Mathematical Tools in Machine Learning

Fadoua Balabdaoui

Seminar für Statistik, ETH

19 septembre 2019

Lecture 1

A general overview

Introduction to Statistical Learning

Towards a formal model

Overfitting and a way to remedy

Types of learning : PAC and Agnostic PAC learning

Lecture 1

A general overview

Introduction to Statistical Learning

Towards a formal model

Overfitting and a way to remedy

Types of learning : PAC and Agnostic PAC learning

Some organizational matters

- Lectures : Thursdays 10h-12h, HG E 5
- Lecturer : Fadoua Balabdaoui
  e-mail : fadoua.balabdaoui@stat.math.ethz.ch
- Teaching assistant : Loris Michel
  e-mail : michel@stat.math.ethz.ch
- Assignments : 4 sessions of in-class exercises (1st one : 03/10/2019)
- Website : https ://stat.ethz.ch/lectures/#overview
  Slides, assignments, etc
- Textbook : Understanding Machine Learning : From Theory to Algorithms, by S. Shalev-Shwartz and S. Ben-David (available online at the ETH electronic library)

Good to know

- Evaluation : a written exam of 120 minutes
- Content : theoretical (with more numerical examples/illustrations during the exercise sessions)
- Prerequisites : Probability Theory, Mathematical Statistics
  Notions about algorithms, empirical processes might be helpful (but not necessary)

Other references

- "**Neural Network Learning : Theoretical Foundations**", by Martin Anthony and Peter Bartlett (available online at the ETH electronic library)
- Lecture notes on "**Mathematics of Machine Learning**", by Phillipe Rigollet (available at the MIT OpenCourseWare)
- YouTube lectures on "**Learning from Data**", by Yaser Abu Mostafa (Caltech)

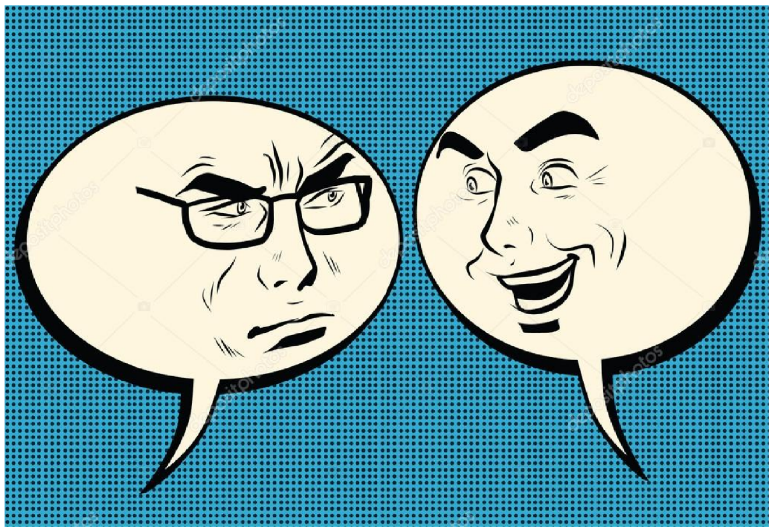
Content

We will mainly cover

- Learning Models (chapters 1-3)
- Learning via Uniform Convergence (chapter 4)
- Linear Predictors (chapter 9)
- The Bias-Complexity Trade-off (chapter 5)
- VC-classes and the VC dimension (chapter 6)
- Model Selection and Validation (chapter 11)
- Convex Learning Problems (chapter 12)
- Regularization and Stability (chapter 13)
- Stochastic Gradient Descent (chapter 14)
- Support Vector Machines (chapter 15)
- Kernels (chapter 16)

Some facts

- There is a significant overlap with Statistics, Inference Theory, Empirical Processes, ...
- Many things are viewed from the standpoint of computer scientists.



Some facts

- Different terminology : Training data instead of **random sample**
  Algorithm instead of **estimator**

