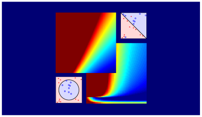
Machine Learning Foundations
(機器學習基石)



Lecture 7: The VC Dimension

Hsuan-Tien Lin (林軒田)
htlin@csie.ntu.edu.tw

Department of Computer Science
& Information Engineering

National Taiwan University
(國立台灣大學資訊工程系)

Roadmap

1. When Can Machines Learn?
2. **Why** Can Machines Learn?

Lecture 6: Theory of Generalization

$E_{\text{out}} \approx E_{\text{in}}$ possible
if $m_{\mathcal{H}}(N)$ **breaks somewhere** and N **large enough**

Lecture 7: The VC Dimension

- Definition of VC Dimension
- VC Dimension of Perceptrons
- Physical Intuition of VC Dimension
- Interpreting VC Dimension

3. How Can Machines Learn?
4. How Can Machines Learn Better?

Recap: More on Growth Function

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

	$B(N,k)$	$k=1$	2	3	4	5
	1	1	2	2	2	2
	2	1	3	4	4	4
	3	1	4	7	8	8
N	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 $N \geq 2, k \geq 3$,

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

Recap: More on Vapnik-Chervonenkis (VC) Bound

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

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

When N is large enough

$\because$

$\therefore$

if (1) $m_{\mathcal{H}}(N)$ breaks at k (**good** $\mathcal{H}$)

(2) N large enough (**good** $\mathcal{D}$)

$\Longrightarrow$ probably generalized '$E_{\text{out}} \approx E_{\text{in}}$', and

if (3) $\mathcal{A}$ picks a g with small E_{in} (**good** $\mathcal{A}$)

$\Longrightarrow$ probably learned! (**:-) good luck**)

VC Dimension

the formal name of **maximum non-break point**

≡ can be shattered
≡ can enumerate all poosibles

Definition

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

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

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

$$N \le d_{\text{VC}} \implies \mathcal{H} \text{ can shatter some } N \text{ inputs}$$
$$\underset{\textbf{N}}{k} > d_{\text{VC}} \implies \underset{\textbf{N}}{k} \text{ is a break point for } \mathcal{H}$$

if $N \ge 2$, $d_{\text{VC}} \ge 2$, $m_{\mathcal{H}}(N) \le N^{d_{\text{VC}}}$

The Four VC Dimensions

- positive rays: $m_{\mathcal{H}}(N) = N + 1$

  $d_{\mathrm{VC}} = 1$

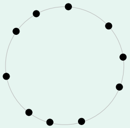
- positive intervals: $m_{\mathcal{H}}(N) = \frac{1}{2}N^2 + \frac{1}{2}N + 1$

  $d_{\mathrm{VC}} = 2$

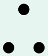
- convex sets: $m_{\mathcal{H}}(N) = 2^N$

  $d_{\mathrm{VC}} = \infty$

- 2D perceptrons: $m_{\mathcal{H}}(N) \le N^3$ for $N \ge 2$

  $d_{\mathrm{VC}} = 3$

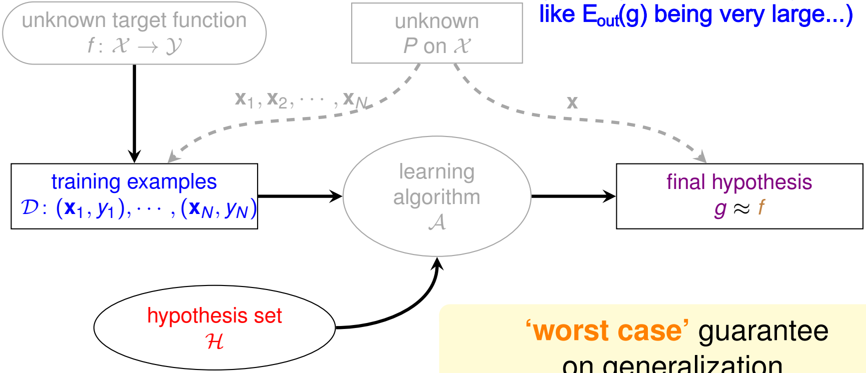
good: **finite** d_{VC} **(≡ break point exists)**

