TITLE

Hung C. Le Tran

January 21, 2024

Course: COURSE

Section: SECTION

Professor: PROFESSOR

At: The University of Chicago

Quarter: QUARTER

Course materials: COURSE MATERIALS

Disclaimer: This document will inevitably contain some mistakes, both simple typos and serious logical and mathematical errors. Take what you read with a grain of salt as it is made by an undergraduate student going through the learning process himself. If you do find any error, I would really appreciate it if you can let me know by email at conghungletran@gmail.com.

Contents

3 A Formal Learning Model

Definition 3.1 (PAC Learnable)

A hypothesis class $\mathcal{H}$ is PAC learnable if there exists $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm such that: for every $\varepsilon, \delta \in (0,1)$, distribution $\mathcal{D}$ over $\mathcal{X}$, labeling function $f : \mathcal{X} \to \{0,1\}$, if the realizable assumption holds with respect to $\mathcal{H}, \mathcal{D}, f$, then when running the algorithm on $m \geq m_{\mathcal{H}}(\varepsilon, \delta)$ i.i.d examples (generated by $\mathcal{D}$ and labeled by f), the algorithm returns a hypothesis h such that, with probability of $1 - \delta$ (over the choice of the m training samples),

$$L_{(D,f)}(h) \leq \varepsilon$$

Corollary 3.2

Every finite hypothesis class is PAC learnable with sample complexity

$$m_{\mathcal{H}}(\varepsilon, \delta) \leq \left\lceil \frac{\log(|\mathcal{H}|/\delta)}{\varepsilon} \right\rceil$$

Definition 3.3 (Agnostic PAC Learnable)

A hypothesis class $\mathcal{H}$ is agnostic PAC learnable if there exists $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm such that: for every $\varepsilon, \delta \in (0,1)$, distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, when running the algorithm on $m \geq m_{\mathcal{H}}(\varepsilon, \delta)$ i.i.d examples (generated by $\mathcal{D}$), the algorithm returns a hypothesis h such that, with probability $1 - \delta$ (over the choice of the m training samples),

$$L_D(h) \leq \min_{h' \in \mathcal{H}} L_D(h') + \varepsilon$$

Remark

If $\mathcal{H}$ is agnostic PAC learnable then it is PAC learnable too. Because in the PAC setting, plus the realizability assumption holds, then

$$\min_{h' \in \mathcal{H}} L_D(h') = 0$$

and we have PAC learnability immediately.

Definition 3.4 (Generalized Loss)

For some loss function $l : \mathcal{H} \times Z \to \mathbb{R}_+$, and predictor $h \in \mathcal{H}$,

$$L_D(h) \coloneqq \mathbb{E}_{z \sim \mathcal{D}}[l(h, z)]$$

Definition 3.5 (Agnostic PAC Learnable for generalized loss)

A hypothesis class $\mathcal{H}$ is agnostic PAC learnable with respect to a set Z, loss function $l : \mathcal{H} \times Z \to \mathbb{R}_+$ if there exists $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm such that: for every $\varepsilon, \delta \in (0,1)$, distribution $\mathcal{D}$ over Z, when running the algorithm on $m \geq m_{\mathcal{H}}(\varepsilon, \delta)$ i.i.d examples (generated by $\mathcal{D}$), the algorithm returns a hypothesis h such that, with probability $1 - \delta$ (over the choice of the m training samples),

$$L_D(h) \leq \min_{h' \in \mathcal{H}} L_D(h') + \varepsilon$$

4 Learning via Uniform Convergence

Definition 4.1 (ε-representative sample)

A training set S is called ε-representative (wrt $Z, \mathcal{H}, l, \mathcal{D}$) if

$$\forall\, h \in \mathcal{H}, |L_S(h) - L_{\mathcal{D}}(h)| \geq \varepsilon$$

The name says it all: the training set is representative of the distribution if the training loss is close to the true loss, regardless of which predictor in the hypothesis class.

