Statistics Learning Theorem:PAC Learing

2019.02.02

Review:Finite Hypothesis Classes- Assumptions

Assume that

- Realizability Assumption:There exist $h^* \in \mathcal{H}$ such that $L_{D,f}(h^*) = 0$
- i.i.d assumption: we assume that the training data $S = \{x_1, x_2, \cdots, x_m\}$ are sampled i,i,d from $\mathcal{D}$. Therefore, the distribution of S is D^m
- $|\mathcal{H}|$ is finite

Review:Finite Hypothesis Classes - Theorem

Under above three assumptions, let $\delta \in (0, 1)$ and $\varepsilon > 0$ and let m be an interger that satisfies

$$m \geq \frac{\log(|\mathcal{H}|/\delta)}{\varepsilon}$$

Then, for any labeling function f, and for any distribution, $\mathcal{D}$, with probability of at least $1 - \delta$ over the choice of an sample S of size m, we have every ERM prediction rule h_S, it holds that

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

Review:Finite Hypothesis Classes -Proof

We try to uppper bound

$$\mathcal{D}^m(\{S : L_{D,f}(h_S) > \varepsilon\})$$

Let

$$\mathcal{H}_B = \{h \in \mathcal{H} : L_{(D,f)}(h) > \varepsilon\}$$

$$M = \{S : \exists h \in \mathcal{H}_B, L_S(h) = 0\}$$

Where $\mathcal{H}_B$ collect those 'bad' prediction rule and M collect those sample to mislead the learner.

Review:Finite Hypothesis Classes–Proof

Then we have

$$\mathcal{D}^m(\{S : L_{D,f}(h_S) > \varepsilon\}) \leq \mathcal{D}^m(M) \leq \sum_{h \in \mathcal{H}_B} \mathcal{D}^m(\{S : L_S(h) = 0\})$$

and

$$\mathcal{D}^m(\{S : L_S(h) = 0\}) = \mathcal{D}^m(\{S : \forall i, h(x_i) = f(x_i))$$

$$= \prod_{i=1}^{m} \mathcal{D}(\{x_i : h(x_i) = f(x_i)\})$$

and we know that $\forall h \in \mathcal{H}_B$

$$\mathcal{D}(\{x_i : h(x_i) = f(x_i)\}) = 1 - L_{D,f}(h) \leq 1 - \varepsilon$$

Review:Finite Hypothesis Classes–Proof

Since $1-\varepsilon \leq e^{-\varepsilon}$, we have

$$\mathcal{D}^m(\{S : L_S(h) = 0\}) \leq (1-\varepsilon)^m \leq e^{-\varepsilon m}$$

Therefore,

$$\mathcal{D}^m(\{S : L_{D,f}(h_S) > \varepsilon\}) \leq |\mathcal{H}_B e^{-\varepsilon m} \leq |H| e^{-\varepsilon m}$$

Therefore, given (δ, ε), we can bound the above set when sample larege than

$$\frac{\log(|\mathcal{H}|/\delta)}{\varepsilon}$$

Solution for problem

Let x_i be the Bernoulli variable with parameter $p = Pr_{x\sim\mathcal{D}}\{x|h(x) \neq f(x)\}$. That is,

$$x_i = \begin{cases} 1 & p \\ 0 & 1-p \end{cases}$$

Then $L_S(h) = \frac{\sum_{i=1}^m x_i}{m}$. The iid assumption implies that

$$E_{S\sim\mathcal{D}^m}[L_S(h)] = E_{S\sim\mathcal{D}^m}[\frac{\sum_{i=1}^m x_i}{m}] = \frac{\sum_{i=1}^m E_{x_i}[x_i]}{m} = E[x_i] = p = L_{\mathcal{D},f}(h)$$

The $E_{S\sim\mathcal{D}^m}[\frac{\sum_{i=1}^m x_i}{m}] = \frac{\sum_{i=1}^m E_{x_i}[x_i]}{m}$ follows from Fubini's theorem.

Definition of PAC Learnability

A hypothesis class $\mathcal{H}$ is PAC learnable if there exist a function $m_{\mathcal{H}} : (0,1)^2 \rightarrow N$ and a learning algorithm with the following property: For every $\varepsilon, \delta \in (0,1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$, and for every labeling function $f : \mathcal{X} \rightarrow \{0,1\}$, if the realizable assumption holds with respect to $\mathcal{H}, \mathcal{D}, f$,then when running the learning algorithm returns a hypothesis h such that, with probability of at least $1 - \delta$, $L_{\mathcal{D},f}(h) \leq \varepsilon$.

