Machine Learning

Lecture 2: Is Learning Feasible?

王浩

信息科学与技术学院

Email: wanghao1@shanghaitech.edu.cn

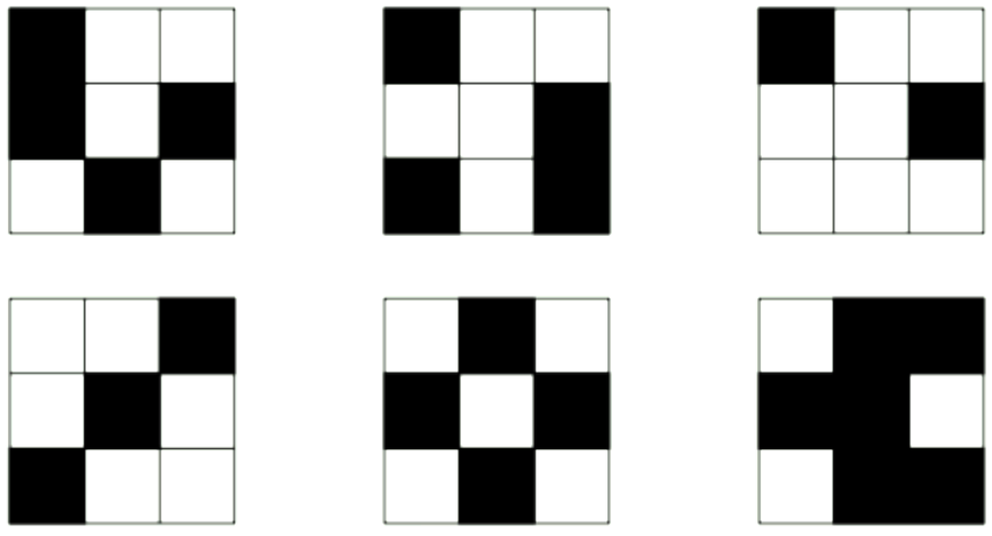
1、期望风险和经验风险

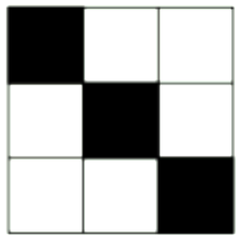
- 数据的联合分布
- 假设类
- 经验风险
- 经验风险极小化

Outside the data set

For example, we have

$$f(\boldsymbol{x}) = \begin{matrix} 8, & 9, & 3, \\ 6, & 8, & 12 \end{matrix}$$





$$f(\boldsymbol{x}) = ?$$

- It seems the bottom line has more impact on f value
- What is the unknown value? – We cannot rule out either possibility

Sample, label, target

- In learning we seek a mapping from the initial data $\mathscr{X}$ (样本空间, sample space) to some label set $\mathscr{Y}$ (label space，标签空间)
- Training set $\mathscr{D} \subset \mathscr{X}$: your sample set to work on
- Target concept（目标概念） $f : \mathscr{X} \to \mathscr{Y}$
- Concept class (概念类) : $\mathscr{C}$

- Example: in character recognition, $\mathcal{X}$ consists of possible images of letters and $\mathcal{Y}$ consists of the twenty-six letters of the Latin alphabet.

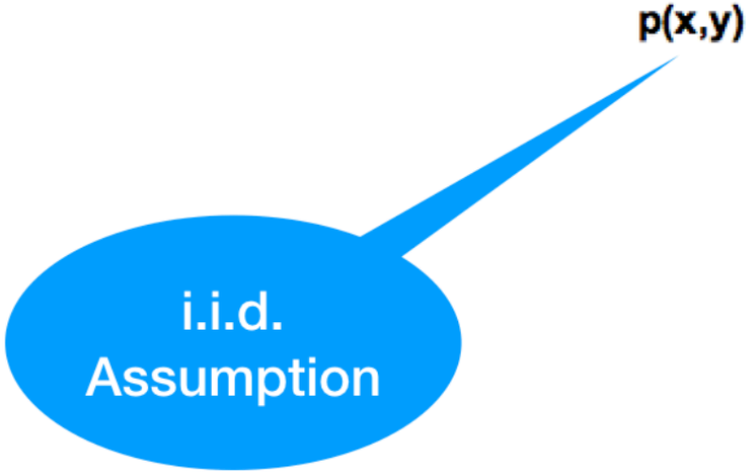
- Example: For simplicity we will use binary labels $\{+1, -1\}$. Whether something is the letter "G" (+1) or not the letter "G" (–1), or whether given image contains a face (+1) or does not contain a face (–1).

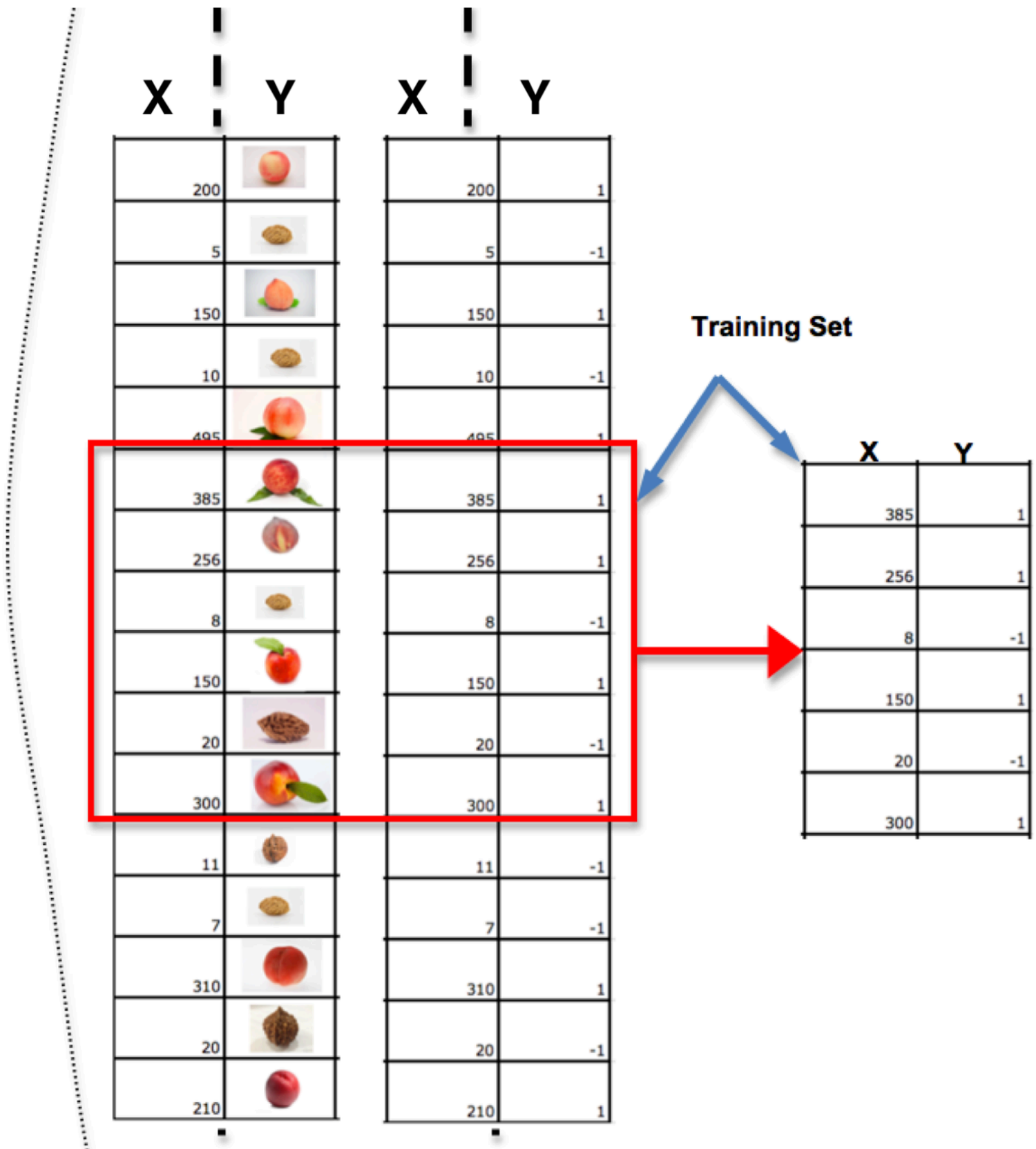
Joint Distribution

- Joint distribution $p_{X,Y}(x, y)$
- Future data is coming from some unknown source joint distribution $p_{X,Y}$ over input objects and their corresponding labels, which we write as the joint distribution $p_{X,Y}$, where $X \in \mathcal{X}, Y \in \mathcal{Y}$
- Example: Character recognition source distribution would assign much more probability to (“image containing a circular shape”, “O”) than to (“image containing a circular shape”, “T”). (why?)

Toy Example:

Peach Example





Conditional Probability/Density

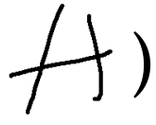
- Conditional distribution $p_{Y|X}(y \mid x)$

- We can define course joint distribution as really having two components

$$p_{XY}(x, y) = p_{Y|X}(y \mid x) \cdot p_X(x)$$

- $p_{Y|X}(y \mid x)$ is the conditional probability of the label random variable Y given the appearance random variable and $p_X(x)$ is marginal probability of the input image.

- Example: In character recognition we may have

$$p_{Y|X}(Y = \text{"A"} \mid X = \text{  }) = 0.9$$

$$p_{Y|X}(Y = \text{"O"} \mid X = \text{  }) = 0.6$$

$$p_{Y|X}(Y = \text{"a"} \mid X = \text{  }) = 0.4$$

Hypothesis

- Hypothesis h (假设): a hypothesis (a predictor) h is a function from $\mathcal{X}$ to $\mathcal{Y}$, $h : \mathcal{X} \to \mathcal{Y}$
- Hypothesis class/set/space（假设类，假设集合，假设空间） $\mathcal{H}$ is a set of predictors $\mathcal{H}$
- In a learning problem, we restrict hypotheses in a certain class.
- Example: Consider binary labels, so the label set is $\mathcal{Y} = \{+1, -1\}$. In addition, we may consider $\mathcal{X} = \mathbb{R}^2$
- Some specific examples of hypothesis classes:

$$\mathcal{H} = \{\text{sign}(w^T x + b) \mid w \in \mathbb{R}^2, b \in \mathbb{R}\}$$

$$\mathcal{H} = \{\sum_{i=1}^{2} x_i \leq \theta \mid \theta \in \mathbb{R}_+\}$$

Loss (cost/risk) function

- Loss function $\text{loss}(x, y)$: How we can measure the performance of h on a given (input, label) pair (x, y)
  $\text{loss}(x, y) = \text{loss}(h(x), y)$

- Example: If the label $h(x)$ does not match the provided label y, we incur a loss of 1 and if the prediction $h(x)$ does match the provided label y, we incur 0 loss.

