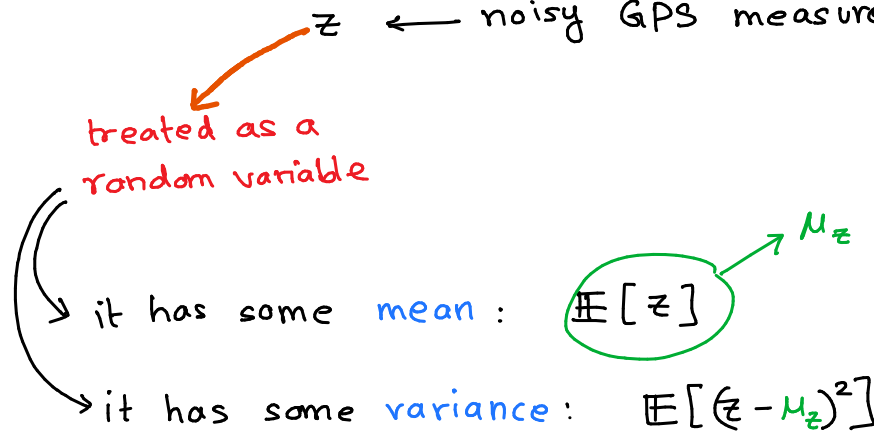


Lecture 10: Bias - Variance Decomposition

Concept of BIAS and VARIANCE

- Consider an example: $z_0 \leftarrow$ true location of an object
- $z \leftarrow$ noisy GPS measurements of the location



- Bias** describes the systematic error in the measurements z (possible offset)

$$\text{BIAS: } \mu_z - z_0$$

- Variance** describes how much the measurements vary (amount of noise in GPS measurements)

$$\text{VARIANCE: } \mathbb{E}[(z - \mu_z)^2] = \mathbb{E}[z^2] - \mu_z^2$$

- Bias : $\mu_z - z_0$

Variance : $\mathbb{E}[z^2] - \mu_z^2$

- Squared error between measurement and true value : $(z - z_0)^2$

- Expected squared error : $\mathbb{E}[(z - z_0)^2]$
(Averaged)

$$\begin{aligned} & \Downarrow \\ &= \mathbb{E} \left[\left((z - \mu_z) + (\mu_z - z_0) \right)^2 \right] \\ &= \mathbb{E} \left[(z - \mu_z)^2 \right] + \mathbb{E} \left[(\mu_z - z_0)^2 \right] + 2 \mathbb{E} \left[(z - \mu_z)(\mu_z - z_0) \right] \\ &= \underbrace{\mathbb{E} \left[(z - \mu_z)^2 \right]}_{\text{variance}} + \underbrace{(\mu_z - z_0)^2}_{\text{bias}} + 2(\mu_z - z_0) \underbrace{\left(\mathbb{E}[z] - \mu_z \right)}_0 \end{aligned}$$

constant *constant* *constant* *constant*

- In other words, the averaged squared error between z and z_0 is the sum of the **squared bias** and **variance**

- To obtain a small expected squared error, we have to consider both **bias** and **variance**

- We will now apply the bias-variance concept to a regression setting
 - z_0 will now correspond to the true relationship between inputs and outputs
 - random variable z will correspond to the model learned from training data since training data T is a random collection from $p(\underline{x}, y)$, the model z learned from it is also random as it is a function of training data T
- Let the true relationship between input \underline{x} and output y be described by some function $f_0(\underline{x})$ plus i.i.d. noise ϵ

True model $\rightarrow y = \underbrace{f_0(\underline{x})}_{\text{similar to } z_0} + \epsilon, \quad \text{with } \mathbb{E}[\epsilon] = 0$
 $\text{Var}[\epsilon] = \sigma^2$

- Learned model is random variable; therefore model prediction $\hat{y}(\underline{x}; T)$ is r.v.!

