# ECE595 / STAT598: Machine Learning I

## Lecture 30.2: Overfit - Analyzing Overfit

Spring 2020

Stanley Chan

School of Electrical and Computer Engineering
Purdue University

PURDUE
UNIVERSITY

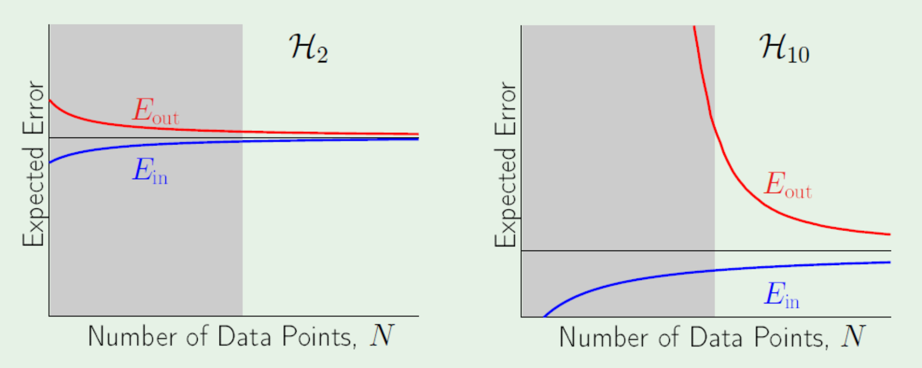
# Outline

- Lecture 30 Overfit
- Lecture 31 Regularization
- Lecture 32 Validation

**Today's Lecture**:

- Source of Overfit
  - Is Noise the Reason?
  - Is Model Complexity the Reason?
  - The Trinity of Noise, Target Complexity, and Training Sample
- Analyzing Overfit
  - Bias and Variance
  - Learning Curve

# Learning Curve




- Noise free
- When $N$ is small, $\mathcal{H}_2$ has lower $E_{\text{out}}$
- $\mathcal{H}_2$ has higher steady state than $\mathcal{H}_{10}$

# Bias-Variance

- Recall this derivation:

$$\begin{aligned}
&\mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_{\text{out}}(g^{(\mathcal{D})})\right] \\
&= \mathbb{E}_{\boldsymbol{x}}\left[\mathbb{E}_{\mathcal{D}}\left[g^{(\mathcal{D})}(\boldsymbol{x})^2\right] - 2\mathbb{E}_{\mathcal{D}}[g^{(\mathcal{D})}(\boldsymbol{x})]f(\boldsymbol{x}) + f(\boldsymbol{x})^2\right] \\
&= \mathbb{E}_{\boldsymbol{x}}\Big[\underbrace{\mathbb{E}_{\mathcal{D}}\left[g^{(\mathcal{D})}(\boldsymbol{x})^2\right] - \overline{g}(\boldsymbol{x})^2}_{\mathbb{E}_{\mathcal{D}}[(g^{(\mathcal{D})}(\boldsymbol{x}) - \overline{g}(\boldsymbol{x}))^2]} + \underbrace{\overline{g}(\boldsymbol{x})^2 - 2\mathbb{E}_{\mathcal{D}}[g^{(\mathcal{D})}(\boldsymbol{x})]f(\boldsymbol{x}) + f(\boldsymbol{x})^2}_{(\overline{g}(\boldsymbol{x}) - f(\boldsymbol{x}))^2}\Big].
\end{aligned}$$

- The bias and variance are defined as

$$\begin{aligned}
\text{bias}(\boldsymbol{x}) &\overset{\text{def}}{=} (\overline{g}(\boldsymbol{x}) - f(\boldsymbol{x}))^2, \\
\text{var}(\boldsymbol{x}) &\overset{\text{def}}{=} \mathbb{E}_{\mathcal{D}}[(g^{(\mathcal{D})}(\boldsymbol{x}) - \overline{g}(\boldsymbol{x}))^2].
\end{aligned}$$

- What if $f(\boldsymbol{x}) \longleftarrow f(\boldsymbol{x}) + \epsilon(\boldsymbol{x})$, where $\mathbb{E}[\epsilon(\boldsymbol{x})] = 0$?

# Bias-Variance with Noise

- We can show the following:

$$\begin{aligned}
&\mathbb{E}_{\mathcal{D},\epsilon}\left[(g^{(D)}(\boldsymbol{x}) - (f(\boldsymbol{x}) + \epsilon(\boldsymbol{x})))^2\right] \\
&= \mathbb{E}_{\mathcal{D},\epsilon}\left[(g^{(D)}(\boldsymbol{x}) - \overline{g}(\boldsymbol{x}) + \overline{g}(\boldsymbol{x}) - f(\boldsymbol{x}) - \epsilon(\boldsymbol{x}))^2\right] \\
&= \mathbb{E}_{\mathcal{D},\epsilon}\left[\left(g^{(D)}(\boldsymbol{x}) - \overline{g}(\boldsymbol{x})\right)^2 + (\overline{g}(\boldsymbol{x}) - f(\boldsymbol{x}))^2 + (\epsilon(\boldsymbol{x}))^2\right]
\end{aligned}$$

- Cross-terms involving $\mathbb{E}[\epsilon(\boldsymbol{x})]$ is zero
- So

$$\begin{aligned}
E_{\text{out}} = \mathbb{E}_{\boldsymbol{x}}[☺] = \mathbb{E}_{\mathcal{D},\boldsymbol{x}}\left[\left(g^{(D)}(\boldsymbol{x}) - \overline{g}(\boldsymbol{x})\right)^2\right] + \mathbb{E}_{\boldsymbol{x}}\left[(\overline{g}(\boldsymbol{x}) - f(\boldsymbol{x}))^2\right] \\
+ \mathbb{E}_{\boldsymbol{x},\epsilon}\left[\epsilon(\boldsymbol{x})^2\right]
\end{aligned}$$

# Bias-Variance with Noise

$$E_{\text{out}} = \mathbb{E}_{\mathcal{D},\boldsymbol{x}}\left[\left(g^{(D)}(\boldsymbol{x}) - \overline{g}(\boldsymbol{x})\right)^2\right] + \mathbb{E}_{\boldsymbol{x}}\left[(\overline{g}(\boldsymbol{x}) - f(\boldsymbol{x}))^2\right] + \mathbb{E}_{\boldsymbol{x},\epsilon}\left[\epsilon(\boldsymbol{x})^2\right]$$

- Variance: $\mathbb{E}_{\mathcal{D},\boldsymbol{x}}\left[\left(g^{(D)}(\boldsymbol{x}) - \overline{g}(\boldsymbol{x})\right)^2\right]$
- Bias: $\mathbb{E}_{\boldsymbol{x}}\left[(\overline{g}(\boldsymbol{x}) - f(\boldsymbol{x}))^2\right]$
- Noise: $\mathbb{E}_{\boldsymbol{x},\epsilon}\left[\epsilon(\boldsymbol{x})^2\right]$
- Overfitting $\downarrow$ if number of data points $\uparrow$
- Overfitting $\uparrow$ if noise $\uparrow$
- Overfitting $\uparrow$ if target complexity $\uparrow$

# Summary

- Overfit happens because of noise, target complexity and training samples.
- Overcoming overfit requires:
  - Reduce the amount of noise in data (Could be hard)
  - Reduce target complexity (May not be possible)
  - Increase training samples
- What else can we do?
  - Choose a low complexity model even though target complexity is high
  - Regularize the model complexity by promoting low order models