Bias-Variance Tradeoff

Ngoc Hoang Luong

University of Information Technology (UIT), VNU-HCM

April 17, 2023

A Theoretical Framework

Player	Height	Weight	Yrs Expr	2 Points	3 Points	Salary
1	...	...	...	...	...	...
2	...	...	...	...	...	...
3	...	...	...	...	...	...
...	...	...	...	...	...	...

- Predict the salary of player - the output variable, denoted as Y.
- The rest of the variables are the inputs, denoted as $X_1, X_2, \ldots, X_p$.
- We assume the existence of a target function $f()$

$$f : \mathcal{X} \to \mathcal{Y},$$

which is a function mapping from the input space $\mathcal{X}$ to the output space $\mathcal{Y}$. This function is unknown.

- We want to "find" this target function.

A Theoretical Framework

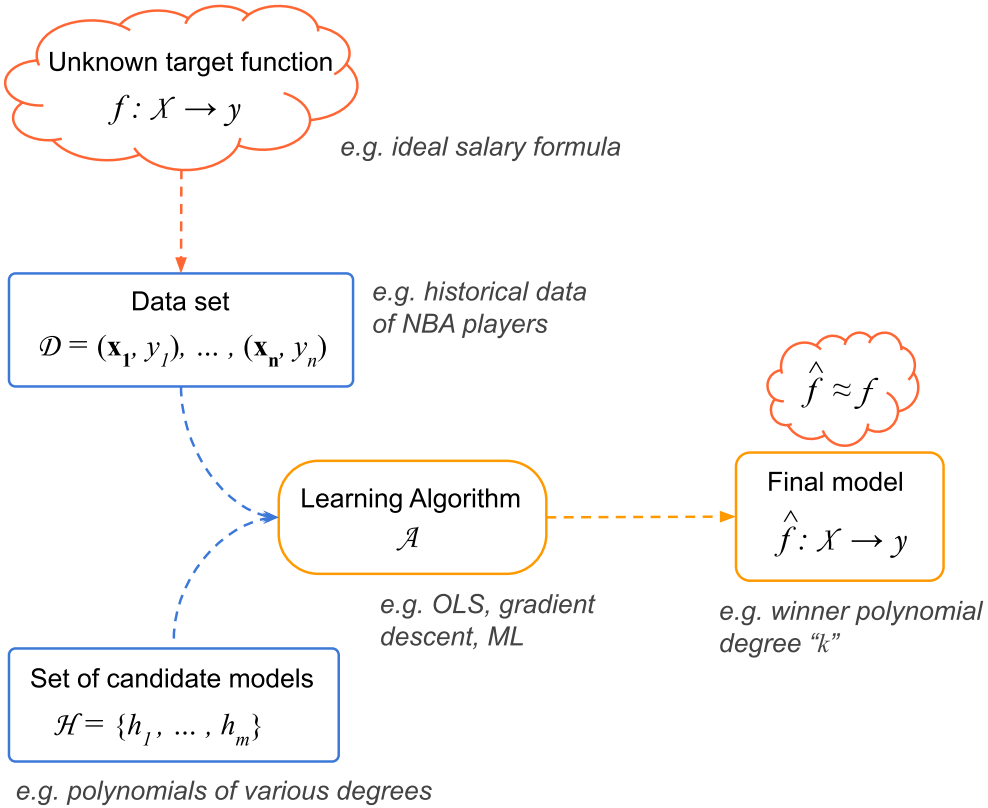
Player	Height	Weight	Yrs Expr	2 Points	3 Points	Salary
1	...	...	...	...	...	...
2	...	...	...	...	...	...
3	...	...	...	...	...	...
...	...	...	...	...	...	...

- We have a dataset $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$ where $\mathbf{x}_i$ is the feature vectors for the i-th player, and y_i is his/her salary.
- From this data, we want to obtain a fitted model, known as a hypothesis model $\hat{f}()$:

$$\hat{f} : \mathcal{X} \to \mathcal{Y}$$

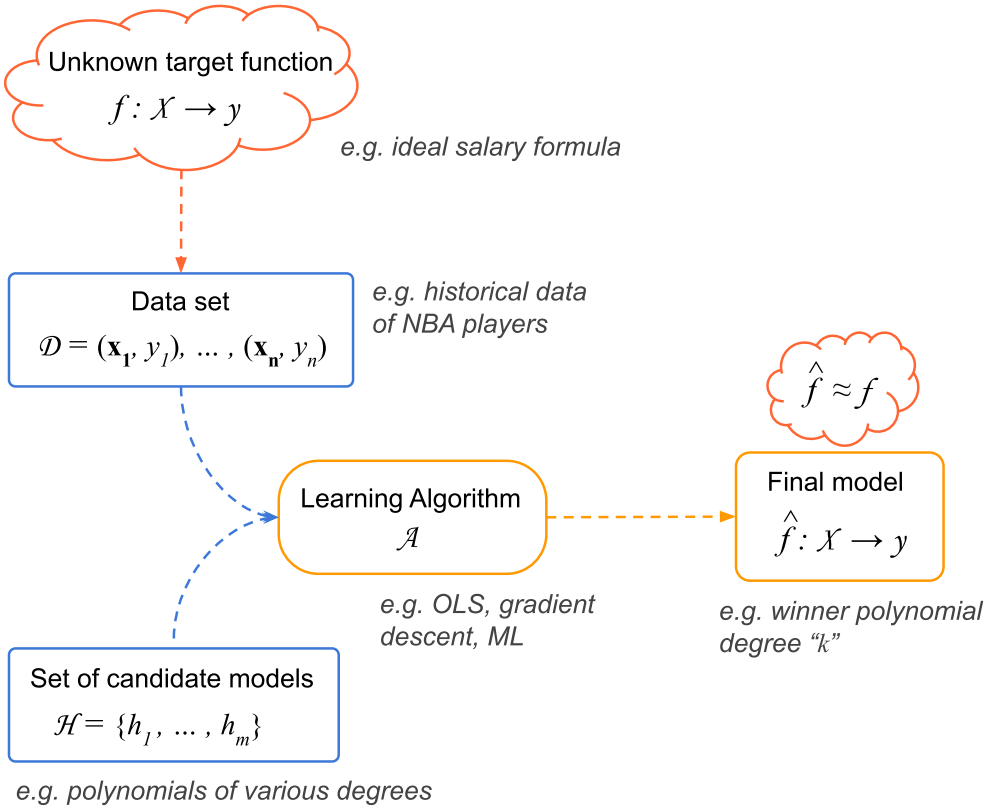
  that approximates the unknown target function $f()$.
- To find $\hat{f}()$, we consider a set of candidate models, known as a hypothesis set $\mathcal{H} = \{h_1, h_2, \ldots, h_m\}$.
- The selected hypothesized model h_m^* will be the one used as the final model $\hat{f}$.

A Theoretical Framework



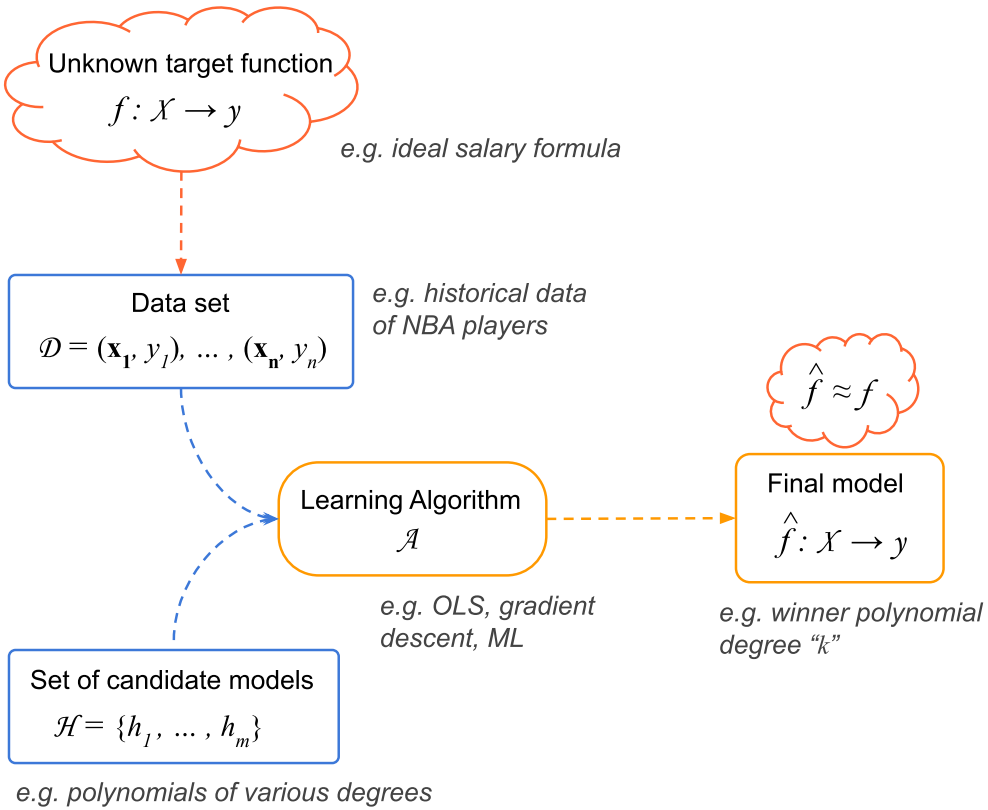
- The target function f is unknown, and we can never really "discover" f. We can only find a good enough approximation to f by estimating $\hat{f}$.
- The idea of a good approximation $\hat{f} \approx f$ is also theoretical because we don't know f.

A Theoretical Framework



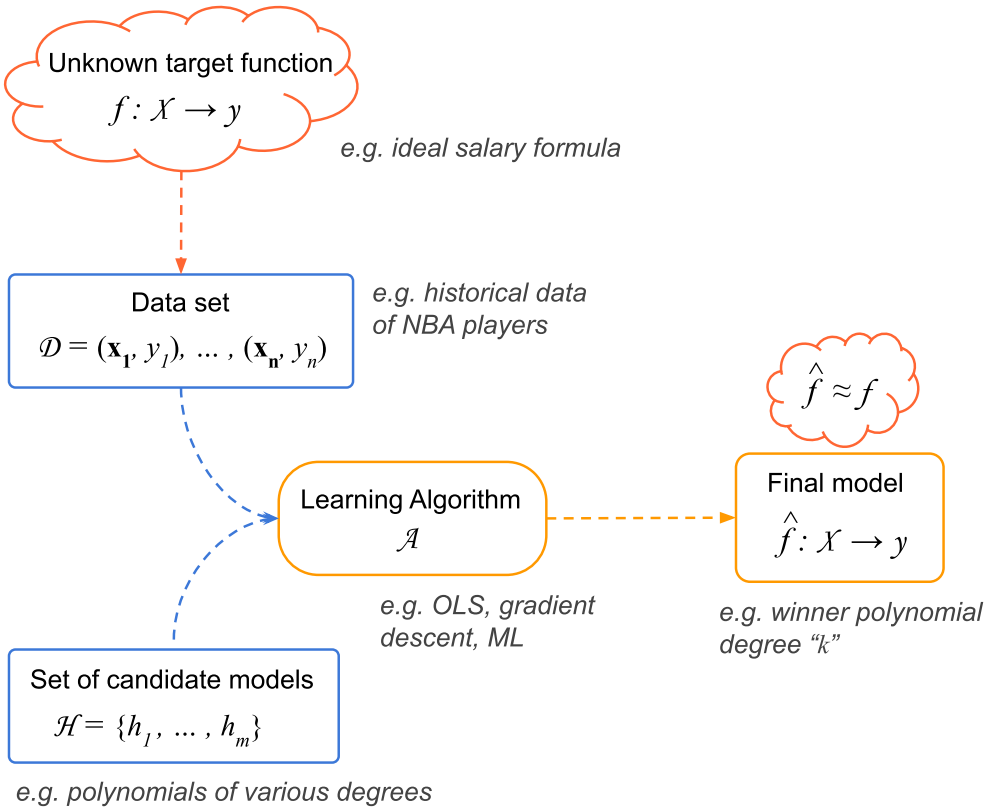
- The dataset $\mathcal{D}$ is influenced by the unknown target function.
- The hypothesis set $\mathcal{H}$ is the set of ML model types that want to try out (e.g., linear model, polynomial models, non-parametric models, etc.)

A Theoretical Framework



- The learning algorithm $\mathcal{A}$ is the set of instructions to be carried out when learning from data.

A Theoretical Framework



- The final model $\hat{f}$ is selected by the learning algorithm from the set of hypothesis models.
- Ideally, $\hat{f}$ should be a good approximation of the target function f.

Types of Predictions

- What does a "good model" mean?
- We want to estimate an unknown function f with some model $\hat{f}$ that gives "good" predictions.
- For a simple linear regression model, a fitted model $\hat{f}(\mathbf{x})$ can be used to make two types of predictions:
  - For an observed point $\mathbf{x}_i$, we can compute $y_i = \hat{f}(\mathbf{x}_i)$. Note that $\mathbf{x}_i$ was part of the learning data used to find $\hat{f}$.
  - For an unseen point $\mathbf{x}_0$, we can compute $\hat{y}_0 = \hat{f}(\mathbf{x}_0)$. Note that $\mathbf{x}_0$ was not part of the learning data used to find $\hat{f}$.
- We have two kinds of dataset:
  - **In-sample data**, denoted by $\mathcal{D}_{in}$, is used to fit a model.
  - **Out-of-sample data**, denoted by $\mathcal{D}_{out}$, is used to measure the predictive quality of a model.

Two Types of Predictions

- With two types of data points, we have two corresponding types of predictions:
  1. predictions $\hat{y}_i$ of observed/seen values $\mathbf{x}_i$
  2. predictions $\hat{y}_0$ of unobserved/unseen values $\mathbf{x}_0$
- The predictions of observed data $\hat{y}_i$ involve the memorizing aspect.
- The predictions of unobserved data $\hat{y}_0$ involve the generalization aspect.
- We are interested in the second type of predictions: we want to find models that are able to give predictions $\hat{y}_0$ as accurate as possible for the real value y_0.
- Having good predictions $\hat{y}_i$ of observed data is often a necessary condition for a good model, but not a sufficient condition.
- Sometimes, we can perfectly fit the observed data, but have a terrible performance for unobserved data $\mathbf{x}_0$.

Error Measure

- We need a way to measure the accuracy of the predictions.
- We need some mechanism to quantify how different the fitted model $\hat{f}()$ is from the target function $f()$: the total amount of error - **Overall Measure of Error**: $E(\hat{f}, f)$.
- The overall measure of error is defined in terms of individual errors $err_i(\hat{y}_i, y_i)$ that quantify the difference between an observed value y_i and its predicted value $\hat{y}_i$.

$$E(\hat{f}, f) = \texttt{measure}\left(\sum err_i(\hat{y}_i, y_i)\right)$$

- We typically use the mean sum of errors as the overall error measure:

$$E(\hat{f}, f) = \frac{1}{n}\left(\sum_i err_i(\hat{y}_i, y_i)\right)$$

Individual Errors

1. Squared error: $err(\hat{f}, f) = (\hat{y}_i - y_i)^2$
2. Absolute error: $err(\hat{f}, f) = |\hat{y}_i - y_i|$
3. Misclassification error: $err(\hat{f}, f) = \mathbf{1}[\hat{y}_i \neq y_i]$
4. ...

- In machine learning, these individual errors are known as **loss functions**.
- We can design different individual error functions, but the above are the most common.

