寶可夢、數碼寶貝分類器

淺談機器學習原理

Review: Basic Idea of ML

https://youtu.be/Ye018rCVvOo

https://youtu.be/bHcJCp2Fyxs

Step 1: function with unknown → Step 2: define loss → Step 3: optimization

Review: Strategy

https://youtu.be/WeHM2xpYQpw

More parameters, easier to overfit. Why?

Case Study: Pokémon v.s. Digimon

https://medium.com/@tyreeostevenson/teaching-a-computer-to-classify-anime-8c77bc89b881

Pokémon vs. Digimon

這是數碼寶貝
的蟲蟲獸

這才是寶可夢
的綠毛蟲

Pokémon vs. Digimon

小智身邊有小火龍

太一身邊有亞古獸

Pokémon/Digimon Classifier

- We want to find a function ……

$f($  $) =$ Pokémon or Digimon

Determine a function with unknown parameters
(based on domain knowledge)

Observation

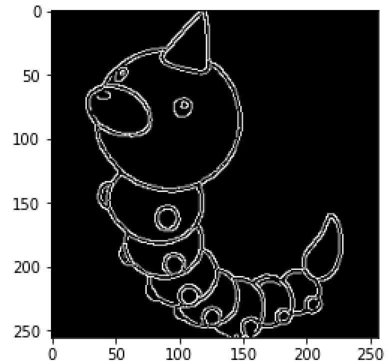
Digimon

線條較複雜?

Pokémon

線條較簡單?

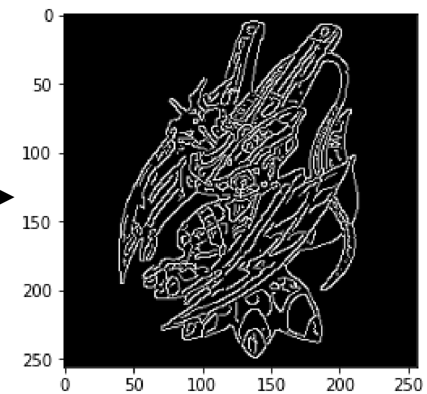
Observation



Edge detection

3558

$e($  $) = 3558$



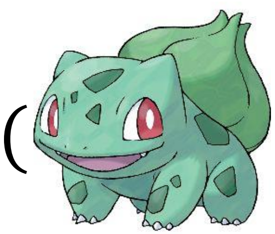
7389

$e($  $) = 7389$

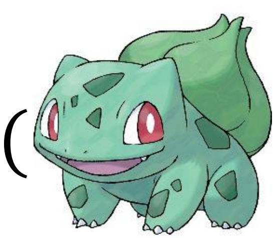
Function with Unknown Parameters

$f($  $) =$

Digimon

If $e($  $) \geq h$

Pokémon

If $e($  $) < h$

f_h: function with threshold h

$\mathcal{H} = \{1,2,\cdots,10{,}000\}$

$|\mathcal{H}|$: number of candidate functions (model "complexity")

Loss of a function (given data)

- Given a dataset $\mathcal{D}$

Pokémon

$$\mathcal{D} = \{(x^1, \hat{y}^1), (x^2, \hat{y}^2), \cdots, (x^N, \hat{y}^N)\}$$

- Loss of a threshold h given data set $\mathcal{D}$

$$L(h, \mathcal{D}) = \frac{1}{N}\sum_{n=1}^{N} \boxed{l(h, x^n, \hat{y}^n)} \rightarrow \mathrm{I}(f_h(x^n) \neq \hat{y}^n)$$

Error rate

If $f_h(x^n) \neq \hat{y}^n$
Output 1
Otherwise
Output 0

Don't like it? Of course, you can choose cross-entropy. ☺

Training Examples

- If we can collect all Pokémons and Digimons in the universe $\mathcal{D}_{all}$, we can find the best threshold h^{all}

$$h^{all} = arg \min_h L(h, \mathcal{D}_{all})$$

- We only collect some examples $\mathcal{D}_{train}$ from $\mathcal{D}_{all}$

$$\mathcal{D}_{train} = \{(x^1, \hat{y}^1), (x^2, \hat{y}^2), \cdots, (x^N, \hat{y}^N)\}$$

$$(x^n, \hat{y}^n) \sim \mathcal{D}_{all}$$

independently and identically distributed (i.i.d.)

$$h^{train} = arg \min_h L(h, \mathcal{D}_{train})$$

Training Examples

- If we can collect all Pokémons and Digimons in the universe $\mathcal{D}_{all}$, we can find the best threshold h^{all}

$$h^{all} = arg \min_h L(h, \mathcal{D}_{all})$$

理想

- We only collect some examples $\mathcal{D}_{train}$ from $\mathcal{D}_{all}$

$$h^{train} = arg \min_h L(h, \mathcal{D}_{train})$$

現實

We hope $L(h^{train}, \mathcal{D}_{all})$ and $L(h^{all}, \mathcal{D}_{all})$ are close.

現實

理想

We hope $L(h^{train}, \mathcal{D}_{all})$ and $L(h^{all}, \mathcal{D}_{all})$ are close.

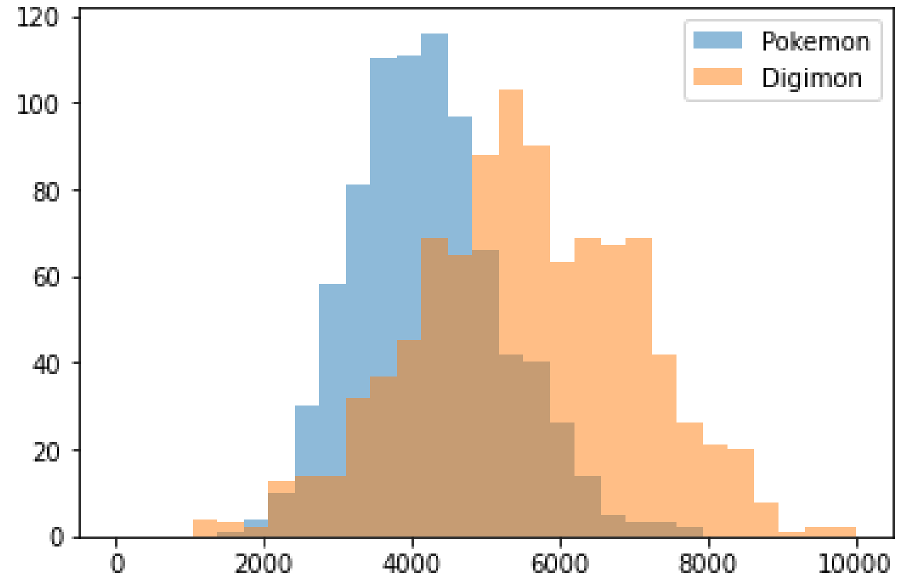
Pokémon: 819

Digimon: 971

In most applications, you cannot obtain $\mathcal{D}_{all}$.

