

CSE 417T

Introduction to Machine Learning

Lecture 6

Instructor: Chien-Ju (CJ) Ho

Recap

Discussion on the VC Bound

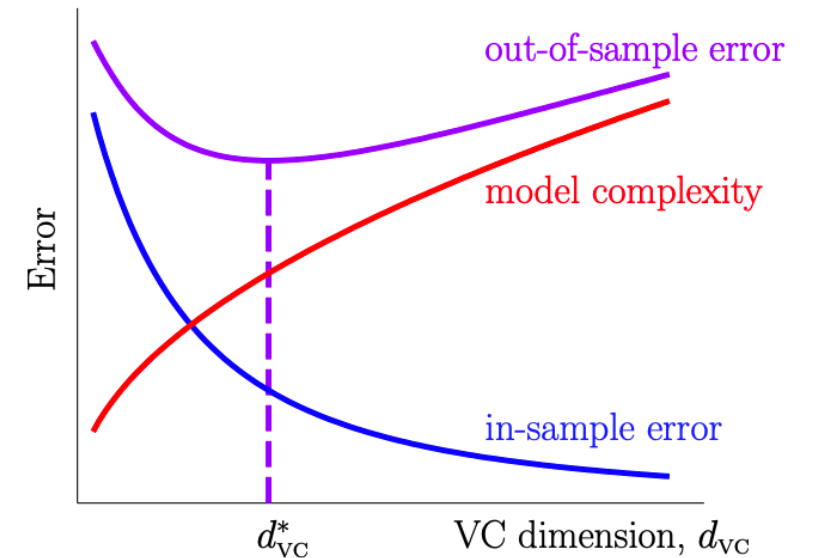
- Think about the high-level tradeoff of choosing d_{VC} and its dependency on N
- The approximation-generalization trade-off

What we want to minimize

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

How well g generalizes

How well g approximates f in training data



Today's Lecture

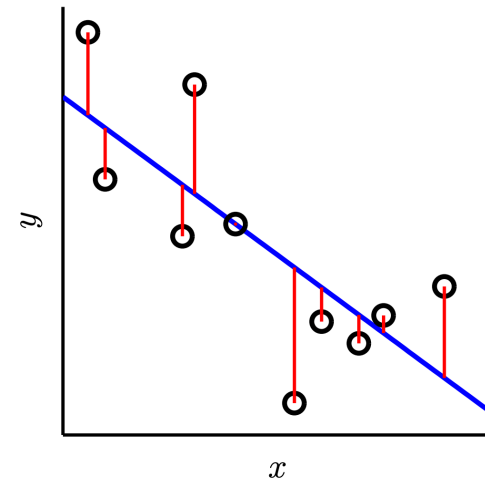
The notes are not intended to be comprehensive. They should be accompanied by lectures and/or textbook.
Let me know if you spot errors.

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 $e(h(\vec{x}), f(\vec{x})) = \mathbb{I}[h(\vec{x}) \neq f(\vec{x})]$
- 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 $m_H(N)$]

- $E_{out}(g^{(D)}) = \mathbb{E}_{\vec{x}}[(g^{(D)}(\vec{x}) - f(\vec{x}))^2]$

- $\mathbb{E}_D[E_{out}(g^{(D)})]$

$$= \mathbb{E}_D \left[\mathbb{E}_{\vec{x}} \left[(g^{(D)}(\vec{x}) - f(\vec{x}))^2 \right] \right]$$

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

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

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

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

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

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

Finishing Up

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

X : a random variable
 μ : the mean of X

Variance of X :
 $Var(X) = \mathbb{E}[(X - \mu)^2]$

- Bias-Variance Decomposition

Discussion

$$\bullet \mathbb{E}_D[E_{out}(g^{(D)})] = \mathbb{E}_{\vec{x}} \left[\overset{\text{Bias}(\vec{x})}{(\bar{g}(\vec{x}) - f(\vec{x}))^2} \right] + \mathbb{E}_{\vec{x}} \left[\overset{\text{Var}(\vec{x})}{\mathbb{E}_D \left[(g^{(D)}(\vec{x}) - \bar{g}(\vec{x}))^2 \right]} \right]$$

- 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

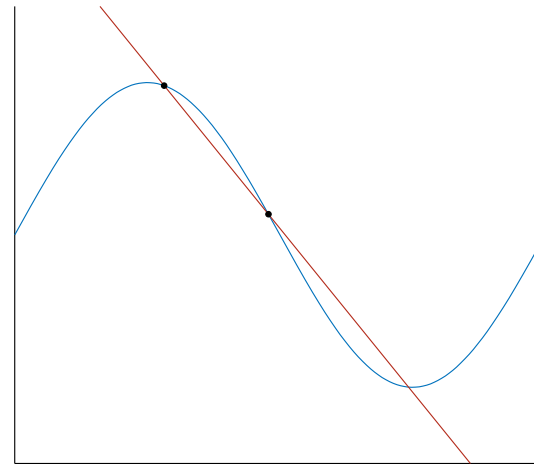
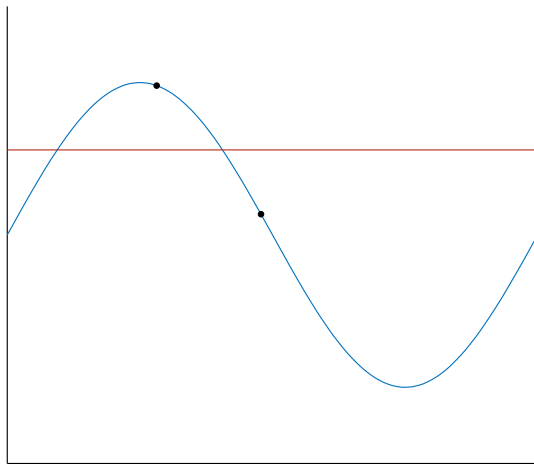
- Fitting a sine function
 - $f(x) = \sin(\pi x)$
 - x is drawn uniformly at random from $[0,2]$
- Two hypothesis set
 - $H_0: h(x) = b$
 - $H_1: h(x) = ax + b$
- Assume our algorithm finds g with minimum in-sample error

Example of Bias-Variance Decomposition

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

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

$N=2$



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

Discussion:

If $N = 2$, would you choose H_0 or H_1 ? Why?

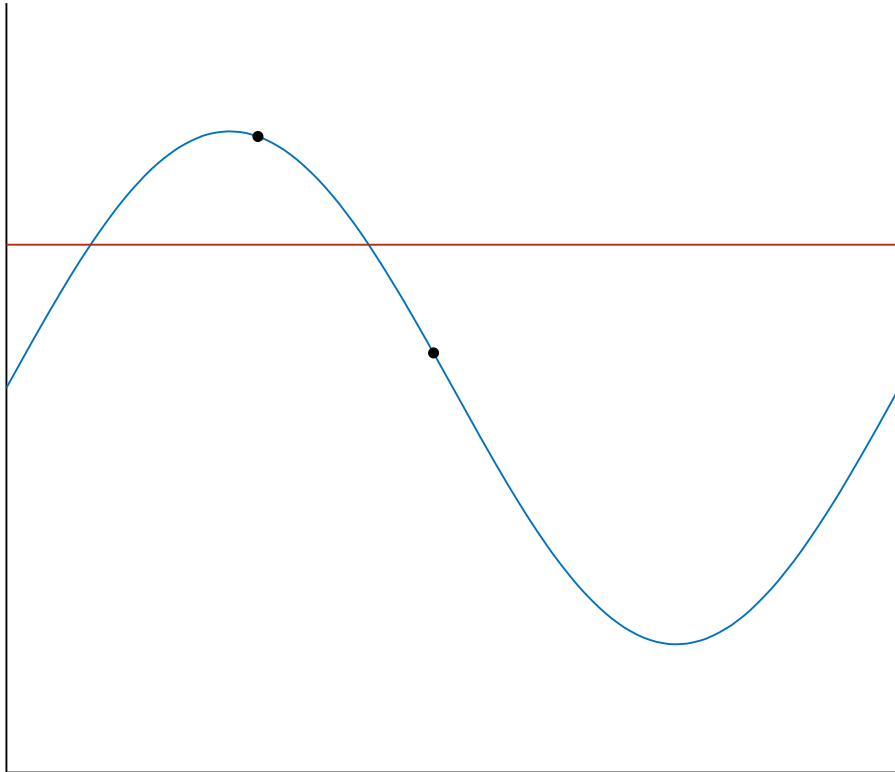
If $N = 5$, would you choose H_0 or H_1 ? Why?

