ECE595 / STAT598: Machine Learning I

Lecture 24.1: Probably Approximately Correct - Two Ingredients of Generalization

Spring 2020

Stanley Chan

School of Electrical and Computer Engineering
Purdue University

PURDUE
UNIVERSITY

Outline

- Lecture 22 Is Learning Feasible?
- Lecture 23 Probability Inequality
- Lecture 24 Probably Approximate Correct

Today's Lecture:

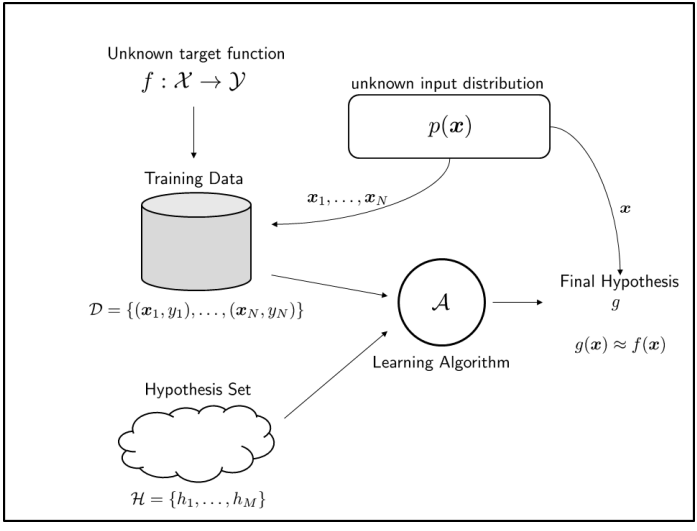
- Two ingredients of generalization
  - Training and testing error
  - Hoeffding inequality
  - Interpreting the bound
- PAC Framework
  - PAC learnable
  - Confidence and accuracy
  - Example

Is Learning Feasible?

- Learning can be **infeasible**.
- Recall the example below.
- Given the training samples, there is no way you can learn and predict.
- You know what you know, and you don't know what you don't know.

$\boldsymbol{x}_n$			y_n	g	f_1	f_2	f_3	f_4
0	0	0	○	○	○	○	○	○
0	0	1	●	●	●	●	●	●
0	1	0	●	●	●	●	●	●
0	1	1	○	○	○	○	○	○
1	0	0	●	●	●	●	●	●
1	0	1	○	○	○	○	○	○
1	1	0		○/●	○	●	○	●
1	1	1		○/●	○	○	●	●

The Power of Probability



In-Sample Error

- Let $\boldsymbol{x}_n$ be a *training* sample
- h: Your hypothesis
- f: The unknown target function
- If $h(\boldsymbol{x}_n) = f(\boldsymbol{x}_n)$, then say training sample $\boldsymbol{x}_n$ is correctly classified.
- This will give you the **in-sample error**

Definition (In-sample Error / Training Error)

Consider a training set $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, and a target function f. The **in-sample error** (or the training error) of a hypothesis function $h \in \mathcal{H}$ is the empirical average of $\{h(\boldsymbol{x}_n) \neq f(\boldsymbol{x}_n)\}$:

$$E_{\text{in}}(h) \stackrel{\text{def}}{=} \frac{1}{N} \sum_{n=1}^{N} [\![h(\boldsymbol{x}_n) \neq f(\boldsymbol{x}_n)]\!], \tag{1}$$

where $[\![\cdot]\!] = 1$ if the statement inside the bracket is true, and $= 0$ if the statement is false.

Out-Sample Error

- Let $\boldsymbol{x}$ be a *testing* sample drawn from $p(\boldsymbol{x})$
- h: Your hypothesis
- f: The unknown target function
- If $h(\boldsymbol{x}) = f(\boldsymbol{x})$, then say testing sample $\boldsymbol{x}$ is correctly classified.
- Since $\boldsymbol{x} \sim p(\boldsymbol{x})$, you need to compute the probability of error, called the **out-sample error**

Definition (Out-sample Error / Testing Error)

Consider an input space $\mathcal{X}$ containing elements $\boldsymbol{x}$ drawn from a distribution $p_{\boldsymbol{X}}(\boldsymbol{x})$, and a target function f. The **out-sample error** (or the testing error) of a hypothesis function $h \in \mathcal{H}$ is

$$E_{\text{out}}(h) \stackrel{\text{def}}{=} \mathbb{P}[h(\boldsymbol{x}) \neq f(\boldsymbol{x})], \tag{2}$$

where $\mathbb{P}[\cdot]$ measures the probability of the statement based on the distribution $p_{\boldsymbol{X}}(\boldsymbol{x})$.