(Testing data $\mathcal{D}_{test}$ as the proxy of $\mathcal{D}_{all}$)

All Pokémons and Digimons we know as $\mathcal{D}_{all}$



$h^{all} = 4824$

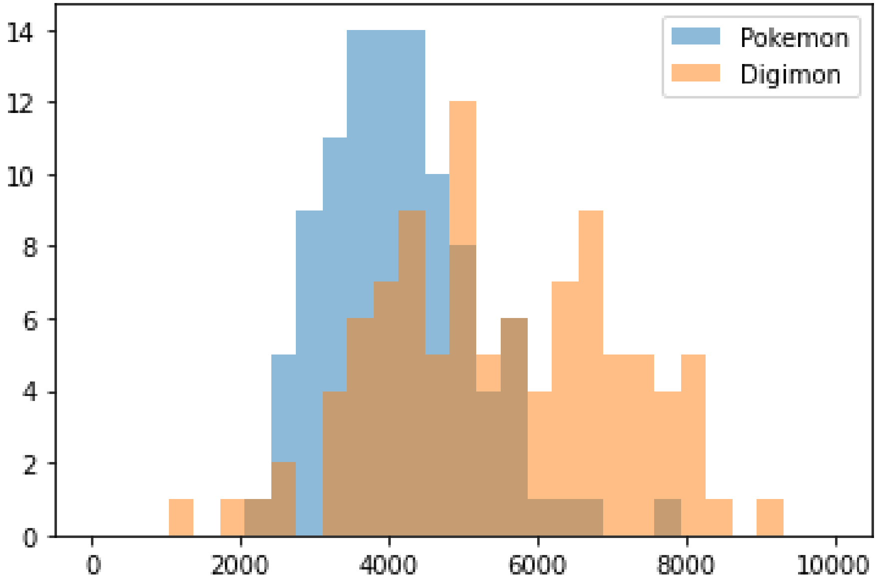
$L(h^{all}, \mathcal{D}_{all}) = 0.28$

Source of Digimon: https://github.com/mrok273/Qiita

Source of Pokémon: https://www.kaggle.com/kvpratama/pokemon-images-dataset/data

We hope $L(h^{train}, \mathcal{D}_{all})$ and $L(h^{all}, \mathcal{D}_{all})$ are close.

Sample 200 Pokémons and Digimons as $\mathcal{D}_{train1}$

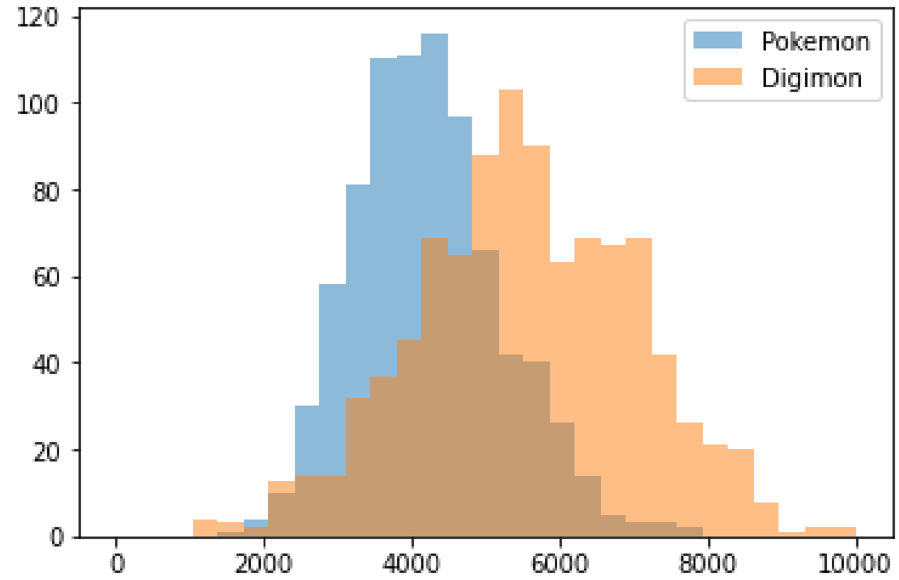


$h^{train1} = 4727$

$L(h^{train1}, \mathcal{D}_{train1}) = 0.27$

Even lower than $L(h^{all}, \mathcal{D}_{all})$?

All Pokémons and Digimons we know as $\mathcal{D}_{all}$



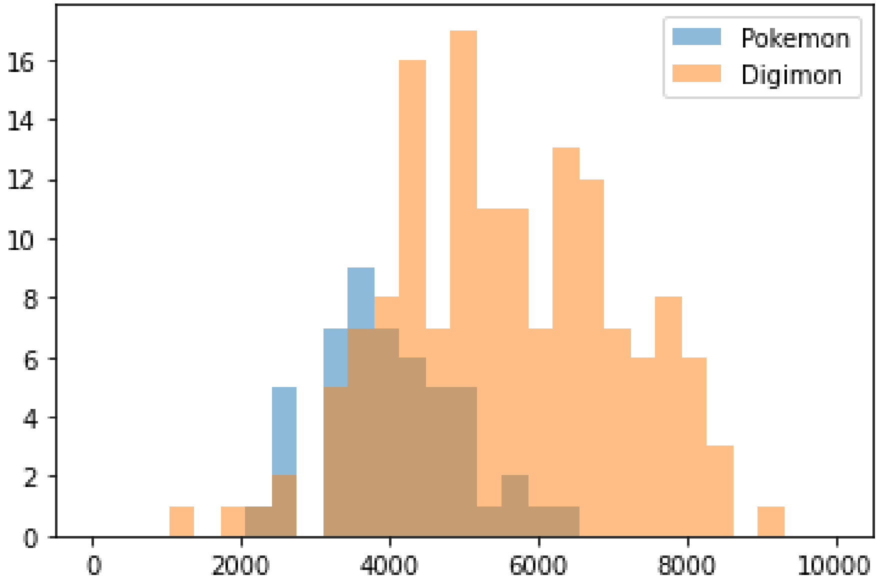
$h^{all} = 4824$

$L(h^{all}, \mathcal{D}_{all}) = 0.28$

$L(h^{train1}, \mathcal{D}_{all}) = 0.28$

We hope $L(h^{train}, \mathcal{D}_{all})$ and $L(h^{all}, \mathcal{D}_{all})$ are close.

