Advanced Machine Learning Seminar 6

Exercise 1 Fix $\epsilon \in \left(0, \dfrac{1}{2}\right)$. Let the training sample be denoted by m points in the plane with $\dfrac{m}{4}$ negative points all at coordinate $(+1,+1)$, another $\dfrac{m}{4}$ negative points all at coordinate $(-1,-1)$, $\dfrac{m(1+\epsilon)}{4}$ positive points all at coordinate $(-1,+1)$, $\dfrac{m(1-\epsilon)}{4}$ positive points all at coordinate $(+1,-1)$.

a. Describe the behavior of AdaBoost when run on this sample using boosting stumps for the first two rounds.

b. What is the error of the optimal classifier chosen at round 1 in the second round?

AdaBoost

- construct distribution $\mathbf{D}^{(t)}$ on $\{1,\ldots,m\}$:
  - $\mathbf{D}^{(t)}(i) = 1/m$
  - given $\mathbf{D}^{(t)}$ and h_t: $D^{(t+1)}(i) = \dfrac{D^{(t)}(i) \times e^{-w_t h_t(x_i) y_i}}{Z_{t+1}}$

where Z_{t+1} normalization factor ($\mathbf{D}^{(t+1)}$ is a distribution): $Z_{t+1} = \sum_{i=1}^{m} D^{(t)}(i) \times e^{-w_t h_t(x_i) y_i}$

$$w_t \text{ is a weight: } w_t = \frac{1}{2}\ln\left(\frac{1}{\epsilon_t} - 1\right) > 0 \text{ as the error } \epsilon_t < 0.5$$

$$\epsilon_t \text{ is the error of } h_t \text{ on } \mathbf{D}^{(t)} : \epsilon_t = \Pr_{i\sim D^{(t)}}[h_t(x_i) \neq y_i] = \sum_{i=1}^{m} D^{(t)}(i) \times \mathbb{1}_{[h_t(x_i)\neq y_i]}$$

If example $\mathbf{x}_i$ is correctly classified, then $h(\mathbf{x}_i) = \mathrm{y}_i$, so at the next iteration $t+1$ its importance (probability distribution) will be decreased to:

$$D^{(t+1)}(i) = \frac{D^{(t)}(i) \times e^{-w_t}}{Z_{t+1}} = \frac{D^{(t)}(i) \times e^{-\frac{1}{2}\ln\left(\frac{1}{\epsilon_t}-1\right)}}{Z_{t+1}} = \frac{D^{(t)}(i) \times \left(\frac{1}{\epsilon_t} - 1\right)^{-\frac{1}{2}}}{Z_{t+1}} = \frac{D^{(t)}(i) \times \sqrt{\frac{\epsilon_t}{1-\epsilon_t}}}{Z_{t+1}}$$

If example x_i is misclassified, then $h(\mathrm{x}_i) \neq \mathrm{y}_i$, so at the next iteration $t+1$ its importance (probability distribution) will be increased to:

$$D^{(t+1)}(i) = \frac{D^{(t)}(i) \times e^{w_t}}{Z_{t+1}} = \frac{D^{(t)}(i) \times e^{\frac{1}{2}\ln\left(\frac{1}{\epsilon_t}-1\right)}}{Z_{t+1}} = \frac{D^{(t)}(i) \times \left(\frac{1}{\epsilon_t} - 1\right)^{\frac{1}{2}}}{Z_{t+1}} = \frac{D^{(t)}(i) \times \sqrt{\frac{1-\epsilon_t}{\epsilon_t}}}{Z_{t+1}}$$

Solution.

$$\frac{m(1+\epsilon)}{4} + \frac{m(1-\epsilon)}{4} = \frac{m}{2} \text{ points with + label}$$
$$\frac{m}{4} + \frac{m}{4} = \frac{m}{2} \text{ points with } - \text{ label}$$

The probability distribution of the training point $(-1,1)$ with label $+$ is $\dfrac{\frac{m(1+\epsilon)}{4}}{m} = \dfrac{1+\epsilon}{4}$. For point $(1,-1)$, we obtain $\dfrac{1-\epsilon}{4}$, for points $(1,1)$ and $(-1,-1)$ with label $-$ we obtain $\frac{1}{4}$.

The initial problem with m points in the training sample is similar with the problem with 4 points with the corresponding probabilities.

$$S = \left\{ (\underbrace{(-1,+1)}_{point}, \underbrace{+1}_{label}), ((+1,-1),+1), ((+1,+1),-1), ((-1,-1),-1) \right\}$$

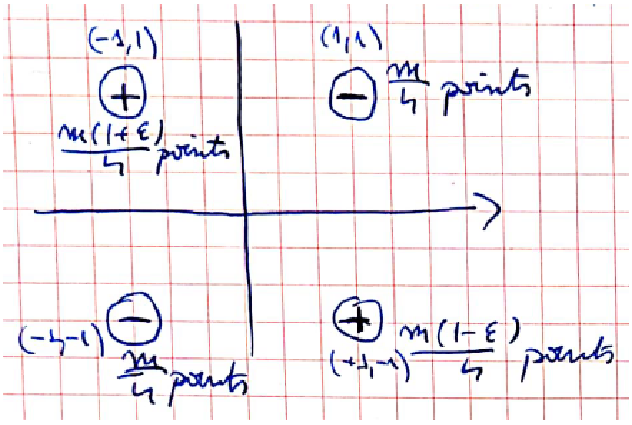


Figure 1: Representation of the m points in the plane.

$$D^{(1)}\colon \left(\begin{array}{cccc} (-1,1) & (1,-1) & (1,1) & (-1,-1) \\ \dfrac{1+\epsilon}{4} & \dfrac{1-\epsilon}{4} & \dfrac{1}{4} & \dfrac{1}{4} \end{array} \right)$$

Base hypothesis class = decision stumps in $\mathbb{R}^2$.

$$\mathcal{H}^2_{\mathrm{DS}} = \left\{ h_{i,\theta,b}\colon \mathbb{R}^2 \to \{-1,1\},\ h_{i,\theta,b}(x_1,x_2) = \operatorname{sign}(\theta - x_i)\cdot b \quad \begin{array}{l} 1 \le i \le 2 \\ \theta \in \mathbb{R} \\ b \in \{+1,-1\} \end{array} \right\}$$

= pick a coordinate i (1 or 2), project the input $x = (x_1, x_2)$ on the i-th coordinate and obtain x_i if $x_i \le \theta$, label the example x_i with label b, else with label $-b$

For our problem, we can see that we can take a set of representation thresholds θ to be $\theta = \{-2, 0, 2\}$. So we have at most 12 base classifiers: $h_{1,-2,1}; h_{1,-2,-1}; \ldots; h_{2,2,-1}$

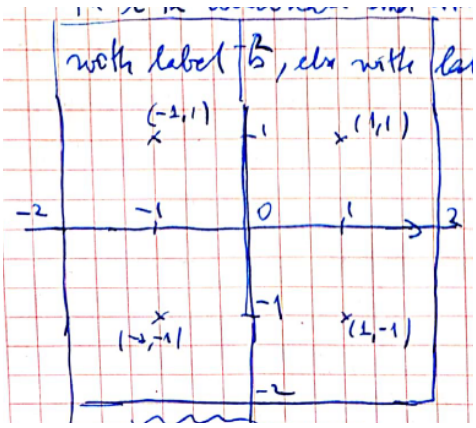


Figure 2: There are 12 base classifiers decision stumps in the plae for our problem: $h_{1,-2,1}; h_{1,-2,-1}; \ldots; h_{2,2,-1}$).

$h_{1,-2,+1} \to$ project on x_1, compare to -2, all points < -2 get label $+1$, all other get label -1

$h_{1,-2,-1} \to$ project on x_1, compare to -2, all points < -2 get label -1, all other get label $+1$

$h_{1,+2,+1} \to$ project on x_1, compare to $+2$, all points $< +2$ get label $+1$, all other get label -1

So we see that on our training set $h_{1,-2,-1}$ and $h_{1,+2,+1}$ will have the same behavior (all points will receive label $+1$).

If we analyze the behavior of all 12 base classifiers (decision stumps in $\mathbb{R}^2$), we will see that in the end there are only 6 unique base classifiers.

$$\underset{h^1}{\begin{array}{c|c} + & + \\ \hline + & + \end{array}} \qquad \underset{h^2}{\begin{array}{c|c} - & - \\ \hline - & - \end{array}} \qquad \underset{h^3}{\begin{array}{c|c} + & - \\ \hline + & - \end{array}} \qquad \underset{h^4}{\begin{array}{c|c} - & + \\ \hline - & + \end{array}} \qquad \underset{h^5}{\begin{array}{c|c} - & - \\ \hline + & + \end{array}} \qquad \underset{h^6}{\begin{array}{c|c} + & + \\ \hline - & - \end{array}}$$

So we have $B = \{h^1, h^2, h^3, h^4, h^5, h^6\}$.

<u>Round 1</u>

- distribution $D^{(1)}$: $\begin{pmatrix} (-1,1) & (1,-1) & (1,1) & (-1,-1) \\ \dfrac{1+\epsilon}{4} & \dfrac{1-\epsilon}{4} & \dfrac{1}{4} & \dfrac{1}{4} \end{pmatrix}$

- select the best classifier from $\mathcal{H}$, the one with minimum empirical risk

$$\begin{aligned}
L_{D^{(1)}}(h^1) &= \frac{1}{4} + \frac{1}{4} = \frac{1}{2} \\
L_{D^{(1)}}(h^2) &= \frac{1+\epsilon}{4} + \frac{1-\epsilon}{4} = \frac{1}{2} \\
L_{D^{(1)}}(h^3) &= \frac{1}{4} + \frac{1-\epsilon}{4} = \frac{1}{2} - \frac{\epsilon}{4} \\
L_{D^{(1)}}(h^4) &= \frac{1+\epsilon}{4} + \frac{1}{4} = \frac{1}{2} + \frac{\epsilon}{4} \\
L_{D^{(1)}}(h^5) &= \frac{1+\epsilon}{4} + \frac{1}{4} = \frac{1}{2} + \frac{\epsilon}{4} \\
L_{D^{(1)}}(h^6) &= \frac{1}{4} + \frac{1-\epsilon}{4} = \frac{1}{2} - \frac{\epsilon}{4}
\end{aligned}$$

So, the minimum achievable error is $\frac{1}{2} - \frac{\epsilon}{4}$ and it is attained by base classifiers h^3 and h^6. Let's choose h^3 as our weak classifier: $h^3 = h_{1,0,+1}$.

So, for $t = 1$ (round 1) we have $h_t = h^3 = h_{1,0,+1}$.

The error of the base classifier is $\epsilon_1 = \frac{1}{2} - \frac{\epsilon}{4}$.

$$w_1 = \frac{1}{2} \ln\left(\frac{1}{\epsilon_1} - 1\right) = \frac{1}{2}\left(\ln\left(\frac{4}{2-\epsilon} - 1\right)\right) = \ln\left(\frac{2+\epsilon}{2-\epsilon}\right)^{\frac{1}{2}} = \ln\sqrt{\frac{2+\epsilon}{2-\epsilon}}$$

Based on $D^{(1)}$ we will build $D^{(2)}$. Examples correctly classified at round 1 will have now the weight decreased, examples misclassified at round 1 will have their weight increased.

$$\begin{aligned}
D^{(2)}((-1,+1)) &= \frac{1}{Z_2} D^{(1)}((-1,+1)) \cdot \sqrt{\frac{\epsilon_1}{1-\epsilon_1}} = \frac{1}{Z_2} \cdot \left(\frac{1+\epsilon}{4}\right) \cdot \sqrt{\frac{2-\epsilon}{2+\epsilon}} \quad \searrow \\
D^{(2)}((+1,-1)) &= \frac{1}{Z_2} \cdot \left(\frac{1-\epsilon}{4}\right) \cdot \sqrt{\frac{2+\epsilon}{2-\epsilon}} \quad \nearrow \\
D^{(2)}((+1,+1)) &= \frac{1}{Z_2} \cdot \frac{1}{4} \cdot \sqrt{\frac{2-\epsilon}{2+\epsilon}} \quad \searrow \\
D^{(2)}((-1,-1)) &= \frac{1}{Z_2} \cdot \frac{1}{4} \cdot \sqrt{\frac{2+\epsilon}{2-\epsilon}} \quad \nearrow
\end{aligned}$$

We can find the value of Z_2 such that $D^{(2)}$ is a probability distribution, meaning that the sum of probability mass should be equal to 1.

$$D^{(2)}((-1,+1)) + D^{(2)}((+1,-1)) + D^{(2)}((+1,+1)) + D^{(2)}((-1,-1)) = 1$$

$$\Rightarrow Z_2 = \frac{1+\epsilon}{4} \cdot \sqrt{\frac{2-\epsilon}{2+\epsilon}} + \frac{1-\epsilon}{4} \cdot \sqrt{\frac{2+\epsilon}{2-\epsilon}} + \frac{1}{4} \cdot \sqrt{\frac{2-\epsilon}{2-\epsilon}} + \frac{1}{4} \cdot \sqrt{\frac{2+\epsilon}{2+\epsilon}}$$
$$= \frac{1}{4} \cdot \sqrt{\frac{2-\epsilon}{2+\epsilon}} \cdot \left((1+\epsilon) + (1-\epsilon) \cdot \frac{2+\epsilon}{2-\epsilon} + 1 + \frac{2+\epsilon}{2-\epsilon} \right)$$
$$= \frac{1}{4} \cdot \sqrt{\frac{2-\epsilon}{2+\epsilon}} \cdot \frac{(1+\epsilon)\cdot(2-\epsilon) + (1-\epsilon)\cdot(2+\epsilon) + (2-\epsilon) + 2 + \epsilon}{2-\epsilon}$$
$$= \frac{1}{4} \cdot \sqrt{\frac{2-\epsilon}{2+\epsilon}} \cdot \frac{2+\epsilon-\epsilon^2+2-\epsilon-\epsilon^2+2-\epsilon+2+\epsilon}{2-\epsilon}$$
$$= \frac{1}{4} \cdot \sqrt{\frac{2-\epsilon}{2+\epsilon}} \cdot \frac{8-2\epsilon^2}{2-\epsilon} = \frac{1}{4} \cdot \sqrt{\frac{2-\epsilon}{2+\epsilon}} \cdot \frac{2(2-\epsilon)(2+\epsilon)}{2-\epsilon}$$
$$= \frac{1}{2} \cdot \sqrt{(2-\epsilon)(2+\epsilon)}$$

$$\Rightarrow D^{(2)}((-1,+1)) = \frac{1+\epsilon}{2(2+\epsilon)}$$
$$D^{(2)}((+1,-1)) = \frac{1-\epsilon}{2(2-\epsilon)}$$
$$D^{(2)}((+1,+1)) = \frac{1}{2(2+\epsilon)}$$
$$D^{(2)}((-1,-1)) = \frac{1}{2(2-\epsilon)}$$

What is the error of the base classifier $h^3 = h_{1,0,+1}$ selected at round 1 on $D^{(2)}$?

$$\text{Loss}(h^3) = \frac{1}{2(2-\epsilon)} + \frac{1-\epsilon}{2(2-\epsilon)} = \frac{2-\epsilon}{2(2-\epsilon)} = \frac{1}{2}$$

Round 2

- distribution $D^{(2)}$: $\begin{pmatrix} (-1,1) & (1,-1) & (1,1) & (-1,-1) \\ \frac{1+\epsilon}{2(2+\epsilon)} & \frac{1-\epsilon}{2(2-\epsilon)} & \frac{1}{2(2+\epsilon)} & \frac{1}{2(2-\epsilon)} \end{pmatrix}$

- select the best classifier from $\mathcal{H}$, the one with minimum empirical risk

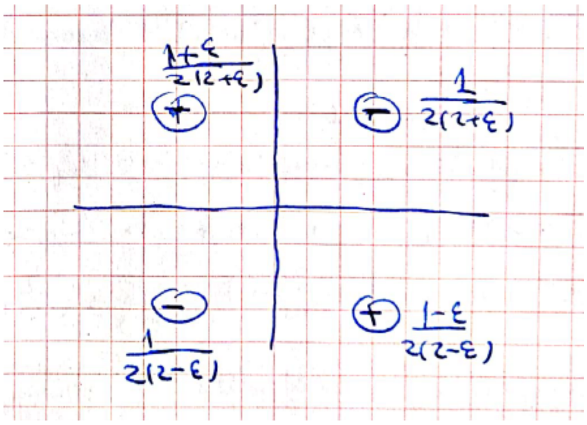


Figure 3: Updated distribution $D^{(2)}$ of samples after round 1 of AdaBoost.

