Aalto University
School of Science

CS-E4070 — Computational learning theory

Slide set 07 : learning in the presence of noise

Cigdem Aslay and Aris Gionis

Aalto University

spring 2019

reading material

- K&V, chapter 5

PAC learning

- PAC **learning**: a concept class $\mathcal{C}$ if PAC learnable, if there exists an algorithm A, so that for every concept $c \in \mathcal{C}$, every distribution $\mathcal{D}$, and every $\epsilon > 0$ and $\delta \in (0, 1)$, the algorithm A outputs a hypothesis $h \in \mathcal{C}$ that satisfies

$$error_{\mathcal{D}}(h) \leq \epsilon$$

  with probability at least $1 - \delta$.

- another limitation of the model: so far we have assumed that the example generator $EX(c, \mathcal{D})$ is noise free

- in this lecture: we will see how to extend the PAC learning framework to deal with noise

a possible extension to introduce noise

- introduce example generator $EX^{\eta}(c, \mathcal{D})$
- an extension of $EX(c, \mathcal{D})$
- η is a noise parameter
- each call to $EX^{\eta}(c, \mathcal{D})$ returns a sample $(\mathbf{x}, y)$ such that
  - $\mathbf{x}$ is sampled from $\mathcal{D}$
  - with probability $1 - \eta$ we set $y = c(\mathbf{x})$
  - with probability η we set $y = \neg c(\mathbf{x})$ (negation)
- we assume $0 \leq \eta < \frac{1}{2}$
- $\eta = \frac{1}{2}$ gives totally random samples
  - no hope in learning anything

our aim

- as before, we want to ensure that for any concept c, any distribution $\mathcal{D}$, any ϵ and δ, the learner A returns a hypothesis h having

$$error_{\mathcal{D}}(h) = \mathbf{Pr}_{\mathbf{x} \sim \mathcal{D}}[h(\mathbf{x}) \neq c(\mathbf{x})] \leq \epsilon$$

  with probability at least $1 - \delta$

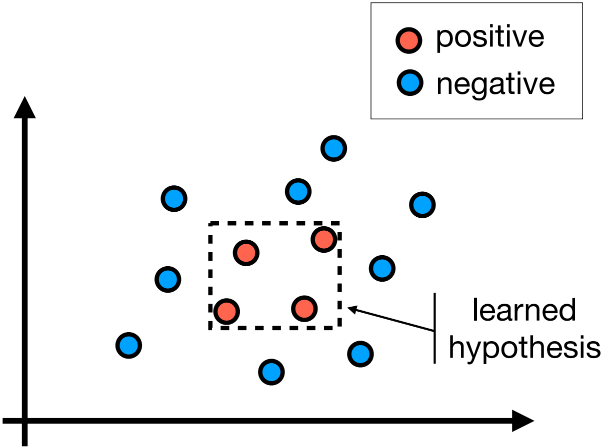
- but now the learner gets samples drawn from $EX^{\eta}(c, \mathcal{D})$

- in addition, we assume that the learner has some knowledge about the amount of noise in the data
  - we assume an upper bound η_0, i.e., $0 \leq \eta \leq \eta_0 < \frac{1}{2}$
  - the learner knows η_0
  - we will allow time polynomial in $\frac{1}{1-2\eta_0}$

is it really a more challenging setting?

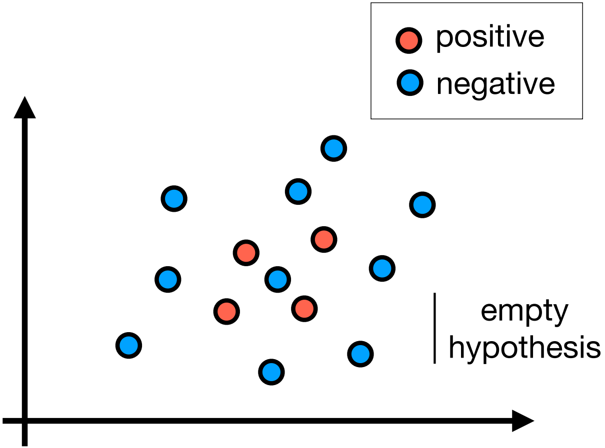
- consider some of the previous PAC learning algorithms in this new noise-tolerant model
  - learning axis-aligned rectangles
  - learning boolean conjunctions
- how do they perform?

learning axis-aligned rectangles



K&V, section 1.1

a single noisy sample can break the algorithm



K&V, section 1.1

learning boolean conjunctions

K&V, section 1.3

learning algorithm

- initial hypothesis

$$h(x_1, \ldots, x_n) = x_1 \wedge \overline{x}_1 \wedge x_2 \wedge \overline{x}_2 \wedge \ldots \wedge x_n \wedge \overline{x}_n$$

  (initially not satisfiable)

- negative examples drawn from $EX(\mathcal{D}, c)$ are ignored
- for positive examples
  - if $a_i = 0$ we delete literal x_i from h
  - if $a_i = 1$ we delete literal $\overline{x}_i$ from h

learning boolean conjunctions

the previous algorithm in the noise-tolerant model:

- consider a boolean conjunction with a literal z
- assume prob. γ to draw a sample that does not satisfy z
- such a sample should be negative
- with probability $\gamma\eta$ the sample becomes positive
  - due to $EX^{\eta}(c, \mathcal{D})$
- leading to eliminating z
- in the same manner, we may eliminate all literals from the target conjunction

learning boolean conjunctions: a different algorithm

still in the original noise-free setting

- consider literal z over boolean variables $x_1, \ldots, x_n$
- $p_0(z)$: prob. z is set to 0 in a sample
  if $p_0(z)$ is *"small"* we can *"ignore"* z – it is always set to 1
- $p_{01}(z)$: prob. z is set to 0 in a positive sample
  notice that if $z \in c$ then $p_{01}(z) = 0$
  if $p_{01}(z)$ is *"large"* we should avoid including z in h
- we say that z is significant if $p_0(z) \geq \epsilon/8n$
- we say that z is harmful if $p_{01}(z) \geq \epsilon/8n$
- since $p_{01}(z) \leq p_0(z)$ any harmful literal is also significant
- we want to include in our hypothesis literals that are significant but not harmful

learning boolean conjunctions: a different algorithm

- we can show:
- **theorem** : if a hypothesis h contains all literals that are significant but not harmful, then $error(h) \leq \epsilon$ with probability at least $1 - \delta$
- this gives a different PAC learning algorithm
  - estimate $p_0(z)$ and $p_{01}(z)$ for all literals z
  - include literals in h based on these estimates
- how can we estimate $p_0(z)$ and $p_{01}(z)$?
  - by sampling from $EX(c, \mathcal{D})$
  - in practice, we get approximations $\hat{p}_0(z)$ and $\hat{p}_{01}(z)$
  - we can control the error by Chernoff bounds

learning boolean conjunctions: a different algorithm

- PAC learnability of new algorithm is shown for the original noise-free setting
- however, intuitively the new algorithm seems more robust
- it seems that can be used for learning in the noise setting
- what is the difference of the two algorithms?
  - previous algorithm examines examples one-by-one and makes a decision upon seen each example
  - a noisy example may force it to make a bad decision from which it cannot recover
  - new algorithm gathers information about statistical properties of the data and makes decision based on those properties
  - the latter idea can be generalized

statistical query learning model

- we replace the oracle $EX(c, \mathcal{D})$ by oracle $STAT(c, \mathcal{D})$
- oracle $STAT(c, \mathcal{D})$ takes input a pair (χ, τ)

  where $\chi : X \times \{0, 1\} \to \{0, 1\}$ indicates the presence of some property in an example $(\mathbf{x}, y)$, and $0 \le \tau \le 1$
- oracle $STAT(c, \mathcal{D})$ outputs an estimate of

$$P_\chi = \mathbf{Pr}_{\mathbf{x} \sim \mathcal{D}}[\chi(\mathbf{x}, c(\mathbf{x})) = 1]$$

- in particular, oracle $STAT(c, \mathcal{D})$ returns a value $\hat{P}_\chi$ s.t.

$$P_\chi - \tau \le \hat{P}_\chi \le P_\chi + \tau$$

- the parameter τ is called tolerance
- example: in the previous algorithm $p_{01}(z) = P_{\chi_z}$

statistical query learning model

- oracle $STAT(c, \mathcal{D})$ can be computed by calls to $EX(c, \mathcal{D})$
- how?
  - draw examples $(\mathbf{x}, c(\mathbf{x}))$ and compute the fraction of which $\chi(\mathbf{x}, c(\mathbf{x})) = 1$ as the estimate $\hat{P}_\chi$ of P_χ
- using Chernoff bounds we can show that, with probability at least $1 - \delta$, the estimate $\hat{P}_\chi$ approximates P_χ within tolerance τ, if the number of calls to $EX(c, \mathcal{D})$ is polynomial in $1/\tau$ and $\ln(1/\delta)$

statistical query learning model

- **definition** : we say that a concept class $\mathcal{C}$ is learnable from statistical queries using a hypothesis class $\mathcal{H}$, if there is an algorithm A with access to queries $STAT(c, \mathcal{D})$, so that for any $c \in \mathcal{C}$, any distribution $\mathcal{D}$, and any $0 \leq \epsilon < \frac{1}{2}$, the algorithm A returns a hypothesis $h \in \mathcal{H}$ that satisfies $error(h) \leq \epsilon$

- we say that such an algorithm is efficient if its running time is polynomial in $1/\tau$, $1/\epsilon$, and n.

- why there is no confidence δ in this definition?

statistical query learning model

- **theorem** : if a concept class $\mathcal{C}$ is efficiently learnable from statistical queries, then $\mathcal{C}$ is efficiently PAC learnable

  still in the noise-free setting

learning in the presence of noise

- we want to achieve PAC learning in the presence of noise
- we can leverage the previous result if we can compute

$$P_\chi = \mathbf{Pr}_{\mathbf{x}\sim\mathcal{D}}[\chi(\mathbf{x}, c(\mathbf{x})) = 1]$$

  by access to queries $EX^\eta(c, \mathcal{D})$

- this is shown in K&V, section 5.4.1

$$P_\chi = p_1 \frac{\mathbf{Pr}_{EX^\eta}[\chi = 1] - \eta}{1 - 2\eta} + \mathbf{Pr}_{EX^\eta}[(\chi = 1) \wedge (\mathbf{x} \in X_2)]$$

details omitted

putting everything together

- **theorem** : if a concept class $\mathcal{C}$ is efficiently learnable from statistical queries, then $\mathcal{C}$ is efficiently PAC learnable in the presence of noise

- **corollary** : the class of boolean conjunctions is efficiently PAC learnable in the presence of noise