Two Types of Errors

- In machine learning, the overall measures of error are known as the **cost functions** or **risks**.
- There are two types of overall error measures, based on the type of data that is used to assess the individual errors.
  1. **In-sample Error**, denoted E_{in}, is the average of individual errors from data points of the in-sample data $\mathcal{D}_{in}$:

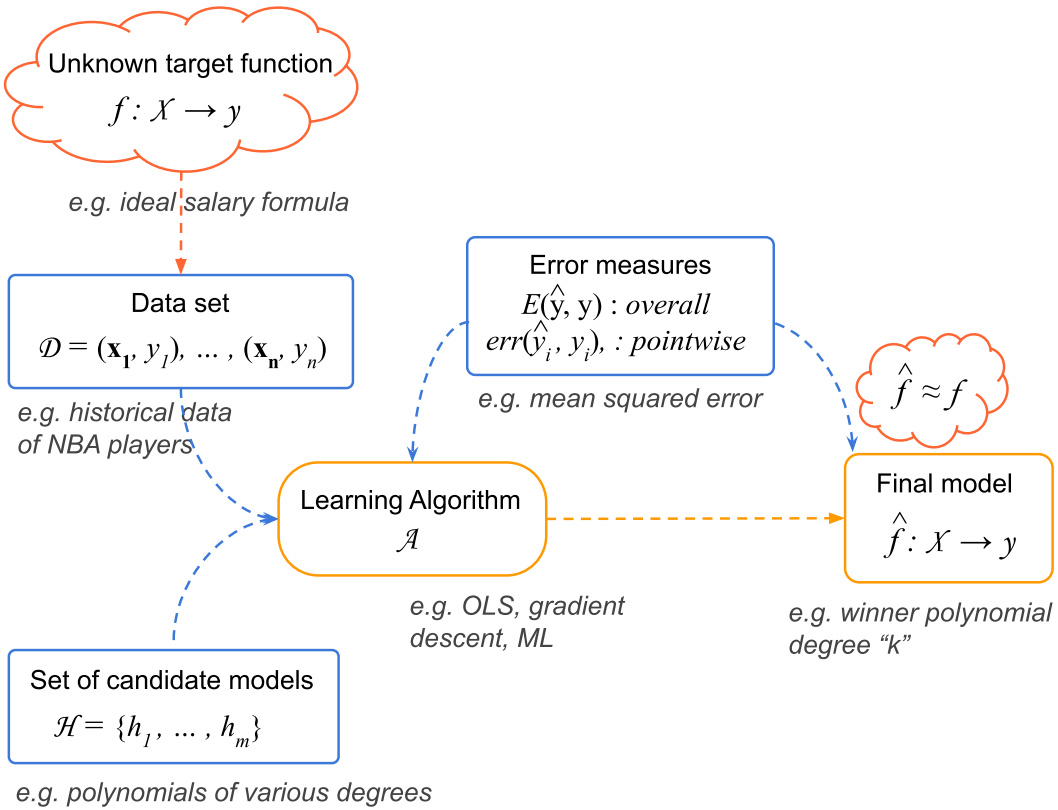
$$E_{in}(\hat{f}, f) = \frac{1}{n} \sum_{i} err_i$$

  2. **Out-of-sample Error**, denoted E_{out}, is the theoretical mean, or expected value, of the individual errors over the entire input space:

$$E_{out}(\hat{f}, f) = \mathbb{E}_{\mathcal{X}} \left[err(\hat{f}(\mathbf{x}), f(\mathbf{x})) \right]$$

- The point $\mathbf{x}$ denotes a general data point in the input space $\mathcal{X}$.
- The expectation is taken over the input space $\mathcal{X}$. Thus, the nature of E_{out} is highly theoretical, we will never able to compute this quantity.

Supervised Learning Diagram

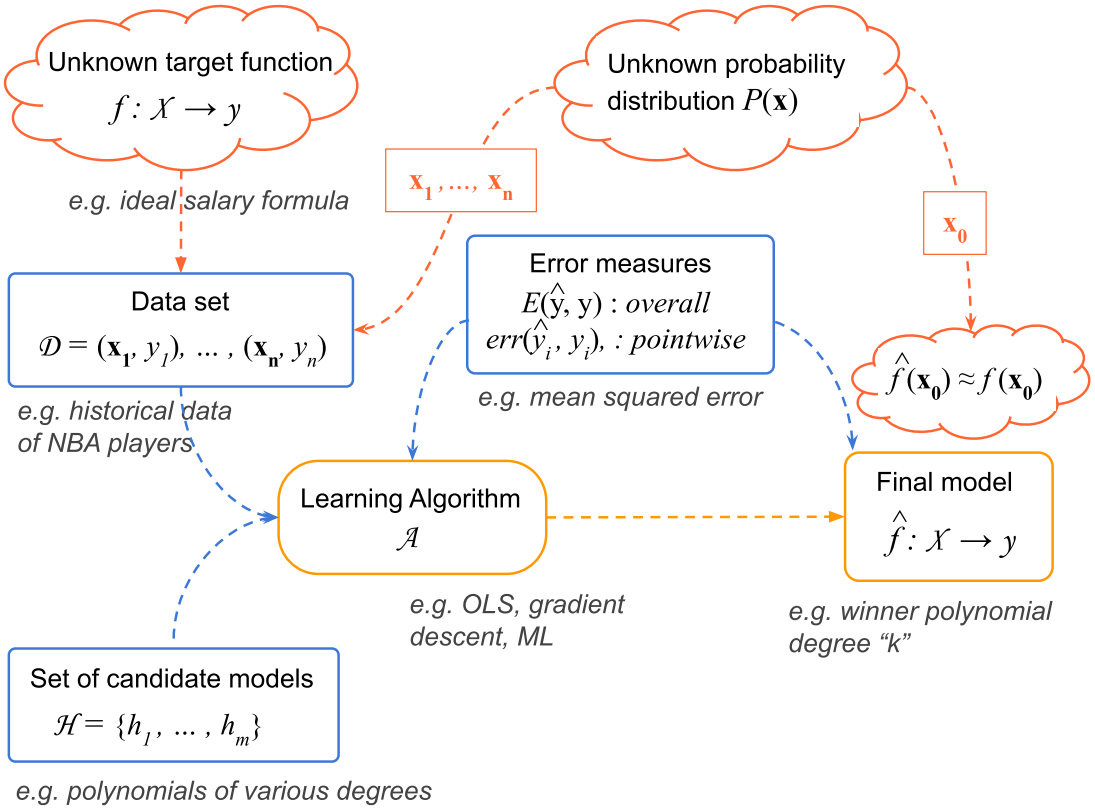


- Learning algorithms $\mathcal{A}$ use individual error function $err()$.
- The overall measure of error $E()$ is used to determine with model $h()$ is the best approximation to the target model $f()$.

Probability Perspective

- Our ultimate goal is to get a good function $\hat{f} \approx f$. Technically, we want $E_{out}(\hat{f}) \approx 0$.
- However, out-of-sample data is theoretical; we can never obtain it entirely. We don't have access to E_{out}.
- The best we can do is to obtain a subset of the out-of-sample data (called the test data).
- We need to assume some probability distribution P over the input space $\mathcal{X}$. Our data points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ are independent identically distributed (iid) samples from this distribution P.
- This links the in-sample error to the out-of-sample data.

Probability Perspective



$$E_{out}(\hat{f}) \approx 0 \Rightarrow \begin{cases} E_{in}(\hat{f}) \approx 0 & \text{practical result} \\ E_{out}(\hat{f}) \approx E_{in}(\hat{f}) & \text{theoretical result} \end{cases}$$

Noisy Targets

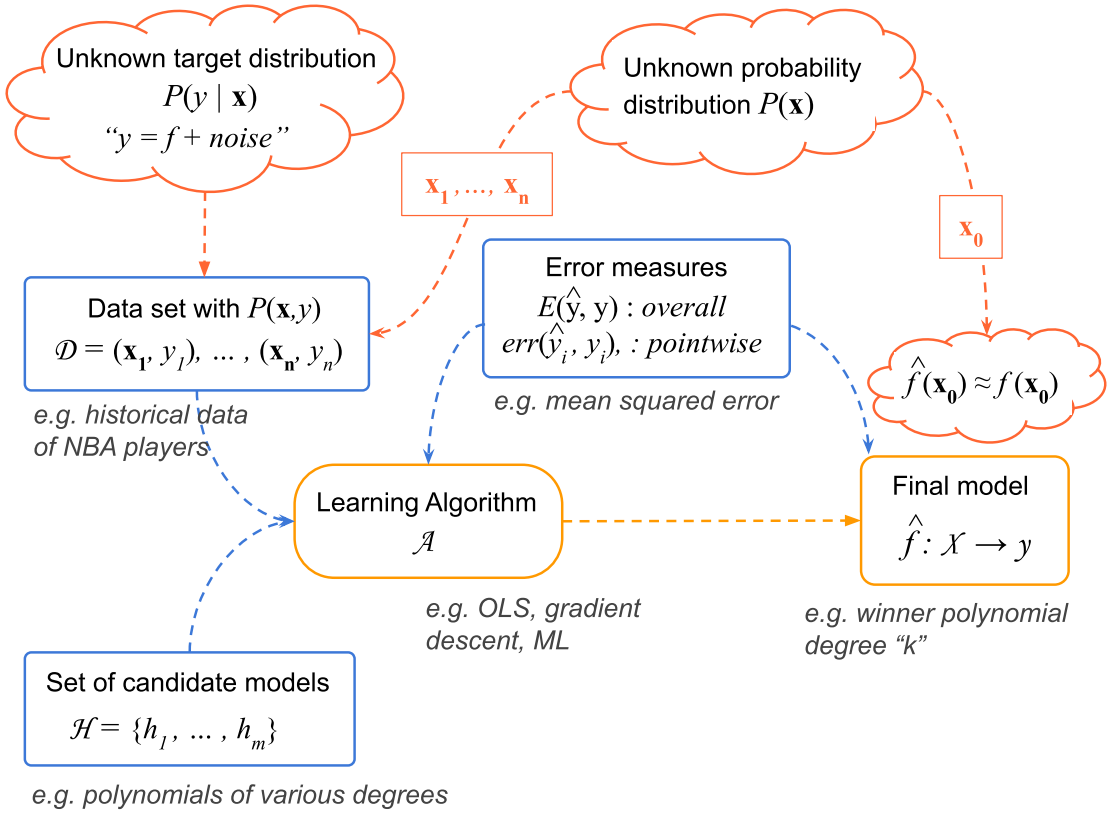
- In practice, there will be some noise. Instead of $y = f(x)$ where $f : \mathcal{X} \to \mathcal{Y}$, it will be:

$$y = f(x) + \epsilon$$

- We could have multiple inputs mapping to the same output..
- We would have two individuals with the exact inputs $\mathbf{x}_A = \mathbf{x}_B$, but with different responses $y_A \neq y_B$.
- We need to consider some target conditional distribution $P(y|\mathbf{x})$.
- Our data can be described as a joint probability distribution $P(\mathbf{x}, y)$:

$$P(\mathbf{x}, y) = P(\mathbf{x})P(y|\mathbf{x})$$

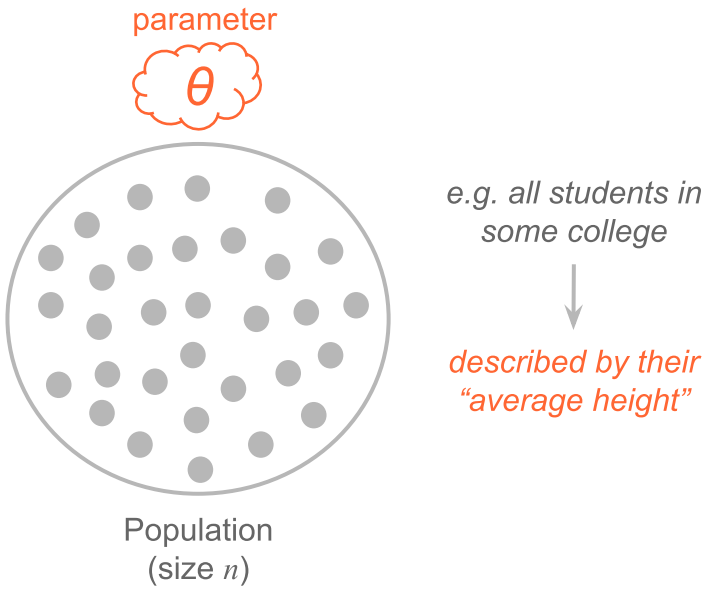
Noisy Targets



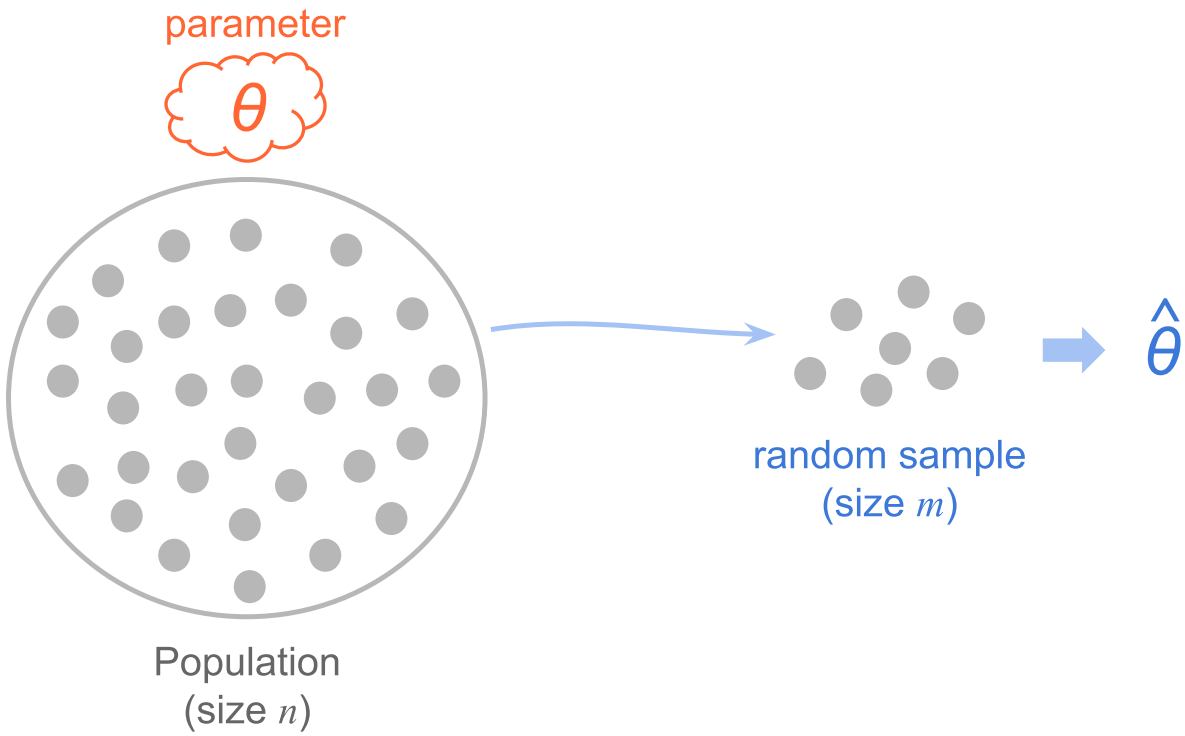
- In supervised learning, we want to learn the conditional distribution $P(y|\mathbf{x})$, where $y = f(\mathbf{x}) + \epsilon$.
- The Hypothesis Set $\mathcal{H}$ and the Learning Algorithm $\mathcal{A}$ are together called the **Learning Model**.

Estimation

- Estimation consists of providing an approximate value to the parameter of a population, using a (random) sample of observations drawn from such population.
- We have a population of n objects, and we want to describe them with some numeric characteristic θ.
- For example, we have a population of all students in a college, and we want to know their average height. We call this (theoretical) average the parameter.