$$
\begin{aligned}
L_{D^{(2)}}(h^1) &= \frac{1}{2(2-\epsilon)} + \frac{1}{2(2+\epsilon)} = \frac{2+\epsilon+2-\epsilon}{2(2-\epsilon)(2+\epsilon)} = \frac{2}{(2-\epsilon)(2+\epsilon)} = \frac{4}{2(2-\epsilon)(2+\epsilon)} \\
L_{D^{(2)}}(h^2) &= \frac{1+\epsilon}{2(2+\epsilon)} + \frac{1-\epsilon}{2(2-\epsilon)} = \frac{(1+\epsilon)\cdot(2-\epsilon)+(1-\epsilon)\cdot(2+\epsilon)}{2(2-\epsilon)(2+\epsilon)} = \frac{4-2\epsilon^2}{2(2-\epsilon)(2+\epsilon)} \\
L_{D^{(2)}}(h^3) &= \frac{1}{2} = \frac{4-\epsilon^2}{2(2-\epsilon)(2+\epsilon)} \\
L_{D^{(2)}}(h^4) &= \frac{1}{2} = \frac{4-\epsilon^2}{2(2-\epsilon)(2+\epsilon)} \\
L_{D^{(2)}}(h^5) &= \frac{1+\epsilon}{2(2+\epsilon)} + \frac{1}{2(2-\epsilon)} = \frac{(1+\epsilon)\cdot(2-\epsilon)+2+\epsilon}{2(2+\epsilon)(2-\epsilon)} = \frac{4+2\epsilon-\epsilon^2}{2(2+\epsilon)(2-\epsilon)} \\
L_{D^{(2)}}(h^6) &= \frac{1}{2(2+\epsilon)} + \frac{1-\epsilon}{2(2-\epsilon)} = \frac{(2-\epsilon)+(1-\epsilon)\cdot(2+\epsilon)}{2(2+\epsilon)(2-\epsilon)} = \frac{4-2\epsilon-\epsilon^2}{2(2+\epsilon)(2-\epsilon)}
\end{aligned}
$$

The smallest error is attained by h^6. This is the base classifier selected at the current round. So, for $t = 2$ (round 2) we have $h_2 = h^6 = h_{2,0,-1}$.

$$
\begin{aligned}
\epsilon_2 &= \frac{4-2\epsilon-\epsilon^2}{2(2+\epsilon)(2-\epsilon)} \\
w_2 &= \frac{1}{2}\ln\left(\frac{1}{\epsilon_2}-1\right) = \frac{1}{2}\ln\left(\frac{4-\epsilon^2+2\epsilon}{4-\epsilon^2-2\epsilon}\right)
\end{aligned}
$$

$\square$

Advanced Machine Learning Seminar 5

Exercise 1 (exercise 8.1 in the book)

Let $\mathcal{H}$ be the class of intervals on the line (formally equivalent to axis aligned rectangles in dimension $n = 1$). Propose an implementation of the $\text{ERM}_{\mathcal{H}}$ learning rule (in the agnostic case) that given a training set of size m, runs in time $\mathcal{O}(m^2)$. Hint: Use dynamic programming.

Solution.

$$\mathcal{H}_{\text{intervals}} = \mathcal{H}^1_{rec} = \left\{ h_{a,b} \colon \mathbb{R} \to \mathbb{R},\ h_{a,b} = \mathbb{1}_{[a,b]},\ h_{a,b}(x) = \begin{cases} 1 & x \in [a,b] \\ 0 & \text{otherwise} \end{cases},\ a, b \in \mathbb{R} \right\}$$

Consider a training set S of size m:

$S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m) \mid x_i \in \mathbb{R},\ y_i \in \{0, 1\}, i = \overline{1, m}\}$

Propose an implementation of the $\text{ERM}_{\mathcal{H}}$ learning rule in the agnostic case that runs in $\mathcal{O}(m^2) \Leftrightarrow$ find a hypothesis h_{a_S, b_S} with the smallest empirical risk.

Example:

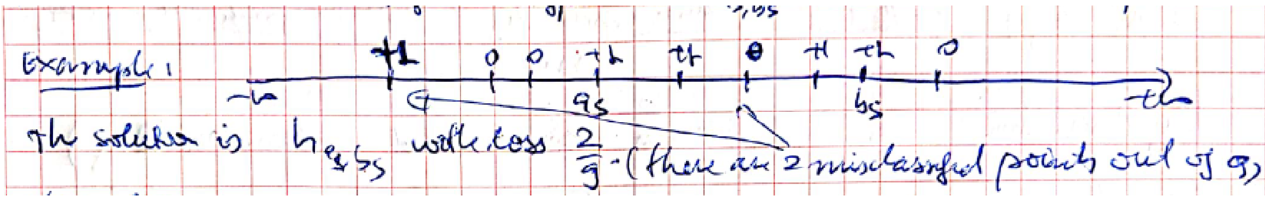


Figure 1: Example for agnostic case: 9 points scattered on the real line with some labels (5 positives and 4 negatives).

The solution for the example in Figure 2 is h_{a_S, b_S} with loss $\dfrac{2}{9}$ (there are 2 misclassified points out of 9).

Observations

1. We are in the agnostic case:
   - it might be the case that there is no labeling function but instead we are dealing with a distribution (same point might have different labels);
   - if there is a labeling function, it might not be in $\mathcal{H}_{\text{intervals}}$
2. If all points are negative, we should return an interval not containing any point in S
3. If all points are positive, we should return an interval containing all points in S

We will first sort the training set S in ascending order of $x's$.

We obtain $S = \{(x_{\sigma(1)}, y_{\sigma(1)}), (x_{\sigma(2)}, y_{\sigma(2)}), \ldots, (x_{\sigma(m)}, y_{\sigma(m)})\}$ with $x_{\sigma(1)} \leq x_{\sigma(2)} \leq \cdots \leq x_{\sigma(m)}$.

As we are in the agnostic case, we can have $x_{\sigma(i)} = x_{\sigma(i+1)}$ and $y_{\sigma(i)} \neq y_{\sigma(i+1)}$.

Consider the set Z containing the values of x' with no repetition:

$$Z = \{z_1, z_2, \ldots, z_n\}$$
$$z_1 = x_{\sigma(1)} < z_2 < \cdots < z_n = x_{\sigma(m)} \quad n \leq m$$

If all initial x values are different, then $z_1 = x_{\sigma(1)}, \ldots, z_n = x_{\sigma(m)}, n = m$.

Idea of the implementation of $\text{ERM}_{\mathcal{H}}$

1. If all $y_i = 0$, return an interval not containing any point x: $[z_1 - 2, z_1 - 1]$.

2. Consider all possible intervals $Z_{i,j} = [z_i, z_j] \qquad i = \overline{1,n}, j = \overline{i,n}$

There are $n + (n-1) + (n-2) + \cdots + 1 = \dfrac{n(n+1)}{2}$ such intervals.

Determine the interval $Z^* = Z_{i^*,j^*}$ with the smallest empirical risk. $Z_{i^*,j^*} = \underset{\substack{i=\overline{1,n} \\ j=\overline{i,m}}}{\text{argmin}} \text{Loss}(Z_{i,j})$

How to compute very fast $\text{Loss}(Z_{i,j})$? Use dynamic programming!

$\text{Loss}(Z_{i,j}) = \dfrac{\#\text{ negative points inside } Z_{i,j} + \#\text{ positive points outside } Z_{i,j}}{m}$

Key observation: $\text{Loss}(Z_{i,j+1})$ can be computed based on $\text{Loss}(Z_{i,j})$.

Simple case: there is just one point (x_k, y_k) in the training set S such that $x_k = z_{j+1}$.

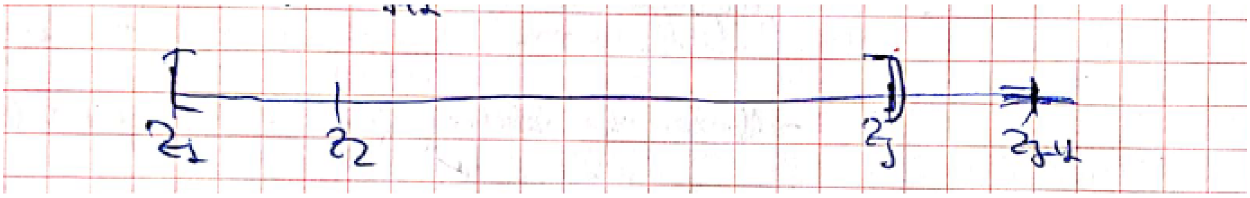


Figure 2: Sorted values $z_1, z_2, \ldots, z_{j+1}$.

$$\text{If } y_k = +1 \text{ then } \text{Loss}(Z_{i,j+1}) = \text{Loss}(Z_{i,j}) - \frac{1}{m} \quad \text{(the loss decreases)}$$

$$\text{If } y_k = 0 \quad \text{then } \text{Loss}(Z_{i,j+1}) = \text{Loss}(Z_{i,j}) + \frac{1}{m} \quad \text{(the loss increases)}$$

General case (in the agnostic scenario)

We have multiple points $x_{k_1}, x_{k_2}, \ldots, x_{k_l} = z_{j+1}$ (l points)

Then: some of the points will have label $1 = p_{j+1}$ some of the points will have label $0 = n_{j+1}$ $p_{j+1} + n_{j+1} = l$

In this case we have that:

$$\text{Loss}(Z_{i,j+1}) = \text{Loss}(Z_{i,j}) - \frac{p_{j+1}}{m} + \frac{n_{j+1}}{m}$$

as p_{j+1} points will be labeled correctly now and n_{j+1} points will be labeled incorrectly now (if $l = 1$, we have $p_{j+1} + n_{j+1} = 1$, so we have just one point labeled positive or negative)

Efficient implementation of the $\text{ERM}_{\mathcal{H}}$ rule for $\mathcal{H}_{\text{intervals}}$

1. Sort S and obtain $x_{\sigma(1)} \le x_{\sigma(2)} \le \cdots \le x_{\sigma(m)}$. Build set Z containing value x without repetition: $Z = \{z_1, z_2, \ldots, z_n\}$, $z_1 = x_{\sigma(1)} < z_2 < \cdots < z_n = x_{\sigma(m)}$
2. Check if all y_i $i = \overline{1,m}$ have value 0. If so, return h_{a_S,b_S}, where $a_S = z_1 - 2$, $b_S = z_1 - 1$. Compute $P = \sum_{i=1}^m y_i$ (# positive examples)
3. For $j = \overline{1,n}$

   compute values $\quad p_j = \;\#$ points $x_i = z_j$ with label $y_i = 1$
   $\qquad\qquad\qquad\quad n_j = \;\#$ points $x_i = z_j$ with label $y_i = 0$
4. min_error $= \frac{m}{m} = 1$, $i^* = []$, $j^* = []$

   for $i = \overline{1,m}$
       for $j = \overline{i,n}$
           $Z_{i,j} = [z_i, z_j]$
           if $(j == i)$
               $\text{Loss}(Z_{i,j}) = \dfrac{P - p_j + n_j}{m}$
           else
               $\text{Loss}(Z_{i,j}) = \text{Loss}(Z_{i,j-1}) + \dfrac{n_j - p_j}{m}$
           if $\text{Loss}(Z_{i,j}) <$ min_error
               min_error $= \text{Loss}(Z_{i,j})$
               $i^* = i$
               $j^* = j$

5. Return i^*, j^*

Complexity:

1. sorting $\mathcal{O}(m \cdot \log m)$
2. computing $P - \mathcal{O}(m)$
3. computing $p_j, n_j - \mathcal{O}(m)$
4. $\text{Loss}(Z_{i,j})$ = constant time

Total: $\mathcal{O}(m^2)$ □

Exercise 2 Let $\mathcal{X} = \mathbf{R}$ and consider $\mathcal{H}$ the class of 3-piece classifiers (signed intervals):

$$\mathcal{H} = \{h_{a,b,s} \colon \mathbf{R} \to \{-1, 1\},\ a \leq b,\ s \in \{-1, +1\}\}$$
$$\text{where } h_{a,b,s}(x) = \begin{cases} s & \text{if } x \in [a,b] \\ -s & \text{if } x \notin [a,b] \end{cases}$$

Give an efficient ERM algorithm for class $\mathcal{H}$ and compute its complexity for each of the following cases:

a. realizable case.

b. agnostic case.

Solution. a. realizable case
There exists a function $h_{a^*,b^*,s^*} \in \mathcal{H}$ that labels the training points

$$S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\} \qquad y_i = h_{a^*,b^*,s^*}(x_i)$$

We can have the following possibilities for examples appearing in S:

$$\begin{array}{ll} + + + + + + + + + & \text{(only positive examples)} \\ - - - - - - - - - & \text{(only negative examples)} \\ + + + - - - - + ++ & \\ - - - + + + - - -- & \\ + + + + - - - - -- & \\ - - - - - + + + ++ & \end{array}$$

Consider the following algorithm

$$\begin{aligned} \text{Initialization:} \quad & a_+ = -\infty \qquad a_- = -\infty \\ & b_+ = +\infty \qquad b_- = +\infty \\ \text{Compute } a_+ &= \min_{\substack{i=\overline{1,m} \\ y_i=+1}} x_i \qquad \text{if there is no } x_i \text{ with } y_i = +1, \text{ then } a_+ = -\infty \\ b_+ &= \max_{\substack{i=\overline{1,m} \\ y_i=+1}} x_i \qquad \text{if there is no } x_i \text{ with } y_i = +1, \text{ then } b_+ = +\infty \\ a_- &= \min_{\substack{i=\overline{1,m} \\ y_i=-1}} x_i \qquad \text{if there is no } x_i \text{ with } y_i = -1, \text{ then } a_- = -\infty \\ b_- &= \max_{\substack{i=\overline{1,m} \\ y_i=-1}} x_i \qquad \text{if there is no } x_i \text{ with } y_i = -1, \text{ then } b_- = +\infty \end{aligned}$$

If $a_+ < a_-$ return $h_{a_-,b_-,-1}$
else return $h_{a+,b+,+1}$

b. agnostic case
Can think of $\mathcal{H}_{\text{signedintervals}} = \mathcal{H}^+_{\text{intervals}} \cup \mathcal{H}^-_{\text{intervals}}$

$$\mathcal{H}^+_{\text{intervals}} = \left\{ h^+_{a,b} \colon \mathbb{R} \to \{-1, 1\},\ a \le b,\ h^+_{a,b}(x) = \begin{cases} 1 & x \in [a,b] \\ -1 & x \notin [a,b] \end{cases} \right\}$$

$$\mathcal{H}^-_{\text{intervals}} = \left\{ h^-_{a,b} \colon \mathbb{R} \to \{-1, 1\},\ a \le b,\ h^-_{a,b}(x) = \begin{cases} -1 & x \in [a,b] \\ 1 & x \notin [a,b] \end{cases} \right\}$$

Use the algorithm in exercise 1 (efficient implementation of the $\mathrm{ERM}_{\mathcal{H}}$ rule) and run it for $\mathcal{H}^+_{\text{intervals}}$ and $\mathcal{H}^-_{\text{intervals}}$.

Obtain the hypotheses $h^+_{a^*,\,b^*}$ and $h^-_{c^*,\,d^*}$.

Choose the one with the minimum empirical risk. □

Exercise 3 (exercise 10.1 in the book)

Boosting the Confidence: Let A be an algorithm that guarantees the following: There exist some constant $\delta_0 \in (0,1)$ and a function $m_{\mathcal{H}} \colon (0,1) \to \mathbb{N}$ such that, for every $\epsilon \in (0,1)$, if $m \ge m_{\mathcal{H}}(\epsilon)$, then, for every distribution $\mathcal{D}$, it holds that, with probability of at least $1-\delta_0$, $L_{\mathcal{D}}(A(S)) \le \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \epsilon$.

Suggest a procedure that relies on A and learns $\mathcal{H}$ in the usual agnostic PAC learning model and has a sample complexity of

$$m_{\mathcal{H}}(\epsilon,\delta) \le k\, m_{\mathcal{H}}(\epsilon/2) + \left\lceil \frac{2\log(4k/\delta)}{\epsilon^2} \right\rceil$$

where

$$k = \lceil \log(\delta/2)/\log(\delta_0) \rceil$$

Hint: Divide tha data into $k+1$ chunks, where each of the first k chunks is of size $m_{\mathcal{H}}(\epsilon/2)$ examples. Train the first k chunks using A. Argue that the probability that for all these chunks we have $L_{\mathcal{D}}(A(S)) > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \epsilon$ is at most $\delta_0^k \le \delta/2$. Finally, use the last chunk to choose from the k hypotheses that A generated from the k chunks (by relying on Corollary 4.6).

