ECE595 / STAT598: Machine Learning I

Lecture 26.1: Growth Function - Overcoming the *M* Factor

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:

- Overcoming the M Factor
  - Decisions based on Training Samples
  - Dichotomy
- Examples of $m_{\mathcal{H}}(N)$
  - Finite 2D Set
  - Positive ray
  - Interval
  - Convex set

Probably Approximately Correct

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

$$\mathbb{P}\Big[\, |E_{\text{in}}(h) - E_{\text{out}}(h)| \le \epsilon \Big] \ge 1 - \delta$$

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

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

- 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**.

The Factor "*M*"

- Testing

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

- Training

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

- 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.

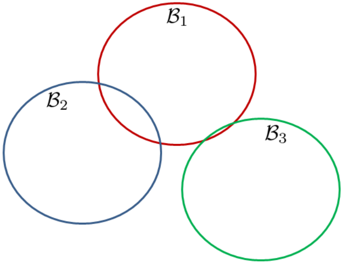
Overcoming the M Factor

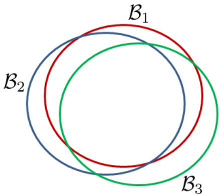
- The $\mathcal{B}$ad events $\mathcal{B}_m$ are

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

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

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

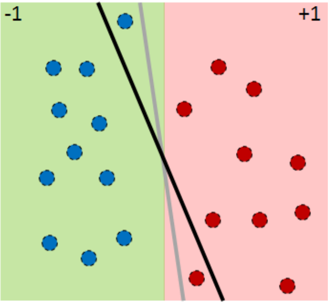




Counting the Overlapping Area

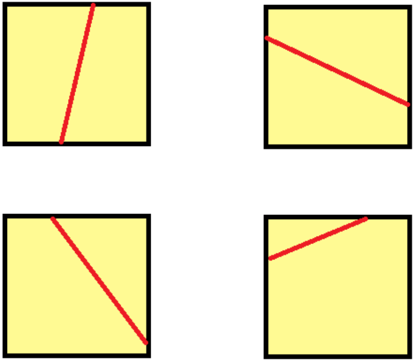
- ΔE_{out} = change in the +1 and -1 area
- Example below: Change a little bit
- ΔE_{in} = change in labels of the training samples
- Example below: Change a little bit, too
- So we should expect the probabilities

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



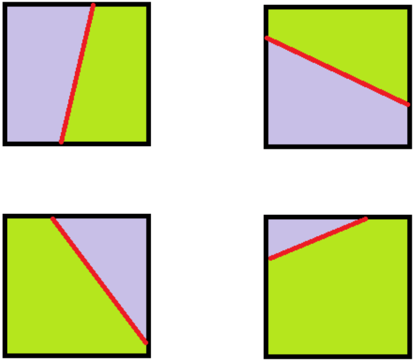
Looking at the Training Samples Only

- 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.



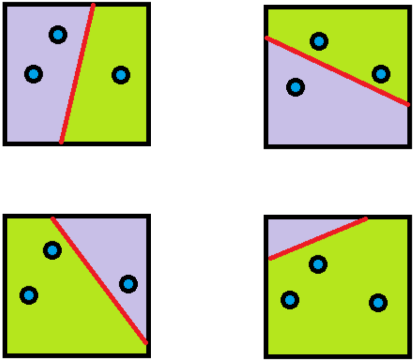
Looking at the Training Samples Only

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



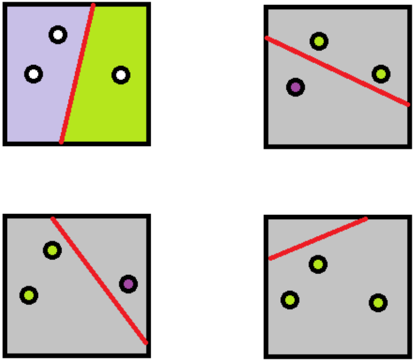
Looking at the Training Samples Only

- 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?



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

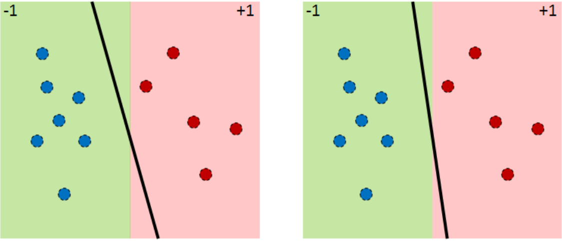


Dichotomies

- We need a new name: dichotomy.
- Dichotomy = mini-hypothesis.

Hypothesis	Dichotomy
$h : \mathcal{X} \rightarrow \{+1, -1\}$	$h : \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\} \rightarrow \{+1, -1\}$
for all population samples	for training samples only
number can be infinite	number is at most 2^N

- Different hypothesis, same dichotomy.

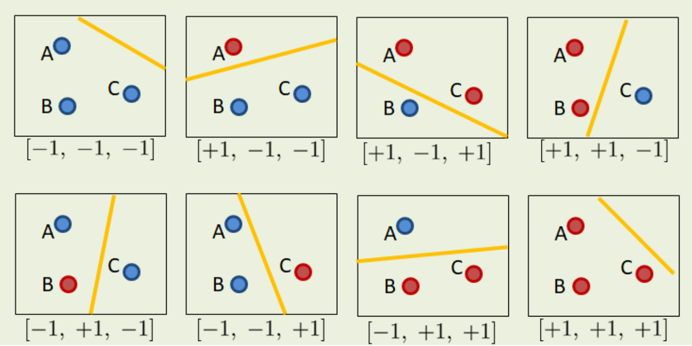


Dichotomy

Definition

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

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

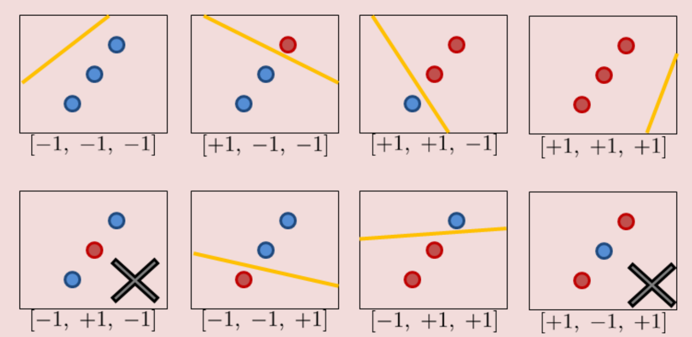


Dichotomy

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.)