Artificial Intelligence

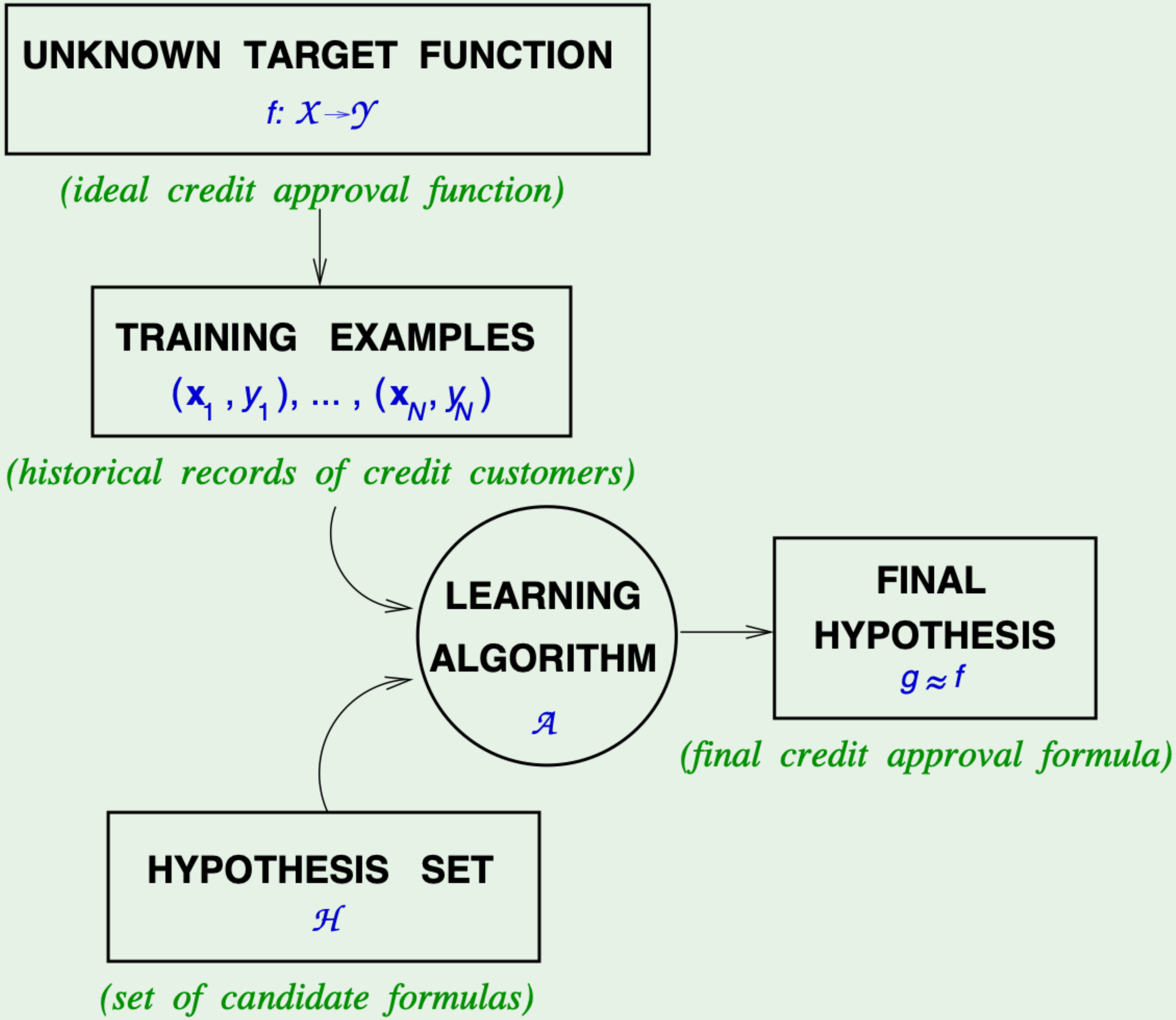
CSC 665

tyler dae devlin

Machine Learning IV

11.7.2023

- **Search:** make decisions by looking ahead
- **Logic:** deduce new facts from existing facts
- **Constraints:** find a way to satisfy a given specification
- **Probability:** reason quantitatively about uncertainty
- **Learning:** make future predictions from past observations

