Learning From Data
Lecture 7
Approximation Versus Generalization

The VC Dimension
Approximation Versus Generalization
Bias and Variance
The Learning Curve

M. Magdon-Ismail
CSCI 4100/6100

RECAP: The Vapnik-Chervonenkis Bound (VC Bound)

$$\mathbb{P}\left[|\boldsymbol{E}_{\text{in}}(\boldsymbol{g}) - \boldsymbol{E}_{\text{out}}(\boldsymbol{g})| > \epsilon\right] \leq 4\boldsymbol{m}_{\mathcal{H}}(2\boldsymbol{N})e^{-\epsilon^2 N/8}, \qquad \textbf{for any } \epsilon > 0.$$

$\mathbb{P}[|E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon] \leq 2|\mathcal{H}|e^{-2\epsilon^2 N}$ $\leftarrow$ finite $\mathcal{H}$

$$\mathbb{P}\left[|\boldsymbol{E}_{\text{in}}(\boldsymbol{g}) - \boldsymbol{E}_{\text{out}}(\boldsymbol{g})| \leq \epsilon\right] \geq 1 - 4\boldsymbol{m}_{\mathcal{H}}(2\boldsymbol{N})e^{-\epsilon^2 N/8}, \qquad \textbf{for any } \epsilon > 0.$$

$\mathbb{P}[|E_{\text{in}}(g) - E_{\text{out}}(g)| \leq \epsilon] \geq 1 - 2|\mathcal{H}|e^{-2\epsilon^2 N}$ $\leftarrow$ finite $\mathcal{H}$

$$\boldsymbol{E}_{\text{out}}(\boldsymbol{g}) \leq \boldsymbol{E}_{\text{in}}(\boldsymbol{g}) + \sqrt{\frac{8}{N}\log\frac{4\boldsymbol{m}_{\mathcal{H}}(2\boldsymbol{N})}{\boldsymbol{\delta}}}, \qquad \text{w.p. at least } \boldsymbol{1-\delta}.$$

$E_{\text{out}}(g) \leq E_{\text{in}}(g) + \sqrt{\frac{1}{2N}\log\frac{2|\mathcal{H}|}{\delta}}$ $\leftarrow$ finite $\mathcal{H}$

$$\boldsymbol{m}_{\mathcal{H}}(\boldsymbol{N}) \leq \sum_{\boldsymbol{i}=\boldsymbol{1}}^{\boldsymbol{k-1}} \binom{N}{i} \leq \boldsymbol{N}^{\boldsymbol{k-1}} + \boldsymbol{1} \qquad k \text{ is a break point.}$$

The VC Dimension $d_{\mathbf{vc}}$

$$m_{\mathcal{H}}(N) \sim N^{k-1}$$

The tightest bound is obtained with the smallest break point k^*.

Definition [VC Dimension] $d_{\text{VC}} = k^* - 1$.

The VC dimension is the largest N which can be shattered ($m_{\mathcal{H}}(N) = 2^N$).

$N \leq d_{\text{VC}}$: $\mathcal{H}$ *could* shatter your data ($\mathcal{H}$ can shatter some N points).

$N > d_{\text{VC}}$: N is a break point for $\mathcal{H}$; $\mathcal{H}$ cannot possibly shatter your data.

$$m_{\mathcal{H}}(N) \leq N^{d_{\text{VC}}} + 1 \sim N^{d_{\text{VC}}}$$

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

The VC-dimension is an Effective Number of Parameters

	N							
	1	2	3	4	5	$\cdots$	#Param	d_{VC}
2-D perceptron	2	4	8	14	$\cdots$		3	3
1-D pos. ray	2	3	4	5	$\cdots$		1	1
2-D pos. rectangles	2	4	8	16	$< 2^5$	$\cdots$	4	4
pos. convex sets	2	4	8	16	32	$\cdots$	∞	∞

There are models with few parameters but infinite d_{VC}.

There are models with redundant parameters but small d_{VC}.

VC-dimension of the Perceptron in $\mathbb{R}^d$ is $d+1$

This can be shown in two steps:

1. $d_{\text{VC}} \geq d+1$.

   What needs to be shown?

   (a) There is a set of $d+1$ points that can be shattered.
   (b) There is a set of $d+1$ points that cannot be shattered.
   (c) Every set of $d+1$ points can be shattered.
   (d) Every set of $d+1$ points cannot be shattered.

2. $d_{\text{VC}} \leq d+1$.

   What needs to be shown?

   (a) There is a set of $d+1$ points that can be shattered.
   (b) There is a set of $d+2$ points that cannot be shattered.
   (c) Every set of $d+2$ points can be shattered.
   (d) Every set of $d+1$ points cannot be shattered.
   (e) Every set of $d+2$ points cannot be shattered.

VC-dimension of the Perceptron in $\mathbb{R}^d$ is $d+1$

This can be shown in two steps:

1. $d_{\text{VC}} \geq d+1$.

   What needs to be shown?

   ✓ **(a)** There is a set of $d+1$ points that can be shattered.
   (b) There is a set of $d+1$ points that cannot be shattered.
   (c) Every set of $d+1$ points can be shattered.
   (d) Every set of $d+1$ points cannot be shattered.

2. $d_{\text{VC}} \leq d+1$.

   What needs to be shown?

   (a) There is a set of $d+1$ points that can be shattered.
   (b) There is a set of $d+2$ points that cannot be shattered.
   (c) Every set of $d+2$ points can be shattered.
   (d) Every set of $d+1$ points cannot be shattered.
   (e) Every set of $d+2$ points cannot be shattered.

VC-dimension of the Perceptron in $\mathbb{R}^d$ is $d+1$

This can be shown in two steps:

1. $d_{\text{VC}} \geq d+1$.

