

# 台湾大学林轩田机器学习基石课程学习笔记7 -- The VC Dimension

作者：红色石头

微信公众号：AI有道 ( ID : redstonewill )

前几节课着重介绍了机器能够学习的条件并做了详细的推导和解释。机器能够学习必须满足两个条件：

- 假设空间H的Size  $M$ 是有限的，即当 $N$ 足够大的时候，那么对于假设空间中任意一个假设 $g$ ， $E_{out} \approx E_{in}$ 。
- 利用算法A从假设空间H中，挑选一个 $g$ ，使 $E_{in}(g) \approx 0$ ，则 $E_{out} \approx 0$ 。

这两个条件，正好对应着test和train两个过程。train的目的是使损失期望 $E_{in}(g) \approx 0$ ；test的目的是使将算法用到新的样本时的损失期望也尽可能小，即 $E_{out} \approx 0$ 。

正因为如此，上次课引入了break point，并推导出只要break point存在，则 $M$ 有上界，一定存在 $E_{out} \approx E_{in}$ 。

本次笔记主要介绍VC Dimension的概念。同时也是总结VC Dimension与 $E_{in}(g) \approx 0$ ， $E_{out} \approx 0$ ，Model Complexity Penalty（下面会讲到）的关系。

## 一、Definition of VC Dimension

首先，我们知道如果一个假设空间H有break point  $k$ ，那么它的成长函数是有界的，它的上界称为Bound function。根据数学归纳法，Bound function也是有界的，且上界为 $N^{k-1}$ 。从下面的表格可以看出， $N(k-1)$ 比 $B(N,k)$ 松弛很多。

$$m_{\mathcal{H}}(N) \text{ of break point } k \leq B(N, k) = \underbrace{\sum_{i=0}^{k-1} \binom{N}{i}}_{\text{highest term } N^{k-1}}$$

$B(N, k)$	$k$				
	1	2	3	4	5
1	1	2	2	2	2
2	1	3	4	4	4
3	1	4	7	8	8
4	1	5	11	15	16
5	1	6	16	26	31
6	1	7	22	42	57

$N^{k-1}$	$k$				
	1	2	3	4	5
1	1	1	1	1	1
2	1	2	4	8	16
3	1	3	9	27	81
4	1	4	16	64	256
5	1	5	25	125	625
6	1	6	36	216	1296

**provably** & loosely, for  $N \geq 2, k \geq 3$ ,

$$m_{\mathcal{H}}(N) \leq B(N, k) = \sum_{i=0}^{k-1} \binom{N}{i} \leq N^{k-1}$$

则根据上一节课的推导，VC bound就可以转换为：

For any  $g = \mathcal{A}(\mathcal{D}) \in \mathcal{H}$  and ‘statistical’ large  $\mathcal{D}$ , for  ~~$N \geq 2$~~ ,  $k \geq 3$

$$\begin{aligned}
 & \mathbb{P}_{\mathcal{D}} \left[ |E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon \right] \\
 & \leq \mathbb{P}_{\mathcal{D}} \left[ \exists h \in \mathcal{H} \text{ s.t. } |E_{\text{in}}(h) - E_{\text{out}}(h)| > \epsilon \right] \\
 & \leq 4m_{\mathcal{H}}(2N) \exp \left( -\frac{1}{8} \epsilon^2 N \right) \\
 & \stackrel{\text{if } k \text{ exists}}{\leq} 4(2N)^{k-1} \exp \left( -\frac{1}{8} \epsilon^2 N \right)
 \end{aligned}$$

这样，不等式只与k和N相关了，一般情况下样本N足够大，所以我们只考虑k值。有如下结论：

- 若假设空间H有break point k，且N足够大，则根据VC bound理论，算法有良好的泛化能力
- 在假设空间中选择一个矩g，使  $E_{\text{in}} \approx 0$ ，则其在全集数据中的错误率会较低

if ①  $m_{\mathcal{H}}(N)$  breaks at  $k$  (good  $\mathcal{H}$ )  
 ②  $N$  large enough (good  $\mathcal{D}$ )  
 $\Rightarrow$  probably generalized ' $E_{\text{out}} \approx E_{\text{in}}$ ', and  
 if ③  $\mathcal{A}$  picks a  $g$  with small  $E_{\text{in}}$  (good  $\mathcal{A}$ )  
 $\Rightarrow$  probably learned! ( $\therefore$ ) good luck)

