Machine Learning B (2025)
Home Assignment 4

Yasin Baysal, cmv882

Contents

1 The Airline Question

Response to Question 1.1. It is given that an airline knows that any person making a reservation on a flight will not show up with probability of 0.05 (5 %). The airline company introduce a policy to sell 100 tickets for a flight that can hold only 99 passengers. Now, we have to bound the probability that the number of people that show up for a flight will be larger than the number of seats, where we assume that they show up independently.

Let X be the number of people who show up. Since $\Pr[\text{do not show up}] = 0.05$, then $\Pr[\text{show up}] = 1 - \Pr[\text{do not show up}] = 1 - 0.05 = 0.95$. Therefore, let

$$X_i = \begin{cases} 1 & \text{if the } i\text{-th ticket-holder shows up} \\ 0 & \text{otherwise,} \end{cases}$$

so that the X_i's are independent Bernoulli random variables with

$$\Pr[X_i = 1] = p = 0.95, \quad \Pr[X_i = 0] = 0.05.$$

Then, $X = \sum_{i=1}^{100} X_i$ is the total number of people who show up, and

$$X \sim \mathrm{Bin}(n = 100, p = 0.95).$$

The flight is overbooked precisely if more than 99 people show up, i.e. $X \geq 100$. But since $X \leq 100$ almost surely (X can never exceed the 100 available tickets), then we get that

$$\Pr[X > 99] = \Pr[X = 100] = p^{100} = (0.95)^{100} \approx 0.0059.$$

Thus, the probability that the number of people that show up for a flight will be larger than the number of seats is approximately 0.59%.

Response to Question 1.2. Now, it is given that an airline has collected i.i.d. sample of 10000 flight reservations and figured out that in this sample 5 % of passengers who made a reservation did not show up for the flight. Therefore, they introduce a policy to sell 100 tickets for a flight that can hold only 99 passengers. We now have to bound the probability of observing such sample and getting a flight overbooked in two different ways:

(a) We let p be the true (unknown) probability of showing up for a flight. We then consider two independent events: the first is that in the sample of 10000 passengers, where each passenger shows up with probability p, we observe 95% of show-ups. The second is that in the sample of 100 passengers, where each passenger shows up with probability p, everybody shows up. Now, we have to bound the probability that they happen simultaneously assuming that p is known, and then we have to find the worst-case p.

First, let the two described events be given by

$$E_1 = \{\text{in 10,000 trials, exactly 9,500 show up}\}, \quad E_2 = \{\text{in 100 trials, all 100 show up}\}.$$

Conditioned on the true show-up probability p, the two are independent, with

$$\Pr[E_1 \mid p] = \binom{10000}{9500} p^{9500}(1-p)^{500}, \quad \Pr[E_2 \mid p] = p^{100}.$$

Since the events are independent, the probability of both happening simultaneously is

$$\Pr[E_1 \cap E_2] = \Pr[E_1 \mid p] \Pr[E_2 \mid p].$$

We now use the Hoeffding bound from Corollary 2.5 and equation (2.5) in Seldin (2025) to control E_1. By letting $\epsilon = p - 0.95$ and $\bar{X} = \frac{1}{10000}\sum_{i=1}^{10000} X_i$, we note also that $\Pr[\bar{X} - \mu = 0.05 - p] = \Pr[\bar{X} - \mu = -\epsilon] \Rightarrow \Pr[\mu - \bar{X} = \epsilon] \leq \Pr[\mu - \bar{X} \geq \epsilon]$, so we would get the same if we instead used the equation (2.4) from Corollary 2.5. Then, on E_1, we have $\mu - \bar{X} = p - 0.95 = \epsilon$. By equation (2.5) with $n = 10000$ and this ϵ, we therefore get that

$$\begin{aligned}
\Pr[E_1 \mid p] &= \Pr[p - 0.95 \geq \epsilon] \\
&\leq \exp\left(-2 \cdot 10000\,(p - 0.95)^2\right) \\
&= \exp\left(-20000\,(p - 0.95)^2\right),
\end{aligned}$$

such that we also get that

$$\Pr[E_1 \cap E_2] = \Pr[E_1 \mid p] \Pr[E_2 \mid p] \leq \exp\left(-20000\,(p-0.95)^2\right) \cdot p^{100} =: g(p).$$