Learned model $\rightarrow \underbrace{\hat{y}(\underline{x}; T)}_{\text{similar to } z} \leftarrow \text{prediction from model trained on dataset } T$

- Let the true relationship between input \underline{x} and output y be described by some function $f_0(\underline{x})$ plus i.i.d. noise ϵ

True model

$$y = f_0(\underline{x}) + \epsilon, \quad \text{with } \mathbb{E}[\epsilon] = 0$$

$\text{Var}[\epsilon] = \sigma^2$

$\xrightarrow{\text{wavy line}} 'z_0'$

- The learned model is a r.v.; therefore model prediction $\hat{y}(\underline{x}; \mathcal{T})$ is r.v.!
- Define average trained model corresponding to $\bar{\epsilon}$:

$$\bar{f}(\underline{x}) = \mathbb{E}_{\mathcal{T}} [\hat{y}(\underline{x}; \mathcal{T})]$$

\uparrow
expected value

$\xrightarrow{\text{wavy line}} 'z'$

average model
we would achieve if we
could re-train the model
infinite # of times on different
training datasets, each of size N

over N training points
drawn from $p(\underline{x}, y)$

— Recall the definition of \bar{E}_{new} (for regression with squared error)

$$E_{\text{new}} = \mathbb{E}_* \left[(y_* - \hat{y}(x_*; \tau))^2 \right]$$

$$\bar{E}_{\text{new}} = \mathbb{E}_\tau [E_{\text{new}}] = \mathbb{E}_\tau \left[\mathbb{E}_* \left[(y_* - \hat{y}(x_*; \tau))^2 \right] \right]$$

— Change the order of integration

$$\bar{E}_{\text{new}} = \mathbb{E}_* \left[\mathbb{E}_\tau \left[(y_* - \hat{y}(x_*; \tau))^2 \right] \right]$$

replace $y_* = f_0(x_*) + \epsilon$

$$= \mathbb{E}_* \left[\mathbb{E}_\tau \left[(f_0(x_*) + \epsilon - \hat{y}(x_*; \tau))^2 \right] \right]$$

$$= \mathbb{E}_* \left[\mathbb{E}_\tau \left[(\hat{y}(x_*; \tau) - f_0(x_*) - \epsilon)^2 \right] \right]$$

$$= \mathbb{E}_* \left[\mathbb{E}_\tau \left[(\hat{y}(x_*; \tau) - \bar{f}(x_*) + \bar{f}(x_*) - f_0(x_*) - \epsilon)^2 \right] \right]$$

$\bar{f}(x) = \mathbb{E}_\tau [\hat{y}(x; \tau)]$

$$\overline{E}_{\text{new}} = \mathbb{E}_* \left[\mathbb{E}_T \left[\underbrace{\left(\hat{y}(x_*; \tau) - \bar{f}(x_*) \right)}_{A_1} + \underbrace{\bar{f}(x_*) - f_0(x_*)}_{A_2} - \underbrace{\epsilon}_{A_3} \right]^2 \right]$$

$$= \mathbb{E}_* \left[\mathbb{E}_T \left[(A_1 + A_2 + A_3)^2 \right] \right] = \mathbb{E}_* \left[\mathbb{E}_T \left[A_1^2 + A_2^2 + A_3^2 + 2(A_1 A_2 + A_2 A_3 + A_3 A_1) \right] \right]$$

$$\begin{aligned} \mathbb{E}_* \left[\mathbb{E}_T [A_1 A_2] \right] &= \mathbb{E}_* \left[\mathbb{E}_T \left[\left(\hat{y}(x_*; \tau) - \bar{f}(x_*) \right) \left(\bar{f}(x_*) - f_0(x_*) \right) \right] \right] \\ &= \mathbb{E}_* \left[\left(\bar{f}(x_*) - f_0(x_*) \right) \left(\underbrace{\mathbb{E}_T [\hat{y}(x_*; \tau)]}_{\bar{f}(x_*)} - \bar{f}(x_*) \right) \right] = 0 \end{aligned}$$

$$\mathbb{E}_* \left[\mathbb{E}_T [A_2 A_3] \right] = \mathbb{E}_* \left[\mathbb{E}_T \left[\left(\bar{f}(x_*) - f_0(x_*) \right) \epsilon \right] \right] = \mathbb{E}_* \left[\left(\bar{f}(x_*) - f_0(x_*) \right) \mathbb{E}[\epsilon] \right] = 0$$