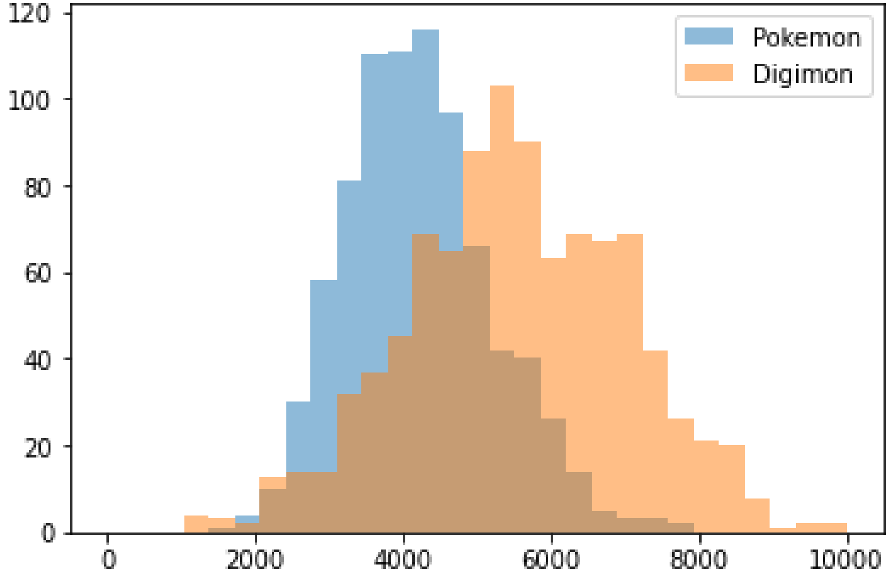
Sample 200 Pokémons and Digimons as $\mathcal{D}_{train2}$



$h^{train2} = 3642$

$L(h^{train2}, \mathcal{D}_{train2}) = 0.20$

All Pokémons and Digimons we know as $\mathcal{D}_{all}$



$h^{all} = 4824$

$L(h^{all}, \mathcal{D}_{all}) = 0.28$

$L(h^{train2}, \mathcal{D}_{all}) = 0.37$

What do we want?

$L(h^{train}, \mathcal{D}_{train})$ can be smaller than $L(h^{all}, \mathcal{D}_{all})$

We want $L(h^{train}, \mathcal{D}_{all}) - L(h^{all}, \mathcal{D}_{all}) \leq \delta$

What kind of $\mathcal{D}_{train}$ fulfill it?

$\forall h \in \mathcal{H}, |L(h, \mathcal{D}_{train}) - L(h, \mathcal{D}_{all})| \leq \delta/2$

$\mathcal{D}_{train}$ is a good proxy of $\mathcal{D}_{all}$ for evaluating loss L given any h.

What do we want?

We want $L\left(h^{train}, \mathcal{D}_{all}\right) - L\left(h^{all}, \mathcal{D}_{all}\right) \leq \delta$

What kind of $\mathcal{D}_{train}$ fulfill it?

$$\forall h \in \mathcal{H}, \left|L(h, \mathcal{D}_{train}) - L(h, \mathcal{D}_{all})\right| \leq \delta/2$$

$$L\left(h^{train}, \mathcal{D}_{all}\right) \leq \underline{L\left(h^{train}, \mathcal{D}_{train}\right)} + \delta/2$$

$$\leq \underline{L\left(h^{all}, \mathcal{D}_{train}\right)} + \delta/2$$

$$h^{train} = arg \min_h L(h, \mathcal{D}_{train})$$

$$\leq L\left(h^{all}, \mathcal{D}_{all}\right) + \delta/2 + \delta/2 \;= L\left(h^{all}, \mathcal{D}_{all}\right) + \delta$$

What do we want?

We want $L(h^{train}, \mathcal{D}_{all}) - L(h^{all}, \mathcal{D}_{all}) \leq \delta$

What kind of $\mathcal{D}_{train}$ fulfill it?

$$\forall h \in \mathcal{H}, |L(h, \mathcal{D}_{train}) - L(h, \mathcal{D}_{all})| \leq \delta/2$$

We want to sample **good** $\mathcal{D}_{train}$

$\varepsilon = \delta/2$

$$\forall h \in \mathcal{H}, |L(h, \mathcal{D}_{train}) - L(h, \mathcal{D}_{all})| \leq \varepsilon$$

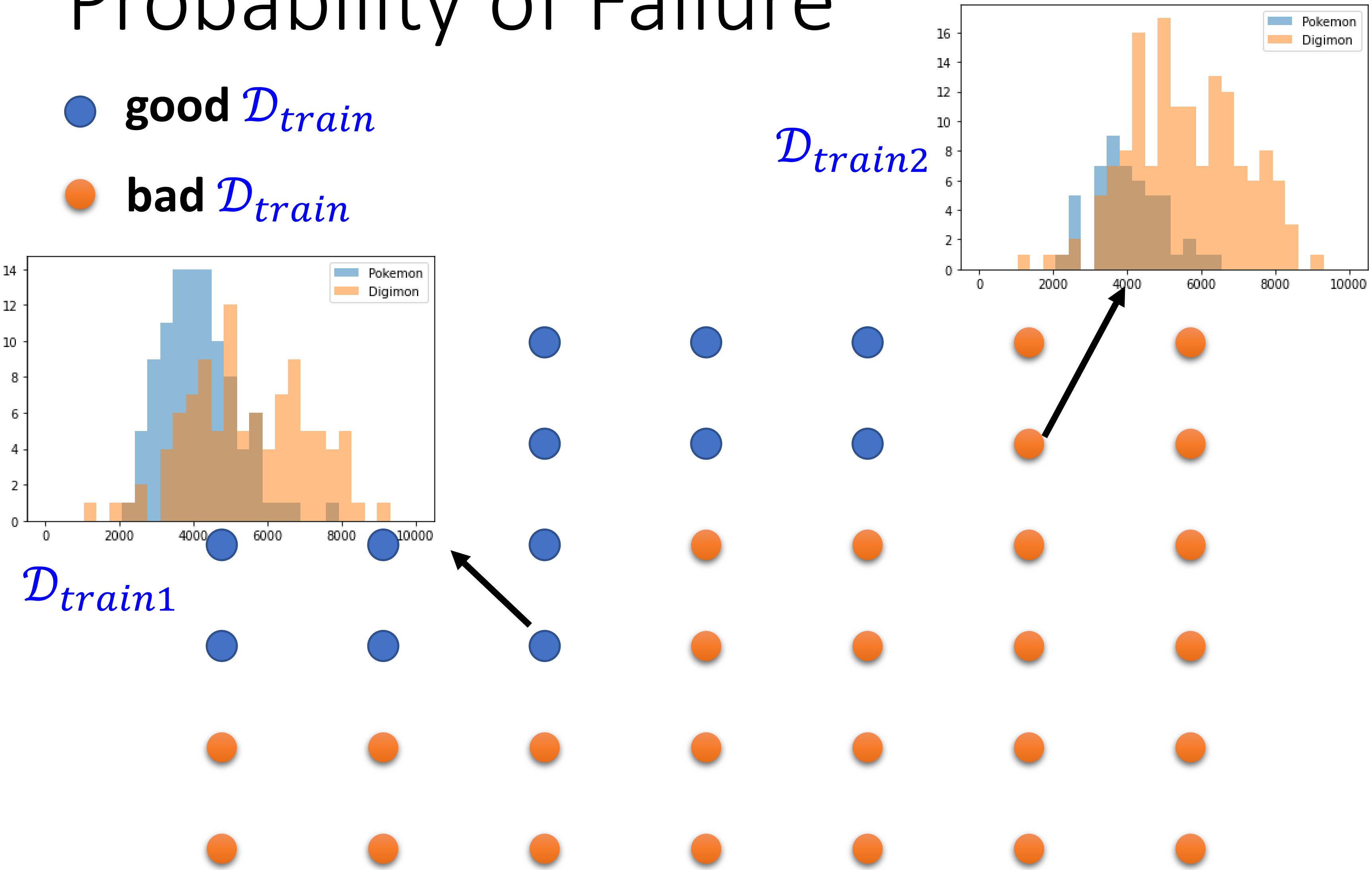
What is the probability of sampling **bad** $\mathcal{D}_{train}$?