Since we are looking for the worst-case p, we maximize $g(p)$ over p by setting the derivative of $\ln g(p)$ to zero and solve for p, such that

$$\begin{aligned}
\frac{d}{dp} \ln g(p) &= \frac{d}{dp} \ln\left(e^{-20000(p-0.95)^2}\, p^{100}\right) \\
&= \frac{d}{dp}\left[-20000(p-0.95)^2\right] \; + \; \frac{d}{dp}[100 \ln p] \\
&= -40000(p - 0.95) \; + \; \frac{100}{p} = 0 \\
&\implies \quad -40000(p - 0.95) + \frac{100}{p} = 0 \\
&\implies \quad -40000p + 38000 + \frac{100}{p} = 0 \\
&\implies \quad -40000p^2 + 38000p + 100 = 0 \\
&\implies \quad p^* = \frac{38000 + \sqrt{38000^2 + 4 \cdot 40000 \cdot 100}}{2 \cdot 40000} \approx 0.95262.
\end{aligned}$$

The second derivative of $\ln g(p)$ is

$$\frac{d^2}{dp^2} \ln g(p) = \frac{d}{dp}\left[-40000(p - 0.95) + \frac{100}{p}\right] = -40000 - \frac{100}{p^2}.$$

At the critical point $p^* \approx 0.95262$, this becomes

$$\frac{d^2}{dp^2} \ln g(p^*) = -40000 - \frac{100}{(0.95262)^2} \approx -40110.19 < 0,$$

so $\ln g$ (and hence g) attains a maximum at p^*. By substituting this back into $g(p)$,

$$g(p^*) = \exp\left(-20000\,(0.95262 - 0.95)^2\right) \cdot 0.95262^{100} \approx 0.0068.$$

So, using Hoeffding's inequality on the 10000-trial event and the exact expression for the 100-trial event, the worst-case (over p) probability of observing both E_1 and E_2 is at most

$$\Pr[E_1 \cap E_2] \leq 0.0068.$$

(b) Now, we consider an alternative way of generating the two samples, using the same idea as in the proof of the VC-bound. Here, we sample 10100 passenger show up events independently at random according to an unknown distribution p. We split them into 10000 passengers in the collected sample and 100 passengers booked for the 99-seats flight. Then, we have to bound the probability of observing a sample of 10000 with 95% show ups and a 99-seats flight with all 100 passengers showing up using the sampling protocol.

First, let us denote by

$$K = \sum_{i=1}^{10100} X_i$$

the total number of show-ups in the combined sample of size $N = 10100$. Under our sampling protocol, once we have drawn these N i.i.d. Bernoulli trials, we partition them into the first 10000 (collected sample) and the last 100 (the overbooked flight). Then,

$$\Pr[E_1 \cap E_2] = \Pr[9500/10000 \text{ show up and } 100/100 \text{ show up}],$$

where E_1, E_2 are the events from part (a). Since these two events can only both happen if in total there are exactly $K = 9500 + 100 = 9600$ show-ups, we may condition on K, so

$$\Pr[E_1 \cap E_2] = \Pr[K = 9600] \Pr[E_2 \mid K = 9600].$$

Given $K = 9600$, the number of successes in the last 100 is hypergeometric, such that

$$\Pr[E_2 \mid K = 9600] = \frac{\binom{9600}{100}\binom{10100-9600}{0}}{\binom{10100}{100}} = \frac{\binom{9600}{100}}{\binom{10100}{100}}.$$