- The loss function that represents this measure of performance is called the 0-1 loss and defined as

$$\text{loss}(h(x), y) = \begin{cases} 1 & \text{if } h(x) \neq y \\ 0 & \text{if } h(x) = y \end{cases}$$

Empirical risk (training error, in-sample error)
Empirical Risk Minimization (ERM)

- The expected loss of a predictor h on a particular observed sample data set $\mathscr{D} = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ could also be referred to as the empirical risk $E_{in}(h)$

$$E_{in}(h) = \frac{1}{m}\sum_{i=1}^{m}[\text{loss}(h(x_i), y_i)] = \frac{1}{m}\sum_{i=1}^{m}\{h(x_i) = y_i\}$$

0-1 loss

- Usually, the target predictor is found by ERM

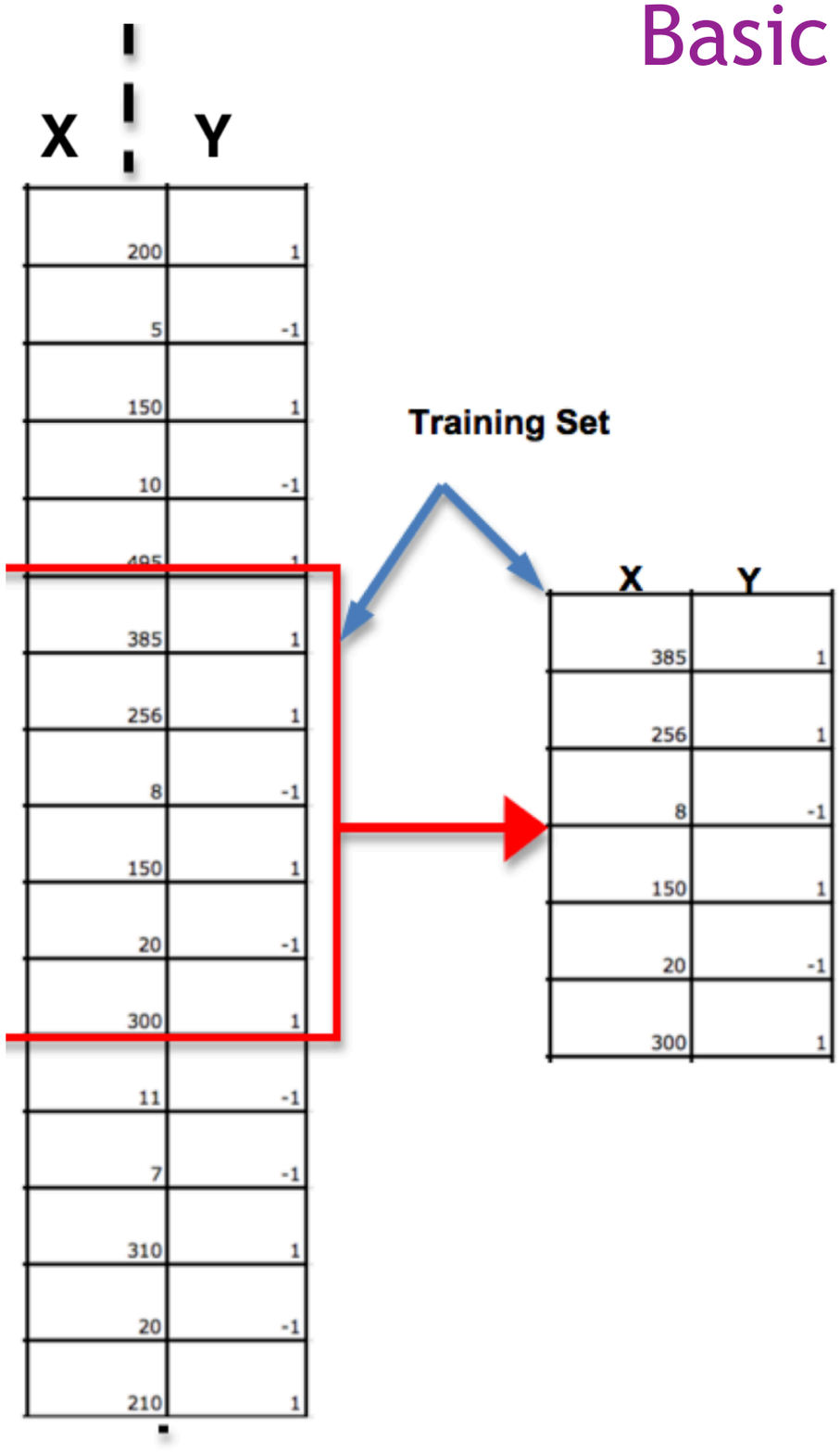
$$\hat{h} = \arg\min_{h} \; E_{in}(h), \qquad h \in \mathcal{H}$$

- Parameterize $h(\cdot\,;\theta) \iff \theta$

$$\hat{\theta} = \arg\min_{\theta} \; E_{in}(\theta), \qquad \theta \in \Theta$$

- If $E_{in}(h) = 0$ on $\mathscr{D}$, we say h agrees with $\mathscr{D}$ (interpolation)

Basic Machine Learning Procedures



Hypothesis (Prediction function):

$$h_\theta(x) = \text{sign}(x - \theta),$$ parameterize by θ

Hypothesis Class

$$\mathscr{H} = \{h_\theta \mid \theta \in [30{,}200]\}$$

Loss Function:

$$\text{loss}(h(x), y) = (h(x) - y)^2$$

Basic Machine Learning Procedures

Training set

385	1
256	1
8	-1
150	1
20	-1
300	1

$h_{30}(x) = \text{sign}(x - 30)$

1	0
-1	0
1	0
-1	0
1	0
1	0

$h_{200}(x) = \text{sign}(x - 200)$

1	0
-1	0
1	0
-1	0
-1	1
1	0

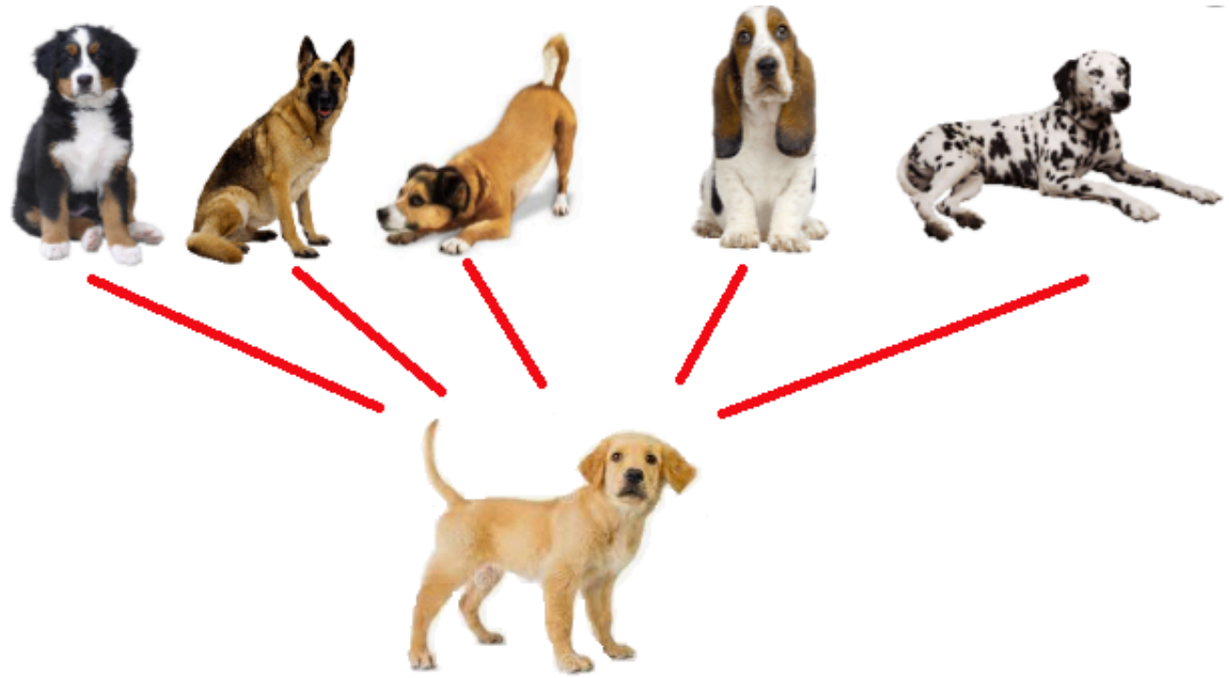
On average over training set: $E_{in}(h) = \frac{1}{m}\sum_{i=1}^{m}(h(x_i) - y_i)^2$
(training error, empirical risk, in-sample error)

$$E_{in}\left(h_{30}(x), y\right) = 0 \quad E_{in}\left(h_{200}(x), y\right) = 1/6$$

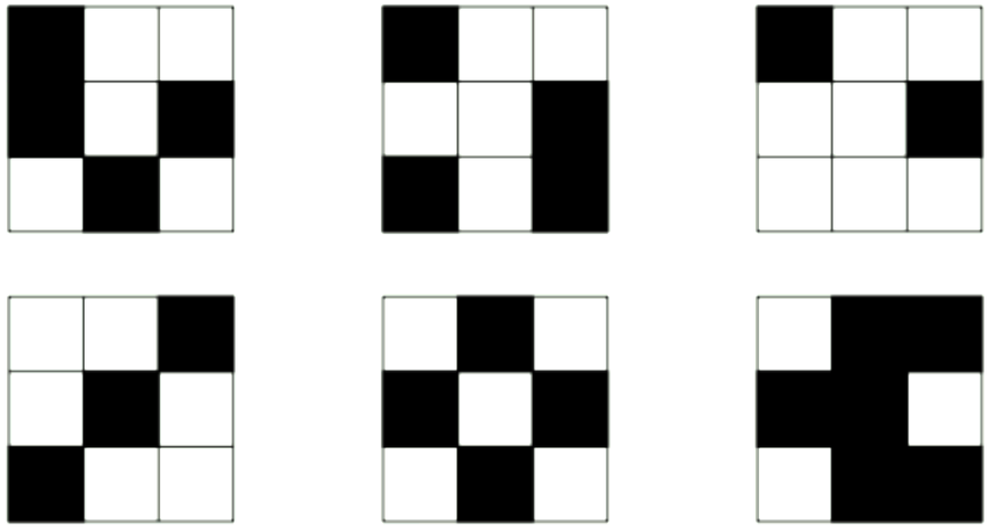
You minimize the loss over $\mathscr{H}$ on $\mathscr{D}$, and end up with the "best" predictor h_{30}