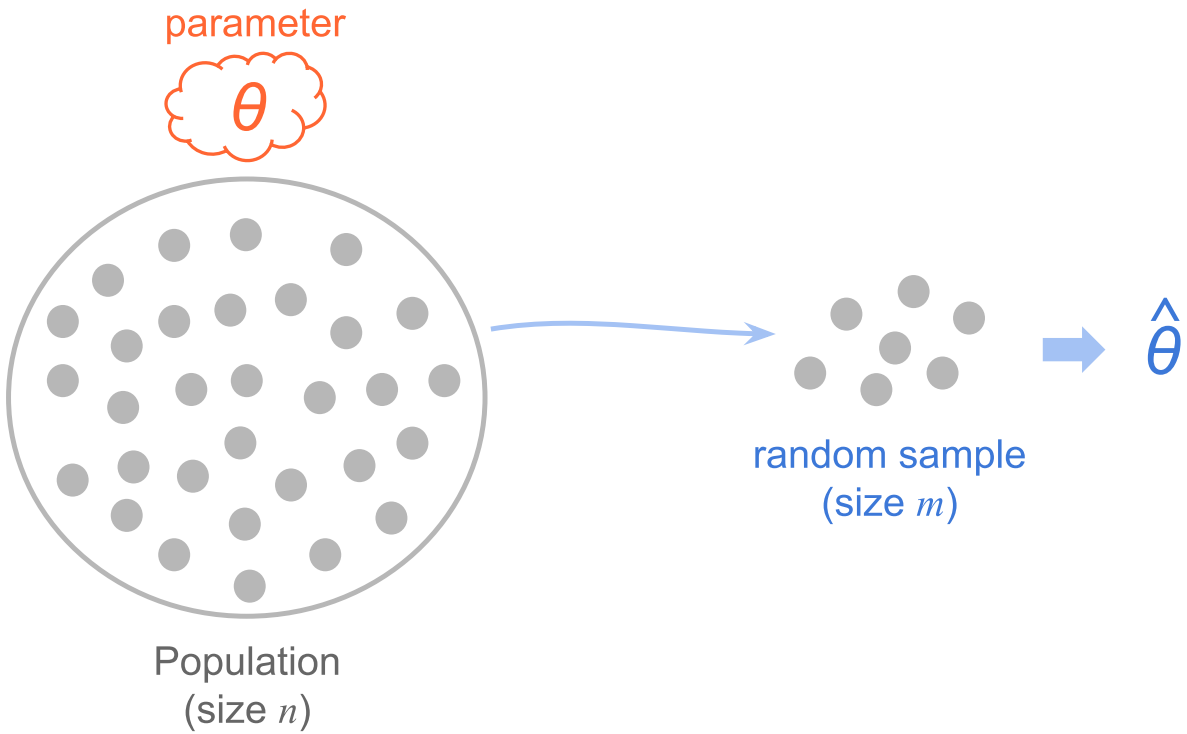


Estimation



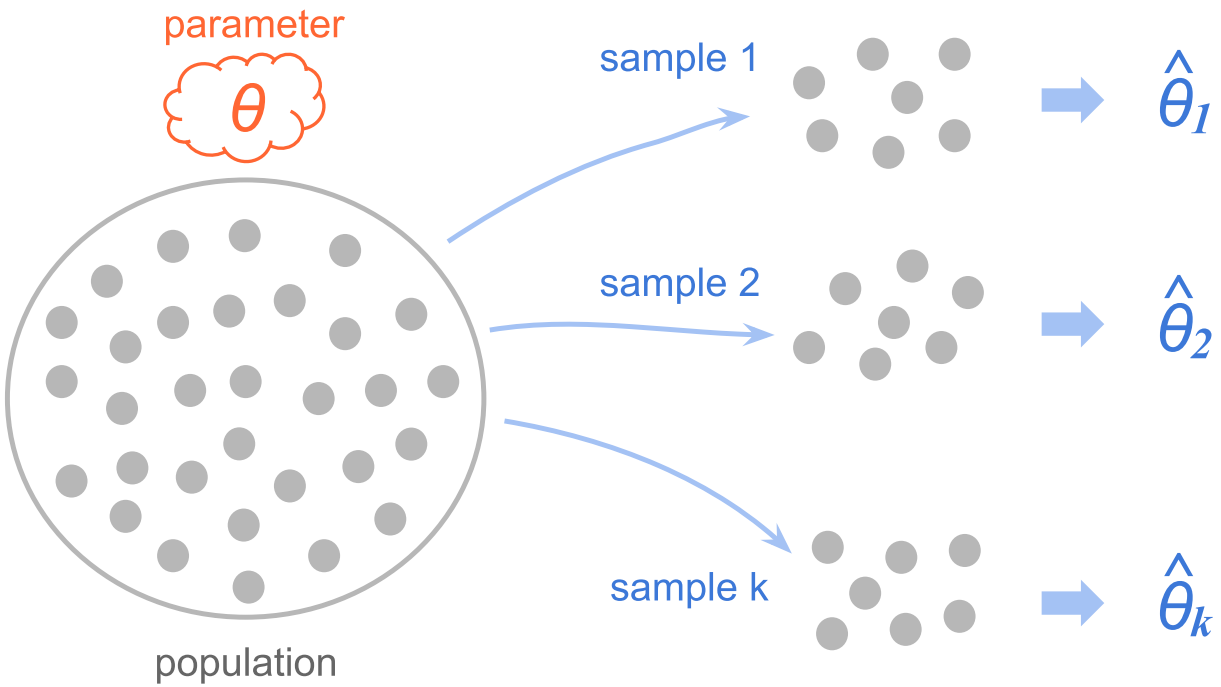
- To estimate the value of the parameter, we draw a sample of $m < n$ students from the population and compute a statistic $\hat{\theta}$.
- Ideally, we would like some statistic $\hat{\theta}$ that approximates well the parameter θ.

Estimation



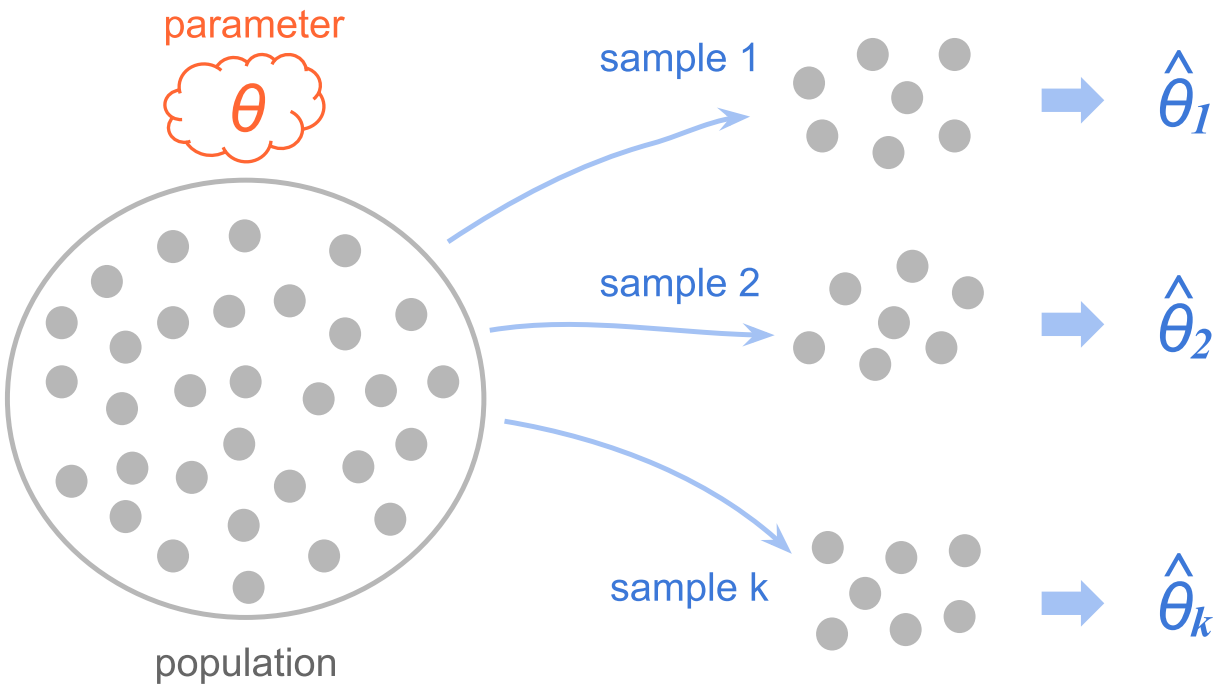
1. Get a random sample from the population.
2. Use the limited amount of data in the sample to estimate θ using some formula to compute $\hat{\theta}$.
3. Make a statement about how reliable an estimator $\hat{\theta}$ is.

Sampling Estimators



- Assume that we can draw multiple random samples, all of the same size m, from the population.
- For each sample, we compute a statistic $\hat{\theta}$.

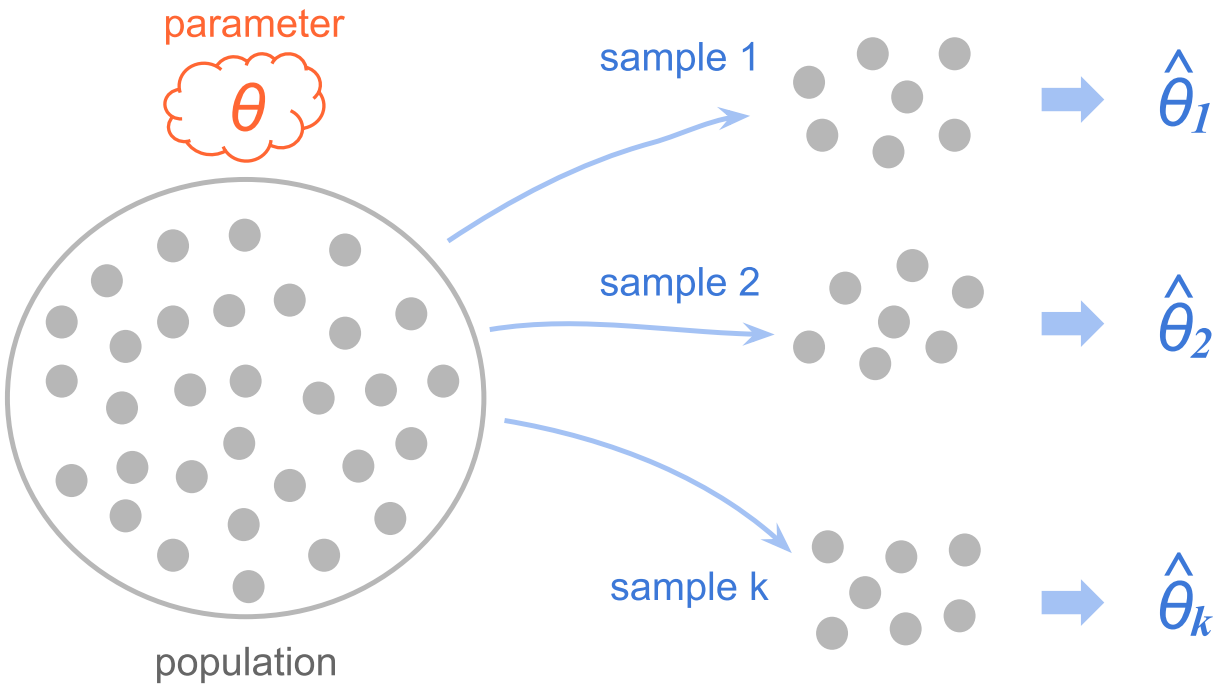
Sampling Estimators



An estimator is a random variable:

- The first sample of size m will result in $\hat{\theta}_1$.
- The second sample of size m will result in $\hat{\theta}_2$.
- The third sample of size m will result in $\hat{\theta}_3$.
- and so on...

Sampling Estimators

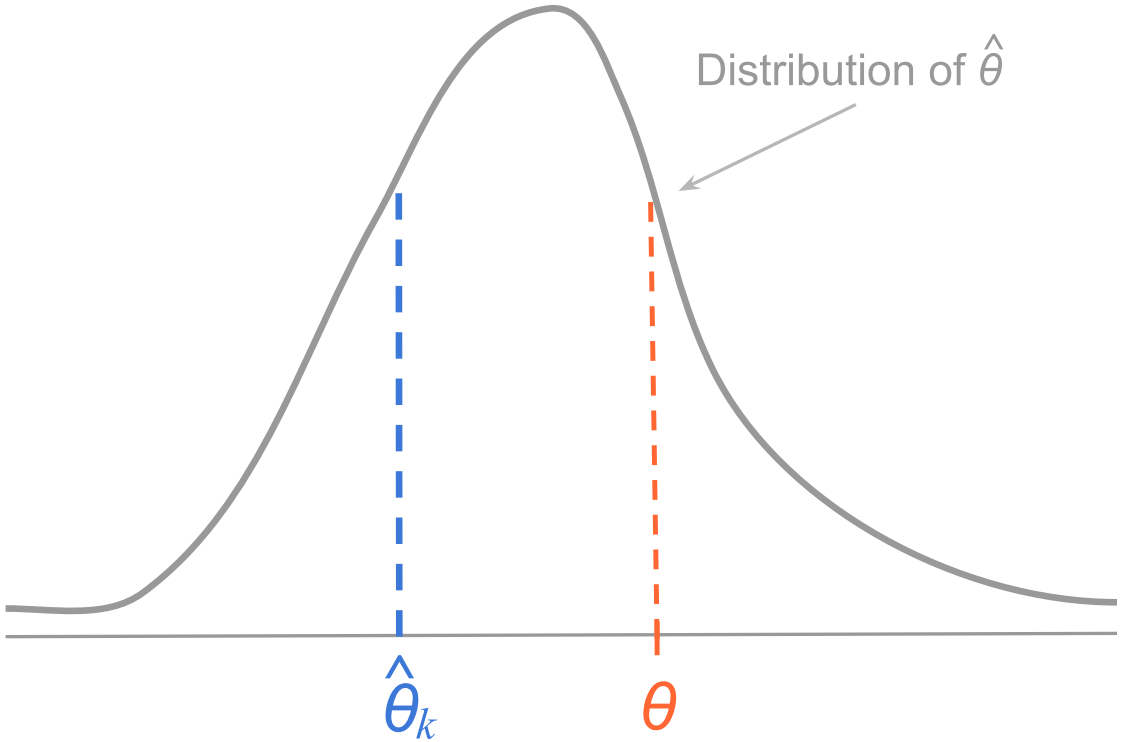


An estimator is a random variable:

- Some samples yield a $\hat{\theta}_k$ that overestimates θ.
- Some samples yield a $\hat{\theta}_k$ that underestimates θ.
- Some samples yield a $\hat{\theta}_k$ that matches θ.

Distribution of Estimators

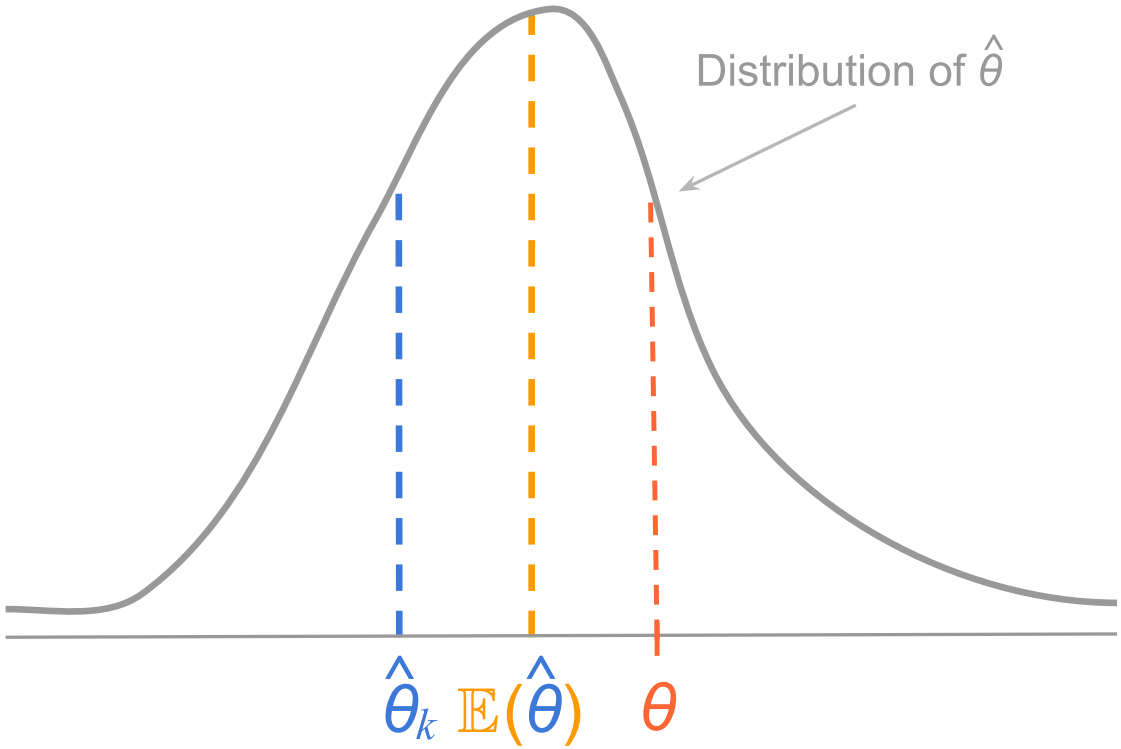
- In theory, we could get a very large number of samples and visualize the distribution of $\hat{\theta}$ as:



- Some estimators will be close to the parameter θ.
- Some estimator will be far away from the parameter θ.

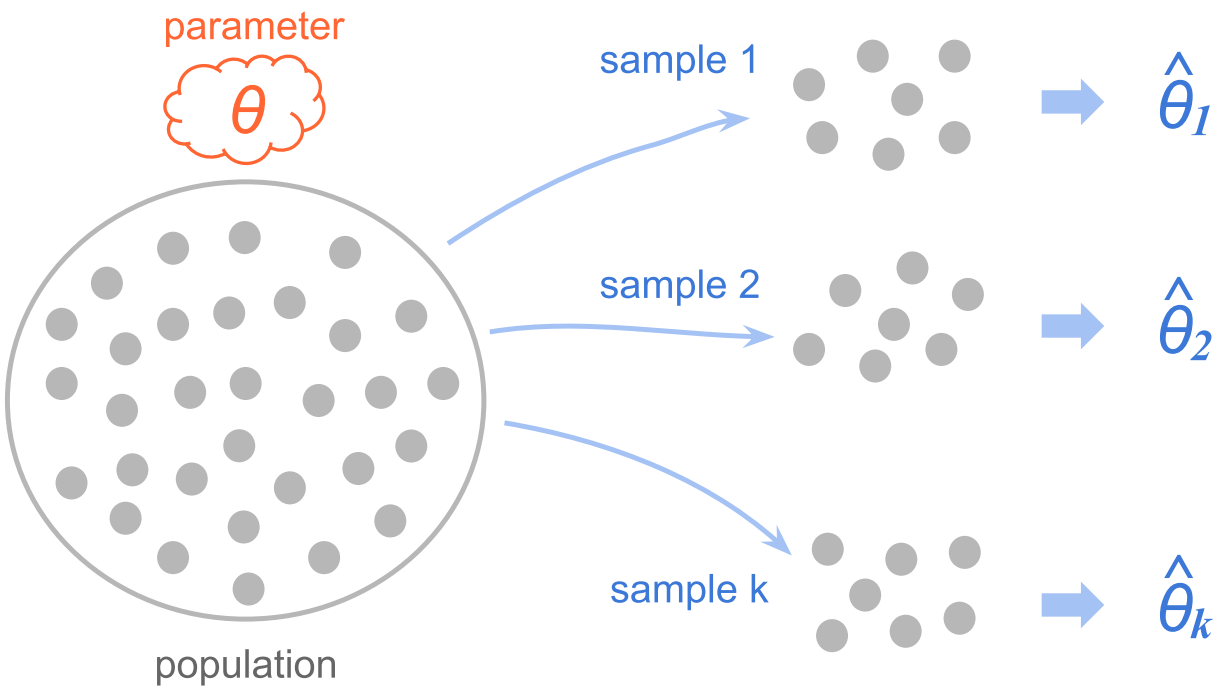
Distribution of Estimators

- The estimator has expected value $\mathbb{E}(\hat{\theta})$ with finite variance $var(\hat{\theta})$.