Since $\Pr[K = 9600] \leq 1$, we have that

$$\Pr[E_1 \cap E_2] = \Pr[K = 9600] \frac{\binom{9600}{100}}{\binom{10100}{100}} \leq \frac{\binom{9600}{100}}{\binom{10100}{100}}.$$

Finally, we calculate that

$$\frac{\binom{9600}{100}}{\binom{10100}{100}} = \frac{\frac{9600!}{100!\,9500!}}{\frac{10100!}{100!\,10000!}} = \frac{9600!}{9500!}\frac{10000!}{10100!}$$
$$= \frac{9600 \cdot 9599 \cdots 9501}{10100 \cdot 10099 \cdots 10001} = \prod_{j=0}^{99} \frac{9600-j}{10100-j} \approx 0.0061.$$

Hence, under the given sampling protocol, the probability of both seeing exactly 95 % in the "collected" 10000 and 100 % show-ups on the overbooked 100 is bounded by

$$\Pr[E_1 \wedge E_2] \leq \frac{\binom{9600}{100}}{\binom{10100}{100}} = \prod_{j=0}^{99} \frac{9600-j}{10100-j} \approx 0.0061.$$

This bound is slightly sharper than the one from (a), but of the same order of magnitude.

2 PAC Learnability

For a target concept class $\mathcal{C}$ and a target concept $c \in \mathcal{C}$, let $\mathcal{D}^+$ and $\mathcal{D}^-$ be arbitrary distributions over the instances labeled positively and negatively by c, respectively. Define the positive example oracle EX_c^+ as $\mathrm{Ex}(c; \mathcal{D}^+)$ and negative example oracle EX_c^- as $\mathrm{Ex}(c; \mathcal{D}^-)$.

Also, a concept class $\mathcal{C}$ is efficiently positively–negatively PAC learnable by hypothesis class $\mathcal{H}$ if $\forall \epsilon, \delta > 0$, there is a polynomial–time algorithm $\mathcal{A}$, which, given access to EX_c^+ and EX_c^-, outputs a hypothesis $h \in \mathcal{H} \cup \{h_0, h_1\}$ satisfying with probability at least $1 - \delta$

$$\Pr_{x \sim \mathcal{D}^+}[h(x) = 0] \leq \epsilon \quad \text{and} \quad \Pr_{x \sim \mathcal{D}^-}[h(x) = 1] \leq \epsilon.$$

Here, h_0, h_1 are the always zero and the always one functions.

Response to Question 2 (a). Show that if $\mathcal{C}$ is efficiently PAC learnable using $\mathcal{H}$ in the standard PAC model, then $\mathcal{C}$ is efficiently positively–negatively PAC learnable using $\mathcal{H}$.

Proof. We assume that the target concept class $\mathcal{C}$ is efficiently PAC learnable using $\mathcal{H}$ in the standard PAC model using the algorithm $\mathcal{A}$. Given access to the two oracles $\mathrm{EX}_c^+ = \mathrm{Ex}(c; \mathcal{D}^+)$ and $\mathrm{EX}_c^- = \mathrm{Ex}(c; \mathcal{D}^-)$, we can simulate a call to $\mathrm{EX}_c = \mathrm{Ex}(c, \mathcal{D})$ by simply flipping an unbiased coin. First, flip a fair coin. If it comes up "heads," query EX_c^+, otherwise query EX_c^-. Then, return whatever labeled example that oracle gives us.

By construction, this delivers exactly one i.i.d. draw from the distribution $\mathcal{D}$ given by

