

Bias-Variance Tradeoff



Process

n : # of data
 X_i : feature $\in \mathbb{R}^d$
 y_i : label

Collect a **data set**

$$\{(X_i, y_i)\}_{i=1}^n$$

Decide on a **model**

function f : $f(x) \approx y$, linear $f(x) = w^T x$

Find the function which fits the data best

Choose a loss function

quadratic

$$(f(x) - y)^2$$

Pick the function which minimizes loss on data

find f

minimizes

Use function to make prediction on new examples

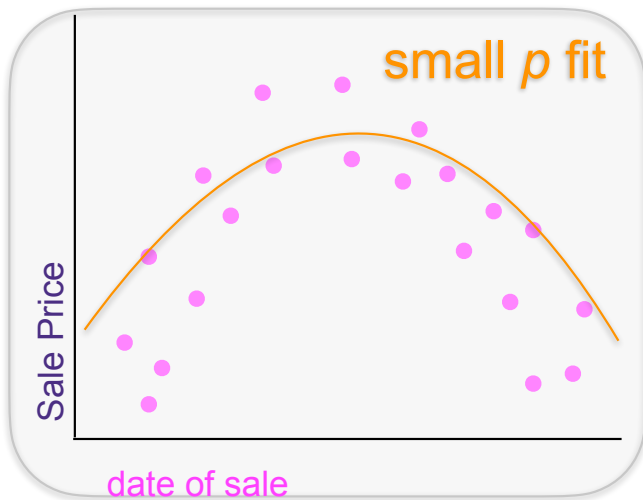
$$X_{new}, \quad f(X_{new}) \approx y_{new}$$

The regression problem

Training Data:

$$\{(x_i, y_i)\}_{i=1}^n \quad \begin{array}{l} x_i \in \mathbb{R}^d \\ y_i \in \mathbb{R} \end{array}$$

$$p=2$$



$$h_1(x) = x$$

$$h_2(x) = x^2$$

$$\vdots$$

$$h_p(x) = x^p$$

Transformed data:

$$h(x) = \begin{bmatrix} h_1(x) \\ h_2(x) \\ \vdots \\ h_p(x) \end{bmatrix}$$

Hypothesis: linear in h

$$y_i \approx h(x_i)^T w \quad w \in \mathbb{R}^p$$

Loss: least squares

$$\min_w \sum_{i=1}^n (y_i - h(x_i)^T w)^2$$

The regression problem

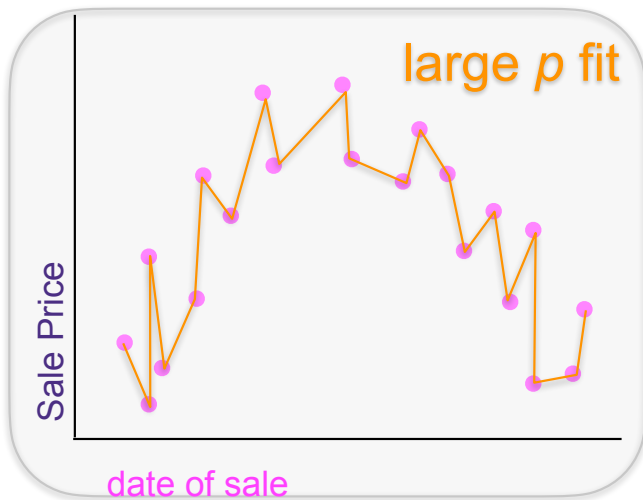
Training Data: $x_i \in \mathbb{R}^d$
 $y_i \in \mathbb{R}$
 $\{(x_i, y_i)\}_{i=1}^n$

$$p=20$$

fit:
+ training error

$$= 0$$

$$h(x_i)^T w = y_i$$



Transformed data:

$$h(x) = \begin{bmatrix} h_1(x) \\ h_2(x) \\ \vdots \\ h_p(x) \end{bmatrix}$$

Hypothesis: linear in h

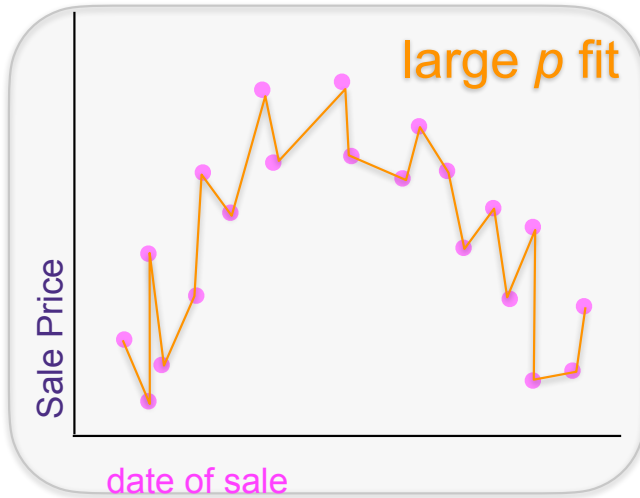
$$y_i \approx h(x_i)^T w \quad w \in \mathbb{R}^p$$