- How different (or similar) is $\hat{\theta}$ from θ. On average, how close we expect the estimator to be from the parameter?
- We need a measure to assess the typical distance of estimators from the parameter.

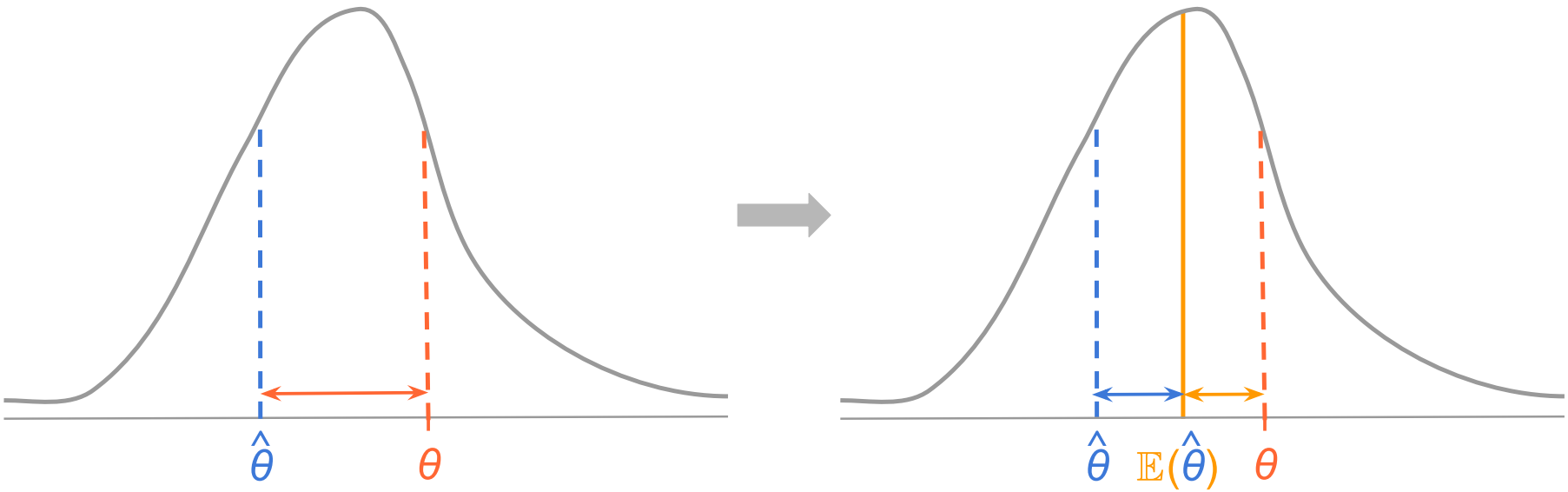
Distribution of Estimators



The difference $\hat{\theta} - \theta$ is the estimation error. The estimation error is also a random variable:

- The first sample yields an error $\hat{\theta}_1 - \theta$.
- The second sample yields an error $\hat{\theta}_2 - \theta$.
- and so on ....

Distribution of Estimators

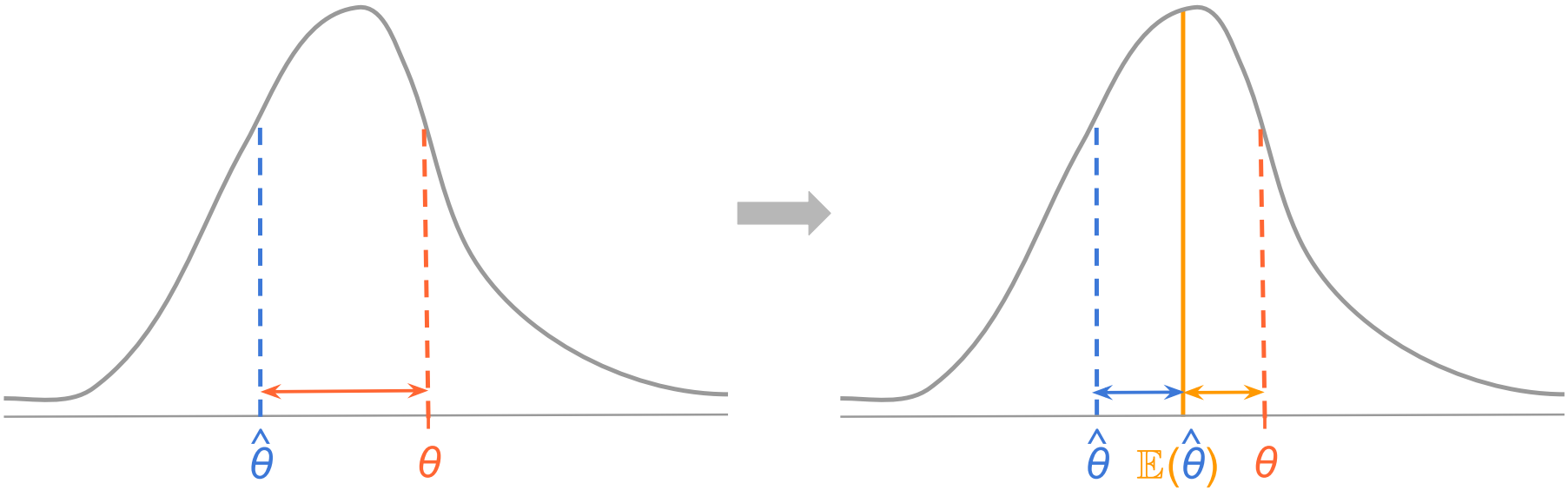


- To measure the size of the estimation errors, we use Mean Squared Error (MSE) of $\hat{\theta}$.

$$\texttt{MSE}(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - \theta)^2]$$

- MSE is the squared distance from our estimator $\hat{\theta}$ to the true value θ, averaged over all possible samples.

Distribution of Estimators



$$
\begin{aligned}
(\hat{\theta} - \theta)^2 &= (\hat{\theta} - \mathbb{E}(\hat{\theta}) + \mathbb{E}(\hat{\theta}) - \theta)^2 \\
&= (\underbrace{\hat{\theta} - \mu_{\hat{\theta}}}_{a} + \underbrace{\mu_{\hat{\theta}} - \theta}_{b})^2 \\
&= a^2 + b^2 + 2ab
\end{aligned}
$$

$$
\Longrightarrow \texttt{MSE}(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - \theta)^2] = \mathbb{E}[a^2 + b^2 + 2ab]
$$

MSE of an Estimator

- The $\texttt{MSE}(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - \theta)^2]$ can be decomposed as:

$$\begin{aligned}\texttt{MSE}(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - \theta)^2] &= \mathbb{E}[a^2 + b^2 + 2ab] \\ &= \mathbb{E}(a^2) + \mathbb{E}(b^2) + 2\mathbb{E}(ab) \\ &= \mathbb{E}[(\hat{\theta} - \mu_{\hat{\theta}})^2] + \mathbb{E}[(\mu_{\hat{\theta}} - \theta)^2] + 2\mathbb{E}(ab)\end{aligned}$$

- We have:

$$\begin{aligned}\mathbb{E}(ab) &= \mathbb{E}[(\hat{\theta} - \mu_{\hat{\theta}})(\mu_{\hat{\theta}} - \theta)] \\ &= (\mu_{\hat{\theta}} - \theta)\mathbb{E}[(\hat{\theta} - \mu_{\hat{\theta}})] \qquad //\mu_{\hat{\theta}} = \mathbb{E}(\hat{\theta}) \text{ and } \theta \text{ are constants} \\ &= (\mu_{\hat{\theta}} - \theta)[\mathbb{E}(\hat{\theta}) - \mathbb{E}(\mu_{\hat{\theta}})] = 0\end{aligned}$$

- Consequently

$$\begin{aligned}\texttt{MSE}(\hat{\theta}) &= \mathbb{E}[(\hat{\theta} - \mu_{\hat{\theta}})^2] + \mathbb{E}[(\mu_{\hat{\theta}} - \theta)^2] \\ &= \mathbb{E}[(\hat{\theta} - \mu_{\hat{\theta}})^2] + \mathbb{E}[(\mu_{\hat{\theta}} - \theta)]^2 = \mathbb{E}[(\hat{\theta} - \mu_{\hat{\theta}})^2] + (\mu_{\hat{\theta}} - \theta)^2 \\ &= \texttt{Var}(\hat{\theta}) + \texttt{Bias}^2(\hat{\theta})\end{aligned}$$

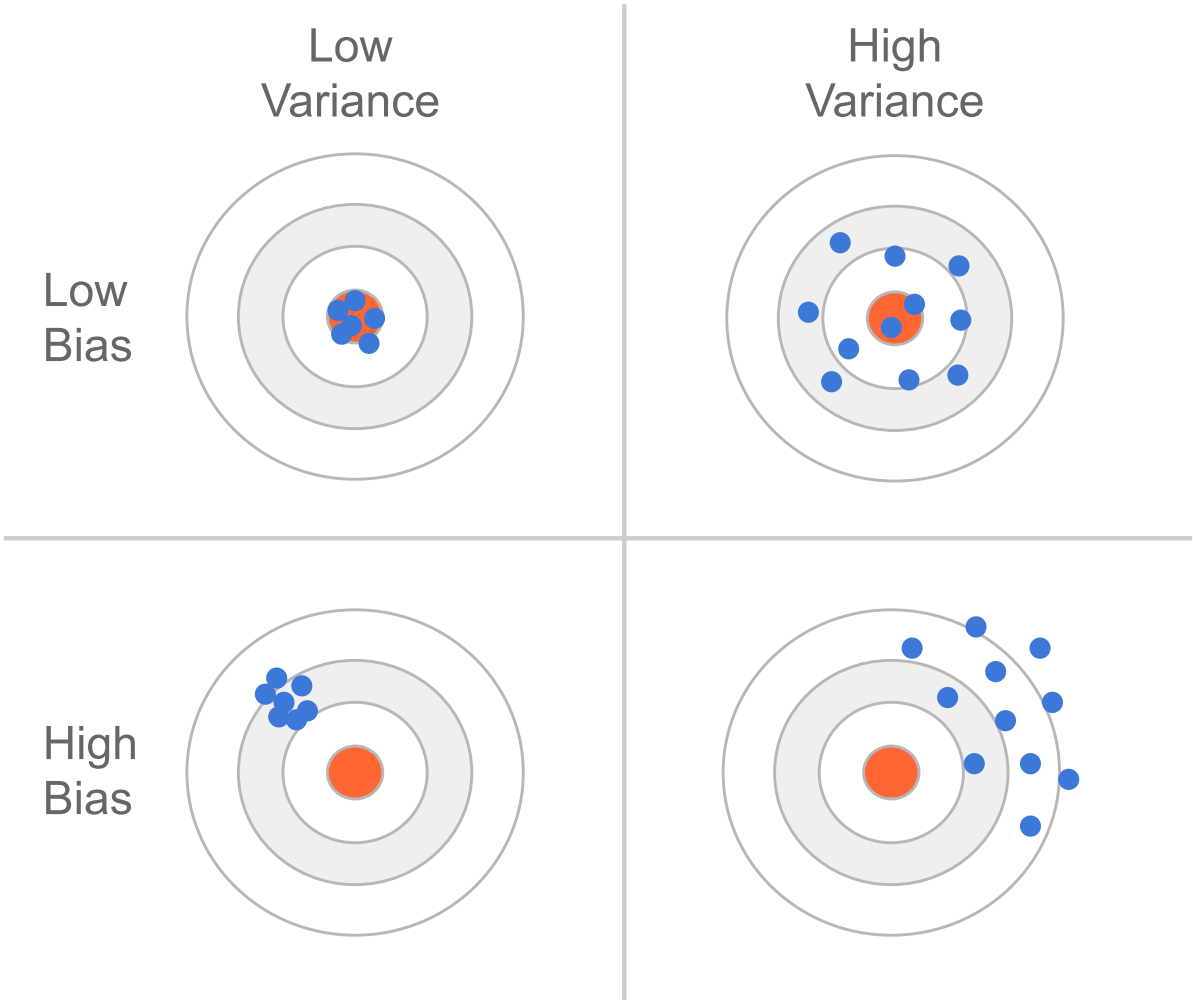
MSE of an Estimator

$$\begin{aligned}\texttt{MSE}(\hat{\theta}) &= \underbrace{\mathbb{E}[(\hat{\theta} - \mu_{\hat{\theta}})^2]}_{\text{Variance}} + \underbrace{(\mu_{\hat{\theta}} - \theta)}_{\text{Bias}}{}^2 \\ &= \texttt{Var}(\hat{\theta}) + \texttt{Bias}^2(\hat{\theta})\end{aligned}$$

- The MSE of an estimator can be decomposed in terms of Bias and Variance.
- **Bias**, $\mu_{\hat{\theta}} - \theta$, is the tendency of $\hat{\theta}$ to overestimate or underestimate θ over all possible samples.
- **Variance**, $\texttt{Var}(\hat{\theta})$, measures the average variability of the estimators around their mean $\mathbb{E}(\hat{\theta})$.

Cases of Biases and Variance

Depending on the type of estimator $\hat{\theta}$ and the sample size m, we can get statistics having different behaviors.



Theoretical Framework of Supervised Learning

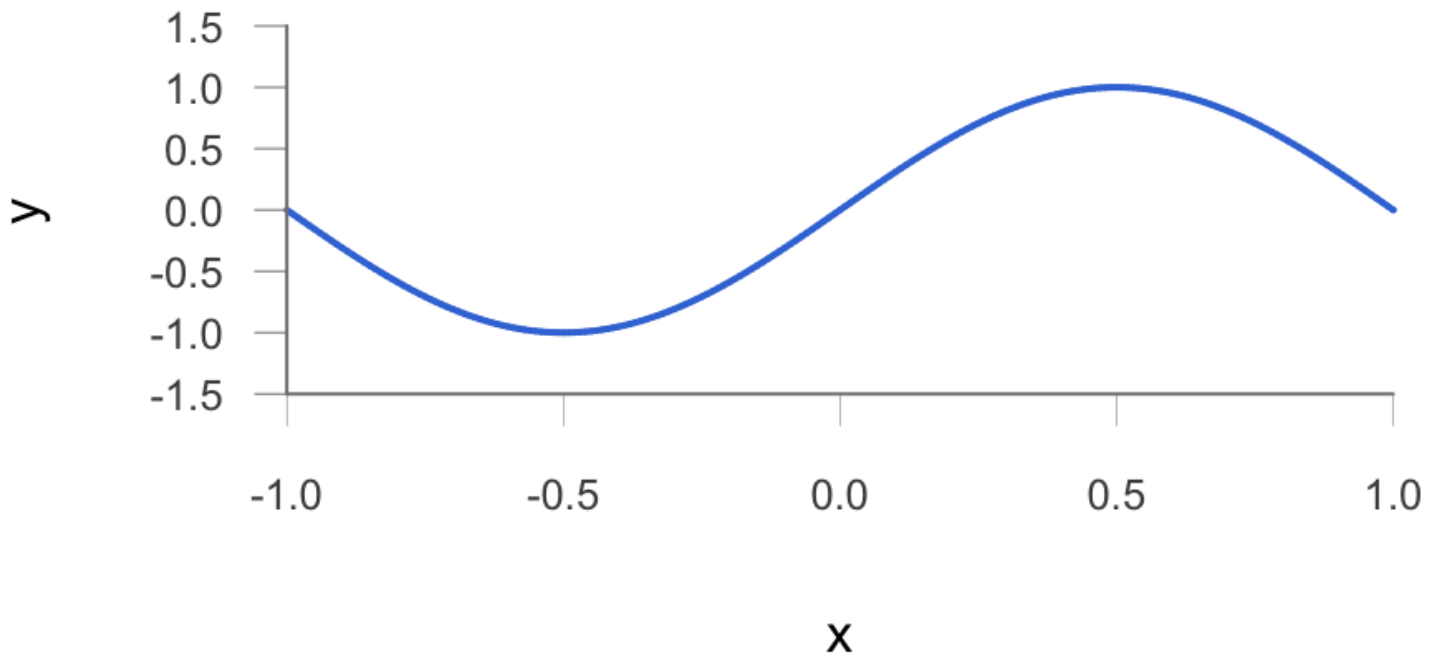
- The main goal is to find a model $\hat{f}$ that approximates well the target function f, i.e., $\hat{f} \approx f$.
- We want to find a model $\hat{f}$ that gives good predictions on both types of data points:
  1. **in-sample data** $\hat{y}_i = \hat{f}(\mathbf{x}_i)$, where $(\mathbf{x}_i, y_i) \in \mathcal{D}_{in}$
  2. **out-of-sample data** $\hat{y}_0 = \hat{f}(\mathbf{x}_0)$, where $(\mathbf{x}_0, y_0) \in \mathcal{D}_{out}$
- Two types of predictions involve two types of errors:
  1. **in-sample error**: $E_{in}(\hat{f})$
  2. **out-of-sample error**: $E_{out}(\hat{f})$
- To have $\hat{f} \approx f$, we need to achieve two goals:
  1. small in-sample error: $E_{in}(\hat{f}) \approx 0$
  2. out-of-sample error similar to in-sample error: $E_{out}(\hat{f}) \approx E_{in}(\hat{f})$
- We will study the theoretical behavior of $E_{out}(\hat{f})$ from a regression perspective with **Mean Squared Error (MSE)**.

