LEARNING

Dr. Matthieu R Bloch

Wednesday, December 1, 2021

LOGISTICS

General announcements

- Assignment 6 posted (*last assignment*)
- Due December 7, 2021 for bonus, deadline December 10, 2021
- 2 lectures left
- Let me know what's missing

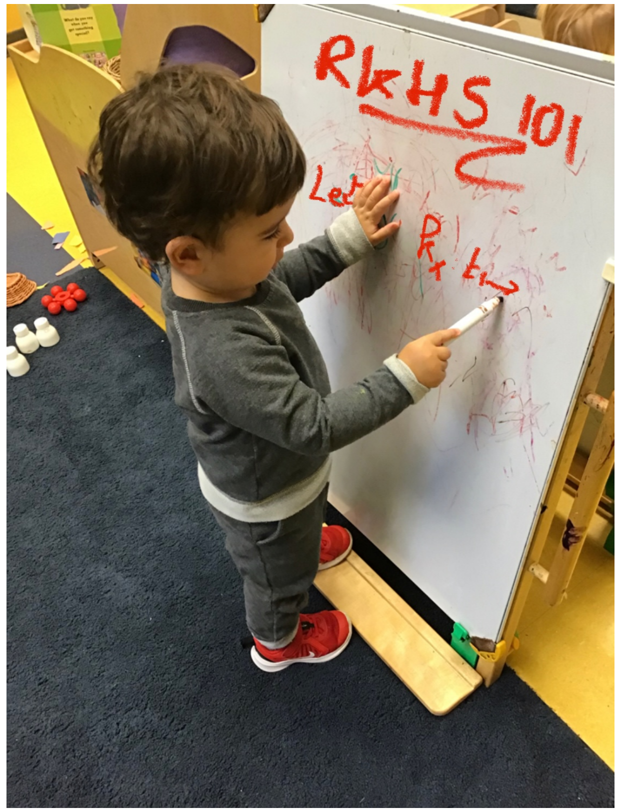
Assignment 5 grades posted

Reviewing Midterm2 grades one last time

WHAT'S ON THE AGENDA FOR TODAY?

The learning problem and why we need probabilities.

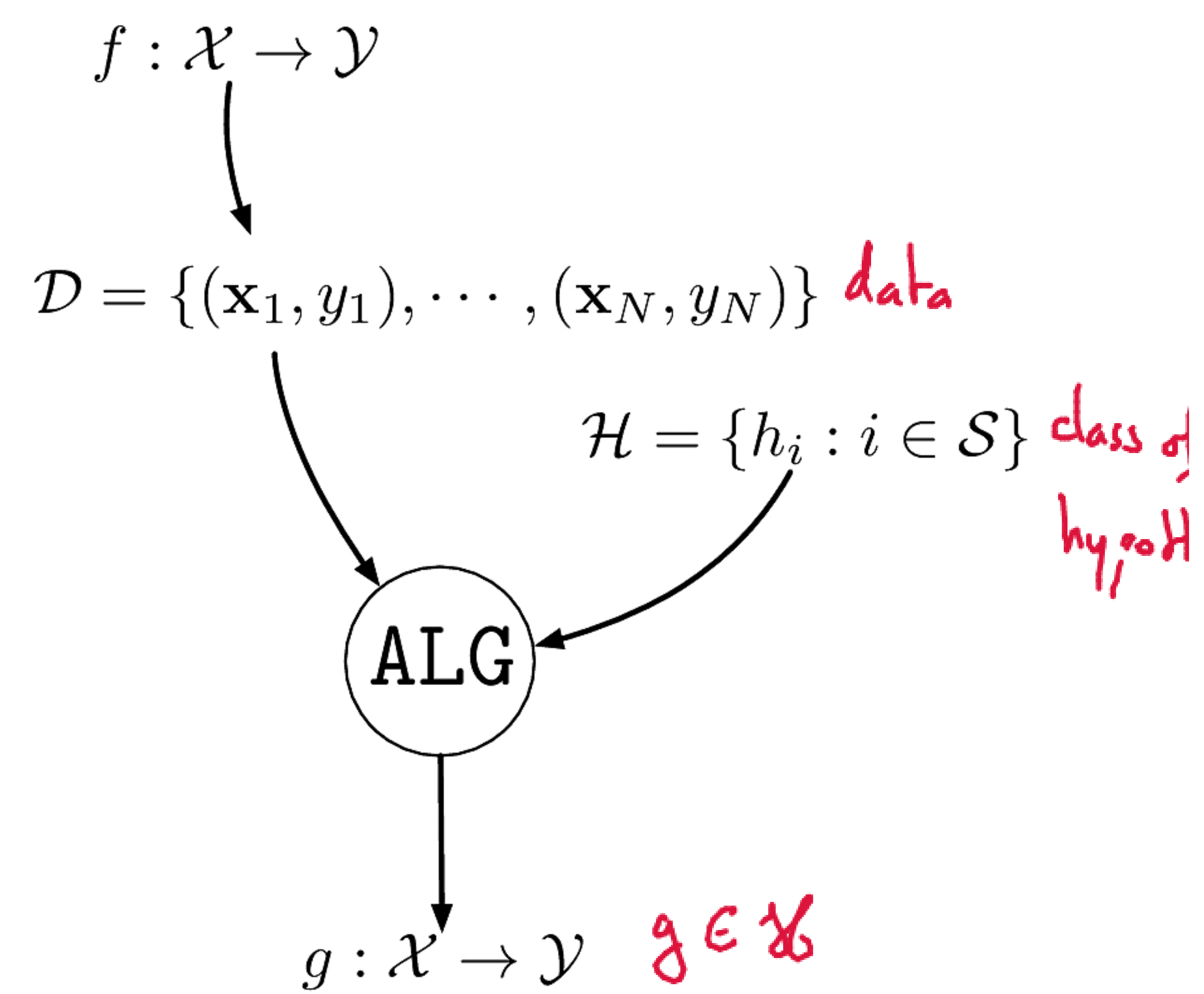
Lecture notes 17 and 23



Toddlers can do it!

COMPONENTS OF SUPERVISED MACHINE LEARNING

1. An *unknown function* $\boxed{f : \mathcal{X} \to \mathcal{Y} : \mathbf{x} \mapsto y = f(\mathbf{x})}$ to learn
   - The formula to distinguish cats from dogs
2. A dataset $\mathcal{D} \triangleq \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$
   - $\mathbf{x}_i \in \mathcal{X} \triangleq \mathbb{R}^d$: picture of cat/dog
   - $y_i \in \mathcal{Y} \triangleq \mathbb{R}$: the corresponding label cat/dog
3. A *set of hypotheses* $\mathcal{H}$ as to what the function could be
   - Example: deep neural nets with AlexNet architecture
4. An *algorithm* $\mathtt{ALG}$ to find the best $h \in \mathcal{H}$ that explains f



Learning model #1

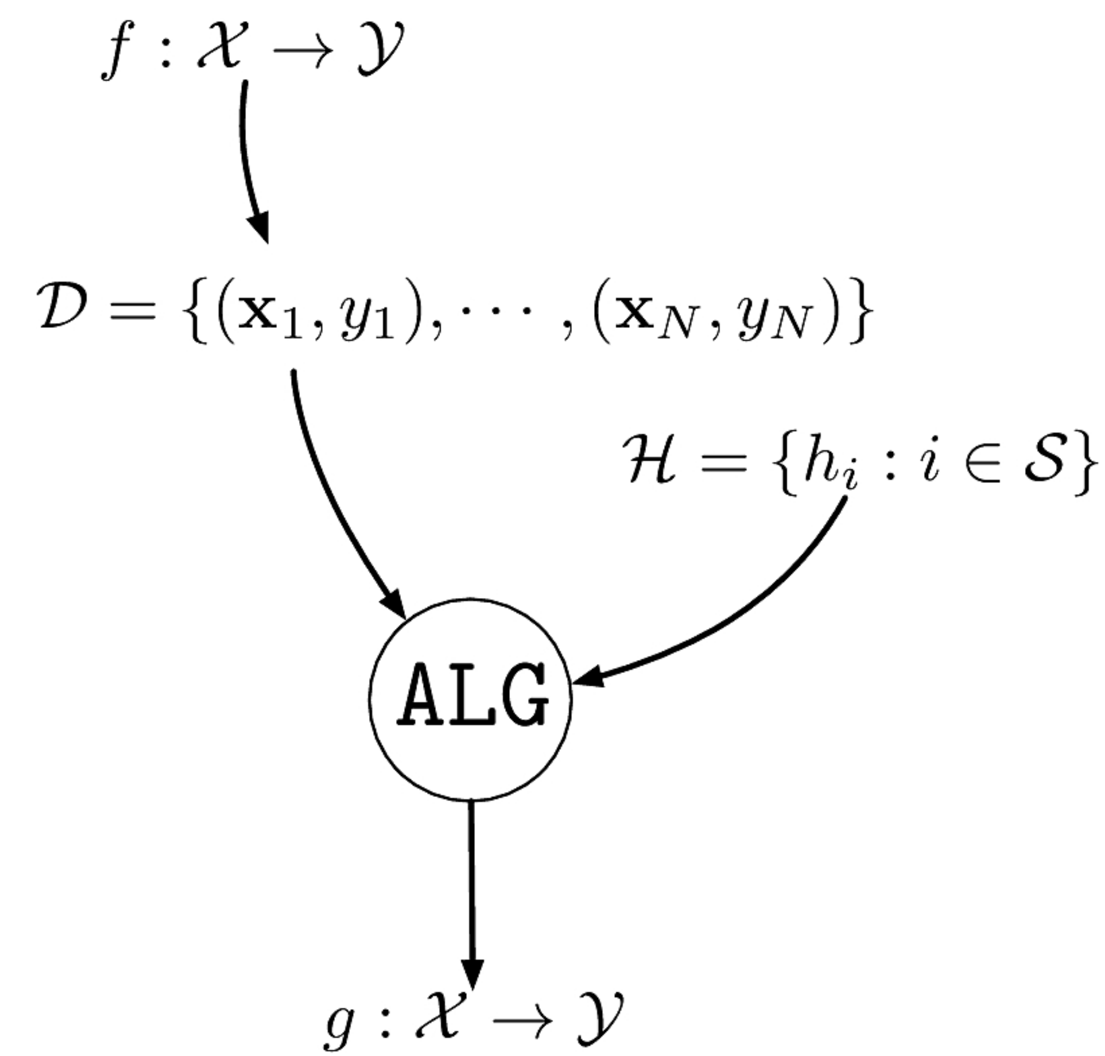
COMPONENTS OF SUPERVISED MACHINE LEARNING

1. An *unknown function* $f : \mathcal{X} \to \mathcal{Y} : \mathbf{x} \mapsto y = f(\mathbf{x})$ to learn
    - The formula to distinguish cats from dogs
2. A dataset $\mathcal{D} \triangleq \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$
    - $\mathbf{x}_i \in \mathcal{X} \triangleq \mathbb{R}^d$: picture of cat/dog
    - $y_i \in \mathcal{Y} \triangleq \mathbb{R}$: the corresponding label cat/dog
3. A *set of hypotheses* $\mathcal{H}$ as to what the function could be
    - Example: deep neural nets with AlexNet architecture
4. An *algorithm* $\mathtt{ALG}$ to find the best $h \in \mathcal{H}$ that explains f

Terminology:

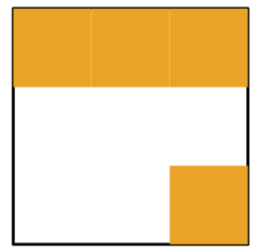
- $\mathcal{Y} = \mathbb{R}$: *regression* problem
- $|\mathcal{Y}| < \infty$: *classification* problem
- $|\mathcal{Y}| = 2$: *binary classification* problem

The goal is to *generalize*, i.e., be able to classify inputs we have *not* seen.