Remark

- Accuracy parameter ε determines how correct the output classifier can be from the optimal one.
- Confidence parameter δ indicates how likely the classifier meet that accuracy requirement.
- Sample Complexity: $m_{\mathcal{H}}$ is the minimal sample to guarantee a probably approximately correct solution.
- As a result, finite hypothesis class is PAC with realizable assumption is PAC learnable with sample complexity

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

Two way to relax the definition

- Removing the realizability assumption: Agnostic PAC
- Learning Problems beyond Binary Classification: Multi-Classification, regression $\rightarrow$ generalized loss function.

Agnostic PAC Learning

There are two aspect we should be able to relax

- Realizability assumption: This assumption assume there exists a almost correct function $h^* \in \mathcal{H}$. This is somehow very strong.
- Labels are fully determined by features: In the original setting, the input features can fully determines the label through f. It is not realistic in many practical problems.

The way out: more realistic model for the data-generating distribution.

Agnostic PAC Learning

To solve the above problems, we allow the distribution $\mathcal{D}$ is over $\mathcal{X} \times \mathcal{Y}$ instead of over $\mathcal{X}$.
In this case, we avoid to introduce the correct labeling function f.
Instead, we define the true risk as

$$L_{\mathcal{D}}(h) = P_{(x,y)}[h(x) \neq y] = \mathcal{D}(\{(x, y) : h(x) \neq y\})$$

and the empirical risk remains the same as before

$$L_S(h) = \frac{|\{i \in [m] : h(x_i) \neq y_i\}|}{m}$$

Agostic PAC Learning

Our goal is to find the some hypothesis, $h : \mathcal{X} \to \mathcal{Y}$ that minimizes the true risk, $L_D(h)$

The Bayes Optimal Predictor:

$$f_{\mathcal{D}}(x) = \begin{cases} 1 & \textit{if} \quad P[y = 1|x] \geq \frac{1}{2} \\ 0 & \textit{otherwise} \end{cases}$$

This solution is optimal in the sense that no other classifier g can have lower risk. That is

$$L_{\mathcal{D}}(f_{\mathcal{D}}) \leq L_{\mathcal{D}}(g)$$

Agnostic PAC Learning

Before giving the definition of agnostic PAC learning, we give two remarks:

- To avoid putting the realizability assumtption, we cannot expect the learning algorithm can achieve the minimal possible error, that of the Bayes predictor.
- We will prove later, once we make no prior assumptions about the data generate process, no algorithm can be guaranteed to find a predictor as good as the Bayes oprimal one.(No Free Lunch Theorem)

To sum up, we require that the best possible error of a predictor in some given Benchmark hypothesis class.

Agonostic PAC Learning: Definition

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

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

Generalized Loss Functions

There are two important way to extend our PAC model

- Multiclass Classification:Document classification(news,sports,biology,medicine): way out:the same structure as before with multiple labels
- Regression: House price, way out: need to modify the loss function

Regression

In the linear regression problem, we general use the following loss function to compute

$$L_{\mathcal{D}}(h) = E_{(x,y)\sim\mathcal{D}}(h(x) - y)^2$$

However, the learning framework should be able to accommendate more type of measure of success.

Generalized Loss Functions

Given the class $\mathcal{H}$, we can define the generalized loss function as follows:

$$l : \mathcal{H} \times \mathcal{X} \times \mathcal{Y} \to R_+$$

However, if our learning task beyond prediction tasks, we can put $l : \mathcal{H} \times Z$ where Z can be any domain of example.(for example, unsupervised learning)

Generalized Loss Functions

we now define the true risk and empirical risk as follows:

$$L_{\mathcal{D}}(h) = E_{z \sim \mathcal{D}}[l(h, z)]$$

$$L_S(h) = \frac{1}{m} \sum_{i=1}^{m} l(h, z_i)$$

Generalized Loss Function:examples

- 0-1 loss:

$$l_{0-1}(h,(x,y)) = \begin{cases} 0 & \textit{if} \quad h(x) = y \\ 1 & \textit{otherwise} \end{cases}$$

- square loss:

$$l_{sq}(h,(x,y)) = (h(x) - y)^2$$

Agnostic PAC Learnability for General Loss Functions

A hypothesis class $\mathcal{H}$ is agnostic PAC learnable with respect to a set Z and a loss function $l : \mathcal{H} \times Z \to R_+$, if there exist a function $m_{\mathcal{H}} : (0,1)^2 \to N$ and a learning algorithm with the following preperty: For every $\varepsilon, \delta \in (0,1)$ and for every distribution $\mathcal{D}$ over Z, when running the learning algorithm on $m \geq m_{mathcalH}(\varepsilon, \delta)$ iid samples generated by $\mathcal{D}$, the algorithm returns $h \in \mathcal{H}$ such that, with probability of at least $1 - \delta$

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

where $L_{\mathcal{D}}(h) = E_{z \sim \mathcal{D}}[l(h, z)]$