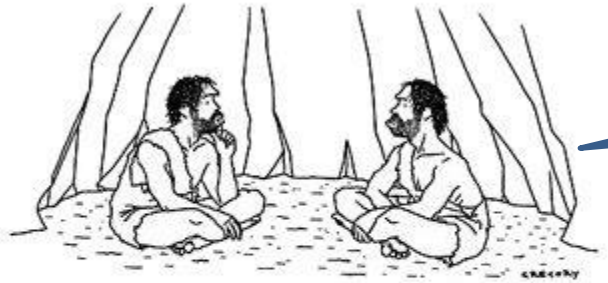


"Something's just not right—our air is clean, our water is pure, we all get plenty of exercise, everything we eat is organic and free-range, and yet nobody lives past thirty."

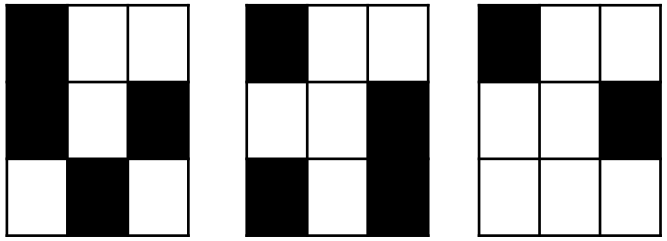
Learning general rules from experience

Fundamental of Inductive Learning:

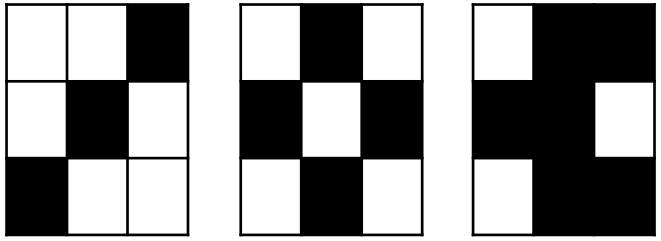
The Empirical Risk Minimization(ERM) rule

Is Learning Feasible?

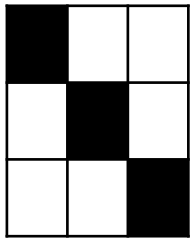
- Let us consider the following two examples:



f = -1



f = 1



f = ?

$f: \{0,1\}^3 \rightarrow \{0,1\}$

We know f only partially in its domain

0 0 0	0
0 0 1	1
0 1 0	1
0 1 1	0
1 0 0	1
1 0 1	?
1 1 0	?
1 1 1	?

How is f in the last 3 elements?

It is Not…

	g	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
0 0 0	0	0	0	0	0	0	0	0	0
0 0 1	1	1	1	1	1	1	1	1	1
0 1 0	1	1	1	1	1	1	1	1	1
0 1 1	0	0	0	0	0	0	0	0	0
1 0 0	1	1	1	1	1	1	1	1	1
1 0 1	?	0	0	0	0	1	1	1	1
1 1 0	?	0	0	1	1	0	1	0	1
1 1 1	?	0	1	0	1	0	0	1	1

- We can't learn this function !
- Try to verify that the eight solutions are equivalent, this is, all provide the same error.
  - Fix one of them as true solution and count how many of the others provide one, two o three errors on the unknown values.

What then ?

- **Inductive Learning** is a hopeless approach:

In a strict sense learning out of the sample is not possible!!

(see Inductivist Turkey (Bertrand Russell) ☺)

Is there any hope to know anything about *f* outside the data set **without making assumptions** about *f* ?

Yes, if we are willing to give up "for sure".

Try to learn something less exigent than the proper **unknown** function,

i.e. some useful property about the **unknown** function

Let's try to exploit randomness….

- NEW Hypothesis: items inside $\mathcal{D}$ are i.i.d samples from a probability distribution $\mathcal{P}$

- **Consequences:**
  - $\mathcal{D}$ **is the output of a random variable (vector)**
  - **It is not realistic to expect that every sample** $\mathcal{D}$ represent equally well the distribution $\mathcal{P}$
  - **The function *g* depends on** $\mathcal{D}$, hence its election is also a random process .

- **Where is the novelty ?**
  - **Probability theory shows that there are probabilistic dependencies between a random variable and a sample of it (under conditions).**
    - **Example : Confidence interval for the sample mean** $P(|\bar{x} - \mu| < \epsilon) > 1 - \delta, \qquad \delta(\epsilon) \ll 1$

But is probability enough?

- MAIN QUESTION:

  There exists a learning algorithm A and a sample size m such that for every distribution P, if A receives m i.i.d. samples from P, there is a high chance it outputs a predictor g with low error?

- No-Free-Lunch (NFL) Theorem (Informal): "For every algorithm there exist a $\mathcal{P}$ on which it fails, even though that $\mathcal{P}$ can be sucessfully learned by another learner. Moreover, all algorithms are equivalent in average on all possible target functions f"

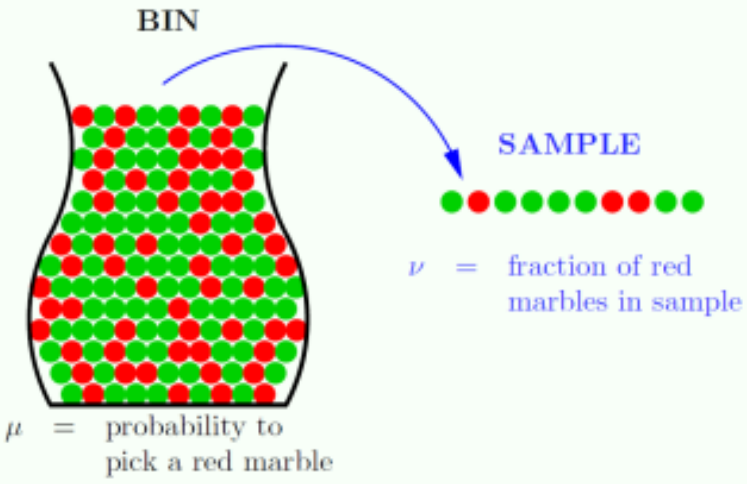
- In order to succeed each learner ($\mathcal{A}, \mathcal{H}$) must be applied on the class of distributions $\mathcal{P}$ that it can learn.