2、从训练到泛化

- 期望风险
- 罐子模型
- 从样本到分布
- 有限假设的泛化

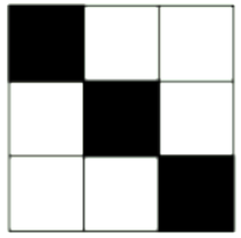


Outside the data set



$f(x) = -1$

$f(x) = +1$



$f(\boldsymbol{x}) = ?$

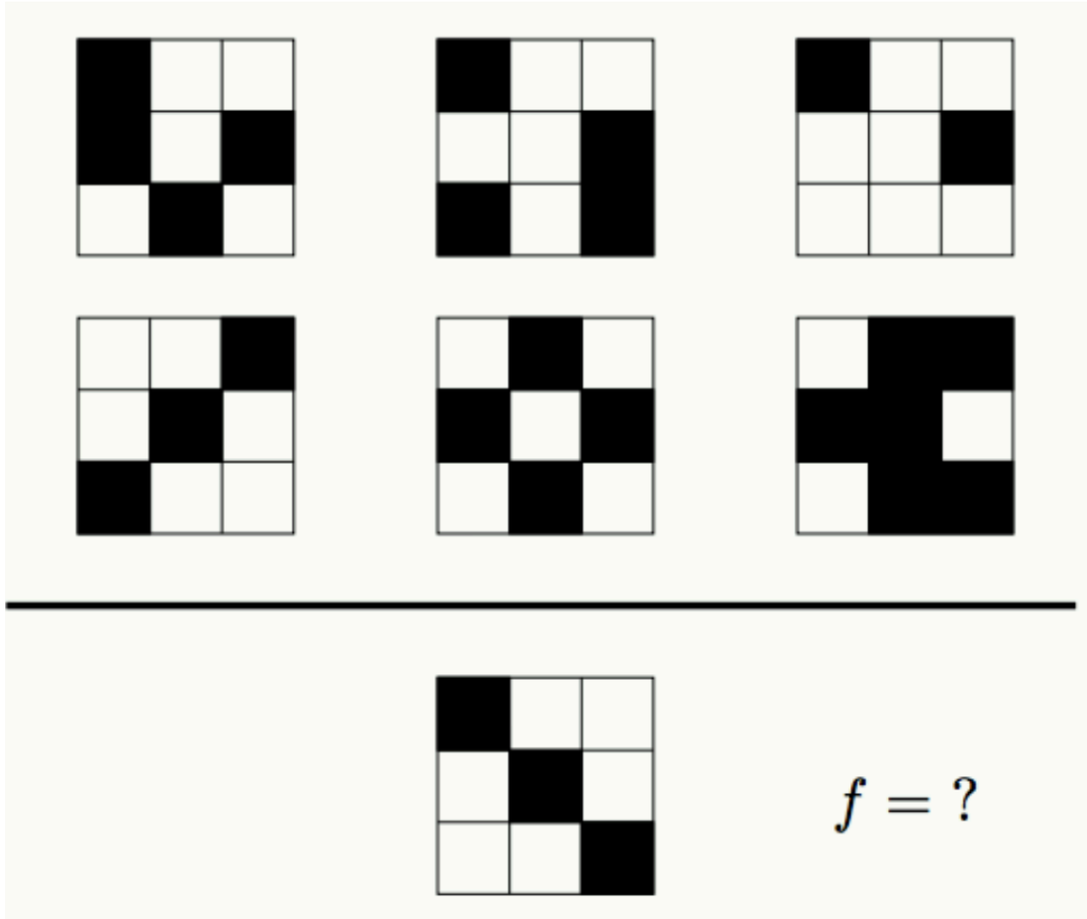
- Should we have $f(\boldsymbol{x}) = +1$? (f is measuring symmetry)
- Should we have $f(\boldsymbol{x}) = -1$? (f only cares about the upper left pixel)
- Which is correct? – We cannot rule out either possibility

Outside the data set

- Any easy visual learning problem just got very messy

  - For every f that fits the data and is "+1" on the new point, there is one that is "−1"

  - Since f is unknown, it can take on any value outside the data, no matter how large the data is



- This is called No Free Lunch (NFL). You cannot know anything for sure about f outside the data without making assumptions

$\mathbf{x}_n$	y_n
0 0 0	○
0 0 1	●
0 1 0	●
0 1 1	○
1 0 0	●

$\mathscr{D}$

- f is a boolean function on 3 Boolean inputs, there are only $2^{2^3} = 256$ distinct boolean functions on $\mathscr{D}$
- Since f is unknown, any boolean function that agrees with* $\mathscr{D}$ could conceivable be f
- It does't matter what $\mathscr{H}$ is used or what algorithm is used. It is simply because you choose g based on $\mathscr{D}$

	$\mathbf{x}$	y	g	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
$\mathscr{D}$	0 0 0	○	○	○	○	○	○	○	○	○	○
	0 0 1	●	●	●	●	●	●	●	●	●	●
	0 1 0	●	●	●	●	●	●	●	●	●	●
	0 1 1	○	○	○	○	○	○	○	○	○	○
	1 0 0	●	●	●	●	●	●	●	●	●	●
	1 0 1		?	○	○	○	○	●	●	●	●
	1 1 0		?	○	○	●	●	○	○	●	●
	1 1 1		?	○	●	○	●	○	●	○	●

- Can we say anything about the f based on $\mathscr{D}$?

*这种表现与训练集上一致的函数所构成的集合，有的书上称为“版本空间”（version space）

Expected risk/loss/cost (Generalization error, Out-of-sample error)

- Expected Risk $R[h]$, $E_{out}(h)$

- How well we expect to do (on average) over the entire (admittedly known) source joint distribution $p_{X,Y}(x, y)$?

- The expected risk $R[h]$ of a hypothesis h on that distribution, measures the performance of this hypothesis by evaluating its expected loss over pairs (x, y) drawn from the distribution

$$R[h] = \mathbb{E}_{(X,Y)\sim p_{X,Y}}[\text{loss}(h(x), y)] = \sum_{X,Y} p(x, y)\text{loss}(h(x), y)$$

- Note that the randomness comes from the data distribution…

- Other terms with the same meaning are *expected loss, generalization error, or source-distribution risk*

Connecting in-sample to out-of-sample: population mean from sample mean

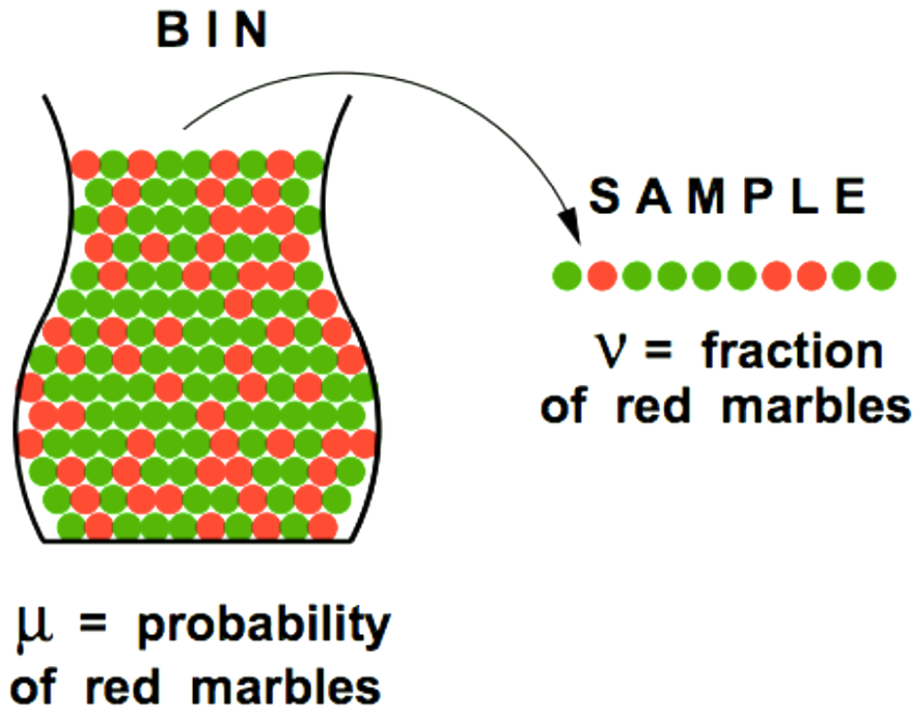
- Consider a "bin" with red and green marbles

- P(picking a red marble) = μ
  P(picking a green marble) = 1–μ

- The value of μ is unknown to us

- We pick m marbles independently

- The fraction of red marbles in sample = ν



Population mean from sample mean

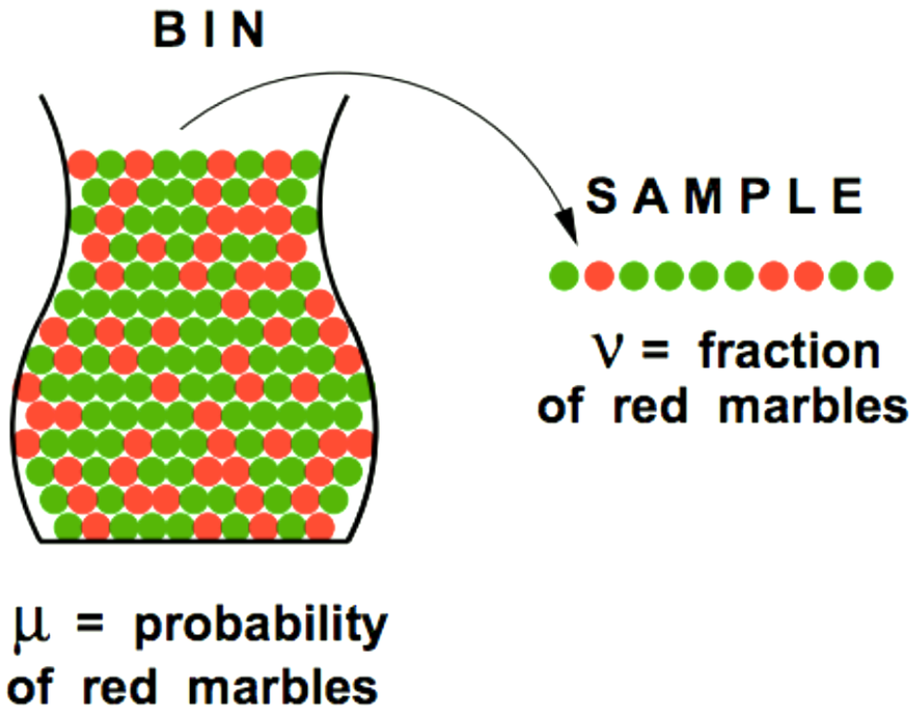
- The bin model
  sample —> the data set —> ν
  bin —> outside the data —> μ

- Can we say anything about μ (outside the data) after observing ν (the data)?

  Answer: No. It is possible for the sample to be all green marbles and the bin to be mostly red

- Then, why do we trust polling (e.g. audience measurement)

  Answer: The bad case is possible, but not probable.



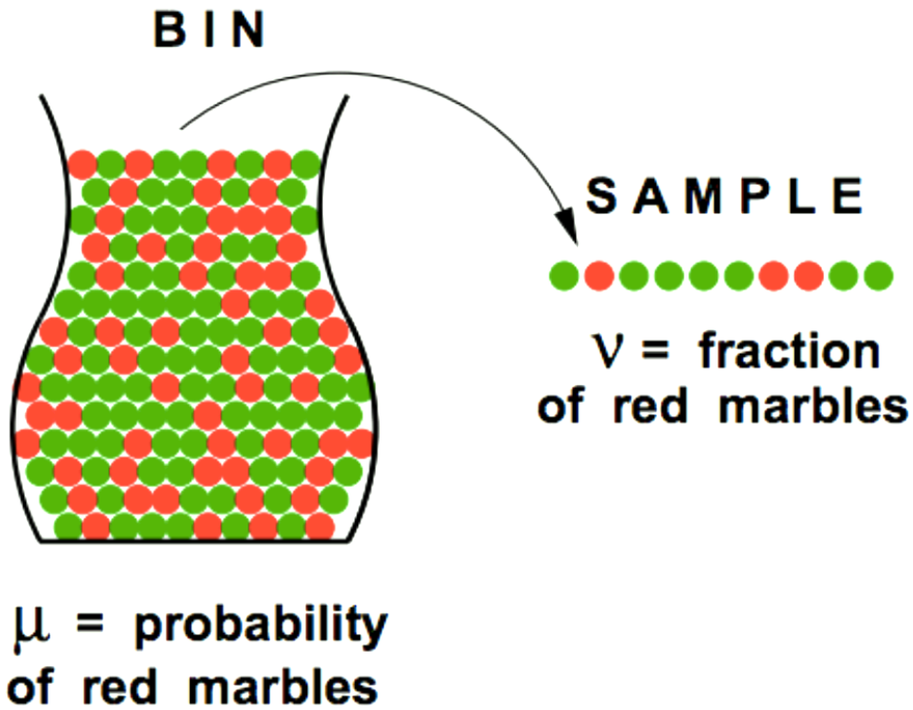
Does ν say anything about μ?

- No!

  Sample can be mostly green while bin is mostly red

- Yes!

