Lecture 5: Training versus Testing

Hsuan-Tien Lin (林軒田)
htlin@csie.ntu.edu.tw

Department of Computer Science
& Information Engineering
National Taiwan University
(國立台灣大學資訊工程系)
When Can Machines Learn?

Lecture 4: Feasibility of Learning

Learning is PAC-possible if enough statistical data and finite $|\mathcal{H}|$.

Why Can Machines Learn?

Lecture 5: Training versus Testing

- Recap and Preview
- Effective Number of Lines
- Effective Number of Hypotheses
- Break Point

How Can Machines Learn?

How Can Machines Learn Better?
Recap: the ‘Statistical’ Learning Flow

if $|\mathcal{H}| = M$ finite, $N$ large enough, for whatever $g$ picked by $\mathcal{A}$, $E_{out}(g) \approx E_{in}(g)$
Recap: the ‘Statistical’ Learning Flow

-if $|\mathcal{H}| = M$ finite, $N$ large enough,
- for whatever $g$ picked by $\mathcal{A}$, $E_{\text{out}}(g) \approx E_{\text{in}}(g)$
-if $\mathcal{A}$ finds one $g$ with $E_{\text{in}}(g) \approx 0$,
PAC guarantee for $E_{\text{out}}(g) \approx 0 \implies$ learning possible :-)

unknown target function $f: \mathcal{X} \to \mathcal{Y}$
(ideal credit approval formula)

training examples $\mathcal{D}: (x_1, y_1), \ldots, (x_N, y_N)$
(historical records in bank)

unknown $P$ on $\mathcal{X}$

learning algorithm $\mathcal{A}$

final hypothesis $g \approx f$
('learned' formula to be used)

hypothesis set $\mathcal{H}$
(set of candidate formula)
### Recap: the ‘Statistical’ Learning Flow

**if** $|\mathcal{H}| = M$ finite, $N$ large enough,

for whatever $g$ picked by $\mathcal{A}$, $E_{out}(g) \approx E_{in}(g)$

**if** $\mathcal{A}$ finds one $g$ with $E_{in}(g) \approx 0$,

PAC guarantee for $E_{out}(g) \approx 0 \implies$ learning possible :-)

---

**unknown target function** $f : \mathcal{X} \to \mathcal{Y}$

(ideal credit approval formula)

**unknown** $P$ on $\mathcal{X}$

**training examples** $\mathcal{D} : (x_1, y_1), \ldots, (x_N, y_N)$

(historical records in bank)

**learning algorithm** $\mathcal{A}$

**final hypothesis** $g \approx f$

(‘learned’ formula to be used)

**hypothesis set** $\mathcal{H}$

(set of candidate formula)

$E_{in}(g) \approx 0$

train
Recap: the ‘Statistical’ Learning Flow

if $|\mathcal{H}| = M$ finite, $N$ large enough,
for whatever $g$ picked by $\mathcal{A}$, $E_{out}(g) \approx E_{in}(g)$
if $\mathcal{A}$ finds one $g$ with $E_{in}(g) \approx 0$,
PAC guarantee for $E_{out}(g) \approx 0 \implies$ learning possible :-)

unknown target function $f : \mathcal{X} \to \mathcal{Y}$
(ideal credit approval formula)

unknown $P$ on $\mathcal{X}$

training examples $\mathcal{D} : (x_1, y_1), \cdots, (x_N, y_N)$
(historical records in bank)

learning algorithm $\mathcal{A}$

final hypothesis $g \approx f$
(‘learned’ formula to be used)

hypothesis set $\mathcal{H}$
(set of candidate formula)

$E_{out}(g) \approx E_{in}(g) \approx 0$
(test train)
Two Central Questions

\[ g \approx f \iff E_{\text{out}}(g) \approx 0 \]

lecture 1
Two Central Questions

\[ g \approx f \iff E_{\text{out}}(g) \approx 0 \]

- lecture 1

\[ E_{\text{in}}(g) \approx 0 \]

- lecture 2

Can we make sure that \( E_{\text{out}}(g) \) is close enough to \( E_{\text{in}}(g) \)?

Can we make \( E_{\text{in}}(g) \) small enough?

What role does \( |H| \) play for the two questions?
Two Central Questions

for batch & supervised binary classification,

\[ g \approx f \iff E_{out}(g) \approx 0 \]

lecture 3

\[ E_{in}(g) \approx 0 \]

lecture 2

lecture 1

\[ \iff \]

\[ \iff \]
Two Central Questions

for batch & supervised binary classification,

\[ g \approx f \iff E_{out}(g) \approx 0 \]

achieved through

\[ E_{out}(g) \approx E_{in}(g) \quad \text{and} \quad E_{in}(g) \approx 0 \]
Two Central Questions

for batch & supervised binary classification, $g \approx f \iff E_{out}(g) \approx 0$

achieved through $E_{out}(g) \approx E_{in}(g)$ and $E_{in}(g) \approx 0$

learning split to two central questions:

1. can we make sure that $E_{out}(g)$ is close enough to $E_{in}(g)$?
Two Central Questions

for batch & supervised binary classification, \( g \approx f \iff E_{\text{out}}(g) \approx 0 \)

achieved through \( E_{\text{out}}(g) \approx E_{\text{in}}(g) \) and \( E_{\text{in}}(g) \approx 0 \)

learning split to two central questions:

1. can we make sure that \( E_{\text{out}}(g) \) is close enough to \( E_{\text{in}}(g) \)?
2. can we make \( E_{\text{in}}(g) \) small enough?
Two Central Questions

for batch & supervised binary classification, \( g \approx f \iff E_{\text{out}}(g) \approx 0 \)

achieved through \( E_{\text{out}}(g) \approx E_{\text{in}}(g) \) and \( E_{\text{in}}(g) \approx 0 \)

learning split to two central questions:

1. can we make sure that \( E_{\text{out}}(g) \) is close enough to \( E_{\text{in}}(g) \)?
2. can we make \( E_{\text{in}}(g) \) small enough?

what role does \( M \) play for the two questions?
Trade-off on $M$

1. Can we make sure that $E_{\text{out}}(g)$ is close enough to $E_{\text{in}}(g)$?
2. Can we make $E_{\text{in}}(g)$ small enough?

