TFML HW-1

SIDDHANT CHAUDHARY
BMC201953

The reference book used was *Understanding Machine Learning by Shai Shalev-Shwartz, Shai Ben-David*.

(1) and (2): Axis Aligned Rectangles (Problem 2.3 of the book). In this problem, we will prove the learnability of *axis aligned rectangles* in $\mathbf{R}^d$. We will first deal only with $d = 2$, i.e rectangles in $\mathbf{R}^2$. Note that we will have the *realizability assumption* throughout.

1. Let A be the algorithm that returns the smallest rectangle enclosing all positive examples in the training set. We show that A is an ERM. Here the loss function is the 0-1 loss (i.e an error incurs a loss of 1 and no error means 0 loss).

Because we have the *realiziability assumption* in place, we know that there is some rectangle R in $\mathbf{R}^2$ such that

$$L_{\mathcal{D},f}(R) = 0$$

This means that the rectangle R contains all the positive points, and that no negative point is contained inside R. Note that if R^* is the *smallest rectangle* containing all the positive points, then clearly $R^* \subseteq R$, and hence R^* cannot contain any negative points. Now, if S is any training set, then clearly $A(S)$, the smallest rectangle containing all positive points in S, must satisfy $A(S) \subseteq R^*$. This means that $A(S)$ does not contain any negative points at all, and clearly this means that

$$L_S(A(S)) = 0$$

because $A(S)$ correctly classifies the positive points within S, and does not contain any negative points. So, A is indeed an ERM.

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

So, let $\mathcal{D}$ be any distribution on $\mathcal{X} = \mathbf{R}^2$, which is the domain set. Let $R^* = R(a_1^*, b_1^*, a_2^*, b_2^*)$ be the rectangle that generates the labels (i.e, let R^* be the smallest rectangle containing all the positive labels, which we know exists by the *realizability assumption*). This implies that f must satisfy the following (where f is the labelling function).

$$f(x_1, x_2) = \begin{cases} 1 & , \quad \text{if } a_1^* \leq x_1 \leq b_1^* \text{ and } a_2^* \leq x_2 \leq b_2^* \\ 0 & , \quad \text{otherwise} \end{cases}$$

Next, we define four special rectangles R_1, R_2, R_3 and R_4 as follows. Let $a_1 \geq a_1^*$ be a number such that the probability mass (w.r.t $\mathcal{D}$) of the rectangle $R_1 = R(a_1^*, a_1, a_2^*, b_2^*)$ is $\epsilon/4$ (we allow $a_1 = \infty$). Similarly, we define corresponding numbers b_1, a_2 and b_2 such that the probability masses of the rectangles $R_2 = R(b_1, b_1^*, a_2, b_2)$, $R_3 = R(a_1^*, b_1^*, a_2^*, a_2)$ and $R_4 = R(a_1^*, b_1^*, b_2, b_2^*)$ are all exactly $\epsilon/4$ (and again, we allow these

Date: 12 October, 2021.

rectangles to have infinite area). Let $R(S)$ be the rectangle returned by A on sample S.

(1) First, we will show that $R(S) \subseteq R^*$. This is because of the algorithm A: note that given any set of points S, the algorithm A returns the *smallest rectangle* containing all the positive points in that set. By our assumption, the smallest rectangle containing *all* the positive points is R^*; so, if we take a *subset* of these positive points, then the smallest rectangle containing that subset has to be contained in R^*. So, $R(S) \subseteq R^*$.

(2) Next, suppose S contains a positive point in each of the rectangle R_1, R_2, R_3 and R_4. This will imply that $R(S)$ will intersect with each of R_1, R_2, R_3 and R_4. Now, the error of the hypothesis $R(S)$ is just going to be the following.

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

This above is true because $R(S)$ will correctly classify all positive examples within it, and it will also correctly classify all negative examples (since they lie outside R^*, and $R(S) \subseteq R^*$). So, the only error that can be seen is on the positive examples in the set $R^* \setminus R(S)$. Finally, since $R(S)$ intersects with each of R_1, R_2, R_3 and R_4, and since each R_i is connected to a distinct edge of R^*, it follows that

$$\mathcal{D}^m(R^* \setminus R(S)) \leq \sum_{i=1}^{4} \mathcal{D}^m(R_i) \leq 4 \cdot \frac{\epsilon}{4} = \epsilon$$

So, it follows that

$$L_{\mathcal{D},f}(R(S)) \leq \epsilon$$

in this case.