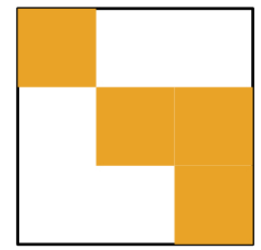


Learning model #1

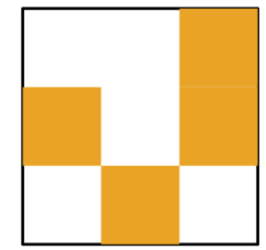
A LEARNING PUZZLE



$\mathbf{x}_1$

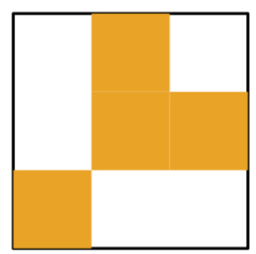


$\mathbf{x}_2$

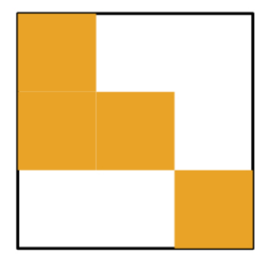


$\mathbf{x}_3$

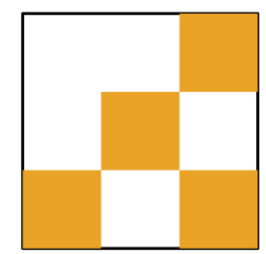
$$f(\mathbf{x}_1) = f(\mathbf{x}_2) = f(\mathbf{x}_3) = +1$$



$\mathbf{x}_4$

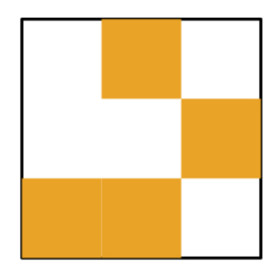


$\mathbf{x}_5$



$\mathbf{x}_6$

$$f(\mathbf{x}_4) = f(\mathbf{x}_5) = f(\mathbf{x}_6) = -1$$



$\mathbf{x}_7$

$$f(\mathbf{x}_7) = ?$$

Learning seems *impossible* without additional assumptions!

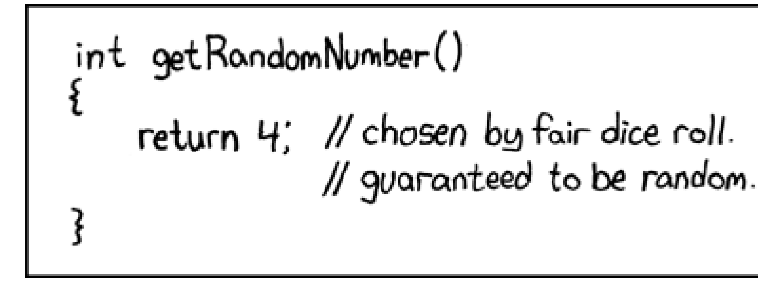
POSSIBLE VS PROBABLE

Flip a biased coin, lands on head with *unknown* probability $p \in [0, 1]$

$\mathbb{P}(\text{head}) = p$ and $\mathbb{P}(\text{tail}) = 1 - p$

Say we flip the coin N times, can we estimate p?

$$\hat{p} = \frac{\backslash\#\ \text{head}}{N}$$



https://xkcd.com/221/

Can we relate $\hat{p}$ to p?

Note: Let $\{X_i\}_{i=1}^{N}$ the coin flips $X_i \in \{\text{head}, \text{tail}\}$ $\forall i\ X_i \sim B(p)$

We compute $\hat{p} = \frac{1}{N} \sum_{i=1}^{n} \mathbb{1}\{X_i = \text{head}\}$

$\triangleq \begin{cases} 1 & \text{if } X_i = \text{head} \\ 0 & \text{else} \end{cases}$

$$\mathbb{E}(\hat{p}) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}(\mathbb{1}\{X_i = \text{head}\}) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{P}(X_i = \text{head}) = \frac{1}{N} \sum_{i=1}^{N} p = p$$

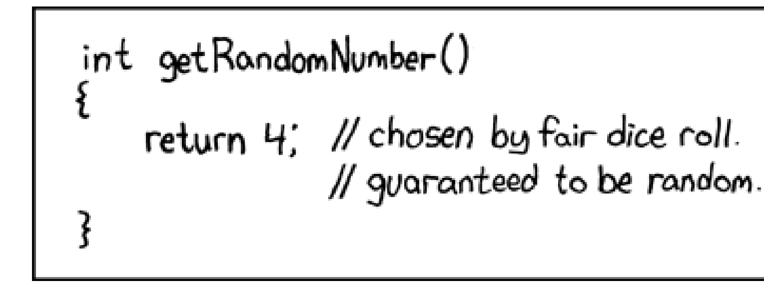
$$\mathbb{E}\{\mathbb{1}\{x \in \mathcal{A}\}\} = \sum_{x \in \mathcal{X}} P_x(x)\, \mathbb{1}\{x \in \mathcal{A}\} = \sum_{x \in \mathcal{A}} P_x(x) = \mathbb{P}(X \in \mathcal{A})$$

POSSIBLE VS PROBABLE

Flip a biased coin, lands on head with *unknown* probability $p \in [0, 1]$

$\mathbb{P}\left(\text{head}\right) = p$ and $\mathbb{P}\left(\text{tail}\right) = 1 - p$

Say we flip the coin N times, can we estimate p?



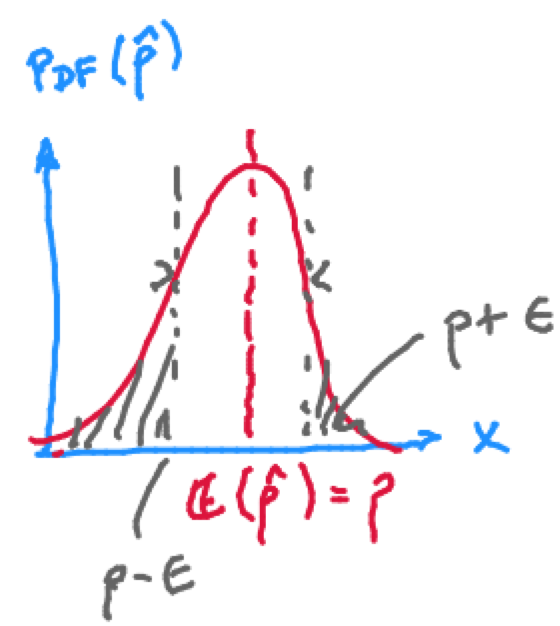
https://xkcd.com/221/

$$\hat{p} = \frac{\backslash\#\ \text{head}}{N}$$

Can we relate $\hat{p}$ to p?

- The law of large numbers tells us that $\hat{p}$ converges in probability to p as N gets large

$$\forall \epsilon > 0 \quad \mathbb{P}\left(|\hat{p} - p| > \epsilon\right) \underset{N \to \infty}{\longrightarrow} 0.$$

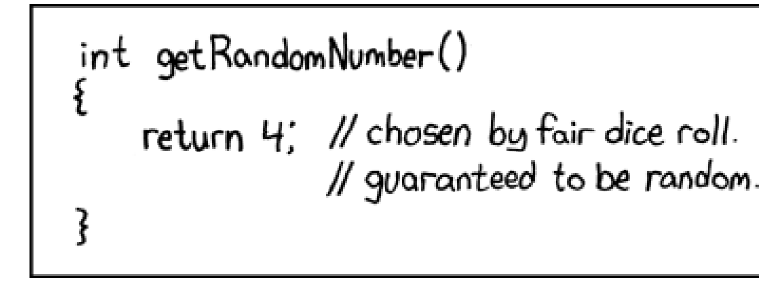


POSSIBLE VS PROBABLE

Flip a biased coin, lands on head with *unknown* probability $p \in [0, 1]$

$\mathbb{P}\left(\text{head}\right) = p$ and $\mathbb{P}\left(\text{tail}\right) = 1 - p$

Say we flip the coin N times, can we estimate p?



https://xkcd.com/221/

$$\hat{p} = \frac{\backslash\#\ \text{head}}{N}$$

Can we relate $\hat{p}$ to p?

