ECE595 / STAT598: Machine Learning I
Lecture 25 Generalization Bound

Spring 2020

Stanley Chan

School of Electrical and Computer Engineering
Purdue University

PURDUE UNIVERSITY

©Stanley Chan 2020. All Rights Reserved.

Outline

- Lecture 25 Generalization
- Lecture 26 Growth Function
- Lecture 27 VC Dimension

Today's Lecture:

- M Hypothesis
  - PAC framework
  - Guarantee and Possibility
  - The M factor
- Generalization Bound
  - $\mathcal{H}$
  - f
  - Lower and upper limits
- Handling M hypothesis
  - A preview

©Stanley Chan 2020. All Rights Reserved.

Probably Approximately Correct

- **Probably**: Quantify error using probability:

$$\mathbb{P}\left[\,|E_{\text{in}}(h) - E_{\text{out}}(h)| \leq \epsilon\right] \geq 1 - \delta$$

- **Approximately Correct**: In-sample error is an approximation of the out-sample error:

$$\mathbb{P}\left[|E_{\text{in}}(h) - E_{\text{out}}(h)| \leq \epsilon\right] \geq 1 - \delta$$

- If you can find an algorithm $\mathcal{A}$ such that for any ϵ and δ, there exists an N which can make the above inequality holds, then we say that the target function is **PAC-learnable**.

©Stanley Chan 2020. All Rights Reserved.

Guarantee VS Possibility

Difference between deterministic and probabilistic learning.

- **Deterministic**:
- "Can $\mathcal{D}$ tell us something *certain* about f outside $\mathcal{D}$?"
- The answer is NO.
- Anything outside $\mathcal{D}$ has uncertainty. There is no way to deal with this uncertainty.
- **Probabilistic**:
- "Can $\mathcal{D}$ tell us something *possibly* about f outside $\mathcal{D}$?"
- The answer is YES.
- If training and testing have the same distribution $p(\boldsymbol{x})$, then training can say something about testing.
- Assume all samples are independently drawn from $p(\boldsymbol{x})$.

©Stanley Chan 2020. All Rights Reserved.

One Hypothesis versus the Final Hypothesis

- In this equation
$$\mathbb{P}\left[|E_{\text{in}}(h) - E_{\text{out}}(h)| > \epsilon\right] \leq 2e^{-2\epsilon^2 N},$$
the hypothesis h is *fixed*.
- This h is chosen **before** we look at the dataset.
- If h is chosen **after** we look at the dataset, then Hoeffding is invalid.
- We have to choose a h from $\mathcal{H}$ during the learning process.
- The h we choose depends on $\mathcal{D}$.
- This h is the final hypothesis g.
- When you need to choose g from $h_1, \ldots, h_M$, you need to repeat Hoeffding M times.

©Stanley Chan 2020. All Rights Reserved.

The Factor "M"

You can show that

$$
|E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon \qquad \Longrightarrow \qquad
\begin{array}{rl}
 & |E_{\text{in}}(h_1) - E_{\text{out}}(h_1)| > \epsilon \\
\textbf{or} & |E_{\text{in}}(h_2) - E_{\text{out}}(h_2)| > \epsilon \\
 & \ldots \\
\textbf{or} & |E_{\text{in}}(h_M) - E_{\text{out}}(h_M)| > \epsilon.
\end{array}
$$

- To have g, you need to consider $h_1, \ldots, h_M$
- You don't know which h_m to pick; So it is a "OR"
- So there is a sequence of "OR"

©Stanley Chan 2020. All Rights Reserved.

The Factor "M"

$$
\begin{aligned}
\mathbb{P}\Big\{ |E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon \Big\} \overset{(a)}{\leq} \mathbb{P}\Big\{ \quad & |E_{\text{in}}(h_1) - E_{\text{out}}(h_1)| > \epsilon \\
\textbf{or} \quad & |E_{\text{in}}(h_2) - E_{\text{out}}(h_2)| > \epsilon \\
\ldots & \\
\textbf{or} \quad & |E_{\text{in}}(h_M) - E_{\text{out}}(h_M)| > \epsilon \quad \Big\} \\
\overset{(b)}{\leq} \sum_{m=1}^{M} \mathbb{P}\Big\{ \quad & |E_{\text{in}}(h_m) - E_{\text{out}}(h_m)| > \epsilon \quad \Big\}
\end{aligned}
$$

- We need two identities
- (a) If-statement. $\mathbb{P}[A] \leq \mathbb{P}[B]$ if $A \Rightarrow B$
- (b) Union Bound. $\mathbb{P}[A \textbf{ or } B] \leq \mathbb{P}[A] + \mathbb{P}[B]$

©Stanley Chan 2020. All Rights Reserved.