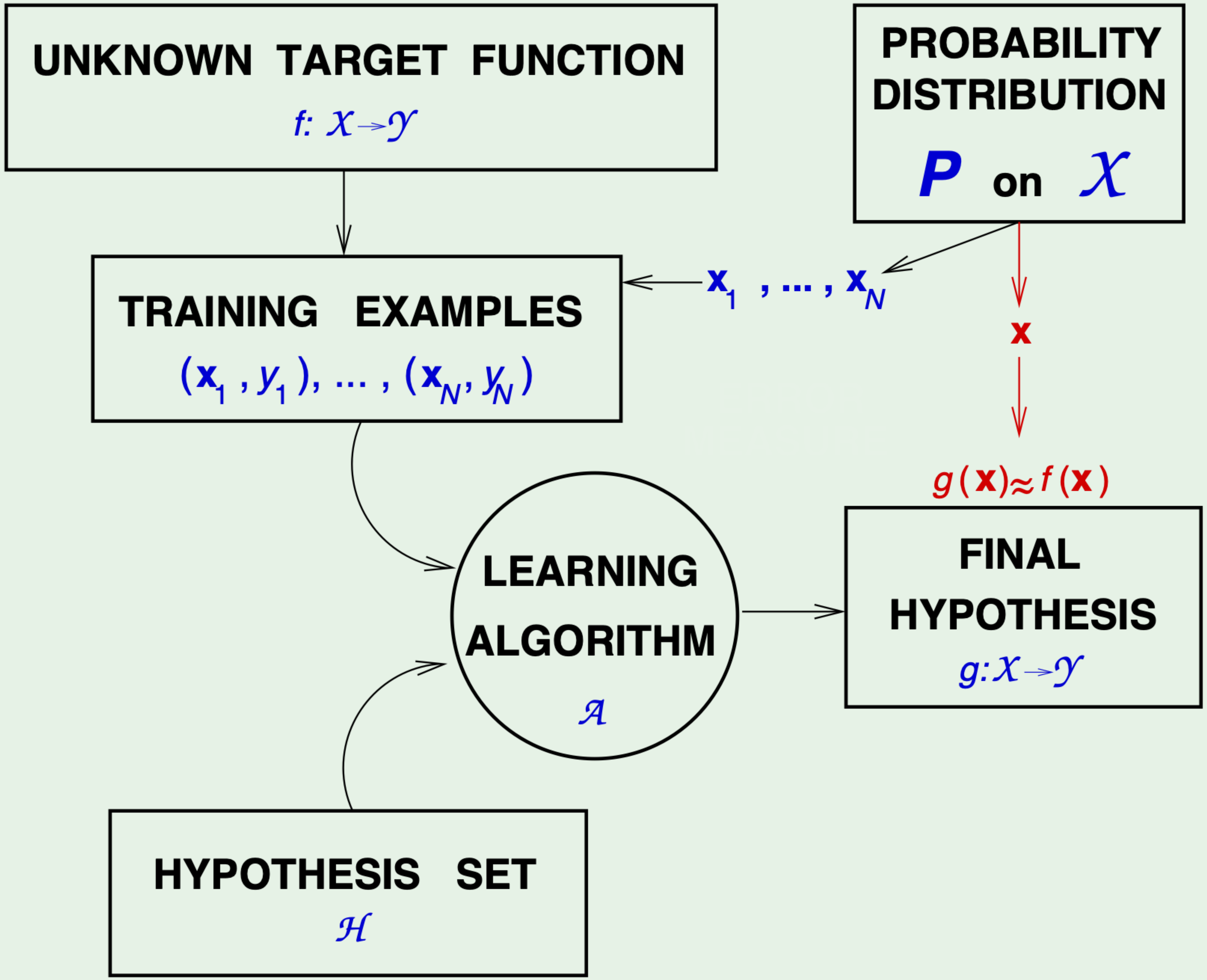

UNKNOWN TARGET FUNCTION
$f: \mathcal{X} \to \mathcal{Y}$
(ideal credit approval function)
TRAINING EXAMPLES
$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$
(historical records of credit customers)
LEARNING ALGORITHM
$\mathcal{A}$
FINAL HYPOTHESIS
$g \approx f$
(final credit approval formula)
HYPOTHESIS SET
$\mathcal{H}$
(set of candidate formulas)

[gradient descent on board]

Generalization

Beyond the training data

- ERM tells us how to pick a good hypothesis to fit a training dataset
- But fitting the training data is not the real goal
- Want a hypothesis h that approximates the target function f on *new unseen examples*
- I.e., want an h that **generalizes**
- This is only possible if the training data is **representative** of examples we are likely to see in the future

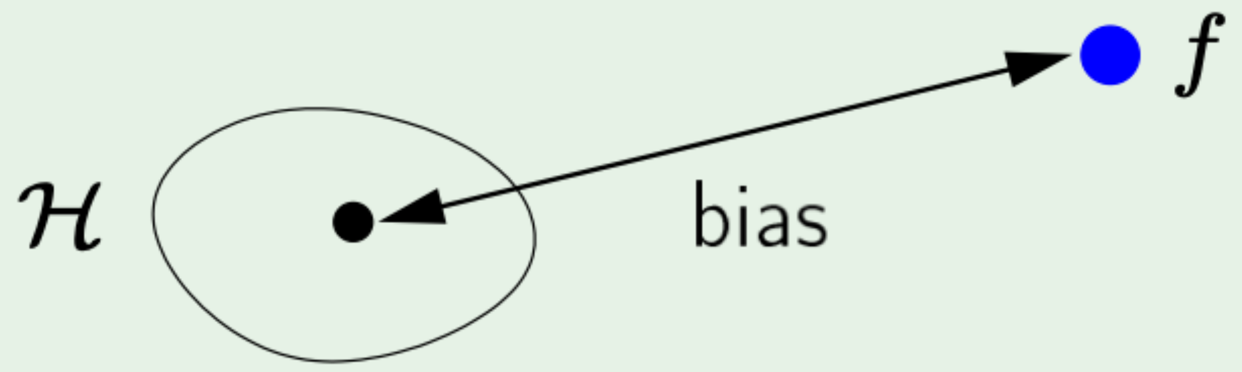

UNKNOWN TARGET FUNCTION
$f: \mathcal{X} \to \mathcal{Y}$
PROBABILITY DISTRIBUTION
P on $\mathcal{X}$
$\mathbf{x}_1, \ldots, \mathbf{x}_N$
$\mathbf{x}$
TRAINING EXAMPLES
$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$
$g(\mathbf{x}) \approx f(\mathbf{x})$
LEARNING ALGORITHM
$\mathcal{A}$
FINAL HYPOTHESIS
$g: \mathcal{X} \to \mathcal{Y}$
HYPOTHESIS SET
$\mathcal{H}$