Understanding the Errors

Let us take a closer look at these two error:

$$E_{\text{in}}(h) \stackrel{\text{def}}{=} \frac{1}{N}\sum_{n=1}^{N}[\![h(\boldsymbol{x}_n) \neq f(\boldsymbol{x}_n)]\!],$$

$$E_{\text{out}}(h) \stackrel{\text{def}}{=} \mathbb{P}[h(\boldsymbol{x}) \neq f(\boldsymbol{x})],$$

- Both error are functions of the hypothesis h
- h is determined by the learning algorithm $\mathcal{A}$
- For every $h \in \mathcal{H}$, there is a different $E_{\text{in}}(h)$ and $E_{\text{out}}(h)$
- The training samples $\boldsymbol{x}_n$ are drawn from $p(\boldsymbol{x})$
- The testing samples $\boldsymbol{x}$ are also drawn from $p(\boldsymbol{x})$
- Therefore, $\mathbb{P}[\cdot]$ in $E_{\text{out}}(h)$ is evaluated over $\boldsymbol{x} \sim p(\boldsymbol{x})$

In-sample VS Out-sample

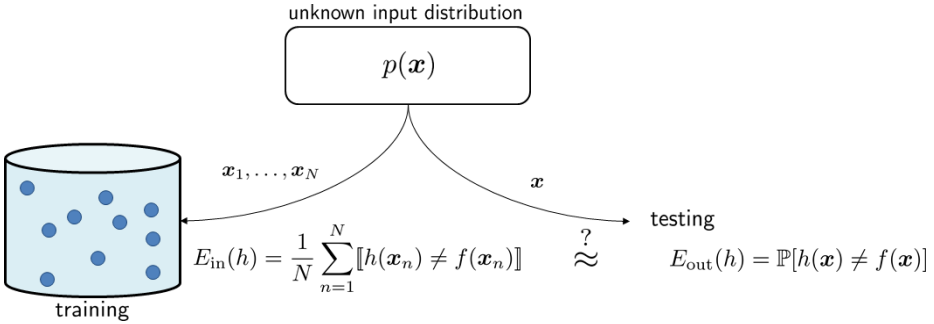
In-Sample Error

$$E_{\text{in}}(h) = \frac{1}{N}\sum_{n=1}^{N}[\![h(\boldsymbol{x}_n) \neq f(\boldsymbol{x}_n)]\!]$$

Out-Sample Error

$$\begin{aligned}
E_{\text{out}}(h) &= \mathbb{P}[h(\boldsymbol{x}) \neq f(\boldsymbol{x})] \\
&= \underbrace{[\![h(\boldsymbol{x}_n) \neq f(\boldsymbol{x}_n)]\!]}_{=1}\mathbb{P}\Big\{h(\boldsymbol{x}_n) \neq f(\boldsymbol{x}_n)\Big\} \\
&\quad + \underbrace{[\![h(\boldsymbol{x}_n) = f(\boldsymbol{x}_n)]\!]}_{=0}\Big(1 - \mathbb{P}\Big\{h(\boldsymbol{x}_n) \neq f(\boldsymbol{x}_n)\Big\}\Big) \\
&= \mathbb{E}\Big\{[\![h(\boldsymbol{x}_n) \neq f(\boldsymbol{x}_n)]\!]\Big\}
\end{aligned}$$

The Role of $p(\boldsymbol{x})$



- Learning is feasible if $\boldsymbol{x} \sim p(\boldsymbol{x})$
- $p(\boldsymbol{x})$ says: Training and testing are related
- If training and testing are unrelated, then hopeless – the deterministic example shown previously
- If you draw training and testing samples with different bias, then you will suffer

A Mathematical Tool

Beside in-sample and out-sample error, we also need a mathematical tool.