What's the change of biases/variances for H_0/H_1 from $N = 2$ to $N = 5$.

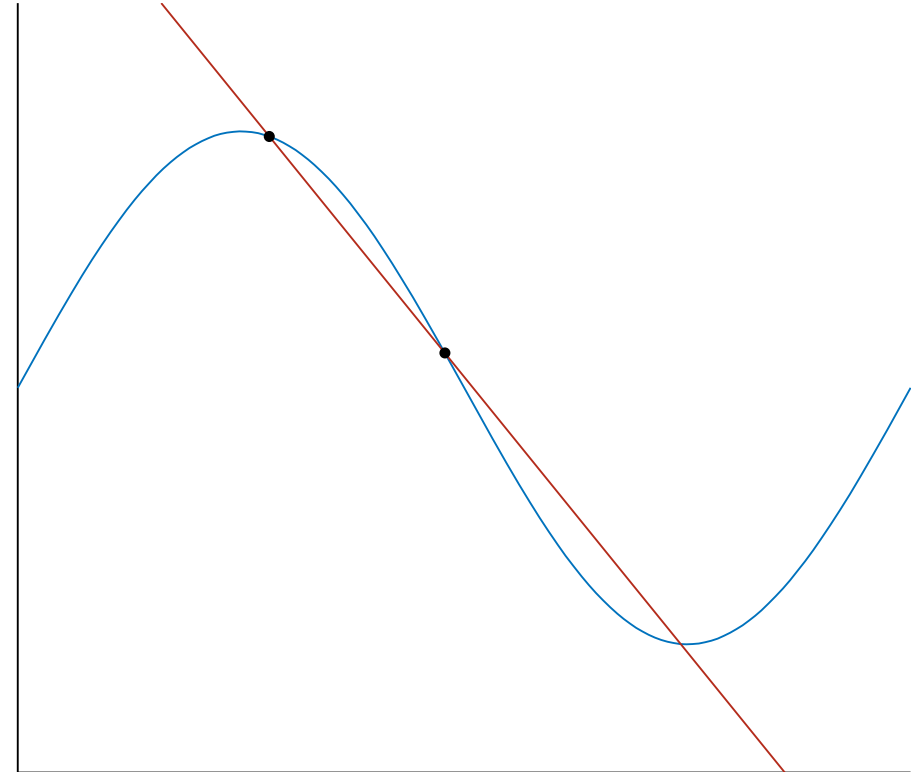
Example of Bias-Variance Decomposition

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

$N=2$



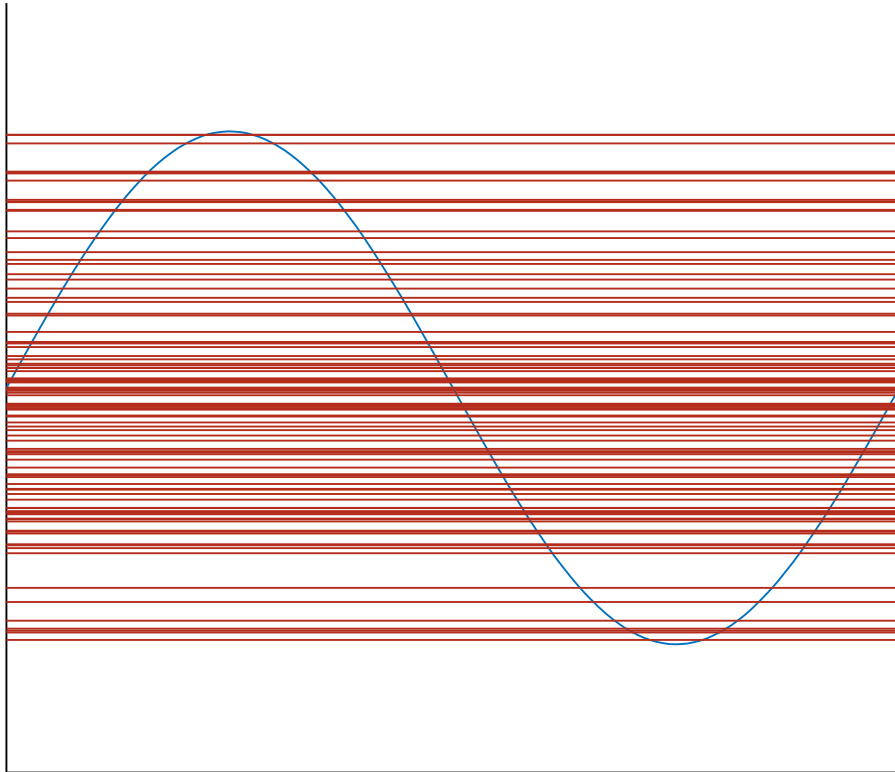
$$H_1: h(x) = ax + b$$



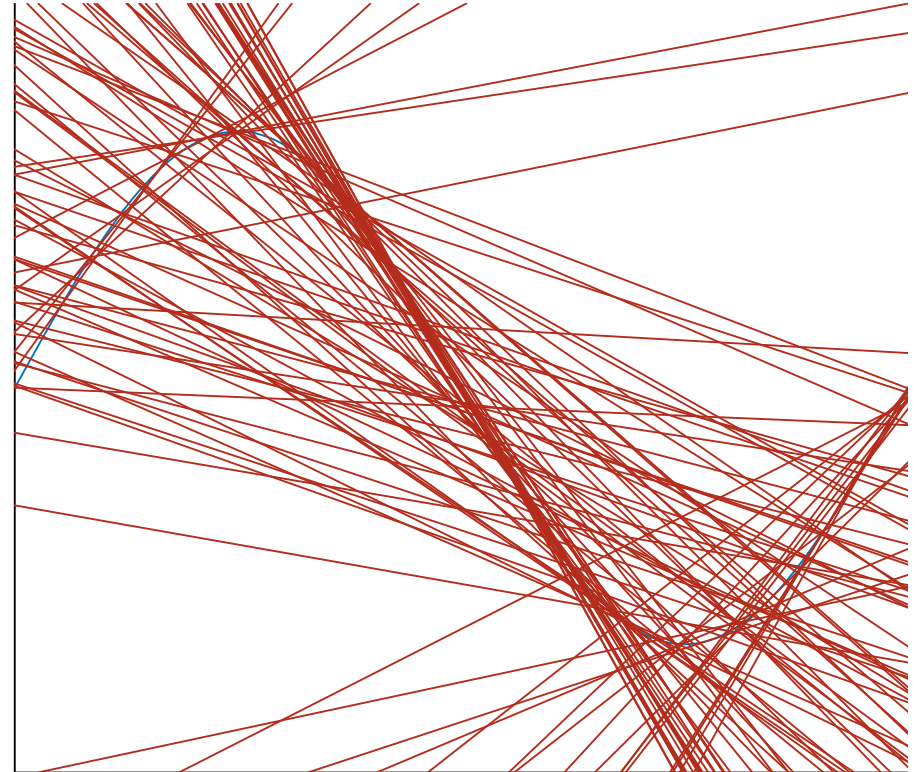
Example of Bias-Variance Decomposition

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

$N=2$



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

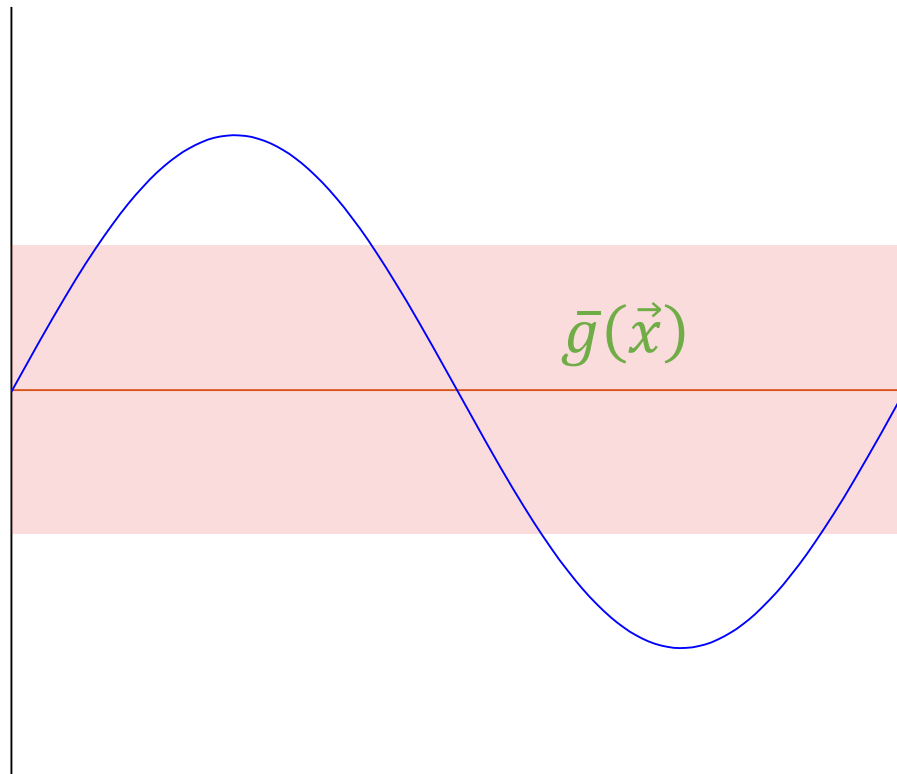


$$\mathbb{E}_D[E_{out}(g^{(D)})] = \mathbb{E}_{\vec{x}} \left[\underbrace{(\bar{g}(\vec{x}) - f(\vec{x}))^2}_{\text{Bias}(\vec{x})} \right] + \mathbb{E}_{\vec{x}} \left[\mathbb{E}_D \left[\underbrace{(g^{(D)}(\vec{x}) - \bar{g}(\vec{x}))^2}_{\text{Var}(\vec{x})} \right] \right]$$

Example of Bias-Variance Decomposition

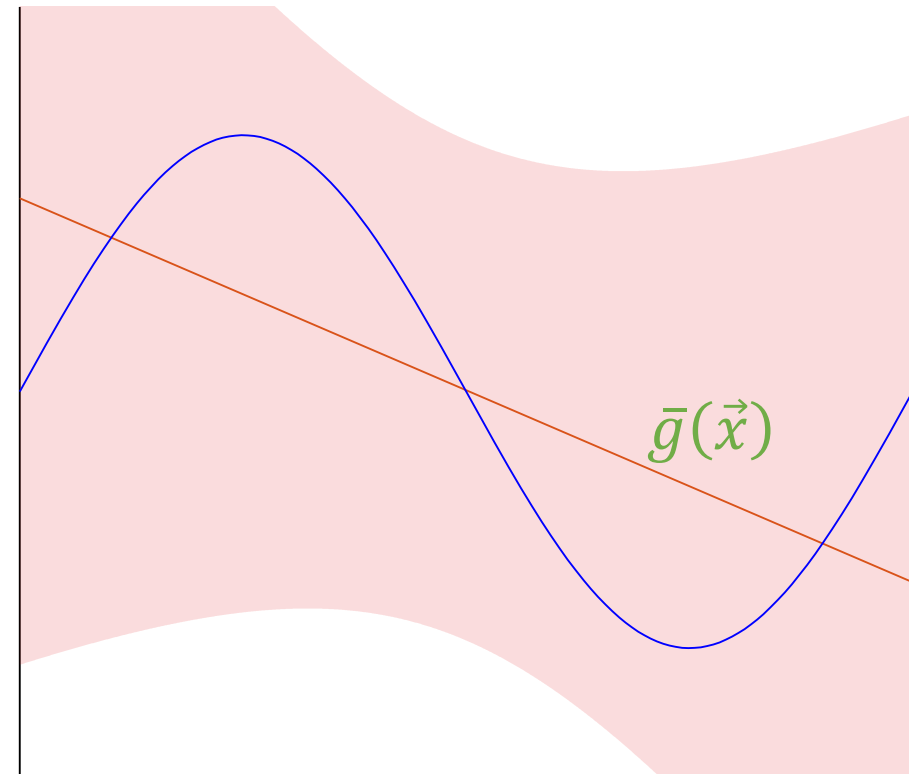
$$H_0: h(x) = b$$

$N=2$



Bias of $\bar{g}(\vec{x}) \approx 0.50$
 Variance of $g_D(\vec{x}) \approx 0.25$
 $\mathbb{E}_D[E_{out}(g_D)] \approx 0.75$

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

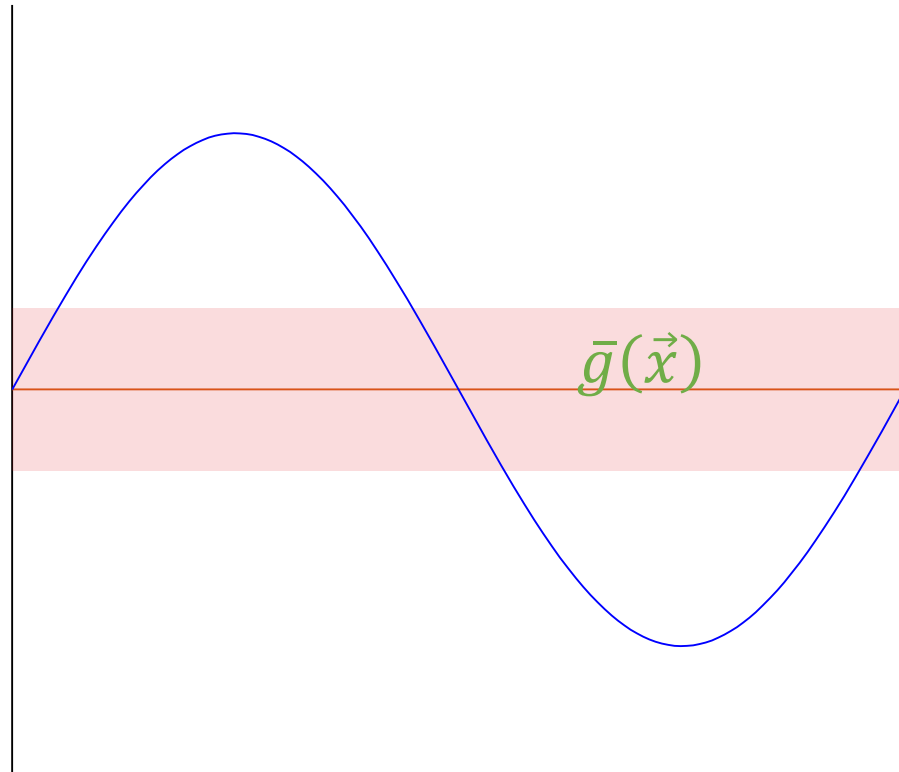


Bias of $\bar{g}(\vec{x}) \approx 0.21$
 Variance of $g_D(\vec{x}) \approx 1.74$
 $\mathbb{E}_D[E_{out}(g_D)] \approx 1.95$

What if we increase N to 5?