- This highlights the need for **exploiting problem-specific knowledge** to achieve better than random performance
  - Geometric constraint
  - Class of function with zero or very small E_{out}
  - Finite class $\mathcal{H}$
  - Finite VC dimension
  - etc

Can we infer something outside the data using only $\mathcal{D}$?:

The PAC answer

Population Mean from Sample Mean



The BIN Model

- Bin with red and green marbles.
- Pick a sample of N marbles *independently*.
- μ: probability to pick a red marble.
  ν: fraction of red marbles in the sample.

Sample $\longrightarrow$ the data set $\longrightarrow$ ν
BIN $\longrightarrow$ outside the data $\longrightarrow$ μ

Can we guarantee 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. to predict the outcome of a presidential election).
ANSWER: The bad case is possible, but **not probable**.

Hoeffding's Inequality

Hoeffding/Chernoff proved that, most of the time, **for a fixed μ** , ν cannot be too far from μ

$$\mathbb{P}(\mathcal{D}\text{: } |\mu - \nu| > \epsilon) \leq 2e^{-2\epsilon^2 N} \qquad \text{for any } \epsilon > 0$$

$$\mathbb{P}(\mathcal{D}\text{: } |\mu - \nu| \leq \epsilon) \geq 1 - 2e^{-2\epsilon^2 N} \qquad \text{for any } \epsilon > 0$$

Question: What does the value of ν tell us on μ: $\mu \approx \nu \Leftrightarrow \nu \approx \mu$

Example: $N = 1,000$; 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

$$\mu \in [\nu - 0.05, \nu + 0.05], \qquad \text{(the error bar is } \pm 0.05\text{)}$$

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

We learned *<u>something</u>*. From *v*, we reached outside the data to μ.

Hoeffding's Inequality: Remarkable facts

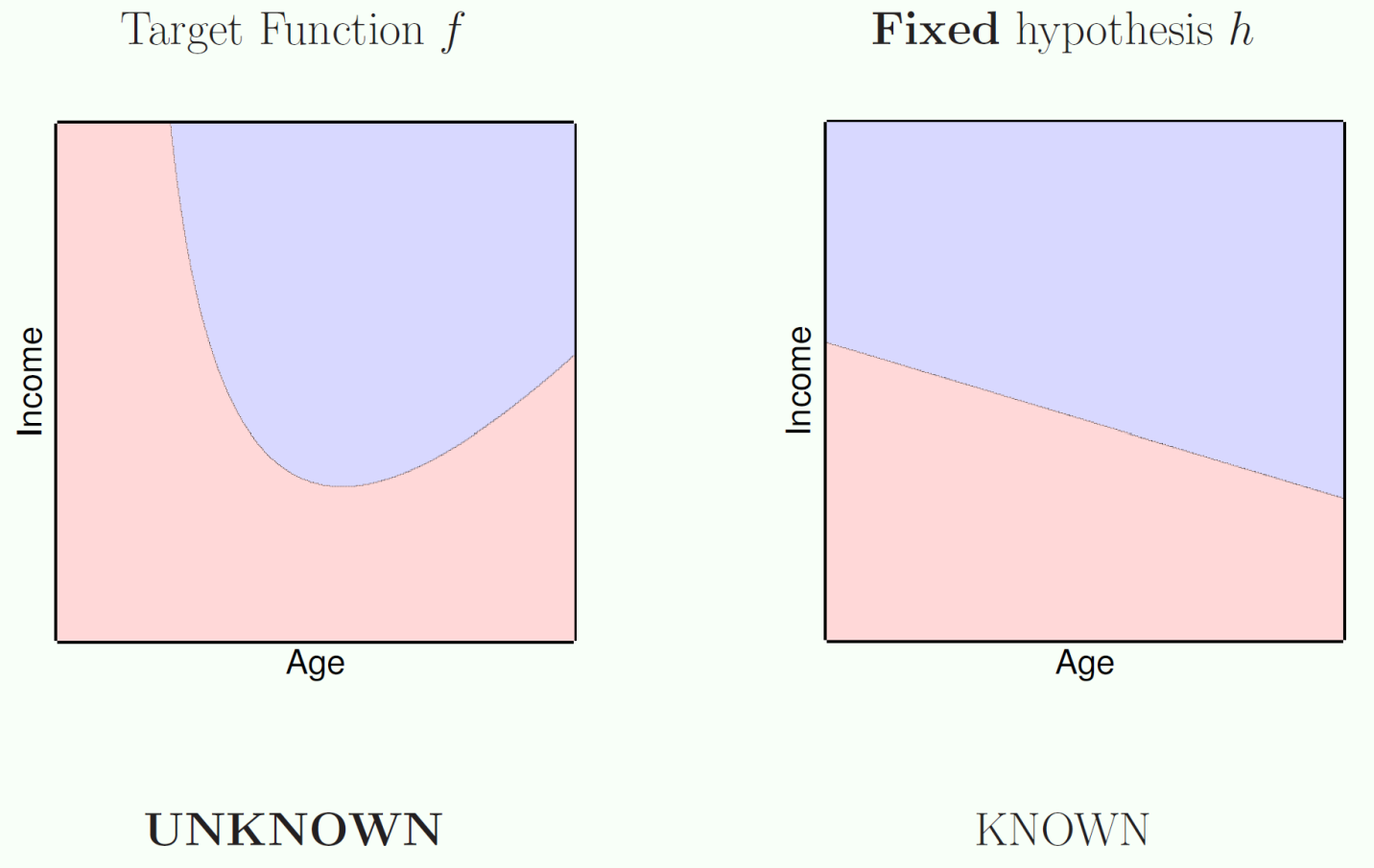
- The key ingredient: samples must be i.i.d.
  - If the sample is constructed in some arbitrary fashion, then indeed we cannot say anything.
  - Even with independence, ν can take on arbitrary values; but some values are more likely than others.
  - This is what allows us to learn something – it is likely that ν ≈ μ.