- **Small $M$**
- **Large $M$**
Trade-off on $M$

1. Can we make sure that $E_{\text{out}}(g)$ is close enough to $E_{\text{in}}(g)$?

2. Can we make $E_{\text{in}}(g)$ small enough?

### Small $M$

1. Yes!,
   \[ \mathbb{P}[\text{BAD}] \leq 2 \cdot M \cdot \exp(\ldots) \]

### Large $M$

No!, too few choices

Yes!, many choices using the right $M$ (or $H$) is important

$M = \infty$ doomed?
### Trade-off on $M$

1. Can we make sure that $E_{out}(g)$ is close enough to $E_{in}(g)$?
2. Can we make $E_{in}(g)$ small enough?

#### small $M$

1. Yes!, $\Pr[\text{BAD}] \leq 2 \cdot M \cdot \exp(\ldots)$
2. No!, too few choices

#### large $M$

---

Hsuan-Tien Lin  (NTU CSIE)
Trade-off on $M$

1. Can we make sure that $E_{out}(g)$ is close enough to $E_{in}(g)$?
2. Can we make $E_{in}(g)$ small enough?

**small $M$**

1. Yes!, $\Pr[\text{BAD}] \leq 2 \cdot M \cdot \exp(...)$
2. No!, too few choices

**large $M$**

2. Yes!, many choices
Trade-off on $M$

1. can we make sure that $E_{out}(g)$ is close enough to $E_{in}(g)$?
2. can we make $E_{in}(g)$ small enough?

**small $M$**

1. Yes!, $\mathbb{P}[BAD] \leq 2 \cdot M \cdot \exp(...)$
2. No!, too few choices

**large $M$**

1. No!, $\mathbb{P}[BAD] \leq 2 \cdot M \cdot \exp(...)$
2. Yes!, many choices
Trade-off on $M$

1. Can we make sure that $E_{out}(g)$ is close enough to $E_{in}(g)$?
2. Can we make $E_{in}(g)$ small enough?

**Small $M$**
1. Yes!, $\mathbb{P}[\text{BAD}] \leq 2 \cdot M \cdot \exp(\ldots)$
2. No!, too few choices

**Large $M$**
1. No!, $\mathbb{P}[\text{BAD}] \leq 2 \cdot M \cdot \exp(\ldots)$
2. Yes!, many choices

Using the right $M$ (or $\mathcal{H}$) is important

$M = \infty$ doomed?


\[ \mathbb{P} \left[ \left| E_{in}(g) - E_{out}(g) \right| > \epsilon \right] \leq 2 \cdot M \cdot \exp \left( -2\epsilon^2 N \right) \]
Training versus Testing

Recap and Preview

Preview

**Known**

\[
\mathbb{P} \left[ \left| E_{\text{in}}(g) - E_{\text{out}}(g) \right| > \epsilon \right] \leq 2 \cdot M \cdot \exp \left( -2\epsilon^2 N \right)
\]

**Todo**

- establish **a finite quantity** that replaces \( M \)

\[
\mathbb{P} \left[ \left| E_{\text{in}}(g) - E_{\text{out}}(g) \right| > \epsilon \right] \stackrel{?}{=} 2 \cdot m_{\mathcal{H}} \cdot \exp \left( -2\epsilon^2 N \right)
\]
Training versus Testing

Recap and Preview

Preview

Known

$$\mathbb{P} \left[ |E_{in}(g) - E_{out}(g)| > \epsilon \right] \leq 2 \cdot M \cdot \exp \left( -2\epsilon^2 N \right)$$

Todo

- establish a finite quantity that replaces $M$

$$\mathbb{P} \left[ |E_{in}(g) - E_{out}(g)| > \epsilon \right] \leq 2 \cdot m_H \cdot \exp \left( -2\epsilon^2 N \right)$$

- justify the feasibility of learning for infinite $M$
Training versus Testing

Recap and Preview

Preview

Known

\[
\mathbb{P} \left[ \left| E_{\text{in}}(g) - E_{\text{out}}(g) \right| > \epsilon \right] \leq 2 \cdot M \cdot \exp \left( -2\epsilon^2 N \right)
\]

Todo

- establish a finite quantity that replaces \( M \)

\[
\mathbb{P} \left[ \left| E_{\text{in}}(g) - E_{\text{out}}(g) \right| > \epsilon \right] \leq 2 \cdot m_{\mathcal{H}} \cdot \exp \left( -2\epsilon^2 N \right)
\]

- justify the feasibility of learning for infinite \( M \)
- study \( m_{\mathcal{H}} \) to understand its trade-off for ‘right’ \( \mathcal{H} \), just like \( M \)
Training versus Testing

Recap and Preview

Preview

**Known**

\[ \mathbb{P} \left[ \left| E_{\text{in}}(g) - E_{\text{out}}(g) \right| > \epsilon \right] \leq 2 \cdot M \cdot \exp \left( -2\epsilon^2 N \right) \]

**Todo**

- establish a finite quantity that replaces \( M \)

\[ \mathbb{P} \left[ \left| E_{\text{in}}(g) - E_{\text{out}}(g) \right| > \epsilon \right] \leq 2 \cdot m_{\mathcal{H}} \cdot \exp \left( -2\epsilon^2 N \right) \]
- justify the feasibility of learning for infinite \( M \)
- study \( m_{\mathcal{H}} \) to understand its trade-off for ‘right’ \( \mathcal{H} \), just like \( M \)

mysterious PLA to be fully resolved

after 3 more lectures :-}
Data size: how large do we need?

