

---

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Supervised learning . . . . .	3
1.2	Unsupervised learning . . . . .	3
1.3	Reinforcement learning . . . . .	3
<b>2</b>	<b>Statistical Learning</b>	<b>5</b>
2.1	Finite hypothesis class . . . . .	5
<b>3</b>	<b>Probably Approximately Correct Learning</b>	<b>7</b>
3.1	Generalized loss functions . . . . .	8
<b>4</b>	<b>Learning via Uniform Convergence</b>	<b>9</b>
<b>5</b>	<b>Bias-Variance Tradeoff</b>	<b>11</b>
5.1	No-Free-Lunch theorem . . . . .	11
5.2	Error decomposition . . . . .	13
<b>6</b>	<b>VC Dimension</b>	<b>15</b>
6.1	VC Dimension . . . . .	15
6.2	Saur's lemma . . . . .	16



---

# Chapter 1

## Introduction

Some kinds of learning include;

### 1.1 Supervised learning

Given a dataset of pair

$$\mathcal{D}_n = \{(x^{(1)}, y^{(1)}), \dots, (x^{(n)}, y^{(n)})\} \quad (1.1)$$

we wish to establish a relationship between  $x^{(i)}$  and  $y^{(i)}$ . Typically,  $x^{(i)} \in \mathbb{R}^d$  is a representation of input, called **feature representation**. Based on the format of the output we can have different types of supervised learning:

**Classification** when the set of possible values of  $y^{(i)}$  is discrete (small finite set). If there two possible values then the classification problem is *binary* otherwise, it is called *multi-class*.

**Regression** when the set of possible values of  $y^{(i)}$  is continuous (or a large finite set). That is,  $y^{(i)} \in \mathbb{R}^k$ .

### 1.2 Unsupervised learning

Given a dataset we wish to find some patterns or structures in it. There are several types of unsupervised learning:

**Density estimation** The data is i.i.d from some distribution  $p_X(x)$ . The goal is to predict the probability  $p_X(x^{(n+1)})$ .

**Clustering** the goal is to find a partitioning of the sample data that groups together samples that are similar. Clustering is sometimes used in density estimation.

**Dimensionality reduction** the goal is to re-represent the same data in  $\mathbb{R}^l$  where  $l < d$ .

### 1.3 Reinforcement learning

The goal is to learn a mapping from input values to output values without a direct supervision signal. There is no training set specified *a priori*. Instead, the learning problem is framed as an agent interacting with an environment. Looking at input as our states and output as a

transition between states, we can assign a reward value  $r^{(i,j)}$  to each such transition. We aim to find a policy  $\pi$  that maximizes the long-term sum or average of rewards.

---

# Chapter 2

## Statistical Learning

A statistical learner needs to know the *domain set*,  $\mathcal{X}$ , *label set*,  $\mathcal{Y}$ , and a training data set (more like a sequence)  $S \subset \mathcal{X} \times \mathcal{Y}$ . Given these, learner outputs a *predictor*  $h : \mathcal{X} \rightarrow \mathcal{Y}$  which is also called *hypothesis* or *classifier*. We can assume that  $\mathcal{D} = \mathbb{P}_{\mathcal{X}}$  is the distribution on  $\mathcal{X}$  and there exists a correct function  $f$  that for each sampled  $x$  output the corresponding label  $y = f(x)$ . Then the *error* of  $h$  is defined as

$$L_{\mathcal{D},f}(h) = \mathbb{P}(h(x) \neq f(x))$$

Since we know neither  $f$  nor  $\mathcal{D}$  we can not find the exact error. To approximate this error, we can use the *empirical error*.

$$L_S(h) = \frac{|\{i \mid h(x_i) \neq y_i\}|}{|S|}$$

Since  $S$  is a representation of the real distribution it makes sense to minimize  $L_S(h)$  and expect that  $L_{\mathcal{D},f}(h)$  is minimized as well. This is called **empirical risk minimization** or ERM for short. *Overfitting* is one drawback of ERM which arises when  $S$  is not fully representative of  $\mathcal{D}$ . In that case, predictor though working well on the training data, fails to generalize and mislabels the new data.