(3) Now, for each i, we will upper bound the probability that S does not contain any positive example from R_i. First, suppose R_i contains a *negative example*. Clearly, this implies that $R^* \subseteq R_i$ (by the way R_i has been defined, this will means that R_i *moves out* of the rectangle R^*), which means $\mathcal{D}^m(R^*) \leq \epsilon/4$. In this case, the probability that S does not contain any positive example in R_i is the same as the probability that S does not contain any positive example at all (because all positive examples are in R^*), and this has probability $\left(1 - \frac{\epsilon}{4}\right)^m \leq e^{-\frac{m\epsilon}{4}}$.

Next, suppose that R_i *does not* contain any negative example, implying that $R_i \subseteq R^*$. In this case, the probability that S does not contain any positive example from R_i is the same as the probability that $S \cap R_i = \phi$, which is again $\left(1 - \frac{\epsilon}{4}\right)^m \leq e^{-\frac{m\epsilon}{4}}$.

So, for each i, the probability that S does not contain any example from R_i is $\leq e^{\frac{-m\epsilon}{4}}$.

(4) Finally, observe the following: points number (2) and (3) above show that

$$\begin{aligned}
\mathop{\mathbf{P}}_{S \sim \mathcal{D}^m}[L_{\mathcal{D},f}(A(S)) > \epsilon] &= \mathop{\mathbf{P}}_{S \sim \mathcal{D}^m}[S \cap R_i \text{ has no positive sample for some } i] \\
&\leq \sum_{i=1}^{4} \mathop{\mathbf{P}}_{S \sim \mathcal{D}^m}[S \cap R_i \text{ has no positive sample}] \\
&\leq 4e^{\frac{-m\epsilon}{4}}
\end{aligned}$$

If $m \geq \frac{4\log(4/\delta)}{\epsilon}$, then

$$4e^{\frac{-m\epsilon}{4}} \leq \delta$$

which means that

$$\mathop{\mathbf{P}}_{S\sim\mathcal{D}^m}[L_{\mathcal{D},f}(A(S)) \leq \epsilon] \geq 1-\delta$$

So, via the above proof, we've shown that rectangles in $\mathbf{R}^2$ are learnable.

3. Let us now consider axis-aligned rectangles in $\mathbf{R}^d$. Most of the above arguments can be repeated for d-dimensions as well, with a bunch of changes.

First, the algorithm A remains the same: given a input set S, it will return the *smallest hypercube* containing all the positive points in the input set. The realizability assumption gives us a hypercube hypercube $R^* = (a_1^*, b_1^*, ..., a_d^*, b_d^*)$, which is the smallest hypercube containing all of the positive points in the domain set. In d dimensions, any hypercube is described by d intervals $[a_i, b_i]$ for $1 \leq i \leq d$; a point $(x_1, ..., x_d)$ is labelled positive if $a_i \leq x_i \leq b_i$ for each i, and is labelled negatively otherwise.

Then, for each $1 \leq i \leq d$, we define two rectangles R_i and R_i' as follows.

$$R_i = R(a_1^*, b_1^*, ..., a_{i-1}^*, b_{i-1}^*, a_i^*, a_i, a_{i+1}^*, b_{i+1}^*, ..., a_d^*, b_d^*)$$
$$R_i' = R(a_1^*, b_1^*, ..., a_{i-1}^*, b_{i-1}^*, b_i, b_i^*, a_{i+1}^*, b_{i+1}^*, ..., a_d^*, b_d^*)$$

Above $a_i \geq a_i^*$ is the real number for which the rectangle R_i has probability mass (w.r.t $\mathcal{D}$) exactly $\frac{\epsilon}{2d}$. Similarly, $b_i \leq b_i^*$ is that real number for which R_i' has mass exactly $\frac{\epsilon}{2d}$.

We again have the following observations.

(1) If S is any input set, then again by the definition of the algorithm A, $R(S) \subseteq R^*$.
(2) If the training set S contains a positive point in each of the rectangles R_i, R_i', then again it will be true that

$$\mathcal{D}^m(R^* \setminus R(S)) \leq \sum_{i=1}^{d} \mathcal{D}^m(R_i) + \mathcal{D}^m(R_i') \leq 2d \cdot \frac{\epsilon}{2d} = \epsilon$$

and hence in this case the error of $R(S)$ will be atmost ϵ.
(3) Just like in the 2 dimensional case, the probability that S does not contain any positive point from the set R_i or R_i' will be bounded above by $e^{-\frac{m\epsilon}{2d}}$ (the same argument goes through).
(4) So, if we choose m such that

$$2de^{\frac{-m\epsilon}{2d}} \leq \delta$$

which is the same as choosing

$$m \geq \frac{2d\log(2d/\delta)}{\epsilon}$$

then we are guaranteed that the error of the output is atmost ϵ.

