3.5, Q1:

Monotonicity of Sample Complexity: Let $\mathcal{H}$ be a hypothesis class for a binary classification task. Suppose that $\mathcal{H}$ is PAC learnable and its sample complexity is given by $m_{\mathcal{H}}(\cdot, \cdot)$. Show that $m_{\mathcal{H}}$ is monotonically increasing in both parameters. That is, show that:

1. for $\delta \in (0, 1)$ for $0 \leq \epsilon_1 \leq \epsilon_2 \leq 1$, show that $m_{\mathcal{H}}(\epsilon_1, \delta) \geq m_{\mathcal{H}}(\epsilon_2, \delta)$.

2. for $\epsilon \in (0, 1)$ for $0 \leq \delta_1 \leq \delta_2 \leq 1$, show that $m_{\mathcal{H}}(\epsilon, \delta_1) \geq m_{\mathcal{H}}(\epsilon, \delta_2)$.

Solution, fixed δ

First recall what it means to be PAC-learnable:

> For every $\epsilon, \delta \in (0, 1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$, for every labelling function $f : \mathcal{X} \to \{0, 1\}$, if the realisability assumption holds with respect to $(\mathcal{H}, \mathcal{D}, f)$, then for all constants $m \geq m_{\mathcal{H}}(\epsilon, \delta)$, on running the learning algorithm on a random i.i.d sample $S \in X^m \sim \mathcal{D}^m$, the algorithm produces a hypothesis h_S, such that with probability $(1 - \delta)$, the learning error $L_{(\mathcal{D},f)}(h) \leq \epsilon$.

ϵ_1 and ϵ_2 are the error rates that we wish to achieve. Intuitively, the smaller the error we want, the more samples we need. Hence $\epsilon_1 \leq \epsilon_2 \implies m_{\mathcal{H}}(\epsilon_1, \delta) \geq m_{\mathcal{H}}(\epsilon_2, \delta)$.

Formally, we shall provide a **proof by contradiction**. Let us assume that:

$$\epsilon_1 \leq \epsilon_2 \implies m_{\mathcal{H}}(\epsilon_1, \delta) \leq m_{\mathcal{H}}(\epsilon_2, \delta) \qquad \text{(assumption for contradiction)}$$

Unwrapping the definition, this means that:

1 Giving the learning algorihm $m_1 = m_{\mathcal{H}}(\epsilon_1, \delta)$ samples, the best error rate we are able to achieve is ϵ_1, $(1 - \delta)$ of the time.

2 Similarly, on giving the learning algorithm learning algorithm $m_2 = m_{\mathcal{H}}(\epsilon_1, \delta)$ samples, the best error rate we are able to achieve is ϵ_2, $(1 - \delta)$ of the time.

Since the fraction $(1 - \delta)$ is the same in both cases, we can focus purely on the error rate.

Recall that $\epsilon_1 < \epsilon_2$ and $(m_2 > m_1)$ from our assumption of contradiction.

Hence, in the m_2 case, we can simply **discard samples** and provide the learning algorithm with m_1 samples, thereby reducing our error rate to ϵ_1. However, we had assumed that for m_2, ϵ_2 was **the best error rate we could have achieved**.

this contradiction arose from our contradictory assumption. Hence, it must be the case that as ϵ decreases, the quantity $m_{\mathcal{H}}(\epsilon, \cdot)$ increses. $\square$.

Solution, fixed ϵ

Once again, recall what it means to be PAC-learnable:

> For every $\epsilon, \delta \in (0, 1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$, for every labelling function $f : \mathcal{X} \to \{0, 1\}$, if the realisability assumption holds with respect to $(\mathcal{H}, \mathcal{D}, f)$, then for all constants $m \geq m_{\mathcal{H}}(\epsilon, \delta)$, on running the learning algorithm on a random i.i.d sample $S \in X^m \sim \mathcal{D}^m$, the algorithm produces a hypothesis h_S, such that with probability $(1 - \delta)$, the learning error $L_{(\mathcal{D},f)}(h) \leq \epsilon$.

δ_1 and δ_2 are the rejection rates that we wish to achieve. Intuitively, the smaller the number of samples we wish to reject, the more general a solution we would need to construct, thus more samples need to be seen. Hence $\delta_1 \leq \delta_2 \implies m_{\mathcal{H}}(\epsilon, \delta_1) \geq m_{\mathcal{H}}(\epsilon, \delta_2)$.