The Factor "M"

- Change this equation

$$\mathbb{P}\left\{ |E_{\text{in}}(h) - E_{\text{out}}(h)| > \epsilon \right\} \le 2e^{-2\epsilon^2 N},$$

- to this equation

$$\mathbb{P}\left\{ |E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon \right\} \le 2Me^{-2\epsilon^2 N}.$$

- So what? M is a constant.
- Bad news: M can be large, or even ∞.
- A linear regression has $M = \infty$.
- Good news: It is possible to bound M.
- We will do it later. Let us look at the interpretation first.

©Stanley Chan 2020. All Rights Reserved.

Outline

- Lecture 25 Generalization
- Lecture 26 Growth Function
- Lecture 27 VC Dimension

Today's Lecture:

- M Hypothesis
  - PAC framework
  - Guarantee and Possibility
  - The M factor
- Generalization Bound
  - $\mathcal{H}$
  - f
  - Lower and upper limits
- Handling M hypothesis
  - A preview

©Stanley Chan 2020. All Rights Reserved.

Learning Goal

- The ultimate goal of learning is to make

$$E_{\text{out}}(g) \approx 0.$$

- To achieve this we need

$$E_{\text{out}}(g) \underset{\substack{\uparrow \\ \text{Hoeffding Inequality}}}{\approx} E_{\text{in}}(g) \underset{\substack{\uparrow \\ \text{Training Error}}}{\approx} 0$$

- Hoeffding inequality holds when N is large
- Training error is small when you train well

©Stanley Chan 2020. All Rights Reserved.

Complex $\mathcal{H}$

- Recall Hoeffding inequality

$$\mathbb{P}\left\{ |E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon \right\} \leq 2Me^{-2\epsilon^2 N}.$$

- If $\mathcal{H}$ is complex, then M will be large. So the approximation by Hoeffding inequality will be worsen.
- But if $\mathcal{H}$ is complex you have more options during training. So training error is improved.
- So there is a trade-off:

$$E_{\text{out}}(g) \quad \underset{\substack{\uparrow \\ \text{worse if } \mathcal{H} \text{ complex}}}{\approx} \quad E_{\text{in}}(g) \quad \underset{\substack{\uparrow \\ \text{good if } \mathcal{H} \text{ complex}}}{\approx} \quad 0$$

- You cannot use a very complex model
- Simple models generalize better

©Stanley Chan 2020. All Rights Reserved.

Complex f

- Recall Hoeffding inequality

$$\mathbb{P}\left\{ |E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon \right\} \le 2Me^{-2\epsilon^2 N}.$$

- Good news: Hoeffding is not affected by f
- So even if f is complex, Hoeffding remains
- Bad news: If f is complex, then very hard to train
- So training error cannot be small
- There is another trade-off:

$$E_{\text{out}}(g) \underbrace{\approx}_{\substack{\uparrow \\ \text{no effect by } f}} E_{\text{in}}(g) \underbrace{\approx}_{\substack{\uparrow \\ \text{worse if } f \text{ complex}}} 0$$

- You can make $\mathcal{H}$ to counteract, but complex $\mathcal{H}$ will make Hoeffding worse.

© Stanley Chan 2020. All Rights Reserved.

Rewriting the Hoeffding Inequality

- Recall the Hoeffding Inequality

$$\mathbb{P}\left\{ |E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon \right\} \le 2Me^{-2\epsilon^2 N}.$$

- This is the same as

$$\mathbb{P}\left\{ |E_{\text{in}}(g) - E_{\text{out}}(g)| \le \epsilon \right\} > 1 - \delta.$$

- Equivalently, we can say: **with probability** $1 - \delta$,

$$E_{\text{in}}(g) - \epsilon \le E_{\text{out}}(g) \le E_{\text{in}}(g) + \epsilon.$$

©Stanley Chan 2020. All Rights Reserved.