  Sample frequency ν is likely close to bin frequency μ



Does ν say anything about μ? Probability to rescue

Law of Large Numbers

In a big sample (large m), ν is probably close to μ (within ϵ), i.e., most of the time, ν cannot be too far from μ

Formally,

$$\begin{cases} \mathbb{P}[|\nu - \mu| > \epsilon] \leq 2e^{-2\epsilon^2 m}, & \text{for any } \epsilon > 0 \\ \mathbb{P}[|\nu - \mu| \leq \epsilon] \geq 1 - 2e^{-2\epsilon^2 m}, & \text{for any } \epsilon > 0 \end{cases}$$

Looks familiar? This is called Hoeffding's Inequality

In other words, the statement $\mu = \nu$ is P.A.C. （probably approximately correct, **可能近似正确**）

We get to select any ϵ we want.

Probability to rescue

$$\begin{cases} \mathbb{P}[|\nu - \mu| > \epsilon] \leq 2e^{-2\epsilon^2 m}, & \text{for any } \epsilon > 0 \\ \mathbb{P}[|\nu - \mu| \leq \epsilon] \geq 1 - 2e^{-2\epsilon^2 m}, & \text{for any } \epsilon > 0 \end{cases}$$

Example

$m = 1000$; draw a sample and observe ν

99% of the time $\mu - 0.05 \leq \nu \leq \mu + 0.05 (\epsilon = 0.05)$

99.9999996% of the time $\mu - 0.10 \leq \nu \leq \mu + 0.10 (\epsilon = 0.10)$

What does this mean? If I repeatedly pick a sample of size 1,000, observe ν and claim that

$$\mu \in [\nu - 0.05, \nu + 0.05]$$

I will be right 99% of the time. On any particular sample you may be wrong, but not often!

$\Rightarrow$ We learned *something*. From ν, we reached outside the data to μ.

How did probability rescue us?

- Key ingredient: samples must be *independent*
  If the sample is constructed in some arbitrary fashion, then indeed we cannot say anything.
  Even with independence, ν can take on arbitrary values; but some values are way more likely than others

  This is what allows us to learn *something* – it is likely that $\nu \approx \mu$

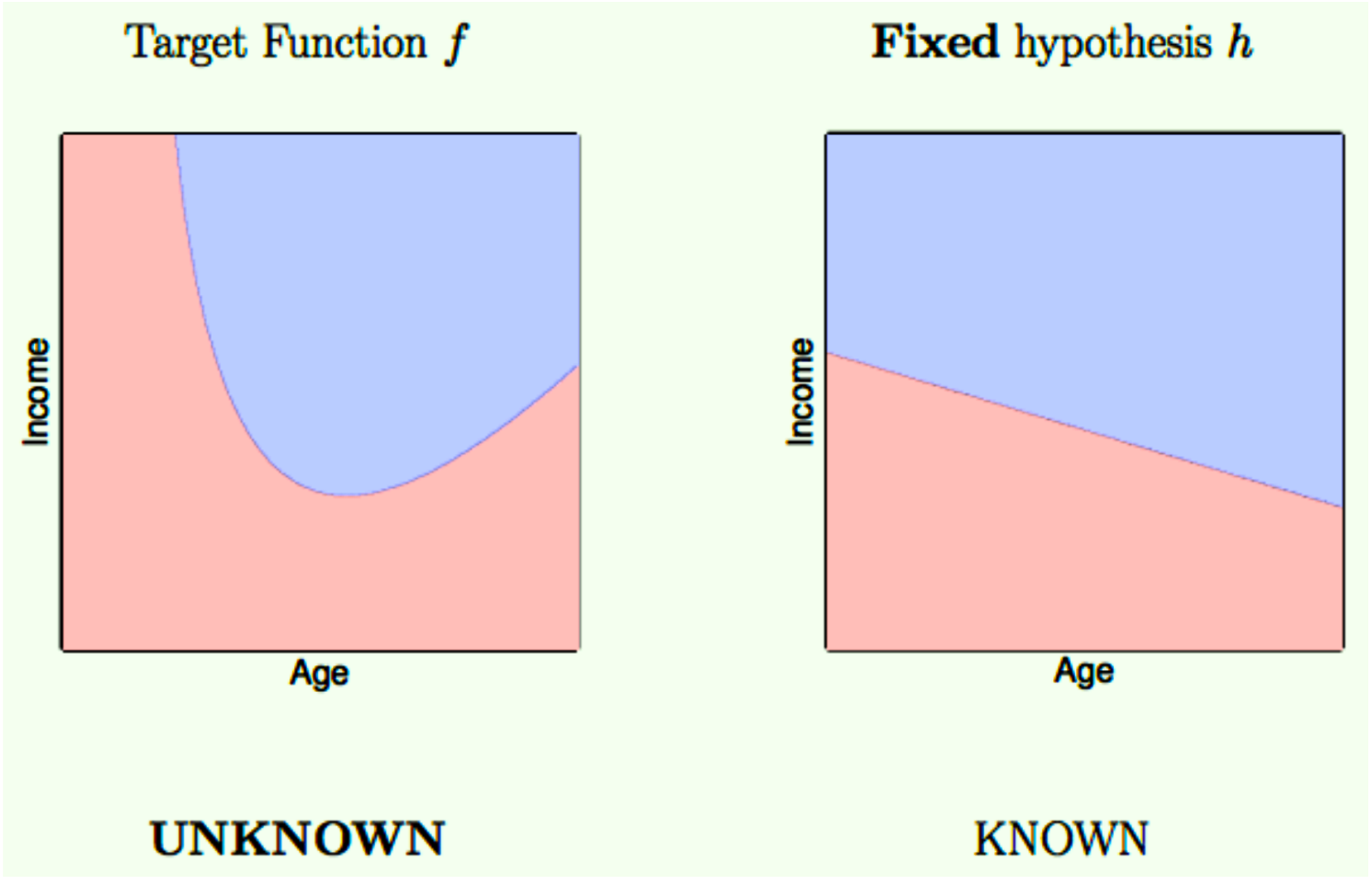
- The bound $2e^{-2\epsilon^2 m}$ does not depend on μ or the size of the bin
  The bin can be infinite

  It's great that it does not depend on μ because μ is unknown

- The key player in the bound $2e^{-2\epsilon^2 m}$ is m
  If $m \to \infty$, $\nu \approx \mu$ with very very very ... high probability, but not for sure

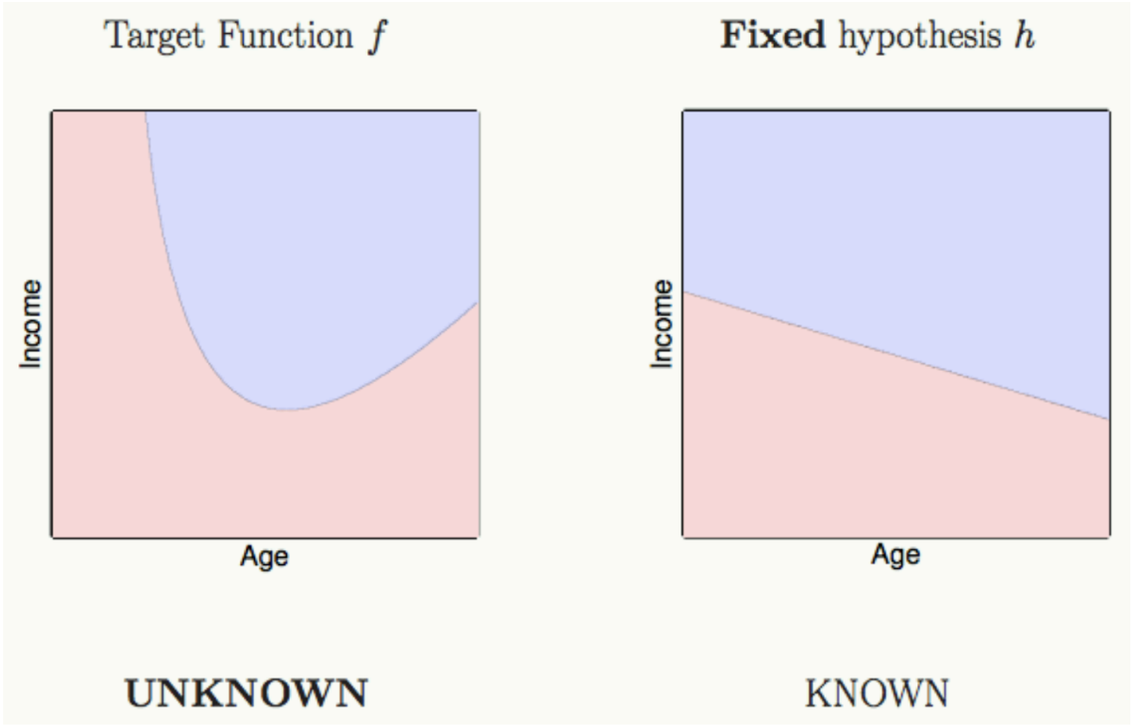
  Can you live with 10^{-100} probability error?

How did probability rescue us?



- In learning, the unknown is an entire function f; in the bin it was a single number μ.