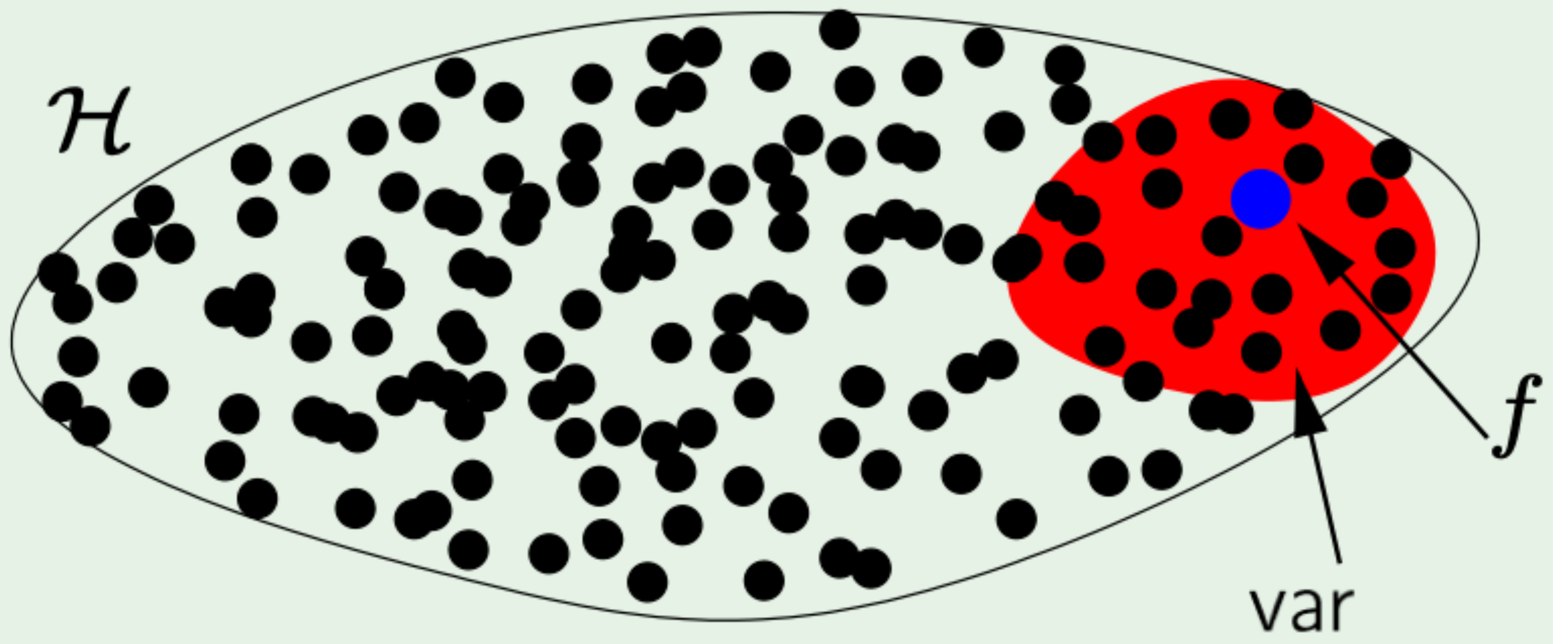
Evaluation

- Testing for generalization is straightforward
- Partition dataset into two groups: **training** dataset and **testing** dataset
- Use only training set to pick an $h \in \mathscr{H}$
- Once you've selected a candidate h, use testing set to obtain an unbiased estimate of performance (cost or error)
- Often, training error will be lower than testing error
- But if the gap is small, you have good generalization
- Good generalization is the central goal of machine learning

Approximation-generalization tradeoff

- Goal is low testing error C_{test}
- Can decompose test error into
  1. **Bias:** how well $\mathcal{H}$ can approximate f
  2. **Variance:** wow well we can zoom in on a good $h \in \mathcal{H}$
- Usually when $\mathcal{H}$ is more complex, (1) is easier but (2) is harder
- I.e., a more complex $\mathcal{H}$ has lower bias, but higher variance
- It's possible to make this decomposition mathematically precise