- The bound $2e^{-2\epsilon^2 N}$ 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; and we mean **unknown**.

- The key player in the bound $2e^{-2\epsilon^2 N}$ is N.
  - If N → ∞, μ ≈ ν with very very very . . . high probabilty, but not for sure.
  - Can you live with 10^{-100} probability of error?

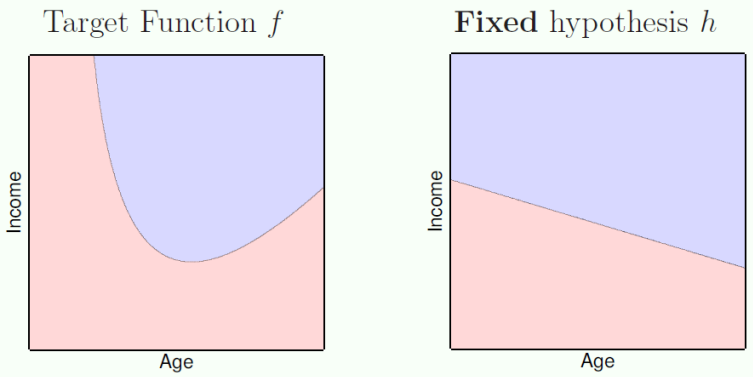
$$\mathbb{P}(\mathcal{D}\text{: } |\mu - \nu| > \epsilon) \leq 2e^{-2\epsilon^2 N}$$

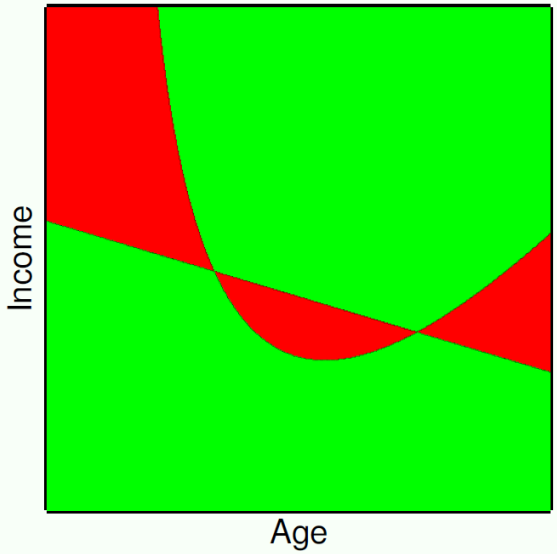
Learning setup



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

The Learning Error Function





UNKNOWN

$$\text{green: } h(\mathbf{x}) = f(\mathbf{x})$$
$$\text{red: } h(\mathbf{x}) \neq f(\mathbf{x})$$

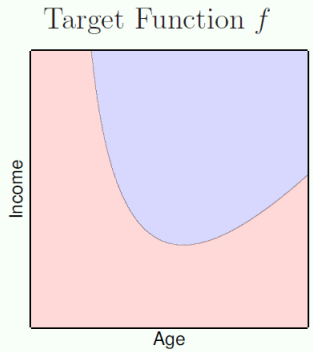
$$E(h) = \mathbb{P}_{\mathbf{x}}[h(\mathbf{x}) \neq f(\mathbf{x})]$$

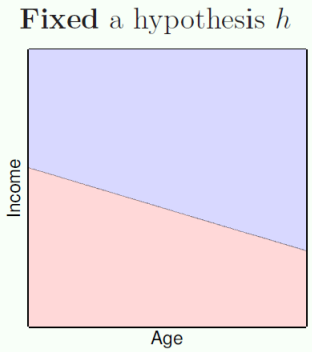
("size" of the red region)

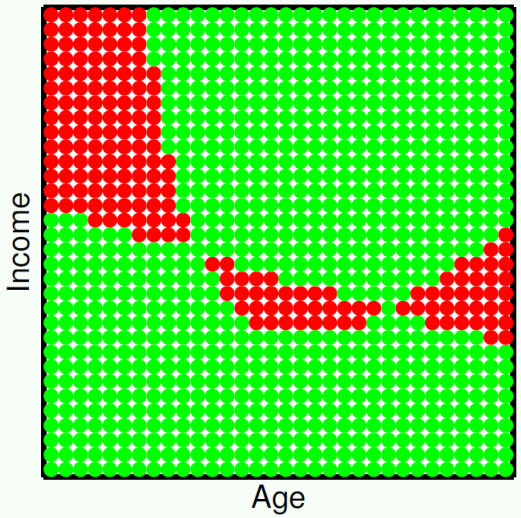
$P(\mathbf{x})$

The function h defines an unknown but fixed probability error E(h)

