CSE 417T
Introduction to Machine Learning

Lecture 6
Instructor: Chien-Ju (CJ) Ho
Recap
Theory of Generalization

• Learning from a finite hypothesis set: learn $g \in \{h_1, ..., h_M\}$

With prob $1 - \delta$, $E_{\text{out}}(g) \leq E_{\text{in}}(g) + \sqrt{\frac{1}{2N} \ln \frac{2^M}{\delta}}$

• What if $M \to \infty$

• Dichotomies
  • Informally, consider a dichotomy as a "data-dependent" hypothesis
  • Characterized by both hypothesis set $H$ and $N$ data points $(\tilde{x}_1, ..., \tilde{x}_N)$
    
    \[ H(\tilde{x}_1, ..., \tilde{x}_N) = \{(h(\tilde{x}_1), ..., h(\tilde{x}_N))|h \in H\} \]
  • The set of possible prediction combinations $h \in H$ can induce on $\tilde{x}_1, ..., \tilde{x}_N$

• Growth function
  • Largest number of dichotomies $H$ can induce across all possible data sets of size $N$
    
    \[ m_H(N) = \max_{(\tilde{x}_1, ..., \tilde{x}_N)} |H(\tilde{x}_1, ..., \tilde{x}_N)| \]

• VC Generalization Bound

With prob $1 - \delta$, $E_{\text{out}}(g) \leq E_{\text{in}}(g) + \sqrt{\frac{8}{N} \ln \frac{4m_H(2N)}{\delta}}$
Bounding Growth Functions

• More definitions....
  • Shatter
    - \( H \) shatters \((\tilde{x}_1, ..., \tilde{x}_N)\) if \( |H(\tilde{x}_1, ..., \tilde{x}_N)| = 2^N \)
    - \( H \) can induce all label combinations for \((\tilde{x}_1, ..., \tilde{x}_N)\)
  • Break point
    - \( k \) is a break point for \( H \) if no data set of size \( k \) can be shattered by \( H \)
    - \( k \) is a break point for \( H \iff m_H(k) < 2^k \)

• VC Dimension: \( d_{vc}(H) \) or \( d_{vc} \)
  - The VC dimension of \( H \) is the largest \( N \) such that \( m_H(N) = 2^N \)
  - Equivalently, if \( k^* \) is the smallest break point for \( H \), \( d_{vc}(H) = k^* - 1 \)
## Examples

<table>
<thead>
<tr>
<th></th>
<th>$m_H(N)$</th>
<th>Break Points</th>
<th>VC Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive Rays</td>
<td>(N=1)</td>
<td>2, 3, 4, ...</td>
<td>1</td>
</tr>
<tr>
<td>Positive Intervals</td>
<td>(N=2)</td>
<td>2, 3, 4, ...</td>
<td>2</td>
</tr>
<tr>
<td>Convex Sets</td>
<td>(N=3)</td>
<td>None</td>
<td>$\infty$</td>
</tr>
<tr>
<td>2D Perceptron</td>
<td>(N=4)</td>
<td>2, 3, 4, ...</td>
<td>3</td>
</tr>
</tbody>
</table>

**Diagrams:**

- **Positive Rays:**
  - Decision boundary at point $a$.
  - Predict +1 and Predict -1.

- **Positive Intervals:**
  - Decision boundary at points $a$ and $b$.
  - Predict +1 and Predict -1.

- **Convex Sets:**
  - Boundary defined by points $a$ and $b$.
  - Predict +1 and Predict -1.