Corollary 4.6. *Let $\mathcal{H}$ be a finite hypothesis class, let Z be a domain, ane let $\ell \colon \mathcal{H} \times Z \to [0,1]$ be a loss function. Then, $\mathcal{H}$ enjoys the uniform convergence property with sample complexity*

$$m^{UC}_{\mathcal{H}}(\epsilon,\delta) \le \left\lceil \frac{\log(2|\mathcal{H}|/\delta)}{2\epsilon^2} \right\rceil$$

Furthermore, the class is agnostically PAC learnable using the ERM algorithm with sample complexity

$$m_{\mathcal{H}}(\epsilon,\delta) \le m^{UC}_{\mathcal{H}}(\epsilon/2,\delta) \le \left\lceil \frac{2\log(2|\mathcal{H}|/\delta)}{\epsilon^2} \right\rceil$$

Solution. A algorithm with the following property: $\exists\, \delta_0 \in (0,1)$ and $m_{\mathcal{H}} \colon (0,1) \to \mathbb{N}$ such that for every $\epsilon \in (0,1)$ if $m \ge m_{\mathcal{H}}(\epsilon)$ then for every distribution $\mathcal{D}$ it holds

$$\underset{S\sim\mathcal{D}^m}{P}\left(L_{\mathcal{D}}(A(S)) \le \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \epsilon \right) \ge 1-\delta_0$$

Suggest a procedure based on algorithm A that learns $\mathcal{H}$ in the agnostic PAC setting and has a sample complexity of

$$m_{\mathcal{H}}(\epsilon,\delta) \le k * m_{\mathcal{H}}(\epsilon/2) + \left\lceil \frac{2\log(4k/\delta)}{\epsilon^2} \right\rceil \qquad \text{where } k = \left\lceil \frac{\log \delta/2}{\log \delta_0} \right\rceil$$

Definition of agnostic PAC: $\mathcal{H}$ ia agnostic PAC if there exists a function $m_{\mathcal{H}} \colon (0,1)^2 \to \mathbb{N}$ and a learning algorithm A' with the following property: $\forall\, \epsilon > 0$, $\forall\, \delta > 0$, $\forall\, \mathcal{D}$ distribution function over $Z = \mathcal{X} \times \{0,1\}$ when we run the algorithm A' on a training set S of $m \ge m_{\mathcal{H}}(\epsilon,\delta)$ examples sampled i.i.d. from $\mathcal{D}$, A' returns $h_S = A'(S)$ such that

$$\underset{S\sim\mathcal{D}^m}{P}\left(L_{\mathcal{D}}(h_S) \le \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \epsilon \right) \ge 1-\delta$$

This is equivalent to:

$$\underset{S\sim\mathcal{D}^m}{P}\left(L_{\mathcal{D}}(h_S) > \underbrace{\min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \epsilon}_{\min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \frac{\epsilon}{2} + \frac{\epsilon}{2}} \right) < \delta$$

Follow the indications.

Let ϵ, $\delta \in (0,1)$. Pick k "chunks" $S_1, S_2, \ldots, S_k$ of size $m_{\mathcal{H}}(\frac{\epsilon}{2})$. Use the property of the algorithm A given.

$$\begin{aligned}
&\forall i = \overline{1,k} \qquad A(S_i) = h_i \\
&\underset{S_i\sim\mathcal{D}^{m_{\mathcal{H}}(\frac{\epsilon}{2})}}{P}\left(L_{\mathcal{D}}(h_i) \le \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h_i) + \frac{\epsilon}{2}\right) \ge 1-\delta_0 \\
\Leftrightarrow\ &\underset{S_i\sim\mathcal{D}^{m_{\mathcal{H}}(\frac{\epsilon}{2})}}{P}\left(L_{\mathcal{D}}(h_i) > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h_i) + \frac{\epsilon}{2}\right) < \delta_0 \quad \text{(the probability of having a bad } h_i\text{)}
\end{aligned}$$

The probability that all $h_i,\ i = \overline{1,k}$ are bad is given by:

$$P\left(L_{\mathcal{D}}(h_1) > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \frac{\epsilon}{2} \text{ and } L_{\mathcal{D}}(h_2) > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \frac{\epsilon}{2} \text{ and } \ldots \right) < (\delta_0)^k$$

$$\begin{aligned}
&\text{Find } k \text{ such that } \delta_0^k < \delta/2 \\
&\Leftrightarrow k\cdot \ln\delta_0 < \ln\frac{\delta}{2} \,\Big|\, :\ \ln\delta_0 \\
&k \ge \left\lceil \frac{\ln\delta - \ln 2}{\ln\delta_0} \right\rceil
\end{aligned}$$

Consider $\mathcal{H}' = \{h_1, h_2, \ldots, h_k\}$. $\mathcal{H}'$ finite, apply Corrolary (4.6).

If $m \ge m_{\mathcal{H}}^{UC}(\epsilon/2, \delta/2) \le \left\lceil \frac{2\log(4k/\delta)}{\epsilon^2} \right\rceil$ we have that

$$\underset{S_{k+1}\sim\mathcal{D}^{m_{\mathcal{H}}^{UC}(\epsilon/2,\delta/2)}}{P}\left(L_{\mathcal{D}}(h_{k+1}) > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \frac{\epsilon}{2}\right) < \frac{\delta}{2}$$

S_{k+1} has $\left\lceil \frac{2\log(4k/\delta)}{\epsilon^2} \right\rceil$ examples.

So: $L_{\mathcal{D}}(h_{k+1}) > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \epsilon$ if either we have

$$\begin{aligned}
\text{A: all } h_i \text{ are bad:} \quad & L_{\mathcal{D}}(h_i) > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \frac{\epsilon}{2} \\
\text{B: } h_{k+1} \text{ is bad:} \quad & L_{\mathcal{D}}(h_{k+1}) > \min_{h\in\mathcal{H}'} L_{\mathcal{D}}(h) + \frac{\epsilon}{2}
\end{aligned}$$

$P(A\cup B) \le P(A) \cup P(B) = \frac{\delta}{2} + \frac{\delta}{2} = \delta.$

So, take $m = k\cdot m_{\mathcal{H}}(\frac{\epsilon}{2}) + \left\lceil \frac{2\log(4k/\delta)}{\epsilon^2} \right\rceil$, $k = \left\lceil \frac{\ln\delta - \ln 2}{\ln\delta_0} \right\rceil$

$$\underset{(\underbrace{S_1, S_2, \ldots, S_k}_{h_1, h_2, \ldots, h_k}, \underset{h_{k+1}}{\overset{S_{k+1}}{\downarrow}})}{P}\left(L_{\mathcal{D}}(h_{k+1}) > \min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) + \epsilon\right) < \delta \quad \checkmark$$

$\square$

Advanced Machine Learning Seminar 4

Exercise 1 $\mathcal{H}^d_{\text{mcon}}$ = class of monotone Boolean conjunctions over $\{0,1\}^d$.

$$\mathcal{H}^d_{\text{mcon}} = \left\{ h\colon \{0,1\}^d \to \{0,1\},\, h_{(x_1,x_2,\ldots,x_d)} = \bigwedge_{i=1}^{d} l(x_i) \right\} \cup \{ \underset{\substack{h^-(x_1,\ldots,x_d)=0 \\ \text{always}}}{\underset{\downarrow}{h^-}} \}$$

$$l(x_i) \in \{\underset{\substack{\text{positive}\\ \text{literal}}}{\underset{\downarrow}{x_i}}, \underset{\substack{\text{missing}\\ \text{literal}}}{\underset{\downarrow}{1}}\}$$

So $|\mathcal{H}^d_{\text{mcon}}| = 2^d + 1$.
Examples:

$$d = 2 \qquad \mathcal{H}^2_{\text{mcon}} = \{ \underset{h^-}{\underset{\downarrow}{0}}, \underset{h_{empty}}{\underset{\downarrow}{1}}, x_1, x_2, x_1 \wedge x_2 \}$$

$$d = 3 \qquad \mathcal{H}^3_{\text{mcon}} = \{0, 1, x_1, x_2, x_3, x_1 \wedge x_2, x_1 \wedge x_3, x_2 \wedge x_3, x_1 \wedge x_2 \wedge x_3\}$$

We need to show that $VC\dim(\mathcal{H}^d_{\text{mcon}}) = d$.

Proof. We know that $|\mathcal{H}^d_{\text{mcon}}| = 2^d + 1$, so $VC\dim(\mathcal{H}^d_{\text{mcon}}) \le \lfloor \log_2(|\mathcal{H}^d_{\text{mcon}}|) \rfloor$, which in turn means

$$VC\dim(\mathcal{H}^d_{\text{mcon}}) \le \lfloor \log_2(2^d + 1) \rfloor = d$$

We only need to find a set $C \subseteq \{0,1\}^d$ with d points that is shattered by $\mathcal{H}^d_{\text{mcon}}$.

Usually, taking $C = \{e_1, e_2, \ldots, e_d\}, e_i = (0, \ldots, 0, 1, 0, \ldots, 0)$ works, but not for this $\mathcal{H} = \mathcal{H}^d_{\text{mcon}}$. You cannot have a conjunction that will have $h(e_1) = h(e_2) = 1$ and $h(e_3) = \cdots = h(e_d) = 0$.

Instead, we choose $C = \{(0,1,1,\ldots,1), (1,0,1,1,\ldots,1), \ldots, (1,1,\ldots,\underset{i}{0},1,1)\}$ set of vectors of the form $c_i = (1,1,\ldots,1) - e_i, i = \overline{1,d}$.

We want to show that, for each possible labeling $(y_1, y_2, \ldots, y_d)$ of the points $c_i = (1,1,\ldots,1) - e_i$, there exists a function $h \in \mathcal{H}^d_{\text{mcon}}$ such that $h(c_i) = y_i, \forall i = \overline{1,d}$.

Consider $(y_1, y_2, \ldots, y_d)$ a labeling and take $\mathcal{J} = \{j \mid y_j = 1\}$.

If $\mathcal{J} = \emptyset$, then h^- realizes the labeling $(0, 0, \ldots, 0)$.

If $\mathcal{J} = \{1, 2, \ldots, d\}$, then $h_{empty} = 1$ (all literals are missing) realizes the labeling $(1, 1, \ldots, 1)$.

If $1 \le |\mathcal{J}| \le d - 1$, then consider $h_{\mathcal{J}}(x_1, x_2, \ldots, x_d) = \bigwedge_{j \notin \mathcal{J}} x_j$.

For example, if $d = 4$ and $\mathcal{J} = \{2, 3\}$, $h_{\mathcal{J}}(x_1, x_2, x_3, x_4) = x_1 \wedge x_4$:

$$\begin{aligned}
h_{\mathcal{J}}(c_1) &= h_{\mathcal{J}}(0,1,1,1) = 0 \\
h_{\mathcal{J}}(c_2) &= h_{\mathcal{J}}(1,0,1,1) = 1 \\
h_{\mathcal{J}}(c_3) &= h_{\mathcal{J}}(1,1,0,1) = 1 \\
h_{\mathcal{J}}(c_4) &= h_{\mathcal{J}}(1,1,1,0) = 0
\end{aligned}$$

We have that $h_{\mathcal{J}}(c_i) = 1$ if $i \in \mathcal{J}$ and $h_{\mathcal{J}}(c_i) = 0$ if $i \notin \mathcal{J}$.

For all indices $i \in \mathcal{J}$, c_i will have value 0 on the position i and 1 in rest, but variable x_i is not considered in the conjunction. So $h_{\mathcal{J}}(c_i) = 1$.

For all indices $i \notin \mathcal{J}$, c_i will have value 0 on the position i and, because the conjunction contains the literal x_i, then we have that $h_{\mathcal{J}}(c_i) = 0$. $\square$

Exercise 2 $\mathcal{X} = \{0,1\}^n$

$$\mathcal{H}_{\text{n-parity}} = \left\{ h_I \mid I \subseteq \{1,2,\dots,n\},\ h_I(x_1,x_2,\dots,x_n) = \left(\sum_{i\in I} x_i\right) \mod 2 \right\}$$

What is $VC\dim(\mathcal{H}_{\text{n-parity}})$?

Proof. For each subset $I \subseteq \{1,2,\dots,n\}$ we have a h_I, so $|\mathcal{H}_{\text{n-parity}}| = 2^n$.

We know that $VC\dim(\mathcal{H}_{\text{n-parity}}) \leq \lfloor \log_2 2^n \rfloor = n$.

So $VC\dim(\mathcal{H}_{\text{n-parity}}) \leq n$.

Can we find a set C with n points that is shattered by $\mathcal{H}_{\text{n-parity}}$?

Let's try the "usual" set of unit vectors $C = \{e_1, e_2, \dots, e_n\}, e_i = (0,\dots,0,\underset{i}{1},0,\dots,0)$.

We want to show that, for each possible labeling $(y_1,y_2,\dots,y_n)$ of $(e_1,e_2,\dots,e_n)$, you can find a corresponding h such that $h(e_i) = y_i, \forall i = \overline{1,n}$.

Consider $(y_1,y_2,\dots,y_n)$ such a labeling and take $I = \{i \mid y_i = 1\}$.

Then we have

$$h_I(e_i) = \left(\sum_{i\in I} x_i\right) \mod 2 = \begin{cases} 1, & \text{if } i \in I \\ 0, & \text{otherwise} \end{cases}$$

So $VC\dim(\mathcal{H}_{\text{n-parity}}) = n$. □

Exercise 3 $\mathcal{X}$ – finite domain, $|\mathcal{X}| = n < \infty$, $k \leq |\mathcal{X}|$

3.1. $\mathcal{H}^{\mathcal{X}}_{=k} = \left\{ h \in \{0,1\}^{\mathcal{X}} \,\middle|\, |\{x\colon h(x) = 1\}| = k \right\}$
$=$ set of all functions that assign the value 1 to exactly k elements of $\mathcal{X}$

$VC\dim(\mathcal{H}^{\mathcal{X}}_{=k}) = ?$

Proof.

$$\begin{aligned}
&\text{If } k = 0 \Rightarrow \mathcal{H}^{\mathcal{X}}_{=0} = \{h^-\},\ \text{all points get the value } 0 \\
&\text{If } k = 1 \Rightarrow \mathcal{H}^{\mathcal{X}}_{=1} \text{ has } |\mathcal{X}| \text{ functions} = n \text{ functions} \\
&\qquad \mathcal{X} = \{x_1, x_2, \dots, x_n\}, n = |\mathcal{X}| \\
&\qquad h_i\colon \{x_1,\dots,x_n\} \to \{0,1\} \quad h_i(x_j) = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \\
&\text{If } k = 2 \Rightarrow \mathcal{H}^{\mathcal{X}}_{=2} \text{ has } C_n^2 \text{ elements} \\
&\vdots \\
&\text{If } k = n-1 \Rightarrow \mathcal{H}^{\mathcal{X}}_{=n-1} \text{ has } n \text{ elements} \\
&\text{If } k = n \Rightarrow \mathcal{H}^{\mathcal{X}}_{=n} \text{ has 1 element } h^+(x_i) = 1\ \forall i = \overline{1,n}
\end{aligned}$$

We first show that $VC\dim(\mathcal{H}^{\mathcal{X}}_{=k}) \leq \min(k, n-k) = VC\dim(\mathcal{H})$.

Case 1) if $n \geq 2k$, in this case, $\min(k, n-k) = k,\ k \leq \dfrac{n}{2}$

$\mathcal{H}$ will consist of functions h that label exactly k elements of $\mathcal{X}$ with label 1. So any set C with more than k points cannot be shattered because the labeling with all 1's $(1,1,1,\dots,1)$ cannot be realized by any $h \in \mathcal{H}$.

Case 2) if $n < 2k$, in this case $\min(k, n-k) = n-k,\ k > \dfrac{n}{2}$

$\mathcal{H}$ will consist of functions h that labels k elements of $\mathcal{X}$ with label 1, and $n-k$ points of $\mathcal{X}$ with label 0. So any set with more than $n-k+1$ points cannot be shattered by $\mathcal{H}$ as the labeling with all 0's $(0,0,\dots,0)$ cannot be realized by any $h \in \mathcal{H}$.

So we have that $VC\dim(\mathcal{H}) \leq \min(k, n-k)$.

We will prove that $VC\dim(\mathcal{H}) = \min(k, n-k)$.

Consider $k' = \min(k, n-k)$.

We need to show that there exists a set of points $A = \{x_{i1}, x_{i2}, \dots, x_{ik'}\} \subseteq \mathcal{X}$ that is shattered by $\mathcal{H}$. This means that, for each subset $B \subseteq A$, we can find $h_B \in \mathcal{H}$ such that

$$h_B(x) = \begin{cases} 1, & x \in B \\ 0, & x \in A \setminus B \end{cases}$$

We choose a set of $k-|B|$ points $B' = \{b_1, b_2, \ldots, b_{k-|B|}\} \subseteqq \mathcal{X} \setminus A$.