$\mathcal{H}$

VC Dimension and Learning

finite $d_{\mathrm{VC}} \Longrightarrow g$ 'will' generalize ($E_{\mathrm{out}}(g) \approx E_{\mathrm{in}}(g)$)

- regardless of learning algorithm $\mathcal{A}$ ⟶ If learning algorithm $\mathcal{A}$ is really bad and we end up getting a very large $E_{in}(g)$, we could still ensure that $E_{out}(g) \approx E_{in}(g)$. (Although we don't really the result like $E_{out}(g)$ being very large...)
- regardless of input distribution P
- regardless of target function f



'worst case' guarantee on generalization

Fun Time

If there is a set of N inputs that cannot be shattered by $\mathcal{H}$. Based only on this information, what can we conclude about $d_{\mathrm{VC}}(\mathcal{H})$?

1. $d_{\mathrm{VC}}(\mathcal{H}) > N$
2. $d_{\mathrm{VC}}(\mathcal{H}) = N$
3. $d_{\mathrm{VC}}(\mathcal{H}) < N$
4. no conclusion can be made

Fun Time

If there is a set of N inputs that cannot be shattered by $\mathcal{H}$. Based only on this information, what can we conclude about $d_{\mathrm{VC}}(\mathcal{H})$?

1. $d_{\mathrm{VC}}(\mathcal{H}) > N$
2. $d_{\mathrm{VC}}(\mathcal{H}) = N$
3. $d_{\mathrm{VC}}(\mathcal{H}) < N$
4. no conclusion can be made

We cannot judge by just a set of inputs.

Reference Answer: 4

It is possible that there is another set of N inputs that can be shattered, which means $d_{\mathrm{VC}} \geq N$. It is also possible that no set of N input can be shattered, which means $d_{\mathrm{VC}} < N$. Neither cases can be ruled out by one non-shattering set.

2D PLA Revisited

linearly separable $\mathcal{D}$ $\downarrow$ PLA can converge $\xrightarrow{T \text{ large}}$ $E_{\text{in}}(g) = 0$

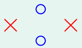
iterations

with $\mathbf{x}_n \sim P$ and $y_n = f(\mathbf{x}_n)$ $\downarrow$ $\mathbb{P}[|E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon] \leq \ldots$ by $d_{\text{VC}} = 3$ $\xrightarrow{N \text{ large}}$ $E_{\text{out}}(g) \approx E_{\text{in}}(g)$

$E_{\text{out}}(g) \approx 0$:-)

general PLA for $\mathbf{x}$ with more than 2 features?

VC Dimension of Perceptrons

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

two steps:

- $d_{\text{VC}} \geq d + 1$
- $d_{\text{VC}} \leq d + 1$

Extra Fun Time

What statement below shows that $d_{\text{vc}} \geq d + 1$?

1. There are some $d + 1$ inputs we can shatter.
2. We can shatter any set of $d + 1$ inputs.
3. There are some $d + 2$ inputs we cannot shatter.
4. We cannot shatter any set of $d + 2$ inputs.

Extra Fun Time

What statement below shows that $d_{\text{VC}} \geq d + 1$?

1. There are some $d + 1$ inputs we can shatter.
2. We can shatter any set of $d + 1$ inputs.
3. There are some $d + 2$ inputs we cannot shatter.
4. We cannot shatter any set of $d + 2$ inputs.

Reference Answer: 1

d_{VC} is the maximum that $m_{\mathcal{H}}(N) = 2^N$, and $m_{\mathcal{H}}(N)$ is the most number of dichotomies of N inputs. So if we can find 2^{d+1} dichotomies on *some* $d + 1$ inputs, $m_{\mathcal{H}}(d + 1) = 2^{d+1}$ and hence $d_{\text{VC}} \geq d + 1$.

$d_{\mathrm{VC}} \geq d + 1$

There are **some $d+1$ inputs** we can shatter.