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

where $\mathcal{D}^+$ and $\mathcal{D}^-$ are arbitrary distributions over the instances labeled positively and negatively by the target concept $c \in \mathcal{C}$, respectively. Let the polynomial-time PAC learner $\mathcal{A}$ for $\mathcal{C}$ w.r.t. $\mathcal{D}$ be given. Feed it the simulated EX_c oracle, with accuracy parameter ϵ and confidence δ. By definition, in time at most $p(d, \mathrm{size}(c), 1/\epsilon, 1/\delta)$ for the polynomial p, the PAC learner $\mathcal{A}$ then outputs a hypothesis $h \in \mathcal{H}$, which satisfies

$$\Pr\left[\mathrm{err}_{\mathcal{D}}(h) \leq \epsilon\right] \geq 1 - \delta.$$

However, since $\mathcal{D} = \frac{1}{2}(\mathcal{D}^- + \mathcal{D}^+)$, for the generalization error of h, we choose δ such that

$$\Pr\left[\mathrm{err}_{\mathcal{D}}(h) \leq \frac{\epsilon}{2}\right] \geq 1 - \delta,$$

where we have invoked $\mathcal{A}$ with accuracy $\epsilon/2$ and confidence δ to guarantee the above bound. By using Definition 2.1 from Mohri et al. (2012), we can write the generalization error as

$$\begin{aligned}
\mathrm{err}_{\mathcal{D}}(h) &= \Pr_{x\sim\mathcal{D}}[h(x) \neq c(x)] \\
&= \frac{1}{2}\left(\Pr_{x\sim\mathcal{D}^-}[h(x) \neq c(x)] + \Pr_{x\sim\mathcal{D}^+}[h(x) \neq c(x)]\right) \\
&= \frac{1}{2}\left(\mathrm{err}_{\mathcal{D}^-}(h) + \mathrm{err}_{\mathcal{D}^+}(h)\right).
\end{aligned}$$

Then, by substituting this expression for $\mathrm{err}_{\mathcal{D}}(h)$ into the bound from before, we obtain

$$\begin{aligned}
\Pr\left[\mathrm{err}_{\mathcal{D}}(h) \leq \frac{\epsilon}{2}\right] \geq 1 - \delta &\Leftrightarrow \Pr\left[\frac{1}{2}\left(\mathrm{err}_{\mathcal{D}^-}(h) + \mathrm{err}_{\mathcal{D}^+}(h)\right) \leq \frac{\epsilon}{2}\right] \geq 1 - \delta \\
&\Leftrightarrow \Pr\left[\mathrm{err}_{\mathcal{D}^-}(h) + \mathrm{err}_{\mathcal{D}^+}(h) \leq \epsilon\right] \geq 1 - \delta \\
&\Leftrightarrow \Pr\left[\mathrm{err}_{\mathcal{D}^-}(h) \leq \epsilon\right] \geq 1 - \delta \,\wedge\, \Pr\left[\mathrm{err}_{\mathcal{D}^+}(h) \leq \epsilon\right] \geq 1 - \delta,
\end{aligned}$$

which exactly implies two-oracle PAC learning with the same computational complexity

$$\Pr_{x\sim\mathcal{D}^+}[h(x) = 0] \leq \epsilon \quad \text{and} \quad \Pr_{x\sim\mathcal{D}^-}[h(x) = 1] \leq \epsilon.$$

with probability at least $1 - \delta$, since on the negative distribution $c(x) = 0$, so

$$\mathrm{err}_{\mathcal{D}^-}(h) = \Pr_{x\sim\mathcal{D}^-}[h(x) \neq c(x)] = \Pr_{x\sim\mathcal{D}^-}[h(x) \neq 0] = \Pr_{x\sim\mathcal{D}^-}[h(x) = 1] \leq \epsilon,$$

and on the positive distribution $c(x) = 1$, so

$$\mathrm{err}_{\mathcal{D}^+}(h) = \Pr_{x\sim\mathcal{D}^+}[h(x) \neq c(x)] = \Pr_{x\sim\mathcal{D}^+}[h(x) \neq 1] = \Pr_{x\sim\mathcal{D}^+}[h(x) = 0] \leq \epsilon.$$

Hence, the output h meets the positive–negative PAC learning criterion. Finally, since we only simulated each call to EX_c by one coin flip plus one call to EX_c^+ or EX_c^-, the total number of oracle-queries and running time remains polynomial in $1/\epsilon$, $1/\delta$ and the input size. Thus, $\mathcal{C}$ is efficiently positively–negatively PAC learnable using $\mathcal{H}$ as claimed. $\square$

Response to Question 2 (b). Show that if $\mathcal{C}$ is efficiently positively–negatively PAC learnable using $\mathcal{H}$, then $\mathcal{C}$ is also efficiently PAC learnable in the standard model.