- Different notation :

$$\mathbb{E}_{x\sim\mathcal{D}}\big(g(x)\big)$$

instead of

$$\mathbb{E}_{\mathcal{D}}\big(g(X)\big).$$

Lecture 1

A general overview

Introduction to Statistical Learning

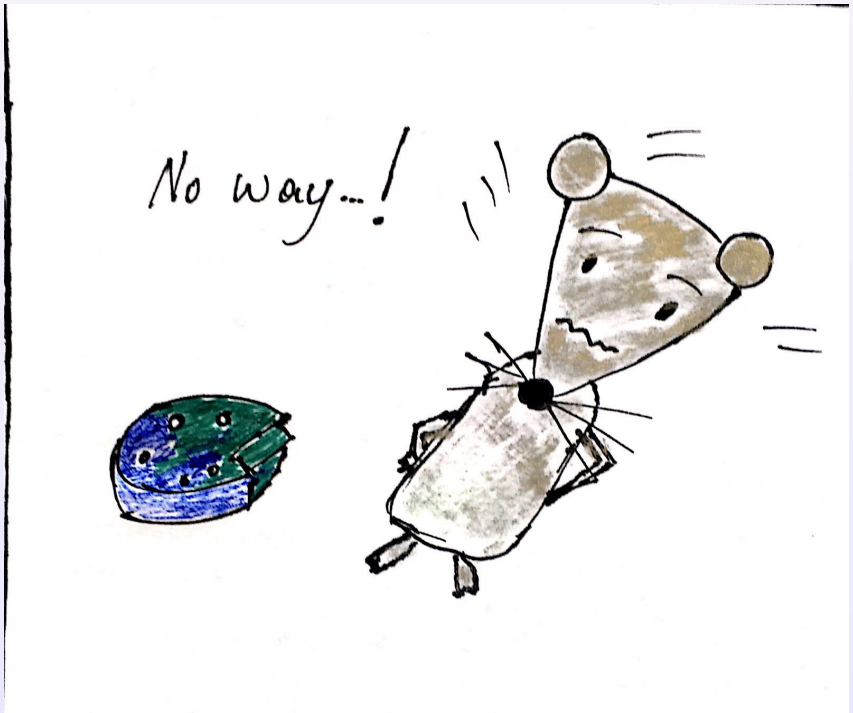
Towards a formal model

Overfitting and a way to remedy

Types of learning : PAC and Agnostic PAC learning

What is learning ? A first example

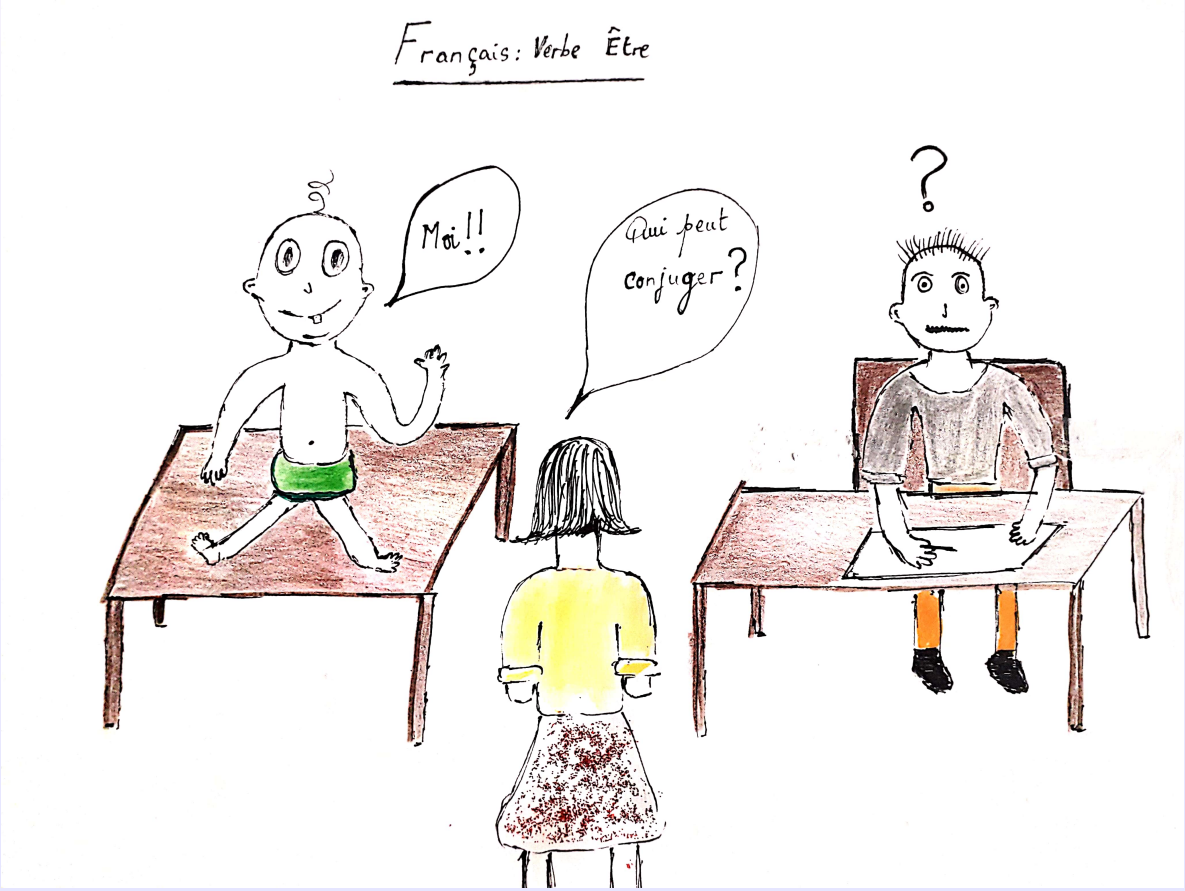
• **Example 1 - Bait shyness** : how does a mouse react when presented some new food ?