Very General!

- The following discussion is **model-agnostic**.
- In the following discussion, we don't have assumption about **data distribution**.
- In the following discussion, we can use any **loss function**.

Probability of Failure

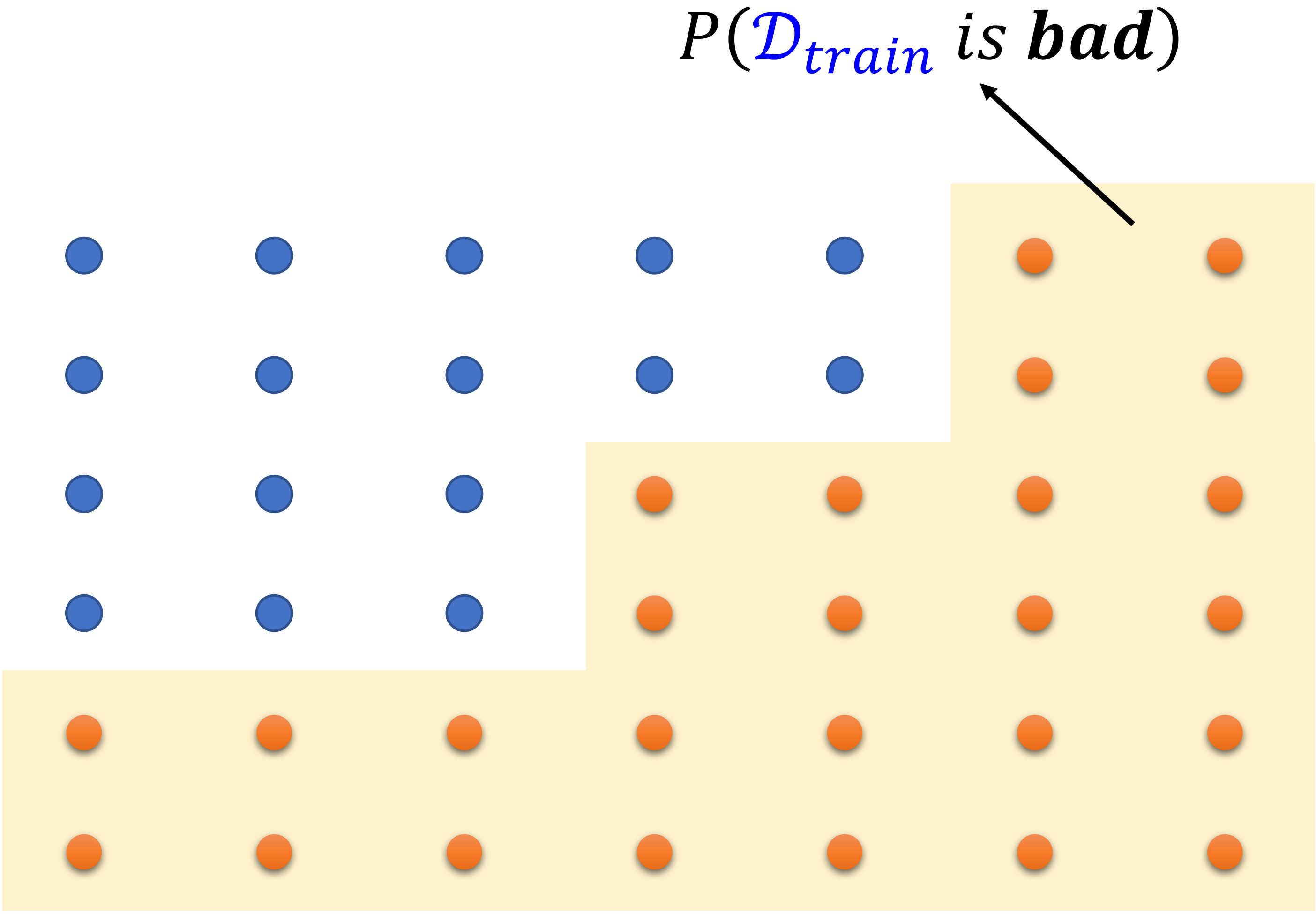
- good $\mathcal{D}_{train}$
- bad $\mathcal{D}_{train}$



Each point is a training set.

Probability of Failure

Each point is a training set.