Relating the Bin to Learning







UNKNOWN

green "marble": $h(\mathbf{x}) = f(\mathbf{x})$

red "marble": $h(\mathbf{x}) \neq f(\mathbf{x})$

BIN: $\mathcal{X}$

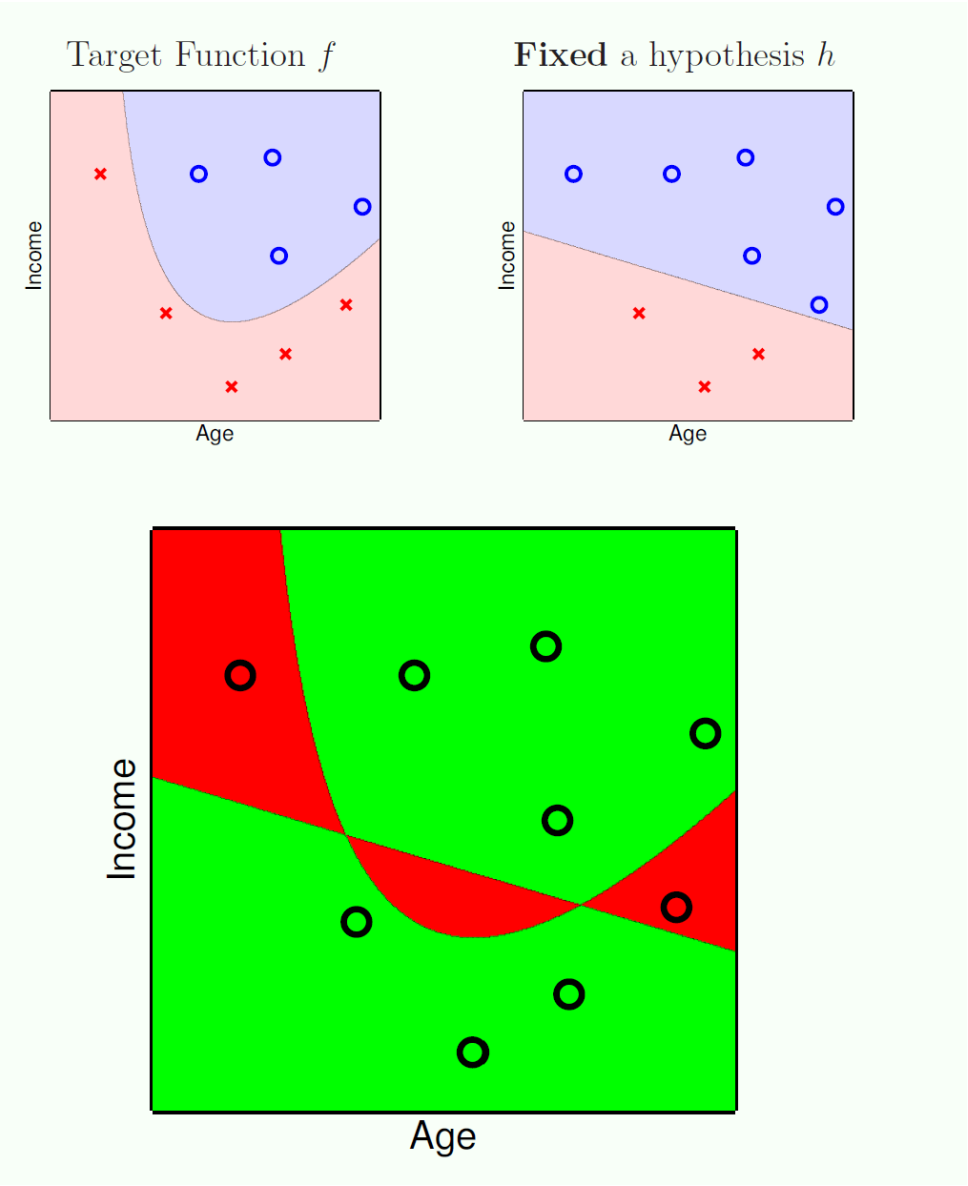
$$E_{\text{out}}(h) = \mathbb{P}_{\mathbf{x}}[h(\mathbf{x}) \neq f(\mathbf{x})]$$

out-of-sample

Let´s consider all possible sample points

Now a Bin Model is defined by *h* and *f*

Relating the Bin to Learning - the Data

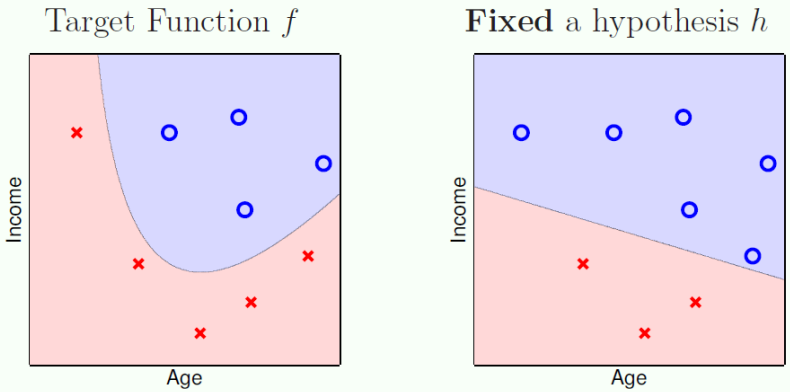


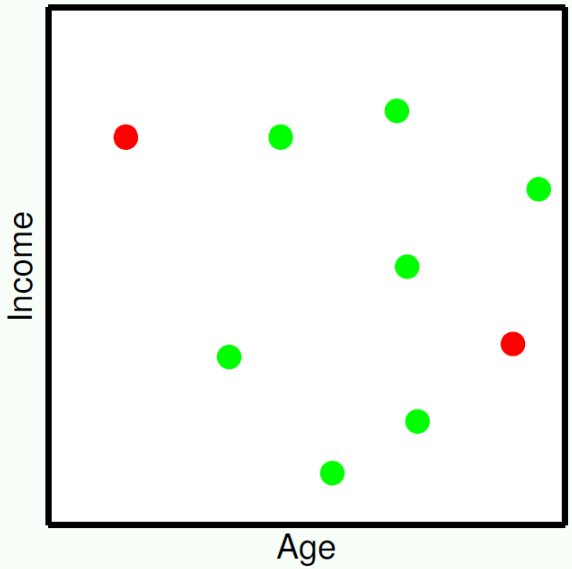
On the same sample, the target function f and the hypothesis h provides us with different labels

We have points in different zones of the error function.

