MACHINE LEARNING

giovanni.neglia@inria.fr
othmane.marfoq@inria.fr

EVALUATION

(1) EXAM (40%)
(2) HOMEWORK (30%)
(3) QUESTION AT EVERY LECTURE (30%)

2020/2021

16/09

INTRODUCTION

MACHINE LEARNING

- way of extract rules from data

DATA → [ML] → RULES/PROGRAM

ISSUES

- To large set of rules (REM Pidgeon)
- To small set of rules (REM Rat)

& TOO MUCH reduces learning flexibility

INDUCTIVE PRIOR

- A PRIORI KNOWLEDGE to prevent useless conclusion

INDUCTIVE INFERENCE

- Ability to proceed from general examples to a broadere generalization

KINDS OF LEARNING

SUPERVISED / UNSUPERVISED

- In supervised learning output label are given; Unsupervised instead finds unexpected correlation. (SPAM MAIL vs ANOMALITY DETECTION)

REINFORCHENT

- Labels are given but after a while

COOPERATIVE / INDIFFERENT / ADVERSARIAL TEACHER

- Change the way the data is given from a teacher

BATCH / ONLINE

- Data can be given all together or PASSO PASSO

! ⇒

- we'll se SUPERVISE STATISTICAL BATCH LEARNING WITH PASSIVE TEACHER

STATISTIC & ML

1. Statistic wants to check an HYPOTHESIS (■ smoke affects heart?)
   ML wants to find the HYPOTHESIS (■ what affects heart?)
2. Statistic starts from the Prob. Distr. of data.
   ↳ ASYNTOTICS: if you look at enough data you find a GAUSSIAN
   ML doesn't know the distribution
   ↳ FINITE SAMPLES: you get the distribution you have

STATISTICAL LEARNING FRAMEWORK

- X input space
- Y output space

} (vector of features)

- S dataset of size m (# SAMPLES)

---

f and D are unknown to the learner

- f correct labeling function ; D distribution over X
- $A : S \subset (X \times Y)^m \longrightarrow \{\text{functions } X \to Y\}$
  $\Rightarrow A(S) = h$

h is the output of an ML Algo. (is a PREDICTION RULE)

- $h : X \to Y$ called HYPOTESIS or PREDICTOR or CLASSIFIER it can be seen as $A(S)$, so the output of the ML ALGO. A given the DS S

INDICATOR FUNCTION: 0 if prediction is ok it is 0 else is 1

EXPECTED LOSS

- $L_D(h) = \mathbb{E}_D\left(\mathbb{1}_{h(x) \neq y}\right) \in [0,1]$

EMPIRICAL LOSS

- $L_S(h) = \dfrac{\sum_{i=1}^{m} \mathbb{1}_{h(x_i) \neq y_i}}{m}$

$\Rightarrow$ for large number theory, EMPIRICAL LOSS will converge to EXPECTED LOSS.

---

note

- we can write $y = f(x)$ so we can rewrite $L_{D,f}(h) = \mathbb{E}_D(\mathbb{1}_{h(x) \neq f(x)})$; note also that our hope is to find h such that $L_{D,f}(h) < \varepsilon$ with Prob $> 1 - \delta$ because in the best case we can have $L_{D,f}(h) = 0$ (that is obv. impossible)

APPROXIMATE WITH HIGH PROBABILITY

?

- from a set of predictor which one should we choose?
  cause we do not have the D variable we cannot calculate the EXPECTED LOSS, so we use the EMPIRICAL LOSS, and take the one that minimize it. (there can be more than one)

ERM

- EMPIRICAL RISK MINIMIZATION

$$h_{ERM} \in \operatorname{argmin} L_S(h)$$

?

ERM is bad?

MEMORIZATION ALGORITHM

- MEMORIZATION → you learn all the couple $\{x, y\}$, so when in future you see a sample that is in the dataset, you know it's label, else you TOSS A COIN (or a DICE, or ...)

$$h_M(x) = \begin{cases} y_i \text{ if } x = x_i \in S \\ \text{TOSS A COIN} \end{cases}$$

- we can proof that $\exists D, f \mid L_{D,f}(h_M) = 1/2$ w.p. 1

↳ Assume uniform Prob. dist. D over X

$f(x)=1$	$f(x)=0$

// 2 feature space

↳ The predictor is wrong 1/2 of the time, because you have infinite new possible sample, so the prob. of getting a sample already in the dataset is $\emptyset$ (NOT IMPOSSIBLE BUT PROB = ϕ), so the Prob. of taking a sample not in original dataset is 1, so the predictor always toss a coin! it explain perfectly the dataset but can't explain anything else.