- **2D Perceptron:**
  - Decision boundary in 2D space.
Bounding Growth Functions

• Theorem statement:
  • If there is no break point for $H$, then $m_H(N) = 2^N$ for all $N$.
  • If $k$ is a break point for $H$, i.e., if $m_H(k) < 2^k$ for some value $k$, then
    $$m_H(N) \leq \sum_{i=0}^{k-1} \binom{N}{i}$$

• Rephrase the 2$^{nd}$ statement of the above theorem
  • If $k$ is a break point for $H$, the following statements are true
    • $m_H(N) \leq N^{k-1} + 1$ [Can be proven using induction from above. See LFD Problem 2.5]
    • $m_H(N) = O(N^{k-1})$
    • $m_H(N)$ is polynomial in $N$

• If $d_{vc}$ is the VC dimension of $H$, then
  • $m_H(N) \leq \sum_{i=0}^{d_{vc}} \binom{N}{i}$
  • $m_H(N) \leq N^{d_{vc}} + 1$
  • $m_H(N) = O(N^{d_{vc}})$

If $d_{vc}$ is the VC dimension of $H$, $d_{vc} + 1$ is a break point for $H$
Vapnik–Chervonenkis (VC) Bound

• VC Generalization Bound
  With prob at least $1 - \delta$
  $$E_{out}(g) \leq E_{in}(g) + \sqrt{\frac{8}{N} \ln \frac{4m_H(2N)}{\delta}}$$

• Let $d_{vc}$ be the VC dimension of $H$, we have $m_H(N) \leq N^{d_{vc}} + 1$.
  With prob at least $1 - \delta$
  $$E_{out}(g) \leq E_{in}(g) + \sqrt{\frac{8}{N} \ln \frac{4((2N)^{d_{vc}} + 1)}{\delta}}$$

• If we treat $\delta$ as a constant, then we can say, with high probability
  $$E_{out}(g) \leq E_{in}(g) + O \left( \sqrt{d_{vc} \frac{\ln N}{N}} \right)$$
Discussion on the VC Bound

• Think about the high-level tradeoff of choosing $d_{VC}$ and its dependency on $N$
• The approximation-generalization trade-off

\[ E_{out}(g) \leq E_{in}(g) + O \left( \sqrt{d_{VC} \frac{\ln N}{N}} \right) \]

What we want to minimize

How well \( g \) generalizes

How well \( g \) approximates \( f \) in training data

Error

out-of-sample error

model complexity

in-sample error

\( d_{vc} \)

VC dimension, \( d_{vc} \)
Today’s Lecture

The notes are not intended to be comprehensive. They should be accompanied by lectures and/or textbook. Let me know if you spot errors.
Bias-Variance Decomposition

Another theory of generalization
Real-Value Target and Squared Error

• So far, we focus on binary target function and binary error
  • Binary target function $f(\vec{x}) \in \{-1,1\}$
  • Binary error $e(h(\vec{x}), f(\vec{x})) = \mathbb{1}[h(\vec{x}) \neq f(\vec{x})]$

• Real-value target functions ["regression"] and squared error?
  • Real-value target function $f(\vec{x}) \in \mathbb{R}$
  • Squared error $e(h(\vec{x}), f(\vec{x})) = (h(\vec{x}) - f(\vec{x}))^2$
Real-Value Target and Squared Error

• Real-value target functions [called ”regression”] and squared error?
  • Real-value target function $f(\vec{x}) \in \mathbb{R}$
  • Squared error $e(h(\vec{x}), f(\vec{x})) = (h(\vec{x}) - f(\vec{x}))^2$

• Errors:
  • In-sample error: $E_{in}(g) = \frac{1}{N} \sum_{n=1}^{N} e(h(\vec{x}_n), f(\vec{x}_n)) = \frac{1}{N} \sum_{n=1}^{N} (h(\vec{x}_n) - f(\vec{x}_n))^2$
  • Out-of-sample error: $E_{out}(g) = \mathbb{E}_{\vec{x}}[e(h(\vec{x}), f(\vec{x}))] = \mathbb{E}_{\vec{x}}[(g(\vec{x}) - f(\vec{x}))^2]$

• Theory of generalization: What can we say about $E_{out}(g)$?
Note that $g$ is learned by some algorithm on the dataset $D$

- We’ll make the dependency on $D$ explicit and write it as $g^{(D)}$ here.
- [In VC theory, we consider the worst-case $D$ through the definition of growth function $m_H(N)$]