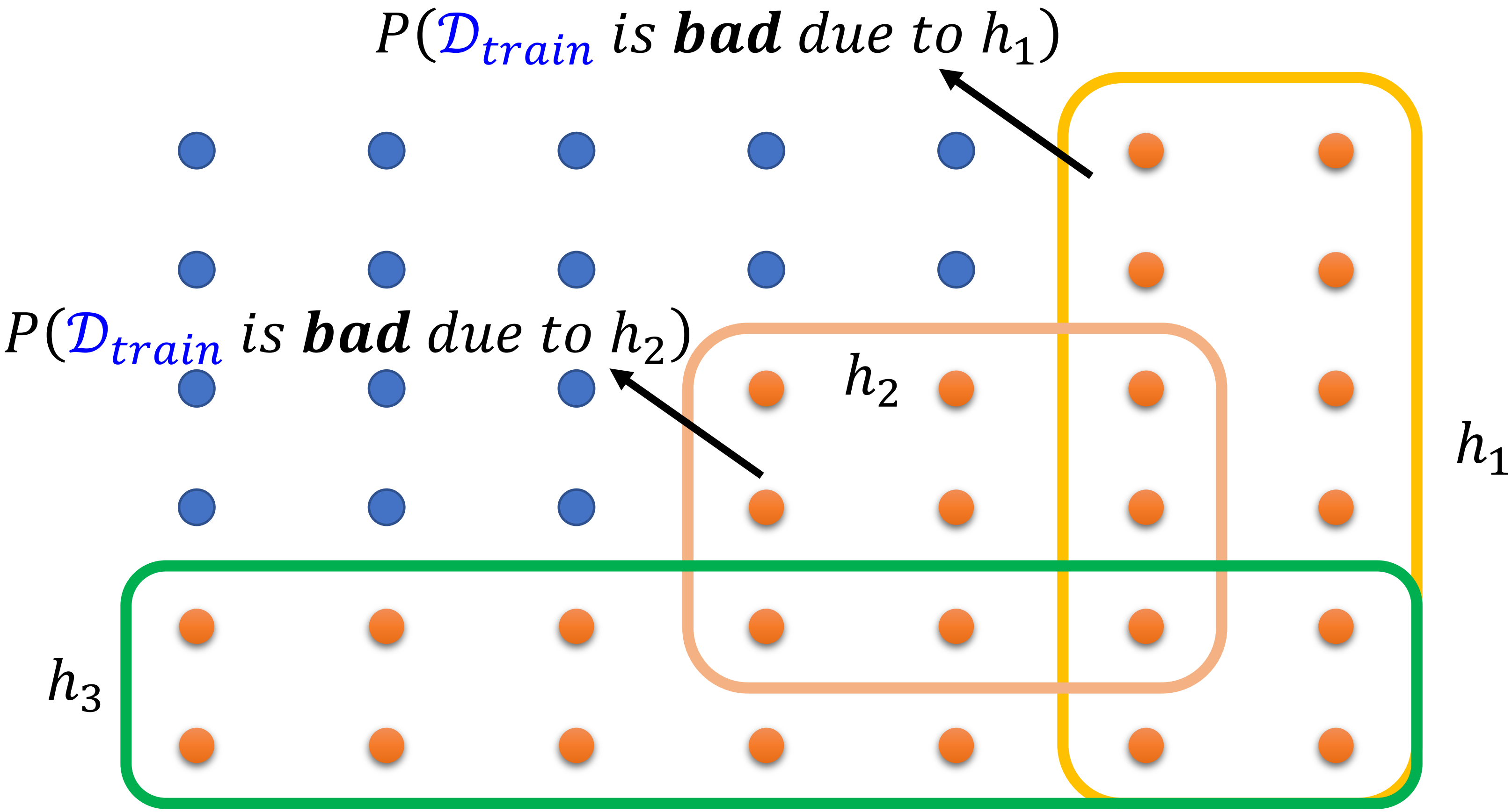


Probability of Failure

Each point is a training set.

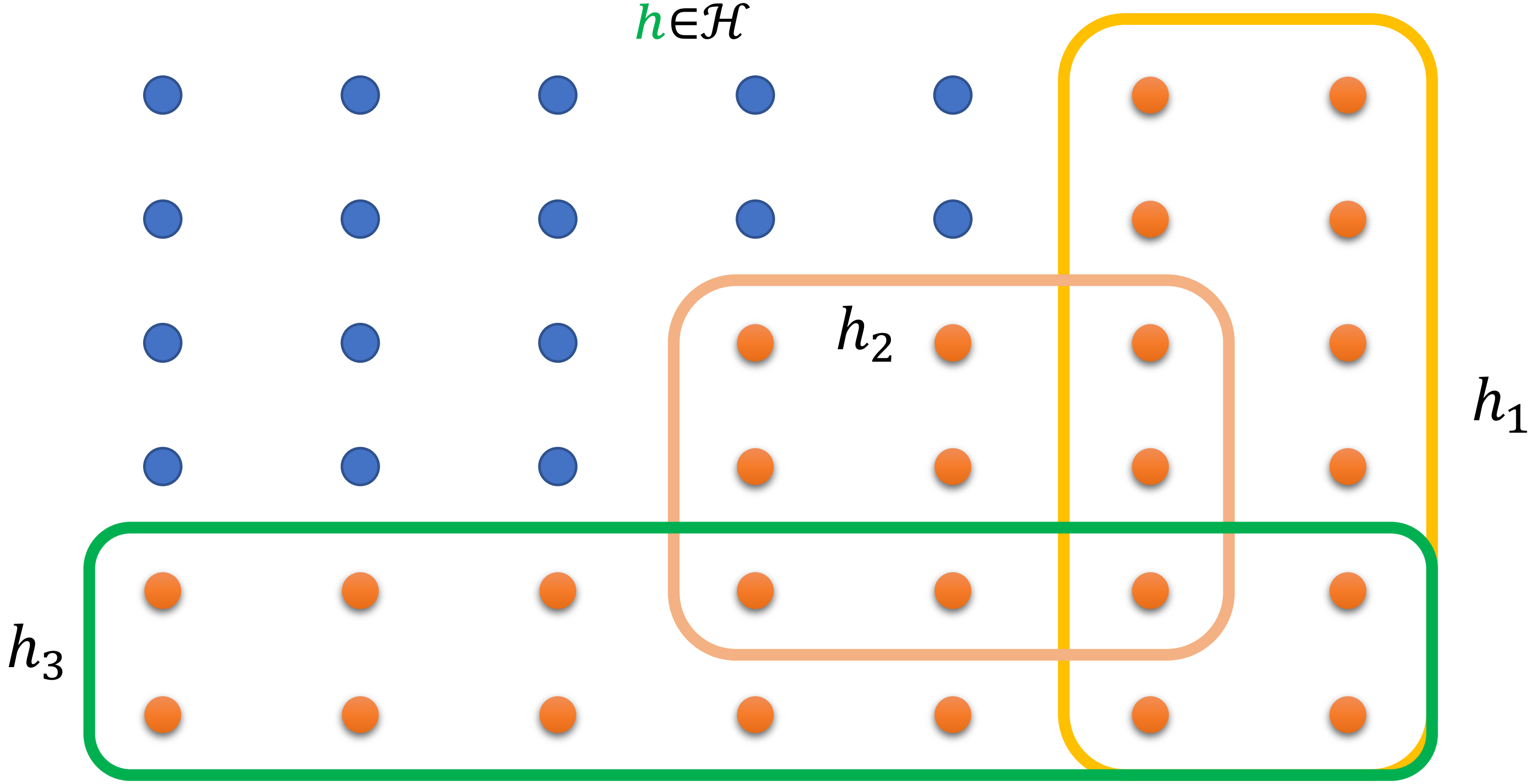
If a $\mathcal{D}_{train}$ is **bad**,

at least one h makes $|L(h, \mathcal{D}_{train}) - L(h, \mathcal{D}_{all})| > \varepsilon$



$$P(\mathcal{D}_{train} \text{ is } \textbf{bad}) = \bigcup_{h \in \mathcal{H}} P(\mathcal{D}_{train} \text{ is } \textbf{bad} \text{ due to } h)$$