One way to avoid overfitting is to restrict possible hypotheses to a class of hypothesis  $\mathcal{H}$ . Then

$$ERM_{\mathcal{H}}(S) \in \underset{h \in \mathcal{H}}{\operatorname{argmin}} L_S(h)$$

This way, we increase the bias toward  $\mathcal{H}$  and possibly increasing the true error.

### 2.1 Finite hypothesis class

Suppose  $\mathcal{D}$  is finite and assume that there exists a  $h^* \in \mathcal{H}$  such that

$$L_{\mathcal{D},f}(h^*) = 0$$

This is called the *realizability assumption*. Furthermore, we can assume that training data are selected independent of each other.

We often assign a probability  $\delta$  to getting a non-representative training data.  $1 - \delta$  is called the *confidence parameter*. We then assign an *accuracy parameter*  $\epsilon$  where  $L_{\mathcal{D},f}(h_S) > \epsilon$  is a failure. We wish to find an upperbound for the probability of getting a training data that results in a failure.

$$\mathbb{P}(S \text{ s.t. } L_{\mathcal{D},f}(h_S) > \epsilon)$$

Let  $\mathcal{H}_B$  be the set of bad hypotheses

$$\mathcal{H}_B = \{h \mid L_{\mathcal{D},f}(h) > \epsilon\}$$

and  $M$  the set of misleading samples

$$M = \{S \mid \exists h \in \mathcal{H}_B, L_S(h) = 0\}$$

---

## Chapter 3

# Probably Approximately Correct Learning

**Definition:** A hypothesis class  $\mathcal{H}$  is PAC learnable if there exist a function  $m_{\mathcal{H}} : ]0, 1]^2 \rightarrow \mathbb{N}$  and a learning algorithm such that:

- For every  $\epsilon, \delta \in ]0, 1[$ , distribution  $\mathcal{D}$  over  $\mathcal{X}$ , and labeling function  $f : \mathcal{X} \rightarrow \{0, 1\}$
- If the realizable assumption hold with respect to  $\mathcal{H}, \mathcal{D}, f$
- Then, when running the algorithm on  $m \geq m_{\mathcal{H}}(\epsilon, \delta)$  of i.i.d. samples generated by  $\mathcal{D}$  and labeled by  $f$ , the algorithm returns a hypothesis  $h$  such that

$$\mathbb{P}(L_{\mathcal{D}, f}(h) \leq \epsilon) \geq 1 - \delta$$

**Remark 1.** The minimal function  $m_{\mathcal{H}}$  determines the sample complexity of learning  $\mathcal{H}$ .

**Corollary 3.1.** *Every finite hypothesis class is PAC learnable with sample complexity*

$$m_{\mathcal{H}}(\epsilon, \delta) \leq \left\lceil \frac{\log(|\mathcal{H}|/\delta)}{\epsilon} \right\rceil$$

Let  $\mathcal{J}$  be the joint distribution over  $\mathcal{X} \times \mathcal{Y}$ . Note that,  $\mathcal{D}$  is the marginal distribution of  $\mathcal{J}$ . Then we can revise the definition for the true error

$$L_{\mathcal{J}}(h) = \mathbb{P}_{(x,y) \sim \mathcal{J}}(h(x) \neq y)$$

Then given  $\mathcal{J}$  the best label prediction function is

$$f_{\mathcal{J}}(x) = \begin{cases} 1 & \text{if } \mathbb{P}(y = 1 \mid x) \geq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

that is, there is no other classifier  $g$  with  $L_{\mathcal{J}}(g) < L_{\mathcal{J}}(f_{\mathcal{J}})$

**Definition:** A hypothesis  $\mathcal{H}$  is **agnostic PAC learnable** if there exist a function  $m_{\mathcal{H}} : ]0, 1]^2 \rightarrow \mathbb{N}$  and a learning algorithm such that:

- For every  $\epsilon, \delta \in ]0, 1[$ , distribution  $\mathcal{J}$  over  $\mathcal{X} \times \mathcal{Y}$
- Then, when running the algorithm on  $m \geq m_{\mathcal{H}}(\epsilon, \delta)$  of i.i.d. samples generated by  $\mathcal{D}$  and labeled by  $f$ , the algorithm returns a hypothesis  $h$  such that

$$\mathbb{P}\left(L_{\mathcal{J}}(h) \leq \min_{h'} L_{\mathcal{J}}(h') + \epsilon\right) \geq 1 - \delta$$

### 3.1 Generalized loss functions

Given any set  $\mathcal{H}$  and some domain  $Z$ , let  $l$  be any function from  $\mathcal{H} \times Z$  to  $\mathbb{R}_+$ . We call such functions *loss functions*. We then define the risk function to be

$$L_{\mathcal{Z}}(h) = \mathbb{E}_{\mathcal{Z}}[l(h, z)]$$

where  $h \in \mathcal{H}$ , and  $\mathcal{Z}$  is the distribution on  $Z$ . Similarly, the empirical risk over a given sample  $S \in Z^m$  is

$$L_S(h) = \frac{1}{m} \sum_{i=1}^m l(h, z_i)$$

Then revising the agnostic PAC learnability definition for general loss function gives

**Definition:** A hypothesis  $\mathcal{H}$  is **agnostic PAC learnable** with respect to  $Z$  and a loss function  $l : \mathcal{H} \times Z \rightarrow \mathbb{R}_+$ , if there exist a function  $m_{\mathcal{H}} : ]0, 1]^2 \rightarrow \mathbb{N}$  and a learning algorithm such that:

- For every  $\epsilon, \delta \in ]0, 1[$ , distribution  $\mathcal{Z}$  over  $Z$
- Then, when running the algorithm on  $m \geq m_{\mathcal{H}}(\epsilon, \delta)$  of i.i.d. samples generated by  $\mathcal{Z}$  and labeled by  $f$ , the algorithm returns a hypothesis  $h$  such that

$$\mathbb{P}\left(L_{\mathcal{Z}}(h) \leq \min_{h'} L_{\mathcal{Z}}(h') + \epsilon\right) \geq 1 - \delta$$

**Remark 2.** In some situations,  $\mathcal{H}$  is a subset of a set  $\mathcal{H}'$ , and the loss function can be naturally extended to be a function from  $\mathcal{H}' \times Z$ . In this cases, we may allow the algorithm to return a hypothesis  $h' \in \mathcal{H}'$  as long as it satisfies the requirement

$$\mathbb{P}\left(L_{\mathcal{Z}}(h') \leq \min_{h \in \mathcal{H}} L_{\mathcal{Z}}(h) + \epsilon\right) \geq 1 - \delta$$

This is called *representation independent learning*, or *improper learning*.



---

# Chapter 4

## Learning via Uniform Convergence

**Definition:** A training set  $S$  is called  $\epsilon$ -representative (w.r.t domain  $Z$ , hypothesis class  $\mathcal{H}$ , loss function  $l$ , and distribution  $\mathcal{Z}$ ) if

$$\forall h \in \mathcal{H}, \quad |L_S(h) - L_{\mathcal{Z}}(h)| < \epsilon$$

**Lemma 4.1.** Suppose  $S$  is  $\frac{\epsilon}{2}$ -representative. Then, any output of  $h_S = \text{ERM}_{\mathcal{H}}(S)$  satisfies

$$L_{\mathcal{Z}}(h_S) \leq \min_{h \in \mathcal{H}} L_{\mathcal{Z}}(h) + \epsilon$$