Loss: least squares

$$\min_w \sum_{i=1}^n (y_i - h(x_i)^T w)^2$$

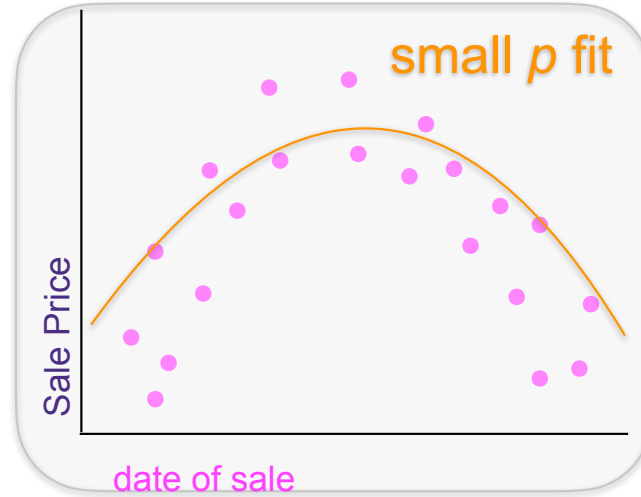
Which is better?

A: large p



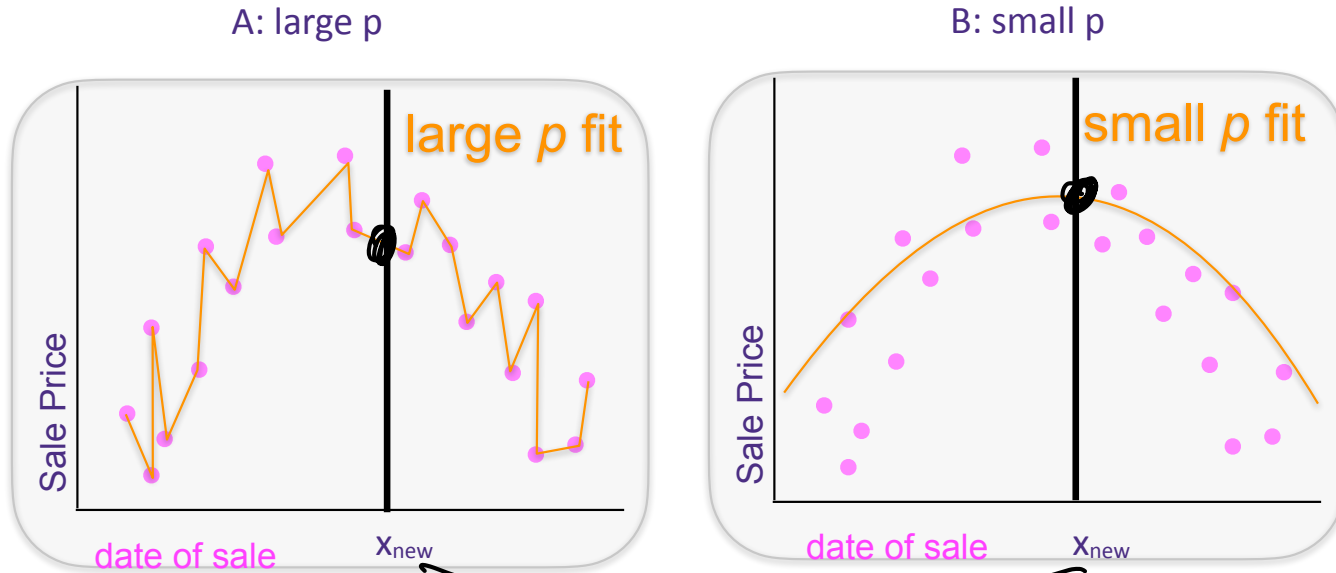
non-smooth

B: small p



smooth

Predicting sale price for a new house: A vs B



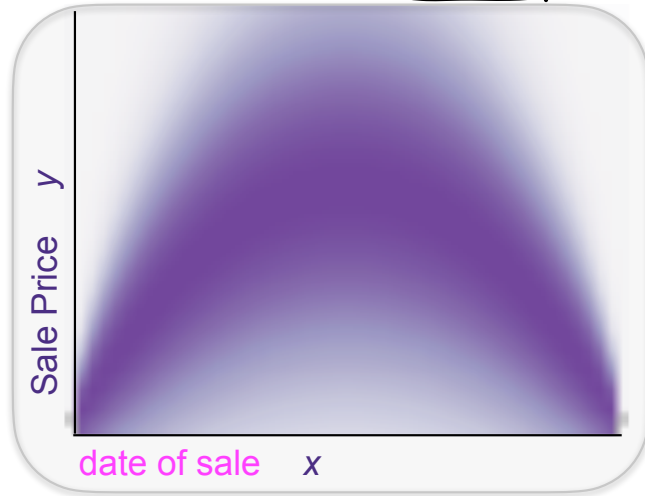
single

Our goal is to predict prices for new houses

Average Accuracy

Joint distribution

$$P_{XY}(X = x, Y = y)$$



On average over a house
drawn from this distribution,
we want to make a good prediction.