$\downarrow$ $\mathcal{H} \uparrow$ $\uparrow$

Example: sine target

$f : [-1, 1] \rightarrow \mathbb{R} \qquad f(x) = \sin(\pi x)$

Only two training examples! $\quad N = 2$

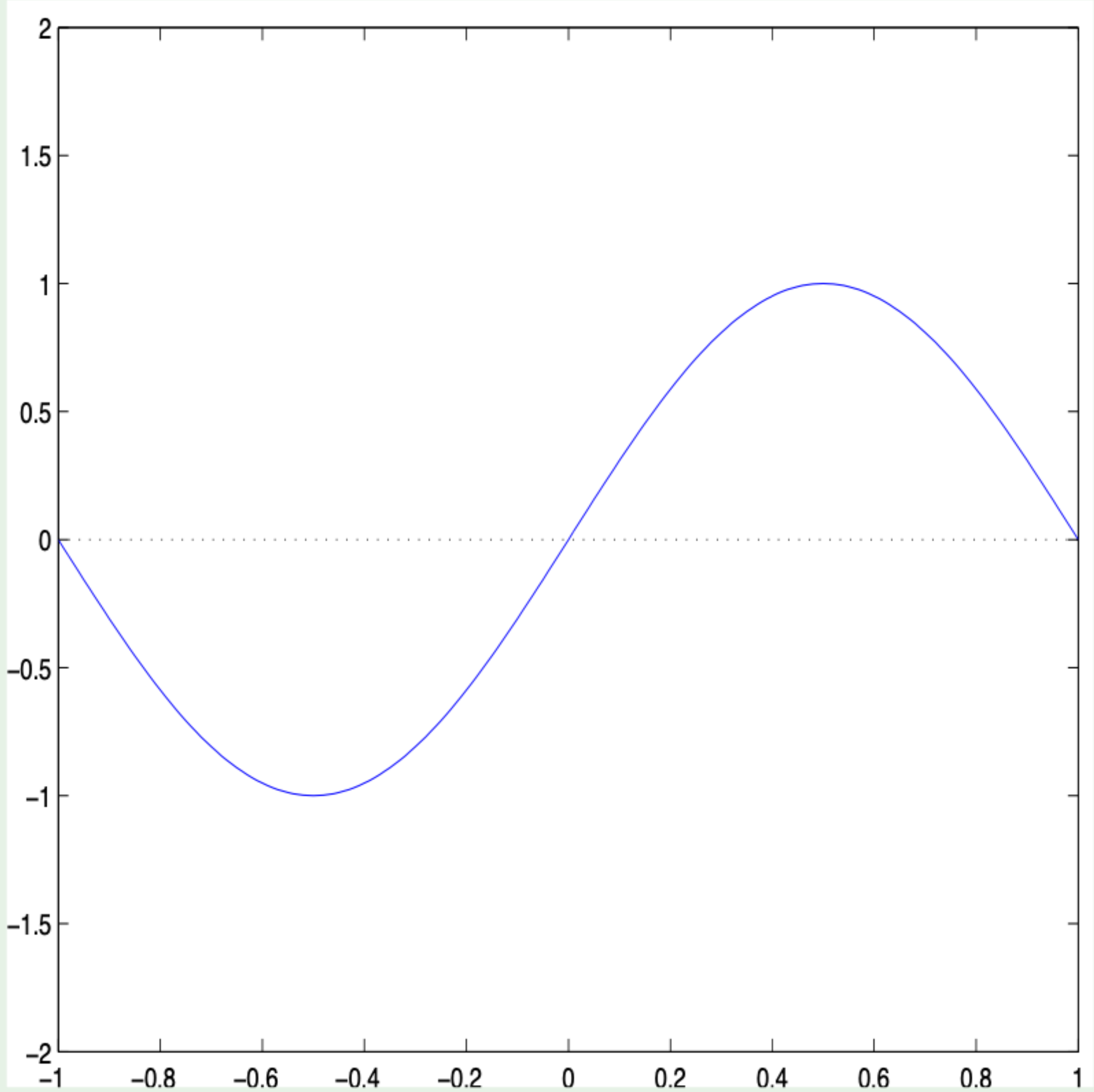
Two models used for learning:

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

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

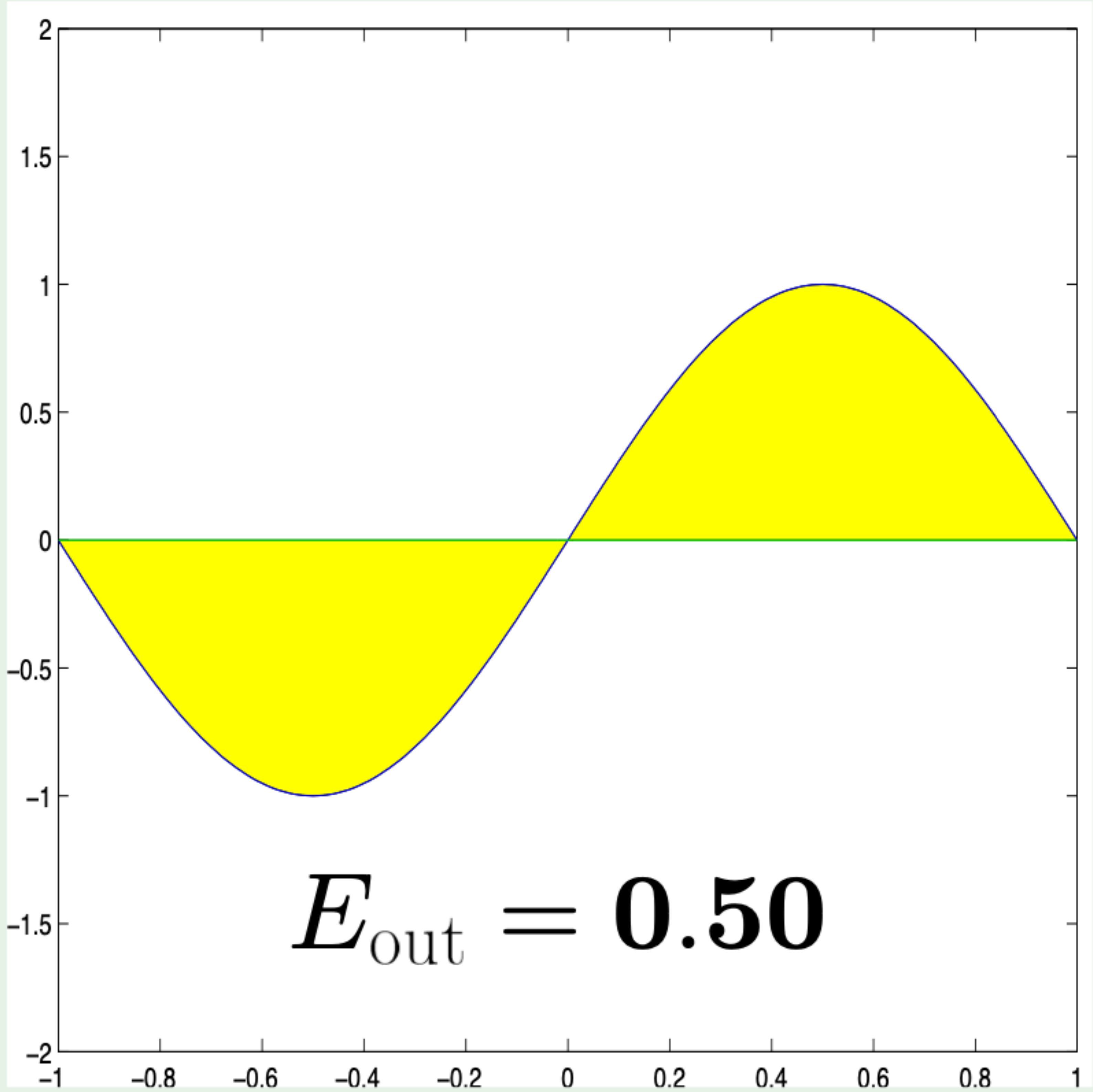
Which is better, $\mathcal{H}_0$ or $\mathcal{H}_1$?