Proof. We assume that $\mathcal{C}$ is efficiently positively–negatively PAC learnable using $\mathcal{H}$, that is, there exists a polynomial-time learning algorithm $\mathcal{A}$ for all $\epsilon, \delta > 0$ that runs in time $p(d, \text{size}(c), 1/\epsilon, 1/\delta)$ for a polynomial p, which given access to the oracles $\text{EX}_c^+ = \text{Ex}(c; D_c^+)$ and $\text{EX}_c^- = \text{Ex}(c; D_c^-)$ for $c \in \mathcal{C}$ returns a hypothesis $h \in \mathcal{H} \cup \{h_0, h_1\}$ satisfying

$$\Pr_{x \sim \mathcal{D}^+}[h(x) = 0] \leq \epsilon \quad \text{and} \quad \Pr_{x \sim \mathcal{D}^-}[h(x) = 1] \leq \epsilon$$

with probability at least $1 - \delta$, where h_0 and h_1 are always zero and always one functions.

First, for the learning algorithm $\mathcal{A}$ with the configurations stated above, there exists polynomials m^+ and m^- in $1/\epsilon$, $1/\delta$ and size(c), such that, provided it is given at least m^+ positive examples and m^- negative examples, outputs $h \in \mathcal{H} \cup \{h_0, h_1\}$, which satisfies

$$\Pr[\text{err}_{\mathcal{D}^-}(h)] \leq \epsilon \quad \text{and} \quad \Pr[\text{err}_{\mathcal{D}^+}(h)] \leq \epsilon$$

with probability at least $1 - \delta$. Suppose now that $\mathcal{D}$ is a probability distribution over the m^+ positive and m^- negative examples. If the m examples were drawn from $\mathcal{D}$ such that $m \geq \max\{m^-, m^+\}$ for a polynomial m in $1/\epsilon$, $1/\delta$ and size(c), then we would have that

$$\begin{aligned}
\Pr[\text{err}_{\mathcal{D}}(h)] &\leq \Pr[\text{err}_{\mathcal{D}}(h) | c(x) = 0] \Pr[c(x) = 0] + \Pr[\text{err}_{\mathcal{D}}(h) | c(x) = 1] \Pr[c(x) = 1] \\
&\leq \epsilon(\Pr[c(x) = 0] + \Pr[c(x) = 1]) \\
&= \epsilon
\end{aligned}$$

with probability at least $1 - \delta$ using law of total probability, which means that the positively-negatively PAC learner $\mathcal{C}$ also would be efficiently PAC learnable in the standard model.

To handle an arbitrary distribution $\mathcal{D}$, we proceed by drawing

$$m = \max\left\{\frac{2m^+}{\epsilon}, \frac{2m^-}{\epsilon}, \frac{8}{\epsilon}\log\frac{4}{\delta}\right\}$$

examples (both positive and negative instances) from the oracle $\text{Ex}(c; \mathcal{D})$ and look at

- **Case 1 (Too few positives)**: If there are fewer than m^+ positive examples in our sample, the algorithm outputs h_0.
- **Case 2 (Too few negatives)**. If there are fewer than m^- negative examples in our sample, the algorithm outputs h_1.
- **Case 3 (Well-balanced sample)**: Otherwise, it selects any m^+ points and any m^- points from the sample, feeds these to $\mathcal{A}$ and outputs the resulting hypothesis h.