- $E_{out}(g^{(D)}) = \mathbb{E}_{\tilde{x}}\left[\left(g^{(D)}(\tilde{x}) - f(\tilde{x})\right)^2\right]$
- $\mathbb{E}_D[E_{out}(g^{(D))}]$
  \[= \mathbb{E}_D\left[\mathbb{E}_{\tilde{x}}\left[\left(g^{(D)}(\tilde{x}) - f(\tilde{x})\right)^2\right]\right]\]
  \[= \mathbb{E}_{\tilde{x}}\left[\mathbb{E}_D\left[\left(g^{(D)}(\tilde{x}) - \bar{g}(\tilde{x}) + \bar{g}(\tilde{x}) - f(\tilde{x})\right)^2\right]\right]\]
  \[= \mathbb{E}_{\tilde{x}}\left[\mathbb{E}_D\left[\left(g^{(D)}(\tilde{x}) - \bar{g}(\tilde{x}) + \bar{g}(\tilde{x}) - f(\tilde{x})\right)^2\right]\right]\]
  \[= \mathbb{E}_{\tilde{x}}\left[\mathbb{E}_D\left[\left(g^{(D)}(\tilde{x}) - \bar{g}(\tilde{x})\right)^2 + \left(\bar{g}(\tilde{x}) - f(\tilde{x})\right)^2 + 2\left(g^{(D)}(\tilde{x}) - \bar{g}(\tilde{x})\right)\left(\bar{g}(\tilde{x}) - f(\tilde{x})\right)\right]\right]\]

- Define “expected” hypothesis $\bar{g}(\tilde{x}) = \mathbb{E}_D[g^{(D)}(\tilde{x})]$

- Note that $\mathbb{E}_D\left[\left(g^{(D)}(\tilde{x}) - \bar{g}(\tilde{x})\right)\left(\bar{g}(\tilde{x}) - f(\tilde{x})\right)\right] = (\bar{g}(\tilde{x}) - f(\tilde{x})) \mathbb{E}_D\left[\left(g^{(D)}(\tilde{x}) - \bar{g}(\tilde{x})\right)\right] = 0$
Finishing Up

\[ \mathbb{E}_D[E_{\text{out}}(g^{(D)})] = \mathbb{E}_{\bar{x}} \left[ \mathbb{E}_D \left[ \left( g^{(D)}(\bar{x}) - \bar{g}(\bar{x}) \right)^2 \right] \right] + \mathbb{E}_{\bar{x}} \left[ \mathbb{E}_D \left[ \left( \bar{g}(\bar{x}) - f(\bar{x}) \right)^2 \right] \right] = \mathbb{E}_{\bar{x}}[\text{Variance of } g^{(D)}(\bar{x}) + \text{Bias of } \bar{g}(\bar{x})] = \text{Variance} + \text{Bias} \]

- **Bias-Variance Decomposition**

\[ \bar{g}(\bar{x}) = \mathbb{E}_D[g^{(D)}(\bar{x})] \]

**X**: a random variable

\[ \mu: \text{the mean of } X \]

**Variance of X**:

\[ \text{Var}(X) = \mathbb{E}[(X - \mu)^2] \]
Discussion

• $\mathbb{E}_D[E_{out}(g^{(D)})] = \mathbb{E}_{\tilde{x}} \left[ (\bar{g}(\tilde{x}) - f(\tilde{x}))^2 \right] + \mathbb{E}_{\tilde{x}} \left[ \mathbb{E}_D \left[ (g^{(D)}(\tilde{x}) - \bar{g}(\tilde{x}))^2 \right] \right]$

• This is a **conceptual** decomposition
  • Both $\bar{g}$ and $f$ are unknown
  • We can’t really calculate bias and variance in practice

• However, it provides a conceptual guideline in decreasing $E_{out}$
Example of Bias-Variance Decomposition

• Fitting a sine function
  • \( f(x) = \sin(\pi x) \)
  • \( x \) is drawn uniformly at random from \([0,2]\)