One way to use the inequality

\[ \mathbb{P} \left[ \left| E_{\text{in}}(g) - E_{\text{out}}(g) \right| > \epsilon \right] \leq 2 \cdot M \cdot \exp \left( \frac{-2\epsilon^2 N}{\delta} \right) \]

is to pick a tolerable difference \( \epsilon \) as well as a tolerable BAD probability \( \delta \), and then gather data with size \( N \) large enough to achieve those tolerance criteria. Let \( \epsilon = 0.1 \), \( \delta = 0.05 \), and \( M = 100 \). What is the data size needed?

1. 215  
2. 415  
3. 615  
4. 815
Data size: how large do we need?

One way to use the inequality

$$\mathbb{P} \left[ \left| E_{\text{in}}(g) - E_{\text{out}}(g) \right| > \epsilon \right] \leq 2 \cdot M \cdot \exp \left( -2\epsilon^2 N \right)$$

is to pick a tolerable difference $\epsilon$ as well as a tolerable BAD probability $\delta$, and then gather data with size $(N)$ large enough to achieve those tolerance criteria. Let $\epsilon = 0.1$, $\delta = 0.05$, and $M = 100$. What is the data size needed?

1. 215
2. 415
3. 615
4. 815

Reference Answer: 2

We can simply express $N$ as a function of those ‘known’ variables. Then, the needed $N = \frac{1}{2\epsilon^2} \ln \frac{2M}{\delta}$. 
Where Did \( M \) Come From?

\[
\mathbb{P} \left[ |E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon \right] \leq 2 \cdot M \cdot \exp \left(-2\epsilon^2 N\right)
\]

- **BAD events** \( \mathcal{B}_m \): \(|E_{\text{in}}(h_m) - E_{\text{out}}(h_m)| > \epsilon\)
Where Did $M$ Come From?

$$\Pr \left[ |E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon \right] \leq 2 \cdot M \cdot \exp \left( -2\epsilon^2 N \right)$$

- **BAD events** $\mathcal{B}_m$: $|E_{\text{in}}(h_m) - E_{\text{out}}(h_m)| > \epsilon$
- to give $A$ freedom of choice: bound $\Pr[\mathcal{B}_1 \text{ or } \mathcal{B}_2 \text{ or } \ldots \mathcal{B}_M]$
Where Did $M$ Come From?

$$
\Pr \left[ |E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon \right] \leq 2 \cdot M \cdot \exp \left( -2\epsilon^2 N \right)
$$

- **BAD events** $B_m$: $|E_{\text{in}}(h_m) - E_{\text{out}}(h_m)| > \epsilon$
- to give $A$ freedom of choice: bound $\Pr[B_1 \text{ or } B_2 \text{ or } \ldots B_M]$  
- worst case: all $B_m$ non-overlapping

$$
\Pr[B_1 \text{ or } B_2 \text{ or } \ldots B_M] \leq \Pr[B_1] + \Pr[B_2] + \ldots + \Pr[B_M]
$$

union bound
Where Did $M$ Come From?

$$
\mathbb{P} \left[ |E_{in}(g) - E_{out}(g)| > \epsilon \right] \leq 2 \cdot M \cdot \exp \left( -2\epsilon^2 N \right)
$$

- **BAD events** $B_m$: $|E_{in}(h_m) - E_{out}(h_m)| > \epsilon$
- to give $A$ freedom of choice: bound $\mathbb{P}[B_1 \text{ or } B_2 \text{ or } \ldots B_M]$
- worst case: all $B_m$ non-overlapping

$$
\mathbb{P}[B_1 \text{ or } B_2 \text{ or } \ldots B_M] \leq \mathbb{P}[B_1] + \mathbb{P}[B_2] + \ldots + \mathbb{P}[B_M]
$$

**union bound**

where did **uniform bound fail**
to consider for $M = \infty$?
Where Did Uniform Bound Fail?

union bound $\mathbb{P}[B_1] + \mathbb{P}[B_2] + \ldots + \mathbb{P}[B_M]$
Where Did Uniform Bound Fail?

union bound $\mathbb{P}[\mathcal{B}_1] + \mathbb{P}[\mathcal{B}_2] + \ldots + \mathbb{P}[\mathcal{B}_M]$

- **BAD events $\mathcal{B}_m$:** $|E_{\text{in}}(h_m) - E_{\text{out}}(h_m)| > \epsilon$

  overlapping for similar hypotheses $h_1 \approx h_2$
Where Did Uniform Bound Fail?

union bound \( \mathbb{P}[\mathcal{B}_1] + \mathbb{P}[\mathcal{B}_2] + \ldots + \mathbb{P}[\mathcal{B}_M] \)

- **BAD events** \( \mathcal{B}_m \): \( |E_{\text{in}}(h_m) - E_{\text{out}}(h_m)| > \epsilon \)

  overlapping for similar hypotheses \( h_1 \approx h_2 \)

- why?
  1. \( E_{\text{out}}(h_1) \approx E_{\text{out}}(h_2) \)
  2. for most \( \mathcal{D} \), \( E_{\text{in}}(h_1) = E_{\text{in}}(h_2) \)
Where Did Uniform Bound Fail?

union bound \( \mathbb{P}[B_1] + \mathbb{P}[B_2] + \ldots + \mathbb{P}[B_M] \)

- **BAD events** \( B_m \): \( |E_{\text{in}}(h_m) - E_{\text{out}}(h_m)| > \epsilon \)
  - overlapping for similar hypotheses \( h_1 \approx h_2 \)
  - why? ① \( E_{\text{out}}(h_1) \approx E_{\text{out}}(h_2) \)
    ② for most \( \mathcal{D} \), \( E_{\text{in}}(h_1) = E_{\text{in}}(h_2) \)
- union bound **over-estimating**
Where Did Uniform Bound Fail?

union bound $\mathbb{P}[\mathcal{B}_1] + \mathbb{P}[\mathcal{B}_2] + \ldots + \mathbb{P}[\mathcal{B}_M]$

- **BAD events** $\mathcal{B}_m$: $|E_{\text{in}}(h_m) - E_{\text{out}}(h_m)| > \epsilon$
  - overlapping for similar hypotheses $h_1 \approx h_2$
- why?
  1. $E_{\text{out}}(h_1) \approx E_{\text{out}}(h_2)$
  2. for most $\mathcal{D}$, $E_{\text{in}}(h_1) = E_{\text{in}}(h_2)$
- union bound over-estimating

To account for overlap, can we group similar hypotheses by kind?
How Many Lines Are There? (1/2)

\[ \mathcal{H} = \{ \text{all lines in } \mathbb{R}^2 \} \]

- how many lines? \(\infty\)
How Many Lines Are There? (1/2)

- how many lines? $\infty$
- how many kinds of lines if viewed from one input vector $x_1$?