下面介绍一个新的名词：VC Dimension。VC Dimension就是某假设集 $\mathcal{H}$ 能够shatter的最多inputs的个数，即最大完全正确的分类能力。（注意，只要存在一种分布的inputs能够正确分类也满足）。

shatter的英文意思是“粉碎”，也就是说对于inputs的所有情况都能列举出来。例如对 $N$ 个输入，如果能够将 $2^N$ 种情况都列出来，则称该 $N$ 个输入能够被假设集 $\mathcal{H}$  shatter。

根据之前break point的定义：假设集不能被shatter任何分布类型的inputs的最少个数。则VC Dimension等于break point的个数减一。





## Definition

VC dimension of  $\mathcal{H}$ , denoted  $d_{\text{VC}}(\mathcal{H})$  is

**largest**  $N$  for which  $m_{\mathcal{H}}(N) = 2^N$

- the **most** inputs  $\mathcal{H}$  that can shatter
- $d_{\text{VC}} = \text{'minimum } k' - 1$

现在，我们回顾一下之前介绍的四种例子，它们对应的VC Dimension是多少：

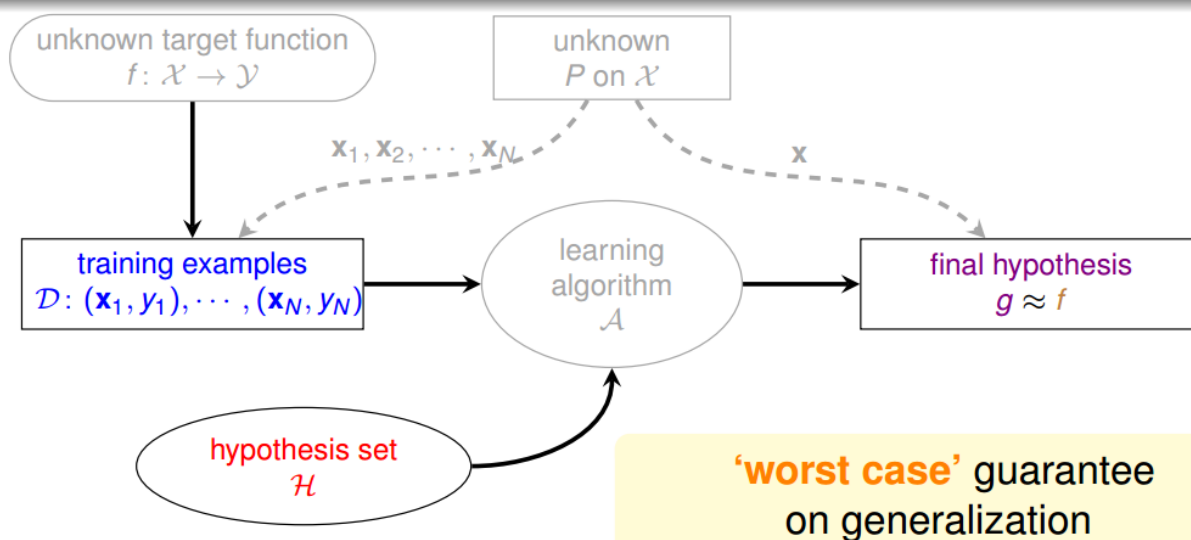
<ul style="list-style-type: none"> <li>positive rays: <math>d_{\text{VC}} = 1</math></li> </ul>		$m_{\mathcal{H}}(N) = N + 1$
<ul style="list-style-type: none"> <li>positive intervals: <math>d_{\text{VC}} = 2</math></li> </ul>		$m_{\mathcal{H}}(N) = \frac{1}{2}N^2 + \frac{1}{2}N + 1$
<ul style="list-style-type: none"> <li>convex sets: <math>d_{\text{VC}} = \infty</math></li> </ul>		$m_{\mathcal{H}}(N) = 2^N$
<ul style="list-style-type: none"> <li>2D perceptrons: <math>d_{\text{VC}} = 3</math></li> </ul>		$m_{\mathcal{H}}(N) \leq N^3$ for $N \geq 2$