   What needs to be shown?

   - **(a)** There is a set of $d+1$ points that can be shattered. ✓
   - **(b)** There is a set of $d+1$ points that cannot be shattered.
   - **(c)** Every set of $d+1$ points can be shattered.
   - **(d)** Every set of $d+1$ points cannot be shattered.

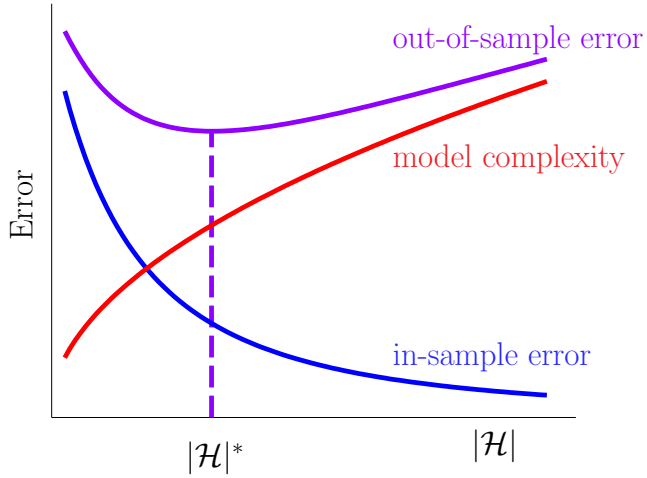
2. $d_{\text{VC}} \leq d+1$.

   What needs to be shown?

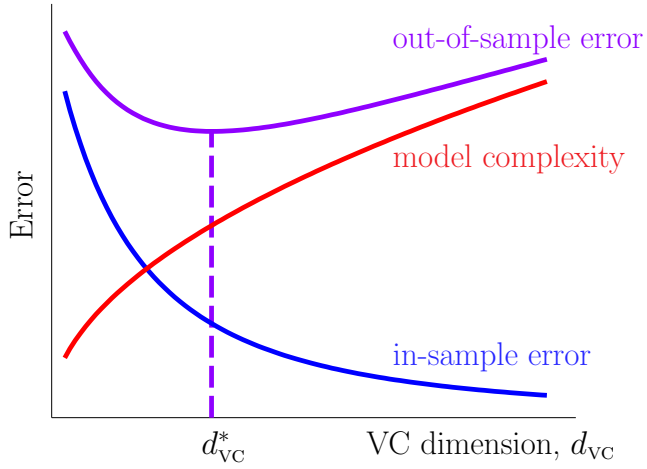
   - **(a)** There is a set of $d+1$ points that can be shattered.
   - **(b)** There is a set of $d+2$ points that cannot be shattered.
   - **(c)** Every set of $d+2$ points can be shattered.
   - **(d)** Every set of $d+1$ points cannot be shattered.
   - **(e)** Every set of $d+2$ points cannot be shattered. ✓

A Single Parameter Characterizes Complexity

$$E_{\text{out}}(g) \leq E_{\text{in}}(g) + \sqrt{\frac{1}{2N}\log\frac{2|\mathcal{H}|}{\delta}}$$



$$E_{\text{out}}(g) \leq E_{\text{in}}(g) + \underbrace{\sqrt{\frac{8}{N}\log\frac{4((2N)^{d_{\text{vc}}}+1)}{\delta}}}_{\text{penalty for model complexity } \Omega(d_{\text{vc}})}$$



Sample Complexity: How Many Data Points Do You Need?

Set the error bar at ϵ.

$$\epsilon = \sqrt{\frac{8}{N} \ln \frac{4((2N)^{d_{\text{VC}}} + 1)}{\delta}}$$

Solve for N:

$$N = \frac{8}{\epsilon^2} \ln \frac{4((2N)^{d_{\text{VC}}} + 1)}{\delta} = O\left(d_{\text{VC}} \ln N\right)$$

Example. $d_{\text{VC}} = 3$; error bar $\epsilon = 0.1$; confidence 90% ($\delta = 0.1$).
A simple iterative method works well. Trying $N = 1000$ we get

$$N \approx \frac{1}{0.1^2} \log \left(\frac{4(2000)^3 + 4}{0.1} \right) \approx 21192.$$

We continue iteratively, and converge to $N \approx 30000$.
If $d_{\text{VC}} = 4$, $N \approx 40000$; for $d_{\text{VC}} = 5$, $N \approx 50000$.

($N \propto d_{\text{VC}}$, but gross overestimates)

Practical Rule of Thumb: $N = 10 \times d_{\text{VC}}$

Theory Versus Practice

The VC analysis allows us to reach outside the data for general $\mathcal{H}$.

- a single parameter characterizes complexity of $\mathcal{H}$
- d_{VC} depends only on $\mathcal{H}$.
- E_{in} can reach outside $\mathcal{D}$ to E_{out} when d_{VC} is finite.

In Practice . . .

- The VC bound is loose.
  - Hoeffding;
  - $m_{\mathcal{H}}(N)$ is a worst case # of dichotomies, not average case or likely case.
  - The polynomial bound on $m_{\mathcal{H}}(N)$ is loose.
- It is a good guide – models with small d_{VC} are good.
- Roughly $10 \times d_{\text{VC}}$ examples needed to get good generalization.

The Test Set

- Another way to estimate $E_{\text{out}}(g)$ is using a *test set* to obtain $E_{\text{test}}(g)$.

- E_{test} is better than E_{in}: you don't pay the price for fitting.

  You can use $|\mathcal{H}| = 1$ in the Hoeffding bound with E_{test}.

- Both a test and training set have variance.

  The training set has *optimistic bias* due to selection – fitting the data.

  A test set has no bias.

- The price for a test set is fewer training examples. (why is this bad?)