$$\begin{aligned} \mathbb{E}_* \left[\mathbb{E}_T [A_3 A_1] \right] &= \mathbb{E}_* \left[\mathbb{E}_T \left[\epsilon \left(\hat{y}(x_*; \tau) - \bar{f}(x_*) \right) \right] \right] \quad (\text{Noise is independent of model}) \\ &= \mathbb{E}_* \left[\mathbb{E}_T [\epsilon] \right] \cdot \mathbb{E}_* \left[\mathbb{E}_T \left[\left(\hat{y}(x_*; \tau) - \bar{f}(x_*) \right) \right] \right] = 0 \end{aligned}$$

$$\bullet \mathbb{E}_* \left[\mathbb{E}_\tau [A_1^2] \right] = \mathbb{E}_* \left[\underbrace{\mathbb{E}_\tau \left[\left(\underbrace{\hat{y}(x_*; \tau)}_{\text{'z'}} - \underbrace{\bar{f}(x_*)}_{\text{'}\mu_z\text{'}} \right)^2 \right]}_{\text{Variance}} \right] \quad \left(\mathbb{E}[(z - \mu_z)^2] \right)$$

describes how much $\hat{y}(x; \tau)$ ^{prediction} varies each time the model is trained on a different training dataset

$$\begin{aligned} \bullet \mathbb{E}_* \left[\mathbb{E}_\tau [A_2^2] \right] &= \mathbb{E}_* \left[\mathbb{E}_\tau \left[\left(\bar{f}(x_*) - f_0(x_*) \right)^2 \right] \right] \\ &= \mathbb{E}_* \left[\underbrace{\left(\underbrace{\bar{f}(x_*)}_{\text{'}\mu_z\text{'}} - \underbrace{f_0(x_*)}_{\text{'}z_0\text{'}} \right)^2}_{\text{Bias}^2} \right] \quad \left((\mu_z - \mu_0)^2 \right) \end{aligned}$$

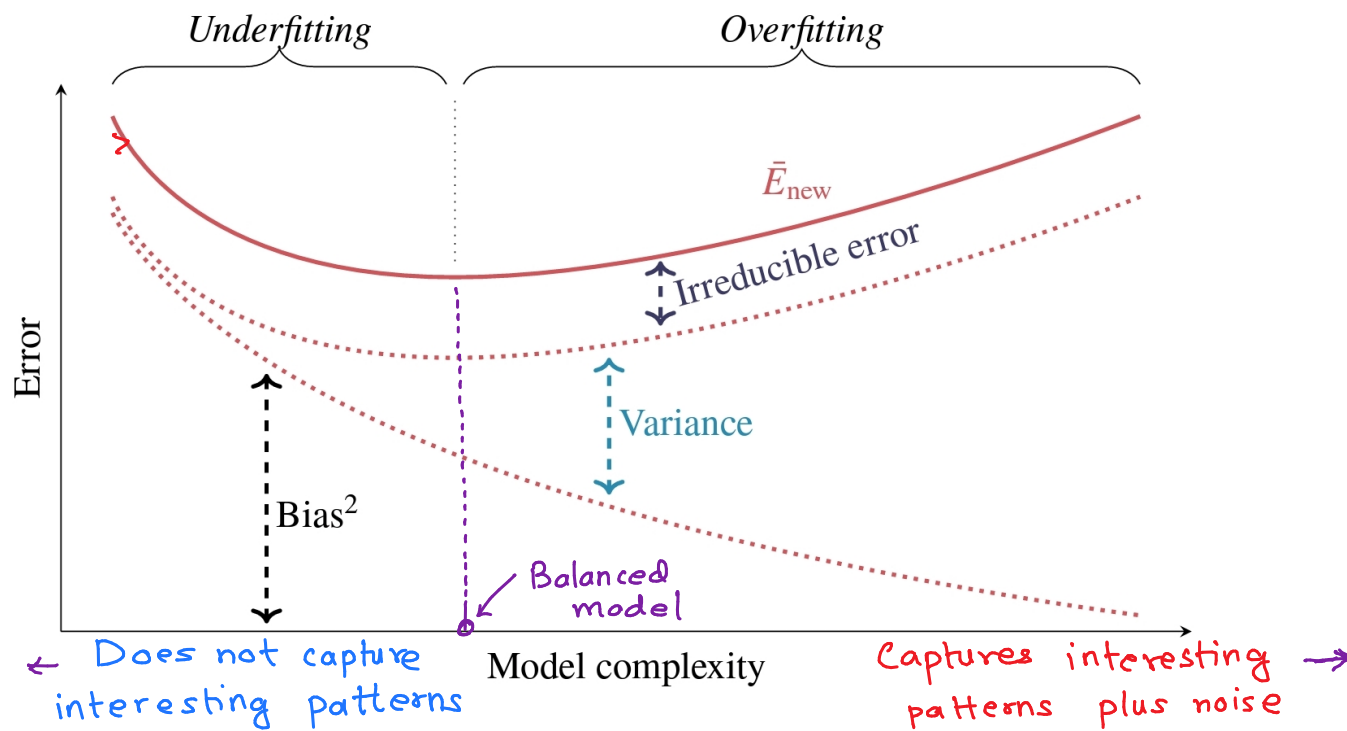
describes how much the average trained model $\bar{f}(x_*)$ differs from the true $f_0(x_*)$