Motivation Example

- Consider a **noiseless** target function:

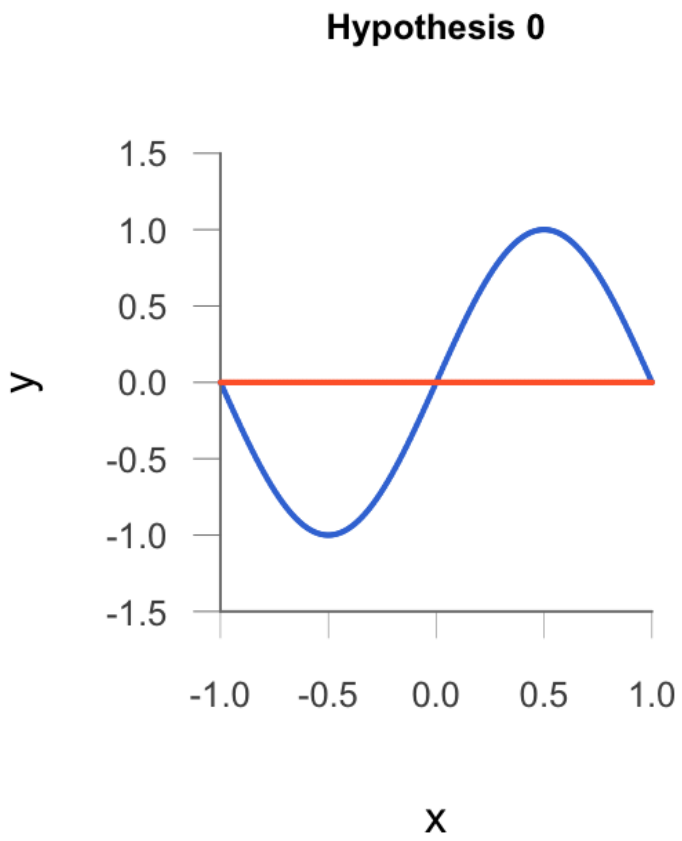
$$f(x) = \sin(\pi x)$$

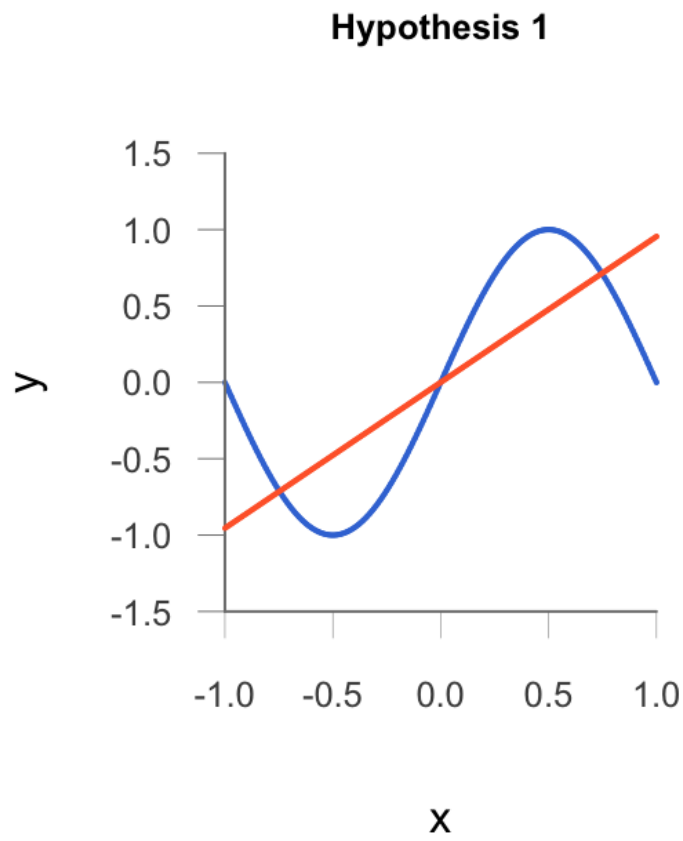
with the input variable $x \in [-1, 1]$.



Motivation Example - Two Hypotheses

- Give a dataset of n data points, we fit the data using one of two hypothesis spaces $\mathcal{H}_0$ and $\mathcal{H}_1$:
- $\mathcal{H}_0$: the set of all lines of the form $h(x) = b$
- $\mathcal{H}_1$: the set of all lines of the form $h(x) = b_0 + b_1 x$





Learning from two points

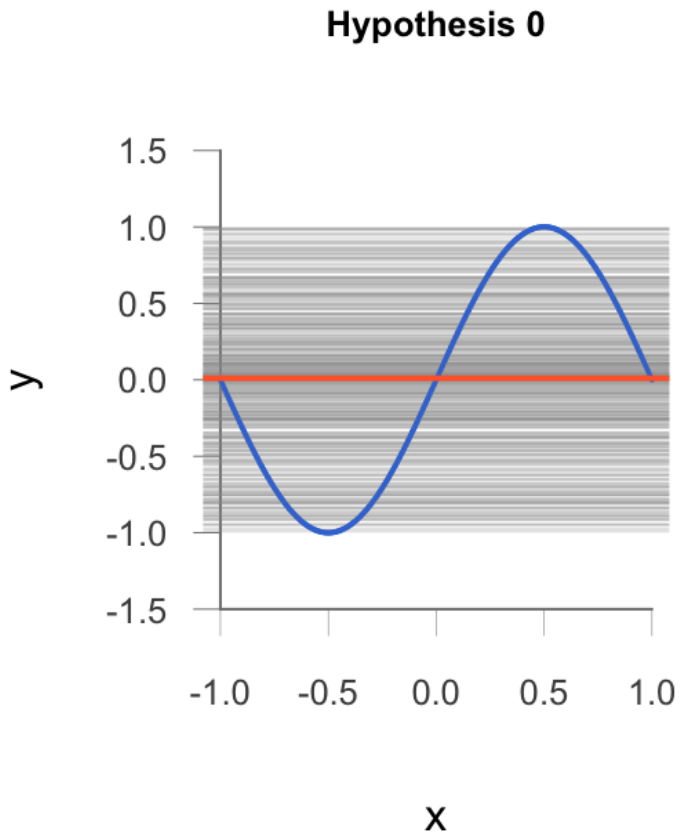
- We assume a dataset of size $n = 2$. $\mathcal{D} = \{(x_1, y_1), (x_2, y_2)\}$, where $x_1, x_2 \in [-1, 1]$.
- For $\mathcal{H}_0$, we choose the constant hypothesis that best fits the data: the horizontal line at the midpoint:

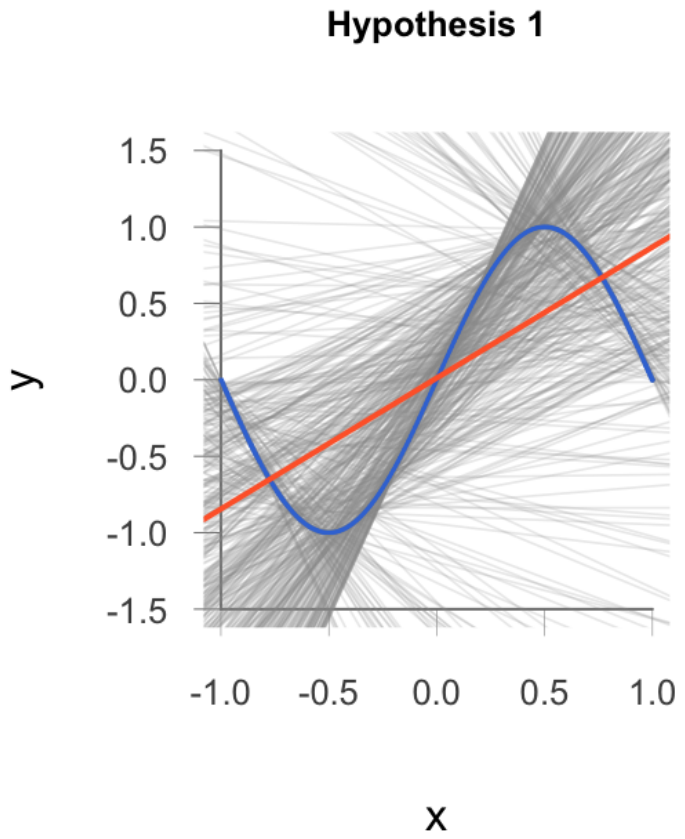
$$b = \frac{y_1 + y_2}{2}$$

- For $\mathcal{H}_1$, we choose the line that passes through the two data points (x_1, y_1) and (x_2, y_2).

Learning from two points

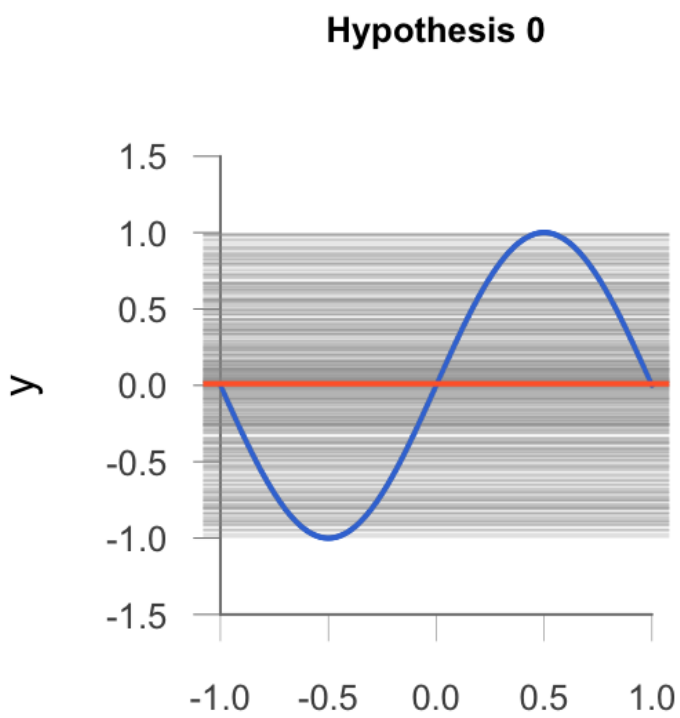
- For 500 times, we randomly sample two points in the interval $[-1, 1]$, and fit both models h_0 and h_1.
- The average hypotheses $\bar{h}_0$ and $\bar{h}_1$ displayed in orange.

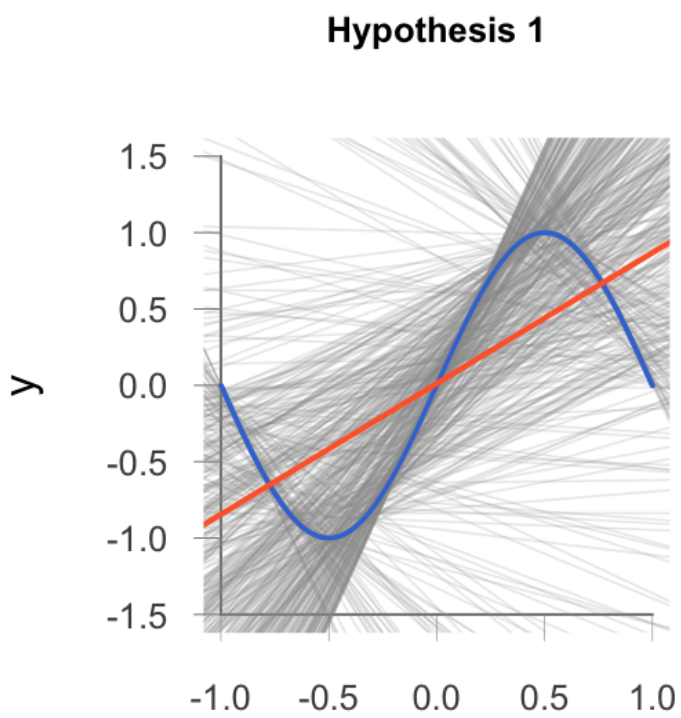




Learning from two points

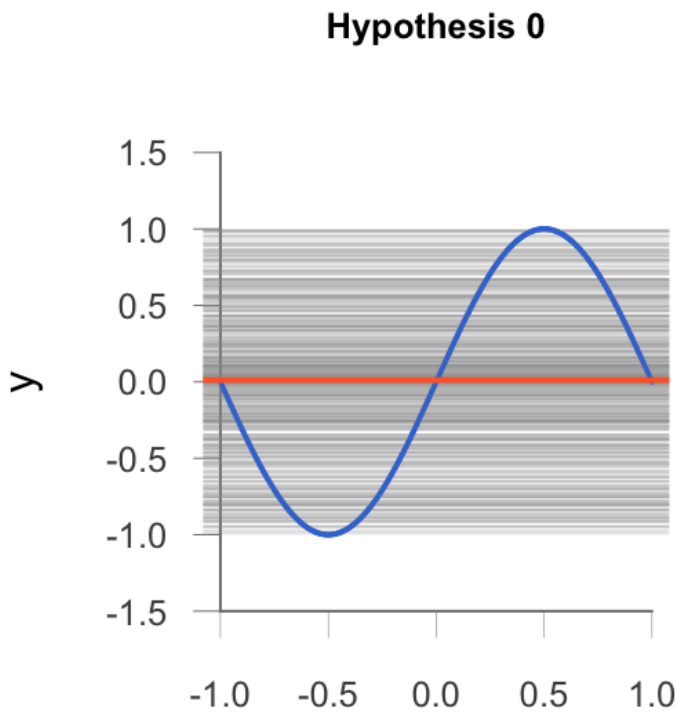
- For $\mathcal{H}_0$ models: if we average all 500 fitted models, we get $\bar{h}_0$ which corresponds to the horizontal line $y = 0$.
- All the individual fitted lines have the same slope of the average hypothesis, but different intercept values.
- The class of $\mathcal{H}_0$ models have **low variance** and **high bias**.

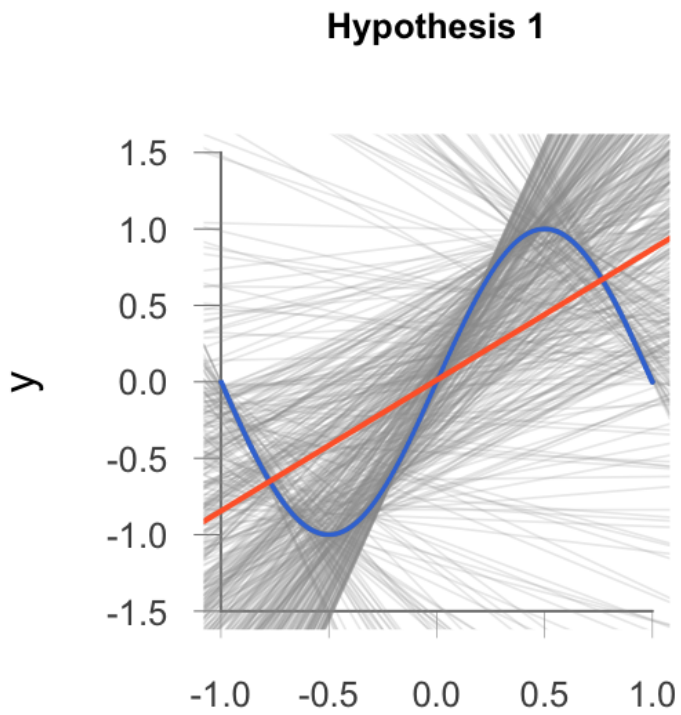




Learning from two points

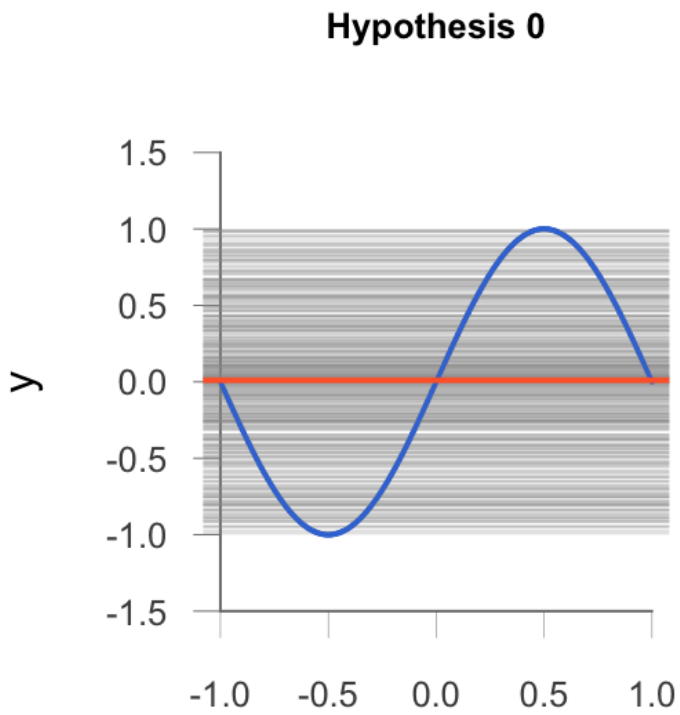
- For $\mathcal{H}_1$ models: if we average all 500 fitted models, we get $\bar{h}_1$ which corresponds to the orange line with positive slope.
- The individual fitted lines have all sorts of slopes: negative, close to zero, zero, positive, etc. There is a substantial amount of variability between the average hypothesis $\bar{h}_1$ and the form of any single fit h_1.

