Chapter 16
Statistical machine learning

16.1 What is machine learning in this book?

Given a set of data $S := \left\{ \left(\mathbf{x}^i, y^i\right) \right\}_{i=1}^N$, where $y^i = f^*\left(\mathbf{x}^i\right)$ for some unknown function f^*, *in this book machine learning is the task of learning the function (or the map) f^* from the incomplete information described by the training set $\left\{ \left(\mathbf{x}^i, y^i\right) \right\}_{i=1}^N$*. Note that we have assume that $y\left(\mathbf{x}\right)$ is a scalar-valued function, and the extension to vector-valued function is straightforward.

Let X be a closed and bounded (and hence compact) subset of $\mathbb{R}^k$, and $y \in \mathbb{R}$. Let h be any approximation of f^* and we are interested in measuring the error that we commit in approximating f^* using h, the error is also known as the "risk" in the machine learning literature. To measure the risk we let π be a (Borel) probability measure on the on product space $Z := X \times y$, $\pi\left(y|\mathbf{x}\right)$ be the conditional probability measure of y given $\mathbf{x}$, and $\pi\left(\mathbf{x}\right) := \int_y d\pi\left(\mathbf{x}, y\right)$ be the marginal probability measure on X. By Bayes' theorem, we know that $\pi\left(\mathbf{x}, y\right) = \pi\left(y|\mathbf{x}\right) \times \pi\left(\mathbf{x}\right)$. One of the popular risk functions is the *least squares error* with respect to the product measure π:

Note the similarity between the risk and the misfit in Chapter 7.

$$\mathscr{R}\left(h\right) := \int_Z \left(h\left(\mathbf{x}\right) - y\right)^2 d\pi\left(\mathbf{x}, y\right).$$

Cleary, the best $h^*\left(\mathbf{x}\right) \in \mathbb{L}^2\left(X, \pi\left(\mathbf{x}\right)\right)$, i.e. $\int_X \left\|h^*\right\|_{\mathbb{R}^d}^2 d\pi\left(\mathbf{x}\right) < \infty$, is the one which minimizes the risk. Chapter 2 shows that the first variation of the risk $\mathscr{R}$ at h^* in any direction $\tilde{h}\left(\mathbf{x}\right) \in \mathbb{L}^2\left(X, \pi\left(\mathbf{x}\right)\right)$ must vanish, i.e.,

$$\mathscr{D}\mathscr{R}\left(h^*; \tilde{h}\right) = 2\int_Z \left(h^*\left(\mathbf{x}\right) - y\right) \tilde{h}\left(\mathbf{x}\right) d\pi\left(\mathbf{x}, y\right) = 0, \quad \forall \tilde{h}\left(\mathbf{x}\right) \in \mathbb{L}^2\left(X, \pi\left(\mathbf{x}\right)\right),$$

which, after using the Bayes' theorem, becomes

Do you see it? Note that h^* and $\tilde{h}$ are functions of only $\mathbf{x}$.

$$\int_X \left(h^*\left(\mathbf{x}\right) - \int_y y\, d\pi\left(y|\mathbf{x}\right)\right) \tilde{h}\left(\mathbf{x}\right) d\pi\left(\mathbf{x}\right) = 0, \quad \forall \tilde{h}\left(\mathbf{x}\right) \in \mathbb{L}^2\left(X, \pi\left(\mathbf{x}\right)\right),$$

See Chapter 2.

which, by Riesz representation theorem, implies that

$$\boxed{h^*(\mathbf{x}) = \int_y y\, d\pi(y|\mathbf{x})\,.} \tag{16.1}$$

Note that $h^*(\mathbf{x})$ is known as the *regresstion function of* $\pi(\mathbf{x}, y)$.

Exercise 16.1. Show that the risk for any function h is given as

$$\mathscr{R}(h) = \int_X (h(\mathbf{x}) - h^*(\mathbf{x}))^2\, d\pi(\mathbf{x}) + \sigma_\pi^2,$$

where we have defined

$$\sigma_\pi^2 := \int_Z (h^*(\mathbf{x}) - y)^2\, d\pi(\mathbf{x}, y)$$

•

Note that since σ_π^2 is independent of h, minimizing the risk is the same as minimizing distance between h and h^*. In particular, h^* is the minimizer of the risk.