If the sample is draw independently acccording to P, each point will be red with probability μ and green with probability $1-\mu$

Relating the Bin to Learning - the Data





KNOWN!

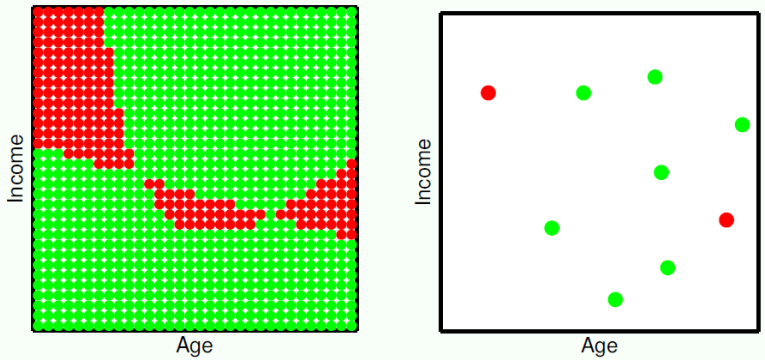
$$\text{green data: } h(\mathbf{x}_n) = f(\mathbf{x}_n)$$
$$\text{red data: } h(\mathbf{x}_n) \neq f(\mathbf{x}_n)$$

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

in-sample

misclassified

Bin Model and Learning



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

Learning

input space $\mathcal{X}$

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

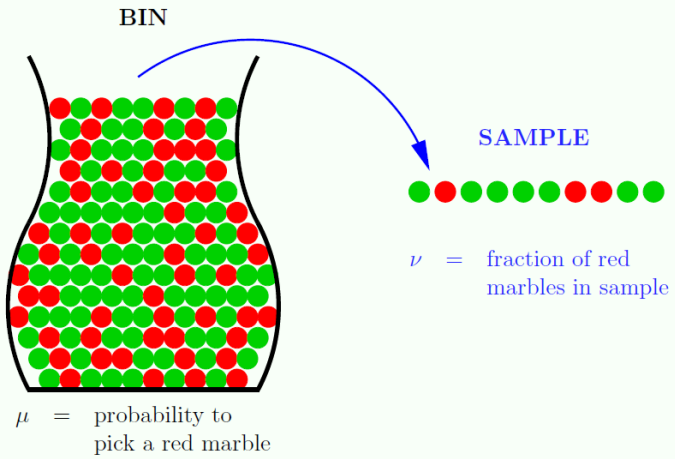
$\mathbf{x}$ for which $h(\mathbf{x}) \neq f(\mathbf{x})$

$P(\mathbf{x})$

data set $\mathcal{D}$

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

In-sample Error: $E_{\text{in}}(h) = \frac{1}{N} \sum_{n=1}^{N} [\![h(\mathbf{x}) \neq f(\mathbf{x})]\!]$



Bin Model

Bin

green marble

red marble

randomly picking a marble

sample of N marbles

μ = probability of picking a red marble

ν = fraction of red marbles in the sample

Hoeffding inequality in Learning

- Let's consider $\mathcal{H}=\{h\}$, only one function, and $f(x)$ the unknown true function.
- Let's $[\![f(x)=h(x)]\!]$ and $[\![f(x)\neq h(x)]\!]$ represent new binary variables in the population. Now $\mu = \Pr([\![f(x)\neq h(x)]\!])$
- For any training sample $\mathcal{D}$ of size N , $\nu = \text{Fraction}([\![f(x)\neq h(x)]\!])$ on $\mathcal{D}$
- Now μ and ν represent the population and sample error respectively.
- Let's denote by $E_{out}(h)=\mu \;\; and \;\; E_{in}(h)=\nu$ the h's global and sample error respectively
- The Hoeffding inequality can be rewritten as:

$$P(\mathcal{D}: |E_{out}(h)-E_{in}(h)|>\epsilon) \leq 2e^{-2\epsilon^2 N} \;\; \text{for any} \;\; \epsilon>0$$

This is called a "Probably Aproximately Correct (PAC)" result

- IMPORTANT: Note that **h** is fixed before knowing the data sample

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

$\mathbb{P}(\mathcal{D}\text{: } |\mu - \nu| > \epsilon) \leq 2e^{-2\epsilon^2 N}$ $\Longleftrightarrow$ $\mathbb{P}(\mathcal{D}\text{: } |E_{\text{out}}(h) - E_{\text{in}}(h)| > \epsilon) \leq 2e^{-2\epsilon^2 N}$

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

- If $E_{\text{in}}(h) \approx 0 \overset{N >> 1}{\Longrightarrow} E_{out}(h) \approx 0$ (with high probability), i.e. $\mathbb{P}_{\mathcal{X}}[h(\mathbf{x}) \neq f(\mathbf{x})] = 0$
  - We have learned something about the <u>*entire*</u> f: $f \approx h \; over \; \mathcal{X}$ (outside $\mathcal{D}$)

- If $E_{\text{in}} \gg 0$, we're out of luck.
  - But, we have still learned something about the <u>*entire*</u> f: $f \not\approx h \; over \; \mathcal{X}$; it is not very useful though.

Questions:

1. Suppose that E_{in} = 1, have we learned something about the entire *f* that is useful?
2. What is the worst E_{in} for inferring about *f* ?