- some 'trivial' inputs:

$$
\mathrm{X} = \begin{bmatrix} -\ \mathbf{x}_1^T\ - \\ -\ \mathbf{x}_2^T\ - \\ -\ \mathbf{x}_3^T\ - \\ \vdots \\ -\ \mathbf{x}_{d+1}^T\ - \end{bmatrix} = \begin{array}{c} \begin{array}{cccccc} \mathbf{x_0} & & & & & \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 0 & 1 & & 0 \\ \vdots & \vdots & & \ddots & 0 \\ 1 & 0 & \dots & 0 & 1 \end{bmatrix} \end{array} \begin{array}{l} \leftarrow \text{origin} \\ \\ \\ \\ \\ \end{array}
$$

- visually in 2D: 

note: X **invertible!**

Can We Shatter X?

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

to shatter ...

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

$$\text{sign}(\mathrm{X}\mathbf{w}) = \mathbf{y} \quad \Longleftarrow \quad (\mathrm{X}\mathbf{w}) = \mathbf{y} \overset{\mathrm{X}\text{ invertible!}}{\Longleftrightarrow} \mathbf{w} = \mathrm{X}^{-1}\mathbf{y}$$

'special' X can be shattered $\Longrightarrow d_{\text{VC}} \geq d + 1$

Extra Fun Time

What statement below shows that $d_{\text{vc}} \leq d + 1$?

1. There are some $d + 1$ inputs we can shatter.
2. We can shatter any set of $d + 1$ inputs.
3. There are some $d + 2$ inputs we cannot shatter.
4. We cannot shatter any set of $d + 2$ inputs.

Extra Fun Time

What statement below shows that $d_{\text{VC}} \le d + 1$?

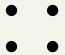
1. There are some $d + 1$ inputs we can shatter.
2. We can shatter any set of $d + 1$ inputs.
3. There are some $d + 2$ inputs we cannot shatter.
4. We cannot shatter any set of $d + 2$ inputs.

Reference Answer: 4

d_{VC} is the maximum that $m_{\mathcal{H}}(N) = 2^N$, and $m_{\mathcal{H}}(N)$ is the most number of dichotomies of N inputs. So if we cannot find 2^{d+2} dichotomies on *any* $d + 2$ inputs (i.e. break point), $m_{\mathcal{H}}(d + 2) < 2^{d+2}$ and hence $d_{\text{VC}} < d + 2$. That is, $d_{\text{VC}} \le d + 1$.

$d_{\mathrm{VC}} \le d + 1$ (1/2)

A 2D Special Case



$$\mathrm{X} = \begin{bmatrix} — \mathbf{x}_1^T — \\ — \mathbf{x}_2^T — \\ — \mathbf{x}_3^T — \\ — \mathbf{x}_4^T — \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$\circ$ **?**

$\times$ $\circ$

? cannot be $\times$

linear dependence:
$x_4 = x_2 + x_3 - x_1$

multiply by w

$\because$ "+" + "+" − "−" = "+"

$$\mathbf{w}^T\mathbf{x}_4 = \underbrace{\mathbf{w}^T\mathbf{x}_2}_{\circ} + \underbrace{\mathbf{w}^T\mathbf{x}_3}_{\circ} - \underbrace{\mathbf{w}^T\mathbf{x}_1}_{\times} > 0$$

$\Rightarrow$ x_4 must be $\circ$

linear dependence **restricts dichotomy**

$d_{\mathrm{VC}} \le d + 1$ (2/2)

d-D General Case

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

In linear algebra: rank = d+1

~~more rows than columns~~:

Can not be all zero.

linear dependence (some a_i non-zero)

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

then

- can you generate $(\mathrm{sign}(a_1), \mathrm{sign}(a_2), \ldots, \mathrm{sign}(a_{d+1}), \times)$? No if so, what $\mathbf{w}$?