(3) Problem 3.5 of the book. Let $\mathcal{X}$ be a domain, and let $\mathcal{D}_1, ..., \mathcal{D}_m$ be a sequence of distributions over $\mathcal{X}$. Let $\mathcal{H}$ be a finite hypothesis class of binary classifiers over $\mathcal{X}$ and let $f \in \mathcal{H}$. Let $\overline{\mathcal{D}}_m$ be the average distribution, i.e

$$\overline{\mathcal{D}}_m = \frac{\mathcal{D}_1 + ... + \mathcal{D}_m}{m}$$

Finally, suppose a training set S with $|S| = m$ is sampled, such that each point in the set is sampled independently, and the ith point is drawn from the distribution $\mathcal{D}_i$, for $1 \leq i \leq m$. So,

$$S = \{(\boldsymbol{x_1}, y_1), ..., (\boldsymbol{x_m}, y_m)\}$$

where $\boldsymbol{x}_i \sim \mathcal{D}_i$ and $y_i = f(\boldsymbol{x_i})$ for each $1 \leq i \leq m$. Let $\epsilon \in (0, 1)$ be fixed. We show that

$$\mathop{\mathbf{P}}_{S}\left[\exists h \in \mathcal{H} \text{ s.t. } L_{(\overline{\mathcal{D}}_m, f)}(h) > \epsilon \text{ and } L_{(S,f)}(h) = 0\right] \leq |\mathcal{H}|e^{-\epsilon m}$$

Consider the following set.

$$M := \left\{S \mid \exists h \in \mathcal{H} \text{ s.t. } L_{(\overline{\mathcal{D}}_m, f)}(h) > \epsilon \text{ and } L_{(S,f)}(h) = 0\right\}$$

Clearly, M is a subset of the following union.

$$M \subseteq \bigcup_{h \in \mathcal{H}} \left\{S \mid L_{(\overline{\mathcal{D}}_m, f)}(h) > \epsilon \text{ and } L_{(S,f)}(h) = 0\right\}$$

This is true because every $S \in M$ clearly belongs to the RHS, but due to repetitions, the inclusion may be strict. So, by a union bound, we have that

$$\mathop{\mathbf{P}}_{S}[M] \leq \sum_{h \in \mathcal{H}} \mathop{\mathbf{P}}_{S}\left[S \mid L_{(\overline{\mathcal{D}}_m, f)}(h) > \epsilon \text{ and } L_{(S,f)}(h) = 0\right] \tag{0.1}$$

So, we need to bound each term of the sum above. So, let $h \in \mathcal{H}$ be fixed such that

$$L_{(\overline{\mathcal{D}}_m, f)}(h) > \epsilon$$

By definition, this gives us the following inequality.

$$\frac{\mathop{\mathbf{P}}_{X_1 \sim \mathcal{D}_1}[h(X_1) = f(X_1)] + \cdots + \mathop{\mathbf{P}}_{X_m \sim \mathcal{D}_m}[h(X_m) = f(X_m)]}{m} < 1 - \epsilon$$

Now, to this we apply the AM-GM inequality. Doing so, we get the following.

$$\prod_{i=1}^{m} \mathop{\mathbf{P}}_{X_i \sim \mathcal{D}_i}[h(X_i) = f(X_i)] \leq \left(\frac{\sum_{i=1}^{m} \mathop{\mathbf{P}}_{X_i \sim \mathcal{D}_i}[h(X_i) = f(X_i)]}{m}\right)^m < (1 - \epsilon)^m$$

Now, we use the fact that the X_i are independent: the left hand product of the above inequality is simply

$$\mathop{\mathbf{P}}_{S}\left[L_{(S,f)}(h) = 0\right] = \prod_{i=1}^{m} \mathop{\mathbf{P}}_{X_i \sim \mathcal{D}_i}[h(X_i) = f(X_i)]$$

So, we see that

$$\mathop{\mathbf{P}}_{S}\left[L_{(S,f)}(h) = 0\right] < (1 - \epsilon)^m \leq e^{-m\epsilon}$$

So, by equation (0.1), we see that

$$\mathop{\mathbf{P}}_{S}[M] \leq |\mathcal{H}|e^{-m\epsilon}$$

and this is exactly what we wanted to prove.

(4) Problem 3.6 of the book. Let $\mathcal{H}$ be a class of binary classifiers. Suppose $\mathcal{H}$ is agnostically PAC learnable with algorithm A. We show that $\mathcal{H}$ is PAC learnable as well, with the same algorithm. (Recall that the definition of PAC learnability involved the *realizability assumption*).