• Two hypothesis set
  • \( H_0: h(x) = b \)
  • \( H_1: h(x) = ax + b \)

• Assume our algorithm finds \( g \) with minimum in-sample error
Example of Bias-Variance Decomposition

\[ H_0: h(x) = b \]
\[ H_1: h(x) = ax + b \]

Discussion:
If \( N = 2 \), would you choose \( H_0 \) or \( H_1 \)? Why?
If \( N = 5 \), would you choose \( H_0 \) or \( H_1 \)? Why?
What’s the change of biases/variances for \( H_0/ H_1 \) from \( N = 2 \) to \( N = 5 \).
Example of Bias-Variance Decomposition

\[ H_0: h(x) = b \]

\[ H_1: h(x) = ax + b \]
Example of Bias-Variance Decomposition

\[ H_0: h(x) = b \]

\[ H_1: h(x) = ax + b \]
Example of Bias-Variance Decomposition

\[ H_0: h(x) = b \]

\[ H_1: h(x) = ax + b \]

\[
E_D [E_{out}(g^D)] = E_x \left[ (\tilde{g}(x) - f(x))^2 \right] + E_x \left[ E_D \left[ (g^D(x) - \tilde{g}(x))^2 \right] \right]
\]

Bias of \( \tilde{g}(\hat{x}) \) ≈ 0.50
Variance of \( g_D(\hat{x}) \) ≈ 0.25
\( E_D [E_{out}(g_D)] \) ≈ 0.75

Bias of \( \tilde{g}(\hat{x}) \) ≈ 0.21
Variance of \( g_D(\hat{x}) \) ≈ 1.74
\( E_D [E_{out}(g_D)] \) ≈ 1.95
What if we increase $N$ to 5?

$H_0$: $h(x) = b$

$H_1$: $h(x) = ax + b$

Bias of $\bar{g}(\hat{x}) \approx 0.50$

Variance of $g_D(\hat{x}) \approx 0.10$

$\mathbb{E}_D[\mathbb{E}_{out}(g_D)] \approx 0.60$

Bias of $\bar{g}(\hat{x}) \approx 0.21$

Variance of $g_D(\hat{x}) \approx 0.21$

$\mathbb{E}_D[\mathbb{E}_{out}(g_D)] \approx 0.42$
Discussion

\[
\mathbb{E}_D[E_{out}(g^{(D)})] = \mathbb{E}_{\bar{x}} \left[ (\bar{g}(\bar{x}) - f(\bar{x}))^2 \right] + \mathbb{E}_{\bar{x}} \left[ \mathbb{E}_D \left[ (g^{(D)}(\bar{x}) - \bar{g}(\bar{x}))^2 \right] \right]
\]

• Increasing the number of data points $N$
  • Biases roughly stay the same
  • Variances decrease
  • Expected $E_{out}$ decreases
Discussion

\[
\mathbb{E}_D [E_{out}(g^{(D)})] = \mathbb{E}_{\tilde{x}} \left[ (\bar{g}(\tilde{x}) - f(\tilde{x}))^2 \right] + \mathbb{E}_{\tilde{x}} \left[ \mathbb{E}_D \left[ (g^{(D)}(\tilde{x}) - \bar{g}(\tilde{x}))^2 \right] \right]
\]

- Increasing the complexity of \( H \)
  - Bias goes down (more likely to approximate \( f \))
  - Variance goes up (The stability of \( g^{(D)} \) is worse)
Discussion

\[ \mathbb{E}_D[E_{out}(g^{(D)})] = \mathbb{E}_{\tilde{x}} \left[ (\tilde{g}(\tilde{x}) - f(\tilde{x}))^2 \right] + \mathbb{E}_{\tilde{x}} \left[ \mathbb{E}_D \left[ (g^{(D)}(\tilde{x}) - \tilde{g}(\tilde{x}))^2 \right] \right] \]

- This is a **conceptual** decomposition
  - Both \( \tilde{g} \) and \( f \) are unknown
  - We can’t really calculate bias and variance for practical problems