Understanding PAC results

$$P(\mathcal{D}: |\mathrm{E}_{out}(h) - \mathrm{E}_{in}(h)| > \epsilon) \leq 2e^{-2\epsilon^2 N} \text{ for any } \epsilon > 0$$

- Let's consider $\delta = 2e^{-2\epsilon^2 N}$ then

$$P(\mathcal{D}: |\mathrm{E}_{out}(h) - \mathrm{E}_{in}(h)| > \epsilon) \leq \delta \Leftrightarrow P(\mathcal{D}: |\mathrm{E}_{out}(h) - \mathrm{E}_{in}(h)| < \epsilon) \geq 1 - \delta$$

- Or equivalently:

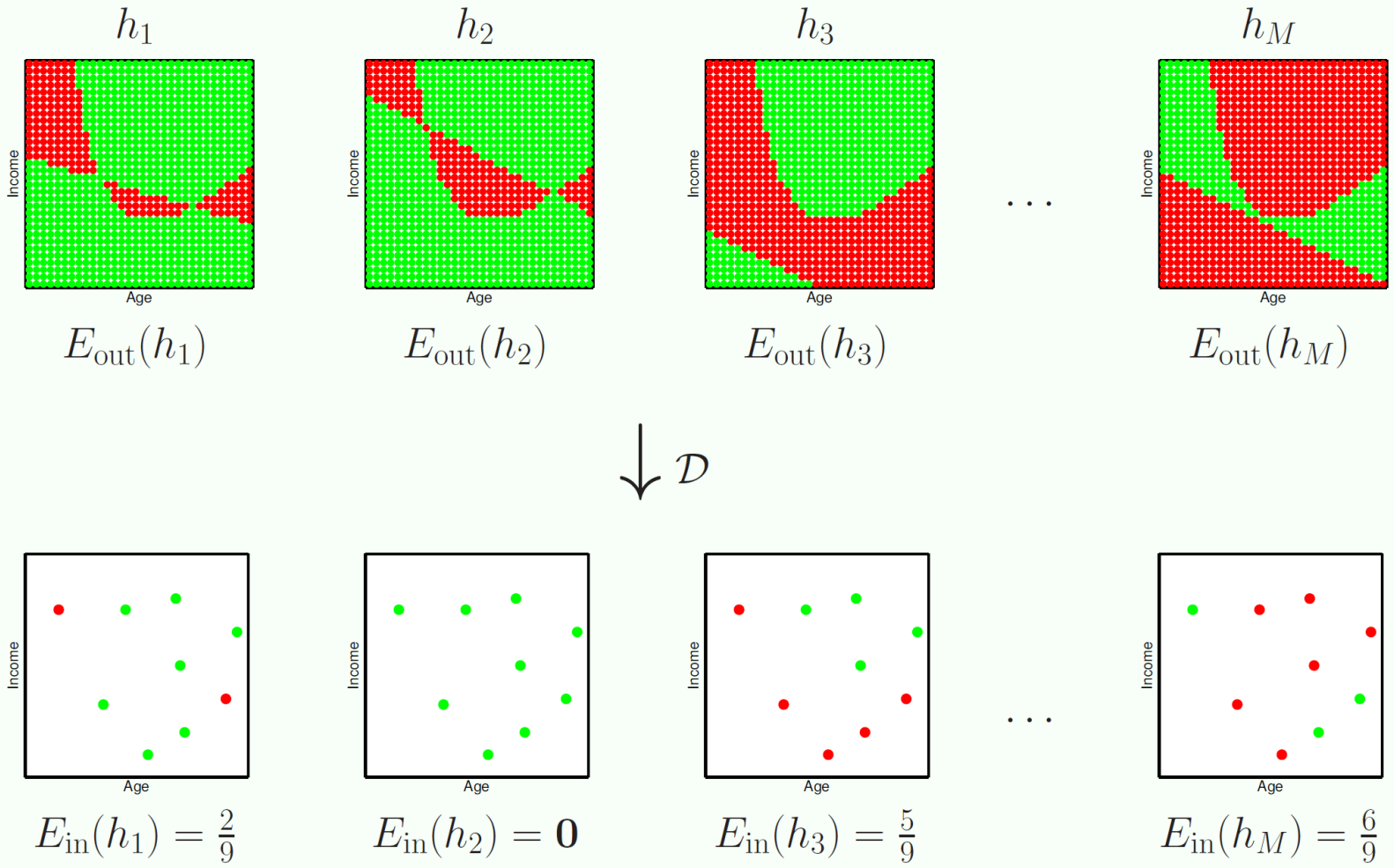
$$\mathrm{E}_{out}(h) \leq \mathrm{E}_{in}(h) + \epsilon \text{ , with probability at least } 1 - \delta \text{ on } \mathcal{D}$$

- Let's write ϵ as a function of N and δ, then

$$\mathrm{E}_{out}(h) \leq \mathrm{E}_{in}(h) + \sqrt{\frac{1}{2N}\log\frac{2}{\delta}} \text{ with probability at least } 1 - \delta \text{ on } \mathcal{D}$$

- The higher N the narrow the interval (The sample size is important !!)
- The smaller δ the larger the interval (The higher guarantee the lesser accuracy)

Real Learning – Finite Learning Model



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

The Hoeffding inequality for multiple hypothesis

THINGS ARE DIFFERENT: In Hoeffding's inequality the h is fixed before knowing the data, BUT in REAL PROBLEMS the chosen hypothesis, $g \in \mathcal{H}$ is identified using the data.

SEARCH CAUSES SELECTION BIAS: THE COIN ANALOGY

Question: if toss a **fair coin** ten times, what is the probability that you will get ten heads ?