We can make the choice since $k - |B| \leq |\mathcal{X} \setminus A|$

$$\begin{aligned} k - |B| &\leq n - k' \\ k' - |B| &\leq n - k \\ k' - |B| &\leq k' \leq n - k \text{ this is true} \end{aligned}$$

So $B \subseteq A$ has $|B|$ elements
$B' \subseteq \mathcal{X} \setminus A$ has $k - |B|$ elements

So $|B \cup B'| = |B| + |B'|$ (as $B \cap B' = \emptyset$) $= k$
So, if we consider the characteristic function of the set $B \cup B'$, we have

$$\mathbb{1}_{B \cup B'}(x) = \begin{cases} 1, & x \in B \cup B' \\ 0, & \text{otherwise} \end{cases}$$

What is more important, $\mathbb{1}_{B \cup B'}$ takes value 1 for exactly k points, so it is a member of $\mathcal{H}$.
So, in this case, we take $h_B = \mathbb{1}_{B \cup B'}$.
h_B will have the desired property that $h_B(x) = \begin{cases} 1, & x \in B \\ 0, & x \in A \setminus B \end{cases}$
So any set A of $k' = min(k, n-k)$ points can be shattered by $\mathcal{H}$.
So $VC\dim(\mathcal{H}) = k' = \min(k, n-k)$. $\square$

3.2. $\mathcal{H}_{\text{at-most-}k} = \left\{h \in \{0,1\}^{\mathcal{X}} : |\{x\colon h(x)=1\}| \le k \text{ or } |\{x\colon h(x)=0\}| \le k\right\}$

Proof.

If $k=0$ $\quad \mathcal{H}_{\text{at-most-0}} = \{h^-, h^+\}$ $\quad$ where $|\{x\colon h^-(x)=1\}| \le 0$ and $|\{x\colon h^+(x)=0\}| \le 0$

If $k=1$ $\quad \mathcal{H}_{\text{at-most-1}} = \{h^-, h^+\} \cup \underbrace{\{\text{functions } h \text{ which label just one point with label } 1\}}_{\mathcal{H}^{\mathcal{X}}_{=1}}$

$\cup \{\text{functions } h \text{ which label just 1 point with label } 0\}$

<u>Case 1</u> If $n = |\mathcal{X}| \le 2k+1$, then we have that $\mathcal{H}_{\text{at-most-}k} = \{0,1\}^{\mathcal{X}} = \{h\colon \mathcal{X} \to \{0,1\}\}$

This is true because any function $h\colon \mathcal{X} \to \{0,1\}$ will have either at most k points labeled with 0 or at most k points labeled with 1.

$\Downarrow$ $h \in \mathcal{H}_{\text{at-most-}k}$

Example (see Table 1): Take $n=7,\ k=4$ $\quad \mathcal{X} = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$

	x	x_1	x_2	x_3	x_4	x_5	x_6	x_7
at most 4 1's or 4 0's $\leftarrow$	$h(x)$	1	1	0	0	1	0	1
at most 4 1's $\leftarrow$	$h(x)$	0	0	1	0	0	0	0
at most 4 1's $\leftarrow$	$h(x)$	0	1	0	0	1	0	0

Table 1

In this case, $VC\dim(\mathcal{H}_{\text{at-most-}k}) = VC\dim(\{0,1\})^{\mathcal{X}} = n = |\mathcal{X}|$.

<u>Case 2</u> If $n = |\mathcal{X}| \ge 2k+2$

We first show that $VC\dim(\underset{\mathcal{H}}{\underset{\shortparallel}{\mathcal{H}_{\text{at-most-}k}}}) = VC\dim(\mathcal{H}) \ge 2k+1$

Consider any set A of $2k+1$ points in $\mathcal{X}$: $A = \{a_1, a_2, \dots, a_{2k+1}\}$.

We will show that A is shattered by $\mathcal{H}$. It is enough to show that, for each possible labeling $(y_1, y_2, \dots, y_{2k+1})$ of the points $(a_1, a_2, \dots, a_{2k+1})$, we can find an $h \in \mathcal{H}$ such that $h(y_i) = a_i$.

Take $\mathcal{J} = \{j \mid y_j = 1\}$, and take $B_{\mathcal{J}} = \{a_j \in A \mid \underset{j \in \mathcal{J}}{y_j = 1}\}$

If $|\mathcal{J}| \le k$ $\quad$ we know that $\mathbb{1}_{B_{\mathcal{J}}} \in \mathcal{H}_{\text{at-most-}k}$, so we take

$$h = \mathbb{1}_{B_{\mathcal{J}}} \quad h(a_j) = \begin{cases} 1, & y_j = 1 \\ 0, & \text{otherwise} \end{cases}$$

If $|\mathcal{J}| > k$ $\quad$ take $C_{\mathcal{J}} = \{a_{\mathcal{J}} \in A \mid \underset{j \notin \mathcal{J}}{y_j = 0}\}$ has at most k elements, so we take in this case

$$h = \mathbb{1}_{A \setminus C_{\mathcal{J}}} \quad h(a_j) = \begin{cases} 1, & y_j = 1 \\ 0, & y_j = 0 \end{cases}$$

So, we have that $VC\dim(\mathcal{H}) \ge 2k+1$.

We show now that $VC\dim(\mathcal{H}) < 2k+2$.

Consider any set A of $2k+2$ points $\quad A = \{a_1, a_2, \dots, a_{2k+2}\}$.

There is no $h \in \mathcal{H}$ that will label the first $k+1$ points with 1 and the rest $k+1$ points with 0.

So, in conclusion, $VC\dim(\mathcal{H}_{\text{at-most-}k}) = \min(|\mathcal{X}|, 2k+1)$. $\square$

Advanced Machine Learning Seminar 3

Exercise 1 $\mathcal{H}$ – finite hypothesis class
$VC\dim(\mathcal{H}) \leq \lfloor \log_2(|\mathcal{H}|) \rfloor$ – upper bound

1. example of $\mathcal{H}$ infinite, $\mathcal{H}$ contains functions $h\colon [0,1] \to \{0,1\}$ and $VC\dim(\mathcal{H}) = 1$
Take $\mathcal{H}_{threshold}$ restricted to $[0,1]$

$$\mathcal{H}_{threshold,[0,1]} = \left\{h_a\colon [0,1] \to \{0,1\},\ h_a(x) = \mathbb{1}_{[x<a]},\ a \in [0,1]\right\}$$

$$h_a(x) = \begin{cases} 1, & 0 \leq x < a \leq 1 \\ 0, & \text{otherwise} \end{cases}$$

$VC\dim(\mathcal{H}_{threshold,[0,1]}) = 1$ (very similar proof with the one provided in lecture 6)

2. $\mathcal{H} = \{h_a, h_b\}$ – has only two functions h_a and h_b
Take $h_a, h_b \in \mathcal{H}_{threshold,[0,1]}$.
$h_a = h_{0.5}, h_b = h_{0.75}$
Take $A = \{0.6\}$. $\mathcal{H}$ shatters A because $\mathcal{H}_A$ has two functions $h_a(0.6) = 0$ and $h_b(0.6) = 1$.
$|\mathcal{H}_A| = 2^{|A|} = 2^1 = 2$
$\mathcal{H}$ cannot shatter any set A of min ≥ 2 points ($\mathcal{H}$ has only 2 functions).
So $VC\dim(\mathcal{H}) = 1 = \lfloor log_2(|\mathcal{H}|) \rfloor$

Exercise 2 $\mathcal{H}^d_{rec}$ – class of axis aligned rectangles in $\mathbb{R}^d$. In lecture 6 we proved that $VC\dim(\mathcal{H}^2_{rec}) = 4$. We want to show, in the general case, that $VC\dim(\mathcal{H}^d_{rec}) = 2d$.

$$\mathcal{H}^d_{rec} = \{h_{(a_1,b_1,a_2,b_2,\ldots,a_d,b_d)} \mid a_i \leq b_i, i = \overline{1,d}\}$$

$$h_{(a_1,b_1,a_2,b_2,\ldots,a_d,b_d)}(\underset{\underline{x}=(x^1,x^2,\ldots,x^d)}{\underline{x}}) = \begin{cases} 1, & a_i \leq x^i \leq b_i \quad \forall i = \overline{1,d} \\ 0, & \text{otherwise} \end{cases}$$

In order to show that $VC\dim(\mathcal{H}^d_{rec}) = 2d$, we need to show that:

1) there exists a set C of $2d$ points that is shattered by $\mathcal{H}^d_{rec}$
(this will mean that $VC\dim(\mathcal{H}^d_{rec}) \geq 2d$)

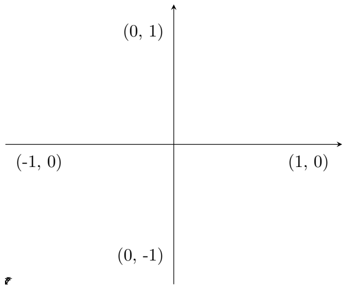
2) every set C of $2d+1$ points is not shattered by $\mathcal{H}^d_{rec}$ (this will mean that $VC\dim(\mathcal{H}^d_{rec}) < 2d+1$)

Let's prove 1).
Consider $C = \{c_1, c_2, c_3, \ldots, c_{2d-1}, c_{2d}\}$ where

$$\begin{aligned}
c_1 &= (1,0,0,\ldots,0) &&= e_1 \\
c_2 &= (0,1,0,\ldots,0) &&= e_2 \\
&\vdots \\
c_d &= (0,0,0,\ldots,1) &&= e_d \qquad c_i = e_i = -c_{i+d} \\
c_{d+1} &= (-1,0,0,\ldots,0) &&= -e_1 \qquad \forall i = \overline{1,d} \\
c_{d+2} &= (0,-1,0,\ldots,0) &&= -e_2 \\
&\vdots \\
c_{2d} &= (0,0,0,\ldots,-1) &&= -e_d
\end{aligned}$$

For $d = 2$, we will have in 2 dimensions:

$$c_1 = (1, 0) \qquad c_2 = (0, 1) \qquad c_3 = (-1, 0) \qquad c_4 = (0, -1)$$



We want to show that, for each labeling $(y_1, y_2, \ldots, y_{2d})$ of the points $(c_1, c_2, \ldots, c_{2d})$ (there are 2^{2d} possible labelings), there exists a function h in $\mathcal{H}^d_{rec}$ such that $h(c_i) = y_i \ \forall i = \overline{1, 2d}$.

Consider a labeling $(y_1, y_2, \ldots, y_{2d}) \in \{0, 1\}^{2d}$.

Each point c_i has all components $= 0$, apart from component i if $i \in \{1, \ldots, d\}$ or $i - d$ if $i \in \{d + 1, \ldots, 2d\}$.

$$\begin{array}{rl|rl|c|rl} c_1 & = (\ \ 1, 0, 0, \ldots, 0) & c_2 & = (0, \ \ 1, 0, \ldots, 0) & & c_d & = (0, 0, 0, \ldots, \ \ 1) \\ c_{d+1} & = (-1, 0, 0, \ldots, 0) & c_{d+2} & = (0, -1, 0, \ldots, 0) & \ldots & c_{2d} & = (0, 0, 0, \ldots, -1) \end{array}$$

We want to find $h = h_{(a_1,b_1,a_2,b_2,\ldots,a_d,b_d)}$ such that $h_{(a_1,b_1,a_2,b_2,\ldots,a_d,b_d)}(c_i) = y_i$.

The choice of the interval $[a_i, b_i]$ is influenced by the labels y_i and y_{i+d} of the points c_i and c_{i+d}. As all other points $c_1, c_2, \ldots, c_{i-1}, c_{i+1}, \ldots, c_{i+d-1}, c_{i+d+1}, \ldots, c_{2d}$ have 0 on the i-th component, we have that $[a_i, b_i]$ should contain 0, otherwise each point will be labeled with 0.

So $[a_i, b_i]$ depends on y_i and y_{i+d}, and $[a_i, b_i]$ decides basically the label of points c_i and c_{i+d}:

$$c_i = (0, \ldots, 0, 1, 0, \ldots, 0) \qquad c_{i+d} = (0, \ldots, 0, -1, 0, \ldots, 0)$$

Possible cases:

I $y_i = 0, y_{i+d} = 0$, then $[a_i, b_i] \cap \{-1, 1\} = \emptyset$
$[a_i, b_i]$ should not contain points -1 and 1.
In this case, take $a_i = -0.5, b_i = 0.5$ (many other choices are possible)

II $y_i = 0, y_{i+d} = 1$, then $[a_i, b_i] \cap \{-1, 1\} = \{-1\}$
$[a_i, b_i]$ should contain only point -1 such that c_{i+d} will get label 1.
In this case, take $a_i = -2, b_i = 0.5$ (many other choices are possible)

III $y_i = 1, y_{i+d} = 0$, then $[a_i, b_i] \cap \{-1, 1\} = \{1\}$
$[a_i, b_i]$ should contain only point $+1$ such that c_i will get label 1.
In this case, take $a_i = -0.5, b_i = 2$ (many other choices are possible)

IV $y_i = 1, y_{i+d} = 1$, then $[a_i, b_i] \cap \{-1, 1\} = \{-1, 1\}$
$[a_i, b_i]$ should contain both points $\{-1, 1\}$ such that c_i and c_{i+d} will get label 1.
In this case, take $a_i = -2, b_i = 2$ (many other choices are possible)

In all cases, we have that $h_{(a_1,b_1,a_2,b_2,\ldots,a_d,b_d)}(c_i) = y_i, \forall i = \overline{1, 2d}$, where each interval $[a_i, b_i]$ was determined based on y_i and y_{i+d}, $i = \overline{1, d}$.

So, $VC\dim(\mathcal{H}^d_{rec}) \geq 2d$.

$\square$

2) Let C be a set of size $2d + 1$ points. We will show that C cannot be shattered by $\mathcal{H}^d_{rec}$.

Because we have $2d + 1$ points in C and there are only d dimensions, there will exist a point $x = (x_1, x_2, \ldots, x_d) \in C$ such that, for each dimension $i = \overline{1, d}$ there will be 2 points x' and $x'' \in C$ such that $x'_i \leq x_i \leq x''_i$ (the point x_i is "inside" the rectangle determined by all other points in dimension i).

So the label for which x has value 0 and all other $2d$ points get label 1 cannot be realized by any function $h \in \mathcal{H}^d_{rec}$ (because x is inside the rectangle) that contain all other points.

$\square$

Exercise 3 $\mathcal{H}_{con}^d$ – class of Boolean conjunctions over the variables $x_1, x_2, \ldots, x_d, d \geq 2$

$$\mathcal{H}_{con}^d = \left\{ h \colon \{0,1\}^d \to \{0,1\},\ h(x_1, x_2, \ldots, x_d) = \bigwedge_{i=\overline{1,d}} l(x_i) \right\}$$
$$l(x_i) = \text{literal of variable } x_i$$
$$l(x_i) \in \{x_i, \overline{x_i}, \underset{missing}{1}\}$$

We also consider that $h^- \in \mathcal{H}_{con}^d$, $\quad h^-(x_1, x_2, \ldots, x_d) = 0$ always.

a) So $|\mathcal{H}_{con}^d| = 3^d + 1$.

b) $VC\dim(\mathcal{H}_{con}^d) \leq \lfloor \log_2(3^d + 1) \rfloor$

c) We will show that $\mathcal{H}_{con}^d$ shatters the set of unit vectors $\{e_i, i \leq d\}$

$e_i = (0, 0, \ldots, 0, \underset{i}{1}, 0, \ldots, 0)$

Consider $C = \{e_1, e_2, \ldots, e_d\}$. We want to prove that, for each possible labeling $(y_1, y_2, \ldots, y_d)$, there exists an $h \in \mathcal{H}_{con}^d$ such that $h(e_i) = y_i$.

Consider a labeling $(y_1, y_2, \ldots, y_d)$ and take $\mathcal{J} = \{j \mid y_j = 1\}$.

If $\mathcal{J} = \emptyset \Rightarrow h^-$ realizes the labeling $(0, 0, \ldots, 0)$.

If $\mathcal{J} = \{1, \ldots, d\} \Rightarrow h_{empty}$ (all literals are missing) $= 1\ \forall x_i$ realizes the labeling $(1, 1, \ldots, 1)$.

In all other cases, define

$$h_{\mathcal{J}} = \bigwedge_{j \notin \mathcal{J}} \overline{x_j} = \bigwedge_{j \in \{1,\ldots,d\} \setminus \mathcal{J}} \overline{x_j}$$

If $\mathcal{J} = \{1, 2, 4\}$, define $h_{\mathcal{J}} = \overline{x_3} \wedge \overline{x_5} \wedge \overline{x_6} \quad (d = 6)$.

$$h_{\mathcal{J}}(e_1) = h_{\mathcal{J}}(1,0,0,0,0,0) = \overline{0} \wedge \overline{0} \wedge \overline{0} = 1$$
$$h_{\mathcal{J}}(e_2) = h_{\mathcal{J}}(0,1,0,0,0,0) = \overline{0} \wedge \overline{0} \wedge \overline{0} = 1$$
$$h_{\mathcal{J}}(e_4) = h_{\mathcal{J}}(0,0,0,1,0,0) = \overline{0} \wedge \overline{0} \wedge \overline{0} = 1$$
$$h_{\mathcal{J}}(e_3) = h_{\mathcal{J}}(0,0,1,0,0,0) = \overline{1} \wedge \overline{0} \wedge \overline{0} = 0$$
$$h_{\mathcal{J}}(e_5) = h_{\mathcal{J}}(0,0,0,0,1,0) = \overline{0} \wedge \overline{1} \wedge \overline{0} = 0$$
$$h_{\mathcal{J}}(e_6) = h_{\mathcal{J}}(0,0,0,0,0,1) = \overline{0} \wedge \overline{0} \wedge \overline{1} = 0$$

So, $\quad h_{\mathcal{J}}(e_j) = 1$ if $j \in \mathcal{J}$

and $\quad h_{\mathcal{J}}(e_j) = 0$ if $j \notin \mathcal{J}$.