We now show that with probability at least $1-\delta$, the algorithm return h of error at most ϵ under $\mathcal{D}$. First, let $\alpha = \Pr_{x\sim\mathcal{D}}[h(x)=1]$. For the well-balanced case, suppose that $\alpha \geq \epsilon$ and $1-\alpha \geq \epsilon$. Then, each sample is positive with probability $\alpha \geq \epsilon$ and negative with probability $1-\alpha \geq \epsilon$. Define for each $i \in [m]$, the total number of positive examples by

$$Z_i = \begin{cases} 1 & \text{if the } i\text{-th draw is positive} \\ 0 & \text{otherwise} \end{cases}, \quad Z = \sum_{i=1}^{m} Z_i.$$

Clearly, $\mathbb{E}[Z] = \alpha m$. By the Chernoff bound with $\alpha \geq \epsilon$, we get that

$$\Pr\left[Z < \frac{1}{2}\alpha m\right] = \Pr\left[Z < \left(1-\frac{1}{2}\right)\alpha m\right] \leq \exp\left(-\frac{\alpha m}{8}\right) \leq \exp\left(-\frac{\epsilon m}{8}\right) \leq \frac{\delta}{4}.$$

Since $m \geq 2m^+/\epsilon$, half the expectation $\frac{1}{2}\alpha m$ is at least m^+, so $\Pr[\#\text{positives} < m^+] \leq \delta/4$ and, if $1-\alpha \geq \epsilon$, we have $\Pr[\#\text{negatives} < m^-] \leq \delta/4$. If both $\alpha \geq \epsilon$ and $1-\alpha \geq \epsilon$, then, by the union bound, with probability at least $1-\delta/2$, the training set contains at least m^+ and m^- positive and negative instances, respectively. Then, with further probability at least $1-\delta/2$, the positively-negatively PAC-guarantee gives us that

$$\Pr[\mathrm{err}_{\mathcal{D}^-}(h)] \leq \epsilon \quad \text{and} \quad \Pr[\mathrm{err}_{\mathcal{D}^+}(h)] \leq \epsilon$$

Under the mixture $\mathcal{D} = \alpha\mathcal{D}^+ + (1-\alpha)\mathcal{D}^-$, the overall error is

$$\Pr[\mathrm{err}_{\mathcal{D}}(h)] = \alpha \Pr[\mathrm{err}_{\mathcal{D}^+}(h)] + (1-\alpha)\Pr[\mathrm{err}_{\mathcal{D}^-}(h)] \leq \alpha\epsilon + (1-\alpha)\epsilon = \epsilon.$$

Next, consider the "too few positives" case. Suppose $\alpha = \Pr_{x\sim\mathcal{D}}[c(x)=1] < \epsilon$ and our sample of m points contains fewer than m^+ positives. Then the algorithm returns h_0, which satisfies $\Pr[\mathrm{err}_{\mathcal{D}}(h_0)] = \alpha < \epsilon$. Moreover, by the same Chernoff-tail argument used above, we have $\Pr[\#\text{positives} \geq m^+] \leq \delta/4$. Now, consider the "too few negatives" case. If $1-\alpha = \Pr_{x\sim\mathcal{D}}[c(x)=0] < \epsilon$ and fewer than m^- negatives appear, the algorithm returns h_1 and hence $\Pr[\mathrm{err}_{\mathcal{D}}(h_1)] = 1-\alpha < \epsilon$. Again, we get $\Pr[\#\text{negatives} \geq m^-] \leq \delta/4$.

Combining these two failure events $(\delta/4 + \delta/4)$ with the $\delta/2$ failure in the well-balanced case, yields total failure probability δ. Therefore, with probability $1-\delta$ the output h satisfies $\Pr[\mathrm{err}_{\mathcal{D}}(h)] \leq \epsilon$, and the running time remains polynomial in $1/\epsilon$, $1/\delta$ and $\mathrm{size}(c)$. Hence, $\mathcal{C}$ is efficiently PAC learnable using $\mathcal{H}$ in the standard model. □

3 Growth Function

Response to Question 3.1. We let $\mathcal{H}$ be a finite hypothesis set with $|\mathcal{H}| = M$ hypotheses and have to prove that $m_{\mathcal{H}}(n) \leq \min\{M, 2^n\}$.