Answer: $\approx$ **0.1** (try it)

Question: if toss 1000 **fair coins** ten times, what is the probability that some coin will get ten heads ?

Answer: $\approx$ **0.63** (try it)

Identifying coins with functions: the higher the size of $\mathcal{H}$ the higher the probability of having a hypothesis with $E_{in} \approx 0$ error , BUT can we expect E_{out} to be small ?

Then what?

- Adapting the Hoeffding's inequality to the case of finite $\mathcal{H}$
  1. The hypothesis solution $\boldsymbol{g}$ **should be fixed before knowing the data sample**. (MANDATORY CONDITION)
  2. Nevertheless, the Learning Algorithm uses the training data to search for $\boldsymbol{g}$.

- A simple solution is to consider an event valid for all functions in $\mathcal{H}$.
  - Let g denote a generic hypothesis solution then,

$$\{\mathcal{D}: |E_{in}(g) - E_{out}(g)| > \epsilon\} = \bigcup_{h_i \in H} (\mathcal{D}: |E_{in}(h_i) - E_{out}(h_i)| > \epsilon)$$

  - Using $P\left(\cup_{i=1:|\mathcal{H}|} B_i\right) \leq \sum_{i=1}^{|\mathcal{H}|} P(B_i)$

$$P(\mathcal{D}: |\mathrm{E}_{in}(g) - \mathrm{E}_{out}(g)| > \epsilon) < 2|\mathcal{H}|e^{-2\epsilon^2 N} \text{ for any } \epsilon > 0$$

Uniform Convergence

Interpreting the bound

$$\mathbb{P}(\mathcal{D}: |\mathrm{E}_{in}(\boldsymbol{g}) - \mathrm{E}_{out}(\boldsymbol{g})| > \epsilon) \leq 2|\mathcal{H}|e^{-2\epsilon^2 N} \qquad \text{for any } \epsilon > 0$$

$$\mathbb{P}(\mathcal{D}: |\mathrm{E}_{in}(\boldsymbol{g}) - \mathrm{E}_{out}(\boldsymbol{g})| \leq \epsilon) \geq 1 - 2|\mathcal{H}|e^{-2\epsilon^2 N} \qquad \text{for any } \epsilon > 0$$

Generalization bar error

Result: With probability at least $1 - \boldsymbol{\delta}$, $E_{out}(g) \leq E_{in}(g) + \sqrt{\frac{1}{2N}\ln\frac{2|\mathcal{H}|}{\delta}}$

Let's denote $\delta = 2|\mathcal{H}|e^{-2\epsilon^2 N}$ and writing ϵ as a function of N, δ and $|\mathcal{H}|$

E_{in} reaches outside to E_{out} when $|\mathcal{H}|$ is small $\qquad E_{out}(g) \leq E_{in}(g) + \mathcal{O}\left(\sqrt{\frac{\ln|\mathcal{H}|}{N}}\right)$

if $N \gg \ln|\mathcal{H}|$, then $E_{out}(g) \approx E_{in}(g)$

- This bound does not depend on $\mathcal{X}$, $\mathbb{P}$(x), f or how g is found. (a worst case bound)
- Only requires P(x) to generate the data points independently *including* the test point.

A particular case: The Realizability Hypothesis

- Definition ($\mathcal{H}$ -realizable): There exist $h^* \in \mathcal{H}\ s.t.\ E_{out_{\mathcal{P},f}}(h^*) = 0$
- This means that the class of functions $\mathcal{H}$ includes at least a function with error zero for any $\mathcal{P}$ and f.
- Being a mathematical hypothesis can be applied in real tasks.
  - Example: Separated clases in classification
- Under this hyphotesis can be shown that the ERM rule on finite clases $\mathcal{H}$ always provides a function h_S s.t. $E_{in}(h_S) = 0$, obtaining the smallest error bound

$$\mathrm{E}_{out}(h_S) \leq \frac{1}{N}\log\frac{|\mathcal{H}|}{\delta} \quad with\ probability\ al\ least\ 1-\delta$$

Formal Definition of Realizable PAC-Learning

- A realizable hypothesis class $\mathcal{H}$ is PAC learnable if there exist a function $m_{\mathcal{H}}(\epsilon, \delta) \to \mathbb{N}$ and a learning algorithm $\mathcal{A}$ with the following property:
  - For every, $\epsilon, \delta \in (0,1)$
  - For every distribution $\mathcal{P}$ over $\mathcal{X}$
  - For every labeling function f

  if the realizable assumption holds with respect to $\mathcal{H}, \mathcal{P}, f$ then when running the learning algorithm on $\mathrm{N} \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d samples generated by $\mathcal{P}$ and labeled by f, the algorithm returns a hypothesis h such that , with probability al least 1- δ (over the choice of the m training samples)

$$ERM_{\mathcal{P}, f}(h) \leq \epsilon$$

This means the learning algorithm always return a hypothesis h with error close to zero

Sample Complexity in PAC-Learnability

- How many examples are required to guarantee a PAC solution in realizable classes?
- This depend on ϵ, δ, $\mathcal{H}$ and the loss function range
- If $\mathcal{H}$ is a PAC-realizable, there exist many functions $m_{\mathcal{H}}(\epsilon, \delta)$ that satisfy the requirements given in the definition of Realizable PAC learnability
- The sample complexity for learning $\mathcal{H}$ is defined as the "minimal integer" $m_{\mathcal{H}}(\epsilon, \delta)$ that satisfies the requirements of realizable PAC learning with accuracy ϵ and confidence δ
- Result for loss function with range in [0,1]: every realizable finite hypothesis class is PAC learnable with sample complexity

$$m_H(\varepsilon, \delta) \leq \left\lceil \frac{1}{\epsilon} \log \frac{|\mathcal{H}|}{\delta} \right\rceil$$

