ECE595 / STAT598: Machine Learning I

Lecture 27.1: VC Dimension - From Dichotomy to Shattering

Spring 2020

Stanley Chan

School of Electrical and Computer Engineering
Purdue University

Outline

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

Today's Lecture:

- From Dichotomy to Shattering
  - Review of dichotomy
  - The Concept of Shattering
  - VC Dimension
- Example of VC Dimension
  - Rectangle Classifier
  - Perceptron Algorithm
  - Two Cases

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**.

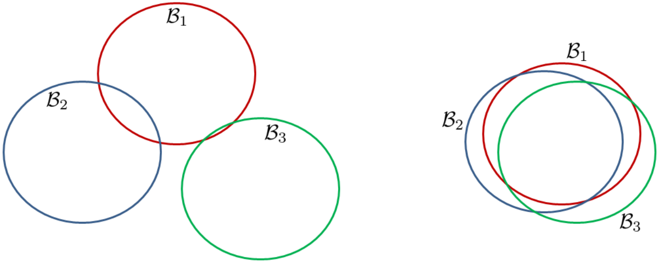
Overcoming the M Factor

- 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:

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

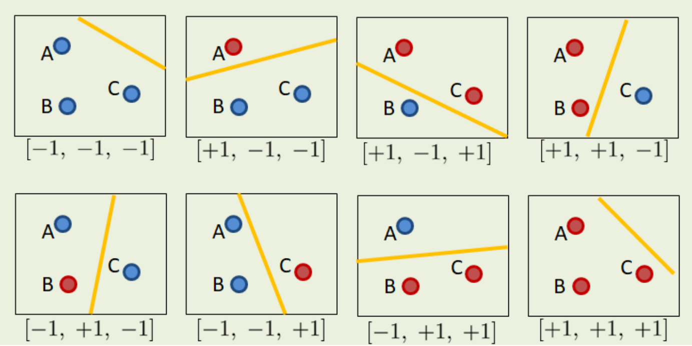


Dichotomy

Definition

Let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathcal{X}$. The **dichotomies** generated by $\mathcal{H}$ on these points are

$$\mathcal{H}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) = \{(h(\boldsymbol{x}_1), \ldots, h(\boldsymbol{x}_N)) \mid h \in \mathcal{H}\}.$$

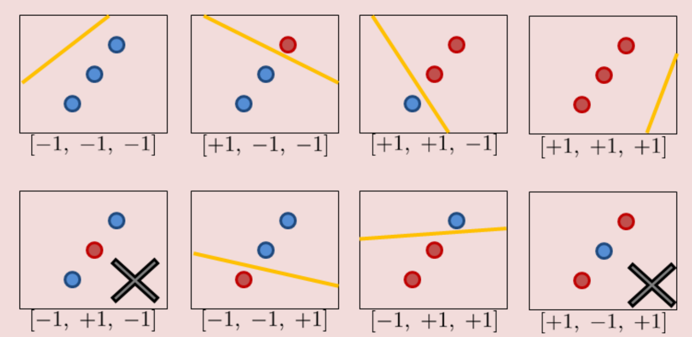


Dichotomy

Definition

Let $\mathbf{x}_1, \ldots, \mathbf{x}_N \in \mathcal{X}$. The **dichotomies** generated by $\mathcal{H}$ on these points are

$$\mathcal{H}(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \{(h(\mathbf{x}_1), \ldots, h(\mathbf{x}_N)) \mid h \in \mathcal{H}\}.$$



Candidate to Replace M

- So here is our candidate replacement for M.
- Define **Growth Function**

$$m_{\mathcal{H}}(N) = \max_{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N \in \mathcal{X}} |\mathcal{H}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)|$$

- You give me a hypothesis set $\mathcal{H}$
- You tell me there are N training samples
- My job: Do whatever I can, by allocating $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$, so that the number of dichotomies is maximized
- Maximum number of dichotomy = the best I can do with your $\mathcal{H}$
- $m_{\mathcal{H}}(N)$: How expressive your hypothesis set $\mathcal{H}$ is
- Large $m_{\mathcal{H}}(N)$ = more expressive $\mathcal{H}$ = more complicated $\mathcal{H}$
- $m_{\mathcal{H}}(N)$ only depends on $\mathcal{H}$ and N
- Doesn't depend on the learning algorithm $\mathcal{A}$
- Doesn't depend on the distribution $p(\boldsymbol{x})$ (because I'm giving you the max.)

Summary of the Examples

- $\mathcal{H}$ is positive ray:

$$m_{\mathcal{H}}(N) = N + 1$$

- $\mathcal{H}$ is positive interval:

$$m_{\mathcal{H}}(N) = \binom{N+1}{2} + 1 = \frac{N^2}{2} + \frac{N}{2} + 1$$

- $\mathcal{H}$ is convex set:

$$m_{\mathcal{H}}(N) = 2^N$$

- So if we can replace M by $m_{\mathcal{H}}(N)$
- And if $m_{\mathcal{H}}(N)$ is a polynomial
- Then we are good.

Shatter

Definition

If a hypothesis set $\mathcal{H}$ is able to generate all 2^N dichotomies, then we say that $\mathcal{H}$ **shatter** $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$.

- $\mathcal{H}$ = hyperplane returned by a perceptron algorithm in 2D.
- If $N = 3$, then $\mathcal{H}$ can shatter
- Because we can achieve $2^3 = 8$ dichotomies
- If $N = 4$, then $\mathcal{H}$ cannot shatter
- Because we can only achieve 14 dichotomies

VC Dimension

Definition (VC Dimension)

The Vapnik-Chervonenkis dimension of a hypothesis set $\mathcal{H}$, denoted by d_{VC}, is the largest value of N for which $\mathcal{H}$ can shatter all N training samples.

- You give me a hypothesis set $\mathcal{H}$, e.g., linear model
- You tell me the number of training samples N
- Start with a small N
- I will be able to shatter for a while, until I hit a bump
- E.g., linear in 2D: $N = 3$ is okay, but $N = 4$ is not okay
- So I find the **largest** N such that $\mathcal{H}$ can shatter N training samples
- E.g., linear in 2D: $d_{\mathrm{VC}} = 3$
- If $\mathcal{H}$ is complex, then expect large d_{VC}
- Does not depend on $p(\boldsymbol{x})$, $\mathcal{A}$ and f