$\mathcal{H} = \{\text{all lines in } \mathbb{R}^2\}$
How Many Lines Are There? (1/2)

\[ \mathcal{H} = \{ \text{all lines in } \mathbb{R}^2 \} \]

- how many lines? \( \infty \)
- how many \textbf{kinds of} lines if viewed from one input vector \( x_1 \)?

2 kinds: \( h_1 \)-like(\( x_1 \)) = \( \circ \) or \( h_2 \)-like(\( x_1 \)) = \( \times \)
How Many Lines Are There? (2/2)

\[ \mathcal{H} = \{ \text{all lines in } \mathbb{R}^2 \} \]

- how many kinds of lines if viewed from two inputs \( x_1, x_2 \)?
How Many Lines Are There? (2/2)

\[ \mathcal{H} = \{ \text{all lines in } \mathbb{R}^2 \} \]

- how many kinds of lines if viewed from two inputs \( x_1, x_2 \)?
How Many Lines Are There? (2/2)

\[ \mathcal{H} = \{ \text{all lines in } \mathbb{R}^2 \} \]

- how many kinds of lines if viewed from two inputs \( x_1, x_2 \)?

4:

One input: 2; two inputs: 4; three inputs?
How Many Kinds of Lines for Three Inputs? (1/2)

\[ \mathcal{H} = \{ \text{all lines in } \mathbb{R}^2 \} \]

for three inputs \( \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3 \)
How Many Kinds of Lines for Three Inputs? (1/2)

\[ H = \{ \text{all lines in } \mathbb{R}^2 \} \]

for three inputs \( \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3 \)
How Many Kinds of Lines for Three Inputs? (1/2)

\[\mathcal{H} = \left\{ \text{all lines in } \mathbb{R}^2 \right\}\]

for three inputs \(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\)

always \(8\) for three inputs?
How Many Kinds of Lines for Three Inputs? (2/2)

\[ \mathcal{H} = \{ \text{all lines in } \mathbb{R}^2 \} \]

for another three inputs \( x_1, x_2, x_3 \)

6:
How Many Kinds of Lines for Three Inputs? (2/2)

\[ \mathcal{H} = \{ \text{all lines in } \mathbb{R}^2 \} \]

for another three inputs \( x_1, x_2, x_3 \)

\[ \text{‘fewer than 8’ when degenerate (e.g. collinear or same inputs)} \]

6:
How Many Kinds of Lines for Three Inputs? (2/2)

\[ \mathcal{H} = \{ \text{all lines in } \mathbb{R}^2 \} \]

for another three inputs \( x_1, x_2, x_3 \)

\[ x_1 \cdot x_2 \cdot x_3 \]

‘fewer than 8’ when degenerate (e.g. collinear or same inputs)
Effective Number of Lines

How Many Kinds of Lines for Four Inputs?

\[ \mathcal{H} = \{ \text{all lines in } \mathbb{R}^2 \} \]

for four inputs \( x_1, x_2, x_3, x_4 \)

\[ x_1 \bullet \quad \bullet x_2 \quad \bullet x_3 \quad \bullet x_4 \]
How Many Kinds of Lines for Four Inputs?

\[ \mathcal{H} = \left\{ \text{all lines in } \mathbb{R}^2 \right\} \]

for four inputs \( x_1, x_2, x_3, x_4 \)

\[ x_1 \cdot x_2 \cdot x_3 \cdot x_4 \]

\( 14 \times 2 \)
How Many Kinds of Lines for Four Inputs?

\[ \mathcal{H} = \{ \text{all lines in } \mathbb{R}^2 \} \]

For four inputs \( x_1, x_2, x_3, x_4 \):

For any four inputs, at most 14

\[ 14: 2 \times \]

\[ \bigtimes \]
Effective Number of Lines

maximum kinds of lines with respect to $N$ inputs $x_1, x_2, \cdots, x_N$

$\iff$ effective number of lines
Effective Number of Lines

maximum kinds of lines with respect to $N$ inputs $x_1, x_2, \cdots, x_N$

$\iff$ effective number of lines

<table>
<thead>
<tr>
<th>$N$</th>
<th>effective($N$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>14</td>
</tr>
</tbody>
</table>

$< 2^N$ if effective($N$) can replace $M$ and $2$ effective($N$) $\ll 2^N$ learning possible with infinite lines :-)}
Training versus Testing

Effective Number of Lines

maximum kinds of lines with respect to \( N \) inputs \( x_1, x_2, \cdots, x_N \)  
\( \iff \) effective number of lines

- must be \( \leq 2^N \) (why?)

