Linear Regression

Machine Learning – CSE546

Kevin Jamieson
University of Washington

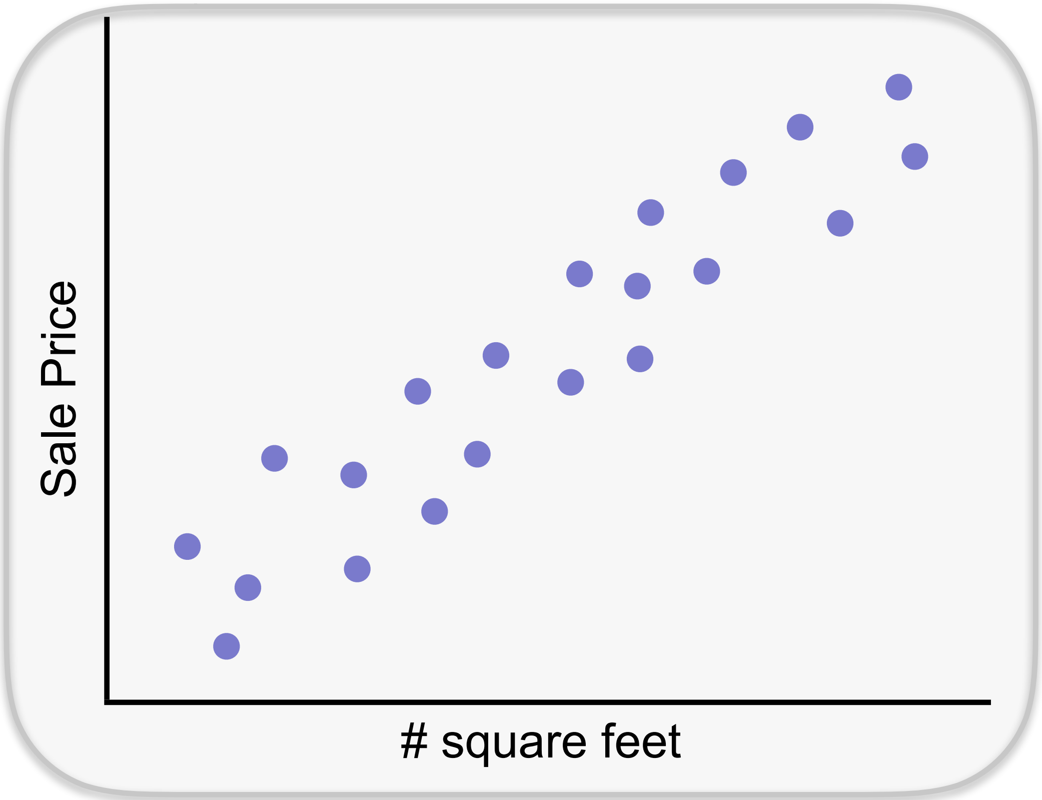
Oct 5, 2017

The regression problem

Given past sales data on zillow.com, predict:

y = **House sale price** *from*

x = **{# sq. ft., zip code, date of sale, etc.}**



Training Data:

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

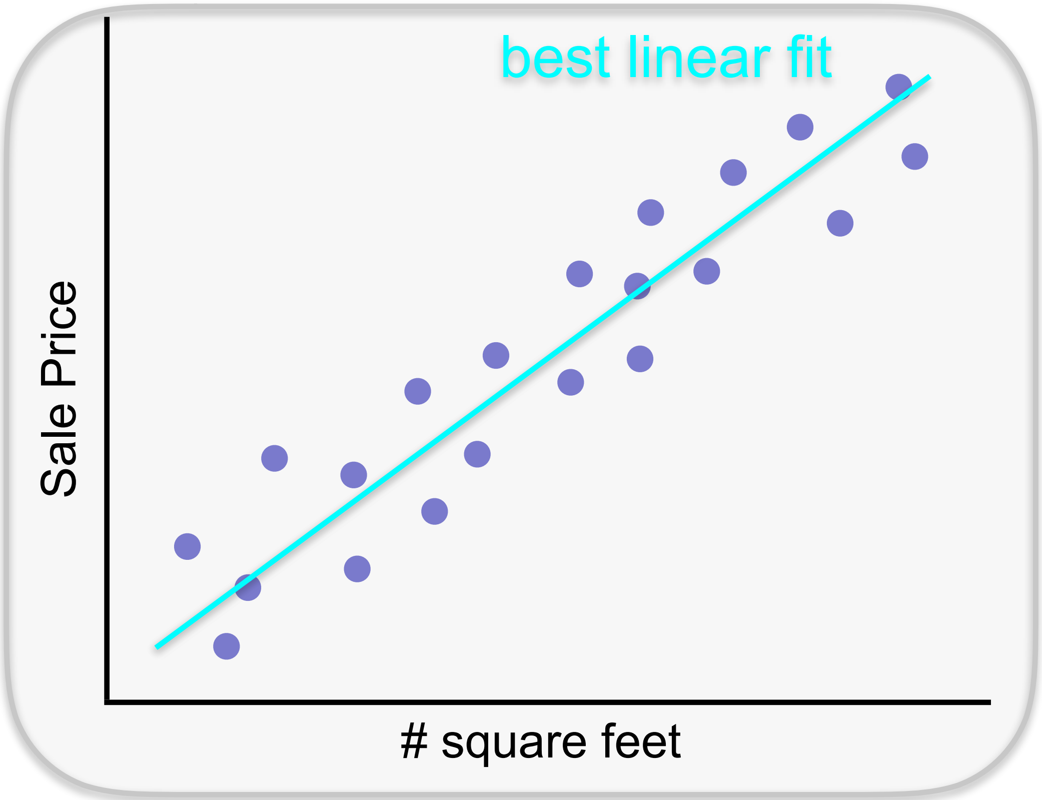
$$x_i \in \mathbb{R}^d$$
$$y_i \in \mathbb{R}$$