*Proof.* Let  $h_{\mathcal{Z}}$  be the  $\text{argmin}_h L_{\mathcal{Z}}(h)$  then

$$\begin{aligned} |L_{\mathcal{Z}}(h_S) - L_{\mathcal{Z}}(h_{\mathcal{Z}})| &= L_{\mathcal{Z}}(h_S) - L_{\mathcal{Z}}(h_{\mathcal{Z}}) \\ &= |L_{\mathcal{Z}}(h_S) - L_S(h_S)| + |L_S(h_S) - L_S(h_{\mathcal{Z}})| + |L_S(h_{\mathcal{Z}}) - L_{\mathcal{Z}}(h_{\mathcal{Z}})| \\ &= \frac{\epsilon}{2} + \frac{\epsilon}{2} + 0 = \epsilon \end{aligned}$$

Another way to prove the statement

$$L_{\mathcal{Z}}(h_S) \leq L_S(h_S) + \frac{\epsilon}{2} \leq L_S(h) + \frac{\epsilon}{2} \leq L_{\mathcal{Z}}(h) + \epsilon, \quad \forall h \in \mathcal{H}$$

**Definition:**  $\mathcal{H}$  has uniform convergence property (w.r.t domain  $Z$  and loss function  $l$ ) if there exists a  $m_{\mathcal{H}}^{UC} : ]0, 1]^2 \rightarrow \mathbb{N}$  such that for all  $\epsilon, \delta \in ]0, 1[$ , for every distribution  $\mathcal{Z}$  over  $Z$ , if  $|S| \geq m_{\mathcal{H}}^{UC}(\epsilon, \delta)$  and  $S$  is i.i.d, then with probability at least  $1 - \delta$ ,  $S$  is  $\epsilon$ -representative.

**Proposition 4.2.** If  $\mathcal{H}$  has uniform convergent property with sample complexity  $m_{\mathcal{H}}^{UC}$ , then it is agnostically PAC learnable with sample complexity  $m_{\mathcal{H}}(\epsilon, \delta) \leq m_{\mathcal{H}}^{UC}(\frac{\epsilon}{2}, \delta)$ . Furthermore, in that case the  $\text{ERM}_h$  paradigm is a successful agnostic learner for  $\mathcal{H}$ .

**Theorem 4.3.** Every finite class has uniform convergence property and thus it is agnostic PAC learnable.

*Proof.* Fix  $\epsilon, \delta$ , we need to show that for any distribution  $\mathcal{Z}$

$$\mathbb{P}(\{S \mid \forall h, |L_S(h) - L_{\mathcal{Z}}(h)| \leq \epsilon\}) \geq 1 - \delta$$

Equivalently

$$\begin{aligned} \mathbb{P}(\{S \mid \exists h, |L_S(h) - L_Z(h)| > \epsilon\}) &\leq \delta \\ &= \mathbb{P}\left(\bigcup_{h \in \mathcal{H}} \{S \mid |L_S(h) - L_Z(h)| > \epsilon\}\right) \leq \sum_{h \in \mathcal{H}} \mathbb{P}(\{S \mid |L_S(h) - L_Z(h)| > \epsilon\}) \end{aligned}$$

Note that  $\forall h$

$$\mathbb{E}[L_S(h)] = \frac{1}{m} \sum_{i=1}^m \mathbb{E}[l(h, Z_i)] = \mathbb{E}[l(h, Z_1)] = L_Z(h)$$

and recall that by Hoeffding inequality, if  $X_1, \dots, X_n$  are i.i.d. random variable with  $\mathbb{E}[X] = \mu$  and  $X \in [a, b]$  then for any  $\epsilon > 0$

$$\mathbb{P}(|\bar{X} - \mu| > \epsilon) \leq 2 \exp\left(-\frac{2n\epsilon^2}{(b-a)^2}\right)$$