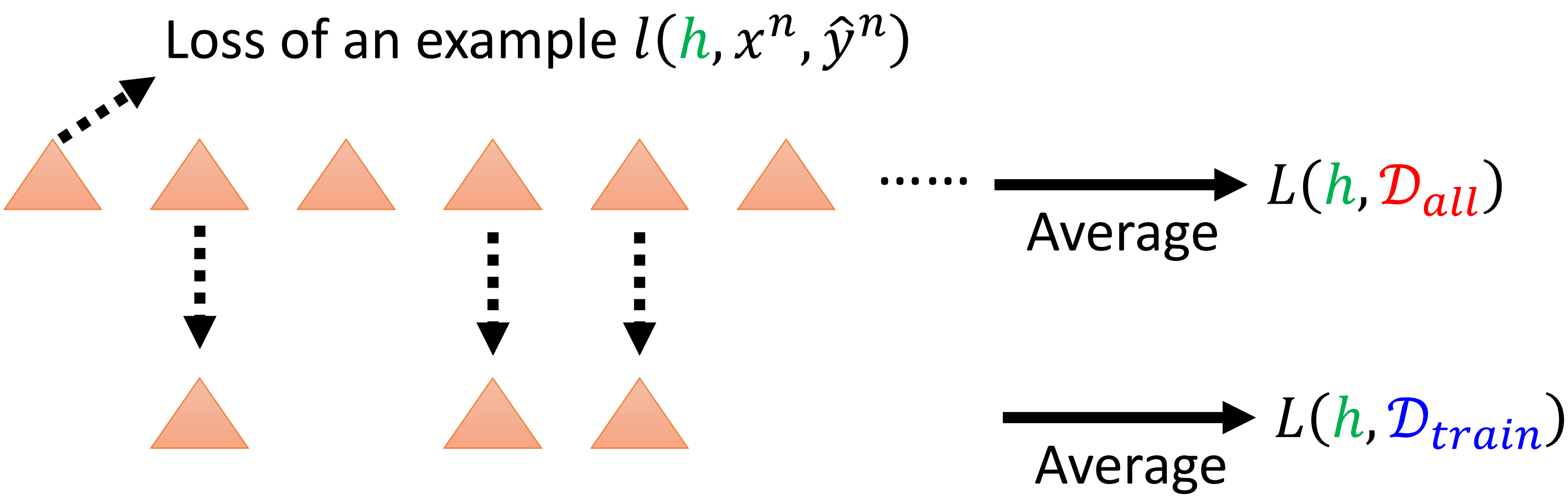
$$\leq \sum_{h \in \mathcal{H}} P(\mathcal{D}_{train} \text{ is } \textbf{bad} \text{ due to } h)$$



$$P(\mathcal{D}_{train} \text{ is } \textbf{bad}) = \bigcup_{h \in \mathcal{H}} P(\mathcal{D}_{train} \text{ is } \textbf{bad} \text{ due to } h)$$

$$\leq \sum_{h \in \mathcal{H}} P(\boxed{\mathcal{D}_{train} \text{ is } \textbf{bad} \text{ due to } h})$$

$$|L(h, \mathcal{D}_{train}) - L(h, \mathcal{D}_{all})| > \varepsilon \qquad L(h, \mathcal{D}) = \frac{1}{N} \sum_{n=1}^{N} l(h, x^n, \hat{y}^n)$$



$$P(\mathcal{D}_{train}\ is\ \boldsymbol{bad}) = \bigcup_{h \in \mathcal{H}} P(\mathcal{D}_{train}\ is\ \boldsymbol{bad}\ due\ to\ h)$$

$$\leq \sum_{h \in \mathcal{H}} \boxed{P(\mathcal{D}_{train}\ is\ \boldsymbol{bad}\ due\ to\ h)}$$

Hoeffding's Inequality:

$$P(\mathcal{D}_{train}\ is\ bad\ due\ to\ h) \leq \boxed{2exp(-2N\varepsilon^2)}$$

- The range of loss L is $[0,1]$
- N is the number of examples in $\mathcal{D}_{train}$

$$P(\mathcal{D}_{train}\ is\ \boldsymbol{bad}) = \bigcup_{h \in \mathcal{H}} P(\mathcal{D}_{train}\ is\ \boldsymbol{bad}\ due\ to\ h)$$

$$\leq \sum_{h \in \mathcal{H}} P(\mathcal{D}_{train}\ is\ \boldsymbol{bad}\ due\ to\ h)$$