用 $d_{\text{VC}}$ 代替 $k$ ，那么VC bound的问题也就转换为与 $d_{\text{VC}}$ 和 $N$ 相关了。同时，如果一个假

设集 $\mathcal{H}$ 的 $d_{vc}$ 确定了，则就能满足机器能够学习的第一个条件 $E_{out} \approx E_{in}$ ，与算法、样本数据分布和目标函数都没有关系。

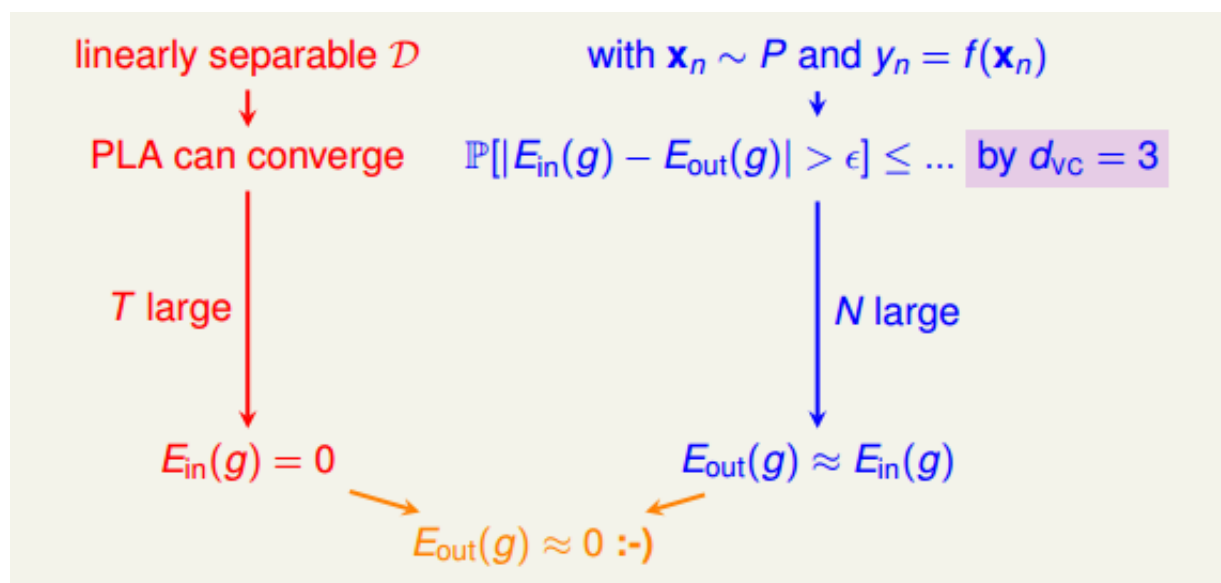
**finite  $d_{vc} \implies g$  'will' generalize ( $E_{out}(g) \approx E_{in}(g)$ )**

- regardless of learning algorithm  $\mathcal{A}$
- regardless of input distribution  $P$
- regardless of target function  $f$



## 二、VC Dimension of Perceptrons

回顾一下我们之前介绍的2D下的PLA算法，已知Perceptrons的 $k=4$ ，即 $d_{vc} = 3$ 。根据VC Bound理论，当 $N$ 足够大的时候， $E_{out}(g) \approx E_{in}(g)$ 。如果找到一个 $g$ ，使 $E_{in}(g) \approx 0$ ，那么就能证明PLA是可以学习的。




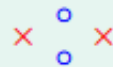
这是在2D情况下，那如果是多维的Perceptron，它对应的 $d_{vc}$ 又等于多少呢？

已知在1D Perceptron， $d_{vc} = 2$ ，在2D Perceptrons， $d_{vc} = 3$ ，那么我们有如下假

设： $d_{vc} = d + 1$ ，其中d为维数。

要证明的话，只需分两步证明：

- $d_{vc} \geq d + 1$
- $d_{vc} \leq d + 1$

- 1D perceptron (pos/neg rays):  $d_{vc} = 2$
- 2D perceptrons:  $d_{vc} = 3$ 
  - $d_{vc} \geq 3$ : 
  - $d_{vc} \leq 3$ : 
- d-D perceptrons:  $d_{vc} \stackrel{?}{=} d + 1$