Let  $X_i = l(h, Z_i)$  then  $\bar{X} = L_S(h)$  and  $\mathbb{E}[X] = L_Z(h)$  and assume that range of  $l$  is  $[0, 1]$ , By Hoeffding inequality

$$\mathbb{P}(\{S \mid |L_S(h) - L_Z(h)| > \epsilon\}) \leq 2e^{-2m\epsilon^2}$$

hence

$$\begin{aligned} \mathbb{P}(\{S \mid \exists h, |L_S(h) - L_Z(h)| > \epsilon\}) &\leq \sum_{h \in \mathcal{H}} 2e^{-2m\epsilon^2} \\ &= 2|\mathcal{H}|e^{-2m\epsilon^2} \end{aligned}$$

which implies for

$$m_{\mathcal{H}}^{UC}(\epsilon, \delta) \geq \frac{\log(2|\mathcal{H}|/\delta)}{2\epsilon^2}$$

proves that any finite  $\mathcal{H}$  has the uniform convergence property. ■

**Corollary 4.4.** *Let  $\mathcal{H}$  be a finite class, let  $Z$  be a domain and let  $l : \mathcal{H} \times Z \rightarrow [0, 1]$  be a loss function. Then,  $\mathcal{H}$  is uniformly convergent with sample complexity*

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

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

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

Glivenko-Cantelli Classes.

---

# Chapter 5

## Bias-Variance Tradeoff

### 5.1 No-Free-Lunch theorem

**Theorem 5.1 (No-Free-Lunch).** *Let  $\mathcal{A}$  be any learning algorithm for task of binary classification w.t.r 0-1 loss over a domain  $\mathcal{X}$ . Let  $m$  be any number smaller than  $\frac{|\mathcal{X}|}{2}$ , representing the training set size, Then, there exists a distribution  $\mathcal{J}$  on  $\mathcal{X} \times \{0, 1\}$  such that*

1. *There exists a function  $f : \mathcal{X} \rightarrow \{0, 1\}$  with  $L_{\mathcal{J}}(f) = 0$ .*
2. *Then probability taken over the choice of  $S \sim \mathcal{J}^m$*

$$\mathbb{P}\left(L_{\mathcal{J}}(\mathcal{A}(S)) \geq \frac{1}{8}\right) \geq \frac{1}{7}$$

*Proof.* Intuitively, if  $C \subset X$  with  $|C| = 2m$ , then we want a distribution that with knowing the labeling one half of  $C$ ,  $\mathcal{A}$  can not label the other half correctly. Let  $T = 2^{2m}$  — the number of binary functions from  $C$  — and  $f_i : C \rightarrow \{0, 1\}$ ,  $i = 1, \dots, T$  and let  $\mathcal{J}_i$  be distributions over  $C \times \{0, 1\}$  such that

$$J_i(x, y) = \begin{cases} \frac{1}{|C|} & \text{if } y = f_i(x) \\ 0 & \text{otherwise} \end{cases}$$

We claim that for all  $\mathcal{A}$  that receives  $m$  samples of  $C$

$$\max_{i \in \mathbb{N}_T} \mathbb{E}_{S \sim \mathcal{J}_i^m} [L_{\mathcal{J}_i}(\mathcal{A}(S))] \geq \frac{1}{4} \quad (5.0) \quad \blacksquare$$

Then for every algorithm  $\mathcal{A}'$  that receives  $m$  samples of  $X$ , there exists  $\mathcal{J}$  and  $f$  such with  $L_{\mathcal{J}}(f) = 0$  and

$$\mathbb{E}_{S \sim \mathcal{J}^m} [L_{\mathcal{J}}(\mathcal{A}'(S))] \geq \frac{1}{4}$$

To do this, consider an extension set  $C$  for that  $m$  samples and consider the  $\mathcal{J}_i$  and  $f_i$  on that set. Then, by Markov inequality we have

$$\begin{aligned} \mathbb{P}\left(L_{\mathcal{J}}(\mathcal{A}'(S)) < \frac{1}{8}\right) &= \mathbb{P}\left(1 - L_{\mathcal{J}}(\mathcal{A}'(S)) > \frac{7}{8}\right) \\ &\leq 8 \frac{\mathbb{E}[1 - L_{\mathcal{J}}(\mathcal{A}'(S))]}{7} \\ &\leq \frac{8}{7} - \frac{2}{7} = \frac{6}{7} \implies \mathbb{P}\left(L_{\mathcal{J}}(\mathcal{A}'(S)) \geq \frac{1}{8}\right) \geq 1 - \frac{6}{7} = \frac{1}{7} \end{aligned}$$

