1 Lecture 6: 2017.02.27

1.1 Learning with Noise

Now we turn our attention to learning with noise. The set up is the same as in the PAC-learning with the difference that the "subroutine" that extracts examples, $EX_\eta(c, D)$, is now a Bernoulli trial. With probability $1-\eta$ it returns the correct label $\langle x, c(x)\rangle$ and with probability η it returns the incorrect label $\langle x, \neg c(x)\rangle$. We also ask for the algorithm to be polynomial on $\frac{1}{1-2\eta}$.

- We assume that the noise is *independent* of x and $c(x)$. This is akin to measurement error. Note that our noise affects only the labels and not on the x's.
- We also assume that $\eta < \frac{1}{2}$. Otherwise it would be impossible to distinguish between the true label and the false label.

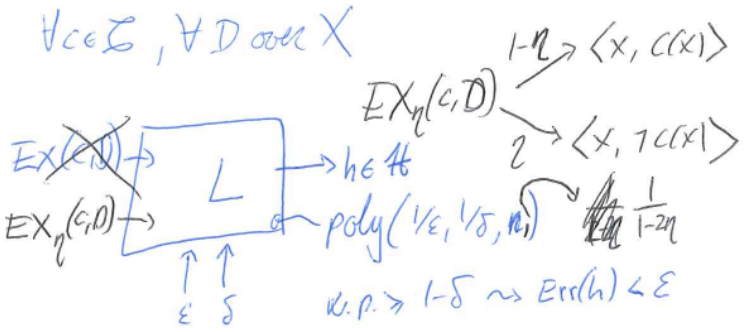

Figure 1: PAC-learning with noise.

Given any $h \in \mathcal{H}$, we can compute the noisy error

$$\mathbb{P}_{\langle x,y\rangle \sim EX_\eta}[h(x) \neq y] = (1-\eta)Err(h) + \eta(1 - Err(h)) = \eta + (1-2\eta)Err(h), \tag{1.1}$$

observe that since we have $\eta < \frac{1}{2}$, this is increasing as a function $Err(h)$. This means that noise preserves the ranking of hypothesis according to their error. So we can still minimize this quantity and obtain a good model.

Remark 1.1. *As a consequence, all the VCD results still hold with the only caveat that we need more data, in particular, we need*

$$m \sim \frac{VCD(\mathcal{C})}{\varepsilon^2(1-2\eta)^2} \log\left(\frac{1}{\delta}\right). \tag{1.2}$$

1.1.1 "Malicious" Errors

Same setup as before but with a slight change.

- with probability $1-\eta$ we return the correct pair $\langle x, c(x)\rangle$,

- with probability η, an "adversary" generates *any* pair $\langle x, y \rangle$.

The general theory still applies and by minimizing observed error we can still pick up a target with error at most η. However, the "malicious" problem is considerably worse and we can no longer ask for $Err << \eta$. To see this consider a fixed distribution D over X and two concepts $c_1, c_2 \in \mathcal{C}$ such that $c_1 \triangle c_2$ has ε weight under D.

Figure 2: Malicious noise.

Then, as long as $\eta \geq \frac{\varepsilon}{1+\varepsilon}$, the adversary can make you confuse c_1 and c_2.

1.2 Learning (monotone) Conjunctions with "Statistics"

Recall the problem of learning conjunctions.

- $X = \{0,1\}^n$, concept class. E.g. $c = x_1 \wedge x_5 \wedge x_6 \wedge x_{19}$.

We started with the hypothesis $h = x_1 \wedge \ldots \wedge x_n$ and we deleted any variable that contradicted the positive examples (all the variables that are zero in some positive example). However, if we introduce noise this algorithm falls apart since we could have the all 0's vector to be incorrectly labeled positive which results in the empty conjunction. The underlying problem is that we made drastic decisions on our hypothesis based on a single examples. To overcome this shortcoming, we will only introduce changes if we see enough *statistical* evidence.

- For each x_i, define

$$p_0(x_i) = \mathbb{P}_{\vec{x} \sim D}[x_i = 0], \tag{1.3}$$
$$p_{01}(x_i) = \mathbb{P}_{\vec{x} \sim D}[x_i = 0 \ \& \ c(\vec{x}) = 1]. \tag{1.4}$$

We call x_i *significant* if $p_0(x_i) \geq \frac{\varepsilon}{4n}$, and *harmful* if $p_{01}(x_i) \geq \frac{\varepsilon}{4n}$.