So, let $f : \mathcal{X} \to \mathcal{Y}$ be a labelling function, and let $\epsilon, \delta \in (0,1)$ be fixed. Let the loss function be the $0-1$ loss. Let $\mathcal{D}$ be *any* distribution on $\mathcal{X}$ such that the realizability assumption holds w.r.t $\mathcal{H}, \mathcal{D}$ and f.

We introduce a joint distribution $\mathcal{D}'$ on $Z = \mathcal{X} \times \{0,1\}$ as follows. For $x \in \mathcal{X}$ and $y \in \{0,1\}$, define

$$\mathop{\mathbf{P}}_{(X,Y)\sim\mathcal{D}'}[X = x, Y = y] = \begin{cases} \mathop{\mathbf{P}}_{X\sim\mathcal{D}}[X = x] & , \quad \text{if } y = f(x) \\ 0 & , \quad \text{otherwise} \end{cases}$$

This can be equivalently stated in terms of conditional probabilities given $x \in \mathcal{X}$: the conditional probability of $Y = f(x)$ given $X = x$ is 1.

So now, $\mathcal{D}'$ is a distribution on $\mathcal{X} \times \{0,1\}$. Since $\mathcal{H}$ is agnostically PAC learnable, there is a number $m_{\mathcal{H}}(\epsilon, \delta)$ such that on a training set with $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d samples drawn with distribution $\mathcal{D}'$, it is true that

$$\mathop{\mathbf{P}}_{S\sim(\mathcal{D}')^m}\left[L_{\mathcal{D}'}(A(S)) \leq \min_{h\in\mathcal{H}} L_{\mathcal{D}'}(h) + \epsilon\right] \geq 1 - \delta \tag{0.2}$$

Note that, by our definition of $\mathcal{D}'$, points of the form $(x, f(x))$ are picked with probability $\mathop{\mathbf{P}}_{X\sim\mathcal{D}}[X = x]$, and points of the form $(x, 1-f(x))$ are never picked (i.e are picked with zero probability). Also, combining this with the fact that the loss function is $0-1$ loss, this means that for any $h \in \mathcal{H}$, we have

$$L_{\mathcal{D}'}(h) = \mathop{\mathbf{P}}_{X\sim\mathcal{D}}[h(X) \neq f(X)] = L_{\mathcal{D},f}(h)$$

Combining all of these facts into equation (0.2), we see that

$$\begin{aligned} \mathop{\mathbf{P}}_{S\sim(\mathcal{D}')^m}\left[L_{\mathcal{D}'}(A(S)) \leq \min_{h\in\mathcal{H}} L_{\mathcal{D}'}(h) + \epsilon\right] &= \mathop{\mathbf{P}}_{S\sim\mathcal{D}^m}\left[L_{\mathcal{D},f}(A(S)) \leq \min_{h\in\mathcal{H}} L_{\mathcal{D},f}(h) + \epsilon\right] \\ &= \mathop{\mathbf{P}}_{S\sim\mathcal{D}^m}[L_{\mathcal{D},f}(A(S)) \leq 0 + \epsilon] \\ &\geq 1 - \delta \end{aligned}$$

where in the last step, we have simply used the realizability assumption. So, we have just shown that $\mathcal{H}$ is PAC learable with the same algorithm A, and this completes the proof.

(5) The Bayes Optimal Predictor (Problem 3.7 of book). In this problem, we will prove the optimality of the *Bayes classifier*. First, we prove a lemma. Given a distribution $\mathcal{D}$ on $\mathcal{X} \times \mathcal{Y}$ and given $x \in \mathcal{X}$, we will use the notation $\mathcal{D}_{\mathcal{Y}|x}$ to denote the induced distribution on $\mathcal{Y}$ given $X = x$.

Lemma 0.1. *Let $\mathcal{D}$ be a distribution on $Z = X \times \{0,1\}$. Let $x \in \mathcal{X}$ be fixed. Let $g : \mathcal{X} \to \{0,1\} = \mathcal{Y}$ be any classifier, and let $f_{\mathcal{D}}$ be the Bayes classifier. Then,*

$$\mathop{\mathbf{P}}_{Y\sim\mathcal{D}_{\mathcal{Y}|x}}[g(X) = Y \mid X = x] \leq \mathop{\mathbf{P}}_{Y\sim\mathcal{D}_{\mathcal{Y}|x}}[f_{\mathcal{D}}(X) = Y \mid X = x]$$

Proof. To prove this, we will deal with the following two cases.