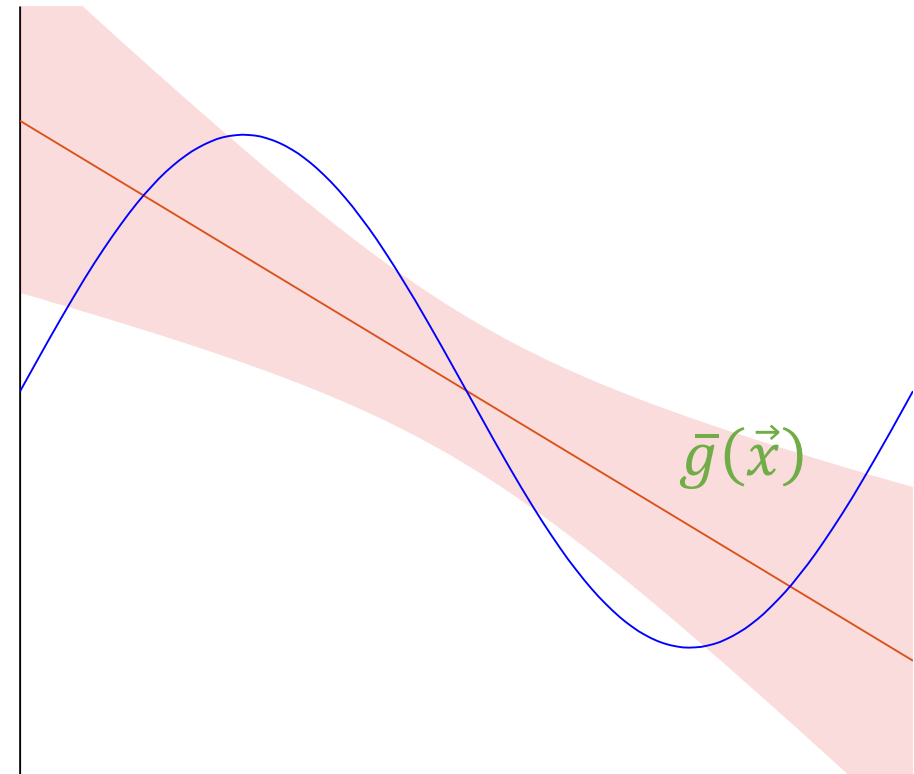
$$\mathbb{E}_{\mathcal{D}}[E_{out}(g^{(\mathcal{D})})] = \mathbb{E}_{\vec{x}} \left[\underbrace{(\bar{g}(\vec{x}) - f(\vec{x}))^2}_{\text{Bias}(\vec{x})} \right] + \mathbb{E}_{\vec{x}} \left[\mathbb{E}_{\mathcal{D}} \left[\underbrace{(g^{(\mathcal{D})}(\vec{x}) - \bar{g}(\vec{x}))^2}_{\text{Var}(\vec{x})} \right] \right]$$

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



$$\begin{aligned} \text{Bias of } \bar{g}(\vec{x}) &\approx 0.50 \\ \text{Variance of } g_{\mathcal{D}}(\vec{x}) &\approx 0.10 \\ \mathbb{E}_{\mathcal{D}}[E_{out}(g_{\mathcal{D}})] &\approx 0.60 \end{aligned}$$

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



$$\begin{aligned} \text{Bias of } \bar{g}(\vec{x}) &\approx 0.21 \\ \text{Variance of } g_{\mathcal{D}}(\vec{x}) &\approx 0.21 \\ \mathbb{E}_{\mathcal{D}}[E_{out}(g_{\mathcal{D}})] &\approx 0.42 \end{aligned}$$

Discussion

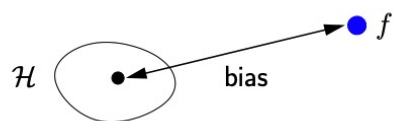
$$\bullet \mathbb{E}_D[E_{out}(g^{(D)})] = \mathbb{E}_{\vec{x}} \left[\overset{\text{Bias}(\vec{x})}{(\bar{g}(\vec{x}) - f(\vec{x}))^2} \right] + \mathbb{E}_{\vec{x}} \left[\overset{\text{Var}(\vec{x})}{\mathbb{E}_D \left[(g^{(D)}(\vec{x}) - \bar{g}(\vec{x}))^2 \right]} \right]$$

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

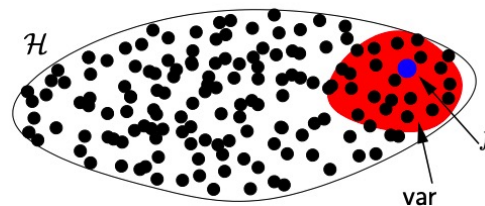
Discussion

$$\bullet \mathbb{E}_D[E_{out}(g^{(D)})] = \mathbb{E}_{\vec{x}} \left[\overset{\text{Bias}(\vec{x})}{(\bar{g}(\vec{x}) - f(\vec{x}))^2} \right] + \mathbb{E}_{\vec{x}} \left[\overset{\text{Var}(\vec{x})}{\mathbb{E}_D \left[(g^{(D)}(\vec{x}) - \bar{g}(\vec{x}))^2 \right]} \right]$$

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



Very small model



Very large model

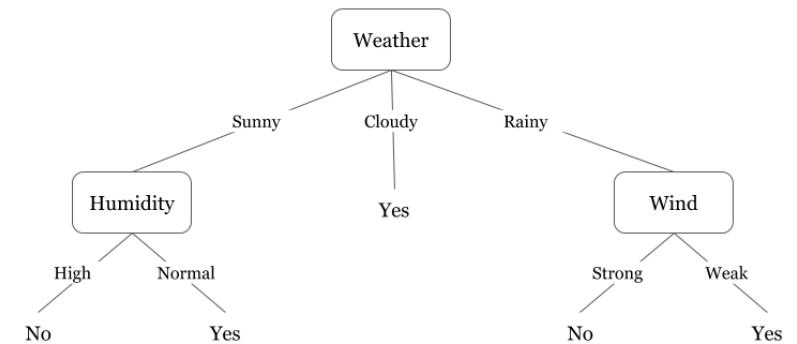
Discussion

$$\bullet \mathbb{E}_D[E_{out}(g^{(D)})] = \mathbb{E}_{\vec{x}} \left[\overset{\text{Bias}(\vec{x})}{(\bar{g}(\vec{x}) - f(\vec{x}))^2} \right] + \mathbb{E}_{\vec{x}} \left[\overset{\text{Var}(\vec{x})}{\mathbb{E}_D \left[(g^{(D)}(\vec{x}) - \bar{g}(\vec{x}))^2 \right]} \right]$$

- This is a **conceptual** decomposition
 - Both \bar{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 2nd 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

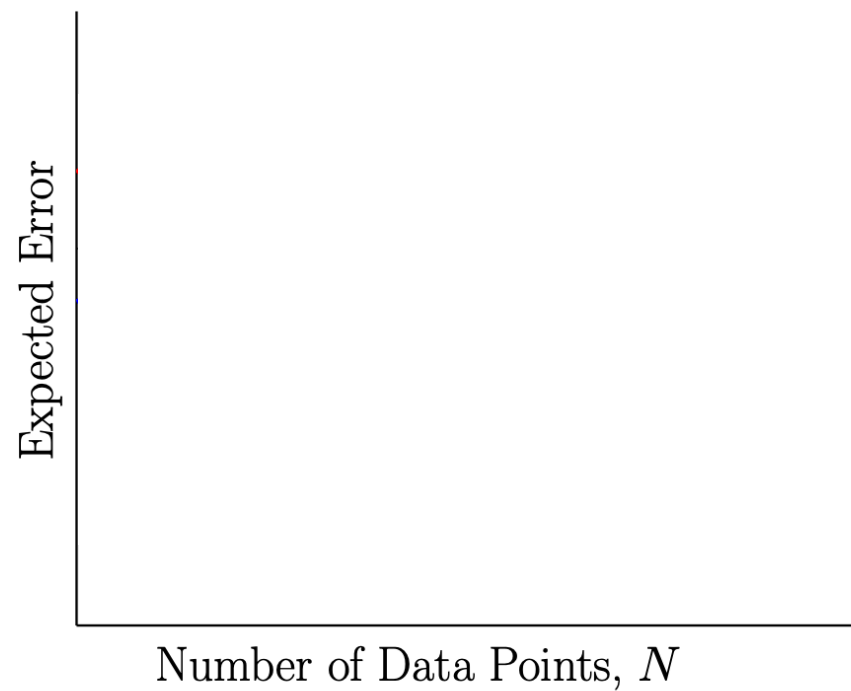
$$E_{out}(g) \leq E_{in}(g) + O\left(\sqrt{d_{vc} \frac{\ln N}{N}}\right)$$