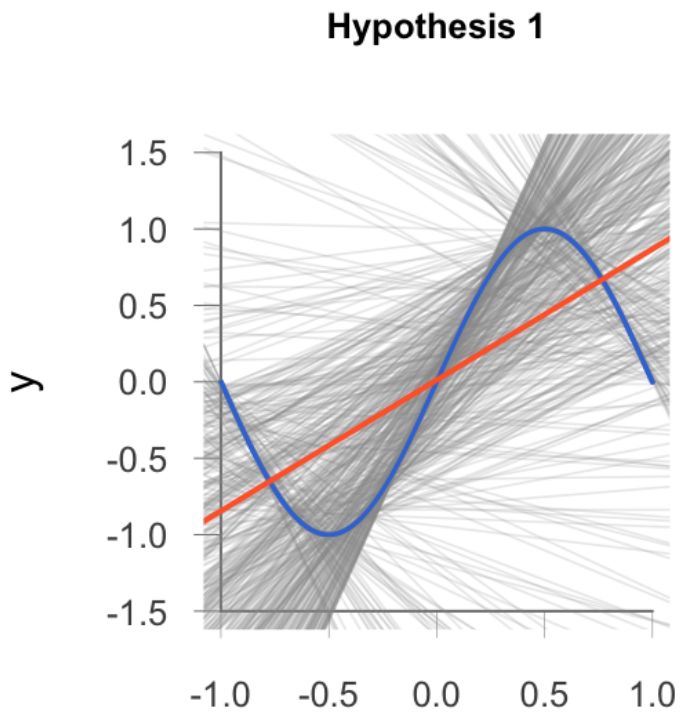




Learning from two points

- For $\mathcal{H}_1$ models: the fact that the average hypothesis $\bar{h}_1$ has positive slope means that the majority of fitted lines also have positive slopes.
- The average hypothesis somewhat matches the overall trend of target function $f()$ around $x \in [-0.5, 0.5]$.
- The class of $\mathcal{H}_1$ models have **high variance** and **low bias**.





Bias-Variance Derivation

- We know that the Mean Squared Error (MSE) of an estimator $\hat{\theta}$ can be decomposed in terms of bias and variance as:

$$\texttt{MSE}(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - \mu_{\hat{\theta}})^2] + (\mu_{\hat{\theta}} - \theta)^2$$

  with $\mu_{\hat{\theta}} = \mathbb{E}(\hat{\theta})$

- Bias, $\mu_{\hat{\theta}} - \theta$, is the tendency of $\hat{\theta}$ to overestimate or underestimate θ over all possible samples.
- Variance, $\texttt{Var}(\hat{\theta})$, measures the average variability of the estimators around their mean $\mathbb{E}(\hat{\theta})$.
- We have $\hat{\theta}$ above is a general estimator.
- Next, we focus on $\hat{f}()$, our approximation of a target function $f()$.

Out-of-Sample Predictions

- To consider the MSE as a theoretical expected value (i.e., not an empirical average), we suppose the existence of an out-of-sample data point $\mathbf{x}_0$.
- Given a learning data set $\mathcal{D}$ of n points, a hypothesis space $\mathcal{H}$ of hypotheses $h(\mathbf{x})$'s, the expectation of the Squared Error for a given out-of-sample point $\mathbf{x}_0$ **over all possible learning sets** is:

$$\mathbb{E}\left[\left(h^{(\mathcal{D})}(\mathbf{x}_0) - f(\mathbf{x}_0)\right)^2\right]$$

- We assume the target function $f()$ is **noiseless**.
- $h^{(\mathcal{D})}$ is obtained by fitting on a specific learning data set $\mathcal{D}$.
- $h^{(\mathcal{D})}(\mathbf{x}_0)$ denotes the predicted value of an out-of-sample point $\mathbf{x}_0$.
- $h^{(\mathcal{D})}(\mathbf{x}_0)$ plays the role of $\hat{\theta}$ and $f(\mathbf{x}_0)$ plays the role of θ.

Out-of-Sample Predictions

- We consider the average hypothesis $\bar{h}(\mathbf{x}_0)$ that plays the role of $\mu_{\hat{\theta}} = \mathbb{E}(\hat{\theta})$:

$$\bar{h}(\mathbf{x}_0) = \mathbb{E}_{\mathcal{D}}\left[h^{(\mathcal{D})}(\mathbf{x}_0)\right]$$

- We have the error for a given out-of-sample point $\mathbf{x}_0$:

$$h^{(\mathcal{D})}(\mathbf{x}_0) - f(\mathbf{x}_0) = h^{(\mathcal{D})}(\mathbf{x}_0) - \bar{h}(\mathbf{x}_0) + \bar{h}(\mathbf{x}_0) - f(\mathbf{x}_0)$$

$$h^{(\mathcal{D})} - f = h^{(\mathcal{D})} - \bar{h} + \bar{h} - f$$

- Same as above, we derive the expectation:

$$\mathbb{E}_{\mathcal{D}}\left[\left(h^{(\mathcal{D})} - f\right)^2\right] = \mathbb{E}_{\mathcal{D}}[(\underbrace{h^{(\mathcal{D})} - \bar{h}}_{a} + \underbrace{\bar{h} - f}_{b})^2]$$

$$= \mathbb{E}_{\mathcal{D}}[(a+b)^2] = \mathbb{E}_{\mathcal{D}}[a^2 + 2ab + b^2]$$

$$= \mathbb{E}_{\mathcal{D}}[a^2] + \mathbb{E}_{\mathcal{D}}[b^2] + \mathbb{E}_{\mathcal{D}}[2ab]$$

Out-of-Sample Predictions

- $\mathbb{E}_{\mathcal{D}}[a^2] = \mathbb{E}_{\mathcal{D}}\left[(h^{(\mathcal{D})} - \bar{h})^2\right] = \texttt{Variance}(h)$
- $\mathbb{E}_{\mathcal{D}}[b^2] = \mathbb{E}_{\mathcal{D}}\left[(\bar{h} - f)^2\right] = \texttt{Bias}^2(h)$
- We have

$$\begin{aligned}
\mathbb{E}_{\mathcal{D}}[2ab] &= \mathbb{E}_{\mathcal{D}}[2(h^{(\mathcal{D})} - \bar{h})(\bar{h} - f)] \\
&= 2\mathbb{E}_{\mathcal{D}}[h^{(\mathcal{D})}\bar{h} - h^{(\mathcal{D})}f - \bar{h}^2 + \bar{h}f] \\
&\propto \bar{h}\mathbb{E}_{\mathcal{D}}[h^{(\mathcal{D})}] - f\mathbb{E}_{\mathcal{D}}[h^{(\mathcal{D})}] - \mathbb{E}_{\mathcal{D}}[\bar{h}^2] + f\mathbb{E}_{\mathcal{D}}[\bar{h}] \\
&= \bar{h}^2 - f\bar{h} - \bar{h}^2 + f\bar{h} = 0
\end{aligned}$$

- Thus, assume that the target function $f()$ is **noiseless**, we have the expectation of the Squared Error for a given out-of-sample point $\mathbf{x}_0$, **over all possible learning sets**, is:

$$\mathbb{E}_{\mathcal{D}}\left[\left(h^{(\mathcal{D})}(\mathbf{x}_0) - f(\mathbf{x}_0)\right)^2\right] = \underbrace{\mathbb{E}_{\mathcal{D}}\left[\left(h^{(\mathcal{D})}(\mathbf{x}_0) - \bar{h}(\mathbf{x}_0)\right)^2\right]}_{\text{variance}} + \underbrace{\left(\bar{h}(\mathbf{x}_0) - f(\mathbf{x}_0)\right)^2}_{\text{bias}^2}$$

Noisy Targets

- When there is noise in the data, we have:

$$y = f(\mathbf{x}) + \epsilon$$

- If ϵ is zero-mean noise random variable with variance σ^2, the bias-variance decomposition becomes:

$$\mathbb{E}_{\mathcal{D}}\left[\left(h^{(\mathcal{D})}(\mathbf{x}_0) - y_0\right)^2\right] = \texttt{var} + \texttt{bias}^2 + \sigma^2$$

- Notice that the above equation involves the squared error corresponds to just one out-of-sample point $(\mathbf{x}_0, y_0)$ (i.e., test point).

Types of Theoretical MSEs

1. MSE involves a single out-of-sample point $\mathbf{x}_0$, measuring the performance of a class of hypotheses $h \in \mathcal{H}$ over multiple learning data sets $\mathcal{D}$ - expected test MSE.

$$\mathbb{E}_{\mathcal{D}}\left[\left(h^{(\mathcal{D})}(\mathbf{x}_0) - f(\mathbf{x}_0)\right)^2\right]$$

2. MSE involves a single hypothesis $h()$, measuring its performance over all out-of-sample points $\mathbf{x}_0$. Notice that $h()$ has been fitted on just one learning set $\mathcal{D}$

$$\mathbb{E}_{\mathcal{X}}\left[(h(\mathbf{x}_0) - f(\mathbf{x}_0))^2\right]$$

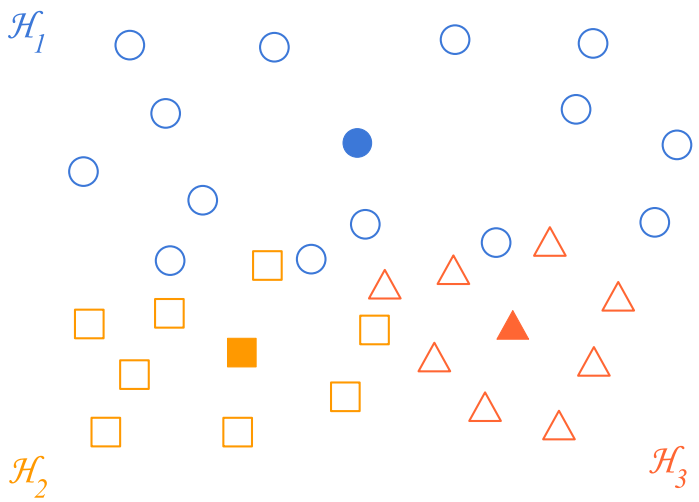
3. MSE measures the performance of a class of hypotheses $h \in \mathcal{H}$, over multiple learning data sets $\mathcal{D}$, over all out-of-sample points $\mathbf{x}_0$ - overall expected test MSE.

$$\mathbb{E}_{\mathcal{X}}\left[\mathbb{E}_{\mathcal{D}}\left[\left(h^{(\mathcal{D})}(\mathbf{x}_0) - f(\mathbf{x}_0)\right)^2\right]\right]$$

Types of Theoretical MSEs

- These types of MSEs are highly theoretical.
- First, we don't know the target function f.
- Second, we don't have access to all out-of-sample points.
- Third, we cannot have infinite learning sets to compute the average hypothesis $\bar{h}$.
- We try to compute an approximation (i.e., an estimate) of an MSE using a test dataset $\mathcal{D}_{test}$.
- $\mathcal{D}_{test}$ is assumed to be a representative subset (i.e., an **unbiased** sample) of the out-of-sample data $\mathcal{D}_{out}$.

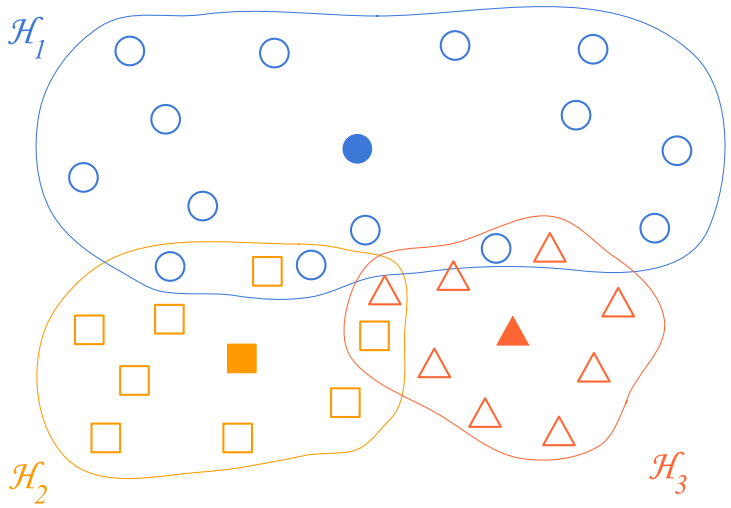
The Bias-Variance Tradeoff Picture



For example, we consider several classes of hypothesis:

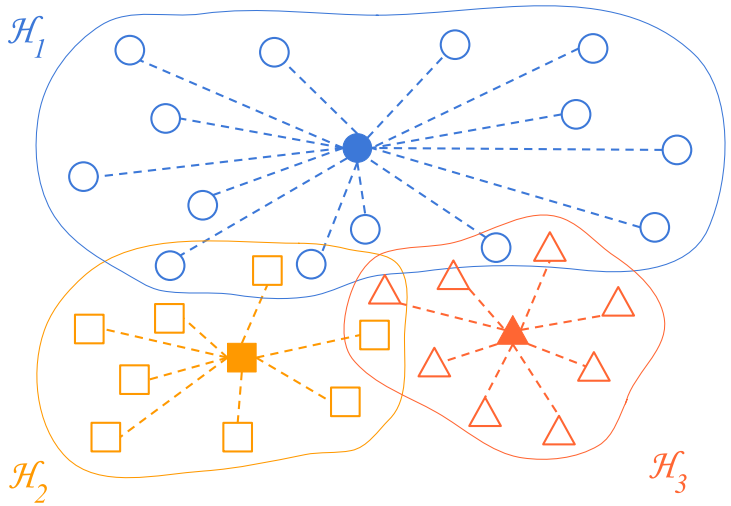
- $\mathcal{H}_1$ of cubic polynomials.
- $\mathcal{H}_2$ of quadratic models.
- $\mathcal{H}_3$ of linear models.

The Bias-Variance Tradeoff Picture



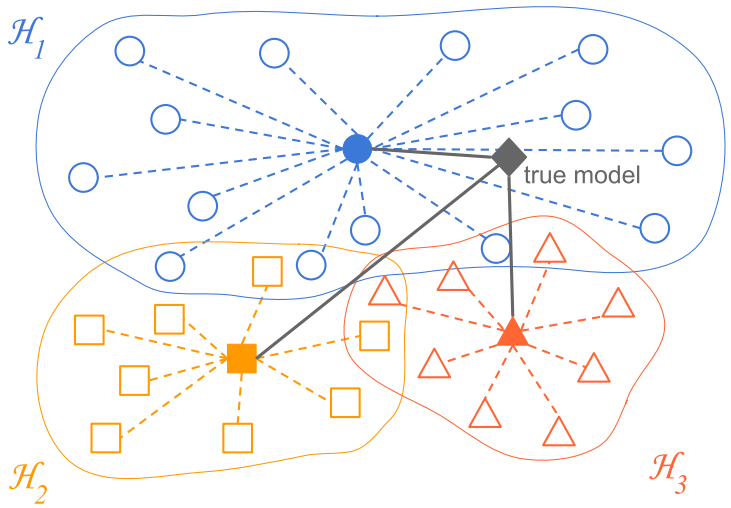
- Each non-filled point is a fitted model $h^{(\mathcal{D})}$ based on some particular dataset $\mathcal{D}$
- Each filled point is the average model of a particular class of hypotheses.
- For example, if $\mathcal{H}_3$ represents the fits based on linear models, each triangle is a linear polynomial $ax + b$ with coefficients a, b that change depending on the sample.

The Bias-Variance Tradeoff Picture



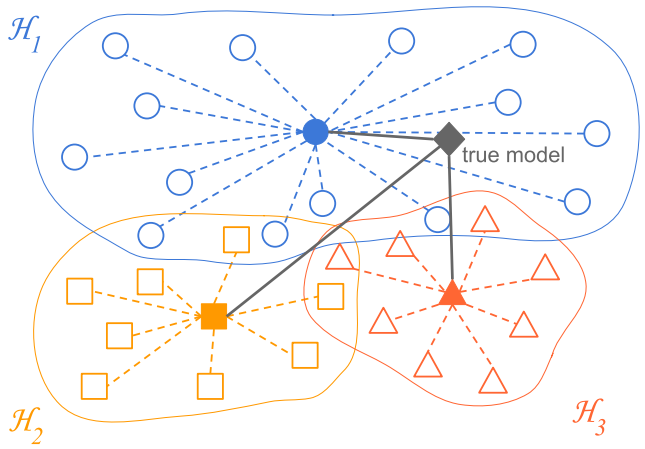
- We measure the variability in each class of models.
- The dashed lines represent the deviations of each fitted model against their average hypothesis.
- The set of all dashed lines shows the **variance** in each class of models, i.e., how spread out the models within each class are.

The Bias-Variance Tradeoff Picture



- Suppose we can locate the true model $f()$ in this space. Assume $f()$ to be of class $\mathcal{H}_1$.
- The solid lines between each average hypothesis and the target function represents the **bias** of each class of model.
- Notice that in practice we don't have access to either the average models $\bar{h}()$ or the true model $f()$.