<table>
<thead>
<tr>
<th>( N )</th>
<th>effective(( N ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>14 ( &lt; 2^N )</td>
</tr>
</tbody>
</table>
Effective Number of Lines

maximum kinds of lines with respect to $N$ inputs $x_1, x_2, \cdots, x_N$

$\iff$ effective number of lines

- must be $\leq 2^N$ (why?)
- finite ‘grouping’ of infinitely-many lines $\in \mathcal{H}$

<table>
<thead>
<tr>
<th>$N$</th>
<th>effective($N$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>14 $&lt; 2^N$</td>
</tr>
</tbody>
</table>
Effective Number of Lines

Maximum kinds of lines with respect to \( N \) inputs \( x_1, x_2, \cdots, x_N \)

\[ \iff \text{effective number of lines} \]

- must be \( \leq 2^N \) (why?)
- finite ‘grouping’ of infinitely-many lines \( \in \mathcal{H} \)
- wish:

\[
P \left[ \left| E_{\text{in}}(g) - E_{\text{out}}(g) \right| > \epsilon \right] \leq 2 \cdot \text{effective}(N) \cdot \exp \left(-2\epsilon^2 N \right)
\]

### Lines in 2D

<table>
<thead>
<tr>
<th>( N )</th>
<th>effective ((N))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>14 (&lt; 2^N)</td>
</tr>
</tbody>
</table>
Effective Number of Lines

- maximum kinds of lines with respect to $N$ inputs $x_1, x_2, \cdots, x_N$
  $\iff$ effective number of lines

- must be $\leq 2^N$ (why?)
- finite ‘grouping’ of infinitely-many lines $\in \mathcal{H}$
- wish:

$$\mathbb{P} \left[ |E_{in}(g) - E_{out}(g)| > \epsilon \right] \leq 2 \cdot \text{effective}(N) \cdot \exp \left( -2\epsilon^2 N \right)$$

<table>
<thead>
<tr>
<th>$N$</th>
<th>effective($N$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>14 $&lt; 2^4$</td>
</tr>
</tbody>
</table>

if 1 effective($N$) can replace $M$ and
2 effective($N$) $\ll 2^N$
learning possible with infinite lines :-)

Hsuan-Tien Lin (NTU CSIE)  Machine Learning Foundations 14/27
What is the effective number of lines for five inputs $\in \mathbb{R}^2$?

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>14</td>
<td>16</td>
<td>22</td>
<td>32</td>
</tr>
</tbody>
</table>

Reference Answer: 3

If you put the inputs roughly around a circle, you can then pick any consecutive inputs to be on one side of the line, and the other inputs to be on the other side. The procedure leads to effectively 22 kinds of lines, which is much smaller than $2^5 = 32$. You shall find it difficult to generate more kinds by varying the inputs, and we will give a formal proof in future lectures.
What is the effective number of lines for five inputs $\in \mathbb{R}^2$?

1. 14
2. 16
3. 22
4. 32

Reference Answer: 3

If you put the inputs roughly around a circle, you can then pick any consecutive inputs to be on one side of the line, and the other inputs to be on the other side. The procedure leads to effectively 22 kinds of lines, which is much smaller than $2^5 = 32$. You shall find it difficult to generate more kinds by varying the inputs, and we will give a formal proof in future lectures.
Dichotomies: Mini-hypotheses

\[ \mathcal{H} = \{ \text{hypothesis } h: \mathcal{X} \rightarrow \{\times, \circ\} \} \]
Dichotomies: Mini-hypotheses

\[ \mathcal{H} = \{ \text{hypothesis } h: \mathcal{X} \rightarrow \{\times, \circ\}\} \]

- call

\[ h(x_1, x_2, \ldots, x_N) = (h(x_1), h(x_2), \ldots, h(x_N)) \in \{\times, \circ\}^N \]

a **dichotomy**: hypothesis ‘limited’ to the eyes of \(x_1, x_2, \ldots, x_N\)
Dichotomies: Mini-hypotheses

\[ \mathcal{H} = \{ \text{hypothesis } h: \mathcal{X} \rightarrow \{ \times, \circ \} \} \]

- call

\[ h(x_1, x_2, \ldots, x_N) = (h(x_1), h(x_2), \ldots, h(x_N)) \in \{ \times, \circ \}^N \]

a **dichotomy**: hypothesis ‘limited’ to the eyes of \( x_1, x_2, \ldots, x_N \)

- \( \mathcal{H}(x_1, x_2, \ldots, x_N) \): all dichotomies ‘implemented’ by \( \mathcal{H} \) on \( x_1, x_2, \ldots, x_N \)
Dichotomies: Mini-hypotheses

\[ \mathcal{H} = \{ \text{hypothesis } h: \mathcal{X} \rightarrow \{\times, \circ}\} \]

- call

\[ h(x_1, x_2, \ldots, x_N) = (h(x_1), h(x_2), \ldots, h(x_N)) \in \{\times, \circ\}^N \]

a **dichotomy**: hypothesis ‘limited’ to the eyes of \( x_1, x_2, \ldots, x_N \)

- \( \mathcal{H}(x_1, x_2, \ldots, x_N) \): all dichotomies ‘implemented’ by \( \mathcal{H} \) on \( x_1, x_2, \ldots, x_N \)

<table>
<thead>
<tr>
<th>hypotheses ( \mathcal{H} )</th>
<th>dichotomies ( \mathcal{H}(x_1, x_2, \ldots, x_N) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>e.g. all lines in ( \mathbb{R}^2 )</td>
<td>{ oooo, ooox, ooxo, ... }</td>
</tr>
<tr>
<td>size possibly infinite</td>
<td>upper bounded by ( 2^N )</td>
</tr>
</tbody>
</table>
Dichotomies: Mini-hypotheses

\[ \mathcal{H} = \{ \text{hypothesis } h: \mathcal{X} \rightarrow \{\times, \circ\} \} \]

- call

\[ h(x_1, x_2, \ldots, x_N) = (h(x_1), h(x_2), \ldots, h(x_N)) \in \{\times, \circ\}^N \]

a **dichotomy**: hypothesis ‘limited’ to the eyes of \( x_1, x_2, \ldots, x_N \)

- \( \mathcal{H}(x_1, x_2, \ldots, x_N) \):
  all dichotomies ‘implemented’ by \( \mathcal{H} \) on \( x_1, x_2, \ldots, x_N \)