- The law of large numbers tells us that $\hat{p}$ converges in probability to p as N gets large

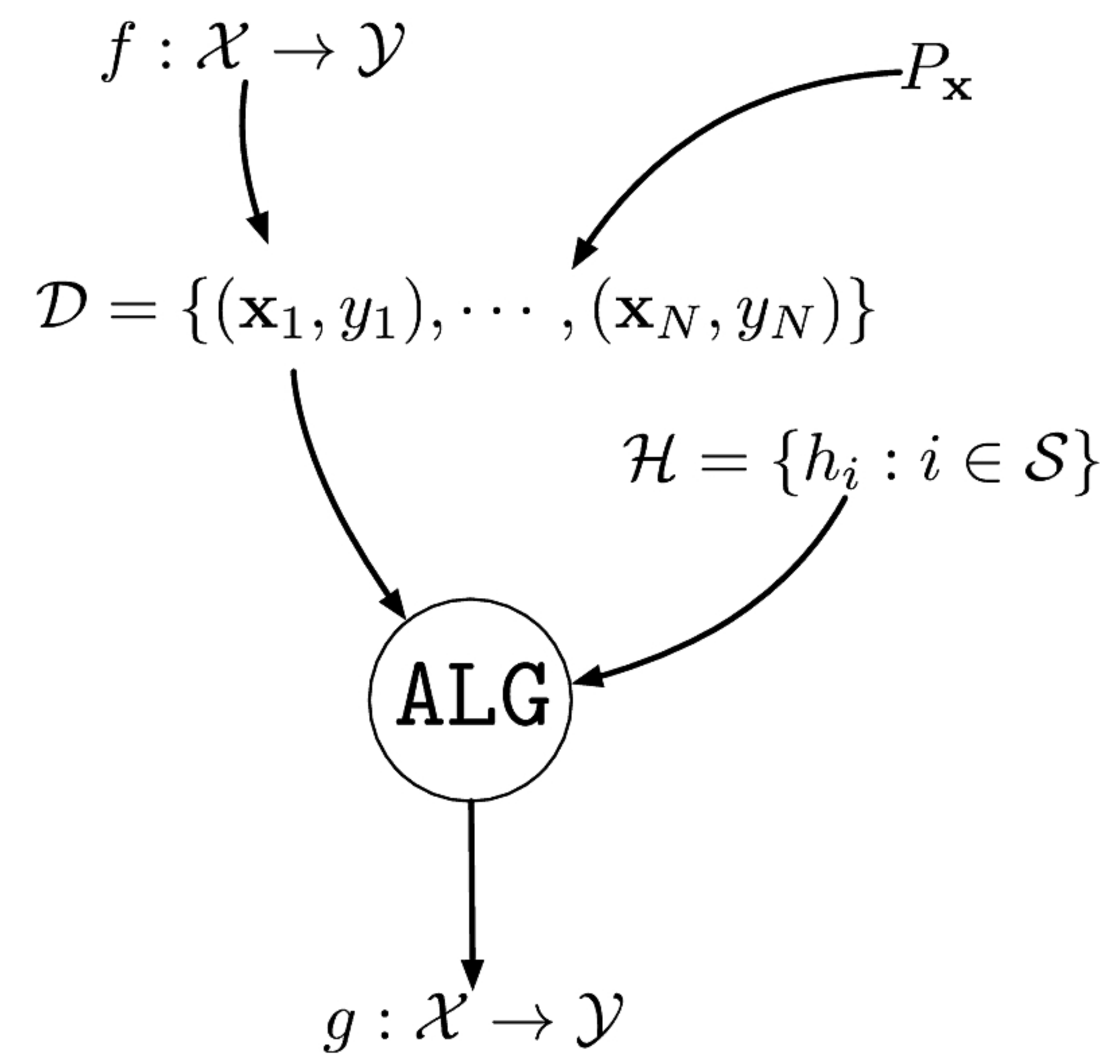
$$\forall \epsilon > 0 \quad \mathbb{P}\left(\left|\hat{p} - p\right| > \epsilon\right) \xrightarrow[N \to \infty]{} 0.$$

It is *possible* that $\hat{p}$ is completely off but it is not *probable*

COMPONENTS OF SUPERVISED MACHINE LEARNING

1. An *unknown function* $f : \mathcal{X} \to \mathcal{Y} : \mathbf{x} \mapsto y = f(\mathbf{x})$ to learn

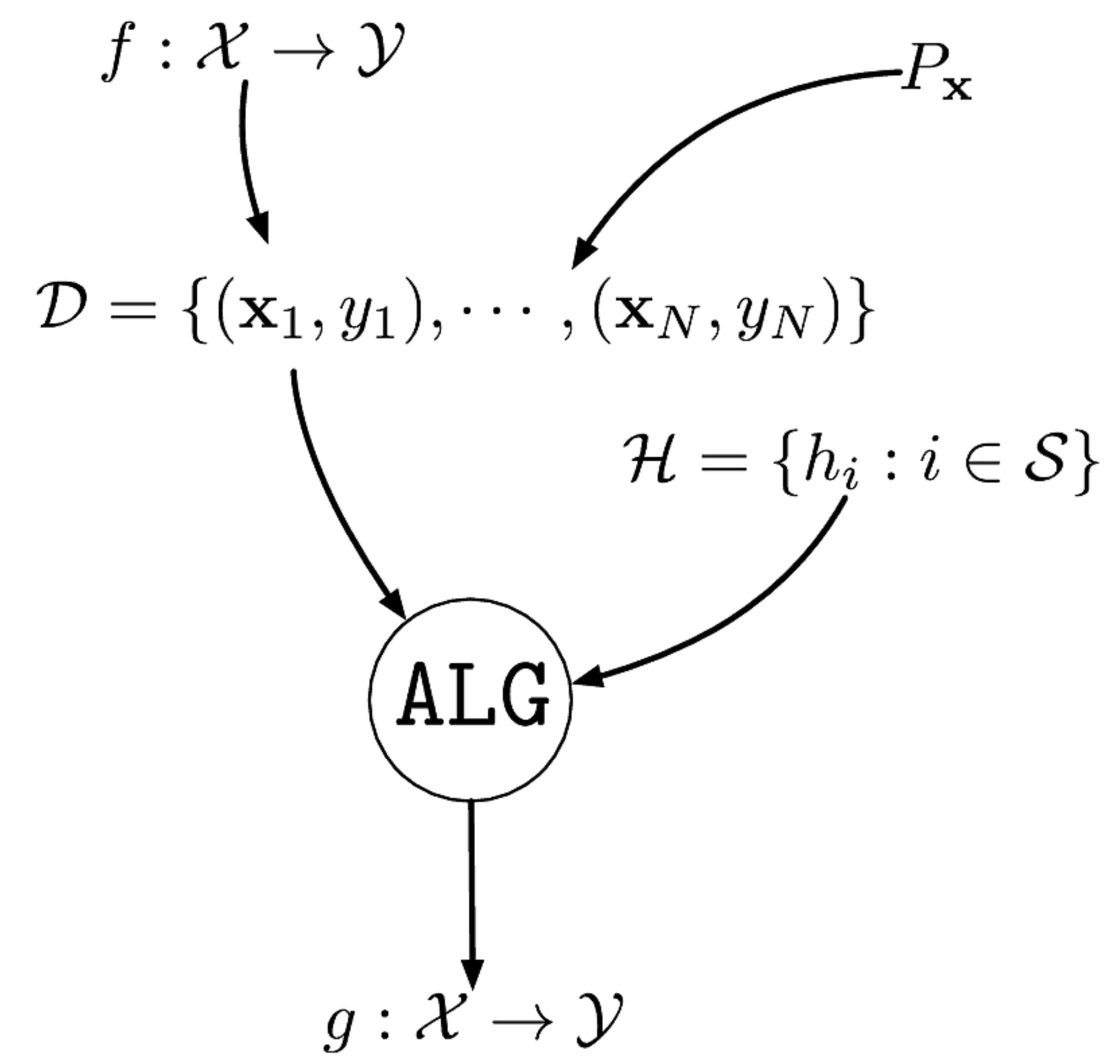
2. A *dataset* $\mathcal{D} \triangleq \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$
   - $\boxed{\{\mathbf{x}_i\}_{i=1}^N \text{ i.i.d. } \textit{from unknown distribution } P_{\mathbf{x}} \text{ on } \mathcal{X}}$
   - $\{y_i\}_{i=1}^N$ are the corresponding labels $y_i \in \mathcal{Y} \triangleq \mathbb{R}$



Learning model #2

COMPONENTS OF SUPERVISED MACHINE LEARNING

1. An *unknown function* $f : \mathcal{X} \to \mathcal{Y} : \mathbf{x} \mapsto y = f(\mathbf{x})$ to learn
2. A *dataset* $\mathcal{D} \triangleq \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$
   - $\{\mathbf{x}_i\}_{i=1}^N$ i.i.d. *from unknown distribution* $P_\mathbf{x}$ on $\mathcal{X}$
   - $\{y_i\}_{i=1}^N$ are the corresponding labels $y_i \in \mathcal{Y} \triangleq \mathbb{R}$
3. A *set of hypotheses* $\mathcal{H}$ as to what the function could be
4. An *algorithm* $\mathtt{ALG}$ to find the best $h \in \mathcal{H}$ that explains f



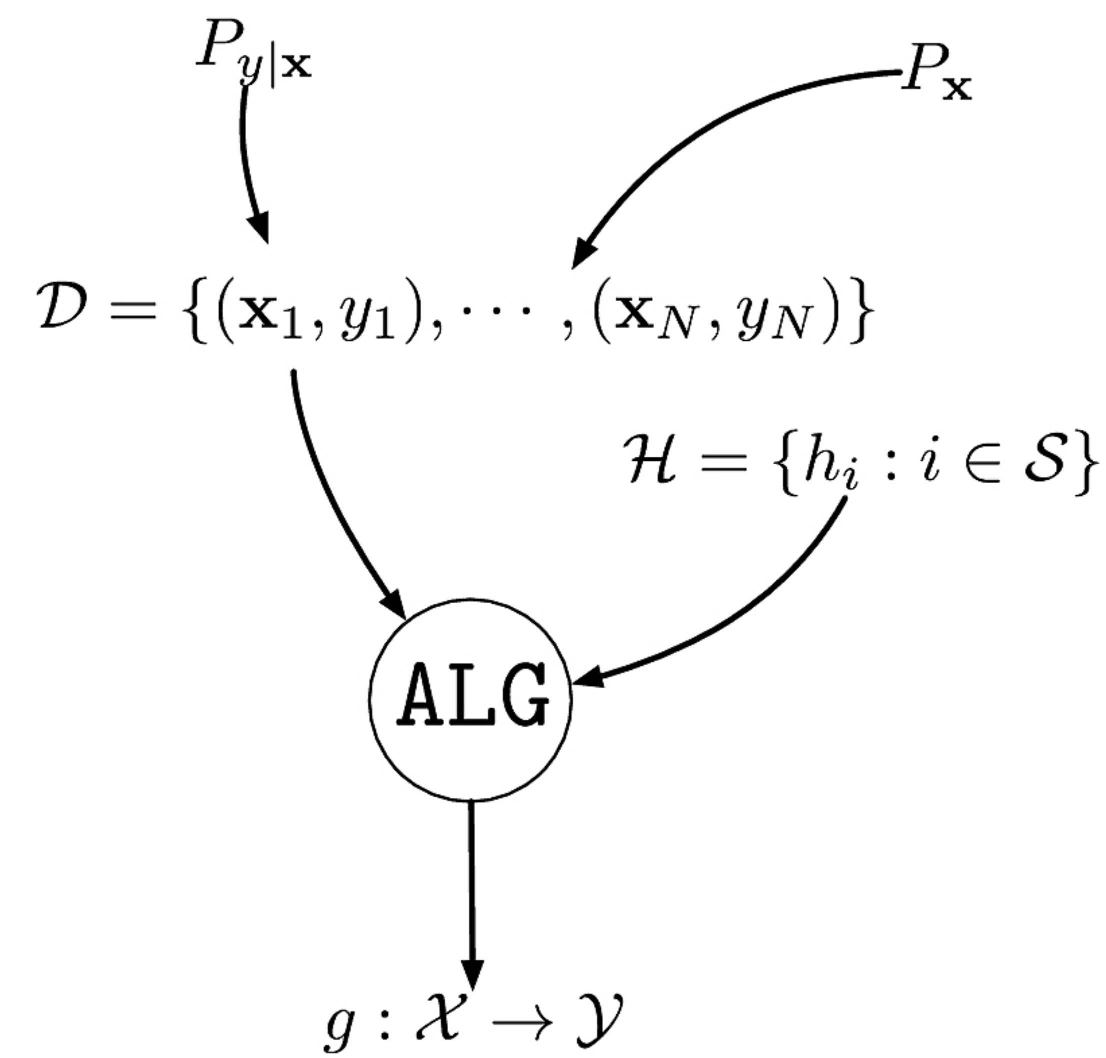
Learning model #2

ANOTHER LEARNING PUZZLE

Which color is the dress?