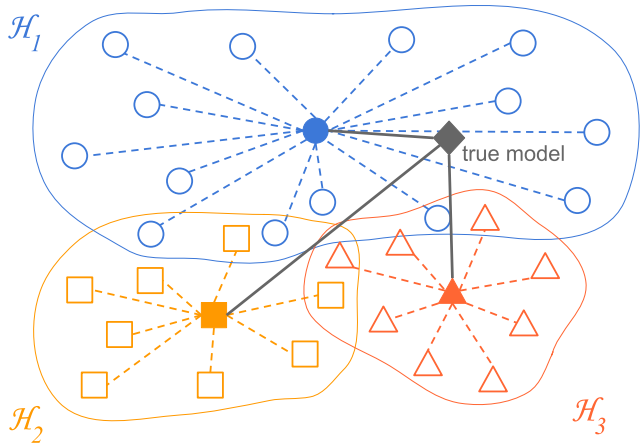
Bias



- The bias term involves $\bar{h}(\mathbf{x}) - f(\mathbf{x})$. The average $\bar{h}$ comes from a hypothesis class $\mathcal{H}$ (e.g., constant models, linear models, etc.).
- $\bar{h}$ is a prototypical example of a certain class of hypotheses.
- The bias term measures how well a hypothesis class $\mathcal{H}$ approximates the target function f.

$$\texttt{MSE} = \texttt{Variance} + \underbrace{\texttt{Bias}^2}_{\text{deterministic noise}} + \underbrace{\texttt{Noise}}_{\text{random noise}}$$

Variance



- The variance term $\mathbb{E}_{\mathcal{D}}\left[(h^{(\mathcal{D})}(\mathbf{x}) - \bar{h})^2\right]$ measures how close a particular hypothesis $h^{(\mathcal{D})}(\mathbf{x})$ can get to the average hypothesis $\bar{h}$.
- The variance term measures how precise is our particular function $h^{(\mathcal{D})}(\mathbf{x})$ compared to the average function $\bar{h}(\mathbf{x})$

Tradeoff

- A model should have both small variance and small bias. But, there is a tradeoff between these two (Bias-Variance tradeoff).
- To actually perform bias-variance decomposition, we need access to $\bar{h}$. But, computing $\bar{h}$ requires computing every model of a hypothesis class $\mathcal{H}$ (e.g., linear, quadratic, etc.)
- More complex/flexible models tend to have less bias, and thus have a better chance to approximate $f(\mathbf{x})$. Complex models tend to have small in-sample error: $E_{in} \approx 0$.
- More complex models tend to have higher variance. They have a higher risk of large out-of-sample error: $E_{out} \gg 0$. We need more resources (training data, computing powers).
- Less complex models tend to have less variance, but more bias, i.e., smaller chance to estimate $f()$, but higher chance to approximate out-of-sample error: $E_{in} \approx E_{out}$, but $E_{in} \approx E_{out} \gg 0$.

Tradeoff

- To decrease bias, we might need "insider" information. That is, to truly decrease bias, we need some information on the form of the unknown target function $f()$.
- It is thus nearly impossible to have zero bias.
- Therefore, to decrease bias, we tend to use more complex/flexible models. We need to put our efforts toward decreasing variance.
  - Adding more training data.
  - Reduce the dimensions of data (e.g. lower-rank data matrices through PCA).
  - Apply regularizations, make the size of model parameters smaller.

Overfitting

- In supervised learning, one of the major risks when fitting a model is to overestimate how well it will do when we use it in the real world. This risk is commonly called **overfitting**.
- Analogy to students' studying before taking a test.
  - Limited capacity: We only grasp the general idea for some topics, and do not understand the details. For example, we know the concept of simple linear regression, but don't know how to derive the formula.
  - Too much focus: We focus too much on certain topics, memorize most of their details, but ignore other topics. For example, we memorize the Normal Equations in linear regression model, but ignore the projection or the probabilistic perspectives.
  - Distraction by "noise": phone notifications, noise from the neighbors, etc.

Bias-Variance

- We assume a response variable $y = f(\mathbf{x}) + \epsilon$.
- We seek a model $h(\mathbf{x})$ that approximates well the target $f()$.
- Given a learning dataset $\mathcal{D}$ of n points, and a hypothesis $h(\mathbf{x})$, the expectation of the Squared Error for a given out-of-sample point $\mathbf{x}_0$, over all possible learning sets, is:

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}\left[\left(h^{(\mathcal{D})}(\mathbf{x}_0) - f(\mathbf{x}_0)\right)^2\right] &= \underbrace{\mathbb{E}_{\mathcal{D}}\left[\left(h^{(\mathcal{D})}(\mathbf{x}_0) - \bar{h}(\mathbf{x}_0)\right)^2\right]}_{\text{variance}} \\ &+ \underbrace{\left(\bar{h}(\mathbf{x}_0) - f(\mathbf{x}_0)\right)^2}_{\text{bias}^2} \\ &+ \underbrace{\sigma^2}_{\text{noise}} \\ &= \texttt{variance} + \texttt{bias}^2 + \sigma^2\end{aligned}$$

where $\bar{h}(\mathbf{x}_0) = \mathbb{E}_{\mathcal{D}}[h^{(\mathcal{D})}(\mathbf{x}_0)]$ is the average hypothesis.

Bias-Variance

- Large Bias: "limited learning capacity" - the class of models $\mathcal{H}$ has too little capacity to get close enough to the true model.
- Large Variance: focusing too much on certain details at the expense of other equally or more important details.
- Large Noise: distracted by the noise in the data - The data for training to obtain $h^{(\mathcal{D})}$ is bad: messy, missing values, poor quality.

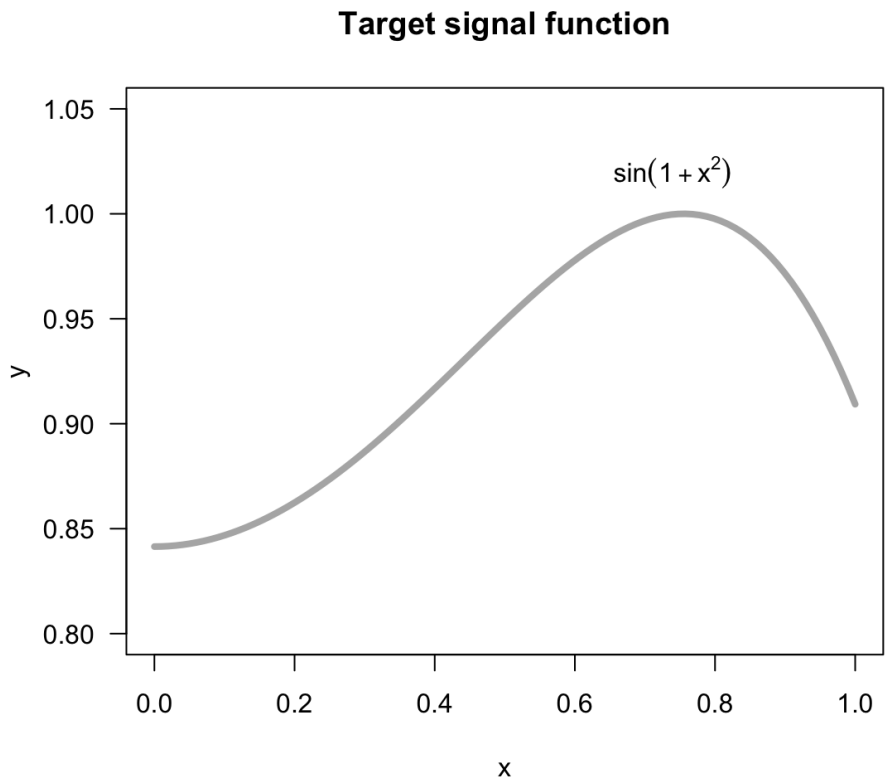
Example

- We consider a target function with some noise:

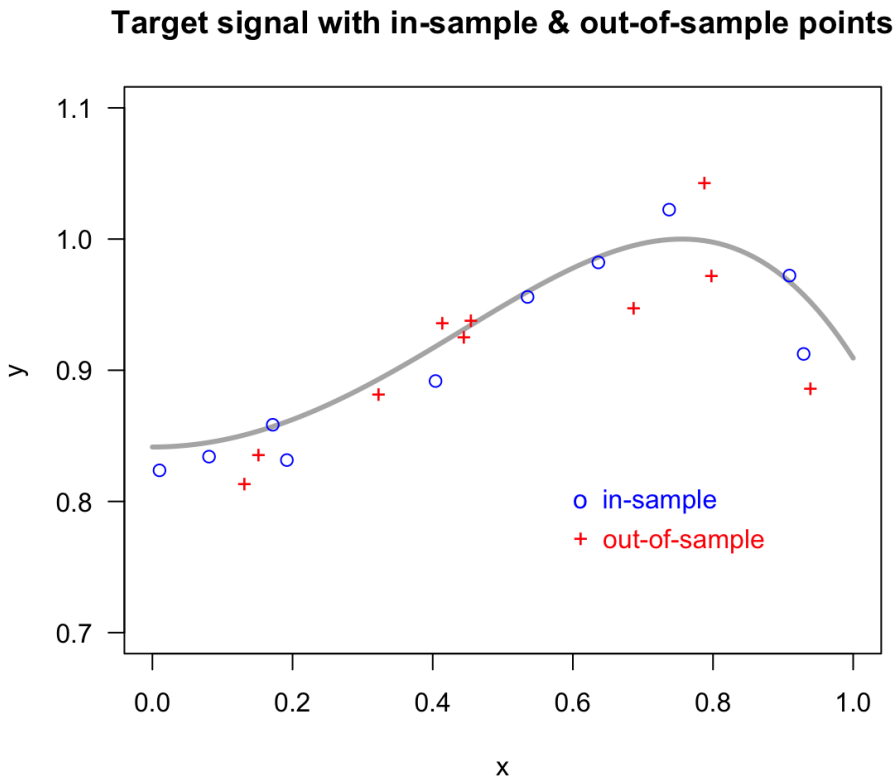
$$f(x) = \sin(1 + x^2) + \epsilon$$

with input variable $x \in [0, 1]$ and the noise term $\epsilon \sim N(\mu = 0, \sigma = 0.03)$.

- The function $\sin(1 + x^2)$ is:

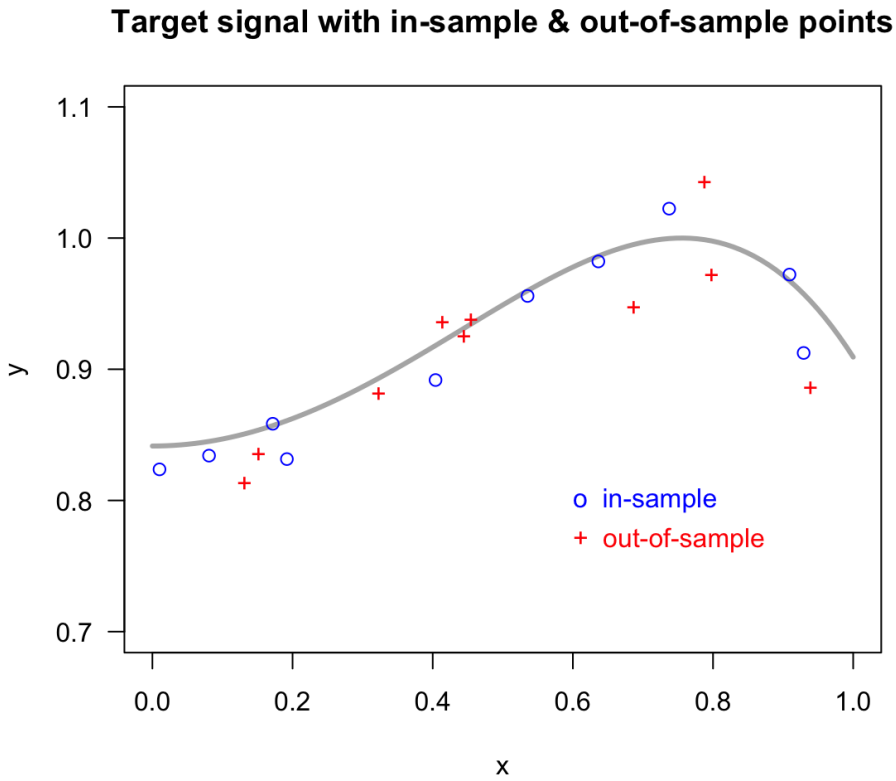


Example



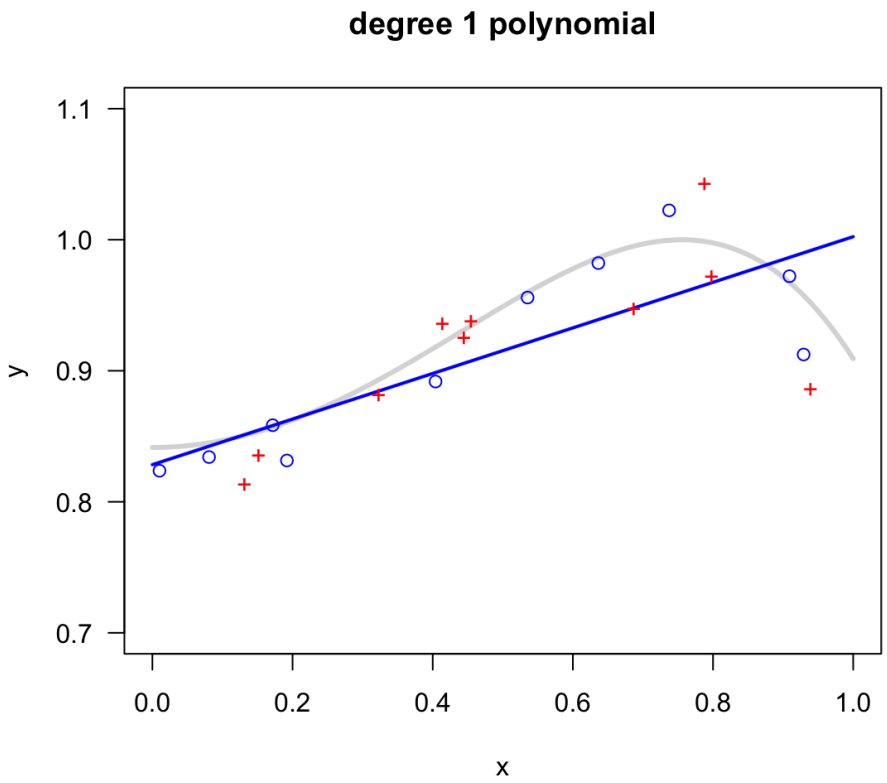
- We sample one in-sample set of 10 points (x_i, y_i) and one out-of-sample set of 10 points (x_0, y_0).
- Notice that a true out-of-sample set should include all $x \in [0, 1]$.

Example



1. First, we fit a linear model (i.e., degree 1 polynomial)
2. Second, we fit a second degree polynomial
3. Third, we fit a third degree polynomial
4. ...
5. With 10 in-sample points, we can fit a 9 degree polynomial.