Let  $K = (2m)^m$  be the number of sequences of examples of  $C$  (with replacement) and  $S_j$  for  $j = 1, \dots, K$  be these sequences. Define

$$S_j^i = ((x_1, f_i(x_1)), \dots, (x_m, f_i(x_m)))$$

to be the sequence  $S_j$  labeled by  $f_i$ . If  $\mathcal{J}_i$  is the distribution and  $\mathcal{A}$  receives  $S_1^i, \dots, S_K^i$  with equal probability then

$$Q_i = \mathbb{E}_{S \sim \mathcal{J}^i}[L_{\mathcal{J}_i}(\mathcal{A}(S))] = \frac{1}{K} \sum_{j=1}^K L_{\mathcal{J}_i}(\mathcal{A}(S_j^i))$$

which implies

$$\begin{aligned} \max_{i \in \mathbb{N}_T} Q_i &\geq \text{average}_{i \in \mathbb{N}} Q_i = \frac{1}{T} \sum_{i=1}^T \left( \frac{1}{K} \sum_{j=1}^K L_{\mathcal{J}_i}(\mathcal{A}(S_j^i)) \right) \\ &= \text{average}_{j \in \mathbb{N}_K} \frac{1}{T} \sum_{i=1}^T L_{\mathcal{J}_i}(\mathcal{A}(S_j^i)) \\ &\geq \min_{j \in \mathbb{N}_K} \frac{1}{T} \sum_{i=1}^T L_{\mathcal{J}_i}(\mathcal{A}(S_j^i)) \end{aligned}$$

Fix  $j$  and let  $v_1, \dots, v_p \in S$  but not in  $S_j$  —  $p \geq m$ . Then for every  $h : C \rightarrow \{0, 1\}$  and every  $i$

$$\begin{aligned} L_{\mathcal{J}_i}(h) &= \frac{1}{2m} \sum_{x \in C} \mathbb{1}_{[h(x) \neq f_i(x)]} \\ &\geq \frac{1}{2m} \sum_{r=1}^p \mathbb{1}_{[h(v_r) \neq f_i(v_r)]} \\ &\geq \frac{1}{2p} \sum_{r=1}^p \mathbb{1}_{[h(v_r) \neq f_i(v_r)]} \end{aligned}$$

Therefore,

$$\frac{1}{T} \sum_{i=1}^T L_{\mathcal{J}_i}(\mathcal{A}(S_j^i)) \geq \frac{1}{2} \min_{r \in \mathbb{N}_p} \frac{1}{T} \sum_{i=1}^T \mathbb{1}_{[\mathcal{A}(S_j^i)(v_r) \neq f_i(v_r)]}$$

For some  $r \in \mathbb{N}$  partition  $f_1, \dots, f_T$  into  $T/2$  disjoint pairs  $f_i, f_{i'}$  where

$$\forall c \in C, f_i(c) \neq f_{i'}(c) \iff c = v_r$$

For any such pair, we must have  $S_j^i = S_j^{i'}$  since  $v_r \notin S_j$ , it follows that

$$\begin{aligned} &\mathbb{1}_{[\mathcal{A}(S_j^i)(v_r) \neq f_i(v_r)]} + \mathbb{1}_{[\mathcal{A}(S_j^{i'})(v_r) \neq f_{i'}(v_r)]} = 1 \\ \implies &\frac{1}{T} \sum_{i=1}^T \mathbb{1}_{[\mathcal{A}(S_j^i)(v_r) \neq f_i(v_r)]} = \frac{1}{T} \cdot \frac{T}{2} = \frac{1}{2} \end{aligned}$$

to conclude

$$\max_{i \in \mathbb{N}_T} Q_i \geq \min_{j \in \mathbb{N}_K} \frac{1}{T} \sum_{i=1}^T L_{\mathcal{J}_i}(\mathcal{A}(S_j^i)) \geq \min_{j \in \mathbb{N}_K} \frac{1}{2} \min_{r \in \mathbb{N}_p} \frac{1}{2} = \frac{1}{4}$$