If food causes illness, the mouse will **associate** it with a **bad experience**
⇒ The mouse will **not** eat a food with a **similar** texture, smell, etc..

What is learning ? Example 2

• **Example 2 - Learning a foreign language** : It is known that children learn and speak a foreign language **effortless**.



What is learning ? Example 2

• In their early life, humans use overlapping brain areas (Broca's and Wernicke's) to learn languages ⇒ **increased efficiency**.

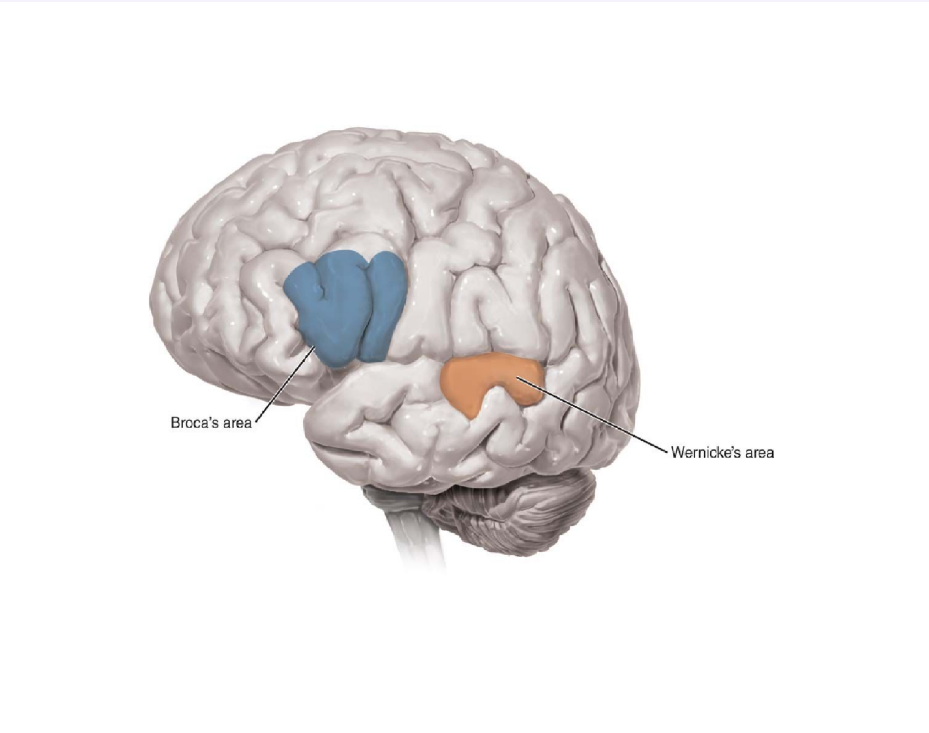


FIGURE – The Broca's (blue) and Wernicke's (orange) areas.

What is learning ? More examples

- **Example 3 - Filtering Spam** :

⋆ Imagine a Spam filter "machine" which **memorizes** several examples of e-mails already **labeled** as Spam.

⋆ Will this machine be able to **correctly classify** this e-mail ?

- **Example 4 - how does a papaya taste ?**

⋆ You just arrived to New Guinea, and you **don't know** how a papaya tastes.

⋆ To **predict** the taste, you decide to restrict yourself to the **features** : color and softness.