Relating the bin to learning

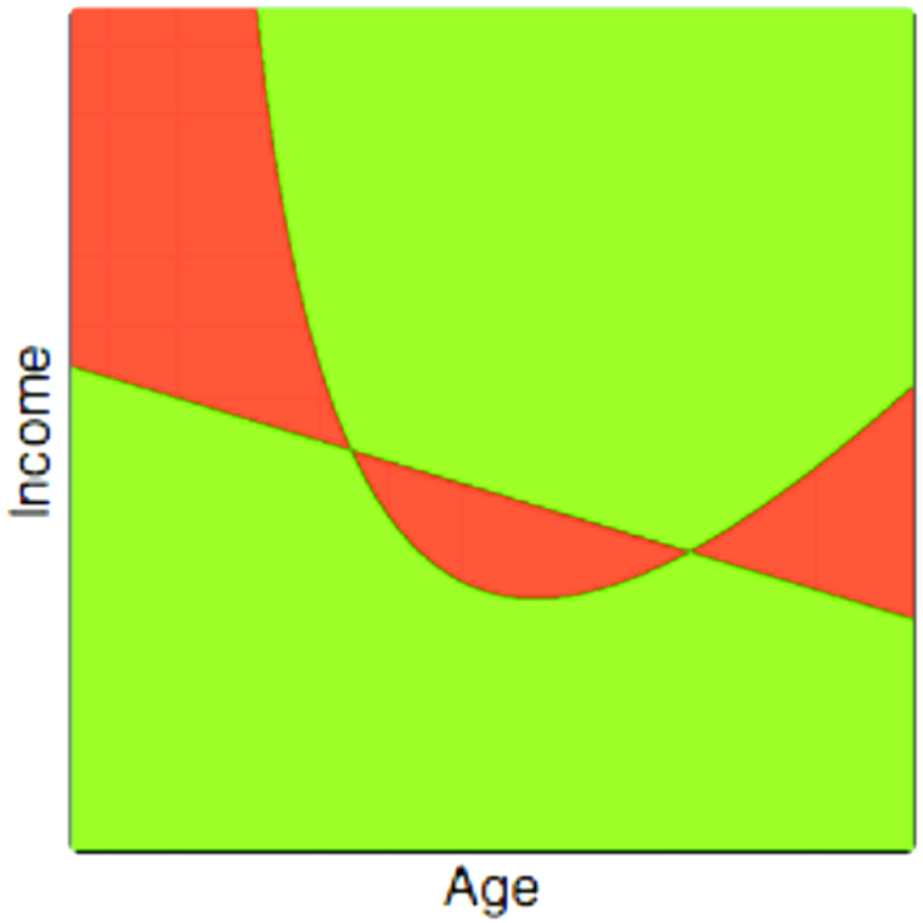


Green: $h(x) = f(x)$

Red: $h(x) \neq f(x)$

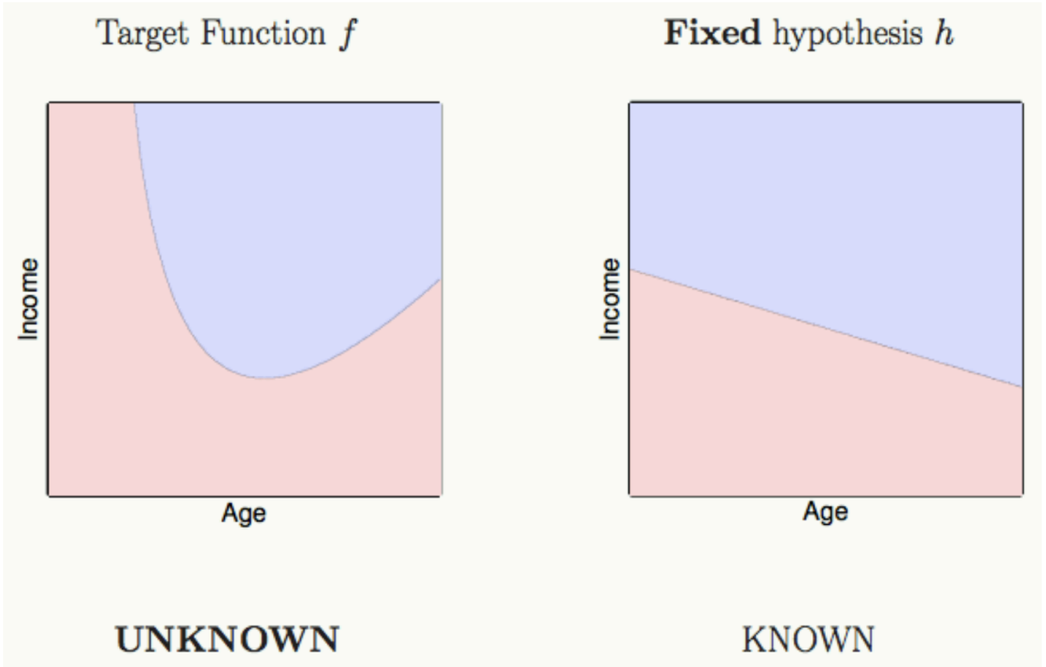
$E_{out}(h) = \mathbb{P}_X[h(x) \neq f(x)]$

$= \mathbb{E}_X[h(x) \neq f(x)]$



UNKNOWN

Relating the bin to learning



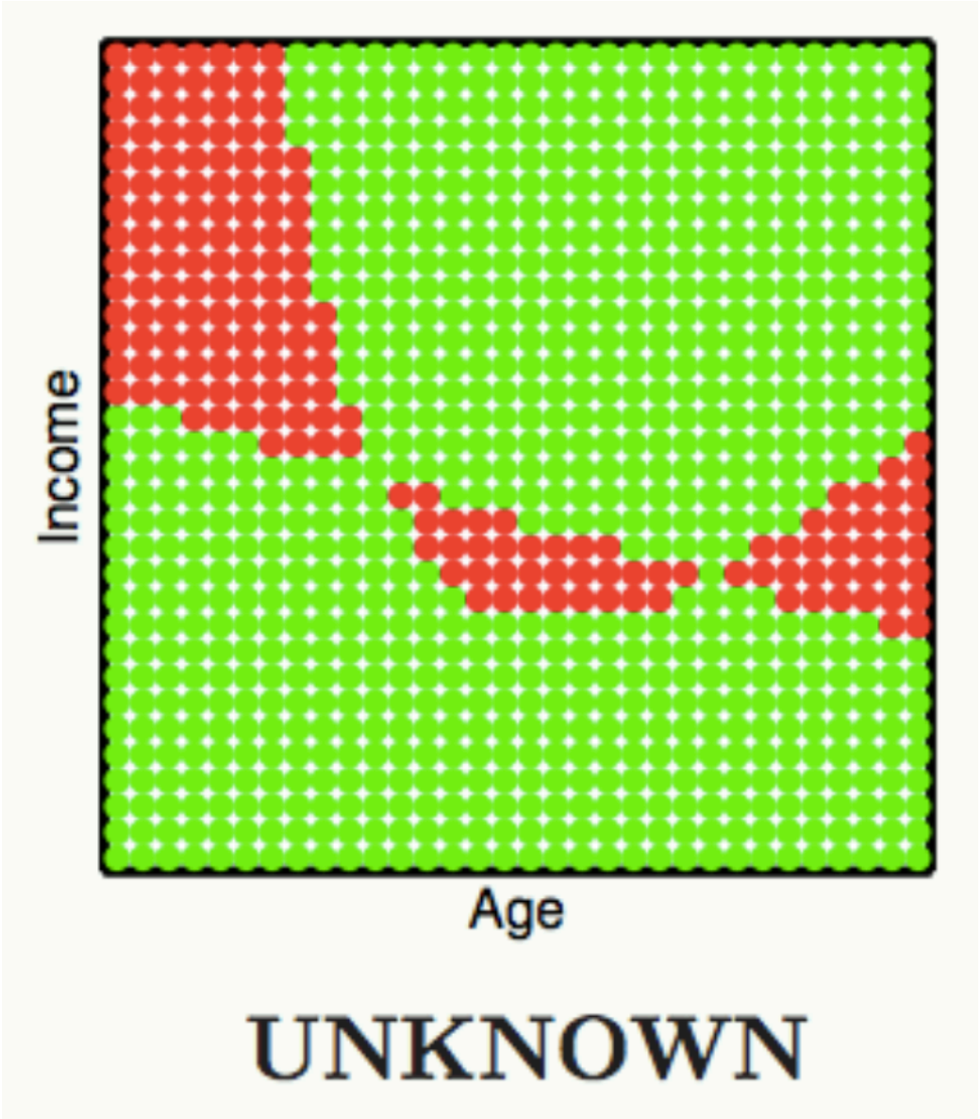
Green "marbles": $h(x) = f(x)$

Red "marbles": $h(x) \neq f(x)$

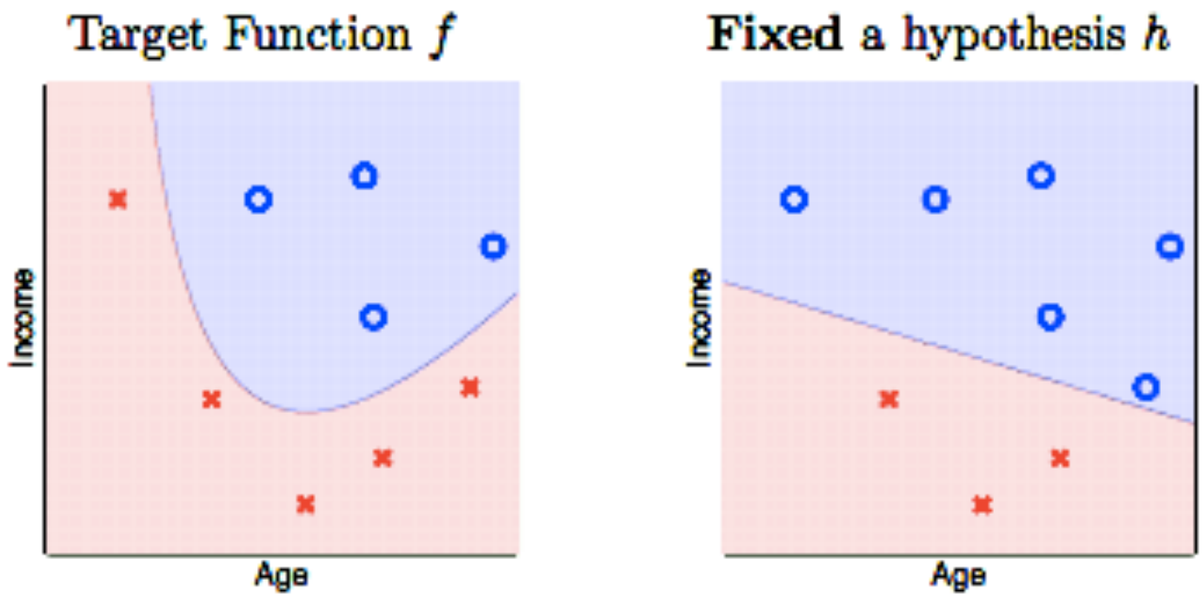
Bin: $\mathcal{X}$

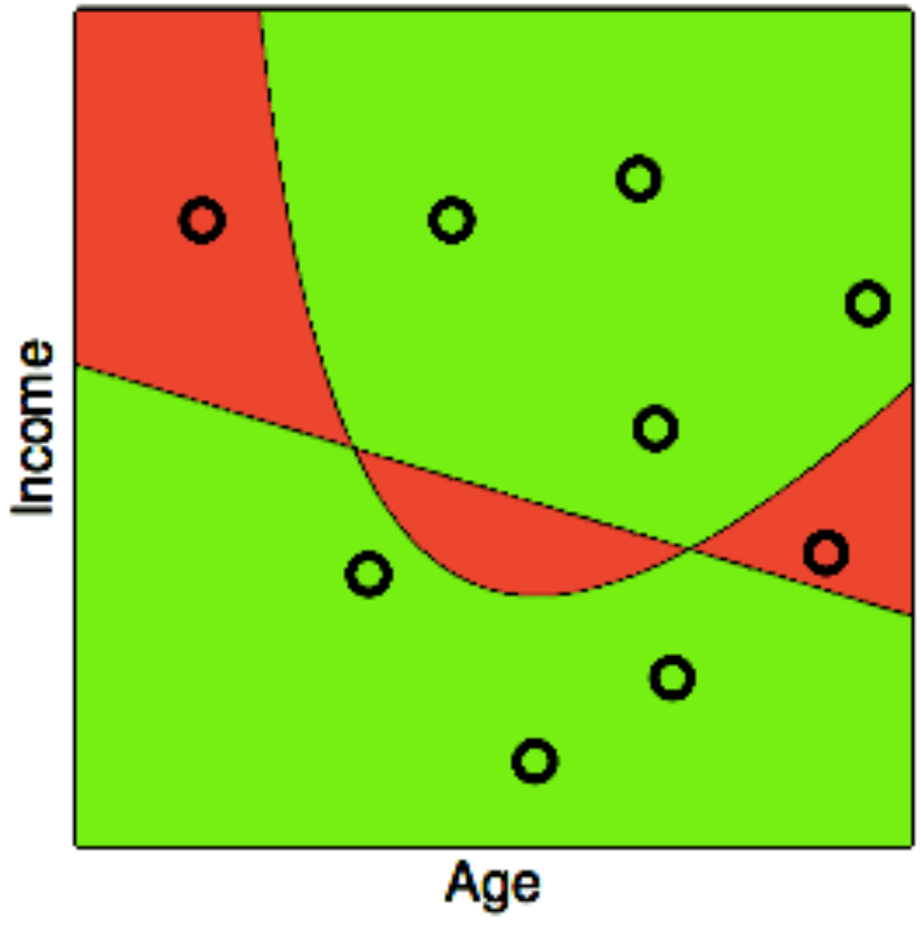
$E_{out}(h) = \mathbb{P}_X[h(x) \neq f(x)]$

(out-of-sample error)