COMPONENTS OF SUPERVISED MACHINE LEARNING

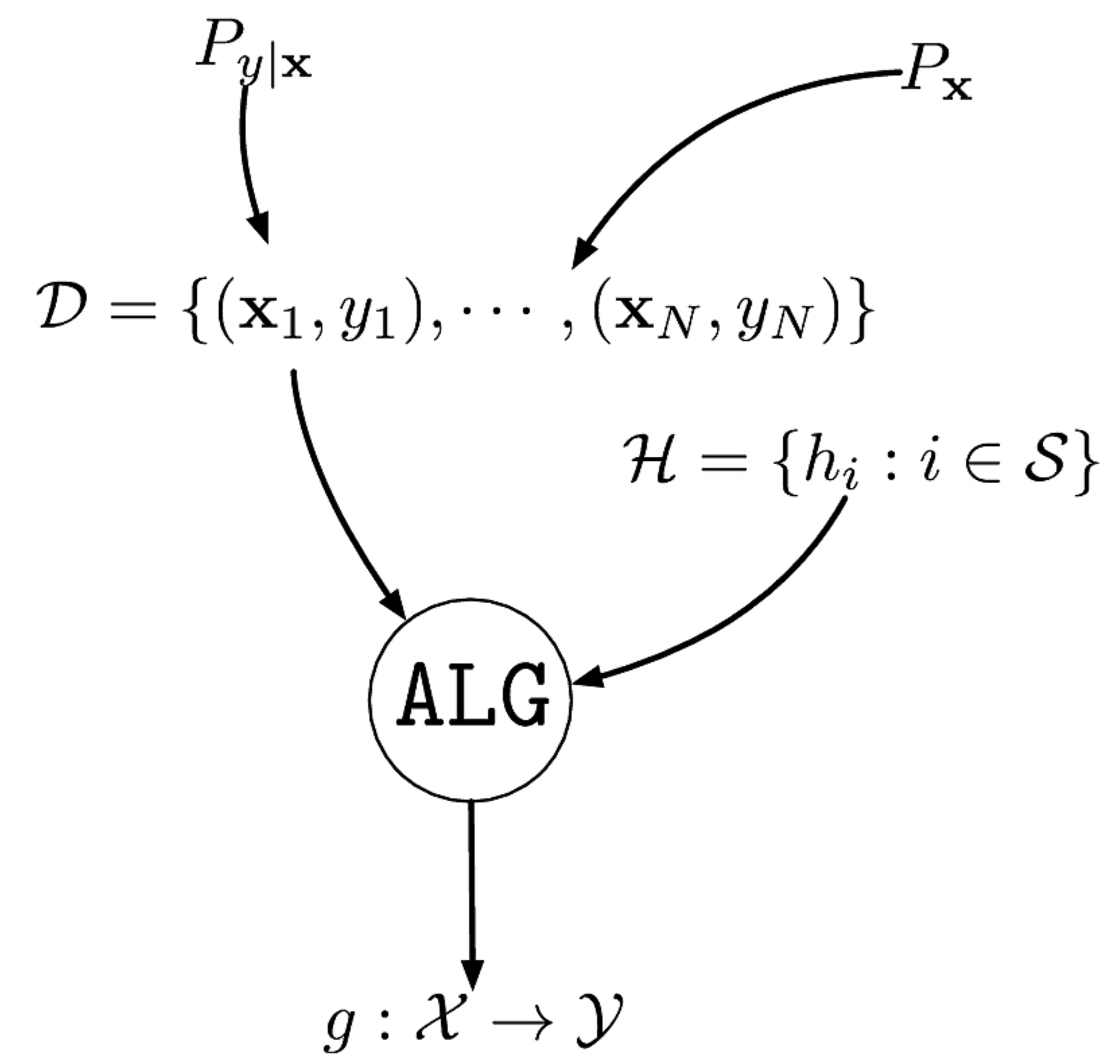
1. An *unknown conditional distribution* $P_{y|\mathbf{x}}$ to learn
   - $\boxed{P_{y|\mathbf{x}}}$ models $f : \mathcal{X} \to \mathcal{Y}$ *with noise*



Learning model #3

COMPONENTS OF SUPERVISED MACHINE LEARNING

1. An *unknown conditional distribution* $P_{y|\mathbf{x}}$ to learn
   - $P_{y|\mathbf{x}}$ models $f : \mathcal{X} \to \mathcal{Y}$ *with noise*
2. A *dataset* $\mathcal{D} \triangleq \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$
   - $\{\mathbf{x}_i\}_{i=1}^N$ i.i.d. from distribution $\boxed{P_{\mathbf{x}}}$ on $\mathcal{X}$
   - $\{y_i\}_{i=1}^N$ are the corresponding labels $y_i \sim \boxed{P_{y|\mathbf{x}=\mathbf{x}_i}}$
3. A *set of hypotheses* $\mathcal{H}$ as to what the function could be
4. An *algorithm* **ALG** to find the best $h \in \mathcal{H}$ that explains f



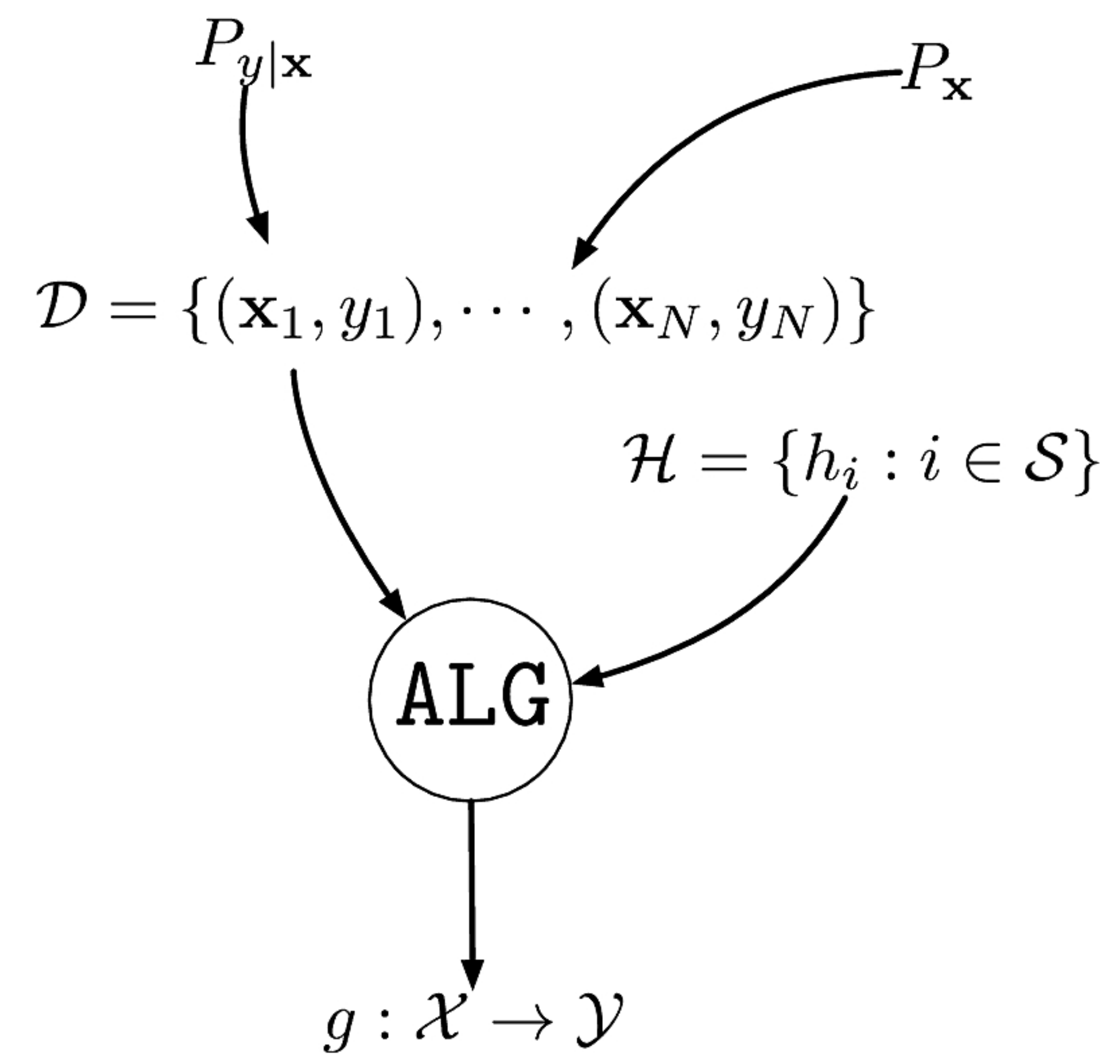
Learning model #3

COMPONENTS OF SUPERVISED MACHINE LEARNING

1. An *unknown conditional distribution* $P_{y|\mathbf{x}}$ to learn
   - $P_{y|\mathbf{x}}$ models $f : \mathcal{X} \to \mathcal{Y}$ *with noise*
2. A *dataset* $\mathcal{D} \triangleq \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$
   - $\{\mathbf{x}_i\}_{i=1}^N$ i.i.d. from distribution $P_{\mathbf{x}}$ on $\mathcal{X}$
   - $\{y_i\}_{i=1}^N$ are the corresponding labels $y_i \sim P_{y|\mathbf{x}=\mathbf{x}_i}$
3. A *set of hypotheses* $\mathcal{H}$ as to what the function could be
4. An *algorithm* **ALG** to find the best $h \in \mathcal{H}$ that explains f

The roles of $P_{y|\mathbf{x}}$ and $P_{\mathbf{x}}$ are *different*

- $P_{y|\mathbf{x}}$ is what we want to learn, captures the underlying function and the noise added to it
- $P_{\mathbf{x}}$ models *sampling* of dataset, need *not* be learned



Learning model #3

YET ANOTHER LEARNING PUZZLE

Assume that you are designing a fingerprint authentication system

- You trained your system with a fancy machine learning system
- The probability of wrongly authenticating is *1%*
- The probability of correctly authenticating is *60%*
- Is this a good system?