16.2 Empirical Risk Minimization (ERM)

Assumption 16.1 (i.i.d. assumption of the training set S). The training set $S = \left\{ \left(\mathbf{x}^i, y^i\right) \right\}_{i=1}^N$ are i.i.d. draws from $\pi(\mathbf{x}, y)$ on the product space $Z = X \times y$. Note that $S = \left\{ \left(\mathbf{x}^i, y^i\right) \right\}_{i=1}^N$ can be equivalently considered as a random draw from on Z^N with the N-fold product measure $\pi(\mathbf{x}, y) \times \pi(\mathbf{x}, y) \times \ldots \times \pi(\mathbf{x}, y)$.

Unfortunately the regression function (16.1) is not computable, and hence the risk, since it would require the joint distribution $\pi(\mathbf{x}, y)$, which is unknown. What available about $\pi(\mathbf{x}, y)$ is the limited information given by the training set S. *We assuming that the training set is i.i.d. in the sense of Assumption 16.1*, then by the law of large numbers (see Chapter **??**) we have

$$\mathscr{R}_N(h) := \frac{1}{N} \sum_{i=1}^N \left(h\left(\mathbf{x}^i\right) - y^i\right)^2 \overset{a.s.}{\rightarrow} \mathscr{R}(h),$$

where $\mathscr{R}_N(h)$ *is known as the empirical risk*. Thus we have to resort to minimizing the empirical risk, i.e.,

$$\boxed{h_N(\mathbf{x}) \in \arg\min_h \mathscr{R}_N(h)\,.} \tag{16.2}$$

This is the well-known *empirical risk minimization* (ERM) problem, and here h_N is a solution to the ERM problem.

Note that since the training set contain random realizations of $\pi(\mathbf{x}, y)$, $h_N(\mathbf{x})$ is a random function.

16.3 Overfitting and No-Free-Lunch theorem

It turns out that there we can always find a *learner*, i.e. a learning algorithm, $h_N(\mathbf{x})$ that makes that empirical risk vanish, i.e. $\mathscr{R}_N(h_N) = 0$ while having (significant) actual risk $\mathscr{R}(h_N) > 0$. This is known as overfitting: the learner does not *generalize* beyond the data, that is, the training error vanishes but the generalized error can be significant. Since the actual risk is our primary object of interest, overfitting is undersirable. You may hope to find a learner that avoid overfitting. Unfortunately this is impossible as pointed out by the following No-Free-Lunch theorem.

Theorem 16.1 (No-Free-Lunch). *For any $N \geq 1$, and any learner $h_N(\mathbf{x})$ built from the training data $S = \left\{\left(\mathbf{x}^i, y^i\right)\right\}_{i=1}^{N}$, where $y \in \{0,1\}$, there exists a joint distribution $\pi(\mathbf{x}, y)$ such that*

- $\mathscr{R}_N(h_N) = 0$ *and*
- $\mathbb{E}_\pi[\mathscr{R}(h_N)] \geq 1/8.$

Proof. See [26]. □

In other words, there is no universal learner—no learner can succeed in all learning tasks—unless further hypothesis or prior structure is present at the beginning of the learning process. This is, in the language of inverse problem, the ill-posedness nature of the learning problem in which we need to make inference on the high (possibly infinite) dimensional learner with limited information from data. We have seen in Chapter 1 that this problem can be overcome by using regularization, and restricting the learner having a prior structure is in fact a regularization technique.

In this book, the prior structure takes the form of a class of functions, namely, the *hypothesis space* in which we find a learner $h_N(\mathbf{x})$ that approximates the regression function (16.1) as well as it can while avoiding the overfitting problem. We shall also determine the compromise between the sample size N and the hypothesis space in order to minimize the *generalized error*, i.e. the actual risk.