What is δ?

- Move around the terms, then we have

$$2Me^{-2\epsilon^2 N} = \delta \quad \Rightarrow \quad \epsilon = \sqrt{\frac{1}{2N}\log\frac{2M}{\delta}}$$

- Plug this result into the previous bound:

$$E_{\text{in}}(g) - \epsilon \leq E_{\text{out}}(g) \leq E_{\text{in}}(g) + \epsilon.$$

- This gives us

$$E_{\text{in}}(g) - \sqrt{\frac{1}{2N}\log\frac{2M}{\delta}} \leq E_{\text{out}}(g) \leq E_{\text{in}}(g) + \sqrt{\frac{1}{2N}\log\frac{2M}{\delta}}.$$

- This is called the **generalization bound**.

©Stanley Chan 2020. All Rights Reserved.

Interpreting the Generalization Bound

$$E_{\text{in}}(g) - \sqrt{\frac{1}{2N}\log\frac{2M}{\delta}} \leq E_{\text{out}}(g) \leq E_{\text{in}}(g) + \sqrt{\frac{1}{2N}\log\frac{2M}{\delta}}.$$

- N: Training sample.
- More is better.
- δ: The probability tolerance level. "Confidence".
- Small δ: You are very conservative. So you need large N to compensate for $\log\frac{1}{\delta}$
- M: Model complexity.
- Large M: You use a very complicated model. So you need large N to compensate for $\log M$

©Stanley Chan 2020. All Rights Reserved.

The Two Sides of the Generalization Bound

- **Upper Limit**

$$E_{\text{in}}(g) - \sqrt{\frac{1}{2N}\log\frac{2M}{\delta}} \leq E_{\text{out}}(g) \leq E_{\text{in}}(g) + \sqrt{\frac{1}{2N}\log\frac{2M}{\delta}}.$$

- $E_{\text{out}}(g)$ cannot be worse than $E_{\text{in}}(g) + \epsilon$.
- Performance guarantee. $E_{\text{in}}(g) + \epsilon$ is the worst you will have. If you can manage this worst case then you are good.
- **Lower Limit**

$$E_{\text{in}}(g) - \sqrt{\frac{1}{2N}\log\frac{2M}{\delta}} \leq E_{\text{out}}(g) \leq E_{\text{in}}(g) + \sqrt{\frac{1}{2N}\log\frac{2M}{\delta}}.$$

- $E_{\text{out}}(g)$ cannot be better than $E_{\text{in}}(g) - \epsilon$.
- Intrinsic limit of your dataset (which controls N), model complexity (which controls M), and how much you want (which determines δ)

©Stanley Chan 2020. All Rights Reserved.

Outline

- Lecture 25 Generalization
- Lecture 26 Growth Function
- Lecture 27 VC Dimension

Today's Lecture:

- M Hypothesis
  - PAC framework
  - Guarantee and Possibility
  - The M factor
- Generalization Bound
  - $\mathcal{H}$
  - f
  - Lower and upper limits
- Handling M hypothesis
  - A preview

©Stanley Chan 2020. All Rights Reserved.

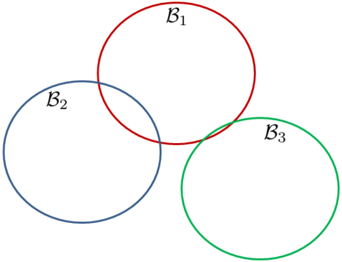
Overcoming the M Factor

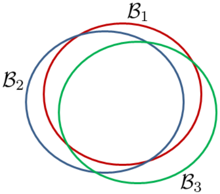
- The $\mathcal{B}$ad events $\mathcal{B}_m$ are

$$\mathcal{B}_m = \{|E_{\text{in}}(h_m) - E_{\text{out}}(h_m)| > \epsilon\}$$

- The factor M is here because of the Union bound:

$$\mathbb{P}[\mathcal{B}_1 \textbf{ or } \ldots \textbf{ or } \mathcal{B}_M] \leq \mathbb{P}[\mathcal{B}_1] + \ldots + \mathbb{P}[\mathcal{B}_M].$$



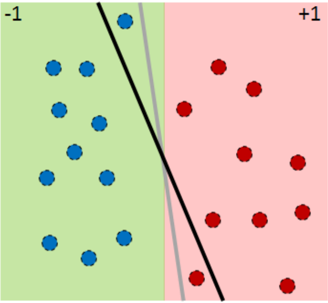


© Stanley Chan 2020. All Rights Reserved.