It depends!

- If you are GTRI, this might be ok (security matters more)
- If you are Apple, this is not acceptable (convenience matters more)

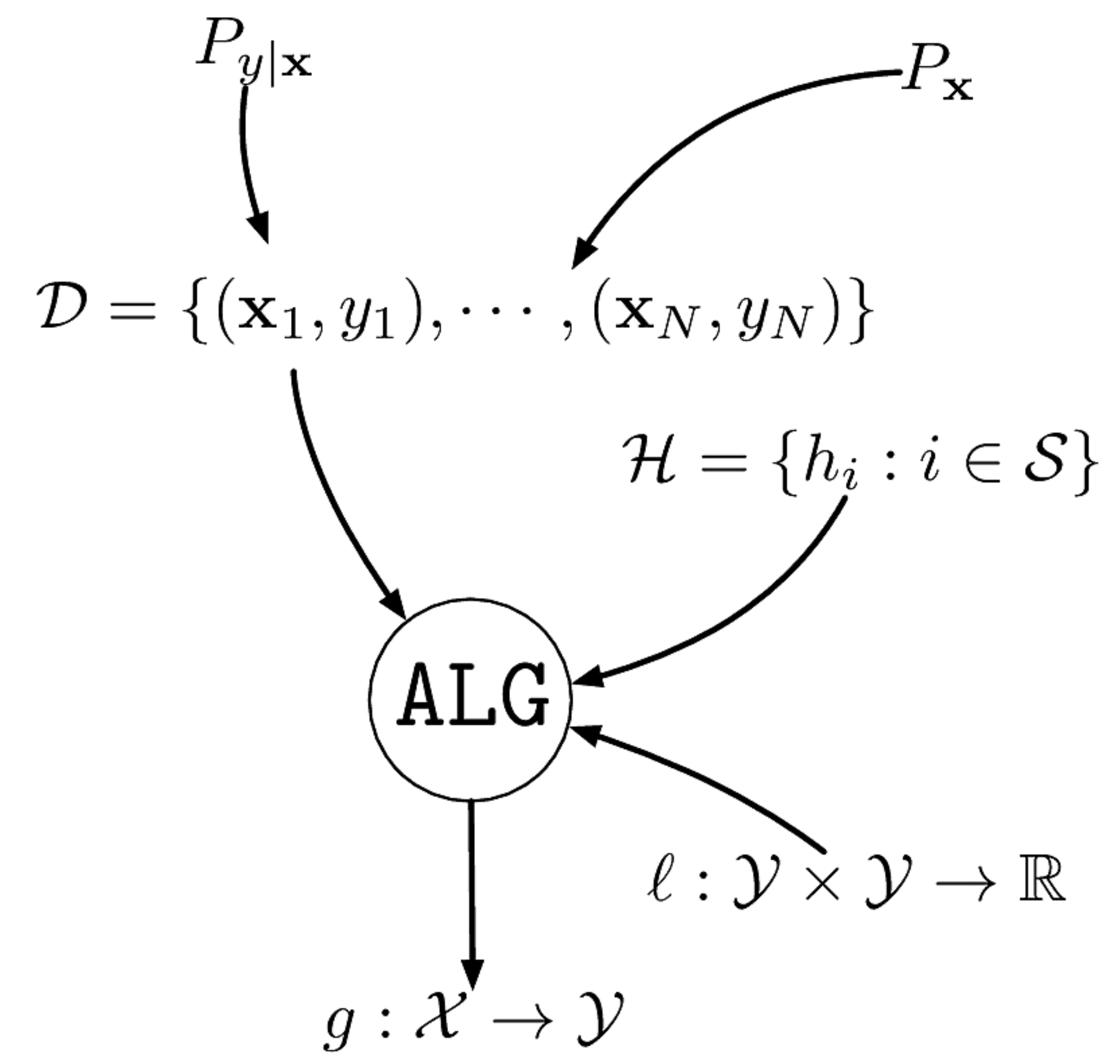
There is an application dependent *cost* that can affect the design

Biometric authentication system

COMPONENTS OF SUPERVISED MACHINE LEARNING

1. A *dataset* $\mathcal{D} \triangleq \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$
   - $\{\mathbf{x}_i\}_{i=1}^N$ i.i.d. from an unknown distribution $P_{\mathbf{x}}$ on $\mathcal{X}$
2. An *unknown conditional distribution* $P_{y|\mathbf{x}}$
   - $P_{y|\mathbf{x}}$ models $f : \mathcal{X} \to \mathcal{Y}$ *with noise*
   - $\{y_i\}_{i=1}^N$ are the corresponding labels $y_i \sim P_{y|\mathbf{x}=\mathbf{x}_i}$
3. A *set of hypotheses* $\mathcal{H}$ as to what the function could be
4. A *loss function* $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+$ capturing the "cost" of prediction

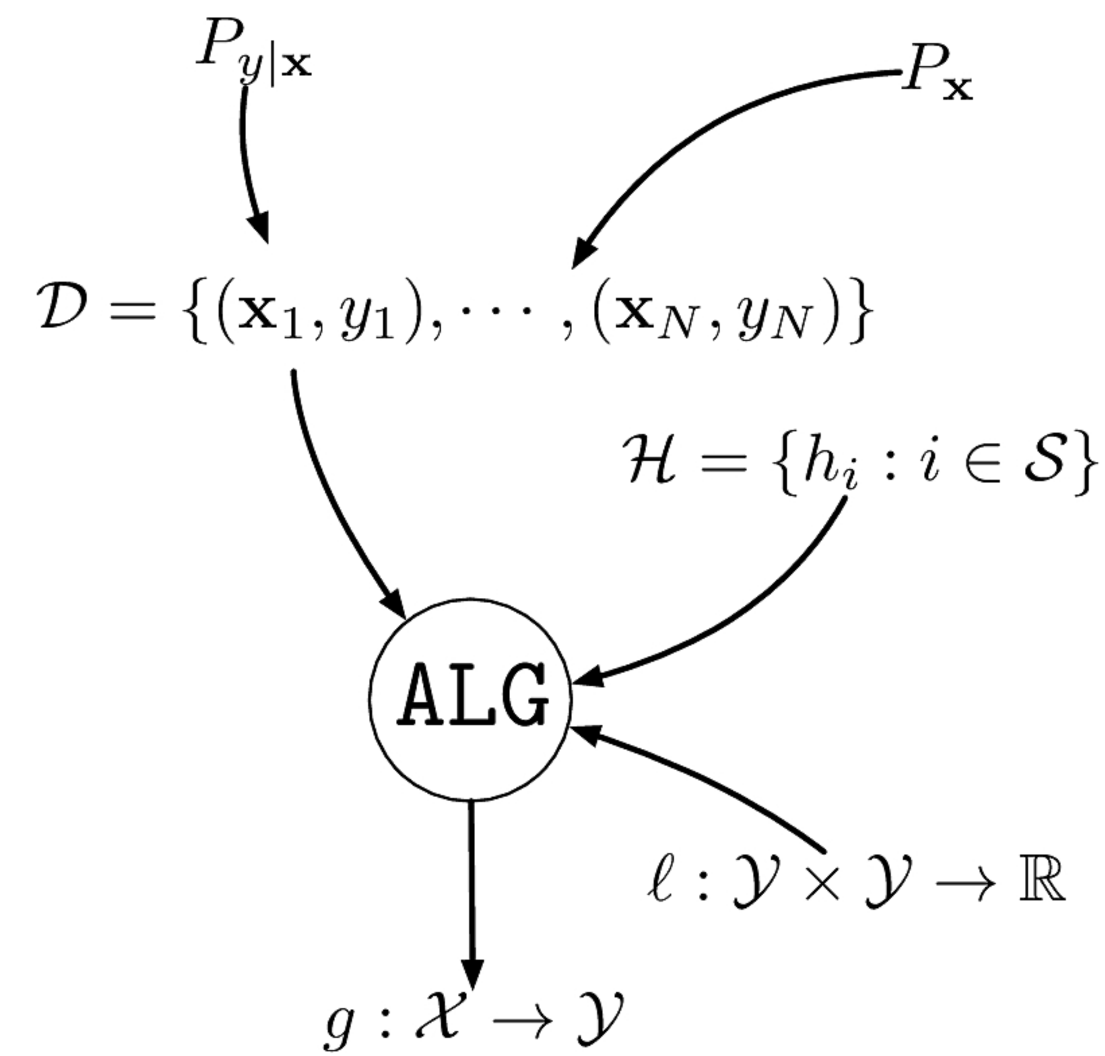
$\ell(g(x), y) \geq 0$ indicates the prediction quality



Final supervised learning model

COMPONENTS OF SUPERVISED MACHINE LEARNING

1. A *dataset* $\mathcal{D} \triangleq \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$
   - $\{\mathbf{x}_i\}_{i=1}^N$ i.i.d. from an unknown distribution $P_{\mathbf{x}}$ on $\mathcal{X}$
2. An *unknown conditional distribution* $P_{y|\mathbf{x}}$
   - $P_{y|\mathbf{x}}$ models $f : \mathcal{X} \to \mathcal{Y}$ *with noise*
   - $\{y_i\}_{i=1}^N$ are the corresponding labels $y_i \sim P_{y|\mathbf{x}=\mathbf{x}_i}$
3. A *set of hypotheses* $\mathcal{H}$ as to what the function could be
4. A *loss function* $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+$ capturing the “cost” of prediction
5. An *algorithm* $\mathtt{ALG}$ to find the best $h \in \mathcal{H}$ that explains f



Final supervised learning model

THE SUPERVISED LEARNING PROBLEM

Learning is not *memorizing*

- Our goal is *not* to find $h \in \mathcal{H}$ that accurately assigns values to elements of $\mathcal{D}$

$h : \mathcal{X} \longmapsto \mathcal{Y} : x_i \longmapsto y_i$ memorizing

knowing $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}$

THE SUPERVISED LEARNING PROBLEM

Learning is not *memorizing*

- Our goal is *not* to find $h \in \mathcal{H}$ that accurately assigns values to elements of $\mathcal{D}$
- Our goal is to find the *best* $h \in \mathcal{H}$ that accurately *predicts* values of *unseen* samples