$$\leq \sum_{h \in \mathcal{H}} 2exp(-2N\varepsilon^2)$$

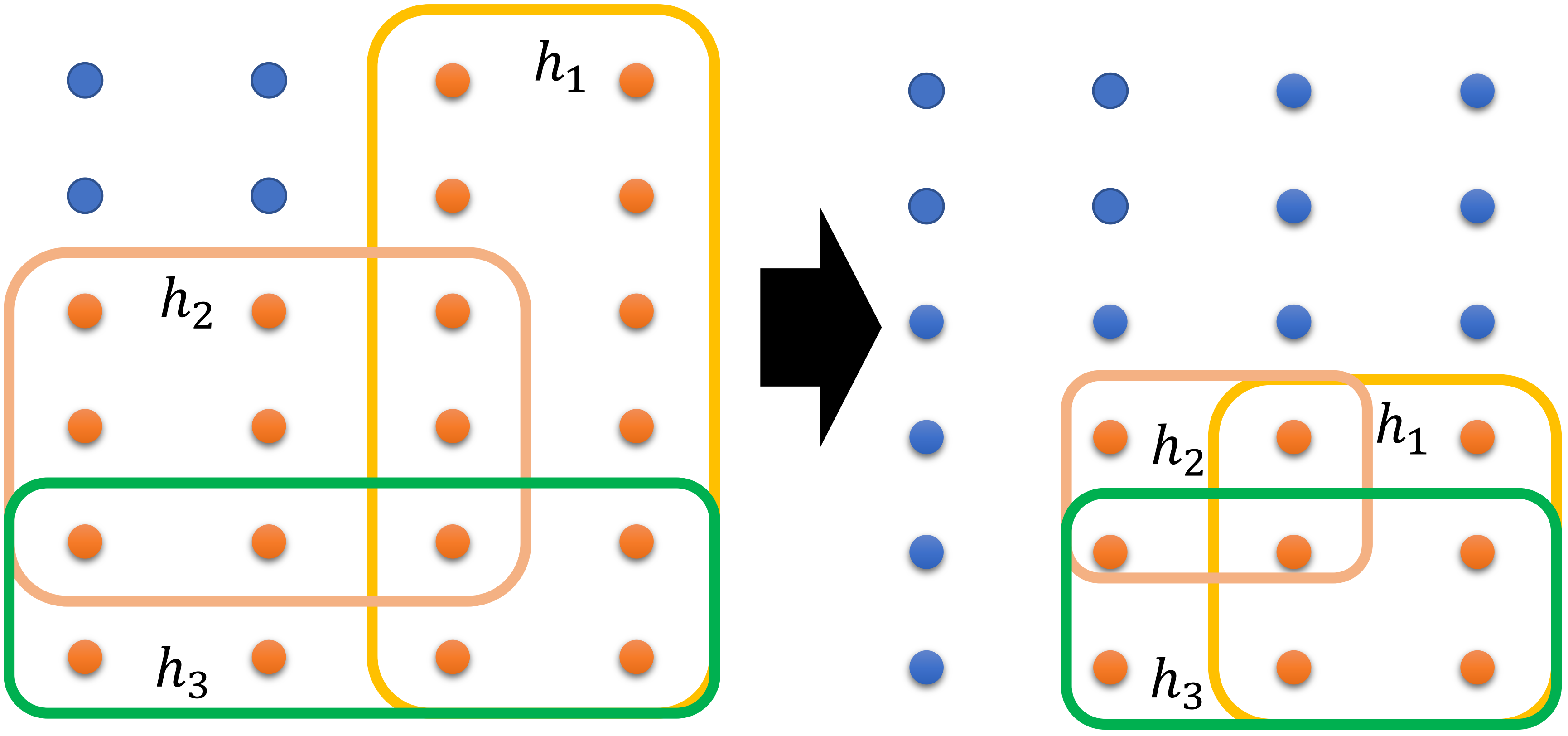
$$= |\mathcal{H}| \cdot 2exp(-2N\varepsilon^2)$$

How to make $P(\mathcal{D}_{train}\ is\ bad)$ smaller?

Larger N and smaller $|\mathcal{H}|$

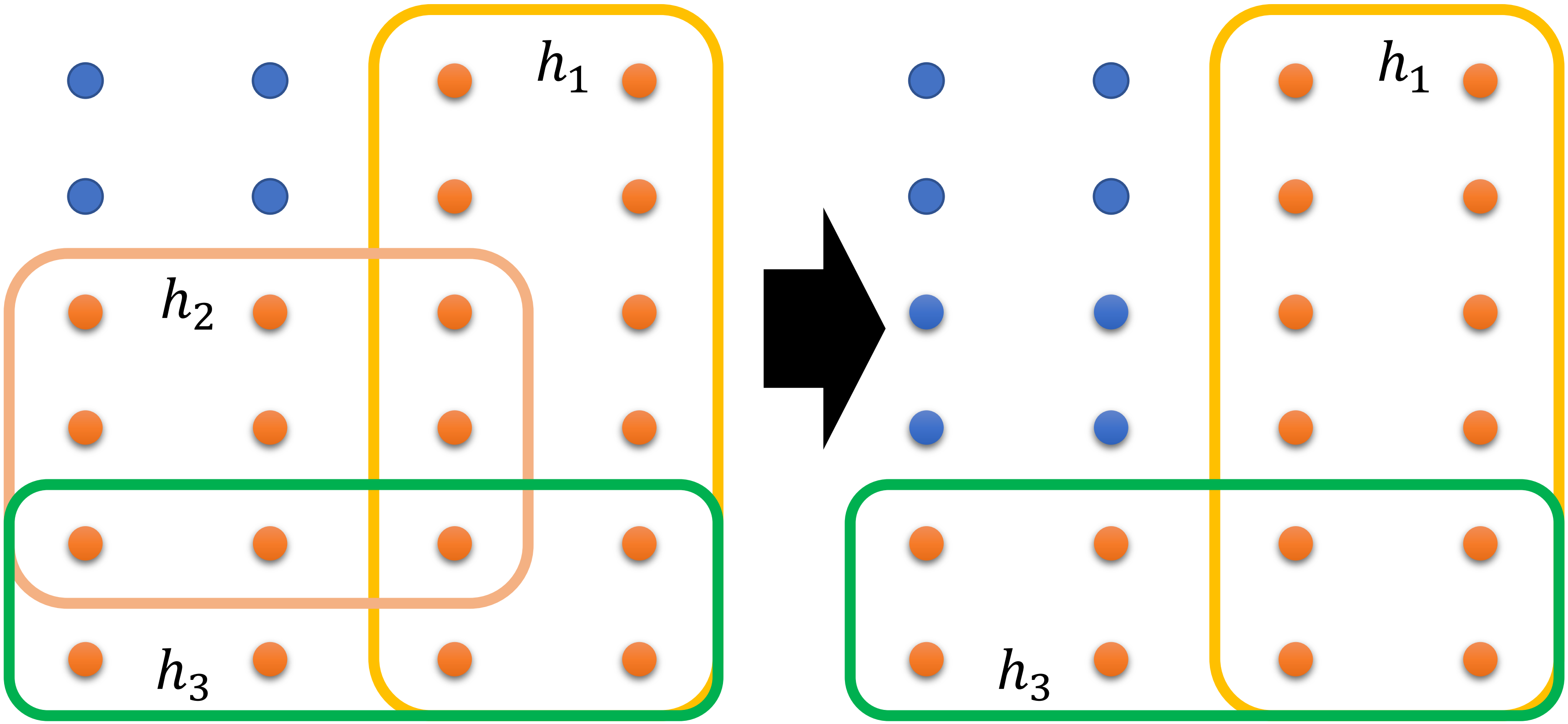
$$P(\mathcal{D}_{train} \text{ is } \boldsymbol{bad}) \leq |\mathcal{H}| \cdot 2exp(-2N\varepsilon^2)$$

Larger N



$$P(\mathcal{D}_{train} \ is \ \boldsymbol{bad}) \leq |\mathcal{H}| \cdot 2exp(-2N\varepsilon^2)$$