Formal Definition of Agnostic PAC-Learning

- A hypothesis class $\mathcal{H}$ is PAC learnable if there exist a function $m_{\mathcal{H}}(\epsilon, \delta) \rightarrow \mathbb{N}$ and a learning algorithm $\mathcal{A}$ with the following property:
  - For every, $\epsilon, \delta \in (0,1)$
  - For every distribution $\mathcal{P}$ over $\mathcal{X}$

  When running the learning algorithm on $\mathrm{N} \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d samples generated by $\mathcal{P}$, the algorithm returns a hypothesis h such that , with probability al least 1- δ (over the choice of the m training samples)

$$E_{in_{\mathcal{P}}}(h) \leq \min_{h' \in \mathcal{H}} E_{in_{\mathcal{P}}}(h') + \epsilon$$

This means the learning algorithm always return a hypothesis h closer to the best possible inside of the class $\mathcal{H}$

The ERM rule is a succesful agnostic PAC learner for finite classes $\mathcal{H}$

Agnostic-PAC Sample Complexity

- How many examples are required to guarantee a PAC solution?
- This depend on ϵ, δ, $\mathcal{H}$ and the loss function range
- If $\mathcal{H}$es PAC learnable, there exist many functions $m_{\mathcal{H}}(\epsilon, \delta)$ that satisfy the requirements given in the definition of PAC learnability
- The sample complexity of learning $\mathcal{H}$ is defined as the "minimal integer" $m_{\mathcal{H}}(\epsilon, \delta)$ that satisfies the requirements of PAC learning with accuracy ε and confidence δ
- Result for loss function with range in [0,1]: every finite hypothesis class is PAC learnable with sample complexity

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

- Compare with the PAC-realizable case!

Feasibility of Learning vs Complexity

- Learning is only possible in a probabilistic setting (under conditions):
  - Samples from X **must be i.i.d**
  - Same probability distribution in training and test

- To be succesful in learning means to find a function g, s.t. $\mathrm{E}_{out}(g) \approx 0$
- Nevertheless, we are only able to guarantee,

$$P(\mathcal{D}\text{: } |\mathrm{E}_{in}(g) - \mathrm{E}_{out}(g)| > \epsilon) < 2|\mathcal{H}|e^{-\epsilon^2 N} \text{ for any } \epsilon > 0$$

- Feasibility of Learning must answer two questions:
  1. Can we make sure that $\mathrm{E}_{out}(g)$ is close enough to $\mathrm{E}_{in}(g)$?
  2. Can we make $\mathrm{E}_{in}(g)$ small enough?

- What is the relationship between Feasibility of Learning and the complexity of $\mathcal{H}$ and f?

Feasibility of learning : $E_{out} \approx 0$

Two conditions:

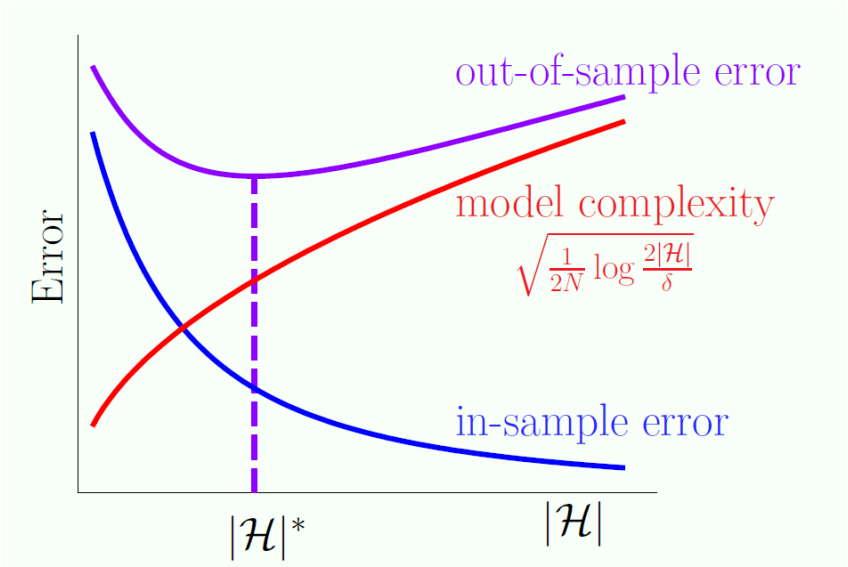
(1) $E_{in} \approx E_{out}$ → Is verified thank to the Hoeffding's inequality

(2) $E_{in} \approx 0$ → Is achieved through the learning algorithm

Together, these ensure $E_{out} \approx 0$

BUT there is a tradeoff on $\mathcal{H}$:

- **Small $|\mathcal{H}| \Rightarrow E_{in} \approx E_{out}$**
- **Large $|\mathcal{H}| \Rightarrow E_{in} \approx 0$ is more likely**



What about the UNKNOWN complexity of f :

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

Feasibility of Learning (finite $\mathcal{H}$): Summary

- Out of $\mathcal{D}$, nothing about *f* can be guaranteed

- If $\mathcal{D}$ is an independent sample from $\mathbb{P}$(**x**).

  $E_{out} \approx E_{in}$ (E_{in} can reach outside the data set to E_{out}).

- But, what we want is $E_{out} \approx 0$.
- The two step solution. We trade $E_{out} \approx 0$ for 2 goals:

  (i) $E_{out} \approx E_{in}$

  (ii) $E_{in} \approx 0$.

  We know E_{in}, not E_{out}, but we can *ensure* (i) if $|\mathcal{H}|$ is small.

Any ERM rule is a succesful PAC learner for finite classes $\mathcal{H}$