- Algorithm: Let h be the conjunction of all x_i's that are significant and not harmful.

Let's look at the analysis of the errors. For the FP type

$$\mathbb{P}_{\vec{x}\sim D}[c(\vec{x}) = 0 \ \& \ h(\vec{x}) = 1], \tag{1.5}$$

must be some x_i such that $x_i \in c$, $c_i \notin h$ and $x_i = 0$. This means that x_i is not harmful which implies that x_i is not significant and $p_0(x_i) \leq \frac{\varepsilon}{4n}$. Thus, the total error is $p_0(n) \leq n\frac{\varepsilon}{4n} \leq \frac{\varepsilon}{4}$. Similarly, for the FN type

$$\mathbb{P}_{\vec{x}\sim D}[c(\vec{x}) = 1 \ \& \ h(\vec{x}) = 0], \tag{1.6}$$

must be some x_i such that $x_i \notin c$, $x_i \in h$ and $x_i = 0$. Therefore, $p_{01}(n) \leq \frac{\varepsilon}{4n}n \leq \frac{\varepsilon}{4}$.

Remark 1.2.

Remark 1.3 (Concentration Inequalities)**.** *Consider a biased coin with $\mathbb{P}[head] = p$ and $\mathbb{P}[tails] = 1 - p$. We flip the coin m times and let $\hat{p} = \frac{\#\ heads\ observed}{m}$. We are interested in bounds like (Hoeffding's)*

$$\mathbb{P}_{m\ flips}\left[|p - \hat{p}| \geq \varepsilon\right] \leq e^{-\frac{\varepsilon^2 m}{3}}, \tag{1.7}$$

and the closely related Chernoff bound. This is like the law of large numbers but with an explicit rate.

We can estimate the $p_0(x_i)$ probabilities with $EX_\eta(c, D)$ since we don't care about the noise of the labels.

The interesting part is that we can do the same for the p_{01}'s.

1.3 Statistical Query (SQ) Learning

Remark 1.4 (Warning)**.** *The professor sort of rushed through the definition of SQ-learning. Check the textbook for more details.*

We modify the PAC learning algorithm. We replace the subroutine $EX(c, D)$ with $SQ(c, D)$ which interacts with the algorithm L in the role of "oracle", answering the questions (queries) posed by L. For example, what is the probability of $x_i = 0$?.

These queries, χ, are *predicates* (0/1 valued functions) on $\langle x, y\rangle$ pairs

$$\chi(x, y) \in \{0, 1\}^n \longrightarrow P_\chi \coloneqq \mathbb{P}_{\langle x,y\rangle\sim EX(c,D)}[\chi(x, y) = 1]. \tag{1.8}$$

E.g. $\chi(\vec{x}, y) = 1 \iff x_{11} = 0$ $[p_0(x_{11})]$
E.g. $\chi(\vec{x}, y) = 1 \iff x_{11} = 0 \ \& \ y = 1$ $[p_{01}(x_{11})]$

On query χ, $SQ(c, D)$ returns any value $\hat{P}_\chi \in \left[P_\chi - \tau, P_\chi + \tau\right]$. To say that $\mathcal{C}$ is *SQ-learnable* if we require $\tau \geq poly\left(\frac{\varepsilon}{n}\right)$

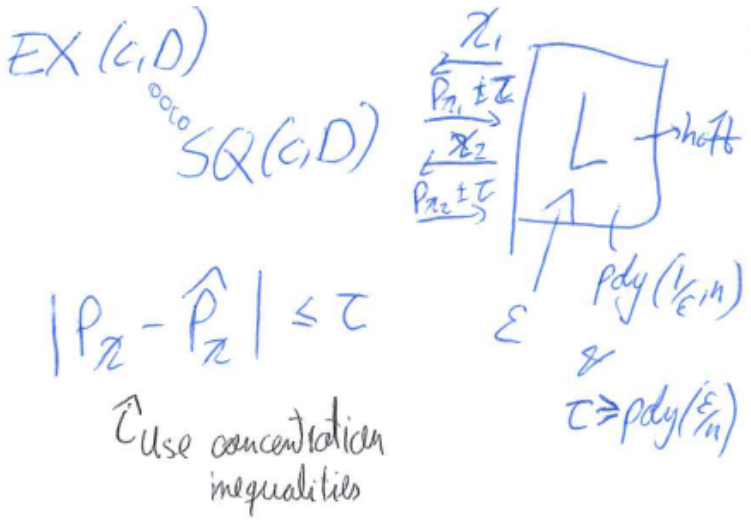

Figure 3: Statistical Query Learning.