<table>
<thead>
<tr>
<th></th>
<th>hypotheses ( \mathcal{H} )</th>
<th>dichotomies ( \mathcal{H}(x_1, x_2, \ldots, x_N) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>e.g.</td>
<td>all lines in ( \mathbb{R}^2 )</td>
<td>{○○○○, ○○○×, ○○××, \ldots}</td>
</tr>
<tr>
<td>size</td>
<td>possibly infinite</td>
<td>upper bounded by ( 2^N )</td>
</tr>
</tbody>
</table>

\[ |\mathcal{H}(x_1, x_2, \ldots, x_N)|: \text{candidate for replacing } M \]
Growth Function

- $|\mathcal{H}(x_1, x_2, \ldots, x_N)|$: depend on inputs $(x_1, x_2, \ldots, x_N)$
Growth Function

- \(|\mathcal{H}(x_1, x_2, \ldots, x_N)|\): depend on inputs \((x_1, x_2, \ldots, x_N)\)
- growth function: remove dependence by taking \(\textbf{max of all possible} \ (x_1, x_2, \ldots, x_N)\)

\[
m_{\mathcal{H}}(N) = \max_{x_1, x_2, \ldots, x_N \in \mathcal{X}} |\mathcal{H}(x_1, x_2, \ldots, x_N)|
\]
Growth Function

- $|\mathcal{H}(x_1, x_2, \ldots, x_N)|$: depend on inputs $(x_1, x_2, \ldots, x_N)$
- Growth function: remove dependence by taking $\max$ of all possible $(x_1, x_2, \ldots, x_N)$

$$m_{\mathcal{H}}(N) = \max_{x_1, x_2, \ldots, x_N \in \mathcal{X}} |\mathcal{H}(x_1, x_2, \ldots, x_N)|$$

<table>
<thead>
<tr>
<th>$N$</th>
<th>$m_{\mathcal{H}}(N)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>$\max(\ldots, 6, 8)$</td>
</tr>
<tr>
<td></td>
<td>= 8</td>
</tr>
<tr>
<td>4</td>
<td>14 &lt; $2^N$</td>
</tr>
</tbody>
</table>
Training versus Testing

Effective Number of Hypotheses

Growth Function

- \(|\mathcal{H}(x_1, x_2, \ldots, x_N)|\): depend on inputs \((x_1, x_2, \ldots, x_N)\)
- growth function: remove dependence by taking \(\max\) of all possible \((x_1, x_2, \ldots, x_N)\)

\[
m_{\mathcal{H}}(N) = \max_{x_1, x_2, \ldots, x_N \in \mathcal{X}} |\mathcal{H}(x_1, x_2, \ldots, x_N)|
\]
- finite, upper-bounded by \(2^N\)

<table>
<thead>
<tr>
<th>(N)</th>
<th>(m_{\mathcal{H}}(N))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>(\max(\ldots, 6, 8)) = 8</td>
</tr>
<tr>
<td>4</td>
<td>14 (&lt; 2^N)</td>
</tr>
</tbody>
</table>
Training versus Testing

Effective Number of Hypotheses

Growth Function

- \(|\mathcal{H}(x_1, x_2, \ldots, x_N)|\): depend on inputs \((x_1, x_2, \ldots, x_N)\)
- growth function: remove dependence by taking \textbf{max of all possible} \((x_1, x_2, \ldots, x_N)\)

\[
m_{\mathcal{H}}(N) = \max_{x_1, x_2, \ldots, x_N \in \mathcal{X}} |\mathcal{H}(x_1, x_2, \ldots, x_N)|
\]

- finite, upper-bounded by \(2^N\)

*how to ‘calculate’ the growth function?*

<table>
<thead>
<tr>
<th>(N)</th>
<th>(m_{\mathcal{H}}(N))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>(\max(\ldots, 6, 8)) = 8</td>
</tr>
<tr>
<td>4</td>
<td>14 (&lt; 2^4)</td>
</tr>
</tbody>
</table>
Growth Function for Positive Rays

- \( \mathcal{X} = \mathbb{R} \) (one dimensional)
- \( \mathcal{H} \) contains \( h \), where each \( h(x) = \text{sign}(x - a) \) for threshold \( a \)
Growth Function for Positive Rays

- $\mathcal{X} = \mathbb{R}$ (one dimensional)
- $\mathcal{H}$ contains $h$, where each $h(x) = \text{sign}(x - a)$ for threshold $a$
- ‘positive half’ of 1D perceptrons
• $X = \mathbb{R}$ (one dimensional)
• $\mathcal{H}$ contains $h$, where each $h(x) = \text{sign}(x - a)$ for threshold $a$
• ‘positive half’ of 1D perceptrons

One dichotomy for $a \in$ each spot $(x_n, x_{n+1})$: $m_\mathcal{H}(N) = N + 1$
Growth Function for Positive Rays

- $\mathcal{X} = \mathbb{R}$ (one dimensional)
- $\mathcal{H}$ contains $h$, where each $h(x) = \text{sign}(x - a)$ for threshold $a$
- ‘positive half’ of 1D perceptrons

One dichotomy for $a \in$ each spot $(x_n, x_{n+1})$:

$$m_{\mathcal{H}}(N) = N + 1$$
**Growth Function for Positive Rays**

- \( \mathcal{X} = \mathbb{R} \) (one dimensional)
- \( \mathcal{H} \) contains \( h \), where each \( h(x) = \text{sign}(x - a) \) for threshold \( a \)
- ‘positive half’ of 1D perceptrons

One dichotomy for \( a \in \text{each spot} \ (x_n, x_{n+1}) \):

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

\((N + 1) \ll 2^N \) when \( N \) large!
Training versus Testing

Effective Number of Hypotheses

Growth Function for Positive Intervals

- $\mathcal{X} = \mathbb{R}$ (one dimensional)
- $\mathcal{H}$ contains $h$, where each $h(x) = +1$ iff $x \in [\ell, r)$, $-1$ otherwise
Growth Function for Positive Intervals

\[ h(x) = -1 \quad h(x) = +1 \quad h(x) = -1 \]