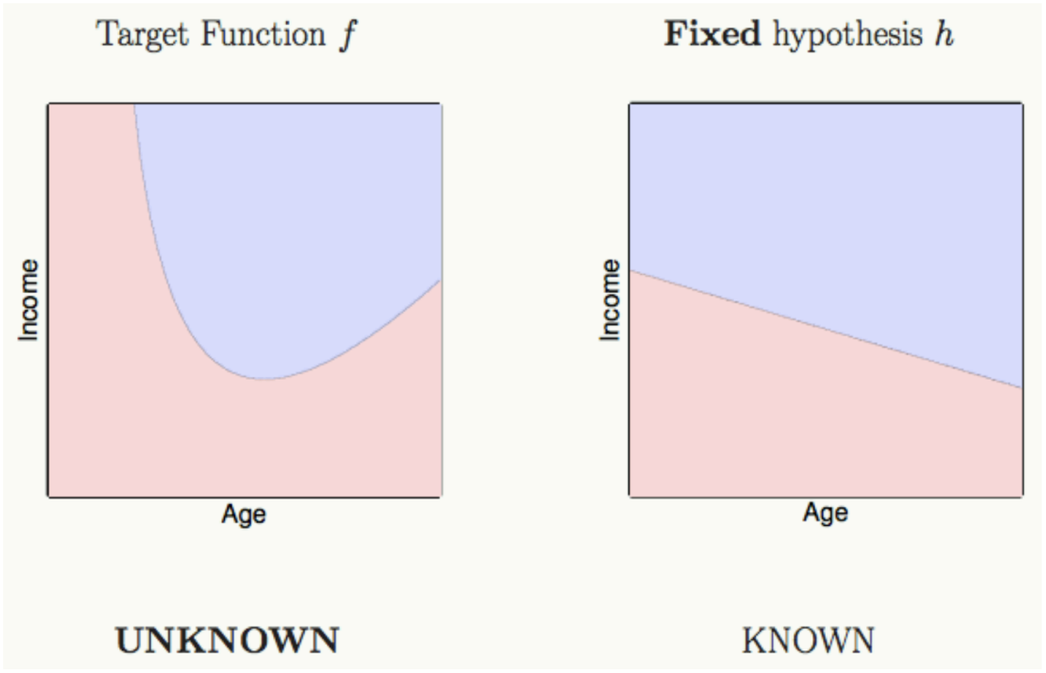


Relating the bin to learning - the data (sample)





Relating the bin to learning



Green data: $h(\mathbf{x}_i) = f(\mathbf{x}_i)$

Red "marbles": $h(\mathbf{x}_i) \neq f(\mathbf{x}_i)$

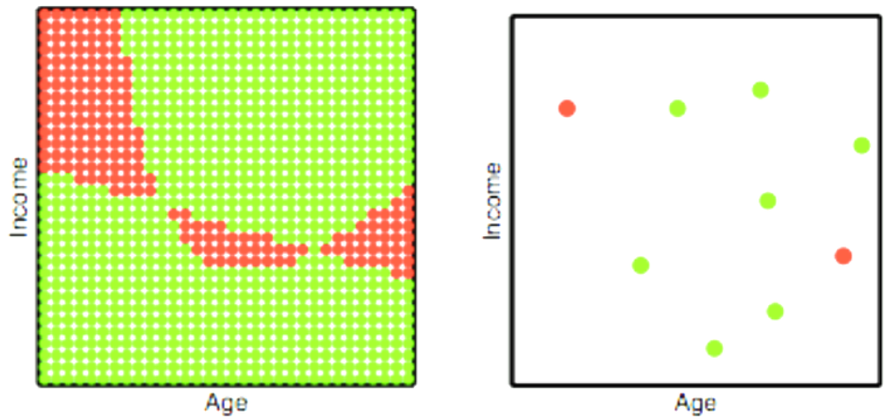
Bin: $\mathcal{X}$

$E_{in}(h)=$ fraction of red data

red data: misclassified

KNOWN!

Relating the bin to learning



Unknown f and $P(\mathbf{x})$, fixed h

Learning

input space $\mathcal{X}$

$\mathbf{x}$ for which $h(\mathbf{x}) = f(\mathbf{x})$

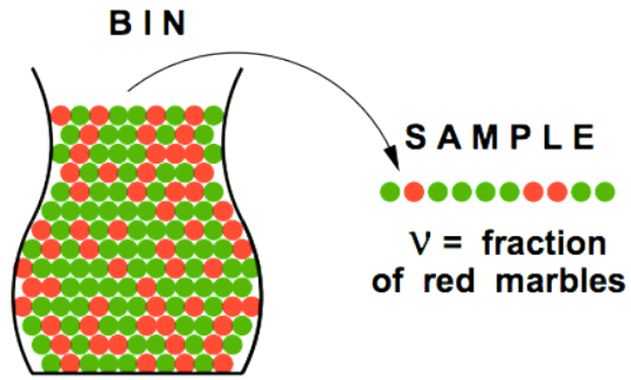
$\mathbf{x}$ for which $h(\mathbf{x}) \neq f(\mathbf{x})$

$\mathbb{P}(\mathbf{x})$

data set $\mathcal{D}$

Out-of-sample error: $E_{out}(h) = \mathbb{P}[h(\mathbf{x}) \neq f(\mathbf{x})]$

In-sample error: $E_{in}(h) = \frac{1}{m}\sum_{i=1}^{m} \mathbb{I}[h(\mathbf{x}_i) \neq f(\mathbf{x}_i)]$



μ = probability
of red marbles

Bin

● green marble

● red marble

randomly picking a marble

sample of m marbles

μ = probability of picking a red marble

ν = fraction of red marbles in the sample

$E_{out}(h)$ and $E_{in}(h)$, Which one is random? Which one is fixed?

Hoeffding says that $E_{in}(h) \approx E_{out}(h)$

$$\begin{cases} \mathbb{P}[|\nu - \mu| > \epsilon] \leq 2e^{-2\epsilon^2 m}, & \text{for any } \epsilon > 0 \\ \mathbb{P}[|\nu - \mu| \leq \epsilon] \geq 1 - 2e^{-2\epsilon^2 m}, & \text{for any } \epsilon > 0 \end{cases}$$

E_{in} is random, but known; E_{out} fixed, but unknown

- If $E_{in} \approx 0 \rightarrow E_{out} \approx 0$ (with high probability), i.e., $\mathbb{P}_X[h(\mathbf{x}) \neq f(\mathbf{x})] \approx 0$

  We have learned something about the entire f: $f \approx h$ over $\mathcal{X}$ (outside $\mathcal{D}$)
- If $E_{in} \gg 0$, we're out of luck

  But, we have still learned something about the entire f: $f \neq h$; it is not very useful though

This process is verification, not real learning

The entire previous argument assumed a **FIXED** h and then came the data

- Given $h \in \mathcal{H}$, a sample can **verify** whether or not it is good (w.r.t. f):
  if E_{in} is small, h is good, with high confidence
  if E_{out} is large, h is bad with high confidence
  We have no control over E_{in}. It is what it is.

- In learning, you actually try to **fit** the data, as with the LS model searching an entire hypothesis set $\mathcal{H}$ for a hypothesis with small E_{in}

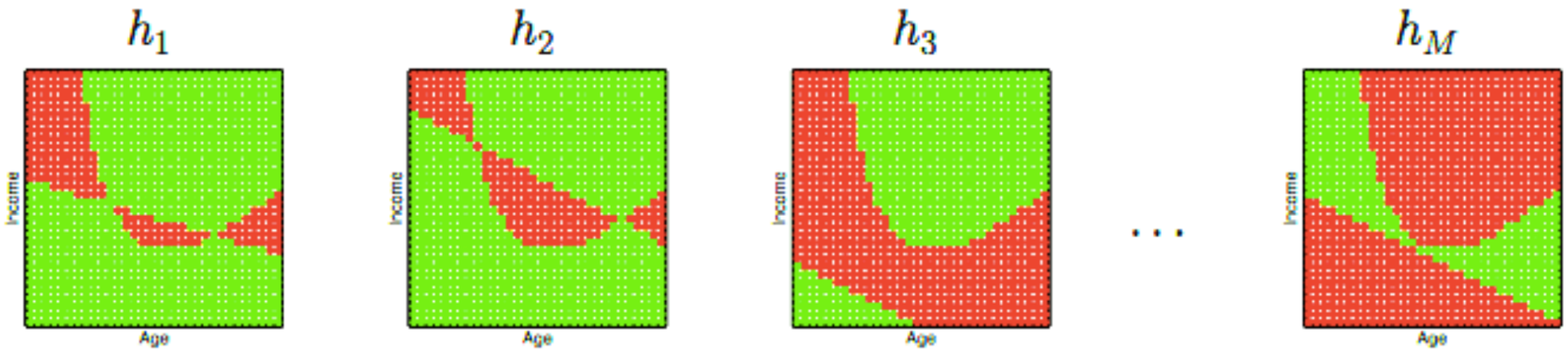
Verification vs learning

Verification	Real learning
Fixed single hypothesis h	Fixed hypothesis set $\mathcal{H}$
h to be certified	g to be selected
h does not depend on $\mathcal{D}$	g results after searching $\mathcal{H}$ to fit $\mathcal{D}$
No control over E_{in}	Pick best E_{in}

Verification: we can say something outside the data about h?

Learning: can we say something outside the data about g?

Real learning - finite learning model



Pick the hypothesis with minimum E_{in}; will E_{out} be small?

Experiment: selecting the best coin

- Everyone takes out a coin
- Each toss your coin 5 times and count the number of heads
- Some guy got the smallest number of heads (probably 0)?
- Let's toss this coin (this coin has never come up heads)

Do we expect P[HEADS] ≈ 0??

Selection bias (林子大了，什么鸟都有)

Coin tossing example:

- If we toss one coin and get no HEADS, its very surprising ($\mathbb{P} = \frac{1}{2^5}$)

- Tossing 70 coins, and find one with no heads. Is it surprising?
  ($\mathbb{P} = 1 - \left(1 - \frac{1}{2^5}\right)^{70} \approx 1$)
  Do we expect $\mathbb{P}[heads] \approx 0$ for the selected coin?

- This is called selection bias
  Selection bias is a very serious trap

If we select an $h \in \mathcal{H}$ with smallest E_{in}, can we expect E_{out} to be small?

Search Causes Selection Bias

Selection bias

Selection bias is

- the selection of individuals, groups or data for analysis in such a way that proper randomization is not achieved
- thereby ensuring that the sample obtained is not representative of the population intended to be analyzed
- refers to the distortion of a statistical analysis, resulting from the method of collecting samples
- if the selection bias is not taken into account, then some conclusions of the study may not be accurate

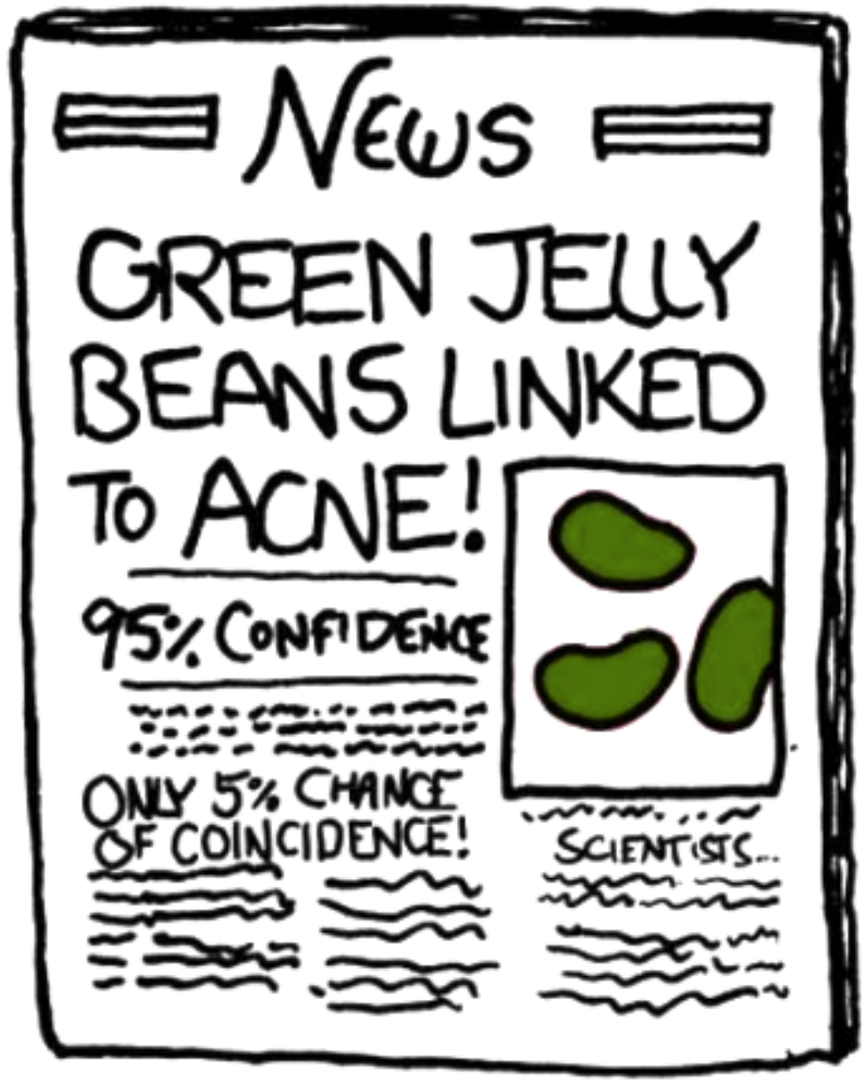
Professor's statement

Jelly Beans Cause Acne?