What is learning ? Example 5

- **Age quiz** : Guess the age of some individual
- **Successful** guessing : $|\text{guess} - \text{true age}| \leq 4$

FIGURE – J. Gordon-Levitt (38)

Age quiz

FIGURE – I. Elba (46)

Age quiz

FIGURE – M. Kunis (35)

Age quiz

FIGURE – J. Garner (47)

Age quiz

FIGURE – Eric Bana (51)

Age quiz

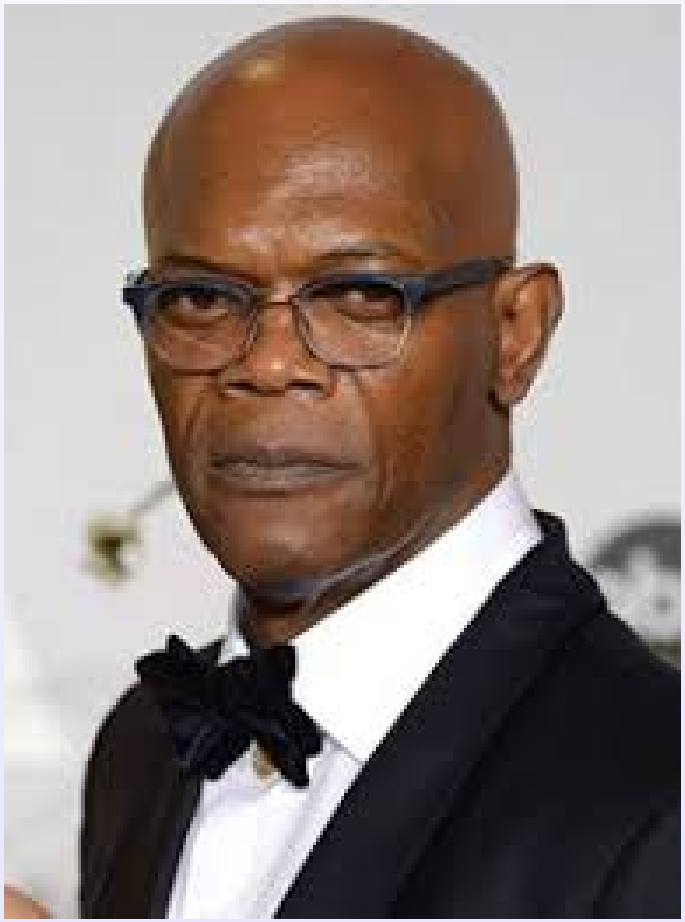


FIGURE – Samuel Jackson (70)

What is learning ? Foundations

- Are there **better** Spam filters ?
- Very informally, the main ingredients in any learning endeavor are :

⋆ A **task** to be learned

⋆ **Features** : can be thought as the main variables used to learn the task

⋆ A **learning machine** : an algorithm,...

⋆ A value for the **Error/Risk** : to assess the quality of learning

Lecture 1

A Learning Model (classification)

- In a statistical learning setting, we have
  - The learner's **input**
  - The learner's **output**

The input :

Domain set	$\mathcal{X}$	set of objects "x" which we want to label
Label set	$\mathcal{Y}$	set of labels "y" : $\{0,1\}$ or $\{-1,1\}$
Training data	$S \in \mathcal{X} \times \mathcal{Y}$	$\Big\{(x_1,y_1),\ldots,(x_m,y_m)\Big\}$

The output :

A prediction rule/hypothesis/classifier, $h : \mathcal{X} \mapsto \mathcal{Y}$

A Learning Model (classification)

Furthermore, we need

- **Data generating model** :
  - A data point $x \in \mathcal{X}$ is assumed to be **generated** from an **unknown** distribution $\mathcal{D}$
  - $y = f(x)$, with $f : \mathcal{X} \mapsto \mathcal{Y}$ gives the **correct labeling**, also **unknown**

Notation : For a measurable set $A \subset \mathcal{X}$, we write

$$\mathcal{D}(A) = \mathbb{P}_{x\sim\mathcal{D}}(x \in A)$$

- **Measure of success** : For a prediction rule h, we define the **prediction error/risk**

$$L_{(D,f)}(h) = \mathbb{P}_{x\sim\mathcal{D}}(h(x) \neq f(x)) = \mathcal{D}(\{x : h(x) \neq f(x)\}).$$

Empirical risk minimization

- Given some training data $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ consider a prediction rule $h_S : \mathcal{X} \mapsto \mathcal{Y}$.
- The true error $L_{(\mathcal{D},f)}(h_S)$ is **unknown** to the learner.
- **But** for this h_S we can compute the **training error**

$$L_S(h_S) = \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}_{\{h_S(x_i) \neq y_i\}} : \quad \text{also the } \textbf{empirical error/risk}.$$