- \( \mathcal{X} = \mathbb{R} \) (one dimensional)
- \( \mathcal{H} \) contains \( h \), where each \( h(x) = +1 \) iff \( x \in [\ell, r) \), \(-1\) otherwise

one dichotomy for each ‘interval kind’

\[
m_{\mathcal{H}}(N) = \left( \begin{array}{c} N+1 \\ 2 \end{array} \right) + 1
\]

interval ends in \( N + 1 \) spots

\[
= \frac{1}{2} N^2 + \frac{1}{2} N + 1
\]
**Growth Function for Positive Intervals**

- \( \mathcal{X} = \mathbb{R} \) (one dimensional)
- \( \mathcal{H} \) contains \( h \), where each \( h(x) = +1 \) iff \( x \in [\ell, r) \), \(-1\) otherwise

one dichotomy for each ‘interval kind’

\[
m_{\mathcal{H}}(N) = \begin{pmatrix} N + 1 \\ 2 \end{pmatrix} + 1
\]

interval ends in \( N + 1 \) spots

\[
= \frac{1}{2} N^2 + \frac{1}{2} N + 1
\]
Growth Function for Positive Intervals

\[ h(x) = -1 \]

\[ h(x) = +1 \]

\[ h(x) = -1 \]

- \( \mathcal{X} = \mathbb{R} \) (one dimensional)
- \( \mathcal{H} \) contains \( h \), where each \( h(x) = +1 \) iff \( x \in [\ell, r) \), \(-1\) otherwise

one dichotomy for each ‘interval kind’

\[
m_{\mathcal{H}}(N) = \left( \frac{N+1}{2} \right) + 1
\]

interval ends in \( N + 1 \) spots

\[
= \frac{1}{2} N^2 + \frac{1}{2} N + 1
\]

\[
\left(\frac{1}{2} N^2 + \frac{1}{2} N + 1\right) \ll 2^N \text{ when } N \text{ large!}
\]
Growth Function for Convex Sets (1/2)

- $\mathcal{X} = \mathbb{R}^2$ (two dimensional)
- $\mathcal{H}$ contains $h$, where $h(x) = +1$ iff $x$ in a convex region, $-1$ otherwise
Growth Function for Convex Sets (1/2)

- $\mathcal{X} = \mathbb{R}^2$ (two dimensional)
- $\mathcal{H}$ contains $h$, where $h(x) = +1$ iff $x$ in a convex region, $-1$ otherwise

what is $m_{\mathcal{H}}(N)$?
Growth Function for Convex Sets (2/2)

- one possible set of $N$ inputs: $x_1, x_2, \ldots, x_N$ on a big circle
• one possible set of $N$ inputs: $x_1, x_2, \ldots, x_N$ on a big circle

• every dichotomy can be implemented by $\mathcal{H}$ using a convex region slightly extended from contour of positive inputs

$$m_{\mathcal{H}}(N) = 2^N$$
• one possible set of $N$ inputs: $x_1, x_2, \ldots, x_N$ on a big circle

• every dichotomy can be implemented by $\mathcal{H}$ using a convex region slightly extended from contour of positive inputs

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

• call those $N$ inputs ‘shattered’ by $\mathcal{H}$
• one possible set of $N$ inputs: $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ on a big circle

• every dichotomy can be implemented by $\mathcal{H}$ using a convex region slightly extended from contour of positive inputs

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

• call those $N$ inputs ‘shattered’ by $\mathcal{H}$

$$m_{\mathcal{H}}(N) = 2^N \iff \exists \text{ } N \text{ inputs that can be shattered}$$
Consider positive and negative rays as $\mathcal{H}$, which is equivalent to the perceptron hypothesis set in 1D. The hypothesis set is often called ‘decision stump’ to describe the shape of its hypotheses. What is the growth function $m_{\mathcal{H}}(N)$?

1. $N$
2. $N + 1$
3. $2N$
4. $2^N$
Consider positive and negative rays as $\mathcal{H}$, which is equivalent to the perceptron hypothesis set in 1D. The hypothesis set is often called ‘decision stump’ to describe the shape of its hypotheses. What is the growth function $m_\mathcal{H}(N)$?

- 1. $N$
- 2. $N + 1$
- 3. $2N$
- 4. $2^N$

Reference Answer: 3

Two dichotomies when threshold in each of the $N - 1$ ‘internal’ spots; two dichotomies for the all-○ and all-× cases.
The Four Growth Functions

- positive rays: \( m_{\mathcal{H}}(N) = N + 1 \)
- positive intervals: \( m_{\mathcal{H}}(N) = \frac{1}{2} N^2 + \frac{1}{2} N + 1 \)
- convex sets: \( m_{\mathcal{H}}(N) = 2^N \)
- 2D perceptrons: \( m_{\mathcal{H}}(N) < 2^N \) in some cases
The Four Growth Functions

- positive rays: \( m_{\mathcal{H}}(N) = N + 1 \)
- positive intervals: \( m_{\mathcal{H}}(N) = \frac{1}{2} N^2 + \frac{1}{2} N + 1 \)
- convex sets: \( m_{\mathcal{H}}(N) = 2^N \)
- 2D perceptrons: \( m_{\mathcal{H}}(N) < 2^N \) in some cases

what if \( m_{\mathcal{H}}(N) \) replaces \( M \)?

\[ \mathbb{P} \left[ |E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon \right] \leq 2 \cdot m_{\mathcal{H}}(N) \cdot \exp \left( -2\epsilon^2 N \right) \]

*polynomial: good; exponential: bad*
### The Four Growth Functions

- **positive rays:**
  \[ m_\mathcal{H}(N) = N + 1 \]

- **positive intervals:**
  \[ m_\mathcal{H}(N) = \frac{1}{2}N^2 + \frac{1}{2}N + 1 \]

- **convex sets:**
  \[ m_\mathcal{H}(N) = 2^N \]

- **2D perceptrons:**
  \[ m_\mathcal{H}(N) < 2^N \text{ in some cases} \]

### what if \( m_\mathcal{H}(N) \) replaces \( M \)?

\[
\mathbb{P} \left[ |E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon \right] \leq 2 \cdot m_\mathcal{H}(N) \cdot \exp \left( -2\epsilon^2 N \right)
\]

**polynomial:** good; **exponential:** bad

### for 2D or general perceptrons, \( m_\mathcal{H}(N) \) **polynomial**?
what do we know about 2D perceptrons now?

three inputs: ‘exists’ shatter;
four inputs, ‘for all’ no shatter
Break Point of $\mathcal{H}$

what do we know about 2D perceptrons now?