Example - Linear Model

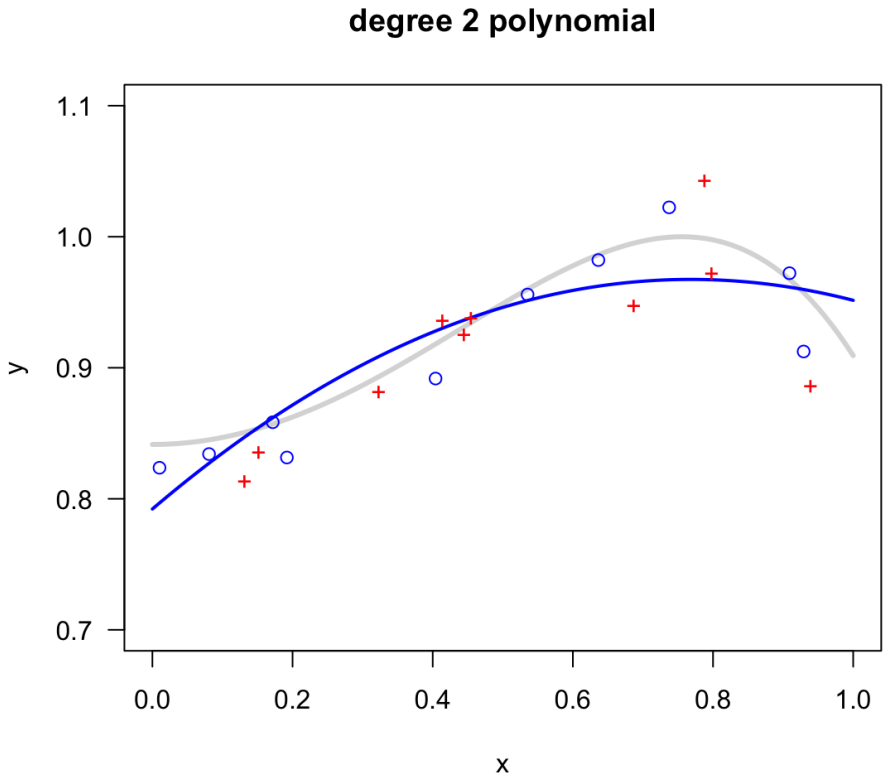


1. The first fitted model is a linear model of the form:

$$h_1(x) = b_0 + b_1 x$$

2. The fitted regression model is the blue line.
3. $E_{in} = 0.00147$ and $E_{out} = 0.00215$

Example - Quadratic Model

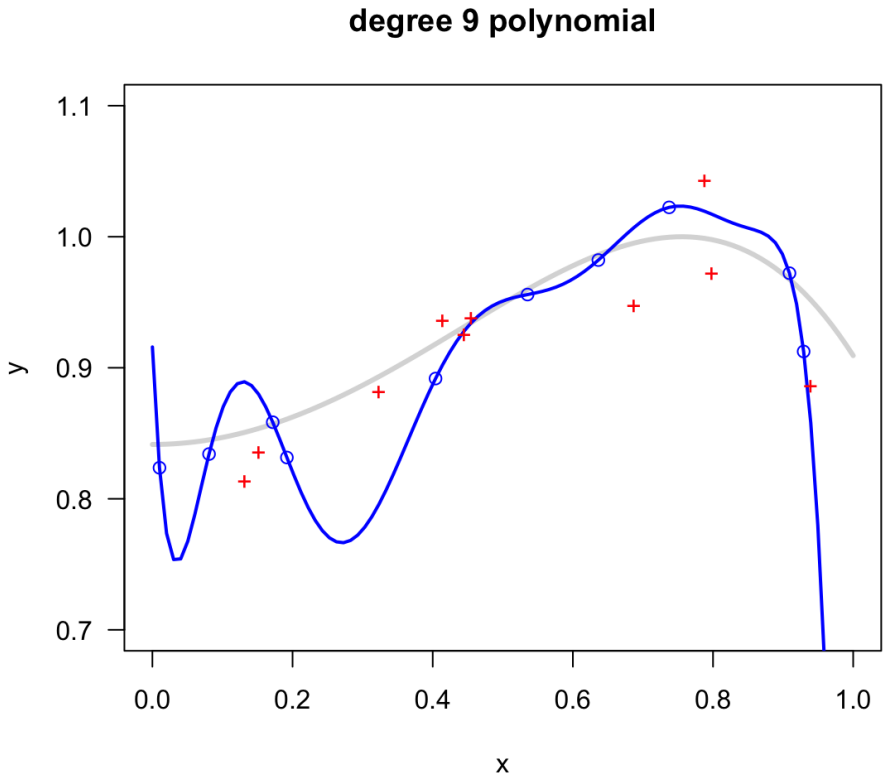


1. The second fitted model is a quadratic model of the form:

$$h_2(x) = b_0 + b_1 x + b_2 x^2$$

2. The fitted regression model is the blue curve.
3. $E_{in} = 0.00093$ and $E_{out} = 0.00137$

Example - Quadratic Model

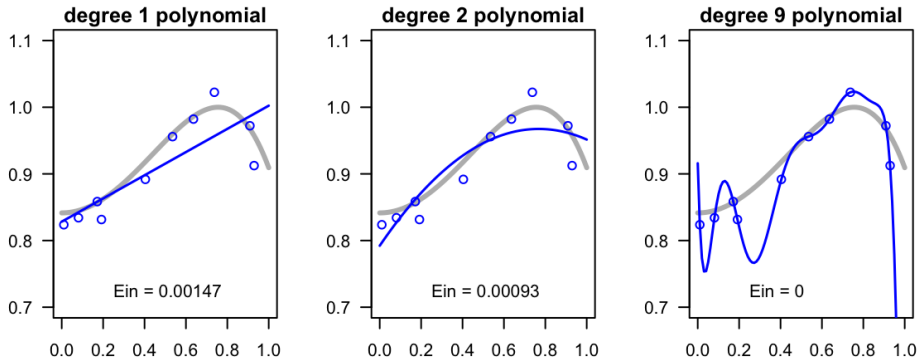


1. The 9-th fitted model is a nonic model of the form:

$$h_9(x) = b_0 + b_1 x + b_2 x^2 + \ldots + b_8 x^8 + b_9 x^9$$

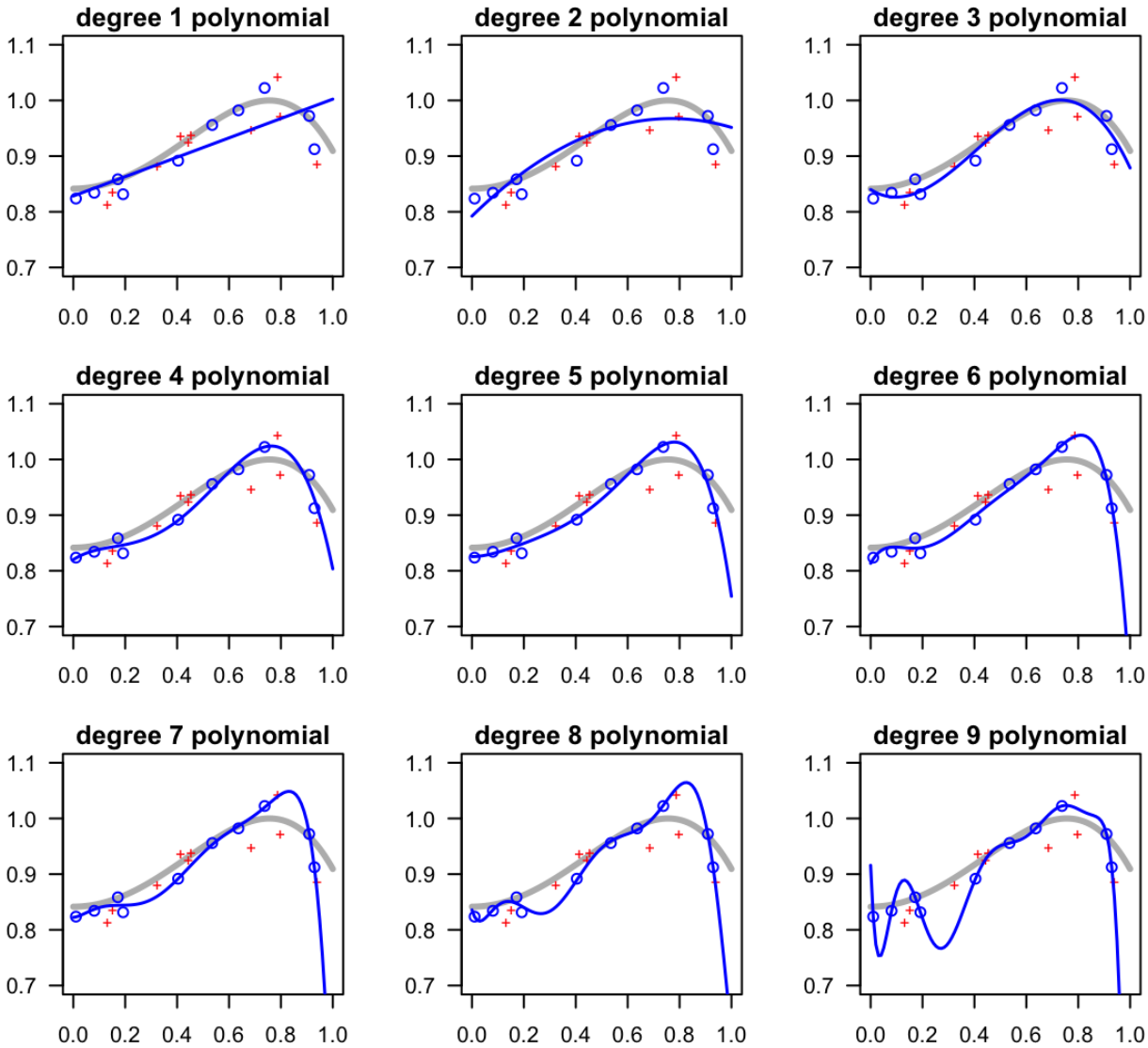
2. The fitted regression model is the blue curve.
3. $E_{in} = 0.00000$ and $E_{out} = 0.00231$

Example - Which Model?



- The 9-degree polynomial achieves $E_{in} = 0.0$
- Should we choose $h_9(x)$ as the final model? Because this is the model with the perfect fit to the learning data.
- No! We should consider their out-of-sample error E_{out}.

Example - Which Model?

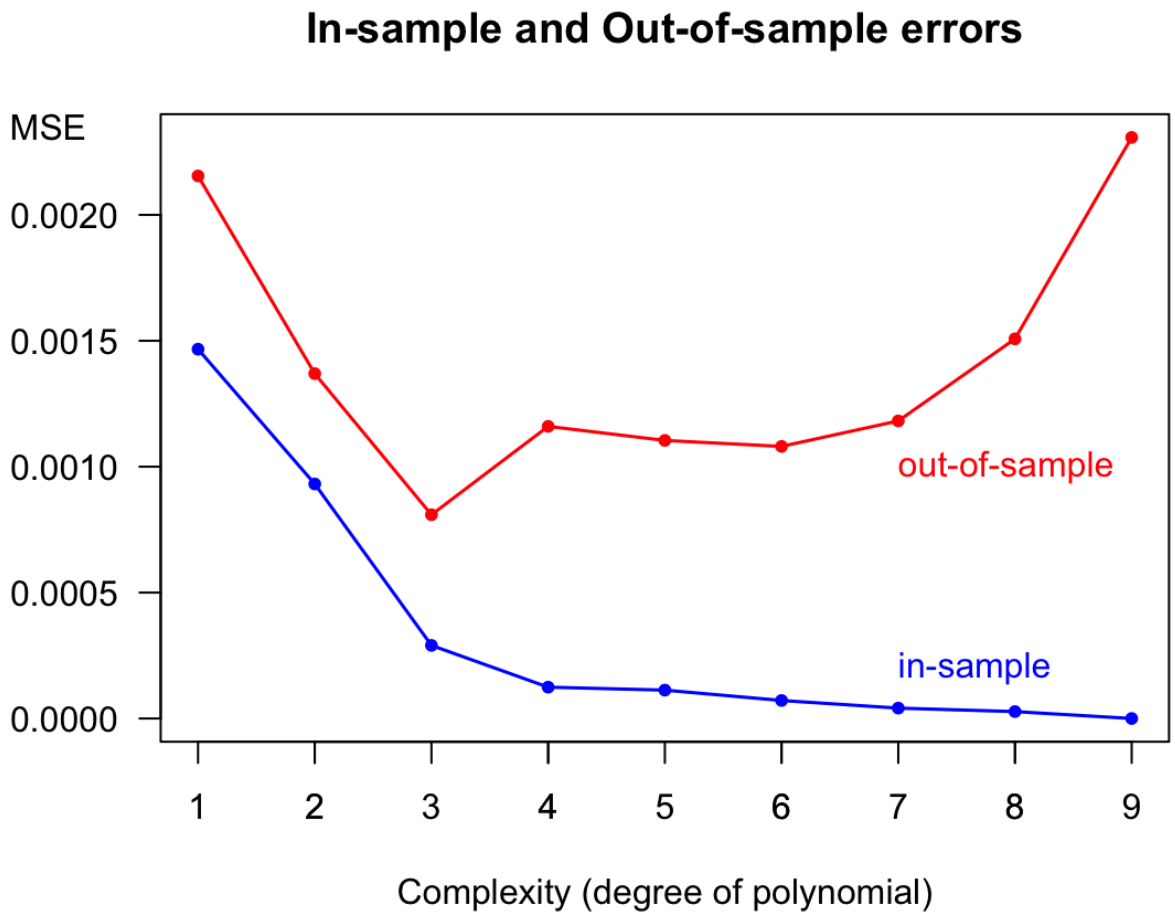


Example - Which Model?

Degree	E_{in}	E_{out}
1	0.00147	0.00215
2	0.00093	0.00137
3	0.00029	0.00081
4	0.00012	0.00116
5	0.00011	0.00110
6	0.00007	0.00108
7	0.00004	0.00118
8	0.00003	0.00151
9	0.00000	0.00231

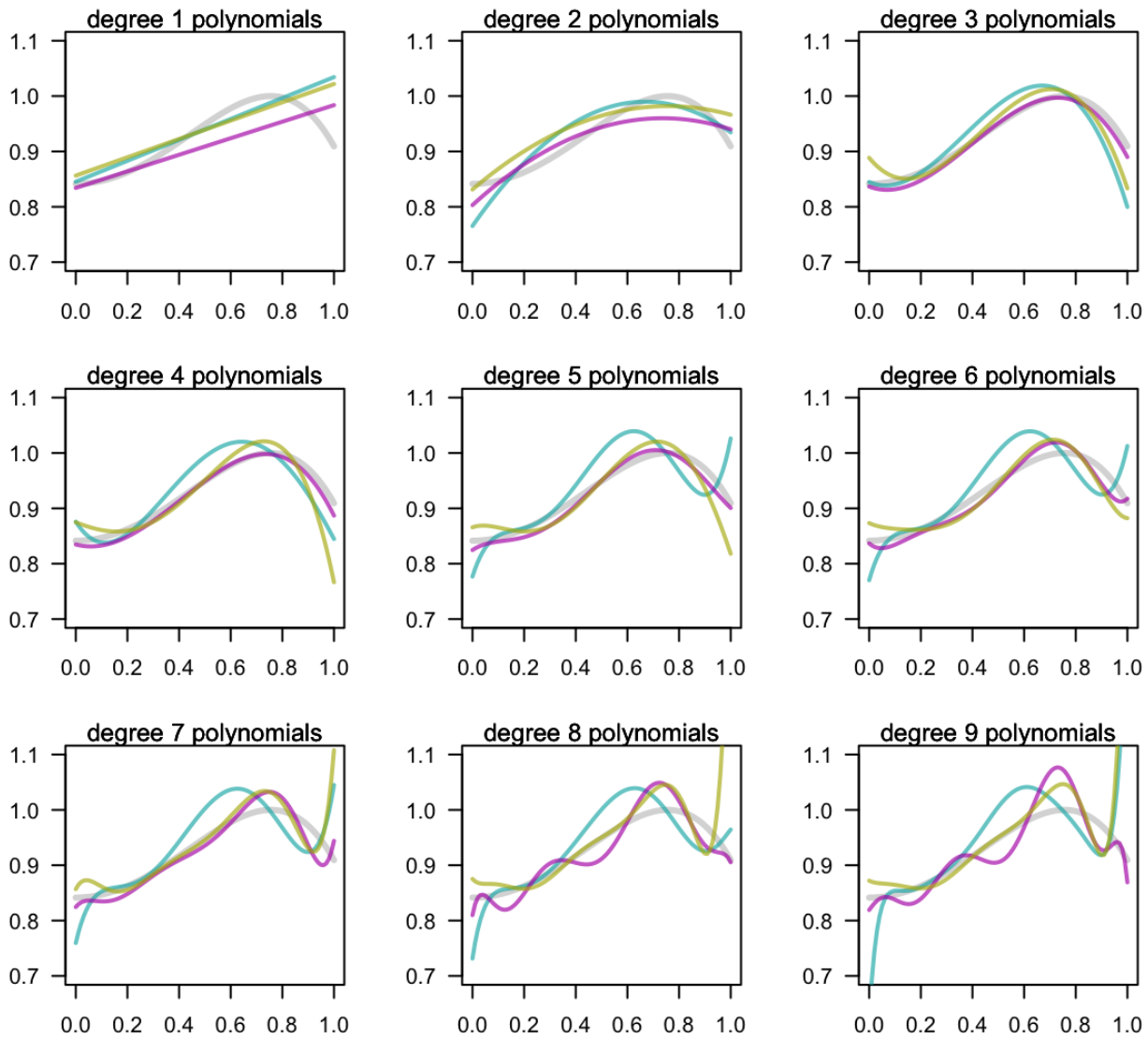
- The linear model underfits the data. Among all models, its in-sample error is the largest. A more complex model will reduce E_{in} and E_{out}.
- The quadratic model is better than the linear model, but still misses the shape of the signals. It still underfits.
- Polynomials of degrees 3, 4, 5 are "okayfit".
- The 9-th degree polynomial is too flexible. Its $E_{in} = 0.0$, but it produces very large E_{out}. This model overfits.

Example - Which Model?



- Overfitting means that an attractively small E_{in} value is no longer a good indicator of a model's out-of-sample performance.

Example - More Learning Sets



- Three new learning sets of size $n = 10$.
- For each learning set, the 9-degree polynomials can fit perfectly. But they are very volatile.
- The 1,2,3-degree polynomials are more stable.

Overfitting and Underfitting

- Overfitting happens when we fit the data more than is necessary. Overfitting is when we choose a model with smaller E_{in} but it turns out in bigger E_{out}. The in-sample error is no longer a good indicator for the chosen model's generalization.
- Underfitting occurs with models that perform poorly on unseen data because of their lack of capacity/flexibility. Underfit models are not of the right class, they suffer from **large bias**.
- Low complexity models tend to be biased. with more complex models the amount of bias decreases. But it is generally impossible to know the true class of model for the target function.
- NOTICE: If none of the proposed hypothesis classes contain the truth, they will all be biased.

Overfitting and Underfitting

- The amount of bias does not depend on the size of the in-sample set.
- This means that increasing the number of learning points won't give us a better chance to approximate $f()$.
- The amount of variance of the model does depend on the number of learning points.
- As n increases, large-capacity models will experience a reduction in variability, and their higher flexibility tends to become an advantage.

Overfitting and Underfitting