Lemma 4.2

If S is $\varepsilon/2$-representative then $ERM_{\mathcal{H}}(S)$ satisfies

$$L_D(h_S) \leq \min_{h \in \mathcal{H}} L_D(h) + \varepsilon$$

Definition 4.3 (Uniform convergence)

A hypothesis class $\mathcal{H}$ has the uniform convergence property (wrt Z, l) if there exists $m_{\mathcal{H}}^{UC} : (0,1)^2 \to \mathbb{N}$ such that: for every $\varepsilon, \delta \in (0,1)$, distribution $\mathcal{D}$ over Z, if S is a sample of $m \geq m_{\mathcal{H}}^{UC}(\varepsilon, \delta)$ examples drawn i.i.d from $\mathcal{D}$, then, with probability of at least $1 - \delta$, S is ε-representative.

The uniform convergence property of a hypothesis class essentially means that it is nice enough for the training sets S to be representative of $\mathcal{D}$.

Corollary 4.4

If $\mathcal{H}$ has uniform convergence with $m_{\mathcal{H}}^{UC}$ then it is agnostic PAC learnable with sample complexity $m_{\mathcal{H}}(\varepsilon, \delta) \leq m_{\mathcal{H}}^{UC}(\varepsilon/2, \delta)$. And $ERM_{\mathcal{H}}$ paradigm is a successful agnostic PAC learner for $\mathcal{H}$.

Corollary 4.5

If $\mathcal{H}$ is finite, $l : \mathcal{H} \times Z \to [0,1]$ (to use Hoeffding's). Then $\mathcal{H}$ has uniform convergence property, that is,

$$m_{\mathcal{H}}^{UC}(\varepsilon, \delta) \leq \left\lceil \frac{\log(2|\mathcal{H}|/\delta)}{2\varepsilon^2} \right\rceil$$

The difference in the bound in this case and the PAC learnable case is that we're approximating around some L_S that is not necessarily 0.

Since uniform convergence can be achieved, finite $\mathcal{H}$ is also agnostic PAC learnable

Corollary 4.6

Using ERM, finite $\mathcal{H}$ is agnostic PAC learnable with sample complexity:

$$m_{\mathcal{H}}(\varepsilon, \delta) \leq m_{\mathcal{H}}^{UC}(\varepsilon/2, \delta) \leq \left\lceil 2\frac{\log(2|\mathcal{H}|/\delta)}{2(\varepsilon/2)^2} \right\rceil = \left\lceil \frac{2\log(2|\mathcal{H}|/\delta)}{\varepsilon^2} \right\rceil$$

5 The Bias-Complexity Tradeoff

Definition 5.1 (Error Decomposition)

$$L_D(h_S) = \varepsilon_{app} + \varepsilon_{est}$$

where $\varepsilon_{app} = \min_{h \in \mathcal{H}} L_D(h), \varepsilon_{est} = L_{\mathcal{D}}(h_S) - \varepsilon_{app}$.

Note that ε_{app} does not depend on the training set, nor the hypothesis. It solely depends on the hypothesis class $\mathcal{H}$. It decreases when enlarging the hypothesis class. This term measures how much risk we've restricted ourselves to, i.e., how much inductive bias we have.

Meanwhile, for finite $\mathcal{H}$, as we've shown ε_{est} increases (logarithmically) with $|\mathcal{H}|$ and decreases with m.

6 The VC-Dimension

Definition 6.1 (Shattering)

$\mathcal{H}$ shatters a finite set $C \subset \mathcal{X}$ if the restriction of $\mathcal{H}$ to C is the set of all functions from C to $\{0,1\}$. That is, $|\mathcal{H}_C| = 2^{|C|}$.

Definition 6.2 (VC-dimension)

The VC-dimension of a hypothesis class $\mathcal{H}$, denoted $VCDim(\mathcal{H})$, is the maximal size of a set $C \subset \mathcal{X}$ that can be shattered by $\mathcal{H}$. It is ∞ when $\mathcal{H}$ can shatter sets of arbitrarily large size.

Theorem 6.3

Let $\mathcal{H}$ have $VCDim(\mathcal{H}) = \infty$. Then $\mathcal{H}$ is not PAC learnable.

The converse is also true.