Consider hypothesis $h \in \mathcal{H}$. We can easily compute the *empirical risk* (a.k.a. *in-sample* error)

empirical

$$\widehat{R}_N(h) \triangleq \frac{1}{N}\sum_{i=1}^{N} \ell(\underbrace{y_i}_{\text{true value}}, \underbrace{h(\mathbf{x}_i)}_{\text{prediction}})$$

dataset size

THE SUPERVISED LEARNING PROBLEM

Learning is not *memorizing*

- Our goal is *not* to find $h \in \mathcal{H}$ that accurately assigns values to elements of $\mathcal{D}$
- Our goal is to find the *best* $h \in \mathcal{H}$ that accurately *predicts* values of *unseen* samples

Consider hypothesis $h \in \mathcal{H}$. We can easily compute the *empirical risk* (a.k.a. *in-sample* error)

$$\widehat{R}_N(h) \triangleq \frac{1}{N} \sum_{i=1}^{N} \ell(y_i, h(\mathbf{x}_i))$$

What we really care about is the *true risk* (a.k.a. *out-sample* error) $R(h) \triangleq \mathbb{E}_{\mathbf{x}y}[\ell(y, h(\mathbf{x}))]$

random variables

THE SUPERVISED LEARNING PROBLEM

Learning is not *memorizing*

- Our goal is *not* to find $h \in \mathcal{H}$ that accurately assigns values to elements of $\mathcal{D}$
- Our goal is to find the *best* $h \in \mathcal{H}$ that accurately *predicts* values of *unseen* samples

Consider hypothesis $h \in \mathcal{H}$. We can easily compute the *empirical risk* (a.k.a. *in-sample* error)

$$\widehat{R}_N(h) \triangleq \frac{1}{N} \sum_{i=1}^{N} \ell(y_i, h(\mathbf{x}_i))$$

What we really care about is the *true risk* (a.k.a. *out-sample* error) $R(h) \triangleq \mathbb{E}_{\mathbf{xy}}\left[\ell(y, h(\mathbf{x}))\right]$

Question #1: Can we *generalize*?

- For a given h, is $\widehat{R}_N(h)$ close to $R(h)$?

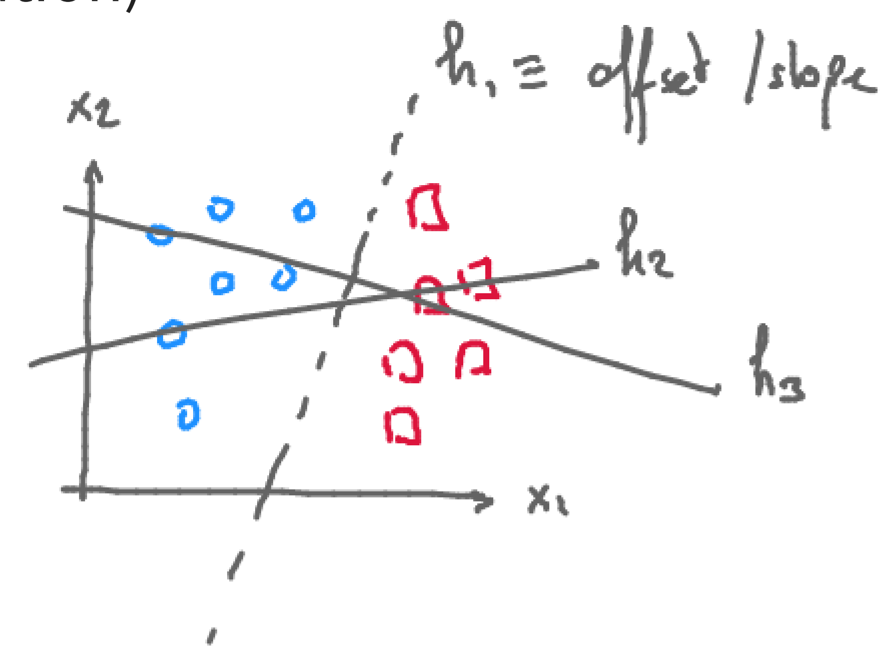
Question #2: Can we learn *well*?

- The *best* hypothesis is $h^\sharp \triangleq \operatorname{argmin}_{h \in \mathcal{H}} R(h)$ but we can only find $h^* \triangleq \operatorname{argmin}_{h \in \mathcal{H}} \widehat{R}_N(h)$
- Is $\widehat{R}_N(h^*)$ close to $R(h^\sharp)$?
- Is $R(h^\sharp) \approx 0$?

A SIMPLER SUPERVISED LEARNING PROBLEM

Consider a special case of the general supervised learning problem

1. Dataset $\mathcal{D} \triangleq \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$
    - $\{\mathbf{x}_i\}_{i=1}^N$ *drawn i.i.d. from unknown* $P_{\mathbf{x}}$ on $\mathcal{X}$
    - $\{y_i\}_{i=1}^N$ labels with $\mathcal{Y} = \{0, 1\}$ (binary classification)
2. Unknown $f : \mathcal{X} \to \mathcal{Y}$, no noise.
3. Finite set of hypotheses $\mathcal{H}$, $|\mathcal{H}| = M < \infty$



A SIMPLER SUPERVISED LEARNING PROBLEM

Consider a special case of the general supervised learning problem

1. Dataset $\mathcal{D} \triangleq \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$
   - $\{\mathbf{x}_i\}_{i=1}^N$ *drawn i.i.d. from unknown* $P_\mathbf{x}$ on $\mathcal{X}$
   - $\{y_i\}_{i=1}^N$ labels with $\mathcal{Y} = \{0, 1\}$ (binary classification)
2. Unknown $f : \mathcal{X} \to \mathcal{Y}$, no noise.
3. Finite set of hypotheses $\mathcal{H}$, $|\mathcal{H}| = M < \infty$
   - $\mathcal{H} \triangleq \{h_i\}_{i=1}^M$
4. Binary loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+ : (y_1, y_2) \mapsto \mathbf{1}\{y_1 \neq y_2\}$

In this very specific case, the true risk simplifies

$$R(h) \triangleq \mathbb{E}_{\mathbf{x}y}\left[\mathbf{1}\{h(\mathbf{x}) \neq y\}\right] = \mathbb{P}_{\mathbf{x}y}\left(h(\mathbf{x}) \neq y\right)$$

The empirical risk becomes

$$\widehat{R}_N(h) = \frac{1}{N}\sum_{i=1}^N \mathbf{1}\{h(\mathbf{x}_i) \neq y_i\}$$

CAN WE LEARN?

Our objective is to find a hypothesis $h^* = \operatorname{argmin}_{h\in\mathcal{H}} \widehat{R}_N(h)$ that ensures a small risk

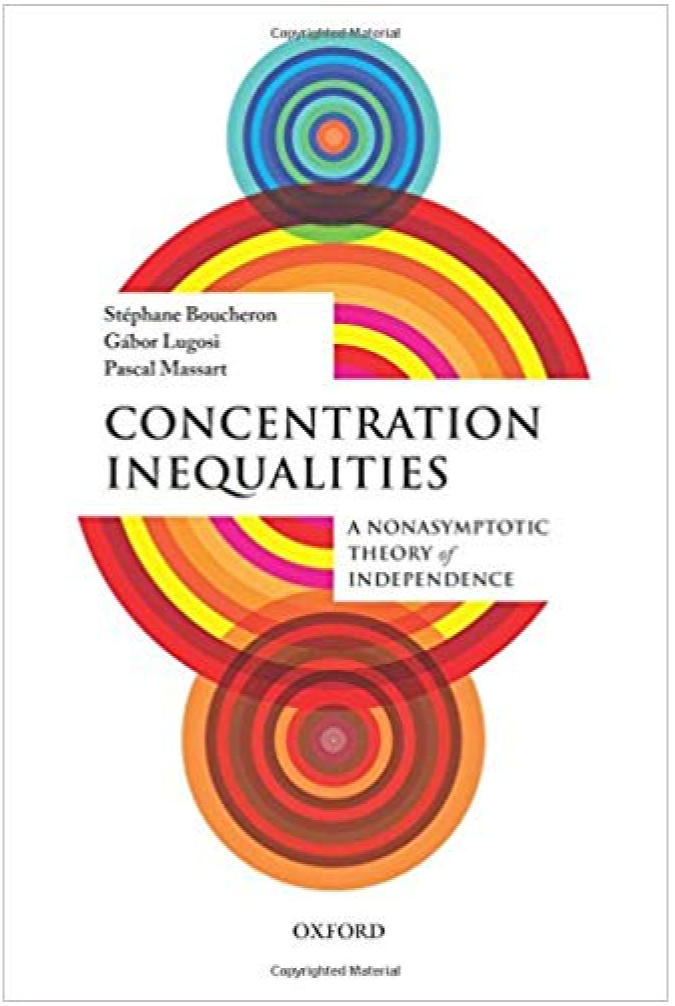
For a *fixed* $h_j \in \mathcal{H}$, how does $\widehat{R}_N(h_j)$ compares to $R(h_j)$?

Observe that for $h_j \in \mathcal{H}$

- The empirical risk is a sum of iid random variables

$$\widehat{R}_N(h_j) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\{h_j(\mathbf{x}_i) \neq y_i\}$$

- $\mathbb{E}_{\mathcal{D}}\left[\widehat{R}_N(h_j)\right] = R(h_j) = \frac{1}{N}\sum_{i=1}^{N} \underbrace{\mathbb{E}_{\mathcal{D}}\left[\mathbb{1}\{h_j(x_i) \neq y_i\}\right]}_{\mathbb{P}_{xy}(h_j(x)\neq y) \,\triangleq\, R(h_j)} = R(h_j)$



CAN WE LEARN?

Our objective is to find a hypothesis $h^* = \mathbf{argmin}_{h \in \mathcal{H}} \widehat{R}_N(h)$ that ensures a small risk

For a *fixed* $h_j \in \mathcal{H}$, how does $\widehat{R}_N(h_j)$ compares to $R(h_j)$?