- Bias-Variance Tradeoff

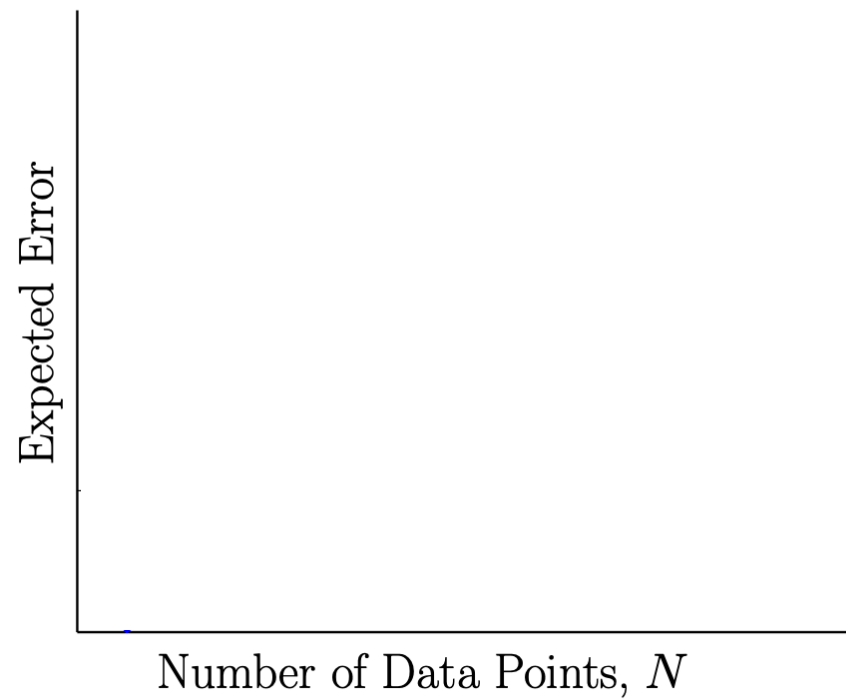
$$\mathbb{E}_D[E_{out}(g^{(D)})] = \mathbb{E}_{\vec{x}} \left[(\bar{g}(\vec{x}) - f(\vec{x}))^2 \right] + \mathbb{E}_{\vec{x}} \left[\mathbb{E}_D \left[(g^{(D)}(\vec{x}) - \bar{g}(\vec{x}))^2 \right] \right]$$