  $E_{\text{test}} \approx E_{\text{out}}$ but now E_{test} may be bad.

VC Bound Quantifies Approximation Versus Generalization

The best $\mathcal{H}$ is $\mathcal{H} = \{f\}$.

You are better off buying a lottery ticket.

$d_{\text{VC}} \uparrow \implies$ better chance of **approximating** f ($E_{\text{in}} \approx 0$).

$d_{\text{VC}} \downarrow \implies$ better chance of **generalizing** to out of sample ($E_{\text{in}} \approx E_{\text{out}}$).

$$E_{\text{out}} \leq E_{\text{in}} + \Omega(d_{\text{VC}}).$$

VC analysis only depends on $\mathcal{H}$.

Independent of $f, P(\mathbf{x})$, learning algorithm.

Bias-Variance Analysis

Another way to quantify the tradeoff:

1. How well *can* the learning approximate f.
   ... as opposed to how well *did* the learning approximate f in-sample (E_{in}).
2. How close can you get to that approximation with a finite data set.
   ... as opposed to how close is E_{in} to E_{out}.

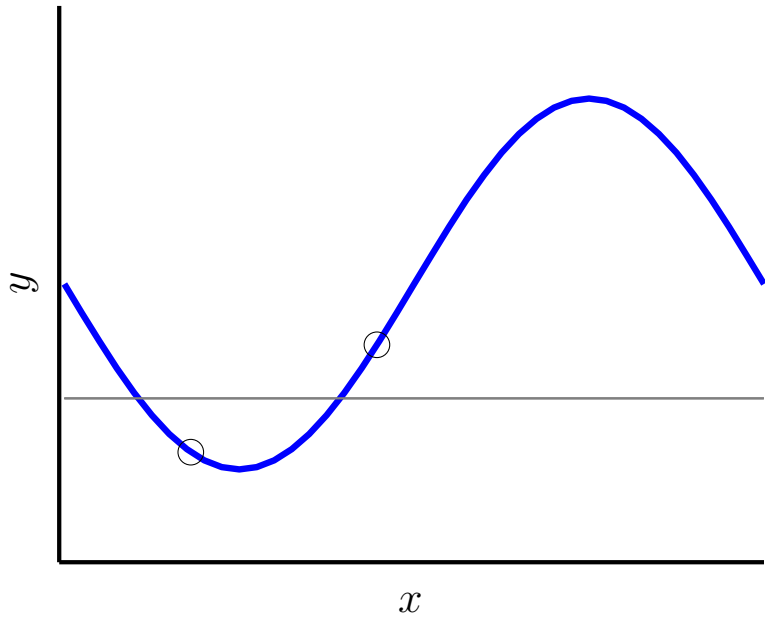
Bias-variance analysis applies to squared errors (classification and regression)

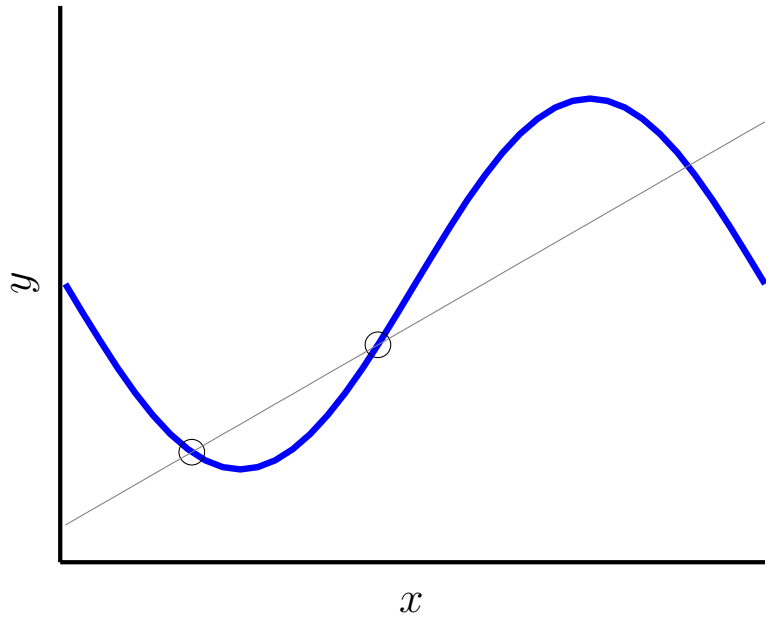
Bias-variance analysis can take into account the *learning algorithm*
Different learning algorithms can have different E_{out} when applied to the same $\mathcal{H}$!