f

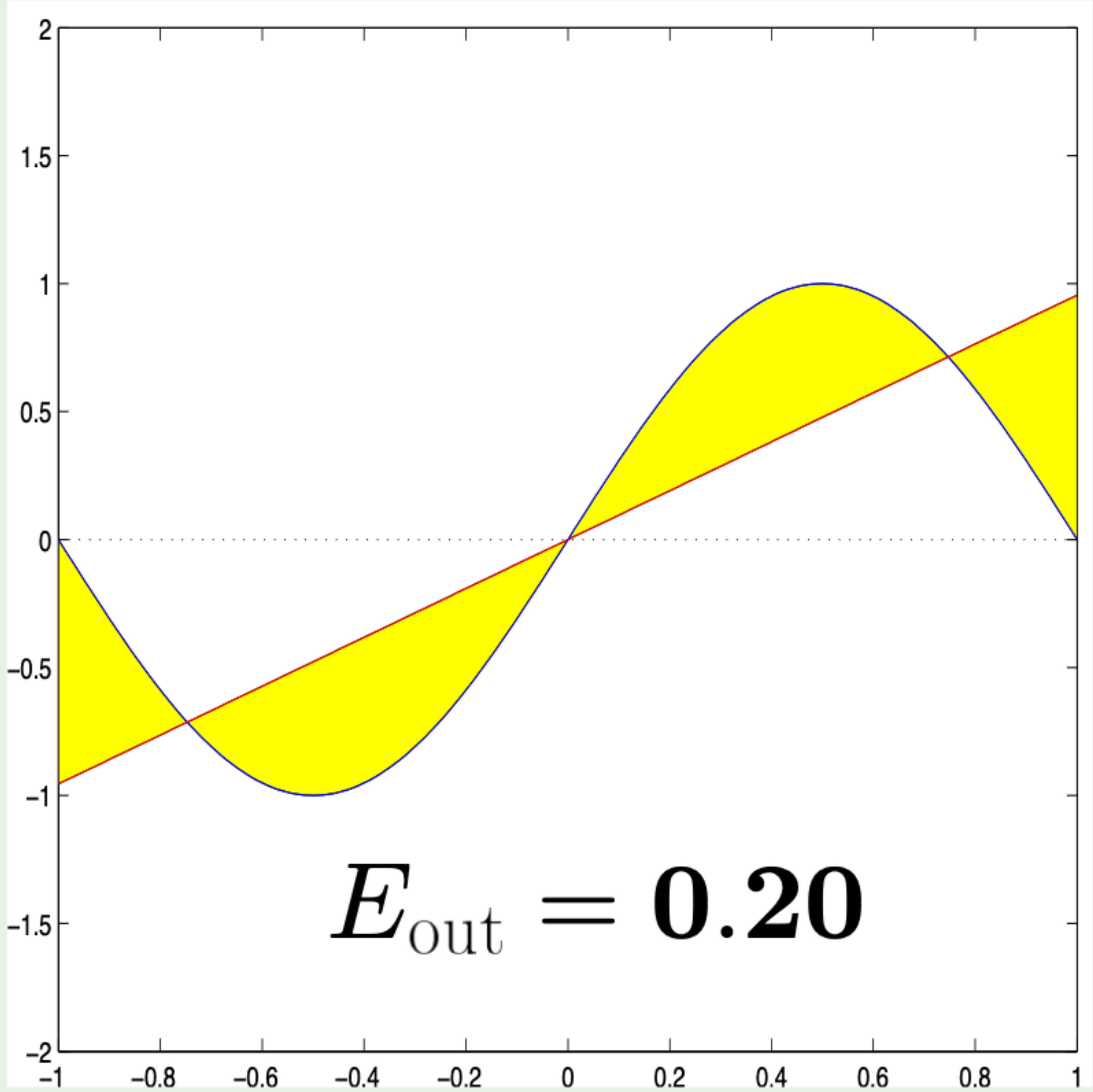


Approximation - $\mathcal{H}_0$ versus $\mathcal{H}_1$

$\mathcal{H}_0$

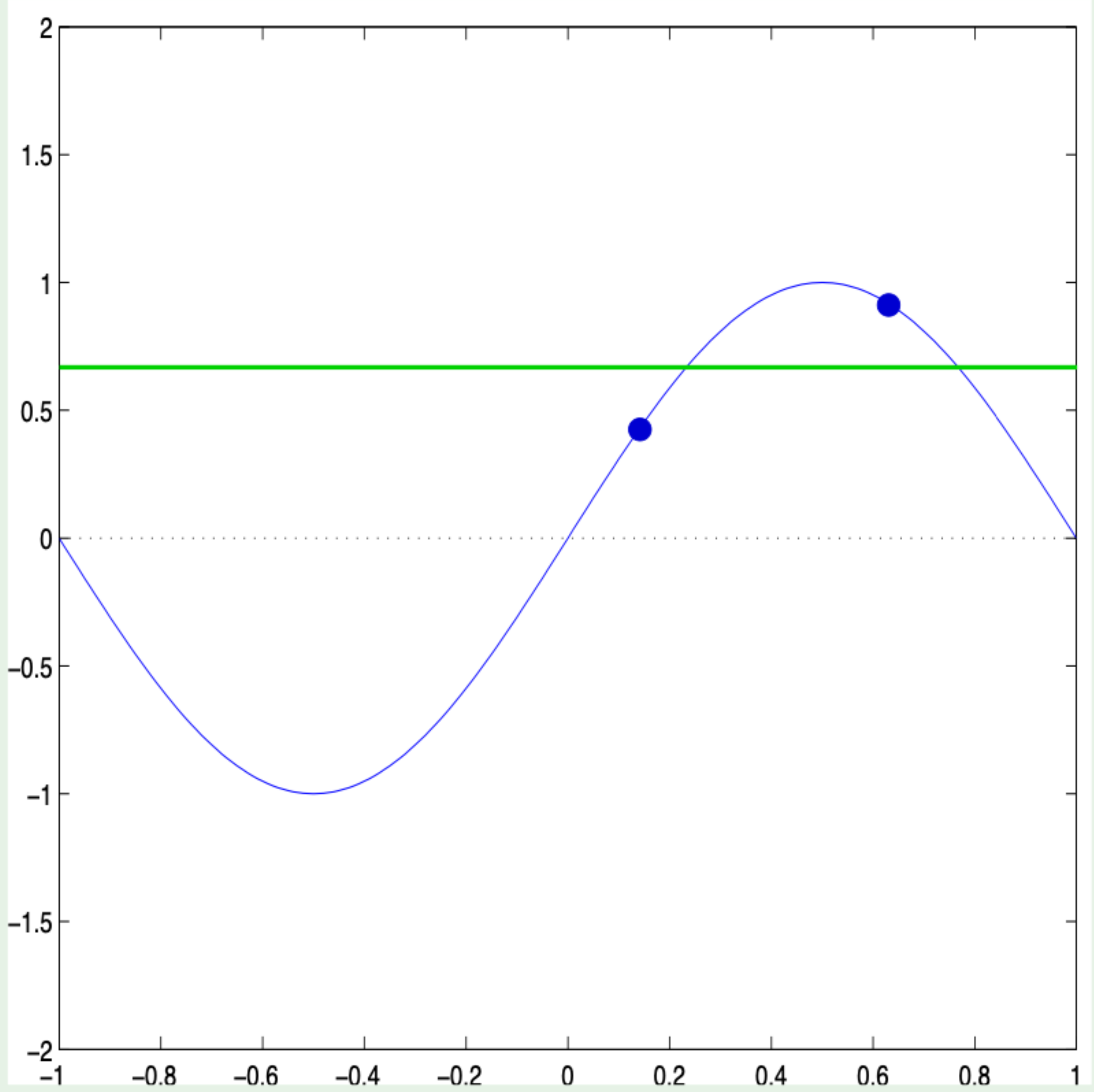


$\mathcal{H}_1$