- S is the only "idea" we have about the world : we search for

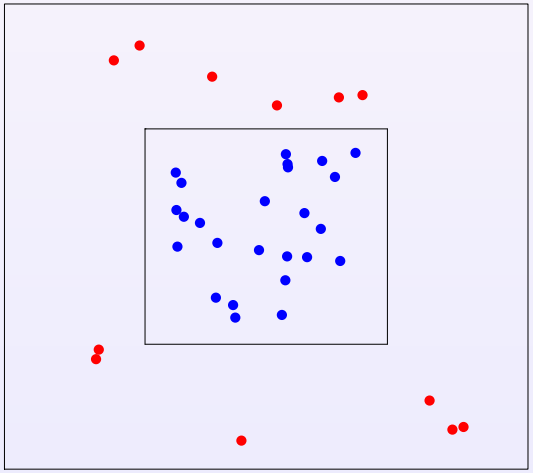
$$h_S = \text{argmin}_h L_S(h)$$

also the **Empirical Risk Minimization (ERM)**.

Lecture 1

"Danger" of overfitting and a possible solution

- Consider the following example



- a point $x \sim \mathcal{D}$: **the uniform distribution** on the larger square,
- the **area** of the larger square is 2 and of the little one is 1,
- the true labeling : 1 inside the inner square and 0 outside.

"Danger" of overfitting and a possible solution

- Let

$$h_S(x) = \begin{cases} y_i & \exists\, x_i : x = x_i \ \text{ for some } i \in [m] \equiv \{1, \ldots, m\} \\ 0 & \text{otherwise.} \end{cases}$$

- Then, $L_S(h_S) = 0$. • Hence, h_S is an ERM predictor, **But**

$$\begin{aligned} L_{(\mathcal{D},f)}(h_S) &= \mathbb{P}_{x\sim\mathcal{D}}\Big(h_S(x) \neq f(x)\Big) \\ &= \mathbb{P}_{x\sim\mathcal{D}}\Big(h_S(x) \neq f(x), x \notin \{x_1, \ldots, x_m\}\Big) \\ &\quad \text{since } \mathcal{D} \text{ is continuous} \\ &= \mathbb{P}_{x\sim\mathcal{D}}\Big(f(x) \neq 0, x \notin \{x_1, \ldots, x_m\}\Big) \\ &= \mathbb{P}_{x\sim\mathcal{D}}\Big(f(x) \neq 0\Big), \text{ using continuity again} \\ &= \mathbb{P}_{x\sim\mathcal{D}}\Big(f(x) = 1\Big) = \frac{1}{2} \ldots ! \end{aligned}$$

"Danger" of overfitting and a possible solution

- In the previous example, the performance of the predictor h_S is
  - **perfect** on the sample (error $= 0$!)
  - **very poor** on the true world (50% chance of being wrong).

  $\rightarrow$ **overfitting/difficulty to generalize**
- The predictor h_S might look "unnatural". $\exists$ a function p_S such that
  - $p_S(x) \leq 1, \quad \forall\, x \in \mathcal{X}$
  - p_S is a polynomial function when restricted on the subsets $\{x \in \mathcal{X} : f(x) = k\}, k \in \{0, 1\}$
  - $h_S = \mathbb{1}_{p_S=1}$ on $\mathcal{X}$

"Danger" of overfitting and a possible solution

• A **possible remedy** : search for an ERM predictor over a **restricted** class $\mathcal{H}$:

$$\mathrm{ERM}_{\mathcal{H}}(S) \in \mathrm{argmin}_{h \in \mathcal{H}} L_S(h)$$

where $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$. $\rightarrow$ **inductive bias**

• Such a class (hypothesis class) $\mathcal{H}$ should be chosen **before** seeing any sample.

• Question : how should such a class be **chosen** ?

• Next : the case of a **finite** class.

Finite classes and a learning guarantee

Definition (The realizability assumption)
There exists $h^* \in \mathcal{H}$ such that

$$L_{(\mathcal{D},f)}(h^*) = 0 \tag{1}$$

Remark. This assumption implies that $L_S(h^*) = 0$ with probability 1. Indeed, (1) means that the event $\{x : h^*(x) = f(x)\}$ occurs with **with probability 1**, and hence

$$L_S(h^*) = \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}_{\{h^*(x_i) \neq f(x_i)\}} = 0 \quad \text{with probability 1.}$$

Consequence. Any ERM hypothesis, h_S, satisfies $L_S(h_S) = 0$ with probability 1

$$L_S(h_S) \leq L_S(h^*).$$

Finite classes and a learning guarantee

- We assume that $x_1, \ldots, x_m$ are i.i.d $\sim \mathcal{D}$.