A Simple Learning Problem

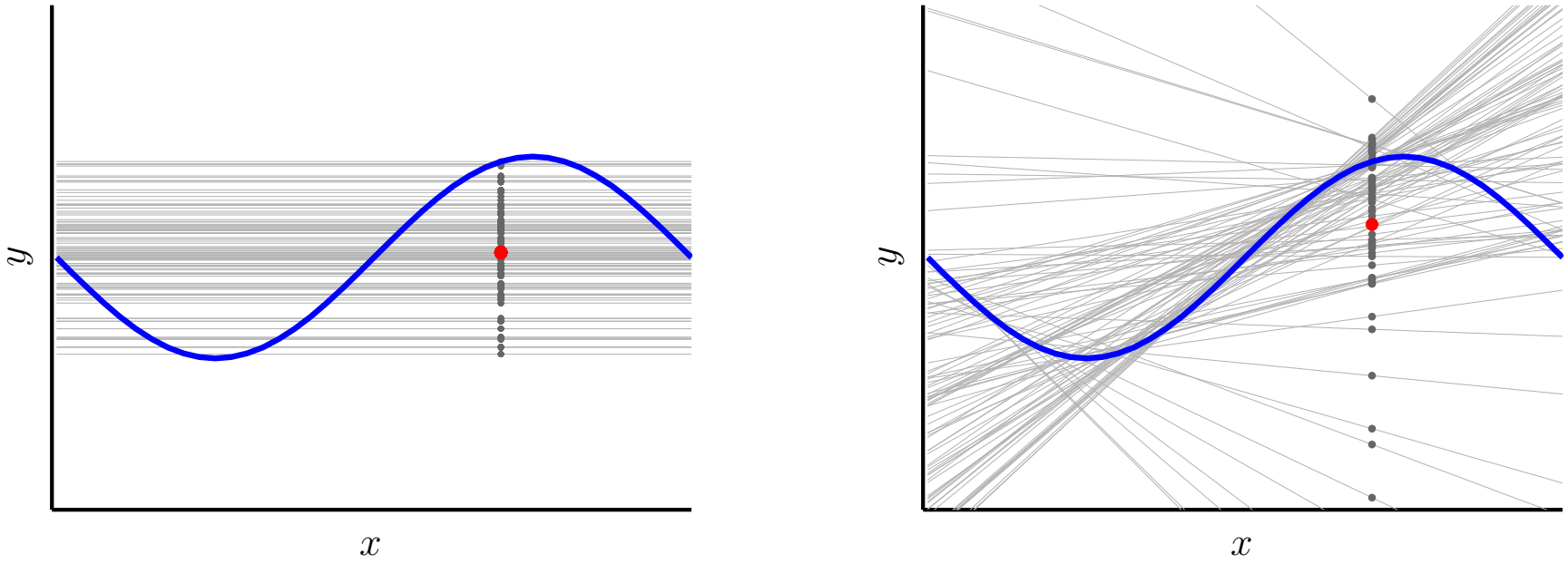
2 Data Points. 2 hypothesis sets:

$$\mathcal{H}_0: \quad h(x) = b$$
$$\mathcal{H}_1: \quad h(x) = ax + b$$





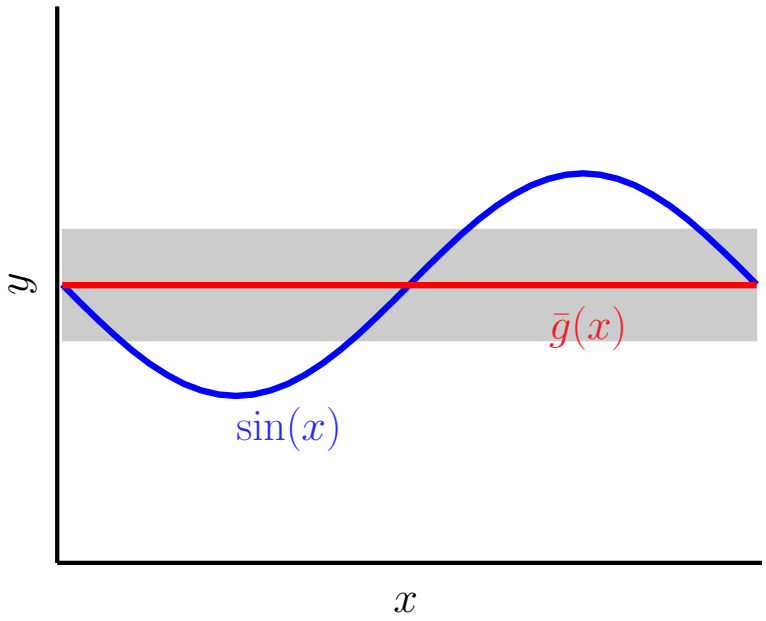
Let's Repeat the Experiment Many Times

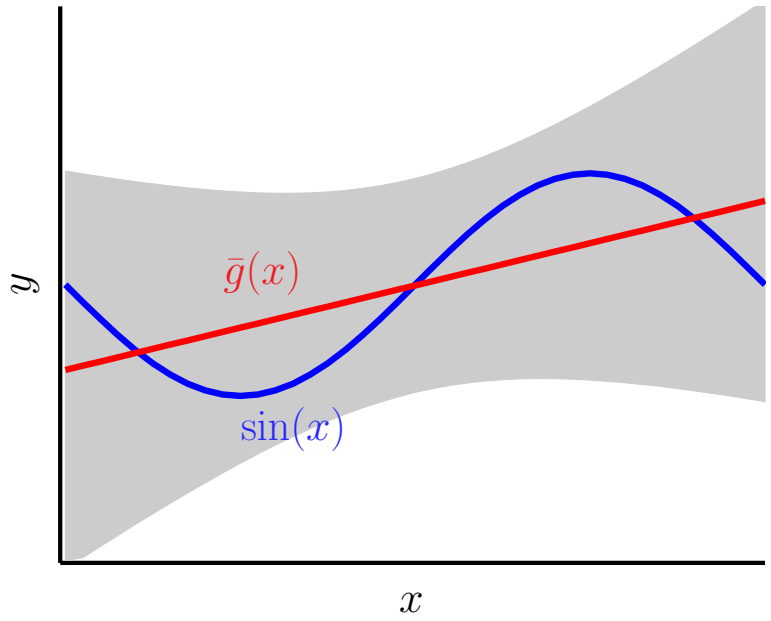


For each data set $\mathcal{D}$, you get a different $g^{\mathcal{D}}$.

So, for a fixed $\mathbf{x}$, $g^{\mathcal{D}}(\mathbf{x})$ is random value, depending on $\mathcal{D}$.