Formally, we once again proceed with contradiction. We assume:

$$\delta_1 \leq \delta_2 \implies m_{\mathcal{H}}(\epsilon, \delta_1) \leq m_{\mathcal{H}}(\epsilon, \delta_2) \qquad \text{(assumption for contradiction)}$$

Unwrapping the definition, this means that:

1 Giving the learning algorihm $m_1 = m_{\mathcal{H}}(\epsilon, \delta_1)$ samples, $(1 - \delta)$ of the time, the learning error will be less than ϵ.

2 Similarly, on giving the learning algorithm learning algorithm $m_2 = m_{\mathcal{H}}(\epsilon_1, \delta)$ samples, $(1 - \delta_2)$ of the time, the learning error will be less that ϵ.

3.5, Q2:

Let $\mathcal{X}$ be a discrete domain, and let $\mathcal{H}_{singleton} \equiv \{[z] : z \in \mathcal{X}\} \cup \{h\}$, where

$$[z] : \mathcal{X} \to \{0, 1\}; \quad [z](x) \equiv \begin{cases} 1 & x = z \\ 0 & \text{otherwise} \end{cases}$$
$$h : \mathcal{X} \to \{0, 1\}; \quad h^{-}(\_) \equiv 0$$

The realizability assumption here implies that the true hypothesis f labels negatively all examples in the domain, perhaps except one.

1. Describe an algorithm that implements the ERM rule for learning $\mathcal{H}_{singleton}$ in the realizable setup.

2. Show that $\mathcal{H}_{singleton}$ is PAC learnable. Prove an upper bound on the sample complexity.

Solution, part (a)

Let $\mathcal{X}$ be the domain, let $f : \mathcal{X} \to \{0, 1\}$ be the underlying target function f that we are trying to approximate using $\mathcal{H}$.

We define the sample loss $L_S(h)$ as the number of elements in S that are mis-classified by h. More formally, $L_S(h) \equiv |\{(x, y) \in \S : h(x) \neq y\}|$.

The ERM algorithm must, given a particular sample set $S \in \mathcal{X}^n \sim \mathcal{D}^n$, provides a function $h_0 \in \mathcal{H} = ERM(S)$ which has minimum sample loss $L_S(h_0)$ across all functions in $\mathcal{H}$.

We can check over the classification of all the samples $s \in S$.

- If all samples $s \in S$ are classified as 0: we return h^- — this will always return 0.
- If some sample $s_1 \in S$ is classified as 1: notice that our hypothesis space $\mathcal{H}$ can only allow us to set *at most one sample to 1*. So, we can pick *any* sample s_1 to create our hypothesis function $h = [s_1]$, since that is the best we can do.

```
def hminus(_): return 0 # h-: sends all samples to 0
def indicator(z): return lambda x: 1 if x == z else 0 #indicator of z
def erm_sample(S):
    # all samples which have label 1
    one_samples = [y for (x, y) in S if y == 1]
    if len(one_samples) == 0: return hminus # send all samples to 0!
    else: # we will have at least on element in one_samples
        return indicator(ones_samples[0])
```

Solution, part (b)

The definition of a hypothesis class $\mathcal{H}$ to be PAC-learnable is that there exists a function $m_{\mathcal{H}} : (0, 1)^2 \to \mathbb{N}$ and a learning algorithm such that:

> For every $\epsilon, \delta \in (0, 1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$, for every labelling function $f : \mathcal{X} \to \{0, 1\}$, if the realisability assumption holds with respect to $(\mathcal{H}, \mathcal{D}, f)$, then for all constants $m \geq m_{\mathcal{H}}(\epsilon, \delta)$, on running the learning algorithm on a random i.i.d sample $S \in X^m \sim \mathcal{D}^m$, the algorithm produces a hypothesis h_S, such that with probability $(1 - \delta)$, the learning error $L_{(\mathcal{D}, f)}(h) \leq \epsilon$.

TODO

4.5, Q1:

Prove that the following two statements are equivalent for any learning algorithm A, any probability distribution $\mathcal{D}$, and any loss function whose loss is in the range $[0, 1]$:

$$\begin{aligned} &(1): \quad \forall \epsilon, \delta > 0, \exists M \equiv m(\epsilon, \delta), \forall m \geq M : \underset{S \sim \mathcal{D}^m}{\mathbb{P}}[L_{\mathcal{D}}(A(S)) > \epsilon] < \delta. \\ &\qquad\quad \Updownarrow \\ &(2): \quad \lim_{m \to \infty} \underset{S \sim \mathcal{D}^m}{\mathbb{E}}[L_{\mathcal{D}}(A(S)) > \epsilon] = 0. \end{aligned}$$

Solution: (2) $\implies$ (1)

