Machine learning ex-ch2 1

Understanding Machine Learning: From Theory to Algorithms [Exercise 2.4]

Mehrnaz jalili
Email: Mehrnazjalili1991@gmail.com

Date: November 20, 2021

Exercises for Section 2.4.1: Overfitting of polynomial matching

We have shown that the predictor defined in Equation (2.3) leads to overfitting. While this predictor seems to be very unnatural,the goal of this exercise is to show that it can be described as a thresholded polynomial. That is, show that given a training set $S = \{(x_i, f(x_i))\}_{i=1}^m \subseteq (\mathbb{R}^d \times \{0,1\})^m$, there exists a polynomial p_S such that $h_S(x) = 1$ if and only if $p_S(x) \geq 0$, where h_S is as defined in Equation (2.3). It follows that learning the class of all thresholded polynomials using the ERM rule may lead to overfitting.

First, we define the concept of "Norm": Given a vector space X over a subfield F of the complex numbers $\mathbb{C}$, a **Norm** on X is a real-valued function $\|*\| : X \to \mathbb{R}$ with the following properties, where $|s|$ denotes the usual absolute value of a scalar s:

1. Subadditivity/Triangle inequality: $\|x + y\| \leq \|x\| + \|y\|$ for all $x, y \in X$.

2. Absolute homogeneity: $\|sx\| = |s| . \|x\|$ for all $x \in X$ and and all scalars s.

3. Positive definiteness/Point-separating: for all $x \in X$, if $\|x\| = 0$ then $x = 0$.

4. Nonnegativity: $\|x\| \geq 0$ for all $x \in X$

Consider an arbitrary norm so that $\|*\| : \mathbb{R}^d \to \mathbb{R}$. We are looking for a polynomial function which returns a negative value whenever $h_S(x) = 1$. A simple solution to this problem is setting the function's value to 0 at points whenever $h_S(x) = 1$. we achieved this by multiplying $\|x - x_j\|$s such that $j \in J$ (It is noteworthy that $J = \{j \in [m] : h_S(x_j) = 1\}$) [1] But we have two main problems. First, the value of the function must be negative at the points $h_s(x) = 0$, which can be reached by multiplying the expression by a (-1). The second problem is that its function will change the sign at the points x_j. So we consider each element to the power of an arbitrary positive number such as 4.

$$p_S(x) = -\prod_{j\in J} \|x - x_j\|^4$$

Now we can see that $p_S(x) = 0$ (or with a deeper look $p_S(x) \geq 0$) if and only if $h_S(x) = 1$

Exercises for Section 2.4.2

Let $\mathcal{H}$ be a class of binary classifiers over a domain $\mathcal{X}$. Let $\mathcal{D}$ be an unknown distribution over $\mathcal{X}$, and let f be the target hypothesis in $\mathcal{H}$. Fix some $h \in \mathcal{H}$. Show that the expected value of $L_S(h)$ over the choice of $S|_x$ equals $L_{(\mathcal{D},f)}(h)$, namely,

$$\underset{S|_x\sim\mathcal{D}^m}{\mathbb{E}}[L_S(h)] = L_{(\mathcal{D},f)}(h)$$

First, let's take a closer look at the definitions and theorems. We have defined the error of a prediction rule, $h : \mathcal{X} \to \mathcal{Y}$, to be

$$L_{\mathcal{D},f}(h) \overset{\text{def}}{=} \underset{x\sim\mathcal{D}}{\mathbb{P}}[h(x) \neq f(x)] \overset{\text{def}}{=} \mathcal{D}(\{x : h(x) \neq f(x)\}) \overset{\text{def}}{=} \mathbb{E}_{x\sim\mathcal{D}}[1_{h(x)\neq f(X)}]$$

[1] m is the cardinal of our training set

In addition, we have proved by the right most expression of the above equalities that $L_{\mathcal{D},f}(h) = \mathbb{E}[L_S(h)]$. Inspired by the aforementioned proof, we will try to start with $\mathbb{E}_{x\sim\mathcal{D}}[1_{h(x)\neq f(X)}]$ and complete the proof by expanding expressions. Before we start proving, we define the set J to be $J = \{j \in [m], 1 \leq j \leq |S|\}$.[2]