What's Happening on Average





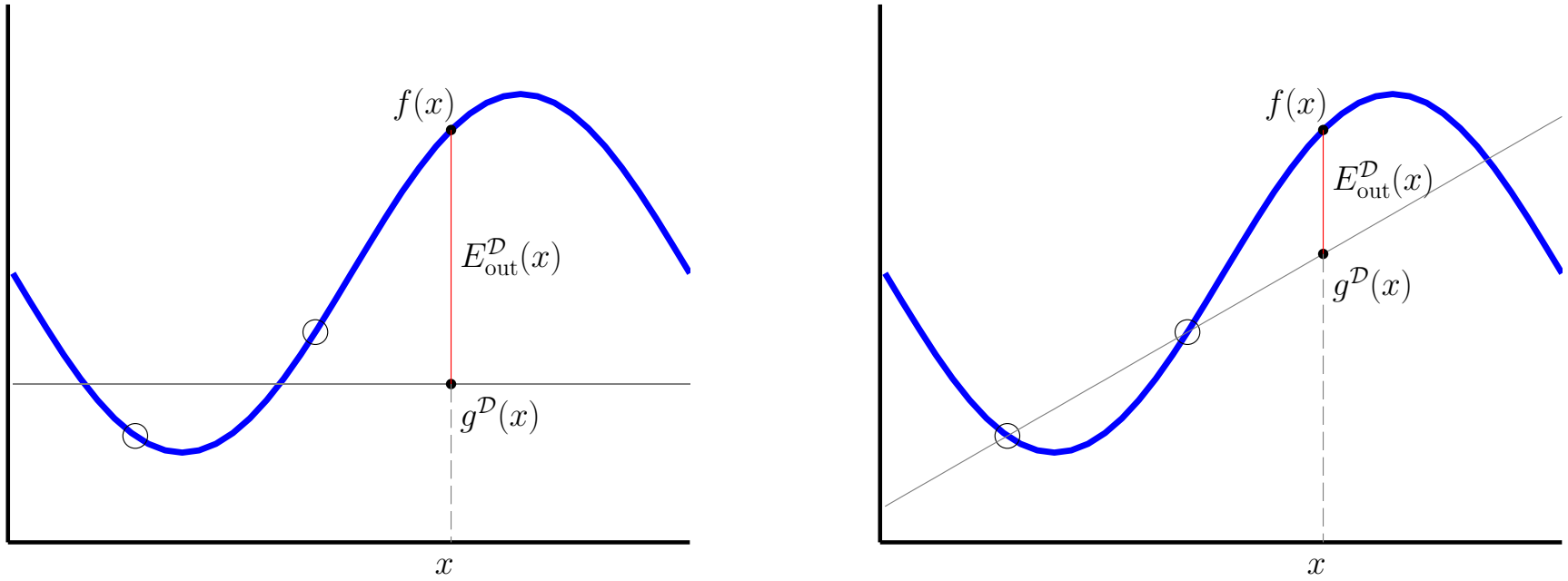
We can define:

$$g^{\mathcal{D}}(\mathbf{x}) \quad \leftarrow \textbf{random value}, \text{ depending on } \mathcal{D}$$

$$\begin{aligned} \bar{g}(\mathbf{x}) &= \mathbb{E}_{\mathcal{D}}\left[g^{\mathcal{D}}(\mathbf{x})\right] \\ &\approx \tfrac{1}{K}(g^{\mathcal{D}_1}(\mathbf{x}) + \cdots + g^{\mathcal{D}_K}(\mathbf{x})) \end{aligned} \quad \leftarrow \text{your average prediction on } \mathbf{x}$$

$$\begin{aligned} \mathsf{var}(\mathbf{x}) &= \mathbb{E}_{\mathcal{D}}\left[(g^{\mathcal{D}}(\mathbf{x}) - \bar{g}(\mathbf{x}))^2\right] \\ &= \mathbb{E}_{\mathcal{D}}\left[g^{\mathcal{D}}(\mathbf{x})^2\right] - \bar{g}(\mathbf{x})^2 \end{aligned} \quad \leftarrow \text{how variable is your prediction?}$$

E_{out} on Test Point x for Data $\mathcal{D}$



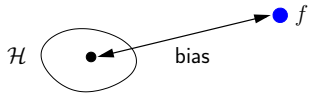
$$E_{\text{out}}^{\mathcal{D}}(\mathbf{x}) = (g^{\mathcal{D}}(\mathbf{x}) - f(\mathbf{x}))^2 \qquad \leftarrow \textbf{squared error}, \text{ a random value depending on } \mathcal{D}$$

$$E_{\text{out}}(\mathbf{x}) = \mathbb{E}_{\mathcal{D}}\left[E_{\text{out}}^{\mathcal{D}}(\mathbf{x})\right] \qquad \leftarrow \text{expected } E_{\text{out}}(\mathbf{x}) \text{ before seeing } \mathcal{D}$$

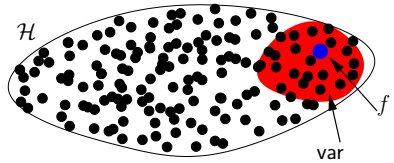
The Bias-Variance Decomposition

$$
\begin{aligned}
E_{\text{out}}(\mathbf{x}) &= \mathbb{E}_{\mathcal{D}}\left[(g^{\mathcal{D}}(\mathbf{x}) - f(\mathbf{x}))^2\right] \\
&= \mathbb{E}_{\mathcal{D}}\left[g^{\mathcal{D}}(\mathbf{x})^2 - 2g^{\mathcal{D}}(\mathbf{x})f(\mathbf{x}) + f(\mathbf{x})^2\right] \\
&= \mathbb{E}_{\mathcal{D}}\left[g^{\mathcal{D}}(\mathbf{x})^2\right] - 2\bar{g}(\mathbf{x})f(\mathbf{x}) + f(\mathbf{x})^2 \qquad \leftarrow \text{understand this; the rest is just algebra} \\
&= \mathbb{E}_{\mathcal{D}}\left[g^{\mathcal{D}}(\mathbf{x})^2\right] - \bar{g}(\mathbf{x})^2 + \bar{g}(\mathbf{x})^2 - 2\bar{g}(\mathbf{x})f(\mathbf{x}) + f(\mathbf{x})^2 \\
&= \underbrace{\mathbb{E}_{\mathcal{D}}\left[g^{\mathcal{D}}(\mathbf{x})^2\right] - \bar{g}(\mathbf{x})^2}_{\mathsf{var}(\mathbf{x})} + \underbrace{(\bar{g}(\mathbf{x}) - f(\mathbf{x}))^2}_{\mathsf{bias}(\mathbf{x})}
\end{aligned}
$$