Goal: predict future sale prices

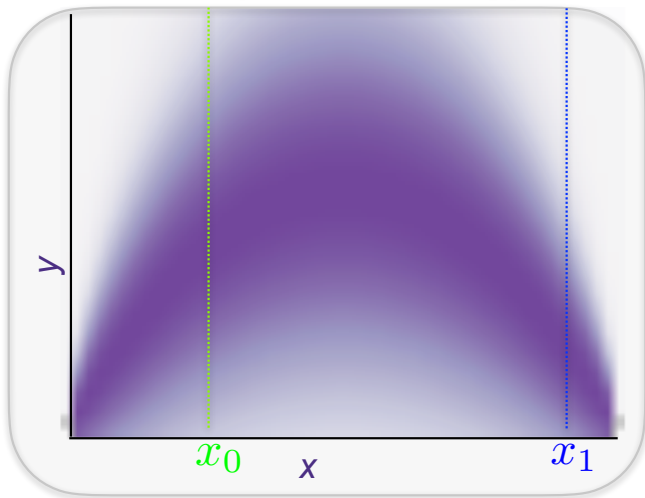
Two sources randomness

1) X is random

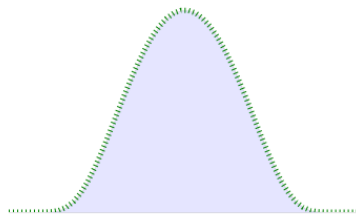
2) given X

y is random

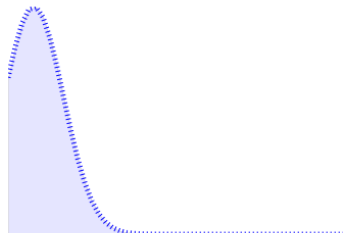
$$P_{XY}(X = x, Y = y)$$



$$P_{XY}(Y = y | X = x_0)$$



$$P_{XY}(Y = y | X = x_1)$$



Statistical Learning

$X \in \mathcal{R}^d$

\mathcal{G} : learned function

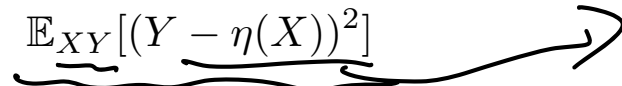
unknown

$P_{XY}(X=x, Y=y)$

Goal: Predict Y given X

Find a function η that minimizes

$$\mathbb{E}_{XY}[(Y - \eta(X))^2]$$



metric

average P_{XY}

of quadratic

h1-loss

logistic

0/1

performance
measure

Thus far, we've been using η which is a:

- Linear functions of X
- Degree p polynomials of X
- Linear "generalization" of X in p dimensions

Statistical Learning

Suppose known

$$P_{XY}(X = x, Y = y)$$

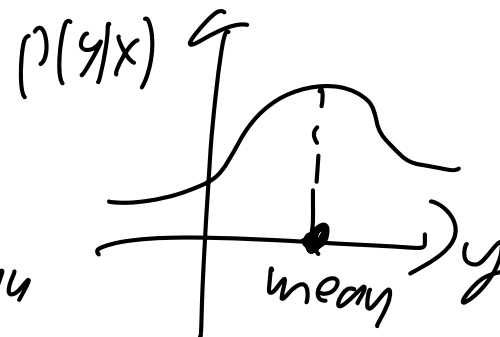
Goal: Predict Y given X

Find a function η that minimizes

$$\mathbb{E}_{XY}[(Y - \eta(X))^2] = \mathbb{E}_X \left[\mathbb{E}_{Y|X}[(Y - \eta(x))^2 | X = x] \right]$$

conditional mean

$$\eta(x) = \arg \min_c \mathbb{E}_{Y|X}[(Y - c)^2 | X = x] = \mathbb{E}_{Y|X}[Y | X = x]$$



Theorem

Under LS loss, optimal predictor: $\eta(x) = \mathbb{E}_{Y|X}[Y | X = x]$

Optimal Prediction

$$\mathbb{E}_{XY}[(Y - \eta(X))^2] = \mathbb{E}_X \left[\mathbb{E}_{Y|X}[(Y - \eta(x))^2 | X = x] \right]$$