Theorem.

$$m \geq \frac{\log\left(|\mathcal{H}|/\delta\right)}{\epsilon}.$$

$$\mathbb{P}_{(x_1,..x_m)\sim\mathcal{D}^m}\left[L_{(\mathcal{D},f)}(h_S) \leq \epsilon\right] \geq 1-\delta$$

for **any** ERM hypothesis, $h_S \in \text{argmin}_{h\in\mathcal{H}} L_S(h)$

Finite classes and a learning guarantee - Proof

- Let $S|_x = \{x_1, \ldots, x_m\}$.

$$\mathcal{D}^m\Big(\{S|_x : L_{(\mathcal{D},f)}(h_S) > \epsilon\}\Big) = \mathbb{P}_{(x_1,\ldots,x_m)\sim\mathcal{D}^m}\Big(\{L_{(\mathcal{D},f)}(h_S) > \epsilon\}\Big).$$

Proof. Let $\mathcal{H}_B$ be the set of **"bad"** hypotheses :

$$\mathcal{H}_B = \Big\{h \in \mathcal{H} : L_{(\mathcal{D},f)}(h) > \epsilon\Big\}.$$

Also, consider the set of **"misleading"** samples

$$M = \Big\{S|_x : \exists\ h \in \mathcal{H}_B, L_S(h) = 0\Big\}.$$

Since the realizability assumption holds, we know that $L_S(h_S) = 0$ with probability 1. Hence,

Finite classes and a learning guarantee - Proof

More formally, we have that $\{S|_x : L_{(\mathcal{D},f)}(h_S) > \epsilon\} \subseteq M$.

Since $M = \cup_{h \in \mathcal{H}_B} \{S|_x : L_S(h) = 0\}$, we have by the union bound

$$\begin{aligned}
\mathcal{D}^m\Big(\{S|_x : L_{(\mathcal{D},f)}(h_S) > \epsilon\}\Big) &\leq \sum_{h \in \mathcal{H}_B} \mathcal{D}^m\Big(\{S|_x : L_S(h) = 0\}\Big) \\
&= \sum_{h \in \mathcal{H}_B} \mathcal{D}^m\Big(\{S|_x : \forall i \; h(x_i) = f(x_i)\}\Big) \\
&= \sum_{h \in \mathcal{H}_B} \prod_{i=1}^{m} \mathcal{D}\Big(\{x_i : h(x_i) = f(x_i)\}\Big)
\end{aligned}$$

under the i.i.d. assumption. Also, we have that

$$\begin{aligned}
\mathcal{D}\Big(\{x_i : h(x_i) = f(x_i)\}\Big) &= \mathcal{D}\Big(\{x : h(x) = f(x)\}\Big) \\
&= 1 - L_{(\mathcal{D},f)}(h) \leq 1 - \epsilon.
\end{aligned}$$

Finite classes and a learning guarantee - Proof

It follows that

$$\begin{aligned} \mathcal{D}^m\Big(\{S|_x : L_{(\mathcal{D},f)}(h_S) > \epsilon\}\Big) &\leq |\mathcal{H}_B|(1-\epsilon)^m \\ &\leq |\mathcal{H}_B| e^{-\epsilon m} \end{aligned}$$

Now :

$$|\mathcal{H}| e^{-\epsilon m} \leq \delta$$

is equivalent to

$$m \geq \frac{\mathbf{1}}{\epsilon} \log\left(\frac{|\mathcal{H}|}{\delta}\right)$$

as stated. □

Before we close this section

- Consider again the classification problem on the square with

$$h_S(x) = \begin{cases} y_i & \exists\, x_i : x = x_i \ \text{ for some } i \in [m] \\ 0 & \text{otherwise.} \end{cases}$$

- Consider the class $\mathcal{P}$ of all **piecewise polynomials** p such that $p(x) \leq 1,\ \forall\, x \in \mathcal{X}$.

$$\begin{aligned} g(x) &= 0, && \text{if } p(x) < 1 \\ &= p(x) = 1, && \text{if } p(x) = 1 \end{aligned}$$

- Then, we can easily show that $h_S = g_S$ with

$$p_S(x) = f(x)\left(1 - \prod_{i=1}^{m}(x - x_i)^2\right)$$

is the associated piecewise polynomial.