首先证明第一个不等式： $d_{vc} \geq d + 1$ 。

在d维里，我们只要找到某一类的d+1个inputs可以被shatter的话，那么必然得到  $d_{vc} \geq d + 1$ 。所以，我们有意构造一个d维的矩阵X能够被shatter就行。X是d维的，有d+1个inputs，每个inputs加上第零个维度的常数项1，得到X的矩阵：

$$X = \begin{bmatrix} -\mathbf{x}_1^T - \\ -\mathbf{x}_2^T - \\ -\mathbf{x}_3^T - \\ \vdots \\ -\mathbf{x}_{d+1}^T - \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 0 & 1 & & 0 \\ \vdots & \vdots & & \ddots & 0 \\ 1 & 0 & \dots & 0 & 1 \end{bmatrix}$$

矩阵中，每一行代表一个inputs，每个inputs是d+1维的，共有d+1个inputs。这里构造的X很明显是可逆的。shatter的本质是假设空间H对X的所有情况的判断都是对的，即总能找到权重W，满足  $X * W = y$ ， $W = X^{-1} * y$ 。由于这里我们构造的矩阵X的逆矩阵存在，那么d维的所有inputs都能被shatter，也就证明了第一个不等式。

$$X = \begin{bmatrix} -\mathbf{x}_1^T \\ -\mathbf{x}_2^T \\ \vdots \\ -\mathbf{x}_{d+1}^T \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & & \ddots & 0 \\ 1 & 0 & \dots & 0 & 1 \end{bmatrix} \text{ invertible}$$

to shatter ...

for any  $\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_{d+1} \end{bmatrix}$ , find  $\mathbf{w}$  such that

$$\text{sign}(\mathbf{X}\mathbf{w}) = \mathbf{y} \iff (\mathbf{X}\mathbf{w}) = \mathbf{y} \xrightarrow{\text{X invertible!}} \mathbf{w} = \mathbf{X}^{-1}\mathbf{y}$$

然后证明第二个不等式： $d_{vc} \leq d + 1$ 。

在d维里，如果对于任何的d+2个inputs，一定不能被shatter，则不等式成立。我们构造一个任意的矩阵 $X$ ，其包含d+2个inputs，该矩阵有d+1列，d+2行。这d+2个向量的某一行一定可以被另外d+1个向量线性表示，例如对于向量 $X_{d+2}$ ，可表示为：

$$X_{d+2} = a_1 * X_1 + a_2 * X_2 + \dots + a_d * X_d$$

其中，假设 $a_1 > 0$ ， $a_2, \dots, a_d < 0$ 。

那么如果 $X_1$ 是正类， $X_2, \dots, X_d$ 均为负类，则存在 $W$ ，得到如下表达式：

$$X_{d+2} * W = a_1 * X_1 * W + a_2 * X_2 * W + \dots + a_d * X_d * W > 0$$

因为其中蓝色项大于0，代表正类；红色项小于0，代表负类。所有对于这种情况， $X_{d+2}$ 一定是正类，无法得到负类的情况。也就是说，d+2个inputs无法被shatter。证明完毕！

## d-D General Case

$$X = \begin{bmatrix} \text{---} \mathbf{x}_1^T \text{---} \\ \text{---} \mathbf{x}_2^T \text{---} \\ \vdots \\ \text{---} \mathbf{x}_{d+1}^T \text{---} \\ \text{---} \mathbf{x}_{d+2}^T \text{---} \end{bmatrix}$$

more rows than columns:

linear dependence (some  $a_i$  non-zero)

$$\mathbf{x}_{d+2} = a_1 \mathbf{x}_1 + a_2 \mathbf{x}_2 + \dots + a_{d+1} \mathbf{x}_{d+1}$$

- can you generate  $(\text{sign}(a_1), \text{sign}(a_2), \dots, \text{sign}(a_{d+1}), \times)$ ? if so, what  $\mathbf{w}$ ?