**Corollary 5.2.** *Let  $\mathcal{X}$  be an infinite domain set and  $\mathcal{H}$  be the set of all functions  $f : \mathcal{X} \rightarrow \{0, 1\}$ . Then  $\mathcal{H}$  is not PAC learnable.*

## 5.2 Error decomposition

Let  $h_S \in \text{ERM}_{\mathcal{H}}$  then we can write the risk of  $h_S$  as

$$L_{\mathcal{J}}(h_S) = \epsilon_{app} + \epsilon_{est}$$

where  $\epsilon_{app} = \min_{h \in \mathcal{H}} L_{\mathcal{J}}(h)$ . Intuitively,  $\epsilon_{app}$  is the *approximation error* – also called *inductive bias* – which is the minimum error achieved by  $\mathcal{H}$  and  $\epsilon_{est}$  is the *estimation error* which is due to the fact that  $\text{ERM}_{\mathcal{H}}$  depends on  $m$  and the complexity of  $\mathcal{H}$ . Hence, this is called bias-complexity or bias-variance tradeoff.



---

# Chapter 6

## VC Dimension

### 6.1 VC Dimension

**Definition:** Let  $\mathcal{H}$  be a class of functions from  $\mathcal{X} \rightarrow \{0, 1\}$  and let  $C = \{c_1, \dots, c_m\} \subset \mathcal{X}$ . The *restriction* of  $\mathcal{H}$  to  $C$  is the set of functions from  $C$  to  $\{0, 1\}$  that can be derived from  $\mathcal{H}$ .

$$\mathcal{H}_C = \{(h(c_1), \dots, h(c_m)) \mid h \in \mathcal{H}\}$$

**Definition:** A hypothesis class  $\mathcal{H}$  *shatters* a finite set  $C \subset \mathcal{X}$  if the restriction of  $\mathcal{H}$  to  $C$  is the set of all functions from  $C$  to  $\{0, 1\}$ . That is,  $|\mathcal{H}_C| = 2^{|C|}$ .

Going back to the proof of No-Free-Lunch theorem, if  $C$  is shattered by  $\mathcal{H}$  then adversary is not restricted by  $\mathcal{H}$  and can construct any target function from  $C$  to  $\{0, 1\}$ . Formally,

**Corollary 6.1.** Let  $\mathcal{H}$  be a hypothesis class from  $\mathcal{X}$  to  $\{0, 1\}$ . Let  $m$  be the size of the training set. Assume there exists a set  $C \subset \mathcal{X}$  with size  $2m$  that is shattered by  $\mathcal{H}$ . Then, for any learning algorithm,  $\mathcal{A}$ , there exist a distribution  $\mathcal{J}$  on  $\mathcal{X} \times \{0, 1\}$  and a predictor  $h \in \mathcal{H}$  such that

$$L_{\mathcal{J}}(h) = 0 \text{ but } \mathbb{P}\left(L_{\mathcal{J}}(\mathcal{A}(S)) \geq \frac{1}{8}\right) \geq \frac{1}{7}$$

Intuitively, if  $\mathcal{H}$  shatters  $C$ , knowing  $m$  elements of does not give much information on the other  $m$  elements.

**Definition:** The *VC dimension* of  $\mathcal{H}$  denoted by  $\text{VCdim } \mathcal{H}$  is the maximal size of a set  $C \subset \mathcal{X}$  that can be shattered by  $\mathcal{H}$ . If  $\mathcal{H}$  can shatter set of arbitrarily large size, we say  $\mathcal{H}$  has infinite VC-dimension.