(1) In the first case, suppose that $f_{\mathcal{D}}(x) = 1$. By definition, this means that

$$\mathop{\mathbf{P}}_{Y \sim \mathcal{D}_{\mathcal{Y}|x}} [Y = 1 \mid X = x] \geq \mathop{\mathbf{P}}_{Y \sim \mathcal{D}_{\mathcal{Y}|x}} [Y = 0 \mid X = x]$$

Now, if $g(x) = 1$, then the claim trivially holds (because the two probabilities are equal). If $g(x) = 0$, then the above inequality is the inequality we want to prove.

(2) In the second case, we have $f_{\mathcal{D}}(x) = 0$. This case is symmetric to the above case.

So the claim has been proven. ■

Now, let us prove the original claim. Let $\mathcal{D}$ be any distribution on $Z = \mathcal{X} \times \mathcal{Y}$, and let $\mathcal{D}_{\mathcal{X}}$ be the marginal distribution over $\mathcal{X}$. Let X, Y be random variables denoting the values of x and y.

We want to show that

$$\mathop{\mathbf{P}}_{(X,Y) \sim \mathcal{D}} [f_{\mathcal{D}}(X) \neq Y] \leq \mathop{\mathbf{P}}_{(X,Y) \sim \mathcal{D}} [g(X) \neq Y]$$

Note that this is equivalent to showing that

$$\mathop{\mathbf{P}}_{(X,Y) \sim \mathcal{D}} [f_{\mathcal{D}}(X) = Y] \geq \mathop{\mathbf{P}}_{(X,Y) \sim \mathcal{D}} [g(X) = Y]$$

Intuitively, this just means that the success probability of the Bayes classifier is the maximum possible success probability. We now have the following.

$$\begin{aligned}
\mathop{\mathbf{P}}_{(X,Y) \sim \mathcal{D}} [g(X) = Y] &= \sum_{x \in \mathcal{X}} \mathop{\mathbf{P}}_{(X,Y) \sim \mathcal{D}} [g(X) = Y \wedge X = x] \\
&= \sum_{x \in \mathcal{X}} \mathop{\mathbf{P}}_{X \sim \mathcal{D}_{\mathcal{X}}} [X = x] \mathop{\mathbf{P}}_{Y \sim \mathcal{D}_{\mathcal{Y}|x}} [g(X) = Y \mid X = x] \\
&\leq \sum_{x \in \mathcal{X}} \mathop{\mathbf{P}}_{X \sim \mathcal{D}_{\mathcal{X}}} [X = x] \mathop{\mathbf{P}}_{Y \sim \mathcal{D}_{\mathcal{Y}|x}} [f_{\mathcal{D}}(X) = Y \mid X = x] \\
&= \sum_{x \in \mathcal{X}} \mathop{\mathbf{P}}_{(X,Y) \sim \mathcal{D}} [f_{\mathcal{D}}(X) = Y \wedge X = x] \\
&= \mathop{\mathbf{P}}_{(X,Y) \sim \mathcal{D}} [f_{\mathcal{D}}(X) = Y]
\end{aligned}$$

where in one of the steps above, we used **Lemma** 0.1. This proves the claim.

(6) Problem 5.2 of the book. As given in the problem statement, the features available to us are the blood pressure (BP), body-mass index (BMI), age (A), physical activity (P) and income (I).

Let $\mathcal{H}_2$ be the class of two-dimensional axis aligned rectangles, and let $\mathcal{H}_5$ be the class of five-dimensional axis aligned rectangles. Clearly, we see that $\mathcal{H}_2 \subseteq \mathcal{H}_5$.

(1) The pros of choosing the class $\mathcal{H}_2$ with the features BP and BMI are straightforward: a person's BP and BMI is more likely to affect a person's chances of getting a heart attack than the other features. Also, learning the class $\mathcal{H}_2$ is much simpler than learning the class $\mathcal{H}_5$, because we not only need fewer samples to learn, but also the complexity of the learning algorithm is smaller.

On the other hand, the major con of the class $\mathcal{H}_2$ is accuracy: if we include all the parameters like age, physical activity and income, our learner will be more accurate if it is given enough samples. Even though the complexity of the class $\mathcal{H}_5$ is more, it obviously provides a much flexible model and a model

which might generalise well. So overall, we are essentially trading between the complexity of our class and the accuracy.

(2) If we have a small number of samples and we have to learn from only those samples, it's a better choice to go with the class $\mathcal{H}_2$, because it's sample complexity is smaller and it will provide a lower generalisation error with the same training data as compared to the class $\mathcal{H}_5$. However, if we have a large number of samples, enough to train the class $\mathcal{H}_5$, and if we are willing to go with a more complex learning class, then the class $\mathcal{H}_5$ is a better choice because it is more accurate in terms of it's generalization error. So overall, one really has to see all factors before choosing the algorithm.