$$\bullet \mathbb{E}_* \left[\mathbb{E}_\tau [A_3^2] \right] = \mathbb{E}_* \left[\mathbb{E}_\tau [\epsilon^2] \right] = \mathbb{E}_* \left[\cancel{\text{Var}(\epsilon)} + \cancel{\mu_\epsilon^2} \right] = \mathbb{E}_* [\sigma^2] = \underbrace{\sigma^2}_{\text{Irreducible error}}$$

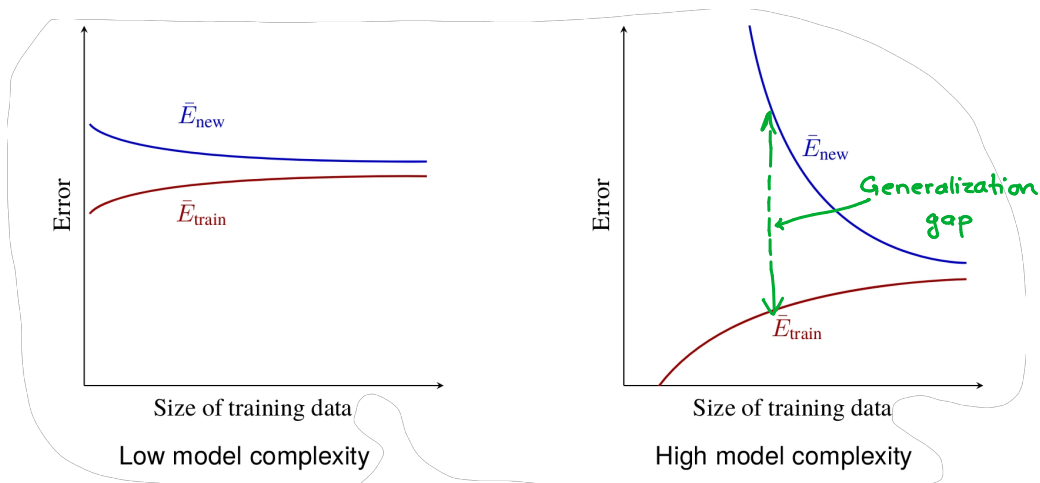
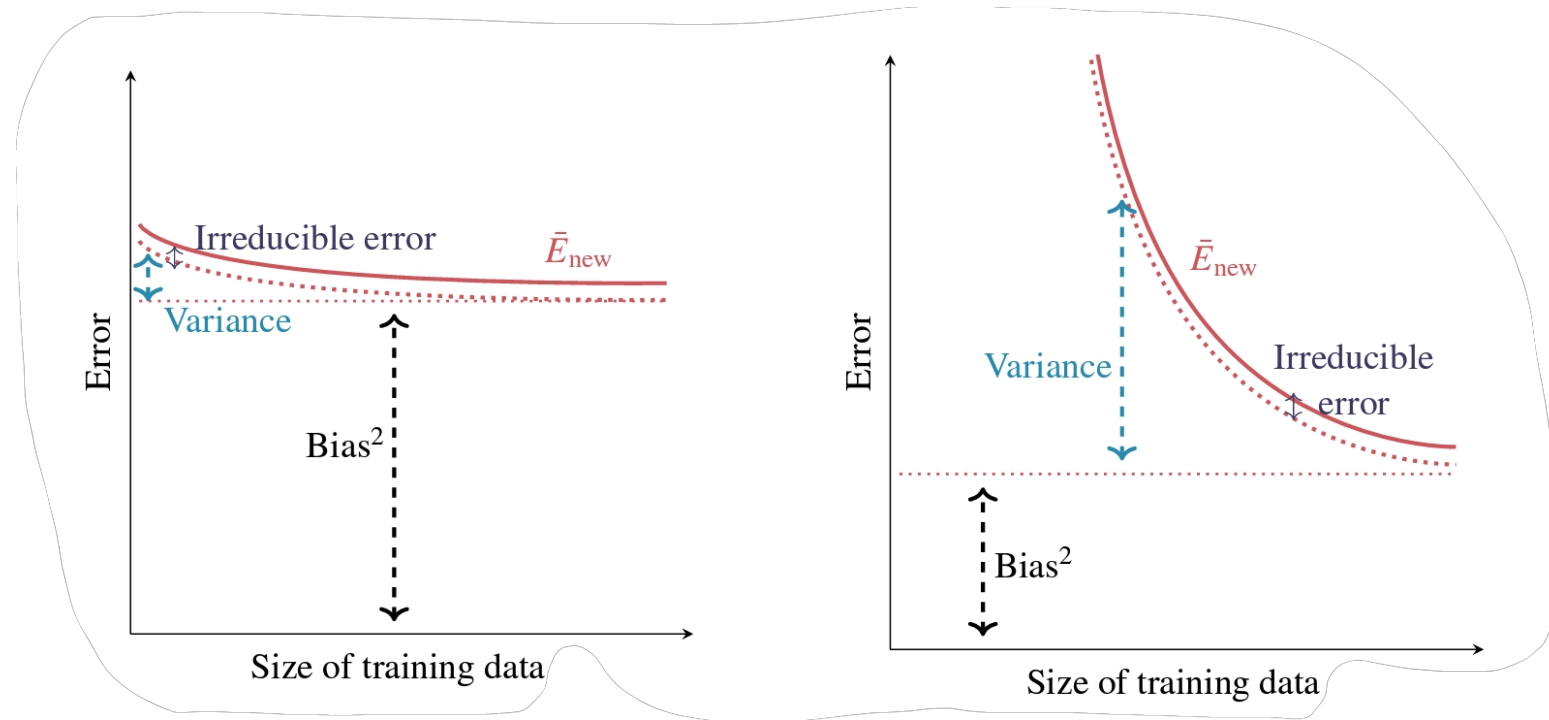
BIAS - VARIANCE TRADE-OFF

$$\bar{E}_{\text{new}} = \underbrace{\mathbb{E}_* \left[\left(\bar{f}(x_*) - f_0(x_*) \right)^2 \right]}_{\text{Bias}^2} + \underbrace{\mathbb{E}_* \left[\mathbb{E}_{\mathcal{T}} \left[\left(\hat{y}(x_*; \mathcal{T}) - \bar{f}(x_*) \right)^2 \right] \right]}_{\text{Variance}} + \underbrace{\sigma^2}_{\text{Irreducible error}}$$

- for the bias to be small, the model has to be flexible
- For the variance to be small, the model should not be very sensitive to the data points in the training set



- We also know that \bar{E}_{new} typically decreases with increasing training data



Intuitively, as the size of training data increases, we have more info about the parameters, hence the variance of prediction reduces!