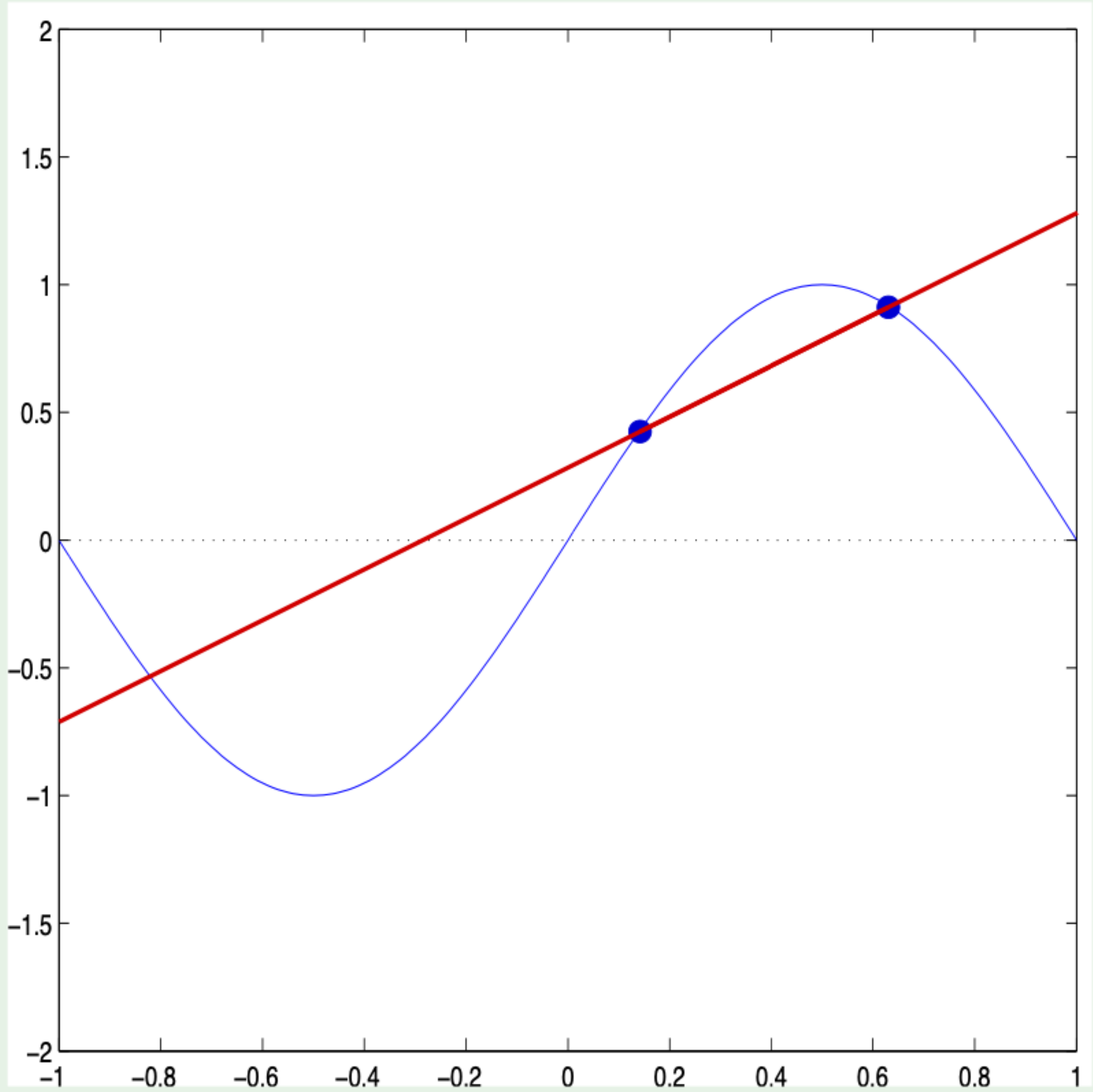


Learning - $\mathcal{H}_0$ versus $\mathcal{H}_1$

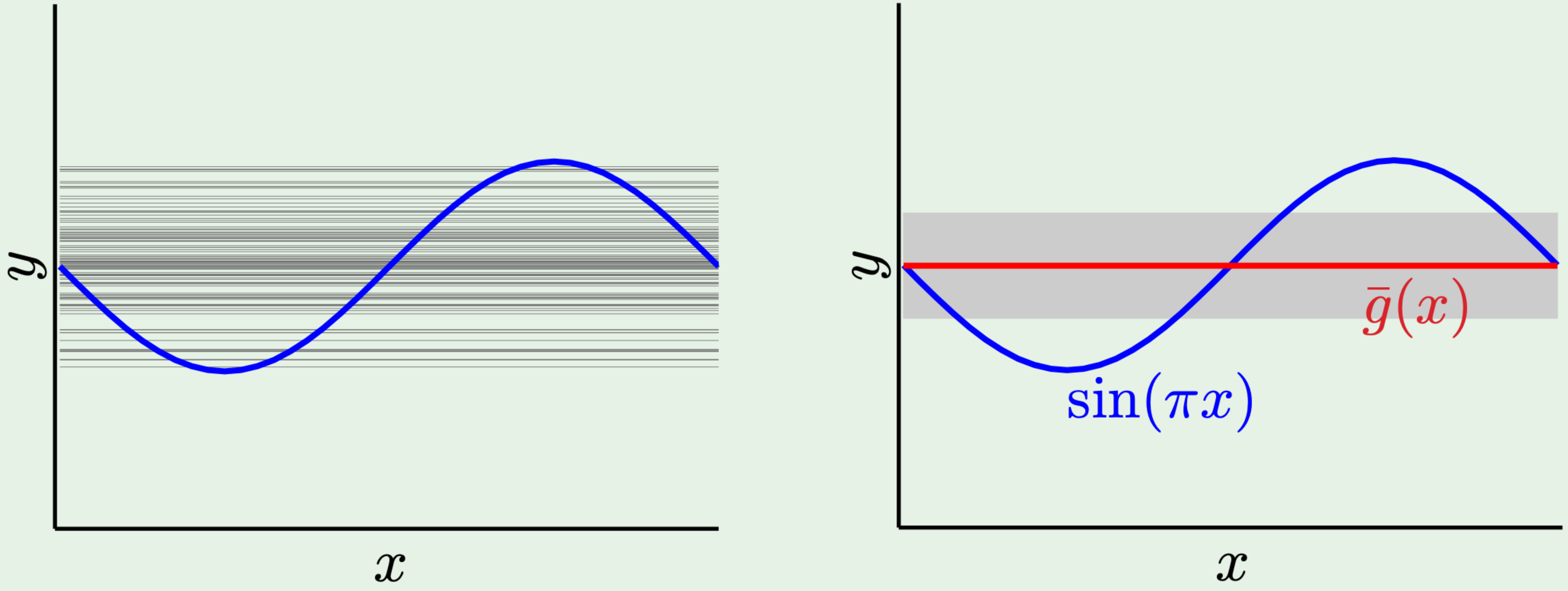
$\mathcal{H}_0$



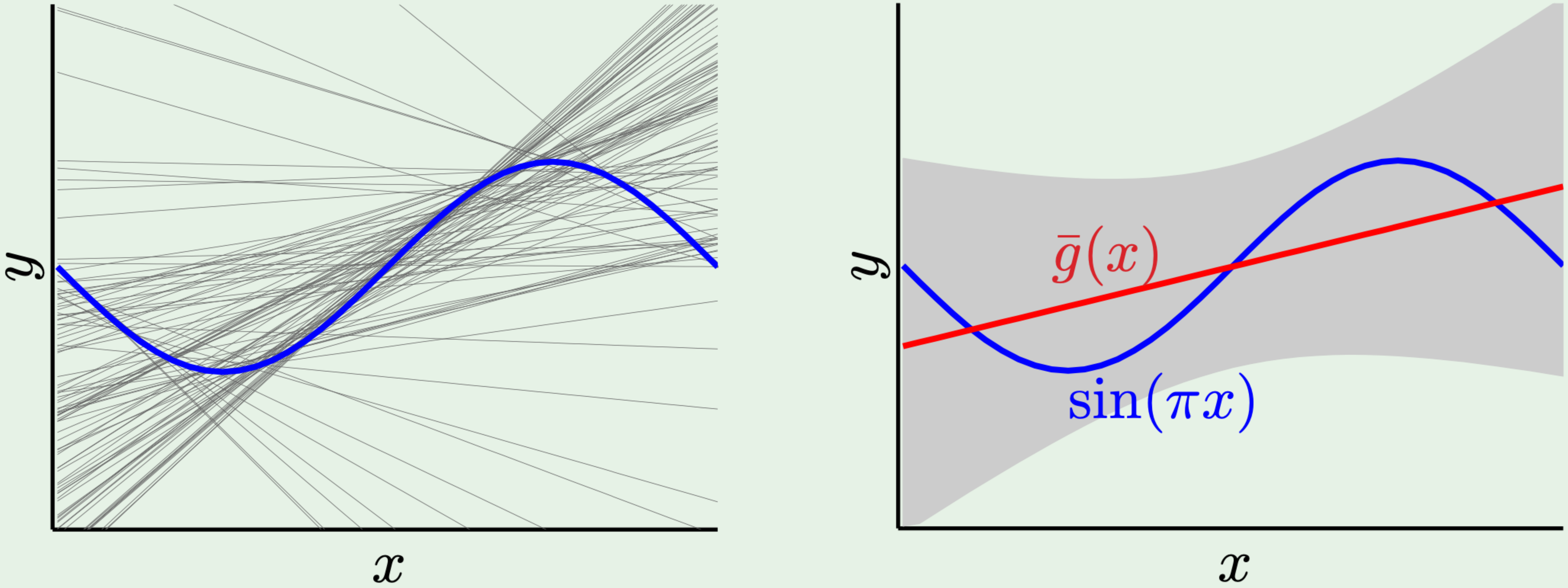
