Assignment 4:
Komal Wavhal
Student Id -20034443

Exercise 1
Prove the **Ω(d/ϵ)** sample complexity lower bound for realizable PAC learning, where 'd' is the VC dimension.

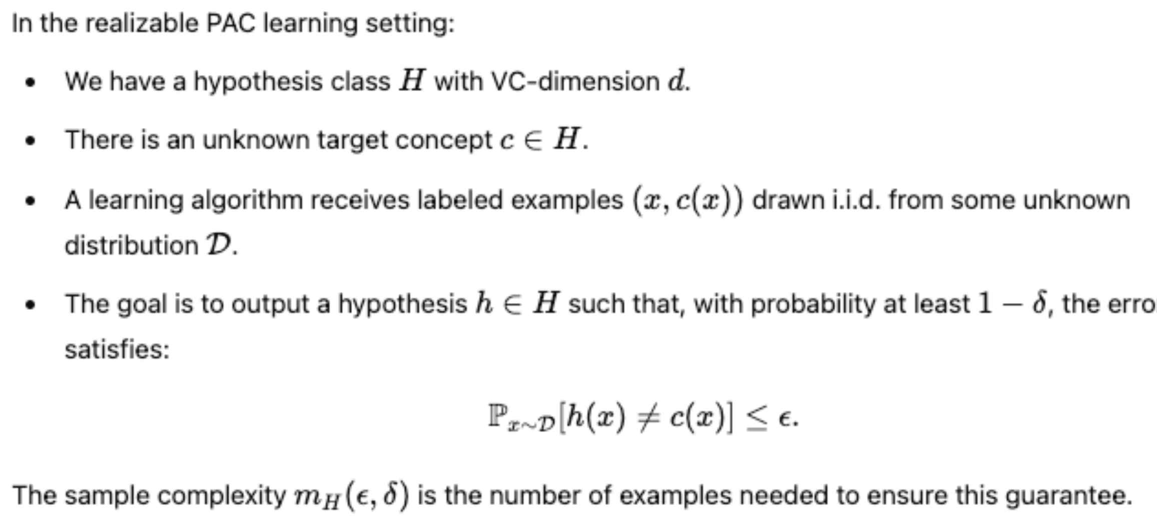
Solution:

The Ω(d/ϵ) sample complexity lower bound for realizable PAC learning states that for any PAC learning algorithm, the number of samples needed to achieve an ϵ-error with high probability is at least on the order of d/ϵ, where d is the VC-dimension of the hypothesis class. This result provides a fundamental limit on how efficiently a learning algorithm can learn from data.

Understanding the Problem:

- **PAC Learning (Probably Approximately Correct):** We aim to learn a hypothesis 'h' from a hypothesis class 'H' that is approximately correct with high probability.
- **Realizable Setting:** We assume that there exists a target function 'f' within our hypothesis class 'H' that perfectly labels all data points.
- **VC Dimension (d):** The VC dimension measures the complexity of a hypothesis class. It's the size of the largest set of points that can be shattered by the hypothesis class. Shattering means that for any possible labeling of those points, there exists a hypothesis in 'H' that achieves that labeling.
- **Sample Complexity:** The number of training examples required to achieve a desired level of accuracy.
- **Lower Bound:** We want to show that *at least* Omega(d/epsilon) samples are necessary in the worst case.

Define PAC Learning in the Realizable Setting:



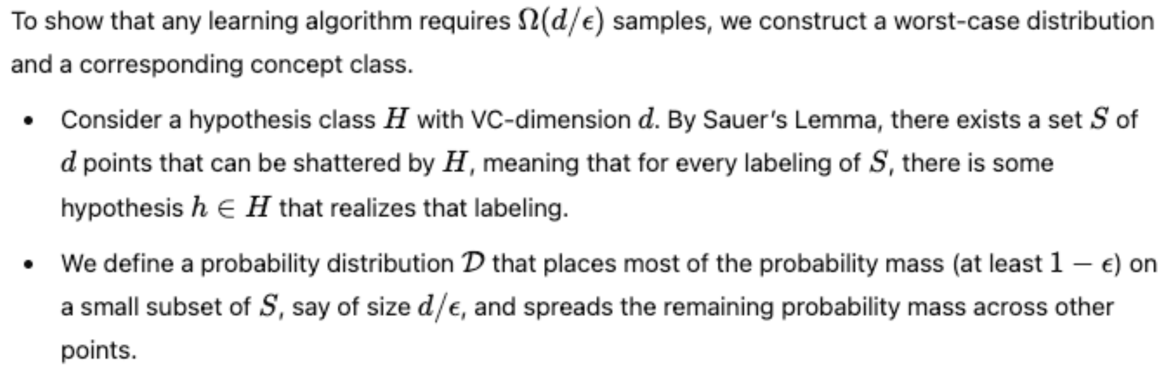
In the realizable PAC learning setting:

- We have a hypothesis class H with VC-dimension d.
- There is an unknown target concept $c \in H$.
- A learning algorithm receives labeled examples $(x, c(x))$ drawn i.i.d. from some unknown distribution $\mathcal{D}$.
- The goal is to output a hypothesis $h \in H$ such that, with probability at least $1 - \delta$, the error of h satisfies:

$$\mathbb{P}_{x \sim \mathcal{D}}[h(x) \neq c(x)] \leq \epsilon.$$

The sample complexity $m_H(\epsilon, \delta)$ is the number of examples needed to ensure this guarantee.

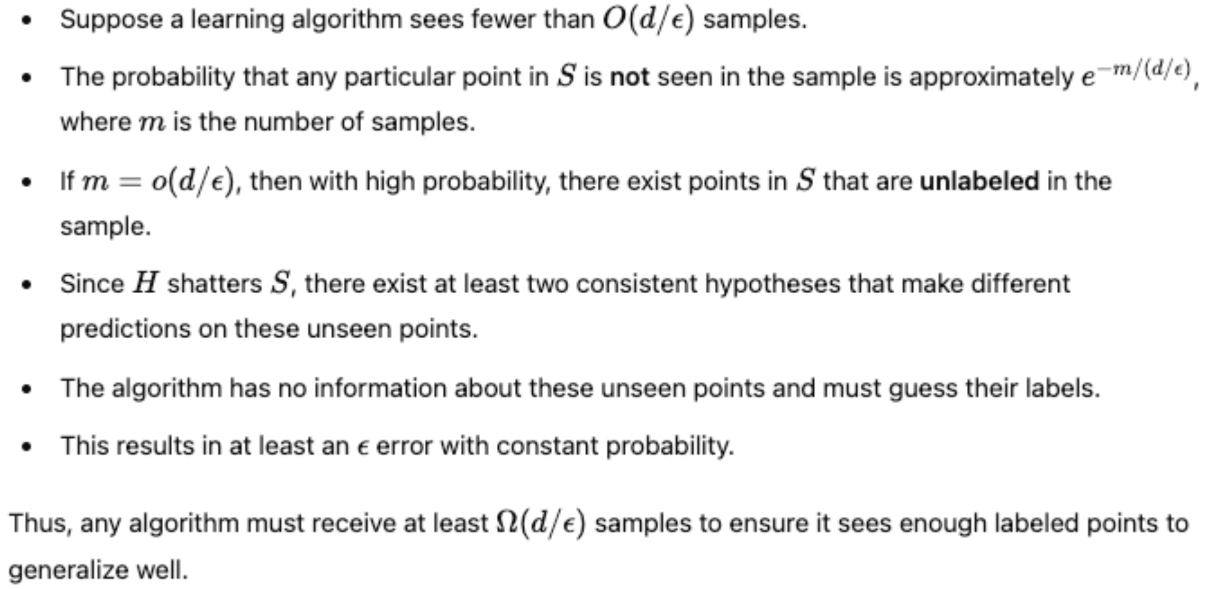
Establish a Hard Instance for Learning



To show that any learning algorithm requires $\Omega(d/\epsilon)$ samples, we construct a worst-case distribution and a corresponding concept class.