**Theorem 6.2.** Let  $\mathcal{H}$  be a class of infinite VC dimension. Then,  $\mathcal{H}$  is not PAC learnable.

Note that, not all infinite  $\mathcal{H}$  has infinite VC dimension. For example, the hypothesis class  $\mathcal{H} = \{h = \mathbb{I}_{[x < a]} \mid a \in \mathbb{R}\}$  is PAC learnable using the ERM rule with sample complexity of  $m_{\mathcal{H}} \leq \left\lceil \frac{\log 2 - \log \delta}{\epsilon} \right\rceil$ .

Read the examples afterward, I'm tired right now, again :)).

**Theorem 6.3 (Fundamental theorem of statistical learning).** Let  $\mathcal{H}$  be a hypothesis class from  $\mathcal{X}$  to  $\{0, 1\}$  with 0-1 loss. Then, the following are equivalent

1.  $\mathcal{H}$  has uniform convergence property.

2. Any ERM rule a successful agnostic PAC learner for  $\mathcal{H}$ .
3.  $\mathcal{H}$  is agnostic PAC learnable.
4.  $\mathcal{H}$  is PAC learnable.
5. Any ERM rule is a successful PAC learner for  $\mathcal{H}$ .
6.  $\mathcal{H}$  has a finite VC dimension.

Similar theorems can be proved for other learning problems such as regression with absolute value or square loss.

**Corollary 6.4.** *Assume that  $\text{VCdim } \mathcal{H} = d < \infty$ , then there are  $C_1, C_2$  constants such that*

1.  $\mathcal{H}$  is uniform convergence property with sample complexity

$$C_1 \frac{d + \log \frac{1}{\delta}}{\epsilon^2} \leq m_{\mathcal{H}}^{UC}(\epsilon, \delta) \leq C_2 \frac{d + \log \frac{1}{\delta}}{\epsilon^2}$$

2.  $\mathcal{H}$  is agnostic PAC learnable with sample complexity

$$C_1 \frac{d + \log \frac{1}{\delta}}{\epsilon^2} \leq m_{\mathcal{H}}(\epsilon, \delta) \leq C_2 \frac{d + \log \frac{1}{\delta}}{\epsilon^2}$$

3.  $\mathcal{H}$  is PAC learnable with sample complexity

$$C_1 \frac{d + \log \frac{1}{\delta}}{\epsilon^2} \leq m_{\mathcal{H}}(\epsilon, \delta) \leq C_2 \frac{d + \log \frac{1}{\delta}}{\epsilon^2}$$

## 6.2 Saur's lemma

**Definition:** The growth function of  $\mathcal{H}$  denoted by  $\tau_{\mathcal{H}} : \mathbb{N} \rightarrow \mathbb{N}$  is defined as

$$\tau_{\mathcal{H}}(m) = \max_{\substack{C \subseteq \mathcal{X} \\ |C|=m}} |\mathcal{H}_C|$$

**Lemma 6.5 (Saur, Shelah, Parles).** *Let  $\mathcal{H}$  be a hypothesis class with  $\text{VCdim } \mathcal{H} \leq d < \infty$ . Then, for all  $m$*

$$\tau_{\mathcal{H}}(m) \leq \sum_{i=1}^d \binom{m}{i}$$

*In particular of  $m > d + 1$  then*

$$\tau_{\mathcal{H}}(m) \leq \left( \frac{em}{d} \right)^d$$

**Theorem 6.6.** *Let  $\mathcal{H}$  be a hypothesis class of with  $\tau_{\mathcal{H}}$ . Then, for every  $\mathcal{J}$  and for every  $\delta \in ]0, 1[$*

$$\mathbb{P} \left( \left| L_{\mathcal{J}}(h) - L_S(h) \right| \geq \frac{4 + \sqrt{\log(\tau_{\mathcal{H}}(2m))}}{\delta \sqrt{2m}} \right) \geq 1 - \delta$$