Proof. First, we recall that the growth function of a hypothesis class $\mathcal{H}$ is defined as

$$m_{\mathcal{H}}(n) := \max_{x_1,\ldots,x_n \in \mathcal{X}} |\Pi_{\mathcal{H}}(x_1,\ldots,x_n)|,$$

where $\Pi_{\mathcal{H}}(x_1, \ldots, x_n) = \{(h(x_1), \ldots, h(x_n)) \colon h \in \mathcal{H} \subseteq \{0,1\}^{\mathcal{X}}\}$ for a finite sequence of instances $x_1, \ldots, x_n \in \mathcal{X}$. Thus, for any fixed sequence $x_1, \ldots, x_n$, each hypothesis $h \in \mathcal{H}$ induces exactly one labeling $(h(x_1), \ldots, h(x_n))$. Hence, by assumption, we have that the number of distinct labelings that $\mathcal{H}$ can induce on any n set of points satisfies the inequality

$$|\Pi_{\mathcal{H}}(x_1, \ldots, x_n)| \leq |\mathcal{H}| = M,$$

since we cannot have more dichotomies than hypotheses, i.e., since there are only $|\mathcal{H}|$ hypotheses, we cannot realize more than $|\mathcal{H}|$ labelings. Taking the maximum over all choices of $(x_1, \ldots, x_n)$ gives

$$m_{\mathcal{H}}(n) \leq M.$$

On the other hand, on n points there are at most 2^n possible binary labelings, so

$$\Pi_{\mathcal{H}}(x_1, \ldots, x_n) \subseteq \{0,1\}^n \quad \Longrightarrow \quad |\Pi_{\mathcal{H}}(x_1, \ldots, x_n)| \leq 2^n.$$

Hence, we also have that

$$m_{\mathcal{H}}(n) \leq 2^n.$$

Combining the two bounds then yields $m_{\mathcal{H}}(n) \leq \min\{M, 2^n\}$ as claimed. □

Response to Question 3.2. We let $\mathcal{H}$ be a hypothesis space with 2 hypotheses (i.e., $|\mathcal{H}| = 2$) and have to prove that $m_{\mathcal{H}}(n) = 2$.

Proof. The claim is that if $\mathcal{H}$ has exactly two distinct hypotheses, then for every $n \geq 1$,

$$m_{\mathcal{H}}(n) = \max_{x_1, \ldots, x_n \in \mathcal{X}} |\{(h(x_1), \ldots, h(x_n)) \colon h \in \mathcal{H}\}| = 2.$$

From the trivial bound on a finite class in Question 1, any hypothesis class of size M can induce at most M dichotomies on n points. Here $M = 2$, so $m_{\mathcal{H}}(n) \leq |\mathcal{H}| = M = 2$.

Let $\{h_1, h_2\}$ with $h_1 \neq h_2$. Since the two hypotheses in $\mathcal{H}$ are distinct by definition, there is at least one point $x^* \in \mathcal{X}$ on which they differ, such that $h_1(x^*) \neq h_2(x^*)$. Now, we choose our n-point sample so that x^* is among $(x_1, \ldots, x_n)$. On that sample, we have that

$$(h_1(x_1), \ldots, h_1(x_n)) \quad \text{and} \quad (h_2(x_1), \ldots, h_2(x_n))$$

differ in the coordinate corresponding to x^*, so they are two distinct labellings. Thus for this choice of sample we realize at least 2 dichotomies, and hence we get

$$m_{\mathcal{H}}(n) = \max_{x_1, \ldots, x_n \in \mathcal{X}} |\{(h(x_1), \ldots, h(x_n)) \colon h \in \mathcal{H}\}| \geq 2.$$

Combining the two inequalities, then gives us $m_{\mathcal{H}}(n) = 2$ as desired. □

Response to Question 3.3. We have to prove that $m_{\mathcal{H}}(2n) \leq m_{\mathcal{H}}(n)^2$.