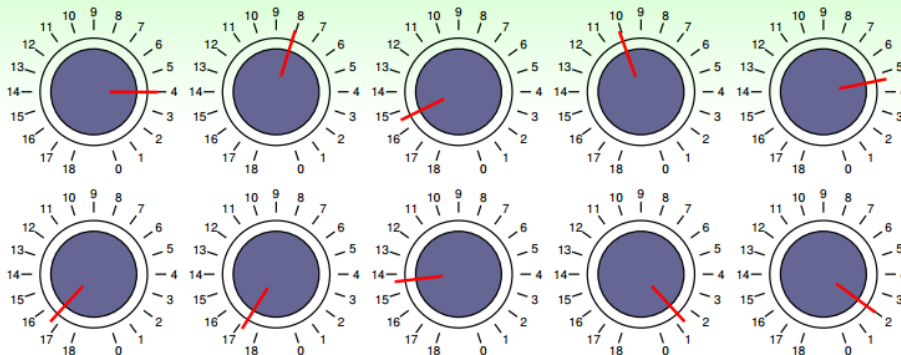
$$\mathbf{w}^T \mathbf{x}_{d+2} = a_1 \underbrace{\mathbf{w}^T \mathbf{x}_1}_0 + a_2 \underbrace{\mathbf{w}^T \mathbf{x}_2}_{\times} + \dots + a_{d+1} \underbrace{\mathbf{w}^T \mathbf{x}_{d+1}}_{\times}$$

$> 0(\text{contradiction!})$

综上所述可得  $d_{vc} = d + 1$ 。

## 三、Physical Intuition VC Dimension

### Degrees of Freedom



(modified from the work of Hugues Vermeiren on <http://www.texample.net>)

- hypothesis parameters  $\mathbf{w} = (w_0, w_1, \dots, w_d)$ :  
**creates degrees of freedom**
- hypothesis quantity  $M = |\mathcal{H}|$ :  
'analog' degrees of freedom
- hypothesis 'power'  $d_{vc} = d + 1$ :  
**effective 'binary' degrees of freedom**

$d_{vc}(\mathcal{H})$ : powerfulness of  $\mathcal{H}$

上节公式中  $\mathbf{W}$  又名 features，即自由度。自由度是可以任意调节的，如同上图中的旋钮一样，可以调节。VC Dimension 代表了假设空间的分类能力，即反映了  $\mathcal{H}$  的自由

度，产生dichotomy的数量，也就等于features的个数，但也不是绝对的。

practical rule of thumb:

$$d_{vc} \approx \text{\#free parameters (but not always)}$$

例如，对2D Perceptrons，线性分类， $d_{vc} = 3$ ，则 $W = \{w_0, w_1, w_2\}$ ，也就是说只要3个features就可以进行学习，自由度为3。

介绍到这，我们发现M与 $d_{vc}$ 是成正比的，从而得到如下结论：

### $M$ and $d_{vc}$

copied from Lecture 5 :-)

- 1 can we make sure that  $E_{out}(g)$  is close enough to  $E_{in}(g)$ ?
- 2 can we make  $E_{in}(g)$  small enough?

#### small $M$

- 1 Yes!,  
 $\mathbb{P}[\text{BAD}] \leq 2 \cdot M \cdot \exp(\dots)$
- 2 No!, too few choices

#### large $M$

- 1 No!,  
 $\mathbb{P}[\text{BAD}] \leq 2 \cdot M \cdot \exp(\dots)$
- 2 Yes!, many choices

#### small $d_{vc}$

- 1 Yes!,  $\mathbb{P}[\text{BAD}] \leq 4 \cdot (2N)^{d_{vc}} \cdot \exp(\dots)$
- 2 No!, too limited power

#### large $d_{vc}$

- 1 No!,  $\mathbb{P}[\text{BAD}] \leq 4 \cdot (2N)^{d_{vc}} \cdot \exp(\dots)$
- 2 Yes!, lots of power

## 四、Interpreting VC Dimension

下面，我们将更深入地探讨VC Dimension的意义。首先，把VC Bound重新写到这里：

For any  $g = \mathcal{A}(\mathcal{D}) \in \mathcal{H}$  and ‘statistical’ large  $\mathcal{D}$ , for  ~~$N \geq 2$~~ ,  $d_{vc} \geq 2$

$$\mathbb{P}_{\mathcal{D}} \left[ \underbrace{|E_{in}(g) - E_{out}(g)|}_{\text{BAD}} > \epsilon \right] \leq \underbrace{4(2N)^{d_{vc}} \exp\left(-\frac{1}{8}\epsilon^2 N\right)}_{\delta}$$