Summary

- In any statistical learning problem, we have
  - A task to be learned
  - Training data (representing the true world)
  - An algorithm to compute our prediction rule/hypothesis
  - A measure for the error or risk of being wrong (to assess the quality of the prediction)
- Prediction method : minimize the empirical risk L_S (ERM rule)
- Danger of overfitting and how to solve it
  - We may overfit when the prediction rule does "too well" on the training data but have difficulty to generalize to test data
  - To solve this issue, we resort to restricting ourselves to some class $\mathcal{H}$
  - When $\mathcal{H}$ is finite and the realizability assumption holds, we have a learning guarantee with $m \geq \log(|\mathcal{H}|/\delta)/\epsilon$.

Lecture 1

Definition of PAC learnability

• We have seen that for a **finite** hypothesis class, an ERM rule will be **probably approximatively correct (PAC)** for large sample sizes.

Definition.

- for all $(\epsilon, \delta) \in (0,1)^2$
- for any distribution $\mathcal{D}$ on $\mathcal{X}$
- for any labeling function $f : \mathcal{X} \to \mathcal{Y} \equiv \{0,1\}$

if the realizability assumption holds with respect to $\mathcal{H}, \mathcal{D}, f$, i.i.d. training examples
$\sim (\mathcal{D}, f)$,

$$\mathbb{P}_{x\sim\mathcal{D}}(L_{(\mathcal{D},f)}(h(x)) \leq \epsilon) \geq 1 - \delta.$$

Definition of PAC learnability and sample complexity

- **Approximatively** correct : refers to $L_{(\mathcal{D},f)}(h) \leq \epsilon$ **only**.
- **Probably** : refers to the confidence parameter $\mathbf{1} - \delta$.
- Let us not lie to ourselves : we cannot avoid such approximations. We are **forgiven** for making mistakes since we only have a **finite random sample** at our disposal.
- **Sample complexity** : is the **minimal sample size** needed to achieve the approximation ϵ and confidence $1 - \delta$. It depends on $\mathcal{H}$!
- Every **finite hypothesis class** $\mathcal{H}$ is, under the realizability assumption, **PAC** learnable with sample complexity

$$m_{\mathcal{H}}(\epsilon, \delta) \leq \left\lfloor \frac{\log(|\mathcal{H}|/\delta)}{\epsilon} \right\rfloor + 1$$

with $\lfloor z \rfloor$ denotes the integer part of $z \in \mathbb{R}$ $\Leftarrow$ **Consequence of the Theorem**

Towards a more general model

- Question : Are there **infinite** hypothesis classes that are PAC learnable ?
- The answer is positive, and is related to the **VC dimension** of the class.
- In order to **generalize** the scope, we can consider
  1. **waiving** the realizability assumption : the latter can be **too strong**,
  2. going **beyond binary classification** : imagine we want to predict a real-valued outcome.
- 1 $\rightarrow$ **Agnostic** PAC learning
- 2 $\rightarrow$ **Multi-class classification, regression,...**

Towards a more general model : (1) Agnostic PAC learning

- Recall that in the realizability assumption we stipulated that $\exists\ h^* \in \mathcal{H}$ such that

$$\mathbb{P}_{x\sim\mathcal{D}}[h^*(x) = f(x)] = 1$$

meaning that with probability 1, h^* is a perfect classifier.

- **BUT**, two papayas might have the same color and softness and still have **different tastes**
- From now on, let $\mathcal{D}$ be a **joint** probability distribution on the product set $\mathcal{X} \times \mathcal{Y}$, where as before
  - $\mathcal{X}$ is the domain set
  - $\mathcal{Y}$ is the set of labels (so far $\{0, 1\}$)
- For a given $h : \mathcal{X} \mapsto \mathcal{Y} \in \mathcal{H}$, the **true error/risk** associated with this h is

$$L_{\mathcal{D}}(h) = \mathbb{P}_{(x,y)\sim\mathcal{D}}[h(x) \neq y] \stackrel{\text{def}}{=} \mathcal{D}\Big(\{(x, y) : h(x) \neq y\}\Big).$$

Towards a more general model : (1) Agnostic PAC learning

- For the same h, the **empirical error/risk** based on the training data S is given by $L_S(h) \stackrel{\text{def}}{=} \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}_{\{h(x_i)\neq y_i\}}$ with $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$.

- **Goal** : As before, we want to find some $h : \mathcal{X} \to \mathcal{Y}$ from $\mathcal{H}$ such that **probably approximatively** h is **correct** (has a small true risk).

- Question : For which h (in the class of all possible classifiers) we have the smallest true risk ?