Counting the Overlapping Area

- ΔE_{out} = change in the $+1$ and -1 area
- Example below: Change a little bit
- ΔE_{in} = change in labels of the training samples
- Example below: Change a little bit, too
- So we should expect the probabilities

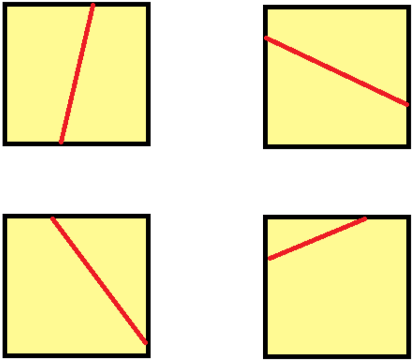
$$\mathbb{P}[|E_{\text{in}}(h_1) - E_{\text{out}}(h_1)| > \epsilon] \approx \mathbb{P}[|E_{\text{in}}(h_2) - E_{\text{out}}(h_2)| > \epsilon].$$



©Stanley Chan 2020. All Rights Reserved.

Looking at the Training Samples Only

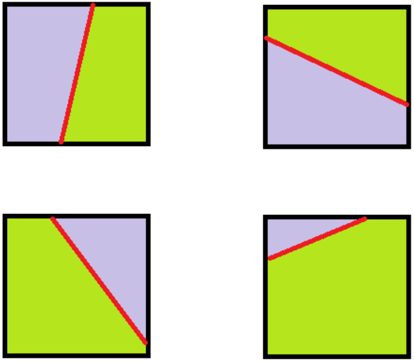
- Here is a our goal: Find something to replace M.
- But M is big because the whole input space is big.
- Let us look at the input space.



©Stanley Chan 2020. All Rights Reserved.

Looking at the Training Samples Only

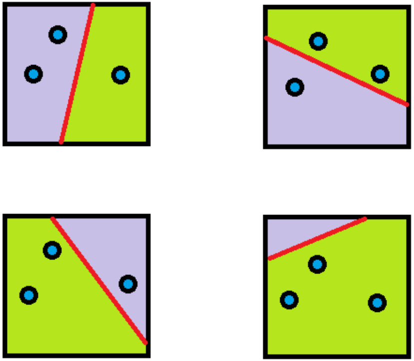
- If you move the hypothesis a little, you get a different partition
- Literally there are infinitely many hypotheses
- This is M



©Stanley Chan 2020. All Rights Reserved.

Looking at the Training Samples Only

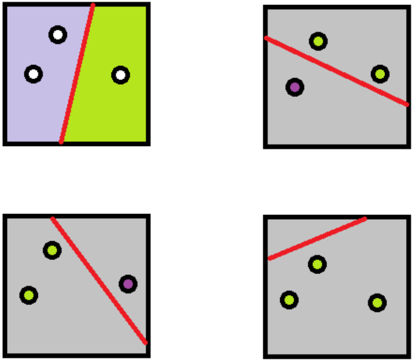
- Here is a our goal: Find something to replace M
- But M is big because the whole input space is big
- Can we restrict ourselves to just the training sets?



©Stanley Chan 2020. All Rights Reserved.

Looking at the Training Samples Only

- The idea is: Just look at the training samples
- Put a mask on your dataset
- Don't care until a training sample flips its sign



©Stanley Chan 2020. All Rights Reserved.

Reading List

- Learning from Data, chapter 2
- Martin Wainwright, High Dimensional Statistics, Cambridge University Press 2019. (Chapter 2)
- CMU Note https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture28-pac.pdf
- Stanford Note http://cs229.stanford.edu/notes/cs229-notes4.pdf

©Stanley Chan 2020. All Rights Reserved.