Example of a simulated problem

Data
Generation

- $N = 10$ data points
- Input $x \sim \text{UniformDist}(-5, 10)$
- $y = \min(0.1x^2, 3) + \epsilon$
- $\epsilon \sim \text{NormalDist}(0, 1)$

Now fit the input-output data $\{x_i, y_i\}_{i=1}^{10}$ using

- (a) Linear regression with L_2 -regularization
- (b) Linear regression with a quadratic polynomial and L_2 -regularization
- (c) Linear regression with a cubic polynomial and L_2 -regularization
- (d) Regression Tree, (e) A random forest with 10 regression trees

--- True model
 $f_0(x)$

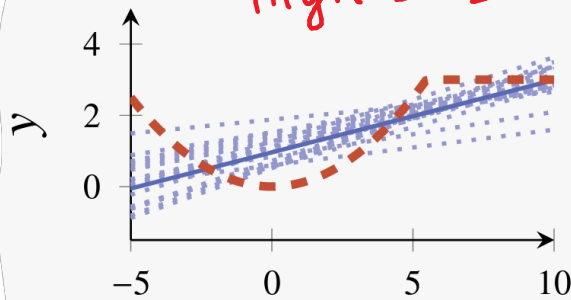
— Mean model
 $\bar{f}(x)$

--- Different
models
 $\hat{y}(x_*; T)$

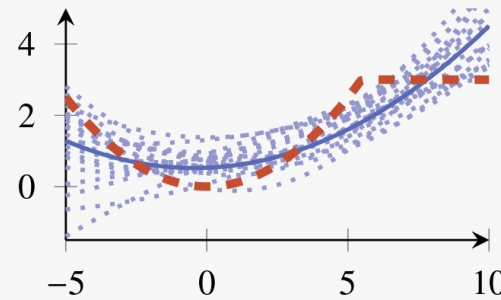
learned from
different training datasets of same size

Linear regression, $\lambda = 0.1$

← High bias

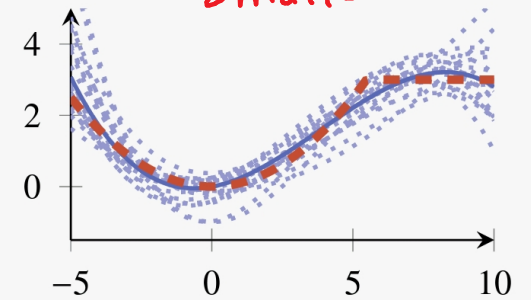


2nd order polynomial, $\lambda = 0.1$

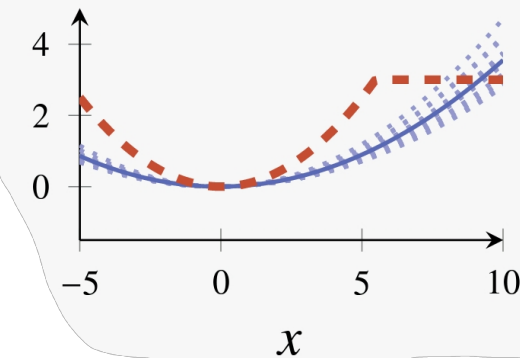


3rd order polynomial, $\lambda = 0.1$

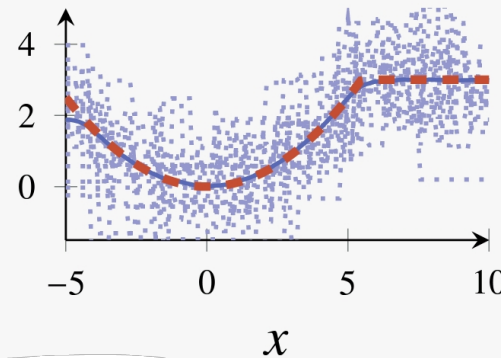
→ Smaller bias



2nd order polynomial, $\lambda = 1000$



Regression tree, max depth 5



Random forest, max depth 5