This proves that $\mathcal{H}_{con}^d$ shatters $C \Rightarrow VC\dim(\mathcal{H}_{con}^d) \geq d$.

d) We want to show that $VC\dim(\mathcal{H}^d_{con}) < d+1$.

Assume that there exists a set $C = \{c_1, c_2, \ldots, c_{d+1}\}$ of points from $\{0,1\}^d$ that is shattered by $\mathcal{H}^d_{con}$, so $|\mathcal{H}^d_{con_C}| = |\{h\colon C \to \{0,1\}, h \in \mathcal{H}\}| = 2^{d+1}$.

$$\begin{array}{ll}
c_1 \in \{0,1\}^d \Rightarrow c_1 = (c_1^1, c_1^2, \ldots, c_1^d) & \in \{0,1\}^d \\
c_2 \in \{0,1\}^d \Rightarrow c_2 = (c_2^1, c_2^2, \ldots, c_2^d) & \text{Each point } c_i \text{ has} \\
\ldots & d \text{ components from } \{0,1\} \\
c_i \in \{0,1\}^d \Rightarrow c_i = (c_i^1, c_i^2, \ldots, c_i^d) &
\end{array}$$

We want to find a contradiction and show that $\mathcal{H}^d_{con}$ doesn't shatter any set C of $d+1$ points.

If $\mathcal{H}^d_{con}$ shatters C, then among the 2^{d+1} function $h\colon C \to \{0,1\}$ we will have the following $d+1$ functions (for simplicity we will denote this functions with $h_1, h_2, \ldots, h_{d+1}$):

$$\begin{array}{ll}
h_1\colon \{0,1\}^d \to \{0,1\} \text{ such that} & h_1(c_1) = 0, h_1(c_2) = 1, h_1(c_3) = 1, \ldots,\ h_1(c_{d+1}) = 1 \\
h_2\colon \{0,1\}^d \to \{0,1\} \text{ such that} & h_2(c_1) = 1, h_2(c_2) = 0, h_2(c_3) = 1, \ldots,\ h_2(c_{d+1}) = 1 \\
h_3\colon \{0,1\}^d \to \{0,1\} \text{ such that} & h_3(c_1) = 1, h_3(c_2) = 1, h_3(c_3) = 0, \ldots,\ h_3(c_{d+1}) = 1 \\
\vdots & \\
h_{d+1}\colon \{0,1\}^d \to \{0,1\} \text{ such that} & h_{d+1}(c_1) = 0, h_{d+1}(c_2) = 1, h_{d+1}(c_3) = 1, \ldots,\ h_{d+1}(c_{d+1}) = 0
\end{array}$$

So h_i, with $i \in \{1, \ldots, d+1\}$ realizes the labels $(1, 1, \ldots, 1, \underset{i}{0}, 1, 1, \ldots, 1)$.

So we have
$$h_i(c_j) = \begin{cases} 0, & \text{if } i = j \\ 1, & \text{if } i \neq j \end{cases}$$

We will use the functions $h_1, h_2, \ldots, h_{d+1}$ to arrive at a contradiction. Each h_i is in $\mathcal{H}^d_{con}$, so it can be written as a conjunction of literals, where each literal from the writing of h_i can have three values for a variable x_k: $l_i(x_k) = \begin{cases} x_k, & \text{positive literal} \\ \overline{x_k}, & \text{negaitive literal} \\ 1, & \text{missing literal} \end{cases}$

For example, if we consider $d = 3$, a possible h from $\mathcal{H}^3_{con}$ could be $h = x_1 \wedge \overline{x_2}$, in this case $h = l(x_1) \wedge l(x_2) \wedge l(x_3)$ with $l(x_1) = x_1, \quad l(x_2) = \overline{x_2}, \quad l(x_3) = 1$.

In the general case, we have

$$h_i(x_1, x_2, \ldots, x_d) = \bigwedge_{k=1}^{d} l_i(x_k), \qquad l_i(x_k) \in \{x_k, \overline{x_k}, 1\}$$

Now, we go back to our $h_1, h_2, \ldots, h_{d+1}$.

$$\begin{array}{l}
h_1 \text{ realizes the labels } (0, 1, 1, 1, \ldots, 1) \text{ on } \{c_1, c_2, \ldots, c_{d+1}\} = C \\
h_2 \text{ realizes the labels } (1, 0, 1, 1, \ldots, 1) \text{ on } \{c_1, c_2, \ldots, c_{d+1}\} = C \\
\vdots \\
h_{d+1} \text{ realizes the labels } (1, 1, 1, 1, \ldots, 0) \text{ on } \{c_1, c_2, \ldots, c_{d+1}\} = C
\end{array}$$

We will use the labels 0 to come up with a contradiction.

$$\text{Because } h_1(c_1) = 0 \Leftrightarrow h_1(c_1^1, c_1^2, \ldots, c_1^d) = \bigwedge_{k=1}^{d} l_1(c_1^k) = l_1(c_1^1) \wedge l_1(c_1^2) \wedge \cdots \wedge l_1(c_1^d) = 0$$
$$\Rightarrow \exists k_1 \in \{1, \ldots, d\} \text{ such that } l_1(c_1^{k_1}) = 0$$
$$\text{Because } h_2(c_2) = 0 \Leftrightarrow h_2(c_2^1, c_2^2, \ldots, c_2^d) = \bigwedge_{k=1}^{d} l_2(c_2^k) = l_2(c_2^1) \wedge l_2(c_2^2) \wedge \cdots \wedge l_2(c_2^d) = 0$$
$$\Rightarrow \exists k_2 \in \{1, \ldots, d\} \text{ such that } l_2(c_2^{k_2}) = 0$$
$\ldots$
$$\text{Because } h_{d+1}(c_{d+1}) = 0 \Leftrightarrow h_{d+1}(c_{d+1}^1, c_{d+1}^2, \ldots, c_{d+1}^d) = \bigwedge_{k=1}^{d} l_{d+1}(c_{d+1}^k) =$$
$$= l_{d+1}(c_{d+1}^1) \wedge l_{d+1}(c_{d+1}^2) \wedge \cdots \wedge l_{d+1}(c_{d+1}^d) = 0$$
$$\Rightarrow \exists k_{d+1} \in \{1, \ldots, d\} \text{ such that } l_{d+1}(c_{d+1}^{k_{d+1}}) = 0$$

$$\begin{aligned}
\text{So we have that } l_1(x_{k_1}) &= 0 \quad \text{where } x_{k_1} = c_1^{k_1} \text{ variable on position } k_1 \\
l_2(x_{k_2}) &= 0 \quad \text{where } x_{k_2} = c_2^{k_2} \text{ variable on position } k_2 \\
&\vdots \\
l_{d+1}(x_{k_{d+1}}) &= 0 \quad \text{where } x_{k_{d+1}} = c_{d+1}^{k_{d+1}} \text{ variable on position } k_{d+1}
\end{aligned}$$

We have $d+1$ literals that use variables $x_1, x_2, \ldots, x_d$. So there are at least two literals using the same variable. Let these literals be l_i and l_j and assume that the variable they use is x_k.

$$\begin{aligned}
&\ldots \\
&h_i = l_i(x_1) \wedge l_i(x_2) \wedge \cdots \wedge \underline{l_i(x_k)} \wedge \ldots \\
&\ldots \\
&h_j = l_j(x_1) \wedge l_j(x_2) \wedge \cdots \wedge \underline{l_j(x_k)} \wedge \ldots \\
&\ldots
\end{aligned}$$

We will gone use l_i and l_j to arrive at a contradiction.
We know that l_i and l_j satisfy the following conditions:
$l_i(c_i^k) = 0$ (because $h_i(c_i) = 0$ and the conjunction contains literal $l_i(c_i^k)$ which is 0)
$l_j(c_j^k) = 0$ (because $h_j(c_j) = 0$ and the conjunction contains literal $l_j(c_j^k)$ which is 0)
In general we have that $l_i(x_k) \in \{x_k, \overline{x_k}, 1\}$, $\quad l_j(x_k) \in \{x_k, \overline{x_k}, 1\}$
But $l_i(x_k) \neq 1$ because we have that $l_i(c_i^k) = 0$.
Same argument goes for $l_j(x_k) \neq 1$.
So $l_i(x_k)$ can take values in $\{x_k, \overline{x_k}\}$ and $l_j(x_k)$ can take values in $\{x_k, \overline{x_k}\}$.
There are 4 possible cases.

Case 1: $l_i(x_k) = x_k$, $l_j(x_k) = x_k$

$$h_i(c_i) = l_i(c_i^1) \wedge l_i(c_i^2) \wedge \cdots \wedge \underset{\substack{\shortparallel \\ 0}}{l_i(c_i^k)} \wedge \cdots = 0$$

We have that $l_i(c_i^k) = c_i^k = 0$.

But we also have that $h_j(c_i) = 1 \Leftrightarrow l_j(c_i^1) \wedge l_j(c_i^2) \wedge \cdots \wedge l_j(c_i^k) \wedge \cdots = 1$

This means that all literals are 1, including $l_j(c_i^k)$.

But $l_j(c_i^k) = c_i^k = 0$. So we have a contradiction.

Case 2: $l_i(x_k) = \overline{x_k}$, $l_j(x_k) = \overline{x_k}$

$$h_i(c_i) = l_i(c_i^1) \wedge l_i(c_i^2) \wedge \cdots \wedge \underset{\substack{\shortparallel \\ 0}}{l_i(c_i^k)} \wedge \cdots = 0$$

We have that $l_i(c_i^k) = \overline{c_i^k} = 1 - c_i^k = 0 \Rightarrow c_i^k = 1$.

But we have that $h_j(c_i) = 1 \Leftrightarrow l_j(c_i^1) \wedge l_j(c_i^2) \wedge \cdots \wedge l_j(c_i^k) \wedge \cdots = 1 \Rightarrow l_j(c_i^k) = 1$.

But $l_j(c_i^k) = 1 - c_i^k = 0$. Contradiction.

Case 3: $l_i(x_k) = x_k, l_j(x_k) = \overline{x_k}$

Take another point c_m that is different than c_i and c_j, $m \neq i, m \neq j$ and $1 \leq m \leq d+1$.

So we have $h_i(c_m) = h_j(c_m) = 1$

$$h_i(c_m) = \cdots \wedge l_i(c_m^k) \wedge \cdots = 1 \Rightarrow l_i(c_m^k) = c_m^k = 1$$
$$h_j(c_m) = \cdots \wedge l_j(c_m^k) \wedge \cdots = 1 \Rightarrow l_j(c_m^k) = 1 - c_m^k = 1 \Rightarrow c_m^k = 0. \text{ Contradiction.}$$

Case 4: $l_i(x_k) = \overline{x_k}, l_j(x_k) = x_k$

Same as Case 3, you will see that

$l_i(c_m^k) = 1 - c_m^k = 1 \Rightarrow c_m^k = 0$

$l_j(c_m^k) = c_m^k = 1$. Contradiction.

Exercise 4

$$\mathcal{H} = \left\{ \begin{array}{l} h_{a,b,s} \colon a \leq b,\, s \in \{-1, 1\}, \\ h_{a,b,s}(x) = \begin{cases} s, & x \in [a,b] \\ -s, & x \notin [a,b] \end{cases} \end{array} \right\}$$

See label 0 as label -1.

$VC\dim(\mathcal{H}) = ?$

$\mathcal{H}$ contains functions parametrized by 3 params (a, b, s). Intuition tells us that $VC\dim(\mathcal{H}) = 3$ (not always, but usually).

Let's consider $C = \{c_1, c_2, c_3\}$ a set of 3 distinct points with $c_1 < c_2 < c_3$ (for example, take $c_1 = 0, c_2 = 1, c_3 = 2$).

To obtain labels $(0,0,0)((-1,-1,-1))$, take $a = b = c_1 - 1$, $s = 1$ or $a = c_1$, $b = c_3$, $s = -1$ ✓

To obtain labels $(1,1,1)$, take $a = c_1$, $b = c_3$, $s = 1$ ✓

To obtain labels $(1,-1,-1)$, take $a = c_1$, $b = \frac{c_1 + c_2}{2}$, $s = 1$

To obtain labels $(-1,1,1)$, take $a = c_1$, $b = \frac{c_1 + c_2}{2}$, $s = -1$

To obtain labels $(-1,1,-1)$, take $a = \frac{c_1 + c_2}{2}$, $b = \frac{c_2 + c_3}{2}$, $s = 1$

To obtain labels $(1,-1,1)$, take $a = \frac{c_1 + c_2}{2}$, $b = \frac{c_2 + c_3}{2}$, $s = -1$

To obtain labels $(-1,-1,1)$, take $a = \frac{c_2 + c_3}{2}$, $b = c_3 + 1$, $s = 1$

To obtain labels $(1,1,-1)$, take $a = \frac{c_2 + c_3}{2}$, $b = c_3 + 1$, $s = -1$

So $\mathcal{H}$ shatters C, so $VC\dim(\mathcal{H}) \geq 3$.

Now, take C, a set of 4 points, $C = \{c_1, c_2, c_3, c_4\}$, $c_1 \leq c_2 \leq c_3 \leq c_4$.

Then $\mathcal{H}$ cannot realize the labels (1, -1, 1, -1).

This happens for any C. So $VC\dim(\mathcal{H}) < 4$. $\boxed{\text{So } VC\dim(\mathcal{H}) = 3}$

Advanced Machine Learning Seminar 2

Exercise 1. Prove that the Bayes optimal predictor has the smallest error among all possible classifiers.

Proof. Let $\mathcal{D}$ be a probability distribution over $X \times \{0,1\}$. The Bayes classifier is defined as:

$$f_{\mathcal{D}}\colon X \to \{0,1\},\ f_{\mathcal{D}} = \begin{cases} 1, & \overbrace{\underset{(x,y)\sim\mathcal{D}}{P}(y=1 \mid x)}^{\eta(x)} \geq \frac{1}{2} \\ 0, & otherwise \end{cases}$$

Let $g\colon X \to \{0,1\}$ be a random classifier. We want to show that $L_{\mathcal{D}}(f_{\mathcal{D}}) \leq L_{\mathcal{D}}(g)$.

$$L_{\mathcal{D}}(f_{\mathcal{D}}) = \underset{(x,y)\sim\mathcal{D}}{E}(l(f_{\mathcal{D}}(x,y))) = \underset{(x,y)\sim\mathcal{D}}{E}(\mathbb{1}_{[f_{\mathcal{D}}(x)\neq y]}) = \underset{(x,y)\sim\mathcal{D}}{P}(f_{\mathcal{D}}(x) \neq y)$$

$$l(f_{\mathcal{D}}(x,y)) = \text{0-1 loss} = \begin{cases} 1, f_{\mathcal{D}}(x) \neq y \\ 0, f_{\mathcal{D}}(x) = y \end{cases} = \mathbb{1}_{[f_{\mathcal{D}}(x)\neq y]}$$

$$\underset{(x,y)\sim\mathcal{D}}{E}(\mathbb{1}_{[f_{\mathcal{D}}(x)\neq y]}) = \underset{x\sim\mathcal{D}_x}{E}[\underbrace{\underset{x\sim\mathcal{D}_{y|x}}{E}[\mathbb{1}_{[f_{\mathcal{D}}(x)\neq y]} \mid x]}_{=\underset{y\sim\mathcal{D}_{y|x}}{P}(f_{\mathcal{D}}(x)\neq y|x)}]$$

$$\begin{aligned}\underset{y\sim\mathcal{D}_{y|x}}{P}(f_{\mathcal{D}}(x) \neq y \mid x) &= P(y=1 \mid x) \cdot \mathbb{1}_{[\eta(x)<\frac{1}{2}]} + P(y=0 \mid x) \cdot \mathbb{1}_{[\eta(x)\geq\frac{1}{2}]} \\ &= \eta(x) \cdot \mathbb{1}_{[\eta(x)<\frac{1}{2}]} + (1-\eta(x)) \cdot \mathbb{1}_{[\eta(x)\geq\frac{1}{2}]} \\ &= \begin{cases} \eta(x), \eta(x) < \frac{1}{2} \\ 1-\eta(x), \eta(x) \geq \frac{1}{2} \end{cases} = min(\eta(x), 1-\eta(x))\end{aligned}$$

$$L_{\mathcal{D}}(g) = \underset{(x,y)\sim\mathcal{D}}{E}(l(g((x,y))) = \underset{(x,y)\sim\mathcal{D}}{E}(\mathbb{1}_{g(x)\neq y}) = \underset{x\sim\mathcal{D}_x}{E}[\underbrace{\underset{x\sim\mathcal{D}_{y|x}}{E}[\mathbb{1}_{g(x)\neq y} \mid x]}_{=\underset{y\sim\mathcal{D}_{y|x}}{P}(g(x)\neq y|x)}] \qquad (*)$$

