(\xi)$. Thus,

$$\mathbf{E}_{\mathbf{z}}(\xi) = \frac{1}{m} \sum_{i=1}^{m} \xi(z_i).$$

For any function $f : X \to Y$ we denote by f_Y the function

$$\begin{aligned} f_Y : X \times Y &\to Y \\ (x, y) &\mapsto f(x) - y. \end{aligned}$$

With these notations we may write $\mathcal{E}(f) = \mathbf{E}(f_Y^2)$ and $\mathcal{E}_{\mathbf{z}}(f) = \mathbf{E}_{\mathbf{z}}(f_Y^2)$. We have already remarked that the expected value of $(f_\rho)_Y$ is 0; we now remark that its variance is σ_ρ^2.

Remark 1.9 Consider the PAC learning setting discussed in Case 1.5 where $X = \mathbb{R}^n$ and T is a subset of $\mathbb{R}^n$.[2] The measure ρ_X described there can be extended to a measure ρ on Z by defining, for $A \subset Z$,

$$\rho(A) = \rho_X(\{x \in X \mid (x, f_T(x)) \in A\}),$$

where, we recall, f_T is the characteristic function of the set T. The marginal measure of ρ on X is our original ρ_X. In addition, $\sigma_\rho^2 = 0$, the error $\mathcal{E}$ specializes to the error mentioned in Case 1.5, and the regression function f_ρ of ρ coincides with f_T except for a set of measure zero in X.

[2] Note, in this case, that X is not compact. In fact, most of the results in this book do not require compactness of X but only completeness and separability.

1.3 Hypothesis spaces and target functions

Learning processes do not take place in a vacuum. Some structure needs to be present at the beginning of the process. In our formal development, we assume that this structure takes the form of a class of functions (e.g., a space of polynomials, of splines, etc.). The goal of the learning process is thus to find the best approximation of f_ρ within this class.

Let $\mathscr{C}(X)$ be the Banach space of continuous functions on X with the norm

$$\|f\|_\infty = \sup_{x\in X} |f(x)|.$$

We consider a subset $\mathcal{H}$ of $\mathscr{C}(X)$ – in what follows called *hypothesis space* – where algorithms will work to find, as well as is possible, the best approximation for f_ρ. A main choice in this book is a compact, infinite-dimensional subset of $\mathscr{C}(X)$, but we will also consider closed balls in finite-dimensional subspaces of $\mathscr{C}(X)$ and whole linear spaces.

If $f_\rho \in \mathcal{H}$, simplifications will occur, but in general we will not even assume that $f_\rho \in \mathscr{C}(X)$ and we will have to consider a *target function* $f_\mathcal{H}$ in $\mathcal{H}$. Define $f_\mathcal{H}$ to be any function minimizing the error $\mathcal{E}(f)$ over $f \in \mathcal{H}$, namely, any optimizer of

$$\min_{f\in\mathcal{H}} \int_Z (f(x) - y)^2 \, d\rho.$$

Notice that since $\mathcal{E}(f) = \int_X (f - f_\rho)^2 + \sigma_\rho^2$, $f_\mathcal{H}$ is also an optimizer of

$$\min_{f\in\mathcal{H}} \int_X (f - f_\rho)^2 \, d\rho_X.$$

Let $\mathbf{z} \in Z^m$ be a sample. We define the *empirical target function* $f_{\mathcal{H},\mathbf{z}} = f_\mathbf{z}$ to be a function minimizing the empirical error $\mathcal{E}_\mathbf{z}(f)$ over $f \in \mathcal{H}$, that is, an optimizer of

$$\min_{f\in\mathcal{H}} \frac{1}{m} \sum_{i=1}^m (f(x_i) - y_i)^2. \tag{1.1}$$

Note that although $f_\mathbf{z}$ is not produced by an algorithm, it is close to algorithmic. The statement of the minimization problem (1.1) depends on ρ only through its dependence on $\mathbf{z}$, but once $\mathbf{z}$ is given, so is (1.1), and its solution $f_\mathbf{z}$ can be looked for without further involvement of ρ. In contrast to $f_\mathcal{H}$, $f_\mathbf{z}$ is "empirical" from its dependence on the sample $\mathbf{z}$. Note finally that $\mathcal{E}(f_\mathbf{z})$ and $\mathcal{E}_\mathbf{z}(f)$ are different objects.

We next prove that $f_\mathcal{H}$ and $f_\mathbf{z}$ exist under a mild condition on $\mathcal{H}$.

Definition 1.10 Let $f : X \to Y$ and $\mathbf{z} \in Z^m$. The *defect* of f (w.r.t. $\mathbf{z}$) is

$$L_{\mathbf{z}}(f) = L_{\rho,\mathbf{z}}(f) = \mathcal{E}(f) - \mathcal{E}_{\mathbf{z}}(f).$$

Notice that the theoretical error $\mathcal{E}(f)$ cannot be measured directly, whereas $\mathcal{E}_{\mathbf{z}}(f)$ can. A bound on $L_{\mathbf{z}}(f)$ becomes useful since it allows one to bound the actual error from an observed quantity. Such bounds are the object of Theorems 3.8 and 3.10.

Let $f_1, f_2 \in \mathscr{C}(X)$. Toward the proof of the existence of $f_{\mathcal{H}}$ and $f_{\mathbf{z}}$, we first estimate the quantity

$$|L_{\mathbf{z}}(f_1) - L_{\mathbf{z}}(f_2)|$$

linearly by $\|f_1 - f_2\|_\infty$ for almost all $\mathbf{z} \in Z^m$ (a Lipschitz estimate). We recall that a set $U \subseteq Z$ is said to be *full measure* when $Z \setminus U$ has measure zero.

Proposition 1.11 *If, for $j = 1, 2$, $|f_j(x) - y| \le M$ on a full measure set $U \subseteq Z$ then, for all $\mathbf{z} \in U^m$,*

$$|L_{\mathbf{z}}(f_1) - L_{\mathbf{z}}(f_2)| \le 4M \|f_1 - f_2\|_\infty.$$

Proof. First note that since

$$(f_1(x) - y)^2 - (f_2(x) - y)^2 = (f_1(x) + f_2(x) - 2y)(f_1(x) - f_2(x)),$$

we have

$$\begin{aligned} |\mathcal{E}(f_1) - \mathcal{E}(f_2)| &= \left| \int_Z (f_1(x) + f_2(x) - 2y)(f_1(x) - f_2(x))\, d\rho \right| \\ &\le \int_Z |(f_1(x) - y) + (f_2(x) - y)|\, \|f_1 - f_2\|_\infty\, d\rho \\ &\le 2M \|f_1 - f_2\|_\infty. \end{aligned}$$

Also, for all $\mathbf{z} \in U^m$, we have

$$\begin{aligned} |\mathcal{E}_{\mathbf{z}}(f_1) - \mathcal{E}_{\mathbf{z}}(f_2)| &= \frac{1}{m} \left| \sum_{i=1}^m (f_1(x_i) + f_2(x_i) - 2y_i)(f_1(x_i) - f_2(x_i)) \right| \\ &\le \frac{1}{m} \sum_{i=1}^m |(f_1(x_i) - y) + (f_2(x_i) - y_i)|\, \|f_1 - f_2\|_\infty \\ &\le 2M \|f_1 - f_2\|_\infty. \end{aligned}$$