Learning Curves

Simple Model

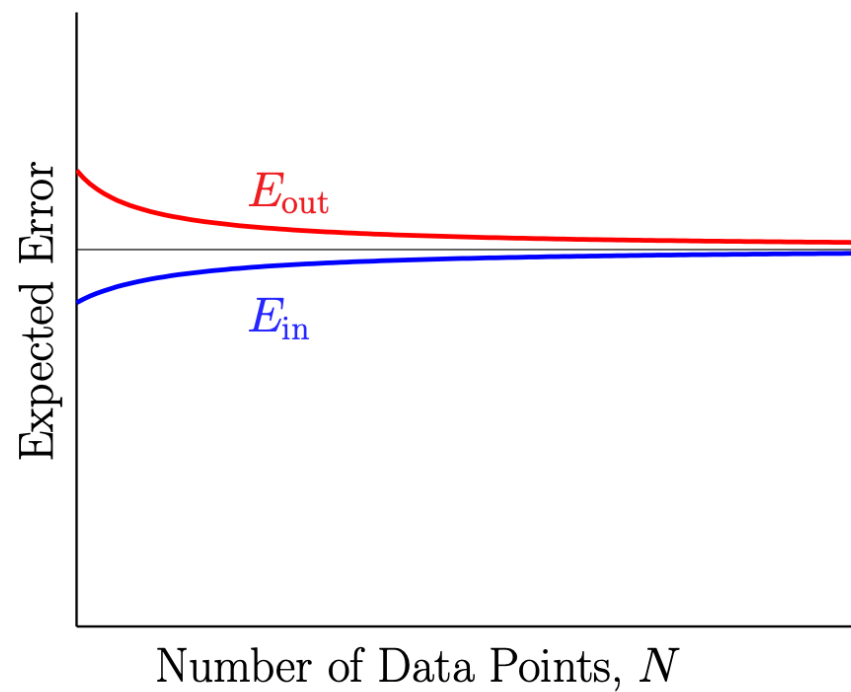


Complex Model

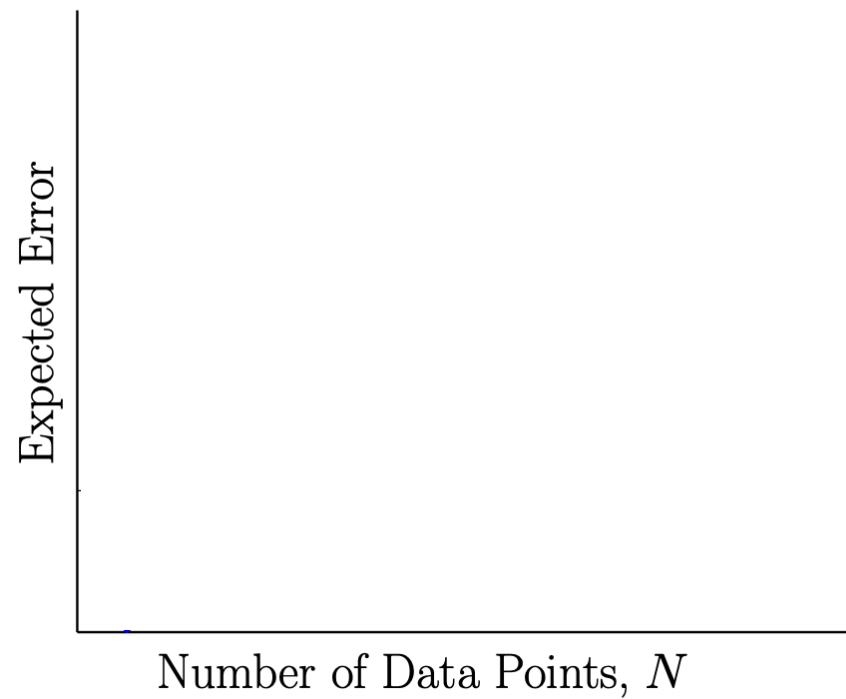


Learning Curves

Simple Model

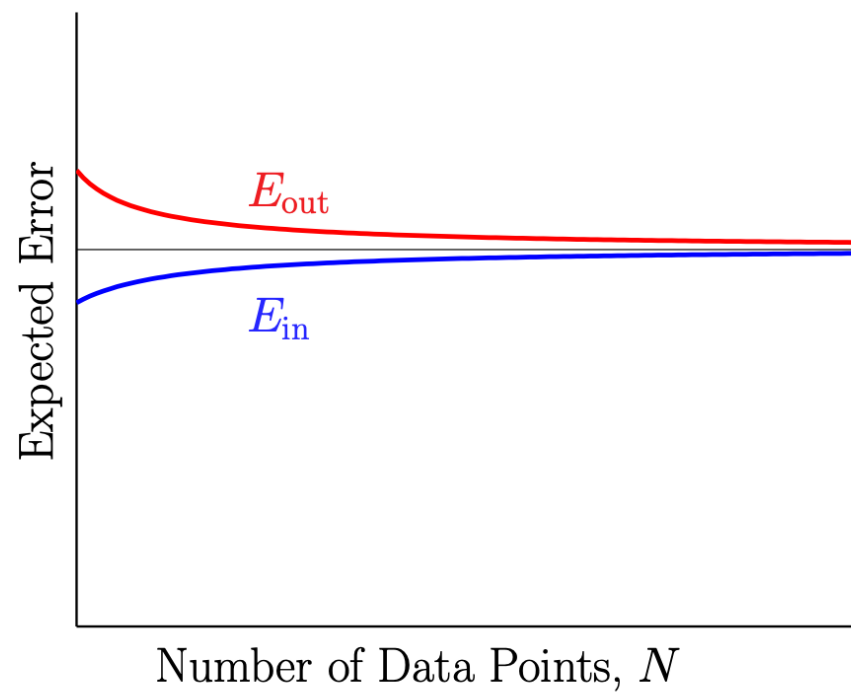


Complex Model

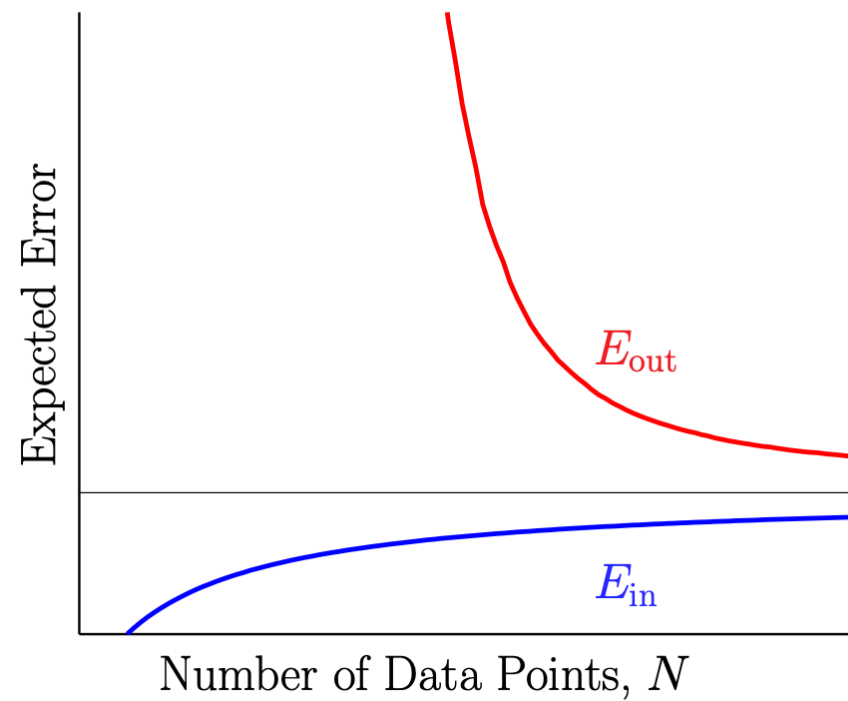


Learning Curves

Simple Model

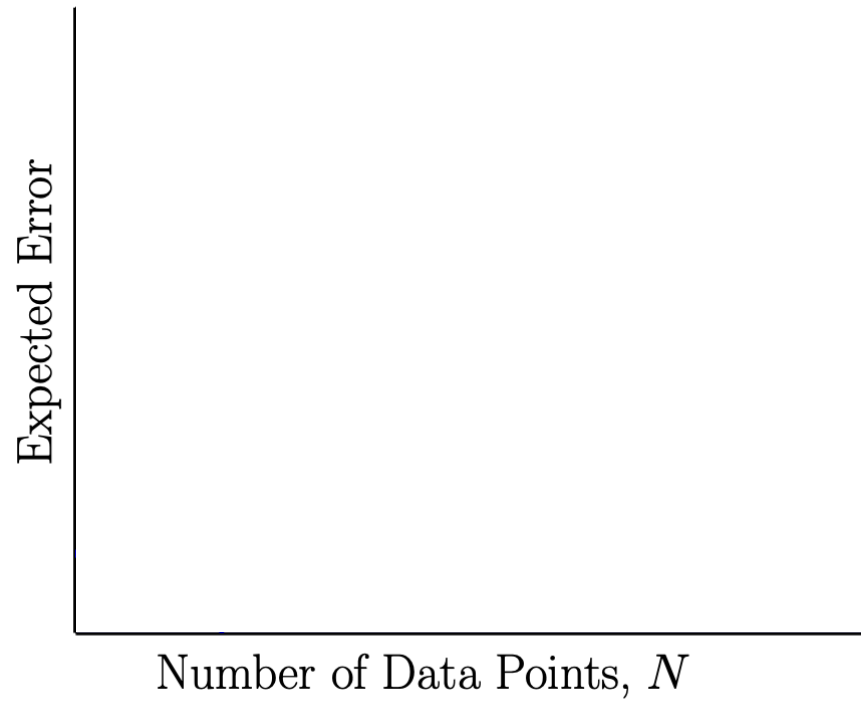


Complex Model

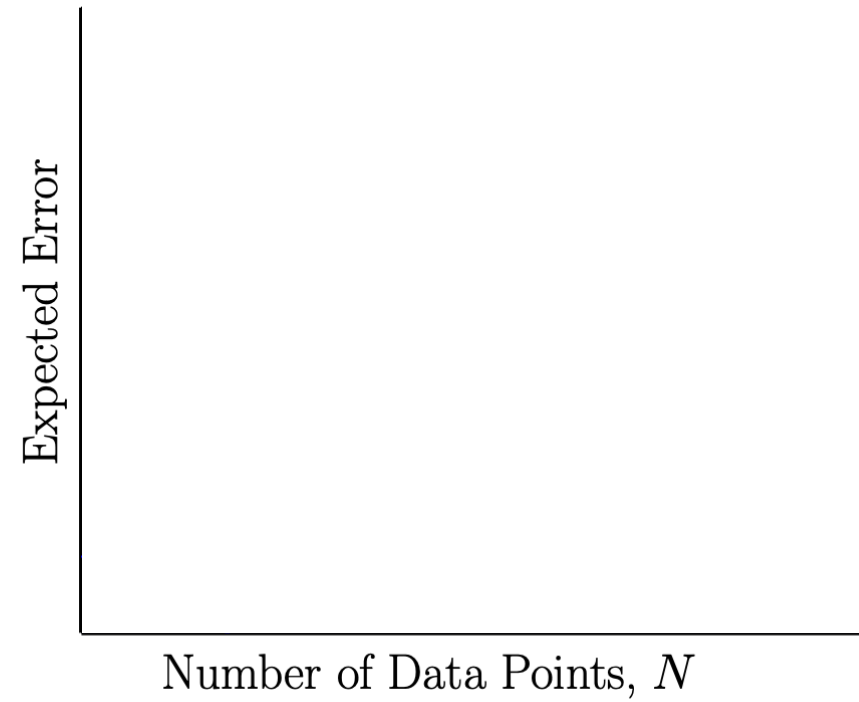


Learning Curves

VC Analysis

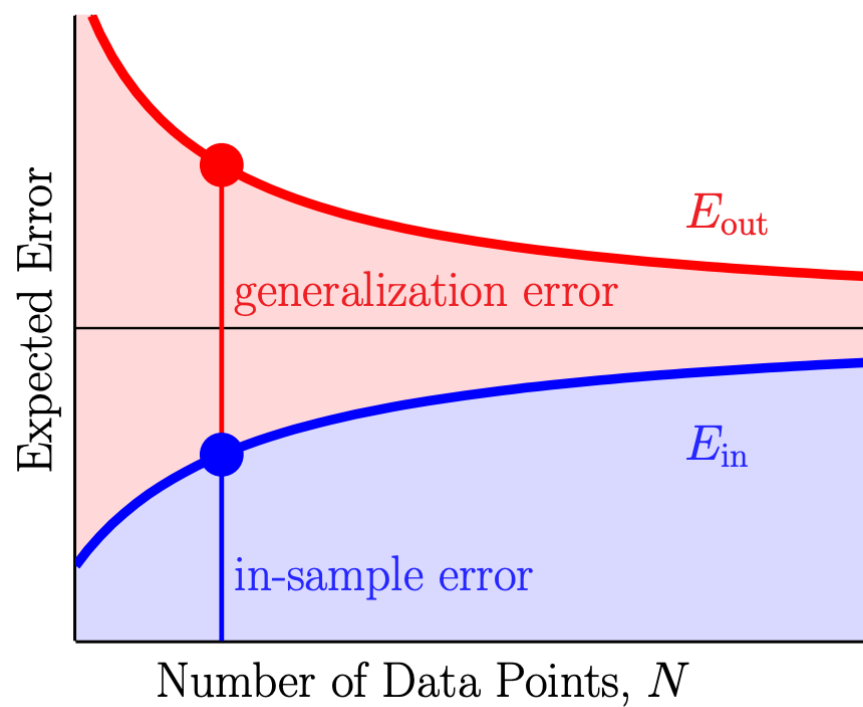


Bias-Variance Analysis

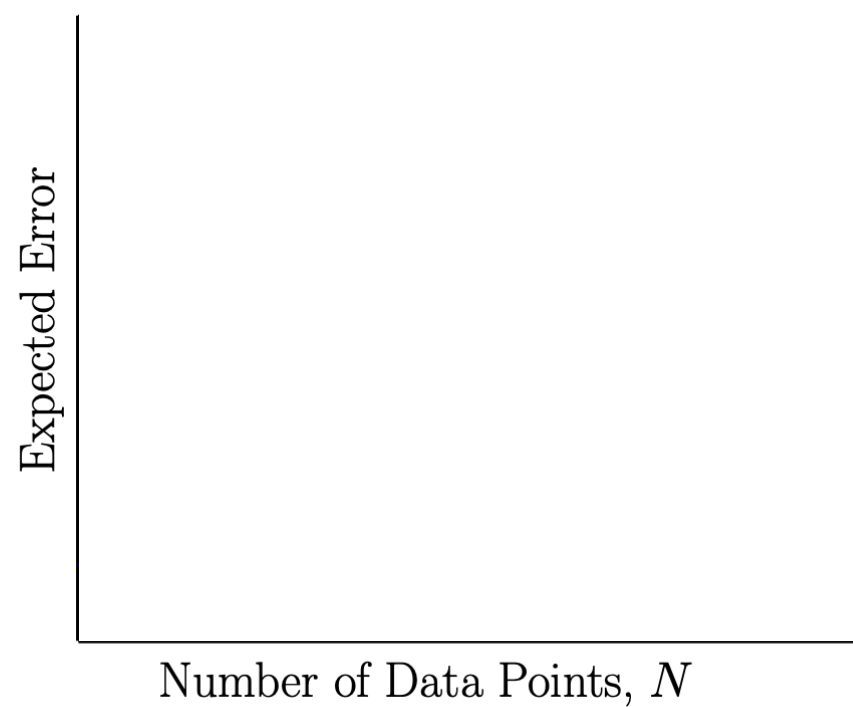


Learning Curves

VC Analysis

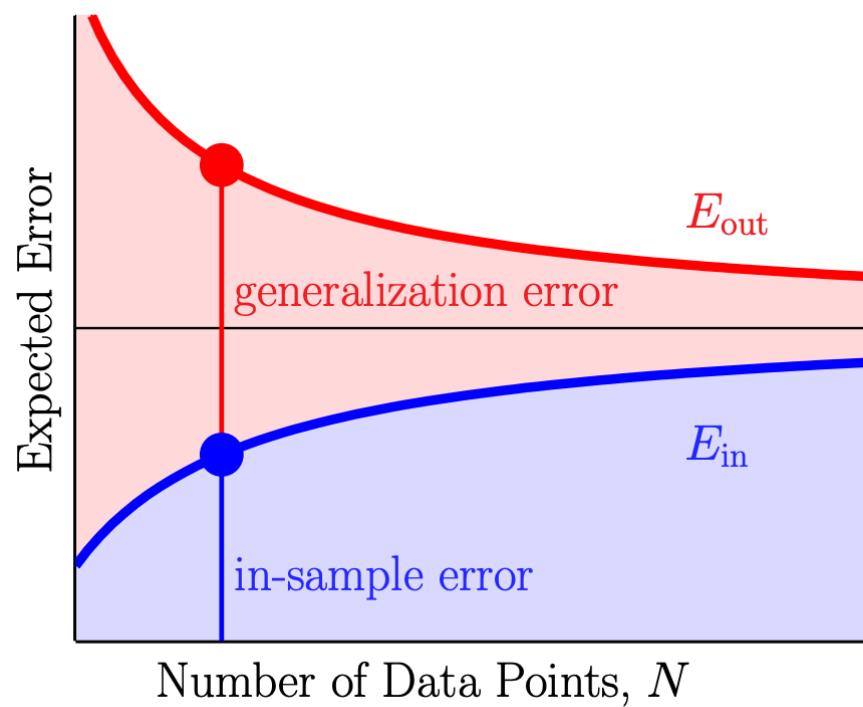


Bias-Variance Analysis

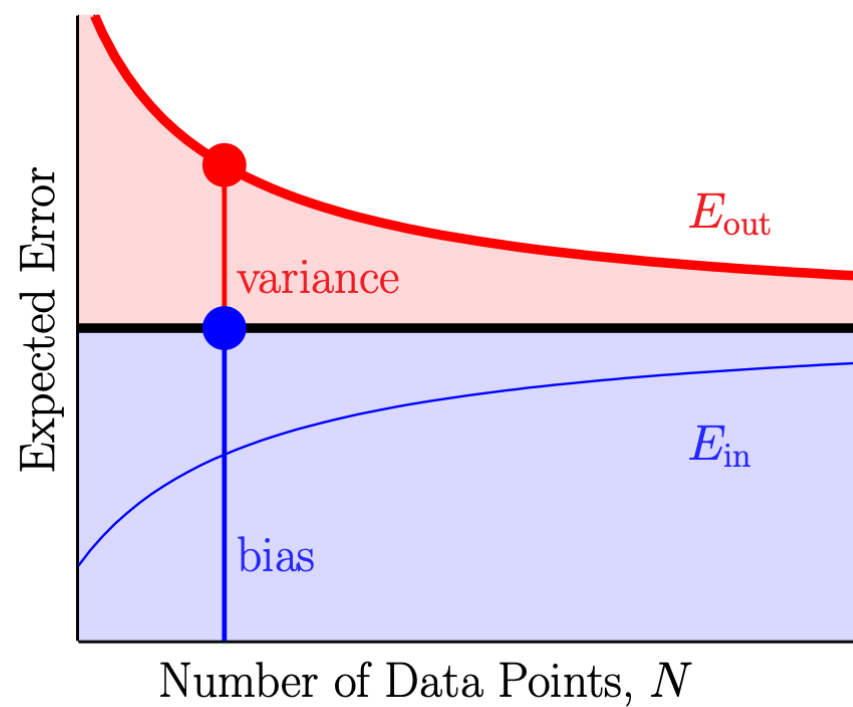


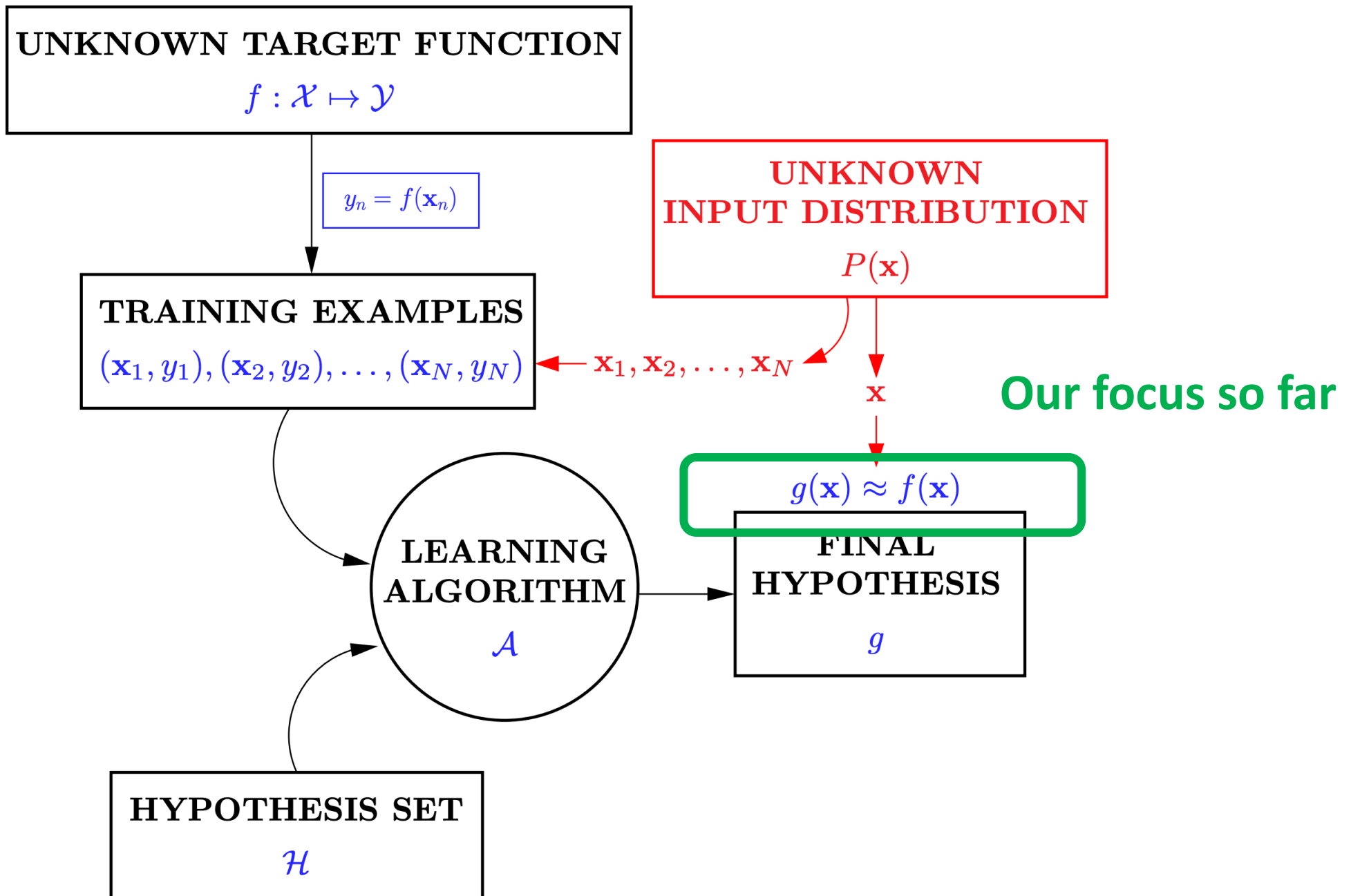