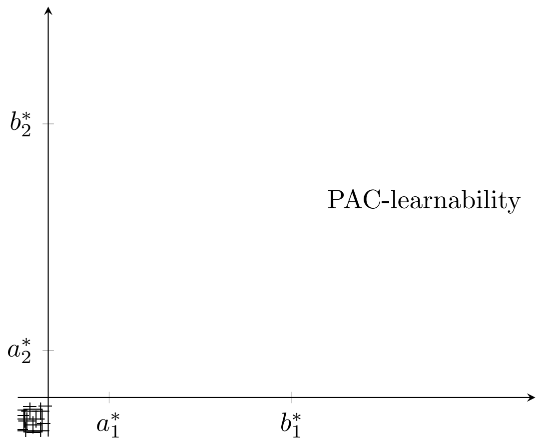
$$\begin{aligned}\underset{y\sim\mathcal{D}_{y|x}}{P}(g(x) \neq y \mid x) &= P(g(x)=0, y=1 \mid x) + P(g(x)=1, y=0 \mid x) \\ &= P(g(x)=0 \mid x) \cdot P(y=1 \mid x) + P(g(x)=1 \mid x) \cdot P(y=0 \mid x) \\ &= P(g(x)=0 \mid x) \cdot \underbrace{\eta(x)}_{\geq min(\eta(x),1-\eta(x))} + P(g(x)=1 \mid x) \cdot \underbrace{1-\eta(x)}_{\geq min(\eta(x),1-\eta(x))} \\ &\geq P(g(x)=0 \mid x) \cdot min(\eta(x), 1-\eta(x)) + P(g(x)=1 \mid x) \cdot min(\eta(x), 1-\eta(x)) \\ &\geq (P(g(x)=0 \mid x) + P(g(x)=1 \mid x)) \cdot min(\eta(x), 1-\eta(x)) \\ &= min(\eta(x), 1-\eta(x)) = \underset{y\sim\mathcal{D}_{y|x}}{P}(f_{\mathcal{D}}(x) \neq y \mid x) \qquad (**)\end{aligned}$$

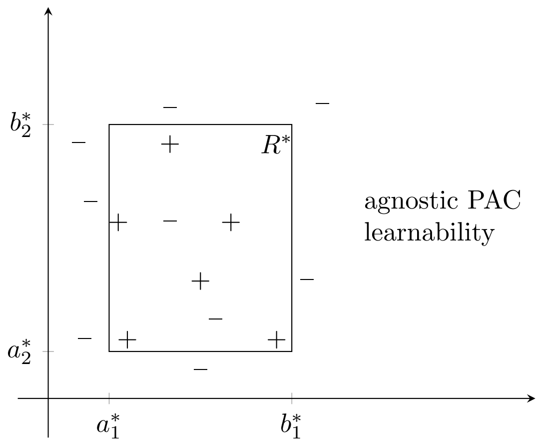
From (*) and (**) it follows that $L_{\mathcal{D}}(f_{\mathcal{D}}) \leq L_{\mathcal{D}}(g)$.

□

Exercise 2 *See the entire statement in the* `Seminar2.pdf`. Show that the algorithm returning the tightest rectangle containing positive points can still PAC-learn axis-aligned rectangles in the presence of this noise.

$$\mathcal{H}^2_{rec} = \left\{ \begin{array}{l} h_{(a_1,b_1,a_2,b_2)} : \mathbb{R}^2 \to \{0,1\}, a_1 \le b_1, a_2 \le b_2, \\ h_{(a_1,b_1,a_2,b_2)}(x) = \mathbb{1}_{[a_1,b_1]\times[a_2,b_2]}(x) = \left\{ \begin{array}{l} 1, x \in [a_1,b_1]\times[a_2,b_2] \\ 0, otherwise \end{array} \right. \end{array} \right\}$$





Positive labels are flipped with probability $0 < \eta < \frac{1}{2}$

Consider the training set $S = \{(\underline{x_1}, y_1), (\underline{x_2}, y_2), \ldots, (\underline{x_m}, y_m)\}$, where the label y_i is given in the agnostic case by a distribution. We have that:

$$y_i = \left\{ \begin{array}{l} 0,\ if\ \underline{x_i} \notin [a_1^*, b_1^*] \times [a_2^*, b_2^*] \\ 0,\ with\ probability\ \eta\ if\ \underline{x_i} \in [a_1^*, b_1^*] \times [a_2^*, b_2^*] \\ 1,\ with\ probability\ 1-\eta\ if\ \underline{x_i} \in [a_1^*, b_1^*] \times [a_2^*, b_2^*] \end{array} \right.$$

We denote with $R^* = [a_1^*, b_1^*] \times [a_2^*, b_2^*]$ the rectangle determined by h^*.

The chance to get a training point labeled as positive is to sample a point from R^* and the label is not flipped so the chance is $\mathcal{D}(R^*) \times (1-\eta)$.

Let A be the learning algorithm that returns the tightest rectangle containing positive points.

$h_S = A(S),\ h_S = h_{(a_{1S}, b_{1S}, a_{2S}, b_{2S})}$, where

$$a_{1S} = \min_{(\underline{x_i},1)\in S} x_{i1} \qquad a_{2S} = \min_{(\underline{x_i},1)\in S} x_{i2}$$

$$b_{1S} = \max_{(\underline{x_i},1)\in S} x_{i1} \qquad b_{2S} = \max_{(\underline{x_i},1)\in S} x_{i2}$$

If S doesn't contain positive samples, then A will return $h_S = h_{(z_1, z_1+1, z_2, z_2+1)}$, where $z = (z_1, z_2)$ is chosen such that the learned classifier has empirical risk equal to zero. For example, take:

$$z_1 = 1 + \max_{(\underline{x_i},0)\in S} x_{i1}, \qquad z_2 = 1 + \max_{(\underline{x_i},0)\in S} x_{i2}$$

We want to show that $\mathcal{H}^2_{rec}$ is agnostic PAC-learnable: there exists a function $m_{\mathcal{H}^2_{rec}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm A such that for every $\epsilon > 0$, for every $\delta > 0$, for every distribution $\mathcal{D}$ over $Z = \mathbb{R}^2 \times \{0,1\}$, $\mathcal{D} = \mathcal{D}_X \times \mathcal{D}_Y$ when we run the learning algorithm A on a training set S consisting of $m \geq m_{\mathcal{H}^2_{rec}}(\epsilon, \delta)$ examples sampled i.i.d. from $\mathcal{D}$, the algorithm A returns a hypothesis $h_S = A(S)$ from $\mathcal{H}^2_{rec}$ such that

$$\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D}}(h_S)) \leq \underset{h\in\mathcal{H}}{min} L_{\mathcal{D}}(h) + \epsilon) \geq 1 - \delta$$

In our case, the smallest achievable real error is

$$\underset{h\in\mathcal{H}}{min} L_{\mathcal{D}}(h) = L_{\mathcal{D}}(h^*) = \eta \cdot \mathcal{D}_X(R^*)$$

Consider $\epsilon > 0, \delta > 0$ and $\mathcal{D}_X$ a distribution over $\mathbb{R}^2$.
<u>Case 1</u>: if $\mathcal{D}_X(R^*) \leq \epsilon \Rightarrow h_S$ can only make errors on points inside R^*, so $\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D}}(h_S) \leq \epsilon) = 1$ ✓
<u>Case 2</u>: if $\mathcal{D}_X(R^*) > \epsilon$
Construct rectangles R_1, R_2, R_3, R_4 (like in seminar 1) such that $\mathcal{D}_X(R_i) = \frac{\epsilon}{4}$

i) if $h_S = A(S)$ intersects all R_i, $i = 1,4$, then h_S will make errors in:

- region R_S, because of flipping $\to \mathcal{D}(R_S) \cdot \eta$
- region $R^* \setminus R$, but we know that $\mathcal{D}_X(R^* \setminus R) < \epsilon$

So in this case we have $\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D}}(h_S) \leq \eta \cdot \mathcal{D}(R^*) + \epsilon) = 1$

ii) if $h_S = A(S)$ doesn't intersect a rectangle R_i

Denote $F_i = \{S \mid S \sim \mathcal{D}^m_X$ such that R_S, the rectangle learned by $A(S)$, doesn't intersect $R_i\}$

$$\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D}}(h_S) > \eta \cdot \mathcal{D}(R^*) + \epsilon) \leq \sum_{i=1}^{4} \mathcal{D}^m_X(F_i)$$

$\mathcal{D}^m_X(F_i) =$ the probability of sampling m points and none of them is a positive point in R_i

$$= \left(\underbrace{1 - \frac{\epsilon}{4}}_{\substack{\text{prob. of sampling} \\ \text{a point outside } R_i}} + \underbrace{\frac{\epsilon}{4} \cdot \eta}_{\substack{\text{prob of sampling} \\ \text{a point in } R_i \\ \text{but flipping its label}}} \right)^m = \left(1 - \frac{\epsilon}{4}(1-\eta)\right)^m$$

$$1 - x \leq e^{-x}$$
$$1 - \frac{\epsilon}{4}(1-\eta) \leq e^{-\frac{\epsilon}{4}(1-\eta)}$$

So:

$$\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D}}(h_S) > \eta \cdot \mathcal{D}(R^*) + \epsilon) \leq \underline{4 \cdot e^{-\frac{\epsilon}{4}(1-\eta)\cdot m}} < \delta$$

$$4e^{-\frac{\epsilon}{4}(1-\eta)\cdot m} < \delta$$

$$e^{-\frac{\epsilon}{4}(1-\eta)\cdot m} < \frac{\delta}{4} \quad \Big| \log_e$$

$$m \cdot \left(-\frac{\epsilon}{4}\right)(1-\eta) < \log\frac{\delta}{4} \quad \Big| \cdot \left(-\frac{4}{\epsilon}\right) \cdot \frac{1}{1-\eta}$$

$$m > -\frac{4}{\epsilon} \cdot \frac{1}{1-\eta} \cdot \log\frac{\delta}{4}$$

$$\boxed{m > \frac{4}{\epsilon} \cdot \frac{1}{1-\eta} \cdot \log\frac{4}{\delta}}$$

Exercise 3

$$\mathcal{C} = \mathcal{H}_{intervals} = \left\{ \begin{array}{l} h_{a,b} \colon \mathbb{R} \to \{0,1\}, a \le b \\ h_{a,b}(x) = \mathbb{1}_{[a,b]}(x) = \left\{ \begin{array}{l} 1, x \in [a,b] \\ 0, otherwise \end{array} \right. \end{array} \right\}$$

Consider a training set $S = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$. We are in the realizability case, $\exists h^* = h_{a^*,b^*} \in \mathcal{H}_{intervals}$ that labels the examples, $y_i = h^*(x_i)$.

We want to show that $\mathcal{H}_{intervals}$ is PAC-learnable.

Consider A the learning algorithm that gets the training set S and outputs $h_S = A(S) =$ the tightest interval containing all the positive examples.

$$h_S = h_{a_S,b_S} \qquad \text{where} \qquad a_S = \min_{(x_i,1)\in S} x_i \qquad b_S = \max_{(x_i,1)\in S} x_i \qquad R_S = [a_S, b_S]$$

If there is no $(x_i, 1) \notin S$ (S doesn't contain positive examples), take $a_S = b_S = z$ a random point such that $(z, 0) \notin S$.

From construction, we see that $L_S(h_S) = 0$.

Let $\epsilon > 0, \delta > 0$ and $\mathcal{D}$ a distribution over R. We need to find how many training examples $m \ge m_{\mathcal{I}}(\epsilon, \delta)$ do we need such that

$$\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) > \epsilon) < \delta$$

Case 1: if $\mathcal{D}([a^*, b^*]) \le \epsilon$ then $\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) > \epsilon) = 0$ ✓

Case 2: if $\mathcal{D}([a^*, b^*]) > \epsilon$

Build R_1 and R_2, $R_1 = [a_1^*, a_1]$, $R_2 = [b_1, b_1^*]$ such that $\mathcal{D}(R_1) = \mathcal{D}(R_2) = \frac{\epsilon}{2}$.

If $R_S \cap R_1 \neq \emptyset$ and $R_S \cap R_2 \neq \emptyset$ then $\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) > \epsilon) = 0$ ✓

Else $\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) > \epsilon) \le 2 \cdot \left(1 - \frac{\epsilon}{2}\right)^m \le 2 \cdot e^{-\frac{\epsilon}{2}m} < \delta \Rightarrow \boxed{m > \frac{2}{\epsilon} \log \frac{2}{\delta}}$

We can see this exercise also as a particular case of class $\mathcal{H}^1_{rec}$ of rectangles in one dimension. We have shown in Seminar 1, exercise 2, that the class $\mathcal{H}^d_{rec}$ can be PAC-learned.

Exercise 4 PAC-learning algorithm for the class $\mathcal{C}_2$ formed by unions of two closed intervals:

$$\mathcal{C}_2 = \left\{ \begin{array}{l} h_{(a,b,c,d)} \colon \mathbb{R} \to [0,1],\ h_{(a,b,c,d)} = \mathbb{1}_{[a,b]\cup[c,d]} \\ a \le b \le c \le d,\ h_{(a,b,c,d)}(x) = \left\{ \begin{array}{l} 1, x \in [a,b] \cup [c,d] \\ 0, otherwise \end{array} \right. \end{array} \right\}$$

Consider $S = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$, where $y_i = h^*(x_i), h^* = h_{(a^*,b^*,c^*,d^*)}$ [1].

Consider the following learning algorithm A that takes input S:

i) Sort S in ascending order of x_i

ii) If all training samples x_i have label $y_i = 0$ then take $a_S = 1 + \max\limits_{\substack{(x_i,y_i)\\ y_i=0}} x_i$, $b_S = c_S = d_S = 1_S + 1$.

   Return $h_S = h_{(a_S,b_S,c_S,d_S)} = \mathbb{1}_{[a_S,b_S]\cup[c_S,d_S]}$.

iii) Take $a_S = \min\limits_{\substack{(x_i,y_i)\\ y_i=1}} x_i$, $d_S = \max\limits_{\substack{(x_i,y_i)\\ y_i=1}} x_i$.

   Case 1: if there exists a negatively label point x_j, with $y_j = 0$ and $a_S < x_j < d_S$ then we are in the case were there is a sequence of positives examples interrupted by a negative example. In this case take: $b_S = \max\limits_{\substack{(x_i,y_i),x_i<x_j\\ y_i=1}} x_i$ and take $c_S = \min\limits_{\substack{(x_i,y_i),x_i>x_j\\ y_i=1}} x_i$.

   Case 2: if there is no negatively labeled point x_j between a_S and d_S then take $b_S = c_S = d_S$.

   Return $h_S = h_{(a_S,b_S,c_S,d_S)} = \mathbb{1}_{[a_S,b_S]\cup[c_S,d_S]}$.

[1] realizability assumption

We need to find $m \geq m_{\mathcal{C}_2}(\epsilon, \delta)$ such that for $\epsilon > 0$, $\delta > 0$ and for every $\mathcal{D}$ distribution over $\mathbb{R}$ we have

$$\underset{S \sim \mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) > \epsilon) < \delta$$

Let $\epsilon > 0$, $\delta > 0$ and let $\mathcal{D}$ be a distribution over $\mathbb{R}$.

The region where h_S can make errors is always $\subseteq [a^*, d^*]$.

<u>Case 1</u>: If $\mathcal{D}([a^*, d^*]) \leq \epsilon$, then $\underset{S \sim \mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) > \epsilon) = 0$.

<u>Case 2</u>: If $\mathcal{D}([a^*, d^*]) > \epsilon$

The types of error that h_S can make are:

- false negatives in $[a^*, b^*]$ and $[c^*, d^*]$
- false positive in (b^*, c^*) if sample S does not contain any points sampled from (b^*, c^*).

Denote L_{FP}, $L_{FN,1}$, $L_{FN,2}$ these type of errors, where:

$$\begin{aligned}
L_{FP}(h_S) &= \underset{x \sim \mathcal{D}}{P}(x \in [a_S, b_S] \cup [c_S, d_S] \setminus ([a^*, b^*] \cup [c^*, d^*])) \\
&= \underset{x \sim \mathcal{D}}{P}(x \in [b^*, c^*] \subseteq [a_S, b_S] \cup [c_S, d_S]) \\
L_{FN,1}(h_S) &= \underset{x \sim \mathcal{D}}{P}(x \in [a^*, b^*] \setminus [a_S, b_S]) \\
L_{FN,2}(h_S) &= \underset{x \sim \mathcal{D}}{P}(x \in [c^*, d^*] \setminus [c_S, d_S])
\end{aligned}$$

So, if we want to have $L_{\mathcal{D},h^*}(h_S) > \epsilon$, then one of the numbers L_{FP}, $L_{FN,1}$, $L_{FN,2}$ must be $> \frac{\epsilon}{3}$.