⇒

- ERM is not bad but learns in a too large set of HYPOTESIS, so it depends from the set of HYPOTESIS you apply it.

?

ERM_H

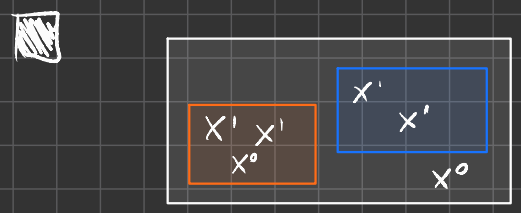
When is good?

↳ we should search the minimum in a finite set of HYPOTESIS, so

$$h_{ERM_H} \in \operatorname{argmin}_{h \in H} L_S(h)$$

REALIZABILITY ASSUMPTION

- Realizability assumption impose $f \in H$, so if H is finite ERM_H can learn



// 2 feature space

. BLUE CH is better because it has positive sample inside and a band outside

- A predictor is bad if:

$$H_B = \{ h \in H \mid L_{D,f}(h) > \varepsilon \}$$

- A Dataset is MISLEADING if

$$S_M = \{ S, h_{ERM_H} \in H_B \} \subset S_P$$

↑ POTENTIALLY MISLEADING DATASET

- $S_P = \{ S \mid \exists h \in H_B \text{ with } L_{D,f}(h) = 0 \}$

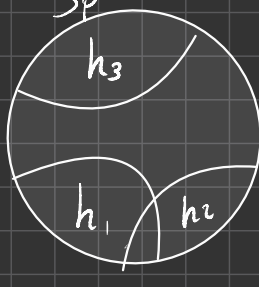
. There is a finite number of $h \in H_B \mid L_S(h) = 0$

PROBABILITIES TO PICK DATASET

- $P_{\mathbb{R}^{D^m}}(S_M) \overset{\text{perché } S_M \subset S_P}{\leq} P_{\mathbb{R}^{D^m}}(S_P) =$

$$= P_{\mathbb{R}^{D^m}}\left(\bigcup_{h \in H_B} \{ S \mid L_S(h) = 0 \} \right) \leq$$

$$\leq \sum_{h \in H_B} P_{\mathbb{R}^{D^m}}(\{ S \mid L_S(h) = 0 \})$$



TRIVIAL

- $L_S(h) = 0$ if $\forall (x_i, y_i) \in S \; h(x_i) = y_i$
- $\text{Prob}(h(x_i) = y_i) \leq 1 - \varepsilon$ so
- $\text{Prob}(h(x_i) \neq y_i \;\; \forall i = 1, \dots, m) \leq (1-\varepsilon)^m$

$$\leq \sum_{h \in H_B} (1-\varepsilon)^m = |H_B| (1-\varepsilon)^m \leq |H| (1-\varepsilon)^m$$

S^m so A BAD ONE

we have proved

- $\text{Prob}(\text{ERM}_H \text{ picks a predictor with } L_{D,f}(h) > \varepsilon) \leq$
  $\leq |H| (1-\varepsilon)^m$
  - ↳ we want this quantity $|H|(1-\varepsilon)^m \leq \delta$

Can we choos how small should be?

↳ sost picking m large enough: $m \geq \dfrac{\ln |H|/\delta}{\varepsilon}$

Conclusion

- given finite H, ERM_H can learn $\forall \varepsilon, \delta, D, f$
  if $m \geq \dfrac{\ln |H|/\delta}{\varepsilon}$

PAC LEARNABILITY

Def

- A class H is probably approximately correct if $\exists$ an algorithm $A : (X \times Y)^m \to H$ (AND $\exists$ also a funct $m_H : (0,1)^2 \to \mathbb{N}$) such that $\forall D$ over X and $\forall f$ over Y, $\forall \varepsilon \in (0,1)$, $\forall \delta \in (0,1)$ whenever we pick a dataset of size $|S| = m \geq m_H(\varepsilon, \delta)$ then we can guarantee that our algo. is such that

$$L_{D,f}(A(S)) \leq \varepsilon \text{ with Pr} \geq (1-\delta)$$

- A finite class H is PAC learnable with ERM_H & δ

$$m_H(\varepsilon, \delta) = \frac{\ln |H| / \delta}{\varepsilon}$$

m_H is called SAMPLE-COMPLEXITY of the class H