The first d+1 dictate the (d+2)th one.

if

$$\boxed{\begin{aligned} \mathbf{w}^T \mathbf{x}_{d+2} &= a_1 \underbrace{\mathbf{w}^T \mathbf{x}_1}_{\circ} + a_2 \underbrace{\mathbf{w}^T \mathbf{x}_2}_{\times} + \ldots + a_{d+1} \underbrace{\mathbf{w}^T \mathbf{x}_{d+1}}_{\times} \\ &> 0 (\text{contradition!}) \end{aligned}}$$

‘general’ X no-shatter $\Longrightarrow d_{\mathrm{VC}} \le d + 1$

Fun Time

Based on the proof above, what is d_{VC} of 1126-D perceptrons?

1. 1024
2. 1126
3. 1127
4. 6211

Fun Time

Based on the proof above, what is d_{VC} of 1126-D perceptrons?

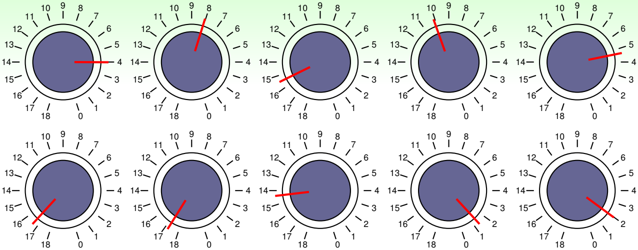
1. 1024
2. 1126
3. 1127
4. 6211

Reference Answer: 3

Well, **too much fun for this section! :-)**

Degrees of Freedom

Knob



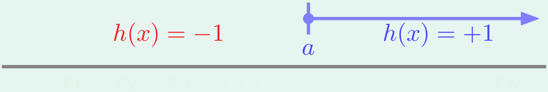
(modified from the work of Hugues Vermeiren on `http://www.texample.net`)

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

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

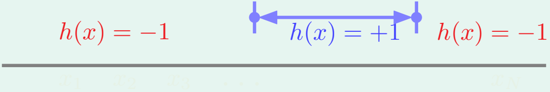
Two Old Friends

Positive Rays ($d_{\text{VC}} = 1$)



free parameters: a

Positive Intervals ($d_{\text{VC}} = 2$)



free parameters: ℓ, r

practical rule of thumb:

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

M and d_{VC}

copied from Lecture 5 :-)

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

small M

1. Yes!, $\mathbb{P}[\textbf{BAD}] \le 2 \cdot M \cdot \exp(\ldots)$
2. No!, too few choices

large M

1. No!, $\mathbb{P}[\textbf{BAD}] \le 2 \cdot M \cdot \exp(\ldots)$
2. Yes!, many choices

small d_{VC}

1. Yes!, $\mathbb{P}[\textbf{BAD}] \le 4 \cdot (2N)^{d_{\text{VC}}} \cdot \exp(\ldots)$
2. No!, too limited power

large d_{VC}

1. No!, $\mathbb{P}[\textbf{BAD}] \le 4 \cdot (2N)^{d_{\text{VC}}} \cdot \exp(\ldots)$
2. Yes!, lots of power

using the right d_{VC} (or $\mathcal{H}$) is important

Fun Time

Origin-crossing Hyperplanes are essentially perceptrons with w_0 fixed at 0. Make a guess about the d_{VC} of origin-crossing hyperplanes in $\mathbb{R}^d$.

1. 1
2. d
3. $d+1$
4. ∞

Fun Time

Origin-crossing Hyperplanes are essentially perceptrons with w_0 fixed at 0. Make a guess about the d_{VC} of origin-crossing hyperplanes in $\mathbb{R}^d$.

1. 1
2. d
3. $d+1$
4. ∞

Reference Answer: 2

The proof is almost the same as proving the d_{VC} for usual perceptrons, but it is the **intuition** ($d_{\text{VC}} \approx$ #free parameters) that you shall use to answer this quiz.

