Understanding Machine Learning – A theory Perspective (part 3)

Shai Ben-David

University of Waterloo

MLSS at MPI Tubingen, 2017

The fundamental theorem (qualitative)

Theorem: Given a class H of binary valued functions the following statements are equivalent:

a. H has the Uniform Convergence Property
b. ERM is an agnostic PAC learner for H
c. H is agnostic PAC learnable
d. H is PAC learnable
e. VCdim(H) is finite

Main tool for (e) implies (a)

The Shatter function

For a class H define a function $\Pi_H: N \rightarrow N$

as $\Pi_H(m) = \max_{\{A: |A|=m\}} |\{h|_A : h \text{ in } H\}|$

Some basic properties of the shatter function:

1. For every $m \leq VCdim(H)$, $\Pi_H(m) = 2^m$
2. For every $m > VCdim(H)$, $\Pi_H(m) < 2^m$

The Sauer (Shelah, Perles) lemma

For every class H of finite VC-dimension, d,

For every m,

$$\Pi_H(m) \leq \Sigma_{i=0}^{d} \text{ (m choose i)} \leq m^d$$

Quantitative version of the Fundamental Theorem

For some constants C_1, C_2, for every d and every class H of binary valued functions such that VCdim(H)=d,

1. H has Uniform Convergence property with
   $C_1(d+\log(1/\delta))/\varepsilon^2 < m^{uc}_H(\varepsilon, \delta) < C_1(d+\log(1/\delta))/\varepsilon^2$

2. H is **agnostic** PAC learnable with
   $C_1(d+\log(1/\delta))/\varepsilon^2 < m_H(\varepsilon, \delta) < C_1(d+\log(1/\delta))/\varepsilon^2$

3. H is PAC learnable with
   $C_1(d+\log(1/\delta))/\varepsilon < m_H(\varepsilon, \delta) < C_1(d+\log(1/\delta))/\varepsilon$

How to compute VC dimension

As a rule of thumb, the VC dimension of a class

Is often equal to the number of parameters need to be set to specify a specific h in H.

(think of H_{init}, $H_{intrvals}$, $H_{rectangles}$, HS^n)

Is the story complete now?

- Issue 1 – finite VC classes may be too restricted.
- Issue 2 – computational complexity

Non-Uniform Learn - Definition

A class H is *NonUniformly learnable* if

There is a function $m_H: H x(0,1)^2 \rightarrow N$

and a learning algorithm A,

s.t. for every distribution P over XxY

and every ε, $\delta > 0$, *for every h in H*

for samples S of size $m > m_H(h, \varepsilon, \delta)$

generated i.i.d.by P,

$Pr[L_P(A(S)) > \mathbf{L_P(h)} + \varepsilon] < \delta$

Non-Uni characterization - Statement

Theorem: A class H is NonUniformly learnable if and only if there exist classes

$\{H_n : n \text{ in } N\}$ such that:

1. Each H_n has the uniform convergence property.

And,

2. $H = U_{n \text{ in } N} H_n$

Some NonUni Learnable classes

- The class of all polynomials epi-sets

  H={h_p: p a polynonial in x}

  where, $h_p(x) = 1$ if and only if $p(x)>0$.
- The class of all (characteristic functions of) finite subsets of (any) X.

- The class of all finite unions of rectangles.

Some classes are not NUL

If H shatters an infinite set, then H is not (even) NonUni learnable.

(in particular, the class of ALL functions over any infinite domain).

Proof of easy direction

Assume H is NonUni learnable,

Define, for every n,

$H_n = \{h \text{ in } H : m_H (h, 1/7, 1/8) < n\}$

Note that each of these classes must have finite VCdim, and therefore has Uniform Convergence, and their union covers H.

Hard direction

Step 1: Weight functions.

We define a *weight function* to be any function $w : N \rightarrow [0,1]$ such that

$\Sigma_n w(n) \leq 1.$

Examples: $w(n) = 1/2n^2$

or $w(n) = 1/2^n$

Rewriting the m function

Given a class H and a representation of H os a union of H_n's, each enjoying uniform convergence, define for any n

$\varepsilon_n(m, \delta)=\min\{\varepsilon: m_{Hn}(\varepsilon, \delta) < m\}$

(namely, the minimal error that an m-size sample can guarantee)

Hard direction

The NonUniform generalization (loss) bound

For every weight function w, every prob. Dist P every δ and every m, with probability

> $(1-\delta)$,

For all h in H

$L_P(h) \leq L_S(h) + \min_{\{n:\ h \text{ in } H_n\}} \varepsilon_n (m, w(n)\ \delta)$

Hard direction

The bound minimization algorithm –

Structural Risk Minimization (SRM):

Given H, a decomposition of H to H_n's of finite VCdim each and a weight function w,

On a labeled training sample S of size m,

Find h in H that minimizes the above error bound:

$L_S(h) + \min_{\{n:\ h \text{ in } H_n\}} \varepsilon_n\,(m, w(n)\,\delta)$

Hard direction

The resulting sample complexity function:

$m(h,\varepsilon,\delta) = m^{uc}{}_{Hn(h)}(\varepsilon/2, w(n(h))\ \delta)$

Applications of SRM

SRM has many applications, usually referred to as “ERM with regularization”:

Adding to the empirical error a “penalty” on complex (or otherwise, undesirable) h’s.

Example include

1. Norm of a linear classifier
2. Description length
3. Small margins
4. Low prior likelihood

Description length - definition

A description language for a class H is a function

G: H ➔ Finite binary strings

Such that the range of G is prefix-free.

Kraft inequality

Any collection T of binary strings that is prefix-free, satisfies

$\Sigma_{\sigma \text{ in } T}\ 2^{-|\sigma|} \leq 1$

Corollary: For $H=\{h_1, h_2, \ldots h_n, \ldots\}$,

We can use any description language for H to define weights $w(n) = 2^{-|G(hn)|}$

Description length bound and Ocam's Razor

The resulting SRM algorithm is :

pick h that minimizes

$$L_S(h) + \mathrm{sqrt}\{(|G(h)| + \ln(1/\delta))/2m\}$$