- Answer : It can be shown that (exercise 3.7) that it is given by the **Bayes classifier** :

$$\begin{aligned} h_{\mathcal{D}}(x) &= 1, \text{ if } \mathbb{P}[y = 1|x] \geq 1/2 \\ &= 0, \text{ otherwise,} \end{aligned}$$

that is, for any other classifier $h : \mathcal{X} \to \{0, 1\}$ we have $L_{\mathcal{D}}(h_{\mathcal{D}}) \leq L_{\mathcal{D}}(h)$.

The Bayes classifier

- Recall that the true risk is given by

$$L_D(h) = \mathbb{P}_{(x,y)\sim\mathcal{D}}\left(h(x) \neq y\right).$$

- The Bayes classifier "decides" to give the label 1 if and only if

$$\mathbb{P}[y = 1|x] \geq \mathbb{P}[y = 0|x].$$

- Consider the function $\eta(x) = \mathbb{P}[y = 1|x],\ x \in \mathcal{X}$.
- We can show the following
  - $L_{\mathcal{D}}(h_{\mathcal{D}}) = \mathbb{E}_x \min\Big(\eta(x), 1 - \eta(x)\Big) \leq 1/2,$
  - for any other prediction rule h

$$L_{\mathcal{D}}(h) - L_{\mathcal{D}}(h_{\mathcal{D}}) = \mathbb{E}_x\Big[\Big|2\eta(x) - 1\Big|\mathbb{1}_{h(x)\neq h_D(x)}\Big].$$

The Bayes classifier : an example

- Consider the random pair $(x, y) \in \mathcal{X} \times \mathcal{Y} \equiv \mathbb{R} \times \{0, 1\}$ such that

$$\begin{aligned} x|y=0 &\sim \mathcal{N}(\mu_0, 1), \text{ and} \\ x|y=1 &\sim \mathcal{N}(\mu_1, 1) \end{aligned}$$

for some $\mu_0 \neq \mu_1 \in \mathbb{R}$.

- Also, let $\pi = \mathbb{P}(y = 1), 1 - \pi = \mathbb{P}(y = 0)$.
- If ϕ denotes the density of $\mathcal{N}(0, 1)$, then x has (unconditional) density

$$g(x) = (1 - \pi)\phi(x - \mu_0) + \pi\phi(x - \mu_1)$$

- By the Bayes Theorem :

$$\mathbb{P}[y = 1|x] = \frac{\pi\phi(x - \mu_1)}{g(x)} = \frac{\pi\phi(x - \mu_1)}{(1 - \pi)\phi(x - \mu_0) + \pi\phi(x - \mu_1)}.$$

The Bayes classifier : an example

- Also,

$$\mathbb{P}[y=0|x] = \frac{(1-\pi)\phi(x-\mu_0)}{(1-\pi)\phi(x-\mu_0)+\pi\phi(x-\mu_1)}.$$

- In this example, $\mathbb{P}[y=1|x] \geq \mathbb{P}[y=0|x]$ if and only if

$$\frac{\pi\phi(x-\mu_1)}{(1-\pi)\phi(x-\mu_0)} \geq 1$$

- In the special case $\pi = 1-\pi = 1/2$, the Bayes classifier is given by

$$\begin{aligned} h_{\mathcal{D}}(x) &= 1, \quad \text{if } |x-\mu_1| \leq |x-\mu_0| \\ &= 0, \quad \text{otherwise.} \end{aligned}$$

Back to Agnostic PAC learning : A formal definition

- **But**, we cannot use $h_{\mathcal{D}}$ since it $\mathcal{D}$ is **unknown**.
- All we know that any algorithm will produce a prediction rule that is **no better** than $h_{\mathcal{D}}$.
- **Definition.** A hypothesis class $\mathcal{H}$ is **Agnostic PAC learnable** if there exist a function $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm such that
  - for any $(\epsilon, \delta) \in (0,1)^2$
  - and a probability distribution $\mathcal{D}$ on $\mathcal{X} \times \mathcal{Y}$,

  if the algorithm runs on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d examples generated by $\mathcal{D}$,

$$\mathbb{P}_{S \sim \mathcal{D}^m}\left[L_{\mathcal{D}}(\tilde{h}_S) \leq \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + \epsilon\right] \geq 1 - \delta.$$

Back to Agnostic PAC learning : Remarks

- In this setting **no learner** can guarantee an arbitrarily small error.
- The learning is still **successful** if the true risk is **not much larger** than the smallest error on the class.
- Schematically :