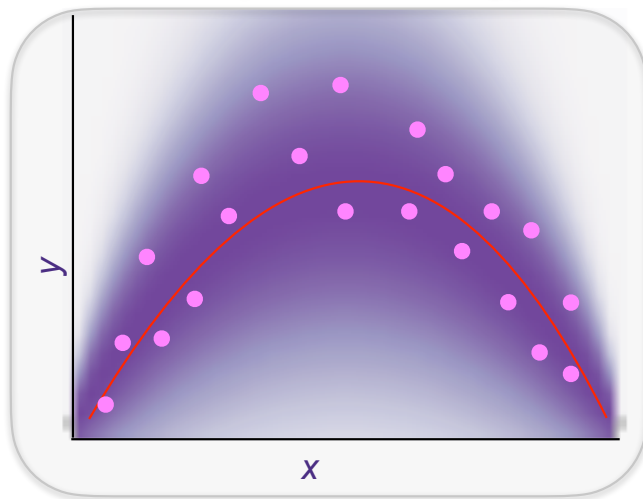
fix x
view $\eta(x)$
as a variable

Under LS loss, optimal predictor: $\eta(x) = \mathbb{E}_{Y|X}[Y | X = x]$

$$\begin{aligned} \text{Pf: } 0 &= \frac{d}{d\eta(x)} \mathbb{E}_{Y|X} [(Y - \eta(x))^2 | X = x] \\ &= \mathbb{E}_{Y|X} \left[\frac{d}{d\eta(x)} (Y - \eta(x))^2 | X = x \right] \\ &= \mathbb{E}_{Y|X} [-2(Y - \eta(x)) | X = x] \\ &= -2 \mathbb{E}_{Y|X} [Y | X = x] + 2\eta(x) \\ \Rightarrow \eta(x) &= \mathbb{E}_{Y|X} [Y | X = x] \quad \square \end{aligned}$$

Statistical Learning

$$P_{XY}(X = x, Y = y)$$



Ideally, we want to find:

$$\eta(x) = \mathbb{E}_{Y|X}[Y|X = x]$$

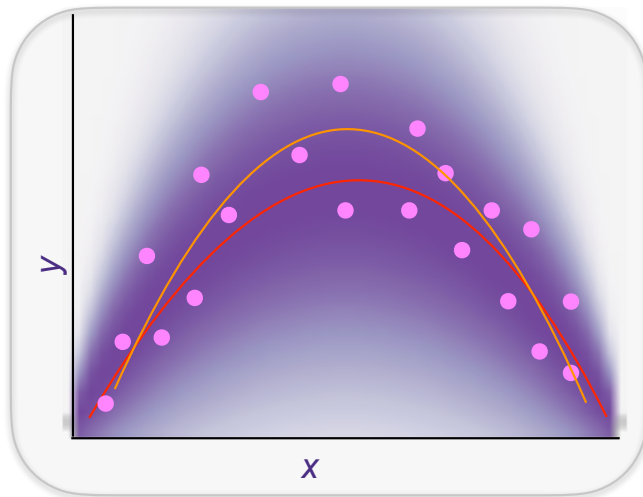
But we only have samples:

$$(x_i, y_i) \stackrel{i.i.d.}{\sim} P_{XY} \quad \text{for } i = 1, \dots, n$$

→ estimate $\eta(x)$

Statistical Learning

$$P_{XY}(X = x, Y = y)$$



We care about future predictions: $\mathbb{E}_{XY}[(Y - \hat{f}(X))^2]$

Ideally, we want to find:

$$\eta(x) = \mathbb{E}_{Y|X}[Y|X = x]$$

But we only have samples:

$$(x_i, y_i) \stackrel{i.i.d.}{\sim} P_{XY} \quad \text{for } i = 1, \dots, n$$

and are restricted to a function class (e.g., linear) so we compute:

$$\hat{f} = \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2$$

\mathcal{F} : linear

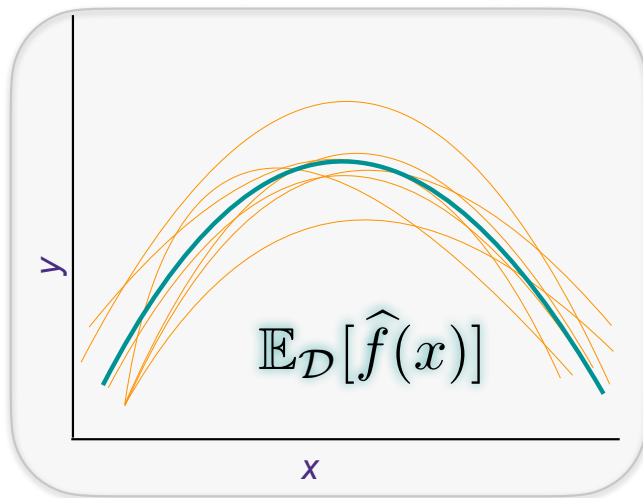
- quadratic
- poly
- neural net

Q: is \hat{f}

random
or deterministic?