The first method:

$$\begin{aligned}
\mathop{\mathbb{E}}_{S|_x\sim\mathcal{D}^m}[L_S(h)] &= \mathop{\mathbb{E}}_{S|_x\sim\mathcal{D}^m}\left[\frac{1}{m}\sum_{j\in J}[1_{h(x_j)\neq f(x_j)}]\right] \\
&= \frac{1}{m}\sum_{j\in J}\left(\mathop{\mathbb{E}}_{x_j\sim\mathcal{D}}\left[1_{h(x_j)\neq f(x_j)}\right]\right) && \text{(by linearity)} \\
&= \frac{1}{m}\sum_{j\in J}\left(\mathop{\mathbb{E}}_{x\sim\mathcal{D}}\left[1_{h(x)\neq f(x)}\right]\right) && \text{(by i.i.d)} \\
&= \frac{1}{m}\times m\times\left(\mathop{\mathbb{E}}_{x\sim\mathcal{D}}\left[1_{h(x)\neq f(x)}\right]\right) && \text{(Since the } |J| = m\text{)} \\
&= L_{\mathcal{D},f}(h) && \text{generalization error}
\end{aligned}$$

The second method:

$$\begin{aligned}
\mathop{\mathbb{E}}_{S|_x\sim\mathcal{D}^m}[L_S(h)] &= \mathop{\mathbb{E}}_{S|_x\sim\mathcal{D}^m}\left[\frac{1}{m}\sum_{j\in J}[1_{h(x_j)\neq f(x_j)}]\right] \\
&= \frac{1}{m}\sum_{j\in J}\left(\mathop{\mathbb{E}}_{x_j\sim\mathcal{D}}\left[1_{h(x_j)\neq f(x_j)}\right]\right) && \text{(by linearity)} \\
&= \frac{1}{m}\sum_{j\in J}\left(\mathop{\mathbb{P}}_{x_j\sim\mathcal{D}}\left[h(x_j)\neq f(x_j)\right]\right) \\
&= \frac{1}{m}\sum_{j\in J}\left(\mathop{\mathbb{P}}_{x\sim\mathcal{D}}\left[h(x)\neq f(x)\right]\right) && \text{(by i.i.d)} \\
&= \frac{1}{m}\times m\times L_{\mathcal{D},f}(h) && \text{(by the definition of } L_{\mathcal{D},f}(h)\text{)} \\
&= L_{\mathcal{D},f}(h) && \text{generalization error}
\end{aligned}$$

Exercises for Section 2.4.3: Axis aligned rectangles

An axis aligned rectangle classifier in the plane is a classifier that assigns the value 1 to a point if and only if it is inside a certain rectangle. Formally, given real numbers $a_1 \leq b_1,\ \ a_2 \leq b_2$, define the classifier $h_{(a_1.b_1,a_2,b_2)}$ by

$$h_{(a_1.b_1,a_2,b_2)}(x_1,x_2) = \begin{cases} 1 & \text{if } a_1 \leq x \leq b_1 \text{ and } a_2 \leq x_2 \leq b_2 \\ 0 & \text{otherwise} \end{cases}$$

The class of all axis aligned rectangles in the plane is defined as

$$\mathcal{H}^2_{\text{rec}} = \{h_{(a_1.b_1,a_2,b_2)} : a_1 \leq b_1 \text{ , and } a_2 \leq b_2\}$$

Note that this is an infinite size hypothesis class. Throughout this exercise we rely on the realizability assumption.

1. Let A be the algorithm that returns the smallest rectangle enclosing all positive examples in the training set. Show that A is an **ERM**.

[2] S is our training set and $|S| = m$

2. Show that if A receives a training set of size $\geq \frac{4\log(\frac{4}{\delta})}{\epsilon}$ then, with probability of at least $1-\delta$ it returns a hypothesis with error of at most ϵ .

   Hint: Fix some distribution $\mathcal{D}$ over $\mathcal{X}$, let $R^* = R(a_1^*, b_1^*, a_2^*, b_2^*)$ be the rectangle that generates the labels, and let f be the corresponding hypothesis. Let $a_1 \geq a_1^*$ be a number such that the probability mass (with respect to $\mathcal{D}$) of the rectangle $R_1 = R(a_1^*, a_1, a_2^*, b_2^*)$ is exactly $\frac{\epsilon}{4}$. Similarly, let b_1, a_2, b_2 be numbers such that the probability masses of the rectangles $R_2 = R(b_1, b_1^*, a_2^*, b_2^*)$, $R_3 = R(a_1^*, b_1^*, a_2^*, a_2)$, $R_4 = R(a_1^*, b_1^*, b_2, b_2^*)$ are all exactly $\frac{\epsilon}{4}$. Let $R(s)$ be the rectangle returned by A. See illustration in Figure 2.2.

   - Show that $R(S) \subseteq R^*$.
   - Show that if S contains (positive) examples in all of the rectangles R_1, R_2, R_3, R_4, then the hypothesis returned by A has error of at most ϵ.
   - For each $i \in \{1, ..., 4\}$, upper bound the probability that S does not contain an example from R_i .
   - Use the union bound to conclude the argument.