Observe that for $h_j \in \mathcal{H}$

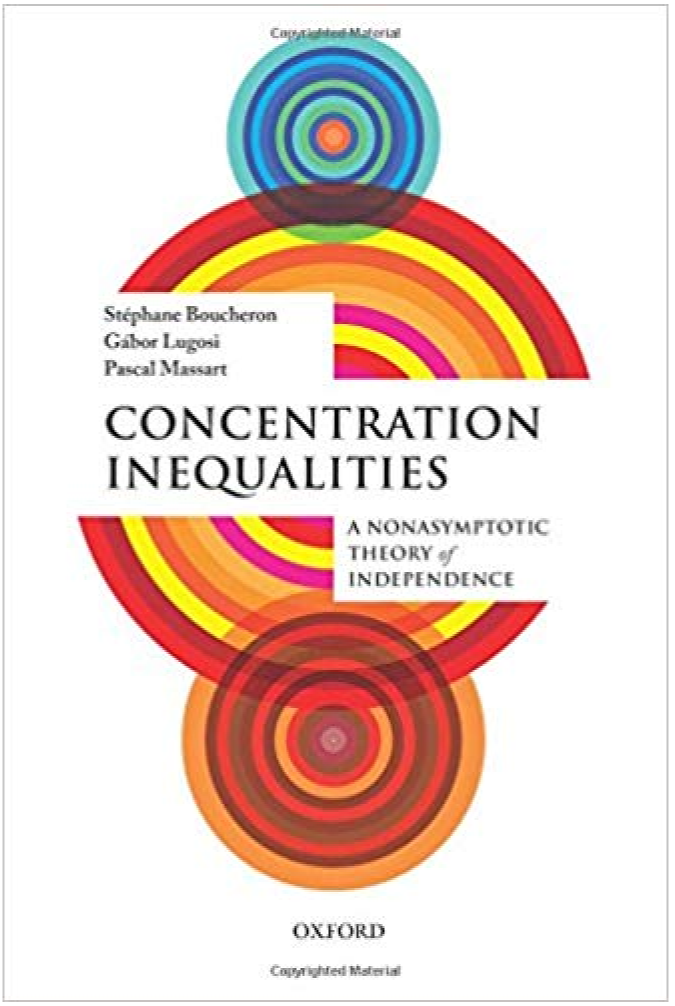
- The empirical risk is a sum of iid random variables

$$\widehat{R}_N(h_j) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}\{h_j(\mathbf{x}_i) \neq y_i\}$$

- $\mathbb{E}\left[\widehat{R}_N(h_j)\right] = R(h_j)$

$\mathbb{P}\left(\left|\widehat{R}_N(h_j) - R(h_j)\right| > \epsilon\right)$ is a statement about the deviation of a normalized sum of iid random variables from its mean

We're in luck! Such bounds, a.k.a, known as *concentration inequalities*, are a well studied subject

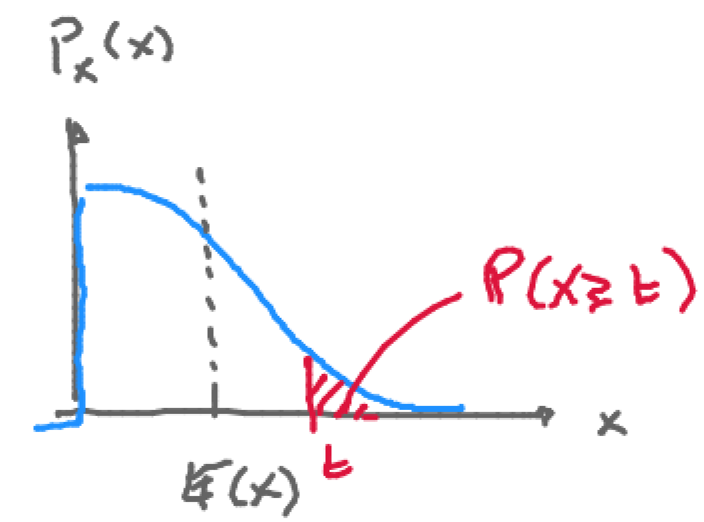


CONCENTRATION INEQUALITIES: BASICS

Lemma (Markov's inequality)

Let X be a *non-negative* real-valued random variable. Then for all $t > 0$

$$\mathbb{P}\left(X \geq t\right) \leq \frac{\mathbb{E}\left[X\right]}{t}.$$



Proof:

$$\mathbb{E}(X) = \mathbb{E}\Big(X\big(\underbrace{\mathbb{1}\{X\geq t\} + \mathbb{1}\{X<t\}}_{=1}\big)\Big)$$

$$= \underbrace{\mathbb{E}(X\,\mathbb{1}\{X\geq t\})}_{(*)\ \geq\ t\,\mathbb{E}(\mathbb{1}\{X\geq t\})\ =\ t\,\mathbb{P}(X\geq t)} + \underbrace{\mathbb{E}(X\,\mathbb{1}\{X<t\})}_{\geq 0}$$

$$\geq t\,\mathbb{P}(X\geq t)$$

$$\mathbb{E}(X) = \int_0^{+\infty} p_X(x)\, x\, dx$$

$$= \underbrace{\int_0^{t} p_X(x)\, x\, dx}_{\geq 0} + \underbrace{\int_t^{+\infty} p_X(x)\, x\, dx}_{\geq \int_t^{+\infty} t\, p_X(x)\, dx\ =\ t \int_t^{+\infty} p_X(x)\ =\ t\,\mathbb{P}(X\geq t}$$

CONCENTRATION INEQUALITIES: BASICS

Lemma (Markov's inequality)

Let X be a *non-negative* real-valued random variable. Then for all $t > 0$

$$\mathbb{P}(X \geq t) \leq \frac{\mathbb{E}[X]}{t}.$$

Lemma (Chebyshev's inequality)

Let X be a real-valued random variable. Then for all $t > 0$

$$\mathbb{P}(|X - \mathbb{E}[X]| \geq t) \leq \frac{\mathrm{Var}(X)}{t^2}.$$

Proof: set $Y = (X - \mathbb{E}(X))^2$; Y is non negative

$\mathbb{P}(Y \geq t^2) \leq \frac{\mathbb{E}(Y)}{t^2}$ note $\mathbb{E}(Y) \overset{\Delta}{=} \mathbb{E}((X - \mathbb{E}(X))^2) \overset{\Delta}{=} \mathrm{Var}(X)$

$Y \geq t \Leftrightarrow (X - \mathbb{E}(X))^2 \geq t^2 \Leftrightarrow |X - \mathbb{E}(X)| \geq t \quad (t > 0)$ ∎

CONCENTRATION INEQUALITIES: BASICS

Lemma (Markov's inequality)

Let X be a *non-negative* real-valued random variable. Then for all $t > 0$

$$\mathbb{P}\left(X \geq t\right) \leq \frac{\mathbb{E}\left[X\right]}{t}.$$

Lemma (Chebyshev's inequality)

Let X be a real-valued random variable. Then for all $t > 0$

$$\mathbb{P}\left(\left|X - \mathbb{E}\left[X\right]\right| \geq t\right) \leq \frac{\mathrm{Var}\left(X\right)}{t^2}.$$

Proposition (Weak law of large numbers)

Let $\{X_i\}_{i=1}^N$ be i.i.d. real-valued random variables with finite mean μ and finite variance σ^2. Then

$$\mathbb{P}\left(\left|\underbrace{\frac{1}{N}\sum_{i=1}^N X_i}_{\text{empirical mean}} - \overset{\text{true mean}}{\mu}\right| \geq \epsilon\right) \leq \frac{\sigma^2}{N\epsilon^2} \qquad \lim_{N\to\infty} \mathbb{P}\left(\left|\frac{1}{N}\sum_{i=1}^N X_i - \mu\right| \geq \epsilon\right) = 0.$$

Proof: apply Chebyshev to $\frac{1}{N}\sum_{i=1}^{N} X_i$

$$\mathbb{E}\left(\frac{1}{n}\sum_{i=1}^{N} X_i\right) = \frac{1}{N}\sum_{i=1}^{N} \underbrace{\mathbb{E}(X_i)}_{\mu} = \mu \qquad \mathrm{Var}\left(\frac{1}{N}\sum_{i=1}^{N} X_i\right) = \frac{1}{N^2}\,\mathrm{Var}\left(\sum_{i=1}^{N} X_i\right) = \frac{1}{N^2}\sum_{i=1}^{N} \sigma^2 = \frac{\sigma^2}{N} \qquad \blacksquare$$

BACK TO LEARNING

By the law of large number, we know that

hypothesis fixed

$$\forall\epsilon>0 \quad \mathbb{P}_{\{(\mathbf{x}_i,y_i)\}}\left(\left|\widehat{R}_N(h_j)-R(h_j)\right|\ge\epsilon\right)\le \frac{\mathrm{Var}\left(\mathbf{1}\{h_j(\mathbf{x}_1)\neq y_1\}\right)}{N\epsilon^2}\le\frac{1}{N\epsilon^2}$$

$\mathcal{D}$

$\triangleq \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}\{h_j(x_i)\neq y_i\}$

BACK TO LEARNING

By the law of large number, we know that

$$\forall \epsilon > 0 \quad \underbrace{\mathbb{P}_{\{(\mathbf{x}_i, y_i)\}}\left(\left|\widehat{R}_N(h_j) - R(h_j)\right| \geq \epsilon\right)}_{\leq \delta} \leq \frac{\operatorname{Var}\left(\mathbf{1}\{h_j(\mathbf{x}_1) \neq y_1\}\right)}{N\epsilon^2} \leq \frac{1}{N\epsilon^2} \leq \delta$$

Given enough data, we can *generalize*

How much data? $N = \frac{1}{\delta\epsilon^2}$ to ensure $\mathbb{P}\left(\left|\widehat{R}_N(h_j) - R(h_j)\right| \geq \epsilon\right) \leq \delta.$

BACK TO LEARNING

By the law of large number, we know that

$$\forall \epsilon > 0 \quad \mathbb{P}_{\{(\mathbf{x}_i, y_i)\}}\left(\left|\widehat{R}_N(h_j) - R(h_j)\right| \geq \epsilon\right) \leq \frac{\mathrm{Var}\left(\mathbf{1}\{h_j(\mathbf{x}_1) \neq y_1\}\right)}{N\epsilon^2} \leq \frac{1}{N\epsilon^2}$$

Given enough data, we can *generalize*

How much data? $N = \frac{1}{\delta\epsilon^2}$ to ensure $\mathbb{P}\left(\left|\widehat{R}_N(h_j) - R(h_j)\right| \geq \epsilon\right) \leq \delta.$

That's not quite enough! We care about $\widehat{R}_N(h^*)$ where $h^* = \mathrm{argmin}_{h \in \mathcal{H}} \widehat{R}_N(h)$

- If $M = |\mathcal{H}|$ is large we should expect the existence of $h_k \in \mathcal{H}$ such that $\widehat{R}_N(h_k) \ll R(h_k)$

$$\mathbb{P}\left(\left|\widehat{R}_N(h^*) - R(h^*)\right| \geq \epsilon\right) \leq \mathbb{P}\left(\exists j : \left|\widehat{R}_N(h_j) - R(h_j)\right| \geq \epsilon\right)$$

$$\mathbb{P}\left(\left|\widehat{R}_N(h^*) - R(h^*)\right| \geq \epsilon\right) \leq \frac{M}{N\epsilon^2}$$

If we choose $N \geq \lceil \frac{M}{\delta\epsilon^2} \rceil$ we can ensure $\mathbb{P}\left(\left|\widehat{R}_N(h^*) - R(h^*)\right| \geq \epsilon\right) \leq \delta.$

- That's a lot of samples!