Statistical Learning

$$P_{XY}(X = x, Y = y)$$



Ideally, we want to find:

$$\eta(x) = \mathbb{E}_{Y|X}[Y|X = x]$$

But we only have samples:

$$(x_i, y_i) \stackrel{i.i.d.}{\sim} P_{XY} \quad \text{for } i = 1, \dots, n$$

and are restricted to a function class (e.g., linear) so we compute:

$$\hat{f} = \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2$$

Each draw $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ results in different \hat{f}

Bias-Variance Tradeoff

$$D = \{(x_i, y_i)\}_{i=1}^M \stackrel{i.i.d.}{\sim} P_{XY}$$

for any x

$$\begin{aligned} & \mathbb{E}_{Y|X} \left[\mathbb{E}_D \left[\frac{(Y - \hat{f}_D(x))^2}{X=x} \right] \right] \\ &= \mathbb{E}_{Y|X} \left[(Y - \eta(x)) \cdot \mathbb{E}_D \left[\frac{(\eta(x) - \hat{f}_D(x))^2}{X=x} \right] \right] \end{aligned}$$

$\mathbb{E}_D(x) = \text{conditional variance}$

$$\eta(x) = \mathbb{E}_{Y|X}[Y|X=x]$$

$$\hat{f} = \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2$$

what we care

$$\mathbb{E}_{XY} \left[(Y - \hat{f}(x))^2 \right]$$

$$\Rightarrow \mathbb{E}_{XY} \left[\mathbb{E}_D \left[(Y - \hat{f}_D(x))^2 \right] \right]$$

$$= \mathbb{E}_X \left[\mathbb{E}_{Y|X} \mathbb{E}_D \left[(Y - \hat{f}_D(x))^2 \mid X=x \right] \right]$$

$$= \mathbb{E}_X \left[\mathbb{E}_{Y|X} \mathbb{E}_D \left[(Y - \eta(x) + \eta(x) - \hat{f}_D(x))^2 \mid X=x \right] \right]$$

$$= \mathbb{E}_X \left[\mathbb{E}_{Y|X} \mathbb{E}_D \left[(Y - \eta(x))^2 + 2(Y - \eta(x))(\eta(x) - \hat{f}_D(x)) + (\eta(x) - \hat{f}_D(x))^2 \mid X=x \right] \right]$$

$$= \mathbb{E}_{XY} \left[(Y - \eta(x))^2 \right] + \mathbb{E}_X \left[\mathbb{E}_{Y|X} \mathbb{E}_D \left[(\eta(x) - \hat{f}_D(x))^2 \mid X=x \right] \right]$$

Bias-Variance Tradeoff

$$\eta(x) = \mathbb{E}_{Y|X}[Y|X=x] \quad \hat{f} = \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2$$

① $\mathbb{E}_{XY} [(Y - \eta(x))^2]$: irreducible error
independent of data

② $\mathbb{E}_X \mathbb{E}_{Y|X} [\mathbb{E}_D (y(x) - \hat{f}_D(x))^2 | X=x]$ learning error
= $\mathbb{E}_X \mathbb{E}_D [(y(x) - \hat{f}_D(x))^2]$. depend on D

Bias-Variance Tradeoff

$$\eta(x) = \mathbb{E}_{Y|X}[Y|X = x]$$

$$\hat{f} = \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2$$

Bias-Variance Tradeoff

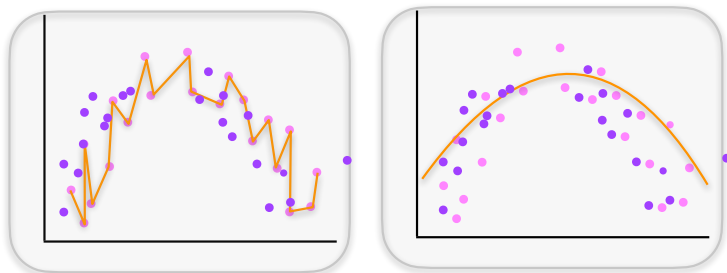
$$\mathbb{E}_{Y|X}[\mathbb{E}_{\mathcal{D}}[(Y - \hat{f}_{\mathcal{D}}(x))^2] | X = x] = \underbrace{\mathbb{E}_{Y|X}[(Y - \eta(x))^2 | X = x]}$$