$\mathcal{H}_1$



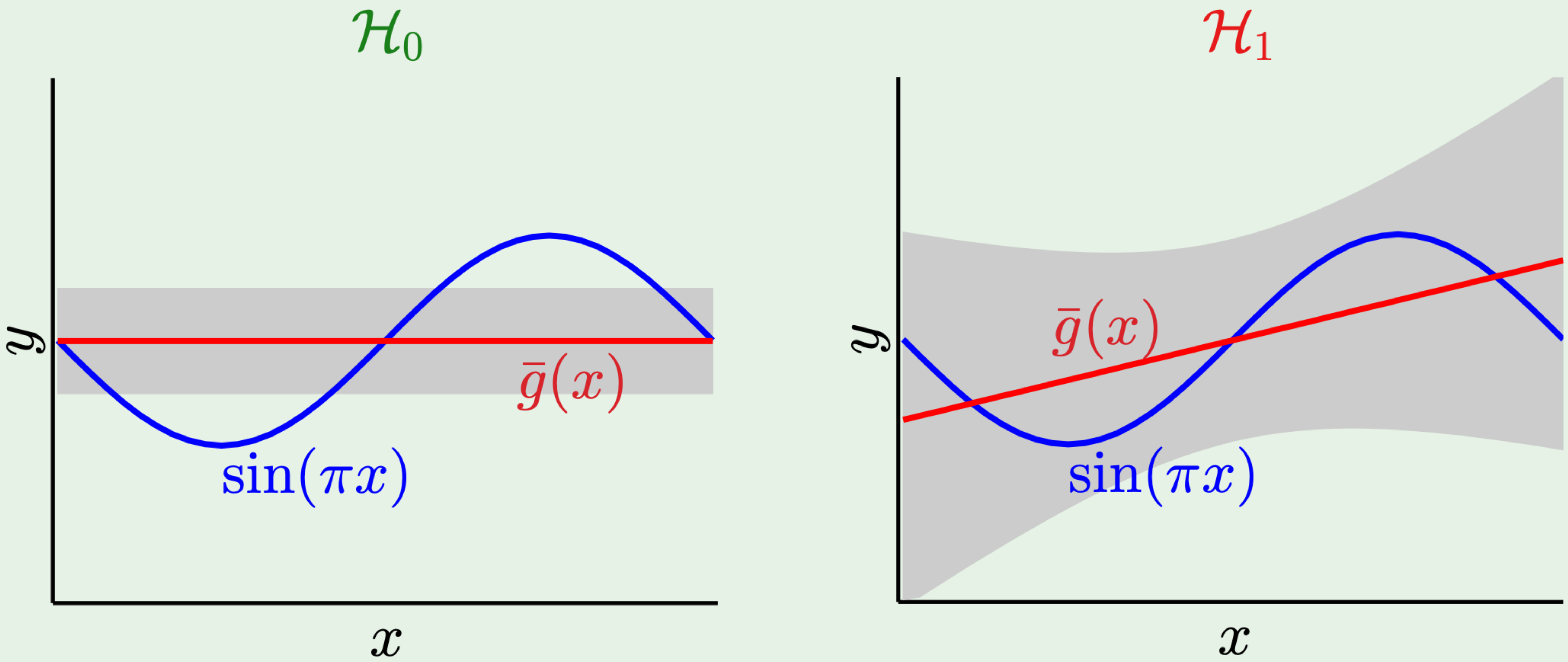
Bias and variance - $\mathcal{H}_0$



Bias and variance - $\mathcal{H}_1$



and the winner is ...



bias $=$ $\mathbf{0.50}$ var $=$ $\mathbf{0.25}$

bias $=$ $\mathbf{0.21}$ var $=$ $\mathbf{1.69}$

Lesson learned

Match the ‘**model complexity**’

to the **data resources**, not to the **target complexity**