- Consider a hypothesis class H with VC-dimension d. By Sauer's Lemma, there exists a set S of d points that can be shattered by H, meaning that for every labeling of S, there is some hypothesis $h \in H$ that realizes that labeling.
- We define a probability distribution $\mathcal{D}$ that places most of the probability mass (at least $1-\epsilon$) on a small subset of S, say of size d/ϵ, and spreads the remaining probability mass across other points.

Show that Fewer Than O(d/ϵ)O(d/\epsilon)O(d/ϵ) Samples Lead to High Error

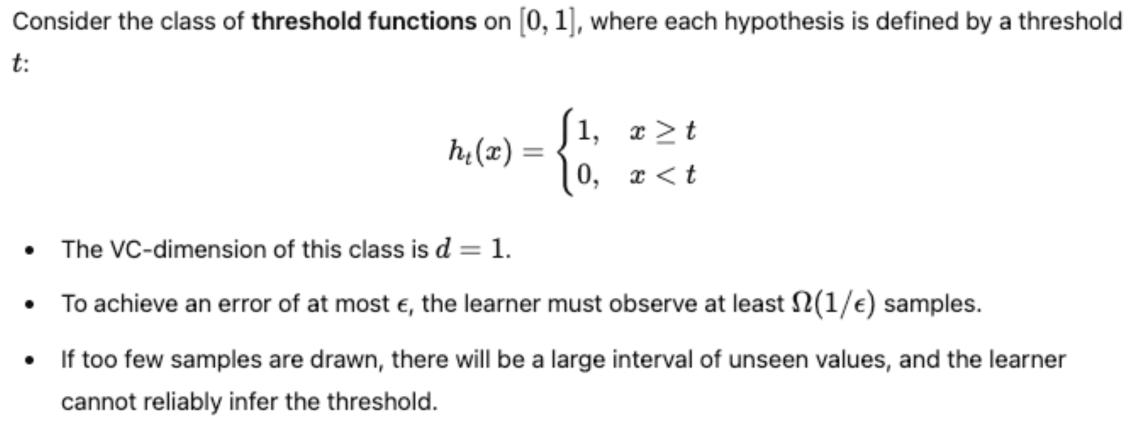


- Suppose a learning algorithm sees fewer than $O(d/\epsilon)$ samples.
- The probability that any particular point in S is **not** seen in the sample is approximately $e^{-m/(d/\epsilon)}$, where m is the number of samples.
- If $m = o(d/\epsilon)$, then with high probability, there exist points in S that are **unlabeled** in the sample.
- Since H shatters S, there exist at least two consistent hypotheses that make different predictions on these unseen points.
- The algorithm has no information about these unseen points and must guess their labels.
- This results in at least an ϵ error with constant probability.

Thus, any algorithm must receive at least $\Omega(d/\epsilon)$ samples to ensure it sees enough labeled points to generalize well.

Since we constructed a worst-case distribution where learning requires at least Ω(d/ϵ) samples to achieve error ϵ, this establishes the lower bound.

Learning a Threshold Function

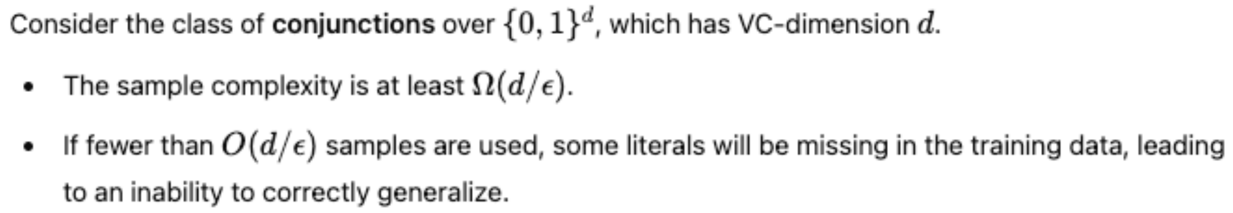


Consider the class of **threshold functions** on $[0,1]$, where each hypothesis is defined by a threshold t:

$$h_t(x) = \begin{cases} 1, & x \geq t \\ 0, & x < t \end{cases}$$

- The VC-dimension of this class is $d = 1$.
- To achieve an error of at most ϵ, the learner must observe at least $\Omega(1/\epsilon)$ samples.
- If too few samples are drawn, there will be a large interval of unseen values, and the learner cannot reliably infer the threshold.