irreducible error

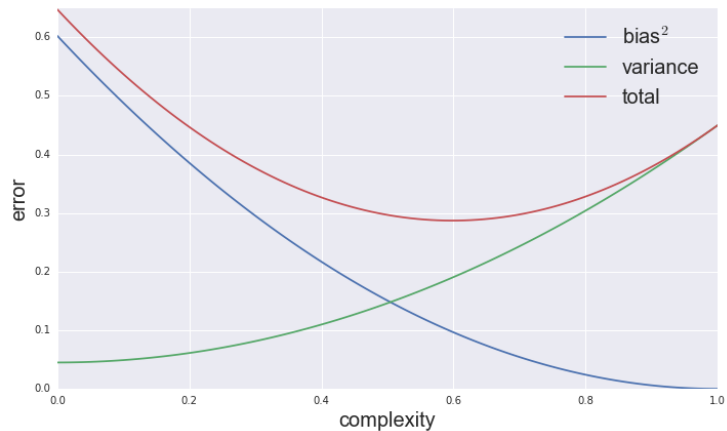
$$+ \underbrace{(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)])^2}_{\text{bias squared}} + \underbrace{\mathbb{E}_{\mathcal{D}}[(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x))^2]}_{\text{variance}}$$

bias squared

variance



If we re-drew our data, what the LS training error estimator look like for generalized linear functions in small p/large p dimensions?



Bias-Variance Tradeoff

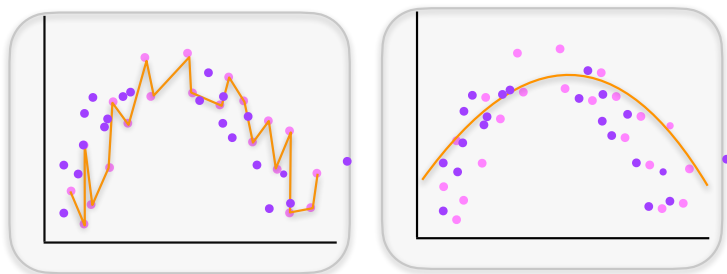
$$\mathbb{E}_{Y|X}[\mathbb{E}_{\mathcal{D}}[(Y - \hat{f}_{\mathcal{D}}(x))^2] | X = x] = \underbrace{\mathbb{E}_{Y|X}[(Y - \eta(x))^2 | X = x]}$$

irreducible error

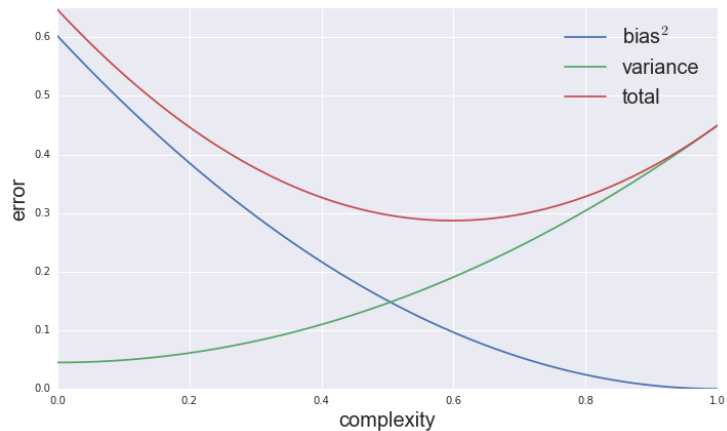
$$+ \underbrace{(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)])^2}_{\text{bias squared}} + \underbrace{\mathbb{E}_{\mathcal{D}}[(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x))^2]}_{\text{variance}}$$

bias squared

variance



If we re-drew our data, what the LS training error estimator look like for generalized linear functions in small p/large p dimensions?



Example: Linear LS

$$\mathbf{Y} = \mathbf{X}w + \epsilon$$

$$\text{if } y_i = x_i^T w + \epsilon_i \quad \text{and} \quad \epsilon_i \stackrel{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$$

Example: Linear LS: compute bias

$$\mathbf{Y} = \mathbf{X}w + \epsilon$$

$$\text{if } y_i = x_i^T w + \epsilon_i \quad \text{and} \quad \epsilon_i \stackrel{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$$

$$\hat{w}_{MLE} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} = w + (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \epsilon$$

$$\eta(x) = \mathbb{E}_{Y|X}[Y|X = x]$$

$$\hat{f}_{\mathcal{D}}(x) = \hat{w}^T x = w^T x + \epsilon^T \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1} x$$

$$\underbrace{(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)])^2}_{\text{bias squared}}.$$

bias squared

Example: Linear LS: compute variance

$$\mathbf{Y} = \mathbf{X}w + \epsilon$$

$$\text{if } y_i = x_i^T w + \epsilon_i \quad \text{and} \quad \epsilon_i \stackrel{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$$

$$\hat{w}_{MLE} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} = w + (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \epsilon$$