Define

$$\begin{aligned}
F_1 &= \left\{ S \sim \mathcal{D}^m \mid L_{FP}(h_S) > \frac{\epsilon}{3} \right\} \\
F_2 &= \left\{ S \sim \mathcal{D}^m \mid L_{FN,1}(h_S) > \frac{\epsilon}{3} \right\} \\
F_3 &= \left\{ S \sim \mathcal{D}^m \mid L_{FN,2}(h_S) > \frac{\epsilon}{3} \right\}
\end{aligned}$$

So,

$$\underset{S \sim \mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) \geq \epsilon) \leq \underset{S \sim \mathcal{D}^m}{P}(F_1 \cup F_2 \cup F_3) \leq \sum_{i=1}^{3} P(F_i)$$

$$\begin{aligned}
P(F_1) &= \underset{S \sim \mathcal{D}^m}{P}\left(L_{FP}(h_S) > \frac{\epsilon}{3}\right) \\
&= \left(\text{this means that } \mathcal{D}([b^*, c^*]) > \frac{\epsilon}{3} \text{ and no point from } [b^*, c^*] \text{ is sampled in } S\right) \\
&\leq \left(1 - \frac{\epsilon}{3}\right)^m \leq e^{-\frac{\epsilon}{3}m} \\
P(F_2) &= \underset{S \sim \mathcal{D}^m}{P}\left(L_{FN,1}(h_S) > \frac{\epsilon}{3}\right)
\end{aligned}$$

Construct $R_1 = [a^*, a_0]$ and $R_2 = [b_0, b^*]$ such that $\mathcal{D}(R_1) = \mathcal{D}(R_2) = \frac{\epsilon}{6}$.

If $[a_S, b_S] \cap R_1 \neq \emptyset$ and $[a_S, b_S] \cap R_2 \neq \emptyset$, then the error made by h_S on $[a^*, b^*]$ is smaller than $\frac{\epsilon}{6} + \frac{\epsilon}{6} \geq \frac{\epsilon}{3}$.

So $L_{FN,1}(h_S) > \frac{\epsilon}{3} \Rightarrow [a_S, b_S] \cap R_1 = \emptyset$ or $[a_S, b_S] \cap R_2 = \emptyset$.

Define

$$\begin{aligned}
F_{21} &= \{S \sim \mathcal{D}^m \mid [a_S, b_S] \cap R_1 = \emptyset\} \\
F_{22} &= \{S \sim \mathcal{D}^m \mid [a_S, b_S] \cap R_2 = \emptyset\}
\end{aligned}$$

$$P(F_2) \leq P(F_{21} \cup F_{22}) \leq P(F_{21}) + P(F_{22}) = 2 \cdot \left(1 - \frac{\epsilon}{6}\right)^m \leq 2 \cdot e^{-\frac{\epsilon}{6}m}$$

In the same way we can prove that $P(F_3) \leq 2 \cdot e^{-\frac{\epsilon}{6}m}$. So we obtain that

$$\underset{S \sim \mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) > \epsilon) \leq e^{-\frac{\epsilon}{3}m} + 4 \cdot e^{-\frac{\epsilon}{6}m} \leq e^{-\frac{\epsilon}{6}m} + 4 \cdot e^{-\frac{\epsilon}{6}m} = 5 \cdot e^{-\frac{\epsilon}{6}m} < \delta$$

$$\Rightarrow e^{-\frac{\epsilon}{6}m} < \frac{\delta}{5} \,\Big|\, \cdot \log$$

$$\Rightarrow -\frac{\epsilon}{6}m < \log\frac{\delta}{5} \,\Big|\, \cdot\left(-\frac{6}{\epsilon}\right)$$

$$\boxed{m > \frac{6}{\epsilon}\cdot\log\frac{5}{\delta}}$$

In the general case, for $\mathcal{C}_p$ = reunion of p intervals, the proof is similar, the only differences are that:

- there are $(p-1)$ regions of false positives
- $2p$ regions of false negatives

So we have

$$\boxed{m \geq \frac{2(2p-1)}{\epsilon}\cdot\log\frac{p+2p-1}{\delta}}$$

$$\boxed{m \geq \frac{2(2p-1)}{\epsilon}\cdot\log\frac{3p-1}{\delta}}$$

time complexity $\rightarrow$ given by sorting $S = \mathcal{O}(m\log m)$

Exercise 5 Let $h \in \mathcal{H}$, with $L_{\overline{D}_m,\delta}(h) > \epsilon \Rightarrow$

$$\underset{x\sim\overline{D}_m}{P}(h(x) \neq f(x)) > \epsilon \Leftrightarrow \underset{x\sim\overline{D}_m}{P}(h(x) = f(x)) = 1 - \underset{x\sim\overline{D}_m}{P}(h(x) \neq f(x)) < 1-\epsilon$$

$$\begin{aligned}\underset{x\sim\overline{D}_m}{P}(h(x) = f(x)) &= \left(x \text{ can be sampled from each } D_i\text{, with probability } \frac{1}{m}\right)\\ &= \frac{1}{m}\cdot\underset{x\sim\overline{D}_1}{P}[h(x)=f(x)] + \cdots + \frac{1}{m}\cdot\underset{x\sim\overline{D}_m}{P}[h(x)=f(x)]\\ &= \frac{1}{m}\sum_{i=1}^{m}\underset{x\sim\overline{D}_i}{P}[h(x)=f(x)] < 1-\epsilon\end{aligned}$$

Consider the training set $S = \{(x_1, f(x_1)), (x_2, f(x_2)), \ldots, (x_m, f(x_m)) \mid \text{where } \; x_i \sim D_i\}$
h consistent with S if $L_S(h) = 0$.

$$\begin{aligned}\underset{S\sim D_1\times D_2\times\cdots\times D_m}{P}[L_S(h)=0] &= \prod_{i=1}^{m}\underset{x_i\sim D_i}{P}[h(x_i)=f(x_i)]\\ &= \prod_{i=1}^{m}\underset{x\sim D_i}{P}[h(x)=f(x)]\\ &= \left[\left(\prod_{i=1}^{m}\underset{x\sim D_i}{P}[h(x)=f(x)]\right)^{\frac{1}{m}}\right]^m\\ &= \text{geometric mean } = (a_1\cdot a_2\cdot\cdots\cdot a_m)^{\frac{1}{m}}\\ &\quad\ \text{where } a_i = \underset{x\sim D_i}{P}[h(x)=f(x)] = \begin{array}{c}\text{probability that } h \text{ correctly}\\ \text{labels a point } x\sim D_i\end{array}\\ &\leq \text{arithmetic mean} = \left(\frac{a_1+a_2+\cdots+a_m}{m}\right)\\ &\leq \left[\frac{1}{m}\sum_{i=1}^{m}\underset{x\sim D_i}{P}[h(x)=f(x)]\right]^m < (1-\epsilon)^m \leq e^{-\epsilon m}\end{aligned}$$

There are at most $|\mathcal{H}|$ number of h hypotheses. So, we observe that

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

Advanced Machine Learning Seminar 1 - solutions

Exercise 1 Consider the training set $S = \{(\mathbf{x}_i, f(\mathbf{x}_i))\}_{i=1}^m \subseteq (\mathbb{R}^d \times \{0,1\})^m$. We consider the classifier from Lecture 2: $h_S \colon \mathbb{R}^d \to \{0,1\}$

$$h_S(\mathbf{x}) = \begin{cases} y_i = f(\mathbf{x}_i), & \text{if } \exists i \in \{1,\dots,m\} \text{ such that } \mathbf{x}_i = \mathbf{x} \\ 0, & \text{otherwise} \end{cases}$$

We want to show that the classifier h_S can be written as a thresholded polynomial P_S, meaning that we want to find a polynomial P_S such that $h_S(\mathbf{x}) = 1 \Leftrightarrow P_S(\mathbf{x}) \geq 0$.

Proof. Let's consider the simpler case, $d = 1$ (so x_i is a scalar).

<u>1^{st} try:</u> Consider the polynomial

$$P_S(x) = -\prod_{i=1}^m (x - x_i)$$

If $x = x_i$ for some $i \in \{1,\dots,m\} \Rightarrow P_S(x) = P_S(x_i) = 0 \Rightarrow h_S(x) = 1$.

This polynomial it will not work if the label of the point x_i is $y_i = 0$ (in this case, $P_S(x_i) = 0 \Rightarrow h_S(x_i) = 1$).

Also, if x doesn't appear in the training data, we don't know if $P_S(x) \geq 0$ or $P_S(x) < 0$.

So, the current polynomial P_S has some drawbacks.

<u>2^{nd} try:</u> Consider the polynomial

$$P_S(x) = -\prod_{i=1}^m (x - x_i)^2$$

If $x = x_i$ for some $i \in \{1,\dots,m\} \Rightarrow P_S(x) = P_S(x_i) = 0 \Rightarrow h_S(x) = 1$.

For points $(x_i, 0) \in S$ it will not work.

For all other points, it will work fine.

<u>3^{rd} try:</u> Consider the polynomial

$$P_S(x) = -\prod_{\substack{i=1 \\ y_i=1}}^m (x - x_i)^2$$

In this case, if all $y_i = 0$, then $P_S(x) = -1$.

$$\text{If } x = x_i, \text{ for some } i \in \{1,\dots,m\} : \begin{array}{lll} \text{if } y_i = 1 & \Rightarrow P_S(x) = 0 & \Rightarrow h_S(x) = 1 \checkmark \\ \text{if } y_i = 0 & \Rightarrow P_S(x) < 0 & \Rightarrow h_S(x) = 0 \checkmark \end{array}$$

If $x \neq x_i$ for all $i \in \{1,\dots,m\} \Rightarrow P_S(x) < 0 \Rightarrow h_S(x) = 0 \checkmark$

Other choices for polynomial P_S could be:

$$P_S(x) = -\prod_{i=1}^m (x - x_i)^{2y_i}$$

$$P_S(x) = -\prod_{i=1}^m \left[(x - x_i)^2 + 1 - y_i\right]$$

Consider now the general case, d can be > 1.

For $d = 1$ we have seen that

$$P_S(x) = -\prod_{\substack{i=1 \\ y_i=1}}^{m} (x - x_i)^2 \text{ works fine.}$$

In the general case, we consider the L_2 distance (Euclidean distance):

$$P_S(\mathbf{x}) = -\prod_{\substack{i=1 \\ y_i=1}}^{m} \|\mathbf{x} - \mathbf{x}_i\|_2^2$$

This polynomial will work fine.

$\square$

Exercise 2

$$\mathcal{H}^2_{rec} = \left\{ \begin{array}{l} h_{(a_1,b_1,a_2,b_2)}\colon \mathbb{R}^2 \to \{0,1\}, a_1 \le b_1 \text{ and } a_2 \le b_2, \\ h_{(a_1,b_1,a_2,b_2)}(x_1,x_2) = \begin{cases} 1, & a1 \le x_1 \le b_1 \text{ and } a_2 \le x_2 \le b_2 \\ 0, & \text{otherwise} \end{cases} \end{array} \right\}$$

$\mathcal{H}^2_{rec}$ is an infinite size hypothesis class, it is called the class of all axis aligned rectangles in the plane. We want to prove that $\mathcal{H}^2_{rec}$ is PAC-learnable.

Proof. From the definition of PAC-learnability, we know that $\mathcal{H} = \mathcal{H}^2_{rec}$ is PAC-learnable if there exists a function $m_{\mathcal{H}}\colon (0,1)^2 \to \mathbb{N}$ and there exists a learning algorithm A with the following property: for every $\epsilon, \delta > 0$, for every labeling function $f \in \mathcal{H}^2_{rec}$ (realizability case), for every distribution $\mathcal{D}$ on $\mathbb{R}^2$ when we run the learning algorithm A on a training set S consisting of $m \ge m_{\mathcal{H}}(\epsilon,\delta)$ examples sampled i.i.d. from $\mathcal{D}$ and labeled by f, the algorithm A returns a hypothesis $h_S \in \mathcal{H}$ such that, with probability at least $1-\delta$ (over the choice of examples), the real risk of h_S is smaller than ϵ:

$$\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D},f}(h_S) \le \epsilon) \ge 1-\delta \text{ or otherwise said}$$
$$\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D},f}(h_S) > \epsilon) < \delta$$

First, we need to find the algorithm A.

We are under the realizability assumption, so there exists a labeling function $f \in \mathcal{H}, f = h^*_{(a_1^*,b_1^*,a_2^*,b_2^*)}$ that labels the training data.

Consider the training set $S = \left\{ (x_1,y_1),(x_2,y_2),\ldots,(x_m,y_m) \;\middle|\; \begin{array}{l} y_i = h^*_{(a_1^*,b_1^*,a_2^*,b_2^*)}(x_i), \\ x_i \in \mathbb{R}^2, x_i = (x_{i1},x_{i2}) \end{array} \right\}$

As in Figure 1, h^* labels each point drawn from the rectangle $R^* = [a_1^*,b_1^*] \times [a_2^*,b_2^*]$ with label 1, and all other points with label 0. So we have $h^*_{(a_1^*,b_1^*,a_2^*,b_2^*)} = \mathbb{1}_{R^*}$

Consider the following algorithm A, that takes as input the training set S and outputs h_S.

$h_S = h_{(a_{1S},b_{1S},a_{2S},b_{2S})}$, where

$$a_{1S} = \min_{\substack{i=\overline{1,m}\\ y_i=1}} x_{i1} \qquad a_{2S} = \min_{\substack{i=\overline{1,m}\\ y_i=1}} x_{i2}$$
$$b_{1S} = \max_{\substack{i=\overline{1,m}\\ y_i=1}} x_{i1} \qquad b_{2S} = \max_{\substack{i=\overline{1,m}\\ y_i=1}} x_{i2}$$

If all $y_i = 0$, then all points x_i have label 0, so there is no positive example. In this case, choose $z = (z_1,z_2)$ a point that is not in the training set S and take $a_{1S} = z_1, b_{1S} = z_1+1$, $a_{2S} = z_2, b_{2S} = z_2+1$. For example, choose:

$$z_1 = 1 + \max_{\substack{i=\overline{1,m}\\ y_i=1}} x_{i1} \qquad z_2 = 1 + \max_{\substack{i=\overline{1,m}\\ y_i=1}} x_{i2}$$

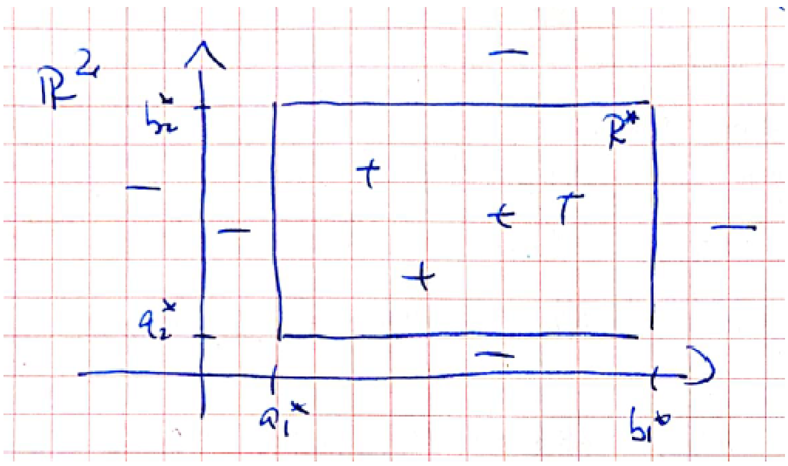


Figure 1: All the points that fall in rectangle R^* will be labeled by h^* with label 1 (+), the other points will be labeled with label 0 (-).

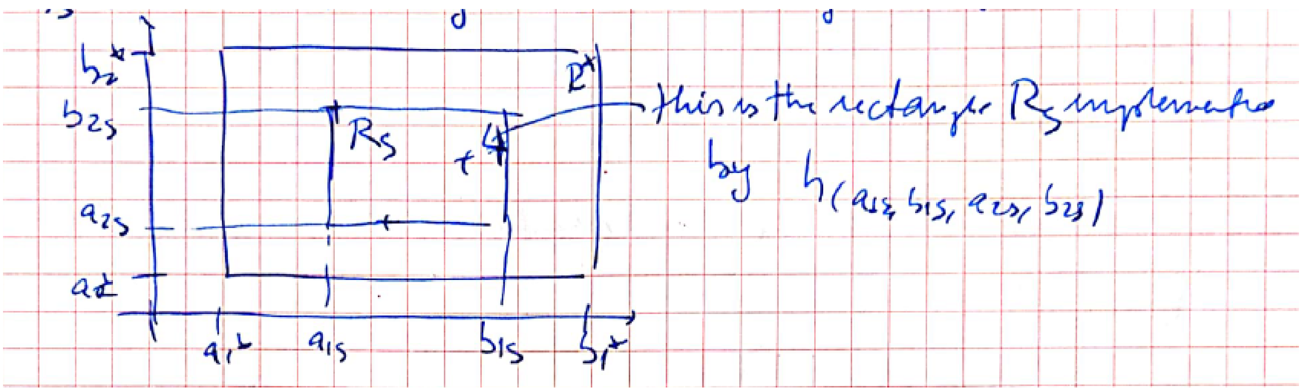


Figure 2: Rectangle R_S is the tightest rectangle enclosing all positive examples.