$$E_{\text{out}}(\mathbf{x}) = \mathsf{bias}(\mathbf{x}) + \mathsf{var}(\mathbf{x})$$



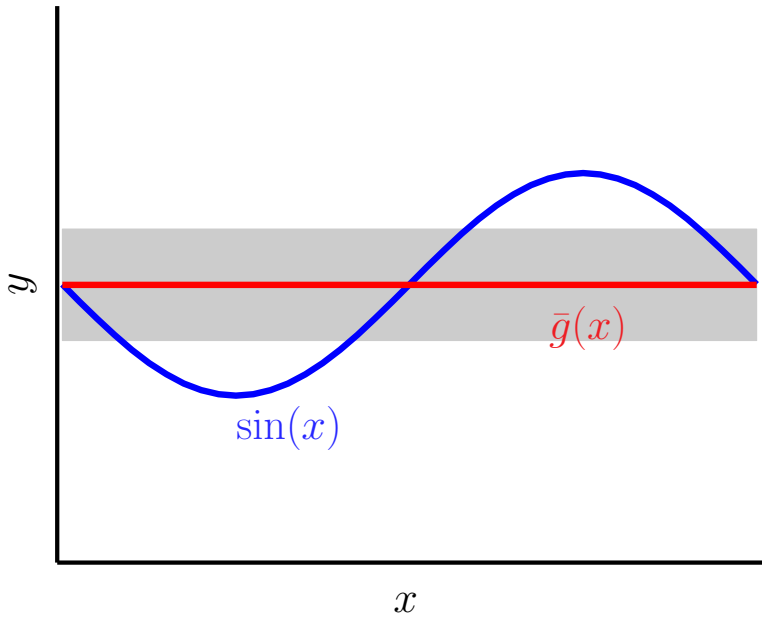
Very small model

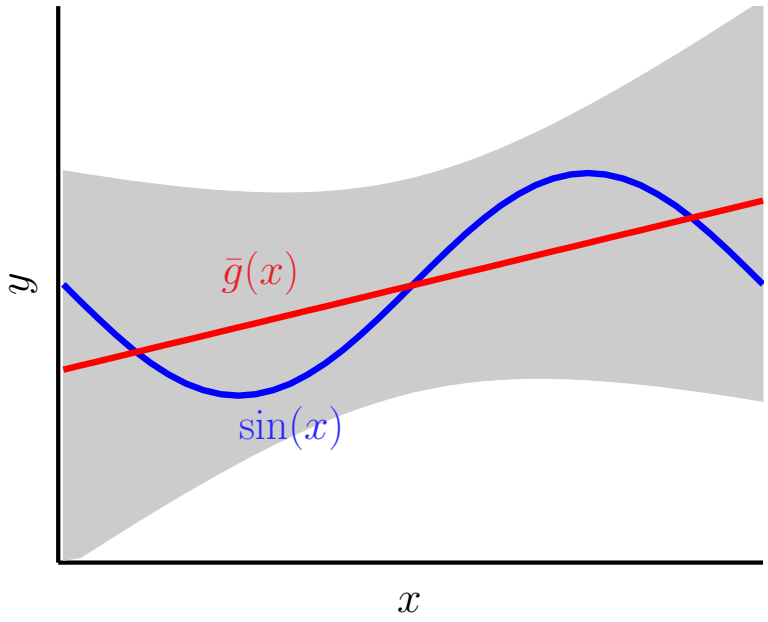


Very large model

If you take average over $\mathbf{x}$: $E_{\text{out}} = \mathsf{bias} + \mathsf{var}$

Back to $\mathcal{H}_0$ and $\mathcal{H}_1$; and, our winner is . . .





