泛化理论

序章 泛化理论简介

§0.3.3 PAC learnable (example)

@ 滕佳烨

[ref] Understanding Machine Learning: From Theory to Algorithms, Shai Shalev-Shwartz and Shai Ben-David (2014)

Recall:

No-free-lunch: there is no universal learner!
PAC learning (**Provably Approximately Correct**): Introduce the prior knowledge via "realizable" over a function class.
Agnostic PAC learning: introducing the prior knowledge via adaptive loss.

Today's topic:
We will give an example to show how to achieve PAC learnable
: finite function class $\mathcal{H}$ is PAC-learnable!

PAC learning

DEFINITION 3.1 (PAC Learnability) A hypothesis class $\mathcal{H}$ is PAC learnable if there exist a function $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm with the following property: For every $\epsilon, \delta \in (0,1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$, and for every 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 learning algorithm on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d. examples generated by $\mathcal{D}$ and labeled by f, the algorithm returns a hypothesis h such that, with probability of at least $1 - \delta$ (over the choice of the examples), $L_{(\mathcal{D},f)}(h) \leq \epsilon$.

$m_{\mathcal{H}}(\epsilon, \delta)$: **sample complexity**. Measure how many samples required to guarantee PAC (minimal).
Informally, when $m_{\mathcal{H}}(\epsilon, \delta) < \infty$, we call a function class “PAC learnable”.

Example: finite function class is PAC-learnable!

COROLLARY 3.2 *Every finite hypothesis class is PAC learnable with sample complexity*

$$m_{\mathcal{H}}(\epsilon,\delta) \le \left\lceil \frac{\log(|\mathcal{H}|/\delta)}{\epsilon} \right\rceil .$$

Hint: for a given **bad** function in $\mathcal{H}$, it predicts each data point correctly with probability $\le 1-\epsilon$. Therefore, for the training set with m samples, the training set has zero loss with probability $\le (1-\epsilon)^m = e^{m\log(1-\epsilon)}$. That is to say:

a bad function has small probability to be chosen at the training phase.

Using union bound over all the functions, the probability becomes $\le |\mathcal{H}|e^{m\log(1-\epsilon)} = e^{\log|\mathcal{H}|+m\log(1-\epsilon)}$. To make the probability equal to δ, we requires that $\log|\mathcal{H}| + m\log(1-\epsilon) = \log\delta$, leading to $m = -\frac{\log\frac{|\mathcal{H}|}{\delta}}{\log(1-\epsilon)} \approx \frac{\log\frac{|\mathcal{H}|}{\delta}}{\epsilon}$

Formally, see Page 39 in the reference book.

Take-away messages

(a) Finite function class is PAC-learnable with $m_{\mathcal{H}} = \frac{\log\frac{|\mathcal{H}|}{\delta}}{\epsilon}$.

All the slides will be available at www.tengjiaye.com/generalization soon.

@ 滕佳烨

Thanks!