$$\hat{f}_{\mathcal{D}}(x) = \hat{w}^T x = w^T x + \epsilon^T \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1} x$$

$$\mathbb{E}_{\mathcal{D}}[(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x))^2] =$$

variance

Example: Linear LS: compute variance

$$\mathbf{Y} = \mathbf{X}w + \epsilon$$

$$\text{if } y_i = x_i^T w + \epsilon_i \quad \text{and} \quad \epsilon_i \stackrel{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$$

$$\hat{w}_{MLE} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} = w + (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \epsilon$$

$$\hat{f}_{\mathcal{D}}(x) = \hat{w}^T x = w^T x + \epsilon^T \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1} x$$

$$\mathbb{E}_{\mathcal{D}}[(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x))^2] =$$

variance

Example: Linear LS

$$\mathbf{Y} = \mathbf{X}w + \epsilon$$

$$\text{if } y_i = x_i^T w + \epsilon_i \quad \text{and} \quad \epsilon_i \stackrel{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$$

$$\hat{w}_{MLE} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} = w + (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \epsilon$$

$$\eta(x) = \mathbb{E}_{Y|X}[Y|X = x]$$

$$\hat{f}_{\mathcal{D}}(x) = \hat{w}^T x = w^T x + \epsilon^T \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1} x$$

$$\underbrace{\mathbb{E}_{XY}[(Y - \eta(x))^2 | X = x]}_{\text{irreducible error}} = \sigma^2 \quad \underbrace{(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)])^2}_{\text{bias squared}} = 0$$

$$\mathbb{E}_{X=x} \left[\underbrace{\mathbb{E}_{\mathcal{D}}[(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x))^2]}_{\text{variance}} \right] = \frac{p\sigma^2}{n}$$

Overfitting



Bias-Variance Tradeoff

- > Choice of hypothesis class introduces learning bias
 - More complex class \rightarrow less bias
 - More complex class \rightarrow more variance
- > But in practice??

Bias-Variance Tradeoff

- > Choice of hypothesis class introduces learning bias
 - More complex class → less bias
 - More complex class → more variance
- > But in practice??
- > Before we saw how increasing the feature space can increase the complexity of the learned estimator:

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \mathcal{F}_3 \subset \dots$$

$$\hat{f}_{\mathcal{D}}^{(k)} = \arg \min_{f \in \mathcal{F}_k} \frac{1}{|\mathcal{D}|} \sum_{(x_i, y_i) \in \mathcal{D}} (y_i - f(x_i))^2$$

Complexity grows as k grows

Training set error as a function of model complexity

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \mathcal{F}_3 \subset \dots$$

$$\hat{f}_{\mathcal{D}}^{(k)} = \arg \min_{f \in \mathcal{F}_k} \frac{1}{|\mathcal{D}|} \sum_{(x_i, y_i) \in \mathcal{D}} (y_i - f(x_i))^2$$



Complexity (k)

TRAIN error:

$$\mathcal{D} \stackrel{i.i.d.}{\sim} P_{XY}$$

$$\frac{1}{|\mathcal{D}|} \sum_{(x_i, y_i) \in \mathcal{D}} (y_i - \hat{f}_{\mathcal{D}}^{(k)}(x_i))^2$$

TRUE error:

$$\mathbb{E}_{XY}[(Y - \hat{f}_{\mathcal{D}}^{(k)}(X))^2]$$

Training set error as a function of model complexity

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \mathcal{F}_3 \subset \dots$$

$$\hat{f}_{\mathcal{D}}^{(k)} = \arg \min_{f \in \mathcal{F}_k} \frac{1}{|\mathcal{D}|} \sum_{(x_i, y_i) \in \mathcal{D}} (y_i - f(x_i))^2$$

TRAIN error:

$$\mathcal{D} \stackrel{i.i.d.}{\sim} P_{XY}$$

$$\frac{1}{|\mathcal{D}|} \sum_{(x_i, y_i) \in \mathcal{D}} (y_i - \hat{f}_{\mathcal{D}}^{(k)}(x_i))^2$$