3. Repeat the previous question for the class of axis aligned rectangles in $\mathbb{R}^d$.

4. Show that the runtime of applying the algorithm A mentioned earlier is polynomial in d, $\frac{1}{\epsilon}$, and in $\log\left(\frac{1}{\delta}\right)$.

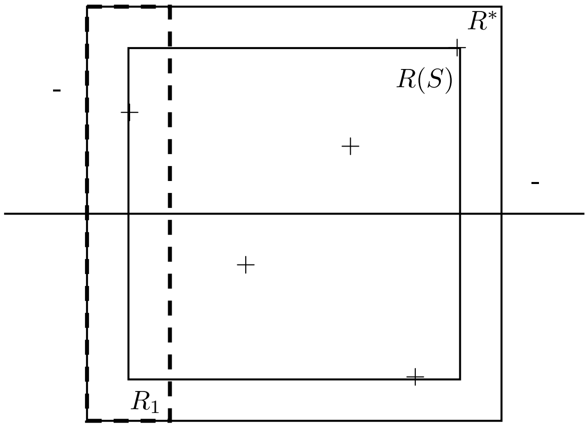

Figure 2.2. Axis aligned rectangles.

1. We have a training set which is called S. We also consider Algorithm A to return the smallest rectangle containing all the positives as the output. It is clear that A is a member of $\mathcal{H}$. Because $\mathcal{H}$ is the set of all rectangles in the universe. Furthermore, considering the realizability assumption, we have at least one member of $\mathcal{H}$ such that $L_s(h) = 0$ Given that the output of Algorithm A contains all positive labels and is clearly due to being the tiniest one, it is not going to mislabel negative elements, so $L_S(A) = 0$, therefore, $A \in \arg\min_{h\in\mathcal{H}} L_s(h)$. So A is an ERM.

2. First, we consider everything that was raised in the Hint section of the question as basic assumptions. From the definition of Algorithm A , it can be easily concluded that $R(s) \subseteq R^*$.

$$L_{(\mathcal{D},f)} = \mathcal{D}(R^* - R(S))$$

   Second, we fix some $\epsilon \in (0,1)$ and then consider R_i as hint.[3] Then we define another family of sets $\mathcal{F}$.

$$F_i = \{S|_x : S|_x \cap R_i = \varnothing\}$$

[3] For each $i \in \{1, 2, 3, 4\}$.

Then we calculate the probability that $L_{(\mathcal{D},f)}$ being greater than ϵ.

$$\mathcal{D}^m\left(\left\{S : L_{(\mathcal{D},f)}(R(s)) > \epsilon\right\}\right) \leq \mathcal{D}^m\left(\bigcup_{i=1}^{4} F_i\right) \leq \sum_{i=1}^{4} \mathcal{D}^m\left(F_i\right)$$

$$\mathcal{D}^m(F_i) = (1 - \frac{\epsilon}{4})^m \leq e^{-m*\frac{\epsilon}{4}}$$

Therefore,

$$ginequation{*}\mathcal{D}^m\left(\left\{S : L_{(\mathcal{D},f)}(R(s)) > \epsilon\right\}\right) \leq 4.e^{-m\cdot\frac{\epsilon}{4}}$$

3. Formally, given real numbers $a_1 \leq b_1,\ \ a_2 \leq b_2\ \ ...\ \ a_d \leq b_d$, define the classifier $h_{(a_1.b_1,...,a_d,b_d)}$ by

$$h_{(a_1.b_1,...,a_d,b_d)}(x_1, ..., x_d) = \begin{cases} 0 & \exists i \in [d], x_i \notin (a_i, b_i) \\ 1 & \text{otherwise} \end{cases}$$

The class of all axis aligned rectangles in the plane is defined as

$$\mathcal{H}^d_{\text{rec}} = \{h_{(a_1.b_1,...,a_d,b_d)} : \forall i \in [d], a_i \leq b_i\}$$

Note that this is an infinite size hypothesis class. Throughout this exercise we rely on the realizability assumption.

I Let A be the algorithm that returns the smallest d-dimensional cube [4] containing all the training set's positive elements. Show that A is an ERM.

II Show that if A receives a training set of size $\geq \frac{2d\log(\frac{2d}{\delta})}{\epsilon}$ then, with probability of at least $1-\delta$ it returns a hypothesis with error of at most ϵ .

This project was written in collaboration with Arash Sajjadi.

[4] In geometry, a **hypercube** is an n-dimensional analogue of a square ($n = 2$) and a cube ($n = 3$). It is a closed, compact, convex figure whose 1-skeleton consists of groups of opposite parallel line segments aligned in each of the space's dimensions, perpendicular to each other and of the same length. A unit hypercube's longest diagonal in n dimensions is equal to $\sqrt{n}$. An **n-dimensional hypercube** is more commonly referred to as an **n-cube** or sometimes as an **n-dimensional cube**.