CONCENTRATION INEQUALITIES: NOT SO BASIC

We can obtain *much* better bounds than with Chebyshev

> **Lemma (Hoeffding's inequality)**
>
> Let $\{X_i\}_{i=1}^N$ be i.i.d. real-valued zero-mean random variables such that $X_i \in [a_i; b_i]$ with $a_i < b_i$. Then for all $\epsilon > 0$
>
> $$\mathbb{P}\left(\left|\frac{1}{N}\sum_{i=1}^N X_i\right| \geq \epsilon\right) \leq 2\exp\left(-\frac{2N^2\epsilon^2}{\sum_{i=1}^N (b_i - a_i)^2}\right).$$

In our learning problem

$$\forall \epsilon > 0 \quad \mathbb{P}\left(\left|\widehat{R}_N(h_j) - R(h_j)\right| \geq \epsilon\right) \leq 2\exp(-2N\epsilon^2)$$

$$\forall \epsilon > 0 \quad \mathbb{P}\left(\left|\widehat{R}_N(h^*) - R(h^*)\right| \geq \epsilon\right) \leq 2M\exp(-2N\epsilon^2)$$

We can now choose $N \geq \lceil \frac{1}{2\epsilon^2}\left(\ln \frac{2M}{\delta}\right)\rceil$

M can be quite large (almost exponential in N) and, with enough data, we can generalize h^*.

How about learning $h^\sharp \triangleq \text{argmin}_{h\in\mathcal{H}} R(h)$?

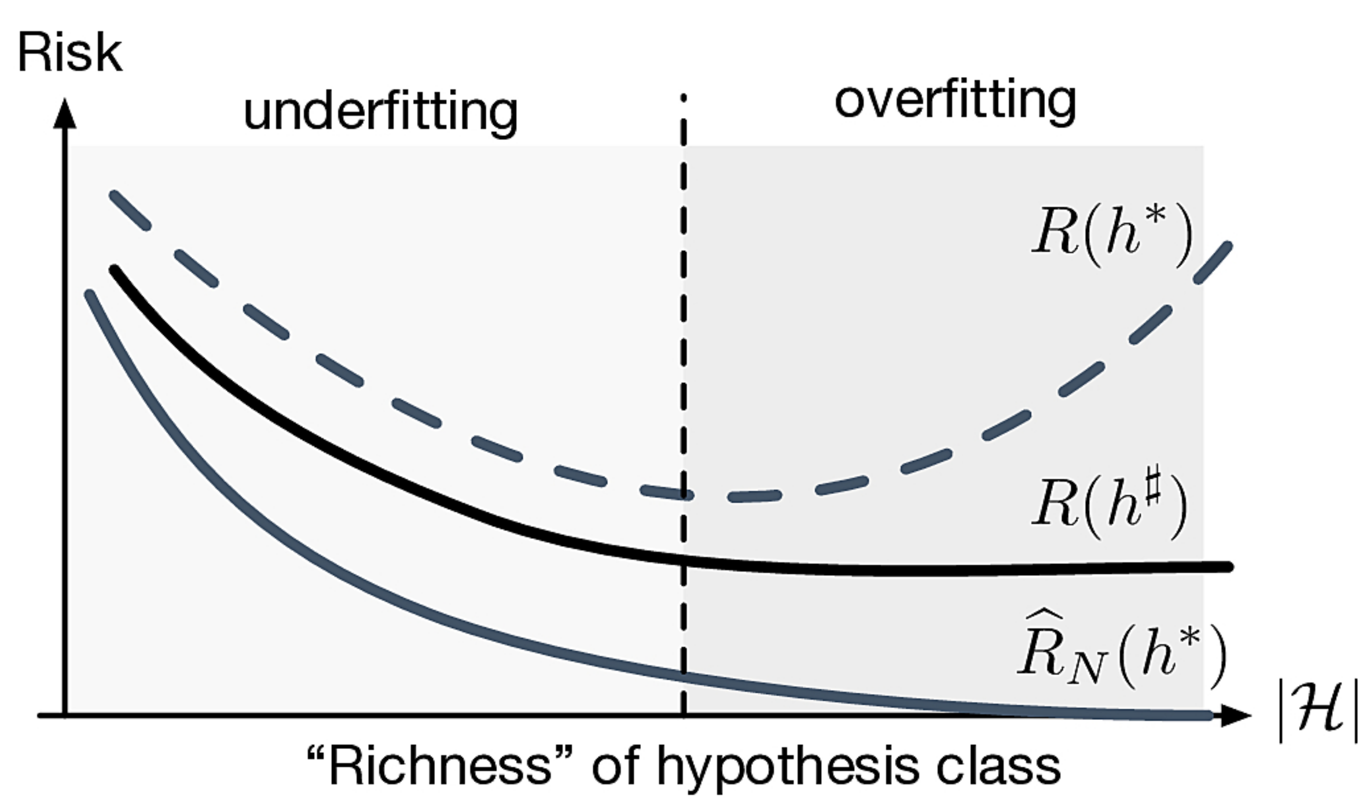
LEARNING CAN WORK!

Lemma.

If $\forall j \in \mathcal{H}$ $\left|\widehat{R}_N(h_j) - R(h_j)\right| \leq \epsilon$ then $\left|R(h^*) - R(h^\sharp)\right| \leq 2\epsilon.$

How do we make $R(h^\sharp)$ small?

- Need bigger hypothesis class $\mathcal{H}$! (could we take $M \to \infty$?)
- Fundamental trade-off of learning



PROBABLY APPROXIMATELY CORRECT LEARNABILITY

Definition. (PAC learnability)
A hypothesis set $\mathcal{H}$ is (agnostic) PAC learnable if there exists a function $N_{\mathcal{H}} :]0;1[^2 \to \mathbb{N}$ and a learning algorithm such that:

- for very $\epsilon, \delta \in]0;1[$,
- for every $P_{\mathbf{x}}, P_{y|\mathbf{x}}$,
- when running the algorithm on at least $N_{\mathcal{H}}(\epsilon, \delta)$ i.i.d. examples, the algorithm returns a hypothesis $h \in \mathcal{H}$ such that

$$\mathbb{P}_{\mathbf{x}y}\left(\left|R(h) - R(h^\sharp)\right| \leq \epsilon\right) \geq 1 - \delta$$

The function $N_{\mathcal{H}}(\epsilon, \delta)$ is called *sample complexity*

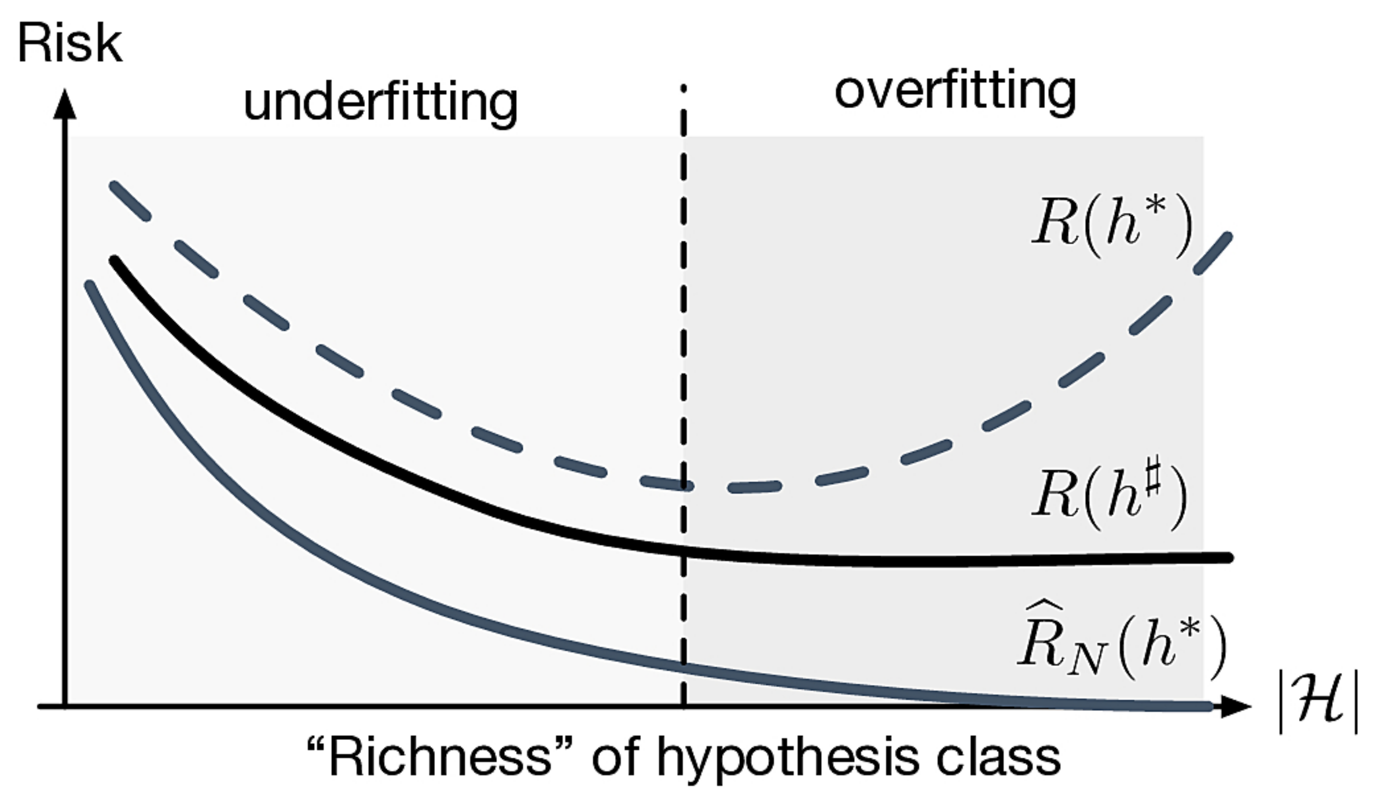
We have effectively already proved the following result

Proposition.

A finite hypothesis set $\mathcal{H}$ is PAC learnable with the Empirical Risk Minimization algorithm and with sample complexity

$$N_{\mathcal{H}}(\epsilon, \delta) = \lceil \frac{2\ln(2|\mathcal{H}|/\delta)}{\epsilon^2} \rceil$$

WHAT IS A GOOD HYPOTHESIS SET?



Ideally we want $|\mathcal{H}|$ small so that $R(h^*) \approx R(h^\sharp)$ and get lucky so that $R(h^*) \approx 0$

In general this is *not* possible

Remember, we usually have to learn $P_{y|\mathbf{x}}$, not a function f

Questions

- What is the optimal binary classification hypothesis class?
- How small can $R(h^*)$ be?