As in the indication, $h_S = h_{(a_{1S}, b_{1S}, a_{2S}, b_{2S})} = \mathbb{1}_{[a_{1S}, b_{1S}] \times [a_{2S}, b_{2S}]}$ is the indicator function of the tightest rectangle $R_S = [a_{1S}, b_{1S}] \times [a_{2S}, b_{2S}]$ enclosing all positive examples (see Figure 2).

By construction, A is an ERM, meaning that $L_{\mathcal{D},h^*}(h_S) = 0$, h_S doesn't make any errors on the training set S.

Now we want to find the sample complexity $m_{\mathcal{H}}(\epsilon, \delta)$ such that the chance to learn a good classifier is very high, thus:

$$\underset{S \sim \mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) \leq \epsilon) \geq 1 - \delta \text{ where } S \text{ contains } m \geq m_{\mathcal{H}}(\epsilon, \delta) \text{ examples.}$$

The chance to learn a good classifier is very high is equivalent to saying that the chance to learn a bad classifier is small, and thus we can write:

$$\underset{S \sim \mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) > \epsilon) < \delta \text{ where } S \text{ contains } m \geq m_{\mathcal{H}}(\epsilon, \delta) \text{ examples.}$$

In order to find the function $m_{\mathcal{H}}(\epsilon, \delta)$ we employ the following technique. We first determine the region where the provided classifier $h_S = A(S)$ makes errors and then discuss the probability of making these errors in the found region according to a distribution fixed $\mathcal{D}$ over $\mathbb{R}^2$.

We make the observation that h_S makes errors in region $R^* \setminus R_S$, assigning the label 0 to points that should get label 1. All points $\in R_S$ will be labeled correctly (label 1), all points outside R^* will be labeled correctly (label 0).

Let's fix $\epsilon > 0, \delta > 0$ and consider a distribution $\mathcal{D}$ over $\mathbb{R}^2$. We now discuss the probability of our classifier h_S to make errors in the found region $R^* \setminus R_S$ according to a distribution fixed $\mathcal{D}$ over $\mathbb{R}^2$. We distinguish two cases.

Case 1)

$$\text{If } \mathcal{D}(R^*) = \underset{x \sim \mathcal{D}}{P}(x \in R^*) \leq \epsilon \text{ then in this case}$$
$$L_{\mathcal{D},h^*}(h_S) = \underset{x \sim \mathcal{D}}{P}(h_S(x) \neq h^*(x)) = \underset{x \sim \mathcal{D}}{P}(x \in R^* \setminus R_S) \leq \underset{x \sim \mathcal{D}}{P}(x \in R^*) \leq \epsilon \text{ so we have that}$$
$$\underset{S \sim \mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) \leq \epsilon) = 1 \text{ (this happens all the time)}$$

Case 2) $\mathcal{D}(R^*) = \underset{x \sim \mathcal{D}}{P}(x \in R^*) > \epsilon$

We construct as in the indication the 4 rectangles R_1, R_2, R_3, R_4 (see Figure 3):

$$\begin{matrix} R_1 = [a_1^*, a_1] \times [a_2^*, b_2^*] & R_2 = [b_1, b_1^*] \times [a_2^*, b_2^*] \\ R_3 = [a_1^*, b_1^*] \times [a_2^*, a_2] & R_4 = [a_1^*, b_1^*] \times [b_2, b_2^*] \end{matrix} \quad \text{with } \mathcal{D}(R_i) = \underset{x \sim \mathcal{D}}{P}(x \in R_i) = \frac{\epsilon}{4}$$

If $R_S = [a_{1S}, b_{1S}] \times [a_{2S}, b_{2S}]$ (the rectangle returned by A, implemented by h_S) intersects each $R_i, i = \overline{1,4}$:

$$L_{\mathcal{D},h^*}(h_S) = \underset{x \sim \mathcal{D}}{P}(h^*(x) \neq h_S(x)) = \underset{x \sim \mathcal{D}}{P}(x \in R^* \setminus R_S) \leq \underset{x \sim \mathcal{D}}{P}(x \in R_1 \cup R_2 \cup R_3 \cup R_4) \leq$$
$$\leq \sum_{i=1}^{4} \underset{x \sim \mathcal{D}}{P}(x \in R_i) = \sum_{i=1}^{4} \mathcal{D}(R_i) = 4 \cdot \frac{\epsilon}{4} = \epsilon$$

So, in this case, $\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) \le \epsilon) = 1$ (this happens always).
In order to have $L_{\mathcal{D},h^*}(h_S) > \epsilon$, we need that R_S will not intersect at least one rectangle R_i.
We denote with F_i this event, so we have $F_i = \{S \sim \mathcal{D}^m \mid R_S \cap R_i = \emptyset\}$. This leads to the following:

$$\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) > \epsilon) \le \underset{S\sim\mathcal{D}^m}{P}(F1 \overset{\substack{\text{at least one } F_i \text{ will happen}\\ \nearrow}}{\cup} F2 \cup F3 \cup F4) \le \sum_{i=1}^{4} \underset{S\sim\mathcal{D}^m}{P}(F_i)$$

$$\begin{aligned}
\text{Now, } \underset{S\sim\mathcal{D}^m}{P}(F_i) &= \text{ what is the probability that } R_S \text{ will not intersect } R_i \\
&= \text{ the probability that no point from } R_i \text{ is sampled in } S \\
&= \left(1 - \frac{\epsilon}{4}\right)^m
\end{aligned}$$

So

$$\underset{S\sim\mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) > \epsilon) \le \sum_{i=1}^{4} \underset{S\sim\mathcal{D}^m}{P}(F_i) = 4 \cdot \left(1 - \frac{\epsilon}{4}\right)^m$$

Now, we know from lecture 2 that $1 - x \le e^{-x}$, so $1 - \frac{\epsilon}{4} \le e^{-\frac{\epsilon}{4}}$, which means that

$$\underset{S\sim\mathcal{D}^m}{P} \underbrace{(L_{\mathcal{D},h^*}(h_S) > \epsilon)}_{\substack{\uparrow \\ \text{this is the probability that} \\ h_S \text{ will make an error} > \epsilon}} \le 4 \cdot \left(1 - \frac{\epsilon}{4}\right)^m \le 4 \cdot e^{-\frac{\epsilon}{4}m}$$

We want to make this probability very small, smaller than δ:

$$\begin{aligned}
4 \cdot e^{-\frac{\epsilon}{4}m} &< \delta \\
e^{-\frac{\epsilon}{4}m} &< \frac{\delta}{4} \quad \Big| \cdot \log_e \\
-\frac{\epsilon}{4} \cdot m &< \log\frac{\delta}{4} \quad \Big| \cdot \left(-\frac{4}{\epsilon}\right) \\
m &> -\frac{4}{\epsilon}\log\frac{\delta}{4} = \frac{4}{\epsilon}\log\frac{4}{\delta}
\end{aligned}$$

So, if we take $m \ge m_{\mathcal{H}}(\epsilon, \delta) = \frac{4}{\epsilon} \cdot \log\frac{4}{\delta}$, we obtain the desired results.

Repeat the previous question for the class of aligned rectangles in $\mathbb{R}^d$.
In $\mathbb{R}^d$, we have

$$\mathcal{H}^d_{rec} = \left\{ \begin{array}{l} h_{(a_1,b_1,a_2,b_2,\dots,a_d,b_d)} : \mathbb{R}^d \to \{0,1\} \mid a_i \le b_i, i = \overline{1,d} \\ h_{(a_1,b_1,a_2,b_2,\dots,a_d,b_d)} = \mathbb{1}_{[a_1,b_1]\times[a_2,b_2]\times\cdots\times[a_d,b_d]} \end{array} \right\}$$

All the arguments used previously will work, the general result will be that $m_{\mathcal{H}}(\epsilon, \delta) = \frac{2d}{\epsilon} \cdot \log\frac{2d}{\delta}$.
For $d = 2$, we obtain the previous result.

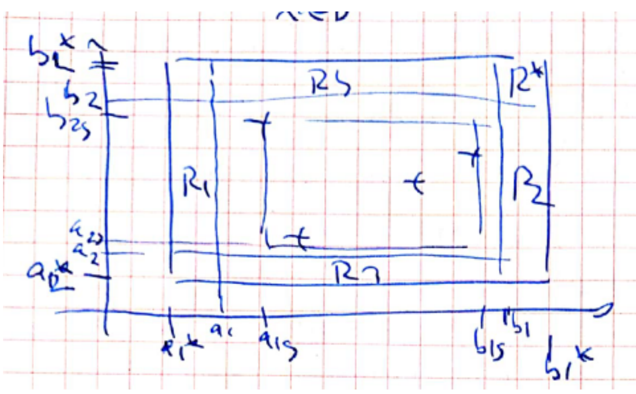


Figure 3: Constructing the rectangles R_1, R_2, R_3 and R_4.

The runtime of algorithm A is given by taking minimum over each dimension, so this means $\mathcal{O}(m * d)$:
$m =$ number of (positive) examples $= \mathcal{O}(\frac{2d}{\epsilon} \cdot \log \frac{2d}{\delta})$
$d =$ number of dimensions.

So we have that the complexity of algorithm A is $\mathcal{O}(\frac{2d^2}{\epsilon} \cdot \frac{2d}{\delta})$, which is polynomial in d, $\frac{1}{\epsilon}$, $\frac{1}{\delta}$.

$\square$

Exercise 3 $\mathcal{H}$ is PAC-learnable and $m_{\mathcal{H}}\colon (0,1)^2 \to \mathbb{N}$ is its sample complexity.

a) Given $\delta \in (0,1)$ and given $0 < \epsilon_1 \leq \epsilon_2 < 1$, we have that $m_{\mathcal{H}}(\epsilon_1, \delta) \geq m_{\mathcal{H}}(\epsilon_2, \delta)$.

Proof. If the hypothesis space $\mathcal{H}$ is PAC-learnable with sample complexity $m_{\mathcal{H}}(\cdot,\cdot)$ this means that there exists a learning algorithm A with the following property: for every $\epsilon, \delta > 0$, when we run the algorithm A on a sample set S of m examples, $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ (samples are labeled by $f \in \mathcal{H}$ and i.i.d. from a distribution $\mathcal{D}$), we have that $h_S = A(S)$ with the real risk fulfilling the relation: $\underset{S\sim\mathcal{D}^m}{P}(L_{f,\mathcal{D}}(h_S) \leq \epsilon) > 1 - \delta$.

We apply this for ϵ_1 and δ:

$$\underset{S\sim\mathcal{D}^m}{P}(L_{f,\mathcal{D}}(h_S) \leq \epsilon_1) > 1 - \delta \text{ if } m \geq m_{\mathcal{H}}(\epsilon_1, \delta)$$

We know that $\epsilon_2 \geq \epsilon_1$, so we have that

$$\underset{S\sim\mathcal{D}^m}{P}(L_{f,\mathcal{D}}(h_S) \leq \epsilon_2) > 1 - \delta \text{ if } m \geq m_{\mathcal{H}}(\epsilon_1, \delta)$$

But $m_{\mathcal{H}}(\epsilon_2, \delta)$ is the smallest number of examples for which the above inequality holds. So, if it holds for $m \geq m_{\mathcal{H}}(\epsilon_1, \delta)$, we have that $m_{\mathcal{H}}(\epsilon_1, \delta) \geq m_{\mathcal{H}}(\epsilon_2, \delta)$. ☐

b) Given $\epsilon \in (0,1)$, $0 < \delta_1 \leq \delta_2 < 1$, we have that $m_{\mathcal{H}}(\epsilon, \delta_1) \geq m_{\mathcal{H}}(\epsilon, \delta_2)$.

Proof. Using the same arguments from a), we have that

$$\underset{S\sim\mathcal{D}^m}{P}(L_{f,\mathcal{D}}(h_S) \leq \epsilon) > 1 - \delta_1 \text{ if } m \geq m_{\mathcal{H}}(\epsilon, \delta_1)$$
$$\delta_1 \leq \delta_2 \Rightarrow 1 - \delta_1 \geq 1 - \delta_2 \Rightarrow \underset{S\sim\mathcal{D}^m}{P}(L_{f,\mathcal{D}}(h_S) \leq \epsilon) > 1 - \delta_2 \text{ if } m \geq m_{\mathcal{H}}(\epsilon, \delta_1)$$

But $m_{\mathcal{H}}(\epsilon, \delta_2)$ is the smallest number of examples for which the above inequality holds (if $m \geq m_{\mathcal{H}}(\epsilon, \delta_2)$). So, if it holds for $m \geq m_{\mathcal{H}}(\epsilon, \delta_1)$, we have that $m_{\mathcal{H}}(\epsilon, \delta_1) \geq m_{\mathcal{H}}(\epsilon, \delta_2)$. ☐

Exercise 4 $\mathcal{X}$ discrete domain, $\mathcal{H}_{singleton} = \{h_z : z \in \mathcal{X}\} \cup \{h^-\}$

$$\forall z \in \mathcal{X} \quad h_z : \mathcal{X} \to \{0,1\}, \quad h_z(x) = \begin{cases} 1, & x = z \\ 0, & x \neq z \end{cases}$$
$$h^- : \mathcal{X} \to \{0,1\}, \quad h^-(x) = 0,\ \forall x \in \mathcal{X}$$

4.1) Describe an algorithm that implements the ERM rule for learning $\mathcal{H}_{singleton}$ in the realizable setup (there exists a target labeling function $f \in \mathcal{H}_{singleton}$ which labels the data, this means that $y_i = f(x_i)$).

Proof. Consider $S = \{(x_i, y_i),\ x_i \text{ i.i.d. from a distribution } \mathcal{D} \text{ over } \mathcal{X}, y_i = f(x_i)\}_{i=1}^m$.
The algorithm A is the following:

Loop over training examples (x_i, y_i), for $i = 1, \ldots, m$.
If there is an $i \in \{1, \ldots, m\}$ such that $y_i = 1$, then return hypothesis $h_S = A(S) = h_{x_i}$
Otherwise return h^-.
From construction, A is ERM, meaning that $L_S(h_S) = 0$. □

4.2) Show that $\mathcal{H}_{singleton}$ is PAC-learnable. Provide an upper bound on the sample complexity.

Proof. Let $\epsilon, \delta > 0$ and fix a distribution $\mathcal{D}$ over $\mathcal{X}$.
The only case in which the algorithm A fails is the case where $f = h_z$ and the sample
$S = \{(x_i, y_i) \mid x_i \text{ sampled i.i.d. from } \mathcal{D}, y_i = f(x_i)\}$ doesn't contain any positive examples, so all $y_i = 0\ \forall i = \overline{1, m}$.
In this case, $h_S = A(S) = h^-$, which is different from f. However, even if the algorithm A fails if $\mathcal{D}(\{z\}) \leq \epsilon$, then everything is ok, as we have that:

$$\underset{S \sim \mathcal{D}^m}{P}(L_{\mathcal{D},f}(h_S) \leq \epsilon) = 1 \quad \checkmark\checkmark$$

So, we have to upper bound the sample complexity in the case where $\mathcal{D}(\{z\}) > \epsilon$ and there is no positive example in the set S (actually, for this problem, there is just one positive possible training point $= z$). We have that

$\underset{S \sim \mathcal{D}^m}{P}(L_{\mathcal{D},f}(h_S) > \epsilon) =$ probability that each point in S
is different than z (which has probability mass $> \epsilon$) $\leq (1-\epsilon)^m \leq e^{-\epsilon m}$

So, if we set $e^{-\epsilon m} < \delta \Rightarrow -\epsilon m < \log \delta$

$$m > -\frac{1}{\epsilon} \log \delta \Rightarrow m > \frac{1}{\epsilon} \log \frac{1}{\delta}$$

If $m \geq \left\lceil \frac{1}{\epsilon} \log \frac{1}{\delta} \right\rceil$ we have that $\underset{S \sim \mathcal{D}^m}{P}(L_{\mathcal{D},h^*}(h_S) > \epsilon) < \delta$

So the upper bound of $m_{\mathcal{H}}(\epsilon, \delta)$ is $m_{\mathcal{H}}(\epsilon, \delta) \leq \left\lceil \frac{1}{\epsilon} \log \frac{1}{\delta} \right\rceil$

□