$\mathcal{H}_0$

$\text{bias} = 0.50$

$\text{var} = 0.25$

$E_{\text{out}} = 0.75$ ✓

$\mathcal{H}_1$

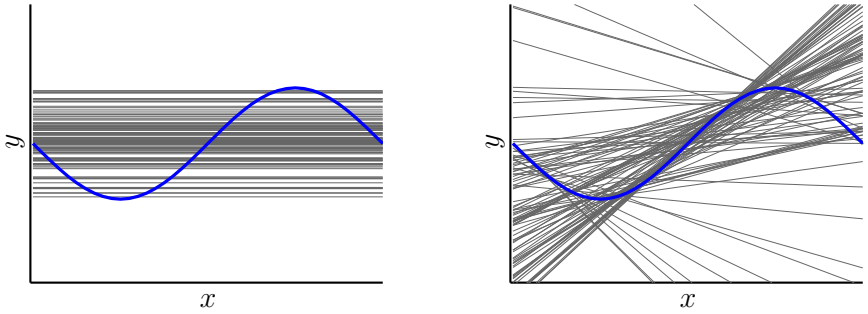
$\text{bias} = 0.21$

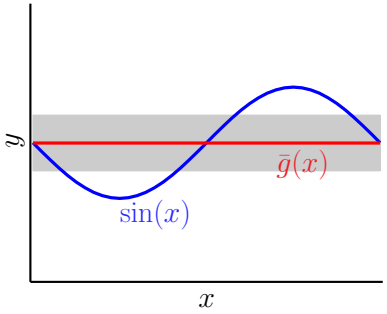
$\text{var} = 1.69$

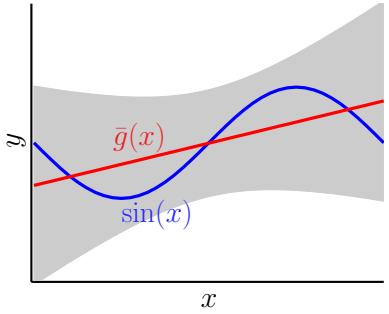
$E_{\text{out}} = 1.90$

Match Learning Power to Data, . . . Not to f

2 Data Points







$\mathcal{H}_0$

$\mathsf{bias} = 0.50;$

$\mathsf{var} = 0.25.$

$E_{\text{out}} = 0.75$ ✓

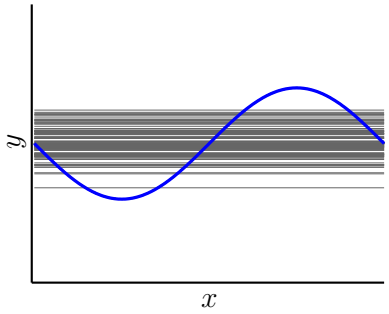
$\mathcal{H}_1$

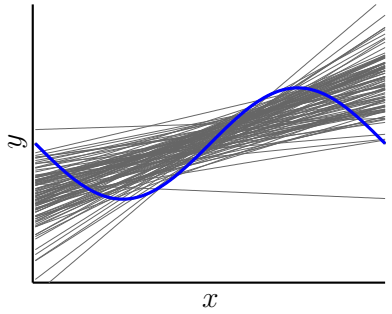
$\mathsf{bias} = 0.21;$

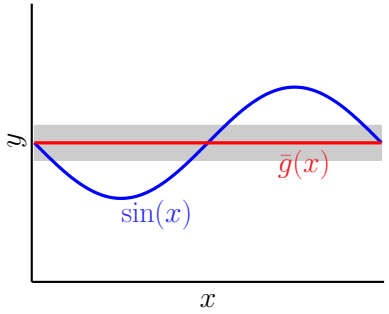
$\mathsf{var} = 1.69.$

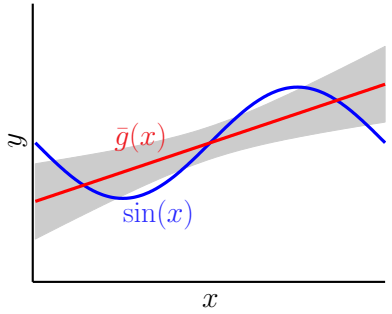
$E_{\text{out}} = 1.90$

5 Data Points









$\mathcal{H}_0$