Learning Curves

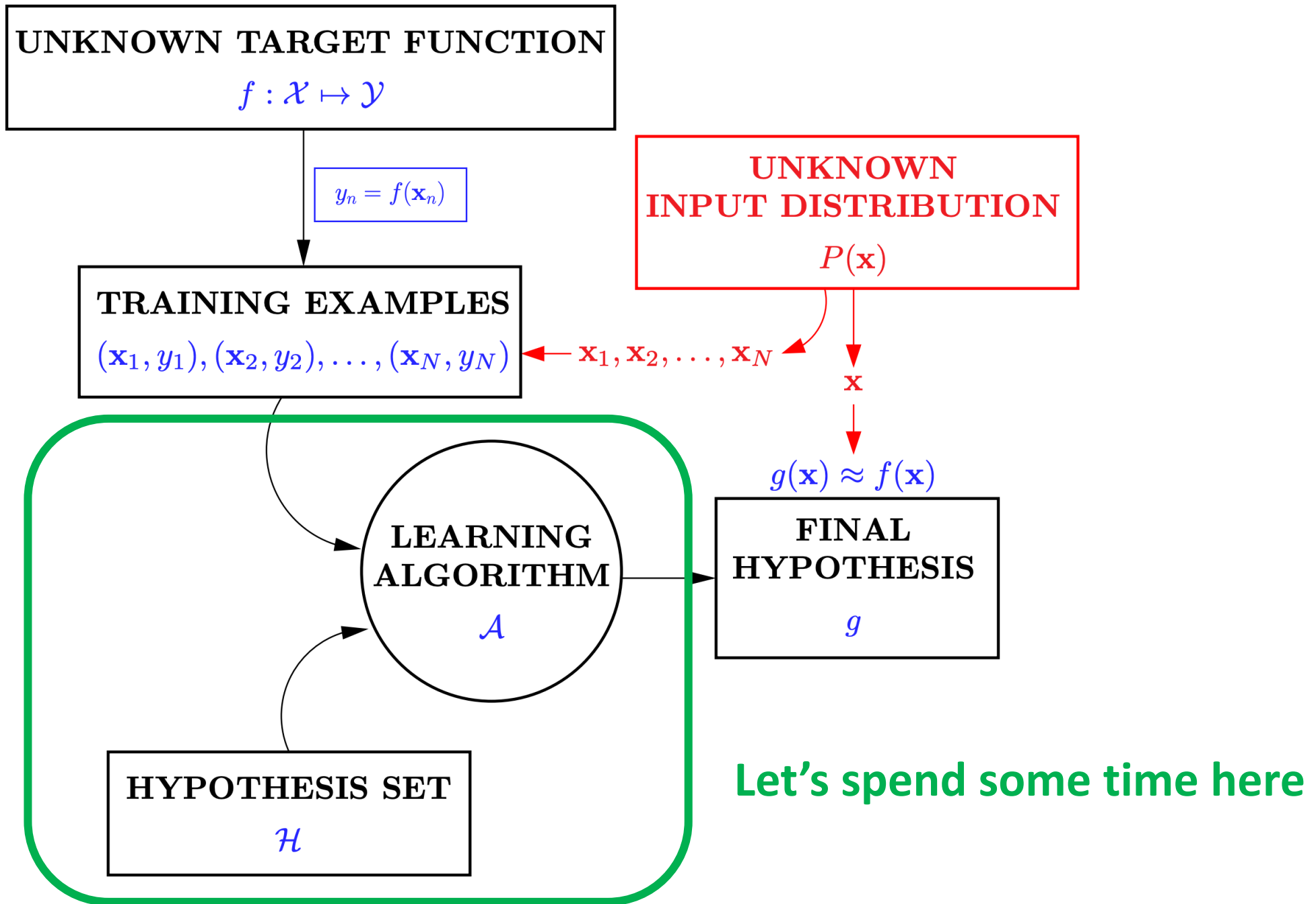
VC Analysis



Bias-Variance Analysis







Linear Models

Linear Models

This is why it's called linear models

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

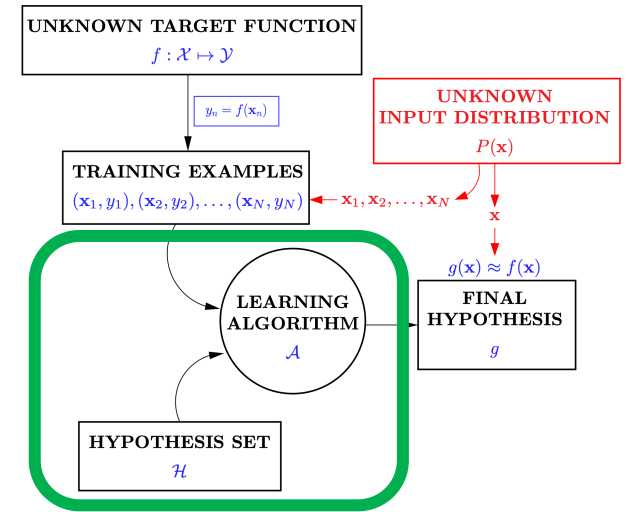
	Domain	Model	Credit Card Example
Linear Classification	$y \in \{-1, +1\}$	$H = \{h(\vec{x}) = \text{sign}(\vec{w}^T \vec{x})\}$	Approve or not
Linear Regression	$y \in \mathbb{R}$	$H = \{h(\vec{x}) = \vec{w}^T \vec{x}\}$	Credit line
Logistic Regression	$y \in [0,1]$	$H = \{h(\vec{x}) = \theta(\vec{w}^T \vec{x})\}$	Prob. of default

$$\theta(s) = \frac{e^s}{1 + e^s}$$

- Linear models:
 - Simple models => Good generalization error
- Reminder:
 - We will **interchangeably use** h and \vec{w} to represent a hypothesis in linear models

Learning Algorithm?

- Goal of the algorithm: Find $g \in H$ that minimizes $E_{out}(g)$
(We don't know E_{out})
- Common algorithms:
 - $g = \operatorname{argmin}_{h \in H} E_{in}(h)$
 - Works well when the model is simple (generalization error is small)
 - Will focus on this in the discussion of linear models
 - $g = \operatorname{argmin}_{h \in H} \{E_{in}(h) + \Omega(h)\}$
 - $\Omega(h)$: penalty for complex h
 - Will discuss this when we get to LFD Section 4



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

- **Optimization** is a key component in machine learning