- three inputs: ‘exists’ shatter;
- four inputs, ‘for all’ no shatter

if no $k$ inputs can be shattered by $\mathcal{H}$, call $k$ a **break point** for $\mathcal{H}$

- $m_{\mathcal{H}}(k) < 2^k$
what do we know about 2D perceptrons now?

three inputs: ‘exists’ shatter;
four inputs, ‘for all’ no shatter

if no $k$ inputs can be shattered by $\mathcal{H}$, call $k$ a **break point** for $\mathcal{H}$

- $m_{\mathcal{H}}(k) < 2^k$
- $k + 1, k + 2, k + 3, \ldots$ also break points!
Break Point of $\mathcal{H}$

**what do we know about 2D perceptrons now?**

- **three inputs**: ‘exists’ shatter;
- **four inputs**: ‘for all’ no shatter

**if no $k$ inputs can be shattered by $\mathcal{H}$, call $k$ a **break point** for $\mathcal{H}$**

- $m_{\mathcal{H}}(k) < 2^k$
- $k + 1, k + 2, k + 3, \ldots$ also break points!
- will study **minimum break point** $k$
Break Point of $\mathcal{H}$

**what do we know about 2D perceptrons now?**

- **three inputs:** ‘exists’ shatter;
- **four inputs:** ‘for all’ no shatter

If no $k$ inputs can be shattered by $\mathcal{H}$, call $k$ a **break point** for $\mathcal{H}$

- $m_\mathcal{H}(k) < 2^k$
- $k + 1, k + 2, k + 3, \ldots$ also break points!
- Will study **minimum break point $k$**

2D perceptrons: **break point at 4**
The Four Break Points

- **positive rays:** \( m_H(N) = N + 1 \)
  break point at 2

- **positive intervals:** \( m_H(N) = \frac{1}{2}N^2 + \frac{1}{2}N + 1 \)
  break point at 3

- **convex sets:**
  no break point

- **2D perceptrons:** \( m_H(N) < 2^N \) in some cases
  break point at 4
The Four Break Points

- **positive rays:**  \( m_H(N) = N + 1 = O(N) \)  
  break point at 2

- **positive intervals:**  \( m_H(N) = \frac{1}{2}N^2 + \frac{1}{2}N + 1 = O(N^2) \)  
  break point at 3

- **convex sets:**  \( m_H(N) = 2^N \)  
  no break point

- **2D perceptrons:**  \( m_H(N) < 2^N \text{ in some cases} \)  
  break point at 4
The Four Break Points

- **positive rays:**
  \[ m_H(N) = N + 1 = O(N) \]
  break point at 2

- **positive intervals:**
  \[ m_H(N) = \frac{1}{2}N^2 + \frac{1}{2}N + 1 = O(N^2) \]
  break point at 3

- **convex sets:**
  \[ m_H(N) = 2^N \]
  no break point

- **2D perceptrons:**
  \[ m_H(N) < 2^N \text{ in some cases} \]
  break point at 4

**Conjecture:**
- no break point: \( m_H(N) = 2^N \) (sure!)
- break point \( k \): \( m_H(N) = O(N^{k-1}) \)
## The Four Break Points

- **positive rays:**
  
  \[ m_{\mathcal{H}}(N) = N + 1 = O(N) \]

  break point at 2

- **positive intervals:**
  
  \[ m_{\mathcal{H}}(N) = \frac{1}{2} N^2 + \frac{1}{2} N + 1 = O(N^2) \]

  break point at 3

- **convex sets:**
  
  \[ m_{\mathcal{H}}(N) = 2^N \]

  no break point

- **2D perceptrons:**
  
  \[ m_{\mathcal{H}}(N) < 2^N \text{ in some cases} \]

  break point at 4

---

conjecture:

- no break point: \( m_{\mathcal{H}}(N) = 2^N \) (sure!)
- break point \( k \): \( m_{\mathcal{H}}(N) = O(N^{k-1}) \)

excited? wait for next lecture :-)
Consider positive and negative rays as $\mathcal{H}$, which is equivalent to the perceptron hypothesis set in 1D. As discussed in an earlier quiz question, the growth function $m_{\mathcal{H}}(N) = 2N$. What is the minimum break point for $\mathcal{H}$?
Consider positive and negative rays as $\mathcal{H}$, which is equivalent to the perceptron hypothesis set in 1D. As discussed in an earlier quiz question, the growth function $m_{\mathcal{H}}(N) = 2N$. What is the minimum break point for $\mathcal{H}$?

1 1 2 2 3 3 4 4

Reference Answer: 3

At $k = 3$, $m_{\mathcal{H}}(k) = 6$ while $2^k = 8$. 
Summary

1. When Can Machines Learn?

Lecture 4: Feasibility of Learning

2. Why Can Machines Learn?

Lecture 5: Training versus Testing

- Recap and Preview
  - Two questions: \( E_{\text{out}}(g) \approx E_{\text{in}}(g) \), and \( E_{\text{in}}(g) \approx 0 \)
- Effective Number of Lines
  - At most 14 through the eye of 4 inputs
- Effective Number of Hypotheses
  - At most \( m_H(N) \) through the eye of \( N \) inputs
- Break Point
  - When \( m_H(N) \) becomes ‘non-exponential’

- Next: \( m_H(N) = \text{poly}(N) \)?

3. How Can Machines Learn?

4. How Can Machines Learn Better?