Theorem 6.4 (Fundamental Theorem of Statistical Learning)

Let $\mathcal{H}$ be a hypothesis class from $\mathcal{X}$ to $\mathcal{Y} = \{0, 1\}$, let loss function be 0-1 loss. Then, TFAE:

(a) $\mathcal{H}$ has uniform convergence property.

(b) Any ERM rule is a successful agnostic PAC learner for $\mathcal{H}$.

(c) $\mathcal{H}$ is agnostic PAC learnable.

(d) $\mathcal{H}$ is PAC learnable.

(e) Any ERM rule is a successful PAC learner for $\mathcal{H}$.

(f) $\mathcal{H}$ has a finite VC-dimension.

Not only does the VC-dimension characterize PAC learnability; it even determines the sample complexity.

Remark

This extends to regression with absolute/squared loss. However, theorem does not hold for all learning tasks. In particular, learnability is sometimes possible, even though uniform convergence does not hold. Someetimes, ERM rule fails but learnability is possible with other learning rules.

7 Nonuniform Learnability

Definition 7.1 (Nonuniformly Learnable)

$\mathcal{H}$ is nonuniformly learnable if there exists a learning algorithm, A, and $m_{\mathcal{H}}^{NUL} : (0,1)^2 \times \mathcal{H} \to \mathbb{N}$ such that, for every $\varepsilon, \delta \in (0,1)$, $h \in \mathcal{H}$, if $m \geq m_{\mathcal{H}}^{NUL}(\varepsilon, \delta, h)$ then for every distribution $\mathcal{D}$, with probability of at least $1 - \delta$ over the choice of $S \sim \mathcal{D}^m$, we have

$$L_{\mathcal{D}}(A(S)) \leq L_{\mathcal{D}}(h) + \varepsilon$$

i.e., that $A(S)$ is (ε, δ)-competitive with every other hypothesis in the class.

Remark

Recall that agnostic PAC learnability also requires

$$L_{\mathcal{D}}(A(S)) \leq \min_{h' \in \mathcal{H}}(L_{\mathcal{D}}(h')) + \varepsilon \leq L_{\mathcal{D}}(h') + \varepsilon \; \forall \; h' \in \mathcal{H}$$

which essentially requires $A(S)$ to be (ε, δ)-competitive with every hypothesis in $\mathcal{H}$ too. But the difference is that in agnostic PAC, $m_{\mathcal{H}}$ is only allowed to depend on (ε, δ) while for nonuniform learnability, $m_{\mathcal{H}}^{NUL}$ is allowed to depend on (ε, δ, h); thus the "non-uniform"ness.

It is thus an easy consequence that agnostic PAC learnable $\Rightarrow$ nonuniformly learnable.

Theorem 7.2

$\mathcal{H}$ of binary classifiers is nonuniformly learnable iff it is a countable union of agnostic PAC learnable hypothesis classes.

Remark

Look up nonuniform learnable again.

Theorem 7.3

Let $\mathcal{H} = \bigcup_{n \in \mathbb{N}} \mathcal{H}_n$ where $\mathcal{H}_n$ has uniform convergence with $m_{\mathcal{H}_n}^{UC}$. Let $w : \mathbb{N} \to [0,1], w(n) = \frac{6}{\pi^2 n^2}$. Then $\mathcal{H}$ is uniformly learnable using SRM rule with rate

$$m_{\mathcal{H}}^{NUL} \leq m_{\mathcal{H}_{n(h)}}^{UC}\left(\varepsilon/2, \frac{6\delta}{(\pi n(h))^2}\right)$$

SRM Rule equivalently by applying Hoeffding's inequality, with $\mathcal{H} = \bigcup_{n \in \mathbb{N}} \{h_n\}$ for countable hypothesis

class.

$$SRM(S) = \arg\min_{h \in \mathcal{H}} \left[L_S(h) + \sqrt{\frac{-\log(w(h)) + \log(2/\delta)}{2m}} \right]$$

so we assign more weight to hypotheses that are more likely to be correct.

So far in the book, we have studied the statistical perspective of learning, namely how many samples are needed for learning (hence, *sample complexity*). Now we turn to *computational complexity*.