Jelly Beans Cause Acne?

Jelly Beans Cause Acne?

Jelly Beans Cause Acne?



What we know so far?

- Verification: if (randomized) sample size m is big enough, what we've learned on $\mathcal{D}$ can be extended to test set:

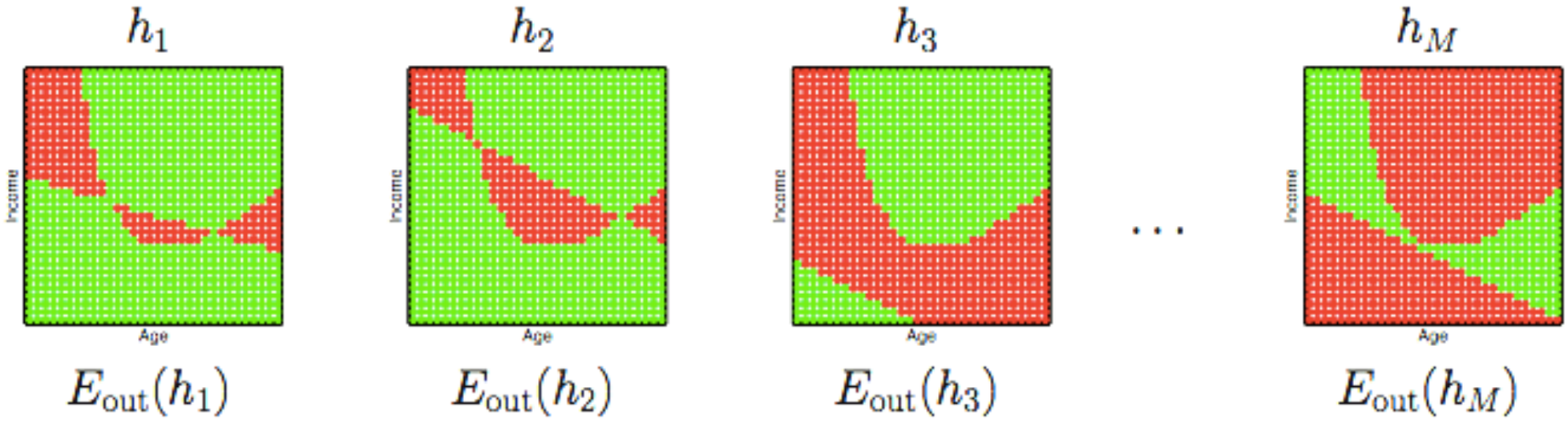
- Formal guarantee:
  for any particular h, in "big" data (m large)

  in-sample error $E_{in}(h)$ is probably close to
  out-of-sample error $E_{out}(h)$ (within ϵ)

$$\mathbb{P}[\ |E_{in}(h) - E_{out}(h)| > \epsilon\ ] \leq 2e^{-2\epsilon^2 m}$$

- We are talking about "extend what we learn", no guarantee "we've learned good information"

Real learning - finite learning model



Pick the hypothesis with minimum E_{in}; will E_{out} be small?

Hoeffding says that $E_{in}(g) \approx E_{out}(g)$ for finite $\mathscr{H}$

$$\begin{cases} \mathbb{P}[|E_{in}(g) - E_{out}(g)| > \epsilon] \leq 2|\mathcal{H}|e^{-2\epsilon^2 m}, & \text{for any } \epsilon > 0 \\ \mathbb{P}[|E_{in}(g) - E_{out}(g)| \leq \epsilon] \geq 1 - 2|\mathcal{H}|e^{-2\epsilon^2 m}, & \text{for any } \epsilon > 0 \end{cases}$$

We don't care how g was obtained, as long as it is from $\mathcal{H}$

g depends on $\mathscr{D}$

Proof.

Let $M = |\mathcal{H}|$. The event "$|E_{in}(g) - E_{out}(g)| > \epsilon$" implies
"$|E_{in}(h_1) - E_{out}(h_1)| > \epsilon$" OR ... OR "$|E_{in}(h_M) - E_{out}(h_M)| > \epsilon$"
So by the implication and union bounds:

$$\begin{aligned} &\mathbb{P}[|E_{in}(g) - E_{out}(g) > \epsilon] \\ \leq &\mathbb{P}[|E_{in}(h_1) - E_{out}(h_1)| > \epsilon] + \ldots + \mathbb{P}[|E_{in}(h_M) - E_{out}(h_M)| > \epsilon] \\ \leq &2e^{-2\epsilon^2 m} + \ldots + 2e^{-2\epsilon^2 m} \\ = &2Me^{-2\epsilon^2 m} \end{aligned}$$

□

Trade-off on $M = |\mathscr{H}|$

Two goals:

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! Hoeffding's inequality
2. No! too few choices

large M

1. No! Hoeffding's inequality
2. Yes! many choices

Using the right M (or $\mathcal{H}$) is important

Infinite hypothesis set

Known:

$$\mathbb{P}[\ |E_{in}(g) - E_{out}(g)| > \epsilon] \leq 2M\ e^{-2\epsilon^2 m}$$

Want:

- establish a finite quantity that replaces M

$$\mathbb{P}[\ |E_{in}(g) - E_{out}(g)| > \epsilon] \leq 2M_{\mathcal{H}}\ e^{-2\epsilon^2 m}$$

- justify the feasibility of learning for infinite M
- study $M_{\mathcal{H}}$ to understand its trade-off for "right" $\mathcal{H}$, just like M

Why we have M in the inequality?

$$\mathbb{P}[\ |E_{in}(g) - E_{out}(g)| > \epsilon] \leq 2M\ \exp(-2\epsilon^2 m)$$

- Bad events $\mathcal{B}_i$: $|E_{in}(h_i) - E_{out}(h_i)| > \epsilon$
- to give freedom of choice: bound $\mathbb{P}[\mathcal{B}_1\ or\ \mathcal{B}_2\ or ... \mathcal{B}_M]$
- worst case: all $\mathcal{B}_i$ non-overlapping (union bound)

$$\mathbb{P}[\mathcal{B}_1\ or\ \mathcal{B}_2\ or ... \mathcal{B}_M] \leq \mathbb{P}[\mathcal{B}_1] + \mathbb{P}[\mathcal{B}_2] + ... + \mathbb{P}[\mathcal{B}_M]$$

Where did uniform bound fail?

union bound $\mathbb{P}[\mathcal{B}_1] + \mathbb{P}[\mathcal{B}_2] + ... + \mathbb{P}[\mathcal{B}_M]$

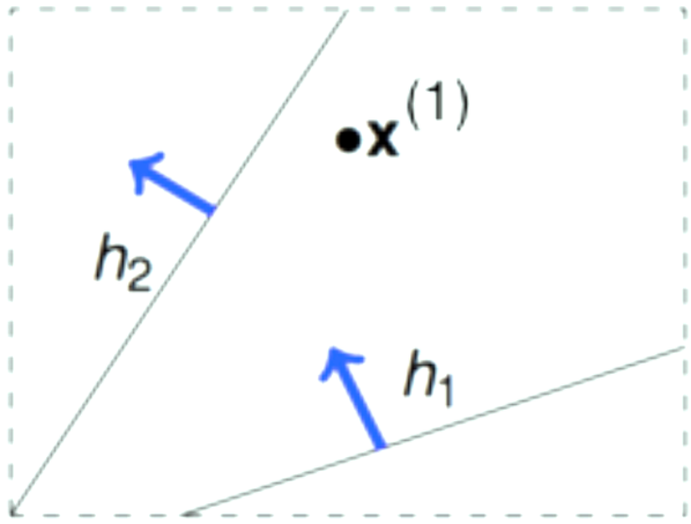
- Bad events $\mathcal{B}_i$: $|E_{in}(h_i) - E_{out}(h_i)| > \epsilon$
  overlapping for similar hypotheses $h_1 \approx h_2$
- why?
  - $E_{out}(h_1) \approx E_{out}(h_2)$
  - for most $\mathcal{D}$, $E_{in}(h_1) \approx E_{in}(h_2)$
- union bound *over-estimating*

to account for overlap, can we group similar hypotheses by kind?

How many lines are there?

$$\mathcal{H} = \{\text{all lines in } \mathbb{R}^2\}$$

- how many lines? ∞
- how many kinds of lines if viewed from one input vector $\mathbf{x}^{(1)}$?



$\rightarrow$ 2 kinds: h_1-like$(\mathbf{x}^{(1)}) = +1$ or h_2-like $(\mathbf{x}^{(1)}) = -1$

Effective number of lines

maximum kinds of lines with respect to m inputs $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(m)}$
$\iff$ effective number of lines ($M_{\mathcal{H}}$)

- must be $\leq 2^m$ (why?)
- finite "grouping" of infinitely-many lines $\in \mathcal{H}$
- wish:

m	$M_{\mathcal{H}}$
1	2
2	4
3	8
4	14 ($< 2^m$)

$$\mathbb{P}[\,|E_{in}(g) - E_{out}(g)| > \epsilon] \leq 2\, M_{\mathcal{H}}\, e^{-2\epsilon^2 m}$$

If $M_{\mathcal{H}}$ can replace M and $M_{\mathcal{H}} \ll 2^m$....
learning possible with infinite lines...

- Simple $f \implies$ can use small $\mathcal{H}$ to get $E_{in} \approx 0$ (need smaller m)
- Complex $f \implies$ need large $\mathcal{H}$ to get $E_{in} \approx 0$ (need larger m)

3、可学性

- 期望风险极小化
- NFL定理
- 可学性

Additional property of expected risk

- A predictor h is "good" on a particular source joint distribution if it has low risk $R[h]$ on that distribution

- The expected risk $R[h]$ is the probability that the predictor h will incorrectly predict the label for any pair (x, y) drawn at random from the source joint distribution:

$$R[h] = \mathbb{E}_{(X,Y)\sim p_{X,Y}}[\text{loss}(h(x), y)] = \mathbb{P}_{(X,Y)\sim p_{X,Y}}\{h(X) \neq Y\}$$

- This equivalence between the risk and the probability of incorrect label prediction holds only for this 0,1 loss

- We will be assuming that the source joint distribution is fixed, samples are i.i.d.