TRUE error:

$$\mathbb{E}_{XY}[(Y - \hat{f}_{\mathcal{D}}^{(k)}(X))^2]$$

TEST error:

$$\mathcal{T} \stackrel{i.i.d.}{\sim} P_{XY}$$

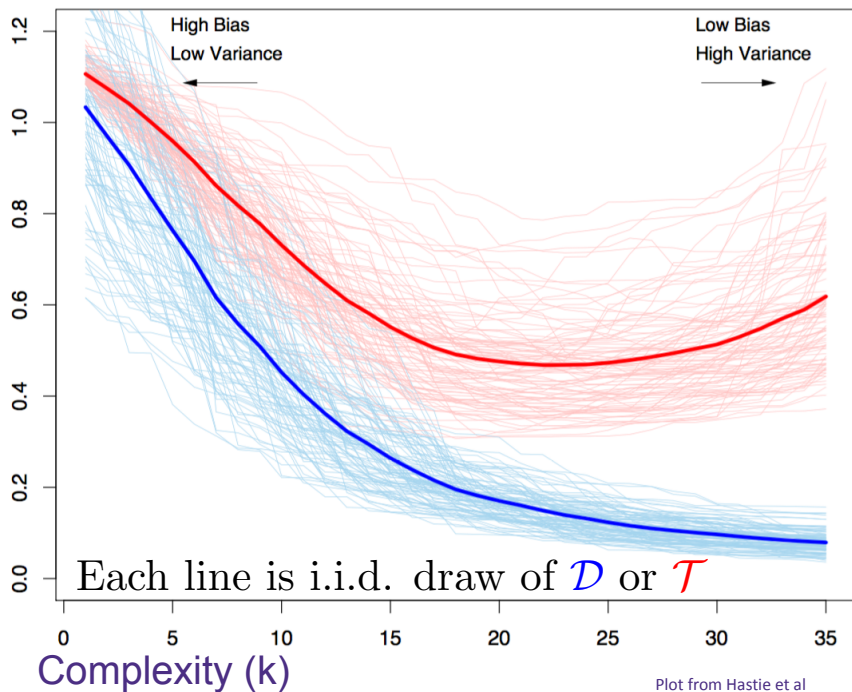
$$\frac{1}{|\mathcal{T}|} \sum_{(x_i, y_i) \in \mathcal{T}} (y_i - \hat{f}_{\mathcal{D}}^{(k)}(x_i))^2$$

Important: $\mathcal{D} \cap \mathcal{T} = \emptyset$

Training set error as a function of model complexity

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \mathcal{F}_3 \subset \dots$$

$$\hat{f}_{\mathcal{D}}^{(k)} = \arg \min_{f \in \mathcal{F}_k} \frac{1}{|\mathcal{D}|} \sum_{(x_i, y_i) \in \mathcal{D}} (y_i - f(x_i))^2$$



TRAIN error:

$$\mathcal{D} \stackrel{i.i.d.}{\sim} P_{XY}$$

$$\frac{1}{|\mathcal{D}|} \sum_{(x_i, y_i) \in \mathcal{D}} (y_i - \hat{f}_{\mathcal{D}}^{(k)}(x_i))^2$$

TRUE error:

$$\mathbb{E}_{XY}[(Y - \hat{f}_{\mathcal{D}}^{(k)}(X))^2]$$

TEST error:

$$\mathcal{T} \stackrel{i.i.d.}{\sim} P_{XY}$$

$$\frac{1}{|\mathcal{T}|} \sum_{(x_i, y_i) \in \mathcal{T}} (y_i - \hat{f}_{\mathcal{D}}^{(k)}(x_i))^2$$

Important: $\mathcal{D} \cap \mathcal{T} = \emptyset$

Training set error as a function of model complexity

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \mathcal{F}_3 \subset \dots$$

$$\hat{f}_{\mathcal{D}}^{(k)} = \arg \min_{f \in \mathcal{F}_k} \frac{1}{|\mathcal{D}|} \sum_{(x_i, y_i) \in \mathcal{D}} (y_i - f(x_i))^2$$

TRAIN error is **optimistically biased** because it is evaluated on the data it trained on. **TEST error** is **unbiased** only if \mathcal{T} is never used to train the model or even pick the complexity k .

TRAIN error:

$$\mathcal{D} \stackrel{i.i.d.}{\sim} P_{XY}$$

$$\frac{1}{|\mathcal{D}|} \sum_{(x_i, y_i) \in \mathcal{D}} (y_i - \hat{f}_{\mathcal{D}}^{(k)}(x_i))^2$$

TRUE error:

$$\mathbb{E}_{XY}[(Y - \hat{f}_{\mathcal{D}}^{(k)}(X))^2]$$

TEST error:

$$\mathcal{T} \stackrel{i.i.d.}{\sim} P_{XY}$$

$$\frac{1}{|\mathcal{T}|} \sum_{(x_i, y_i) \in \mathcal{T}} (y_i - \hat{f}_{\mathcal{D}}^{(k)}(x_i))^2$$

Important: $\mathcal{D} \cap \mathcal{T} = \emptyset$

How many points do I use for training/testing?

- > **Very hard question to answer!**
 - Too few training points, learned model is bad
 - Too few test points, you never know if you reached a good solution
- > **More on this later the quarter, but still hard to answer**
- > **Typically:**
 - If you have a reasonable amount of data 90/10 splits are common
 - If you have little data, then you need to get fancy (e.g., bootstrapping)