Theorem 1.5 (uninteresting)**.** *If $\mathcal{C}$ is SQ-learnable, then $\mathcal{C}$ is PAC-learnable.*

Theorem 1.6. *If $\mathcal{C}$ is SQ-learnable, then $\mathcal{C}$ is PAC-learnable with noise.*

Proof. We want to estimate

$$P_\chi := \mathbb{P}_{\langle x,y\rangle\sim EX(c,D)}[\chi(x,c(x))=1], \tag{1.9}$$
$$P_\chi^\eta := \mathbb{P}_{\langle x,y\rangle\sim EX^\eta(c,D)}[\chi(x,y)=1], \tag{1.10}$$

but we don't have access to the $EX(c,D)$ subroutine. Consider

$$X_2 := \{x\in X \ : \ \chi(x,0)=\chi(x,1)\}, \tag{1.11}$$

and define the following partition of X

$$X_1 := \{x\in X \ : \ \chi(x,0)\neq\chi(x,1)\}, \tag{1.12}$$

X_1 and X_2 are disjoint and clearly $X_1\cup X_2 = X$. Now consider $p_1 = D[X_1]$ and $p_2 = 1-p_1 = D[X_2]$. For $x\in X$ we have the normalized conditioned distributions

$$D_1[X] = \frac{D[X]}{p_1} \to P_\chi^1 = \mathbb{P}_{EX(c,D_1)}[\chi=1], \tag{1.13}$$
$$D_2[X] = \frac{D[X]}{p_2} \to P_\chi^2 = \mathbb{P}_{EX(c,D_2)}[\chi=1]. \tag{1.14}$$

This just introduced notation. By considering conditioned probabilities, we now can compute

$$P_\chi^\eta = (1-\eta)P_\chi + \eta\left(p_1\underbrace{\mathbb{P}_{x\sim D_1,y=\neg c(x)}[\chi(x,y)=1]}_{\mathbb{P}_{EX(c,D)}[\chi(x,c(x))=0]=:1-P_\chi^1} + p_2\underbrace{\mathbb{P}_{x\sim D_2,y=\neg c(x)}[\chi(x,y)=1]}_{\mathbb{P}_{EX(c,D_2)}[\chi(x,c(x))=1]=:P_\chi^2}\right) \tag{1.15}$$
$$= (1-\eta)P_\chi + \eta\left(p_1(1-P_\chi^1)+p_2P_\chi^2\right). \tag{1.16}$$

We can solve this for P_χ

$$P_\chi = \frac{1}{1-\eta}\left[P_\chi^\eta - \eta\left(p_1(1-P_\chi^1) + p_2 P_\chi^2\right)\right]. \tag{1.17}$$

Bu now we can empirically estimate all the elements in the RHS using noisy data. The only tricky one is P_χ^1 but we have

$$P_\chi^1 = \frac{P_\chi^\eta - p_2 P_\chi^2 - \eta p_1}{(1-2\eta)p_1}, \tag{1.18}$$

so we are fine. To finish the proof in detail one has to do a robustness analysis of the expression that we are using to estimate P_χ. See the textbook for more details. □

Claim 1.7. *"Almost" every PAC-learning algorithm has an SQ-algorithm.*

We can actually find counterexamples to the claim, hence the "almost" in the statement.

1.3.1 PAC-learnable but not SQ-learnable

Consider the space $X = \{0,1\}$. For any subset $S \subset \{1,\dots,n\}$ we can define the functions

$$f_S(\vec{x}) = \sum_i c_i x_i \mod 2 = \bigoplus_{i\in S} x_i, \tag{1.19}$$

and the associated pairs $\langle \vec{x}, f_S(\vec{x})\rangle$. In turns out that this is PAC-learnable because examples correspond to linear equations but you can't can get the same information statistically.