根据之前的泛化不等式，如果 $|E_{in} - E_{out}| > \epsilon$ ，即出现bad坏的情况的概率最大不



超过 $\delta$ 。那么反过来，对于good好的情况发生的概率最小为 $1 - \delta$ ，则对上述不等式进行重新推导：

## Rephrase

..., with probability  $\geq 1 - \delta$ , **GOOD**:  $|E_{\text{in}}(g) - E_{\text{out}}(g)| \leq \epsilon$

$$\begin{aligned} \text{set } \delta &= 4(2N)^{d_{\text{vc}}} \exp\left(-\frac{1}{8}\epsilon^2 N\right) \\ \frac{\delta}{4(2N)^{d_{\text{vc}}}} &= \exp\left(-\frac{1}{8}\epsilon^2 N\right) \\ \ln\left(\frac{4(2N)^{d_{\text{vc}}}}{\delta}\right) &= \frac{1}{8}\epsilon^2 N \\ \sqrt{\frac{8}{N} \ln\left(\frac{4(2N)^{d_{\text{vc}}}}{\delta}\right)} &= \epsilon \end{aligned}$$

$\epsilon$ 表现了假设空间 $H$ 的泛化能力， $\epsilon$ 越小，泛化能力越大。

For any  $g = \mathcal{A}(\mathcal{D}) \in \mathcal{H}$  and 'statistical' large  $\mathcal{D}$ , for  ~~$N \geq 2$~~ ,  $d_{\text{vc}} \geq 2$

$$\mathbb{P}_{\mathcal{D}} \left[ \underbrace{|E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon}_{\text{BAD}} \right] \leq \underbrace{4(2N)^{d_{\text{vc}}} \exp\left(-\frac{1}{8}\epsilon^2 N\right)}_{\delta}$$

## Rephrase

..., with probability  $\geq 1 - \delta$ , **GOOD!**

$$\begin{aligned} \text{gen. error } |E_{\text{in}}(g) - E_{\text{out}}(g)| &\leq \sqrt{\frac{8}{N} \ln\left(\frac{4(2N)^{d_{\text{vc}}}}{\delta}\right)} \\ E_{\text{in}}(g) - \sqrt{\frac{8}{N} \ln\left(\frac{4(2N)^{d_{\text{vc}}}}{\delta}\right)} &\leq E_{\text{out}}(g) \leq E_{\text{in}}(g) + \sqrt{\frac{8}{N} \ln\left(\frac{4(2N)^{d_{\text{vc}}}}{\delta}\right)} \end{aligned}$$

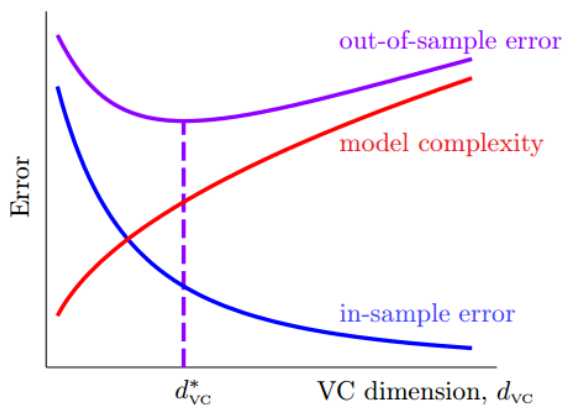
$$\underbrace{\sqrt{\dots}}_{\Omega(N, \mathcal{H}, \delta)} : \text{penalty for model complexity}$$

至此，已经推导出泛化误差 $E_{\text{out}}$ 的边界，因为我们更关心其上界（ $E_{\text{out}}$ 可能的最大值），即：

with a high probability,

$$E_{out}(g) \leq E_{in}(g) + \underbrace{\sqrt{\frac{8}{N} \ln \left( \frac{4(2N)^{d_{vc}}}{\delta} \right)}}_{\Omega(N, \mathcal{H}, \delta)}$$

上述不等式的右边第二项称为模型复杂度，其模型复杂度与样本数量 $N$ 、假设空间 $H(d_{vc})$ 、 $\epsilon$ 有关。 $E_{out}$ 由 $E_{in}$ 共同决定。下面绘出 $E_{out}$ 、model complexity、 $E_{in}$ 随 $d_{vc}$ 变化的关系：



- $d_{vc} \uparrow$ :  $E_{in} \downarrow$  but  $\Omega \uparrow$
- $d_{vc} \downarrow$ :  $\Omega \downarrow$  but  $E_{in} \uparrow$
- best  $d_{vc}^*$  in the middle

powerful  $\mathcal{H}$  not always good!