$\mathsf{bias} = 0.50;$

$\mathsf{var} = 0.1.$

$E_{\text{out}} = 0.6$

$\mathcal{H}_1$

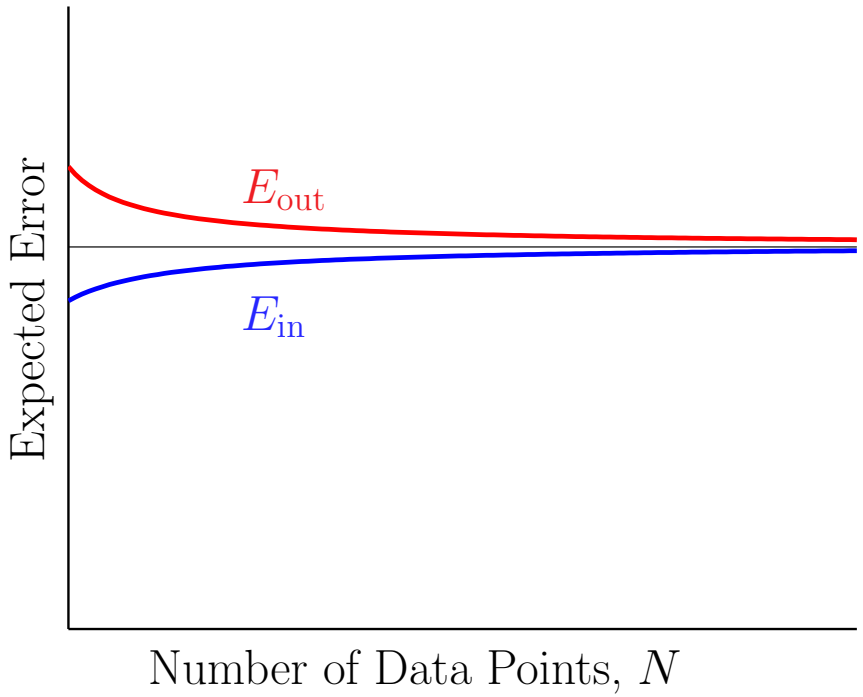
$\mathsf{bias} = 0.21;$

$\mathsf{var} = 0.21.$

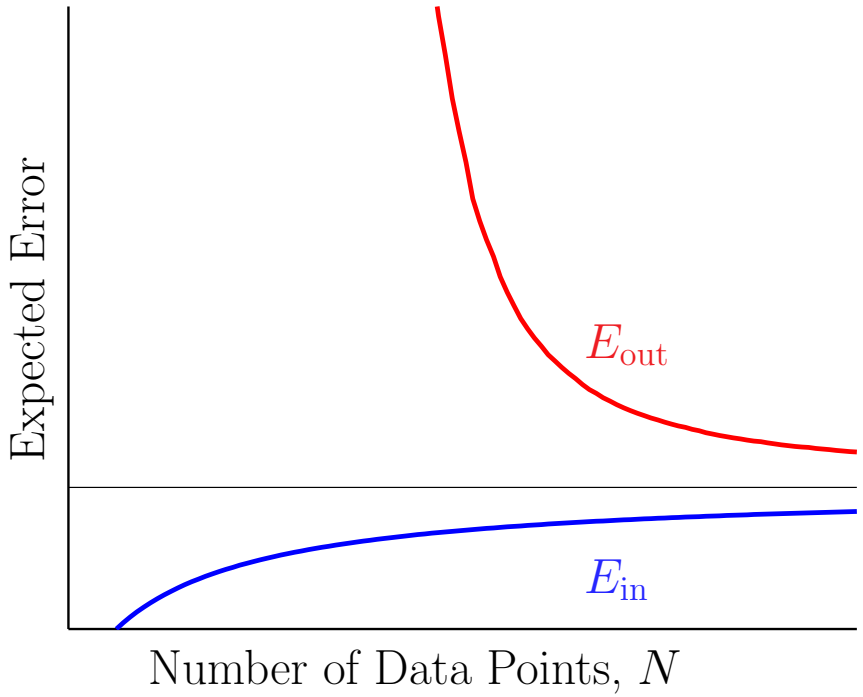
$E_{\text{out}} = 0.42$ ✓

Learning Curves: When Does the Balance Tip?

Simple Model



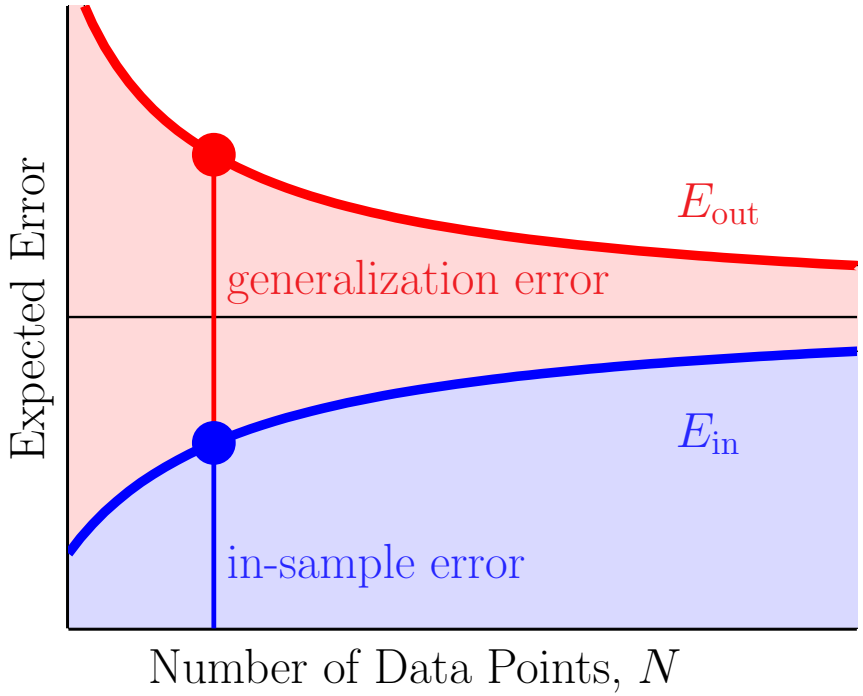
Complex Model



$$E_{\text{out}} = \mathbb{E}_{\mathbf{x}}\left[E_{\text{out}}(\mathbf{x})\right]$$

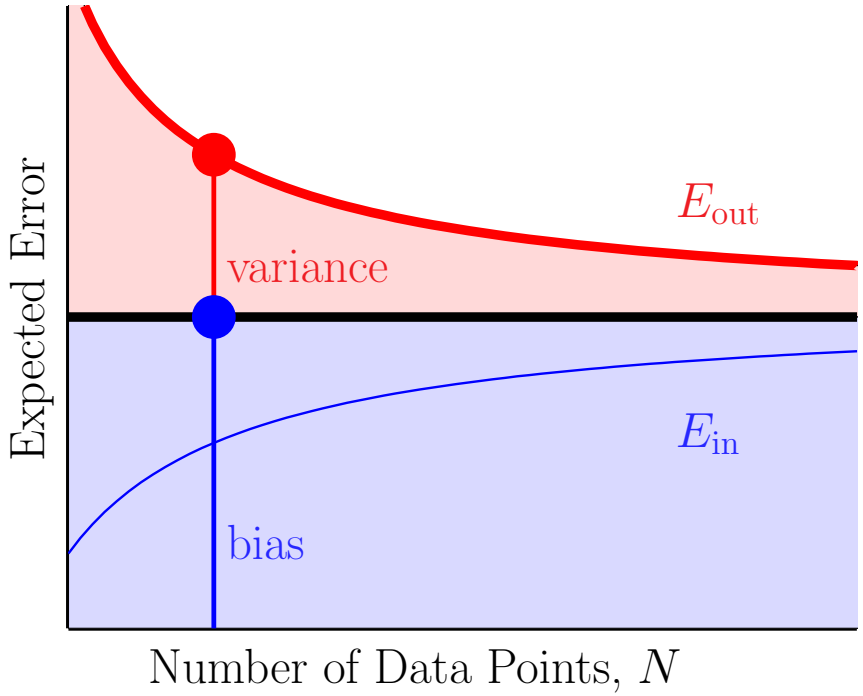
Decomposing The Learning Curve

VC Analysis



Pick $\mathcal{H}$ that can generalize and has a good chance to fit the data

Bias-Variance Analysis



Pick $(\mathcal{H}, \mathcal{A})$ to approximate f and not behave wildly after seeing the data