Continuous Logic and Learning Bounds

Aaron Anderson

UPenn

June 24, 2025

Model Theory to Learnability

- Let $\mathcal{C} = \{c_y : y \in Y\}$ be a class of subsets of X indexed by Y.
- $\mathcal{C}$ is NIP/stable when there is an NIP/stable formula $\phi(x; y)$ such that $x \in c_y \iff \phi(x; y)$.
- The properties in each row are equivalent:

Model Theory	Combinatorics	Learning Theory
NIP	finite VC dimension	PAC learnable
stable	finite Littlestone dimension	online learnable

Continuous Logic to Learnability

- Let $\mathcal{H} = \{h_y : y \in Y\}$ be a class of functions $X \to [0, 1]$ indexed by Y.
- $\mathcal{H}$ is NIP/stable when there is an NIP/stable formula $\phi(x; y)$ of continuous logic such that $h_y(x) = \phi(x; y)$.
- The properties in the table have been generalized to $\mathcal{H}$, but the connections are understudied.

New Learnable Function Classes

Theorem (A., Benedikt)

A class $\mathcal{H}$ of functions $X \to [0,1]$ is stable iff it is online learnable.

Theorem (A., Benedikt)

The randomization of a PAC/online learnable function class $\mathcal{H}$ is also PAC/online learnable.

Generalizing VC Dimension to Continuous Logic

Theorem (Ben Yaacov)

A formula ϕ of continuous logic is NIP iff the class of functions it defines has finite γ-fat-shattering dimension for all $\gamma > 0$.

Definition

Let $\mathcal{H}$ be a class of functions $X \to [0,1]$ and let $\gamma > 0$. We say $\mathcal{H}$ has *γ-fat-shattering dimension* at least n when there are

- $x_1, \ldots, x_n \in X$
- $s_1, \ldots, s_n \in [0,1]$
- For every $E \subseteq \{1, \ldots, n\}$, a function $h_E \in \mathcal{H}$ satisfying
  - if $i \in E$, $h_E(x_i) \geq s_i + \gamma$
  - if $i \notin E$, $h_E(x_i) \leq s_i - \gamma$.

Probably Approximately Correct Learning

A class $\mathcal{H}$ of functions $X \to [0,1]$ is PAC learnable when for every $\varepsilon, \delta > 0$, there is n such that when...

- $(x_1, y_1), \ldots, (x_n, y_n) \in X \times [0,1]$ are i.i.d. random,
- we can choose $h \in \mathcal{H}$ (hoping that $h(x_{n+1}) \approx y_{n+1}$) such that
- with probability at least $1 - \delta$,
- $\mathbb{E}\left[|y_{n+1} - h(x_{n+1})|\right]$ is within ε of the best case for all $h \in \mathcal{H}$.

We call $n = n(\varepsilon, \delta)$ the *sample complexity*.

PAC Learning Bound

Theorem (Bartlett, Long)

The sample complexity $n(\varepsilon, \delta)$ of PAC-learning $\mathcal{H}$ is bounded by

$$O\left(\frac{1}{\epsilon^2} \cdot \left(\mathrm{FatSHDim}_{\frac{\epsilon}{9}}(\mathcal{H}) \cdot \log^2\left(\frac{1}{\epsilon}\right) + \log\left(\frac{1}{\delta}\right)\right)\right)$$

In fact, $\mathcal{H}$ is PAC-learnable if and only if the γ-fat-shattering dimension is finite for all $\gamma > 0$.

Hu et al. extended this to learning a class of measures on $\mathcal{H}$, at the cost of a much worse bound.

We'll use continuous logic to find examples of learnable classes and improve the Hu et al. bound.

The Expectation Class

Suppose (Ω, Σ, μ) is a probability space and $\mathcal{F} = (\mathcal{H}_\omega : \omega \in \Omega)$ is a family of function classes $\mathcal{H}_\omega = (h_{\omega,y} : y \in Y)$.

Definition

Assuming measurability, define $\mathbb{E}\mathcal{F}_y : X \to [0,1]$ by

$$\mathbb{E}\mathcal{F}_y(x) = \mathbb{E}\left[h_{\omega,y}(x)\right].$$

We call the class $\mathbb{E}\mathcal{F} = \{\mathbb{E}\mathcal{F}_y : y \in Y\}$ the *expectation class* of $\mathcal{F}$.

Think of $\mathbb{E}\mathcal{F}$ and every $\mathcal{H}_\omega$ as a class of functions $h : X \to [0,1]$ indexed by Y.

The Randomization

If the classes $\mathcal{H} \in \mathcal{F}$ are uniformly definable in some structure M, then $\mathbb{E}\mathcal{F}$ is definable in a structure M^R (of continuous logic) called the *randomization* of M, whose elements are M-valued random variables.

Theorem (Ben Yaacov, Keisler)

- *If $\mathcal{F}$ is uniformly NIP/stable, then $\mathbb{E}\mathcal{F}$ is NIP/stable.*
- *If a structure M is NIP/stable, then M^R is NIP/stable.*

PAC Learning Expectations

Theorem (A., Benedikt)

If each $\mathcal{H} \in \mathcal{F}$ has $\mathrm{FatSHDim}_{\frac{\epsilon}{50}}(\mathcal{H}) \leq d$, one can PAC learn the expectation class $\mathbb{E}\mathcal{F}$ with sample complexity

$$O\left(\frac{d}{\epsilon^4} \cdot \log^2 \frac{d}{\epsilon} + \frac{1}{\epsilon^2} \cdot \log \frac{1}{\delta}\right).$$

- $\mathrm{FatSHDim}$ can be used to bound *Rademacher mean width*
- Rademacher mean width can be used to bound sample complexity
- Adapt Ben Yaacov's proof that *Gaussian* mean width is preserved under randomization

Online Learning

- At step i, an adversary chooses $(x_i, y_i) \in X \times [0, 1]$
- Given x_i, you guess $y_i' \approx y_i$ (you can use randomness)
- The adversary tells you y_i, penalizes you $|y_i - y_i'|$
- After n steps, compare to the best strategy $y_i' = h(x_i)$ for $h \in \mathcal{H}$.
- Call the difference in penalty the *regret*.
- $\mathcal{H}$ is online learnable if whatever the adversary does, regret is sublinear in n.

Online Learning Bounds

To bound regret in online learning, replace our existing notions with *sequential* versions, replacing subsets $E \subseteq \{1, \ldots, n\}$ with branches of a binary tree of depth n:

Theorem (Rakhlin, Sridharan, Tewari)

Finite γ-sequential-fat-shattering dimension is equivalent to online learnability, with bounds given.

Their proof goes through sequential Rademacher mean width.

Our Online Learning Results

Theorem (A., Benedikt)

- *Stability in continuous logic is equivalent to finite γ-sequential-fat-shattering dimension for all $\gamma > 0$.*
- *Sequential Rademacher mean width, and thus online learnability, is preserved under randomization.*

Theorem (A., Benedikt)

The minimax regret of online learning for the randomization class of $\mathcal{H}$ with γ-sequential-fat-shattering dimension at most d on a run of length n is at most

$$4 \cdot \gamma \cdot n + 12 \cdot (1 - \gamma) \cdot \sqrt{d \cdot n \cdot \log\left(\frac{2 \cdot e \cdot n}{\gamma}\right)}.$$

Thank you!

For downloadable slides, see

`https://awainverse.github.io/talks/learningrandom/`

For valued fields enthusiasts: Talk to me about RCMVF.