The Learning Process (Ideal Case)

Learning from a function from examples! Given

- Domain $\mathscr{X}$, $\mathscr{Y}$
- The target function f. (unknown)
- $\mathscr{H}$: Hypothesis set; the set of all possible hypotheses
- Final hypothesis (your predictor/model): $g \approx f$
- Ideal case: g is obtained by minimizing $R[h]$

$$g = \arg\min_{h} \; R[h], \quad \text{subject to: } h \in \mathscr{H}$$

Goal of Machine Learning

- The core of machine learning deals with representation and generalization:
- Representation (Explanation) of data instances and functions evaluated on these instances are part of all machine learning systems
- Generalization (Prediction) is the property that the system will perform well on unseen data instances

Expected risk minimizer

$$g = \arg\min_{h \in \mathcal{H}} \quad \mathbb{E}_{(X,Y)\sim p_{X,Y}}[\text{loss}(h(x), y)]$$

- We want to find a predictor that minimizes the expected loss on the true joint distribution, but…
- We do not have complete knowledge of the true source joint distribution
- We should choose our predictor to minimize the expected loss on what we do have access to
- We hope that this predictor will do well on the true source joint distribution (However…)

Empirical risk minimizer

- The expected loss of a predictor h on a particular observed sample data set $\mathscr{D} = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ could also be referred to as the empirical risk

$$\hat{R}_{\mathcal{D}}[h] = \frac{1}{m} \sum_{i=1}^{m} [\text{loss}(h(x_i), y_i)]$$

- Usually, the target predictor is found by

$$\hat{h} = \arg\min_{h} \ \hat{R}_{\mathcal{D}}[h], \qquad h \in \mathcal{H}$$

- Parameterize $h(\,\cdot\,;\theta) \iff \theta$

$$\hat{\theta} = \arg\min_{\theta} \ \hat{R}_{\mathcal{D}}[\theta], \qquad \theta \in \ \Theta$$

Expected Risk Min v.s. Empirical Risk Min

- Expected Risk Minimization

$$\min_{h} \ \mathbb{E}[\text{loss}(h(x), y)] \quad \text{s.t. } h \in \mathcal{H} \quad \rightarrow \quad \text{parameterize...}$$

$$\min_{\theta} \ \mathbb{E}[\text{loss}(h(x;\theta), y)] \qquad \text{s.t. } \theta \in \Theta$$

- Empirical Risk Minimization

$$\min_{h} \ \frac{1}{m}\sum_{i=1}^{m}[\text{loss}(h(x_i), y_i)] \ \text{s.t. } h \in \mathcal{H} \quad \rightarrow \quad \text{parameterize...}$$

$$\min_{\theta} \ \frac{1}{m}\sum_{i=1}^{m}[\text{loss}(h(x_i;\theta), y_i)] \qquad \text{s.t. } \theta \in \Theta$$

NFL (no free lunch) Theorem:（无万金油算法定理）

- $P(h \mid X, \mathscr{A})$: the probability of finding hypothesis h when applying learning algorithm $\mathscr{A}$ on training set $X \subset \mathscr{X}$
- f is the target function, $X \subset \mathscr{X}$ is noise-free
- Consider the 0-1 error
- Purpose: compare the performance of different $\mathscr{A}$ on all tasks f

The out-of-sample error of $\mathscr{A}$ on X for task f is given by:

$$E(\mathscr{A} \mid X, f) = \sum_{h} \sum_{\boldsymbol{x} \in \mathscr{X} - X} P(\boldsymbol{x}) \mathbb{I}(h(\boldsymbol{x}) \neq f(\boldsymbol{x})) P(h \mid X, \mathscr{A})$$

---

1. D. H. Wolpert, "The Lack of A Priori Distinctions Between Learning Algorithms," in *Neural Computation*, vol. 8, no. 7, pp. 1341-1390, Oct. 1996.
2. David H. Wolpert and William G. Macready, “No Free Lunch Theorems for Optimization”, *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, VOL. 1, NO. 1, APRIL 1997
3. https://www.kdnuggets.com/2019/09/no-free-lunch-data-science.html

$$E(\mathscr{A}\,|\,X,f)=\sum_{h\in\mathscr{H}}\sum_{\boldsymbol{x}\in\mathscr{X}-X}P(\boldsymbol{x})\mathbb{I}(h(\boldsymbol{x})\neq f(\boldsymbol{x}))P(h\,|\,X,\mathscr{A})$$

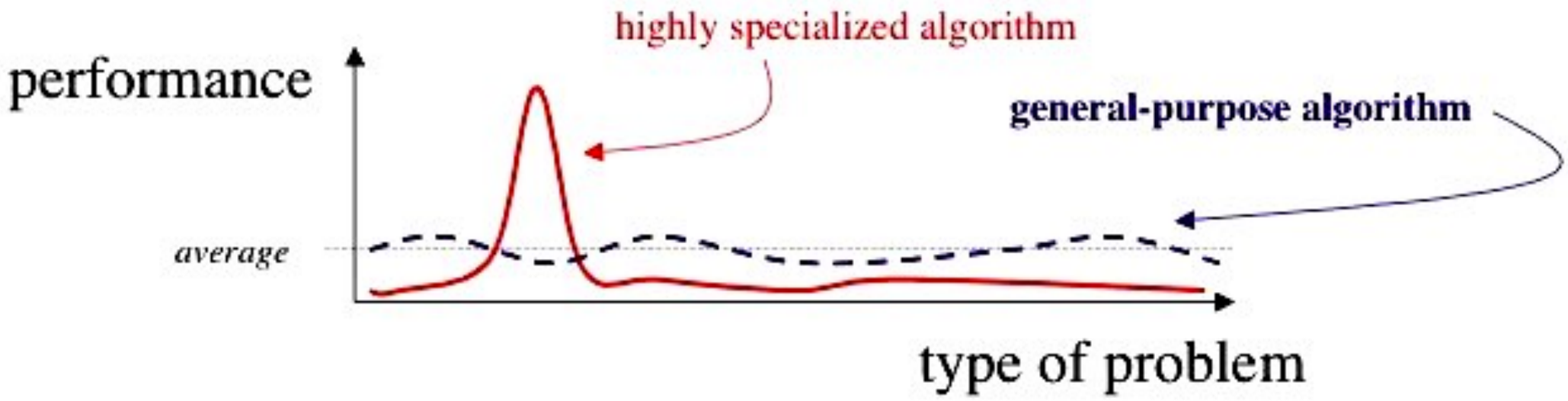
If you apply $\mathscr{A}$ to any problem (any f), the overall out-of-sample error of $\mathscr{A}$ for all f is given by:

$$\begin{aligned}
\sum_{f}E(\mathscr{A}\,|\,X,f)&=\sum_{f}\sum_{h\in\mathscr{H}}\sum_{\boldsymbol{x}\in\mathscr{X}-X}P(\boldsymbol{x})\mathbb{I}(h(\boldsymbol{x})\neq f(\boldsymbol{x}))P(h\,|\,X,\mathscr{A})\\
&=\sum_{h\in\mathscr{H}}\sum_{\boldsymbol{x}\in\mathscr{X}-X}P(\boldsymbol{x})P(h\,|\,X,\mathscr{A})\sum_{f}\mathbb{I}(h(\boldsymbol{x})\neq f(\boldsymbol{x}))\\
&=\sum_{\boldsymbol{x}\in\mathscr{X}-X}P(\boldsymbol{x})\sum_{h\in\mathscr{H}}P(h\,|\,X,\mathscr{A})\sum_{f}\mathbb{I}(h(\boldsymbol{x})\neq f(\boldsymbol{x}))\\
&=\sum_{\boldsymbol{x}\in\mathscr{X}-X}P(\boldsymbol{x})\sum_{h\in\mathscr{H}}P(h\,|\,X,\mathscr{A})\frac{1}{2}2^{|\mathscr{X}|}\\
&=\frac{1}{2}2^{|\mathscr{X}|}\sum_{\boldsymbol{x}\in\mathscr{X}-X}P(\boldsymbol{x})\cdot 1
\end{aligned}$$

Independent of $\mathscr{A}$

Question:

- What does NFL theorem tell us?



- Then why should we trust machine learning? or any algorithms?

What does “learnability” mean? (可学性)

Definition: Consider a concept class C defined over a set of instances X of length n and a learner L using hypothesis space H. C is **PAC-learnable** by L using H if for all $c \in C$, distributions $\mathcal{D}$ over X, ϵ such that $0 < \epsilon < 1/2$, and δ such that $0 < \delta < 1/2$, learner L will with probability at least $(1 - \delta)$ output a hypothesis $h \in H$ such that $error_{\mathcal{D}}(h) \leq \epsilon$, in time that is polynomial in $1/\epsilon$, $1/\delta$, n, and $size(c)$.

- 这里X是训练集，$\mathcal{D}$是分布，H是假设类，C是概念类，c是目标概念
- L是学习器（模型+算法）
- 虽然我们限制ϵ, δ的取值范围，但事实上我们只对充分小的取值情况才感兴趣

Mitchell, T. M. Machine learning, Chapter 7, 1997.

Training Set and Test Set

- A training set is a set data used to discover potentially predictive relationship.
- A test set is a set of data used to assess the strength and utility of a predictive relationship
- Random split

1	-1	5	7	14	19	39	40	51	63	67	73	74	76
2	-1	3	6	17	22	36	41	53	64	67	73	74	76
3	-1	5	6	17	21	35	40	53	63	71	73	74	76
4	-1	2	6	18	19	39	40	52	61	71	72	74	76
5	-1	3	6	18	29	39	40	51	61	67	72	74	76
6	-1	4	6	16	26	35	45	49	64	71	72	74	76
7	1	5	7	17	22	36	40	51	63	67	73	74	76
8	1	2	6	14	29	39	42	52	64	67	72	75	76
9	1	4	6	16	19	39	40	51	63	67	73	75	76
10	1	3	6	18	20	37	40	51	63	71	73	74	76
11	1	2	11	15	19	39	40	52	63	68	73	74	76
12	-1	1	6	15	19	39	42	55	62	67	72	74	76
13	-1	2	6	17	24	38	42	50	64	71	73	74	76
14	1	3	6	15	25	38	40	48	63	68	73	74	76
15	-1	1	7	16	22	36	42	56	62	67	73	74	76
16	-1	2	6	16	22	36	42	54	66	67	73	74	76
17	-1	3	6	14	21	35	40	50	63	67	73	74	76
18	1	4	7	18	29	39	41	51	66	67	72	74	76
19	1	3	6	16	32	39	40	52	63	67	73	74	76
20	-1	5	6	18	22	36	43	49	66	71	72	74	76
21	-1	3	9	14	26	35	40	56	63	71	73	74	76
22	1	5	10	17	19	39	40	47	63	67	73	74	76
23	-1	1	6	16	22	36	42	48	62	67	73	74	76
24	1	5	16	20	37	40	63	68	73	74	76	82	93
25	-1	3	6	18	22	36	41	51	64	67	73	74	76
26	-1	4	6	16	22	36	40	48	63	67	73	74	76
27	-1	1	10	16	24	38	42	59	64	67	73	74	76
28	-1	1	6	18	20	37	42	50	62	71	73	74	76
29	-1	4	6	18	19	39	41	51	62	67	73	74	77
30	-1	2	9	14	20	37	40	55	62	67	73	74	76
31	-1	1	11	18	20	37	40	49	63	71	73	74	76
32	-1	4	6	17	21	35	42	54	66	67	73	74	76
33	-1	1	6	17	20	37	42	54	62	67	73	74	76
34	-1	1	6	18	22	36	46	55	61	67	72	74	76
35	1	2	6	14	20	37	40	50	63	67	73	74	76
36	-1	4	7	18	24	38	40	52	63	67	73	74	76
37	-1	2	6	18	26	35	40	54	63	67	73	74	76