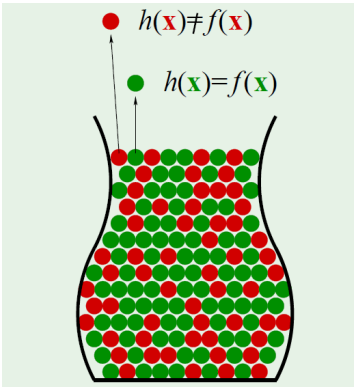
Theorem (Hoeffding Inequality)

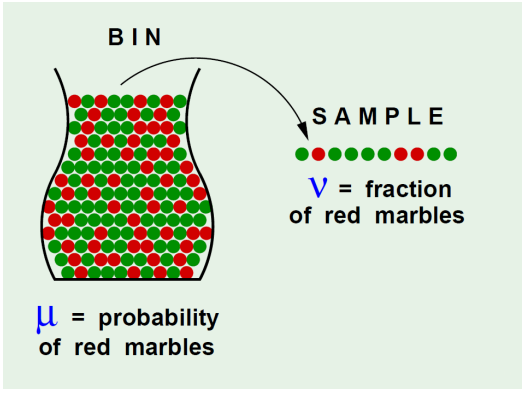
Let $X_1, \ldots, X_N$ be random variables with $0 \leq X_n \leq 1$, then

$$\mathbb{P}\left[|\nu - \mu| > \epsilon\right] \leq 2e^{-2\epsilon^2 N}$$

- We will use Hoeffding inequality to analyze the generalization error
- There are many other inequalities that can serve the same purpose
- Hoeffding requires $0 \leq X_n \leq 1$
- $\nu = \frac{1}{N}\sum_{n=1}^{N} X_n$ is the empirical average
- Probability of how close ν compared to μ
- $\epsilon =$ tolerance level
- $N =$ number of samples

Applying Hoeffinding Inequality to Our Problem





- $X_n = [\![h(\boldsymbol{x}_n) \neq f(\boldsymbol{x}_n)]\!]$ = one sample training error = either 0 or 1
- $\nu = E_{\text{out}} = \frac{1}{N}\sum_{n=1}^{N} X_n$ = training error
- $\mu = E_{\text{in}}$ = testing error

Applying Hoeffinding Inequality to Our Problem

- Therefore, the inequality can be stated as

$$\mathbb{P}\left[|E_{\text{in}}(h) - E_{\text{out}}(h)| > \epsilon\right] \leq 2e^{-2\epsilon^2 N}.$$

- $N =$ number of training samples
- $\epsilon =$ tolerance level
- Hoeffding is applicable because $[\![h(\boldsymbol{x}) \neq f(\boldsymbol{x})]\!]$ is either 1 or 0.
- If you want to be more explicit, then

$$\mathop{\mathbb{P}}_{\boldsymbol{x}_n \sim \mathcal{D}}\left[\left|\frac{1}{N}\sum_{n=1}^{N}[\![h(\boldsymbol{x}_n) \neq f(\boldsymbol{x}_n)]\!] - E_{\text{out}}(h)\right| > \epsilon\right] \leq 2e^{-2\epsilon^2 N}.$$

- The probability is evaluated with respect to $\boldsymbol{x}_n$ drawn from the dataset $\mathcal{D}$

Interpreting the Bound

- Let us look at the bound again:

$$\mathbb{P}\left[|E_{\text{in}}(h) - E_{\text{out}}(h)| > \epsilon\right] \leq 2e^{-2\epsilon^2 N}.$$

- **Message 1**: You can bound $E_{\text{out}}(h)$ using $E_{\text{in}}(h)$.
- $E_{\text{in}}(h)$: You know. $E_{\text{out}}(h)$: You don't know, but you want to know.
- They are close if N is large.

- **Message 2**: The right hand side is independent of h and $p(\boldsymbol{x})$
- So it is a universal upper bound
- Works for any $\mathcal{A}$, any $\mathcal{H}$, any f, and any $p(\boldsymbol{x})$