We start with (2):

$$(2): \lim_{m \to \infty} \underset{S \sim \mathcal{D}^m}{\mathbb{E}}[L_{\mathcal{D}}(A(S)) > \epsilon] = 0.$$

We then look at the definition of the limit abstractly, for a function $f : \mathbb{N} \to \mathbb{R}$ (In our case, the function takes $m \in \mathbb{N}$ as input and produces $\mathbb{E}[\dots]$ as output):

$$\begin{aligned} &\lim_{m \to \infty} f(m) = y^\star \equiv \\ &\qquad \forall p > 0,\ \exists M \in \mathbb{N}, \forall m \in \mathbb{N}, m \geq M \implies |f(m) - y^\star| < p \end{aligned}$$

Plugging in to our case, we recieve:

$$\forall p > 0,\ \exists M \in \mathbb{N}, \forall m \in \mathbb{N}, m \geq M \implies \left| \underset{S \sim \mathcal{D}^m}{\mathbb{E}}[L_{\mathcal{D}}(A(S)) > \epsilon] - 0 \right| < p$$

Recall that our loss function $L_{\mathcal{D}}$ is non-negative, hence we can remove the $|\cdot|$ and -0 completely, giving:

$$\forall p > 0,\ \exists M \in \mathbb{N}, \forall m \in \mathbb{N}, m \geq M \implies \underset{S \sim \mathcal{D}^m}{\mathbb{E}}[L_{\mathcal{D}}(A(S)) > \epsilon] < p$$

However, note that $L_{\mathcal{D}}(A(S)) > \epsilon$ is a binary random variable, and hence:

$$\underset{S \sim \mathcal{D}^m}{\mathbb{E}}[L_{\mathcal{D}}(A(S)) > \epsilon] = 1 \cdot \underset{S \sim \mathcal{D}^m}{\mathbb{P}}[L_{\mathcal{D}}(A(S)) > \epsilon] + 0 \cdot \underset{S \sim \mathcal{D}^m}{\mathbb{P}}[L_{\mathcal{D}}(A(S)) \not> \epsilon] = \underset{S \sim \mathcal{D}^m}{\mathbb{P}}[L_{\mathcal{D}}(A(S)) > \epsilon]$$

Plugging this in, we get:

$$\forall p > 0,\ \exists M \in \mathbb{N}, \forall m \in \mathbb{N}, m \geq M \implies \underset{S \sim \mathcal{D}^m}{\mathbb{P}}[L_{\mathcal{D}}(A(S)) > \epsilon] < p$$

If we replace p with δ in the above, we recover (1).
Hence, we show that (2) $\implies$ (1). □

Solution: (1) $\implies$ (2)

Since to show that (2) $\implies$ (1) we only ever argued with equalities, we can simply run the proof backwards. □

6.8, Q1:

For two hypothesis classes $\mathcal{H}, \mathcal{H}'$, if $\mathcal{H}' \subseteq \mathcal{H}$ then $\text{VCdim}(H') \leq \text{VCdim}(H)$.

Solution

Recall that the VC dimension of a given set family $\mathcal{H}$ is the size of the largest set C such that H shatters C. That is, the intersection of C with every element is H is equal to the powerset of C:

$$\text{VCdim}(\mathcal{H}) \equiv \max_{C} \{h \cap C : h \in \mathcal{H}\} = 2^C \qquad \text{We denote powerset of } C \text{ by } 2^C$$

Now, if a set family $\mathcal{H}'$ is a subset of another set family $\mathcal{H}$, and if $\mathcal{H}'$ shatters C, then:

$$\begin{aligned}
&\mathcal{H}' \text{ shatters C} \equiv \{h \cap C : h \in \mathcal{H}'\} = 2^C \qquad \text{Given, (1)} \\
&\{h \cap C : h \in \mathcal{H}'\} \subseteq \{h \cap C : h \in \mathcal{H}\} \qquad \text{Since } H' \subseteq H \\
&2^C \subseteq \{h \cap C : h \in \mathcal{H}\} \qquad \text{From (1)}
\end{aligned}$$

Hence, any set that can be shattered by H' can be shattered by H if $H' \subseteq H \implies \text{VCdim}(H') \leq \text{VCdim}(H)$.

On the other hand, clearly if H is larger than H', then H can shatter more. For example, let $H' = \emptyset \subsetneq H$. Then H' can only shatter the empty set, while H can in general shatter sets larger than the empty set. Hence, we have have strict inequality: $\text{VCdim}(\emptyset) < \text{VCdim}(H)$ for example.