<u>Smaller $|\mathcal{H}|$</u>



Example

$$\mathcal{H} = \{1,2,\cdots,10{,}000\}$$
$$\mathcal{D}_{train} = \{(x^1,\hat{y}^1),(x^2,\hat{y}^2),\cdots,(x^N,\hat{y}^N)\}$$
$$\forall h \in \mathcal{H}, |L(h,\mathcal{D}_{train}) - L(h,\mathcal{D}_{all})| \le \varepsilon$$

$$P(\mathcal{D}_{train}\ is\ \boldsymbol{bad}) \le |\mathcal{H}| \cdot 2exp(-2N\varepsilon^2)$$

$$|\mathcal{H}| = 10000, N = 100, \varepsilon = 0.1$$

Usually happen QQ

$$P(\mathcal{D}_{train}\ is\ bad) \le 2707$$

$$|\mathcal{H}| = 10000, N = 500, \varepsilon = 0.1$$

$$P(\mathcal{D}_{train}\ is\ bad) \le 0.91$$

$$|\mathcal{H}| = 10000, N = 1000, \varepsilon = 0.1$$

$$P(\mathcal{D}_{train}\ is\ bad) \le 0.00004$$

Example

$$P(\mathcal{D}_{train}\ is\ \boldsymbol{bad}) \leq |\mathcal{H}| \cdot 2exp(-2N\varepsilon^2)$$

If we want $P(\mathcal{D}_{train}\ is\ bad) \leq \delta$

How many training examples do we need?

$$|\mathcal{H}| \cdot 2exp(-2N\varepsilon^2) \leq \delta \Rightarrow N \geq \frac{log(2|\mathcal{H}|/\delta)}{2\varepsilon^2}$$

$$|\mathcal{H}| = 10000, \delta = 0.1, \varepsilon = 0.1$$

$$\Rightarrow N \geq 610$$

Model Complexity

$$P(\mathcal{D}_{train}\ is\ \boldsymbol{bad}) \leq |\mathcal{H}| \cdot 2exp(-2N\varepsilon^2)$$

The number of possible functions you can select

What if the parameters are continuous?

- *Answer 1*: Everything that happens in a computer is discrete. ☺
- *Answer 2*: VC-dimension (not this course)

Model Complexity

$$P(\mathcal{D}_{train}\ \textit{is}\ \textbf{\textit{bad}}) \le |\mathcal{H}| \cdot 2exp(-2N\varepsilon^2)$$

Why don't we simply use a very small $|\mathcal{H}|$?

理想崩壞

"$\mathcal{D}_{train}$ is **good**" means …

$$\boxed{\forall h \in \mathcal{H}, |L(h, \mathcal{D}_{train}) - L(h, \mathcal{D}_{all})| \le \varepsilon}$$

Larger loss

$$\boxed{L\left(h^{train}, \mathcal{D}_{all}\right) - L\left(h^{all}, \mathcal{D}_{all}\right) \le \delta}$$

$\varepsilon = \delta/2$

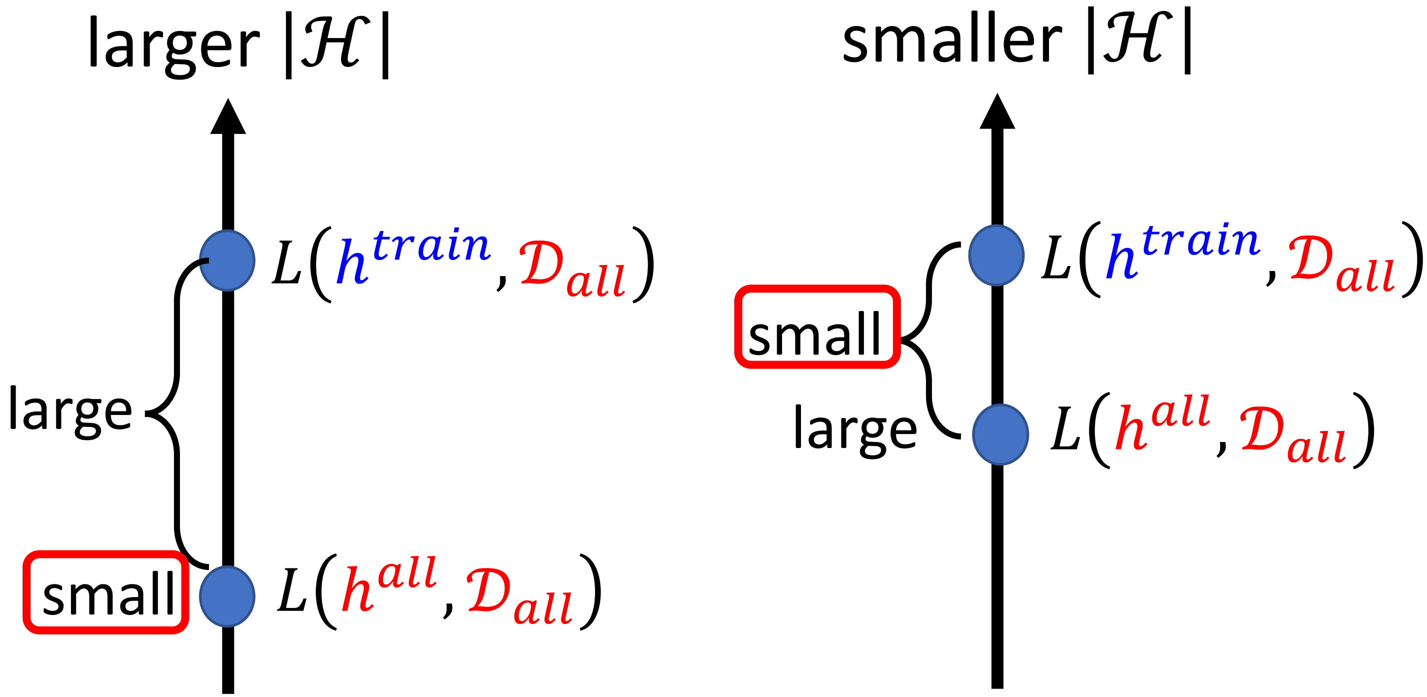
$$h^{all} = arg \min_{h \in \mathcal{H}} L(h, \mathcal{D}_{all})$$

fewer candidates

Tradeoff of Model Complexity

Larger N and smaller $|\mathcal{H}|$ ➡ $L(h^{train}, \mathcal{D}_{all}) - L(h^{all}, \mathcal{D}_{all}) \leq \delta$

Smaller $|\mathcal{H}|$ ➡ Larger $L(h^{all}, \mathcal{D}_{all})$



魚與熊掌可以兼得嗎？ Yes, **Deep Learning.**

https://forms.gle/FKGwMczbJPxnWe9o7