VC Bound Rephrase: Penalty for Model Complexity

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

$$\mathbb{P}_{\mathcal{D}}\Big[\underbrace{\big|E_{\mathrm{in}}(g) - E_{\mathrm{out}}(g)\big| > \epsilon}_{\textbf{BAD}}\Big] \quad \leq \quad \underbrace{4(2N)^{d_{\mathrm{VC}}} \exp\left(-\tfrac{1}{8}\epsilon^2 N\right)}_{\delta}$$

Rephrase

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

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

VC Bound Rephrase: Penalty for Model Complexity

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

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

Rephrase

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

generalization

$$\text{gen. error } \left|E_{\text{in}}(g) - E_{\text{out}}(g)\right| \quad \leq \quad \sqrt{\frac{8}{N}\ln\left(\frac{4(2N)^{d_{\mathrm{VC}}}}{\delta}\right)}$$

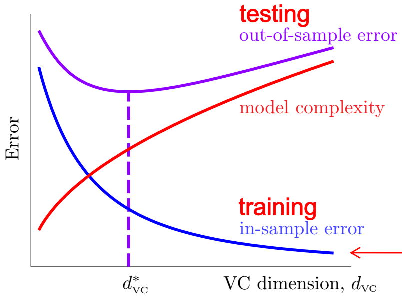
$$E_{\text{in}}(g) - \sqrt{\frac{8}{N}\ln\left(\frac{4(2N)^{d_{\mathrm{VC}}}}{\delta}\right)} \quad \leq \quad E_{\text{out}}(g) \quad \leq \quad E_{\text{in}}(g) + \boxed{\sqrt{\frac{8}{N}\ln\left(\frac{4(2N)^{d_{\mathrm{VC}}}}{\delta}\right)}}$$

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

THE VC Message

with a high probability,

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



- $d_{\text{VC}} \uparrow$: $E_{\text{in}} \downarrow$ but $\Omega \uparrow$
- $d_{\text{VC}} \downarrow$: $\Omega \downarrow$ but $E_{\text{in}} \uparrow$
- best d^*_{VC} in the middle

The premise is that the algorithm can find the best hypothesis.

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

VC Bound Rephrase: Sample Complexity

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

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

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

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

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

practical rule of thumb:

$N \approx 10 d_{\mathrm{VC}}$ **often enough!**

Looseness of VC Bound

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

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

Why?

- Hoeffding for unknown E_{out} — **any distribution, any target**
- $m_{\mathcal{H}}(N)$ instead of $|\mathcal{H}(\mathbf{x}_1, \ldots, \mathbf{x}_N)|$ — **'any' data**
- $N^{d_{\text{VC}}}$ 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'**

Why VC bound?
1. We can't find another theory that has tighter looseness than VC bound.
2. VC bound has similar loose for all models.
3. We only need the message behind VC bound. (We don't need too complicated models.)

philosophical message of VC bound
important for improving ML

Fun Time

Consider the VC Bound below. How can we decrease the probability of getting **BAD** data?

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

1. decrease model complexity d_{VC}
2. increase data size N a lot
3. increase generalization error tolerance ϵ
4. all of the above

Fun Time

Consider the VC Bound below. How can we decrease the probability of getting **BAD** data?

$$\mathbb{P}_{\mathcal{D}}\Big[\big|E_{\text{in}}(g) - E_{\text{out}}(g)\big| > \epsilon\Big] \quad \leq \quad \underbrace{\boxed{4(2N)^{d_{\text{VC}}} \exp\left(-\tfrac{1}{8}\epsilon^2 N\right)}}_{\delta}$$

1. decrease model complexity d_{VC}
2. increase data size N a lot $\boxed{\delta \propto N \cdot e^{-N} = N / e^{N} \Rightarrow N \nearrow \Rightarrow \delta \searrow}$
3. increase generalization error tolerance ϵ
4. all of the above

Reference Answer: 4

Congratulations on being Master of VC bound! :-)

Summary

1. When Can Machines Learn?
2. **Why** Can Machines Learn?

Lecture 6: Theory of Generalization

Lecture 7: The VC Dimension

- Definition of VC Dimension

  maximum non-break point

- VC Dimension of Perceptrons

  $d_{\text{VC}}(\mathcal{H}) = d + 1$

- Physical Intuition of VC Dimension

  $d_{\text{VC}} \approx \#$**free parameters**

- Interpreting VC Dimension

  loosely: model complexity & sample complexity

- **next: more than noiseless binary classification?**

3. How Can Machines Learn?
4. How Can Machines Learn Better?