Learning Conjunctions on d-dimensional Boolean Vectors



Consider the class of **conjunctions** over $\{0,1\}^d$, which has VC-dimension d.

- The sample complexity is at least $\Omega(d/\epsilon)$.
- If fewer than $O(d/\epsilon)$ samples are used, some literals will be missing in the training data, leading to an inability to correctly generalize.

The lower bound Ω(d/ϵ) arises because the VC-dimension determines the **worst-case complexity** of the hypothesis class.

A carefully constructed hard instance forces any learning algorithm to need at least d/ ϵ samples to distinguish hypotheses with small error.

This bound matches the upper bound given by uniform convergence arguments in realizable PAC learning, showing it is **tight**.

Ex: Let's say we have a hypothesis class 'H' with a VC dimension of 'd = 10'. We want to learn a hypothesis with an error of 'epsilon = 0.1'.

- According to the lower bound, we need at least Omega(10 / 0.1) = Omega(100) samples.
- If we have fewer than 100 samples, there will be many possible target functions that are consistent with the samples, and we cannot guarantee that our learned hypothesis will have an error less than 0.1.

Key Points:

- The VC dimension plays a crucial role in determining the sample complexity.
- The 'epsilon' parameter determines the accuracy we need to achieve.
- The Omega notation means that the sample complexity grows at least linearly with d/epsilon.
- The proof uses an adversarial argument, meaning it considers the worst-case scenario.

Detailed Steps

1. **Shattered Set:**
    - Since the VC dimension of 'H' is 'd', there exists a set of 'd' points, denoted as S={x1,x2 ,...,xd}, that can be shattered by 'H'.
    - This means that for any possible labeling of these 'd' points, there is a hypothesis in 'H' that agrees with that labeling.
2. **Uniform Distribution:**
    - Consider a uniform distribution 'D' over the points in 'S'. This means that each point in 'S' has a probability of 1/d of being sampled.
3. **Adversarial Labeling:**
    - Imagine an adversary who chooses a specific labeling of the points in 'S'. This labeling represents the target function 'f'.
    - The adversary can choose any of the 2d possible labelings.
4. **Limited Samples:**

- Suppose we have fewer than $c \cdot d/\epsilon$ samples, where 'c' is a constant. Let's denote the sample size as 'm'.
- Since 'm' is less than $c \cdot d/\epsilon$, there will be many labelings of the 'd' points that are consistent with the 'm' samples.
- This means that the learner has to choose a hypothesis based on very limited information.

5. **Bad Event:**
   - The bad event we want to show is that the learner's chosen hypothesis 'h' will have a high error on the distribution 'D' with a non-negligible probability.
   - Specifically, we want to show that there is a good chance that 'h' will disagree with 'f' on a significant portion of the points in 'S'.
6. **Probability of Error:**
   - Since there are 2d possible labelings, and the learner has only seen a limited number of samples, there will be many labelings that are consistent with the observed samples.
   - Consider a labeling 'f' that disagrees with the learner's hypothesis 'h' on at least $\epsilon \cdot d$ points.
   - The probability of such a disagreement is related to the number of points that are misclassified. Because the distribution is uniform, the error is the proportion of points misclassified.
   - Because the number of samples is less than $c \cdot d/\epsilon$, it can be shown that there is a non-zero probability that the learner's hypothesis will have an error greater than epsilon.
7. **Conclusion:**
   - This implies that if we have fewer than $c \cdot d/\epsilon$ samples, we cannot guarantee that the learner's hypothesis will have an error less than 'epsilon' with high probability.
   - Therefore, we need at least Omega(d/epsilon) samples to achieve PAC learnability.