- However, it provides a conceptual guidelines in decreasing \( E_{out} \)
Example

• Will talk about this in details in the 2\textsuperscript{nd} half of the semester
• Decision tree
  • A low bias but high variance hypothesis set
  • Practical performance is not ideal

• Random forest
  • Trying to reduce the variance while not sacrificing bias
  • Idea: Generate many trees randomly and average them
Two Theories of Generalization

• VC Generalization Bound

\[ E_{out}(g) \leq E_{in}(g) + O\left(\sqrt{d_{vc}} \frac{\ln N}{N}\right) \]

• Bias-Variance Tradeoff

\[ \mathbb{E}_D\left[E_{out}(g^{(D)})\right] = \mathbb{E}_{\tilde{x}}\left[\left(\bar{g}(\tilde{x}) - f(\tilde{x})\right)^2\right] + \mathbb{E}_{\tilde{x}}\left[\mathbb{E}_D\left[\left(g^{(D)}(\tilde{x}) - \bar{g}(\tilde{x})\right)^2\right]\right] \]
Learning Curves

Simple Model

Complex Model

Expected Error

Number of Data Points, $N$
Learning Curves

Simple Model

\[
E_{\text{out}} \quad E_{\text{in}}
\]

Complex Model

\[
\text{Expected Error}
\]

Number of Data Points, \( N \)
Learning Curves

Simple Model

Complex Model

Expected Error

Number of Data Points, $N$

$E_{\text{out}}$

$E_{\text{in}}$

$E_{\text{out}}$

$E_{\text{in}}$
Learning Curves

VC Analysis

Bias-Variance Analysis

Expected Error

Expected Error

Number of Data Points, $N$

Number of Data Points, $N$
Learning Curves

VC Analysis

- Expected Error vs. Number of Data Points, $N$
- Generalization error $E_{\text{out}}$
- In-sample error $E_{\text{in}}$

Bias-Variance Analysis

- Expected Error vs. Number of Data Points, $N$
Learning Curves

VC Analysis

Expected Error

generalization error

$E_{\text{out}}$

in-sample error

$E_{\text{in}}$

Number of Data Points, $N$

Bias-Variance Analysis

Expected Error

variance

$E_{\text{out}}$

bias

$E_{\text{in}}$

Number of Data Points, $N$
Our focus so far
Let’s spend some time here
Linear Models
Linear Models

• $H$ contains hypothesis $h(\vec{x})$ as some function of $\vec{w}^T \vec{x}$

<table>
<thead>
<tr>
<th>Domain</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear Classification</td>
<td>$y \in {-1, +1}$</td>
</tr>
<tr>
<td>Linear Regression</td>
<td>$y \in \mathbb{R}$</td>
</tr>
<tr>
<td>Logistic Regression</td>
<td>$y \in [0,1]$</td>
</tr>
</tbody>
</table>

This is why it’s called linear models

• Linear models:
  • Simple models => Good generalization error

• Reminder:
  • We will interchangeably use $h$ and $\vec{w}$ to represent a hypothesis in linear models

Credit Card Example

- Approve or not
- Credit line
- Prob. of default

\[ \theta(s) = \frac{e^s}{1 + e^s} \]
Learning Algorithm?

• Goal of the algorithm: Find $g \in H$ that minimizes $E_{out}(g)$
  (We don’t know $E_{out}$)

• Common algorithms:
  • $g = \arg\min_{h \in H} E_{in}(h)$
    • Works well when the model is simple (generalization error is small)
    • Will focus on this in the discussion of linear models
  • $g = \arg\min_{h \in H} \{E_{in}(h) + \Omega(h)\}$
    • $\Omega(h)$: penalty for complex $h$
    • Will discuss this when we get to LFD Section 4

• Optimization is a key component in machine learning

VC Bound: $E_{out}(g) \leq E_{in}(g) + O\left(\sqrt{d_{vc} \frac{\ln N}{N}}\right)$