通过该图可以得出如下结论：

- $d_{vc}$ 越大， $E_{in}$ 越小， $\Omega$ 越大（复杂）。
- $d_{vc}$ 越小， $E_{in}$ 越大， $\Omega$ 越小（简单）。
- 随着 $d_{vc}$ 增大， $E_{out}$ 会先减小再增大。

所以，为了得到最小的 $E_{out}$ ，不能一味地增大 $d_{vc}$ 以减小 $E_{in}$ ，因为 $E_{in}$ 太小的时候，模型复杂度会增加，造成 $E_{out}$ 变大。也就是说，选择合适的 $d_{vc}$ ，选择的features个数要合适。

下面介绍一个概念：样本复杂度（Sample Complexity）。如果选定 $d_{vc}$ ，样本数据 $D$ 选择多少合适呢？通过下面一个例子可以帮助我们理解：

given specs  $\epsilon = 0.1$ ,  $\delta = 0.1$ ,  $d_{vc} = 3$ , want  $4(2N)^{d_{vc}} \exp(-\frac{1}{8}\epsilon^2 N) \leq \delta$

$N$	bound
100	$2.82 \times 10^7$
1,000	$9.17 \times 10^9$
10,000	$1.19 \times 10^8$
100,000	$1.65 \times 10^{-38}$
29,300	$9.99 \times 10^{-2}$

sample complexity:  
need  $N \approx 10,000 d_{vc}$  in theory

通过计算得到 $N=29300$ ，刚好满足 $\delta = 0.1$ 的条件。 $N$ 大约是 $d_{vc}$ 的10000倍。这个数值太大了，实际中往往不需要这么多的样本数量，大概只需要 $d_{vc}$ 的10倍就够了。 $N$ 的理论值之所以这么大是因为VC Bound 过于宽松了，我们得到的是一个比实际大得多的上界。

## Looseness of VC Bound

$$\mathbb{P}_{\mathcal{D}} \left[ |E_{in}(g) - E_{out}(g)| > \epsilon \right] \leq 4(2N)^{d_{vc}} \exp \left( -\frac{1}{8} \epsilon^2 N \right)$$

theory:  $N \approx 10,000 d_{vc}$ ; practice:  $N \approx 10 d_{vc}$

### Why?

- Hoeffding for unknown  $E_{out}$  **any distribution, any target**
- $m_{\mathcal{H}}(N)$  instead of  $|\mathcal{H}(\mathbf{x}_1, \dots, \mathbf{x}_N)|$  **'any' data**
- $N^{d_{vc}}$  instead of  $m_{\mathcal{H}}(N)$  **'any'  $\mathcal{H}$  of same  $d_{vc}$**
- union bound on worst cases **any choice made by  $\mathcal{A}$**

—**but hardly better, and 'similarly loose for all models'**

值得一提的是，VC Bound是比较宽松的，而如何收紧它却不是那么容易，这也是机器学习的一大难题。但是，令人欣慰的一点是，VC Bound基本上对所有模型的宽松程度是基本一致的，所以，不同模型之间还是可以横向比较。从而，VC Bound宽松对机器学习的可行性还是没有太大影响。

## 五、总结

本节课主要介绍了VC Dimension的概念就是最大的non-break point。然后，我们得到了Perceptrons在 $d$ 维度下的VC Dimension是 $d+1$ 。接着，我们在物理意义上，将 $d_{vc}$ 与自由度联系起来。最终得出结论 $d_{vc}$ 不能过大也不能过小。选取合适的值，才能让 $E_{out}$ 足够小，使假设空间 $H$ 具有良好的泛化能力。

**注明：**

文章中所有的图片均来自台湾大学林轩田《机器学习基石》课程