Proof. First, the growth function of a hypothesis class $\mathcal{H}$ in this case is defined as

$$m_{\mathcal{H}}(2n) := \max_{x_1,\ldots,x_{2n}\in\mathcal{X}} |\Pi_{\mathcal{H}}(x_1,\ldots,x_{2n})|,$$

where $\Pi_{\mathcal{H}}(x_1,\ldots,x_{2n}) = \{(h(x_1),\ldots,h(x_{2n}))\colon h \in \mathcal{H} \subseteq \{0,1\}^{\mathcal{X}}\}$ for a finite sequence of instances $x_1,\ldots,x_{2n} \in \mathcal{X}$. Any dichotomy on those $2n$ points is completely determined by what happens on the first n and on the last n. Concretely, we write

$$X_1 = (x_1,\ldots,x_n) \in \mathbb{R}^n, \quad X_2 = (x_{n+1},\ldots,x_{2n}) \in \mathbb{R}^n.$$

Then, the map $h \mapsto (h(X_1), h(X_2))$ embeds $\Pi_{\mathcal{H}}(x_1,\ldots,x_{2n})$ into the Cartesian product $\Pi_{\mathcal{H}}(X_1)(x_1,\ldots,x_n) \times \Pi_{\mathcal{H}}(x_{n+1},\ldots,x_{2n})$ with dimension $\mathbb{R}^n \times \mathbb{R}^n = \mathbb{R}^{2n}$, whose generic element is the original concatenated $2n$-vector $(X_1,X_2) = (x_1,\ldots,x_n,x_{n+1},\ldots,x_{2n})$.

We have $\Pi_{\mathcal{H}}(x_1,\ldots,x_n) \subseteq \{0,1\}^{X_1}$ and $\Pi_{\mathcal{H}}(x_{n+1},\ldots,x_{2n}) \subseteq \{0,1\}^{X_2}$. By construction, every full dichotomy on the $2n$ points is just $(h(X_1),h(X_2)) \in \Pi_{\mathcal{H}}(x_1,\ldots,x_n) \times \Pi_{\mathcal{H}}(x_{n+1},\ldots,x_{2n})$. Hence, $\Pi_{\mathcal{H}}(x_1,\ldots,x_{2n}) \subseteq \Pi_{\mathcal{H}}(x_1,\ldots,x_n) \times \Pi_{\mathcal{H}}(x_{n+1},\ldots,x_{2n})$. Thus, for any fixed choice of the $2n$ points, we get

$$|\Pi_{\mathcal{H}}(x_1,\ldots,x_{2n})| \leq |\Pi_{\mathcal{H}}(x_1,\ldots,x_n)| \cdot |\Pi_{\mathcal{H}}(x_{n+1},\ldots,x_{2n})|.$$

Now, when taking the maximum over all choices of $x_1,\ldots,x_{2n} \in \mathcal{X}$ on the left, and over all choices of n points in each block on the right, it yields

$$\begin{aligned}
m_{\mathcal{H}}(2n) &= \max_{x_1,\ldots,x_{2n}} |\Pi_{\mathcal{H}}(x_1,\ldots,x_{2n})| \\
&\leq \left(\max_{x_1,\ldots,x_n} |\Pi_{\mathcal{H}}(x_1,\ldots,x_n)|\right) \cdot \left(\max_{x_{n+1},\ldots,x_{2n}} |\Pi_{\mathcal{H}}(x_{n+1},\ldots,x_{2n})|\right) \\
&= m_{\mathcal{H}}(n) \cdot m_{\mathcal{H}}(n) \\
&= m_{\mathcal{H}}(n)^2,
\end{aligned}$$

which is exactly the inequality that we had to show. □

References

Mohri, M., Rostamizadeh, A., , and Talwalkar, A. (2012). *Foundations of Machine Learning.* Massachusetts Institute of Technology, 1st edition.

Seldin, Y. (2025). Machine learning: The science of selection under uncertainty.