The regression problem

Given past sales data on zillow.com, predict:

y = **House sale price** *from*

x = **{# sq. ft., zip code, date of sale, etc.}**



Training Data:

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

$$x_i \in \mathbb{R}^d$$
$$y_i \in \mathbb{R}$$

Hypothesis: linear

$$y_i \approx x_i^T w$$

Loss: least squares

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

The regression problem in matrix notation

$$\widehat{w}_{LS} = \arg\min_w \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2$$
$$= \arg\min_w (\mathbf{y} - \mathbf{X}w)^T(\mathbf{y} - \mathbf{X}w)$$

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix}$$

The regression problem in matrix notation

$$\widehat{w}_{LS} = \arg\min_w ||\mathbf{y} - \mathbf{X}w||_2^2$$
$$= \arg\min_w (\mathbf{y} - \mathbf{X}w)^T(\mathbf{y} - \mathbf{X}w)$$

The regression problem in matrix notation

$$\widehat{w}_{LS} = \arg\min_w ||\mathbf{y} - \mathbf{X}w||_2^2$$
$$= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

What about an offset?

$$\widehat{w}_{LS}, \widehat{b}_{LS} = \arg\min_{w,b} \sum_{i=1}^n \big(y_i - (x_i^T w + b)\big)^2$$
$$= \arg\min_{w,b} ||\mathbf{y} - (\mathbf{X}w + \mathbf{1}b)||_2^2$$

Dealing with an offset

$$\widehat{w}_{LS}, \widehat{b}_{LS} = \arg\min_{w,b} ||\mathbf{y} - (\mathbf{X}w + \mathbf{1}b)||_2^2$$

Dealing with an offset

$$\widehat{w}_{LS}, \widehat{b}_{LS} = \arg\min_{w,b} ||\mathbf{y} - (\mathbf{X}w + \mathbf{1}b)||_2^2$$

$$\mathbf{X}^T\mathbf{X}\widehat{w}_{LS} + \widehat{b}_{LS}\mathbf{X}^T\mathbf{1} = \mathbf{X}^T\mathbf{y}$$

$$\mathbf{1}^T\mathbf{X}\widehat{w}_{LS} + \widehat{b}_{LS}\mathbf{1}^T\mathbf{1} = \mathbf{1}^T\mathbf{y}$$

If $\mathbf{X}^T\mathbf{1} = 0$ (i.e., if each feature is mean-zero) then

$$\widehat{w}_{LS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

$$\widehat{b}_{LS} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

The regression problem in matrix notation

$$\widehat{w}_{LS} = \arg\min_w ||\mathbf{y} - \mathbf{X}w||_2^2$$
$$= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

But why least squares?

Consider $\quad y_i = x_i^T w + \epsilon_i \quad$ where $\quad \epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$

$$P(y|x, w, \sigma) =$$

Maximizing log-likelihood

Maximize:

$$\log P(\mathcal{D}|w,\sigma) = \log(\frac{1}{\sqrt{2\pi}\sigma})^n \prod_{i=1}^{n} e^{-\frac{(y_i - x_i^T w)^2}{2\sigma^2}}$$

MLE is LS under linear model

$$\widehat{w}_{LS} = \arg\min_w \sum_{i=1}^n \left(y_i - x_i^T w\right)^2$$

$$\widehat{w}_{MLE} = \arg\max_w P(\mathcal{D}|w,\sigma)$$

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

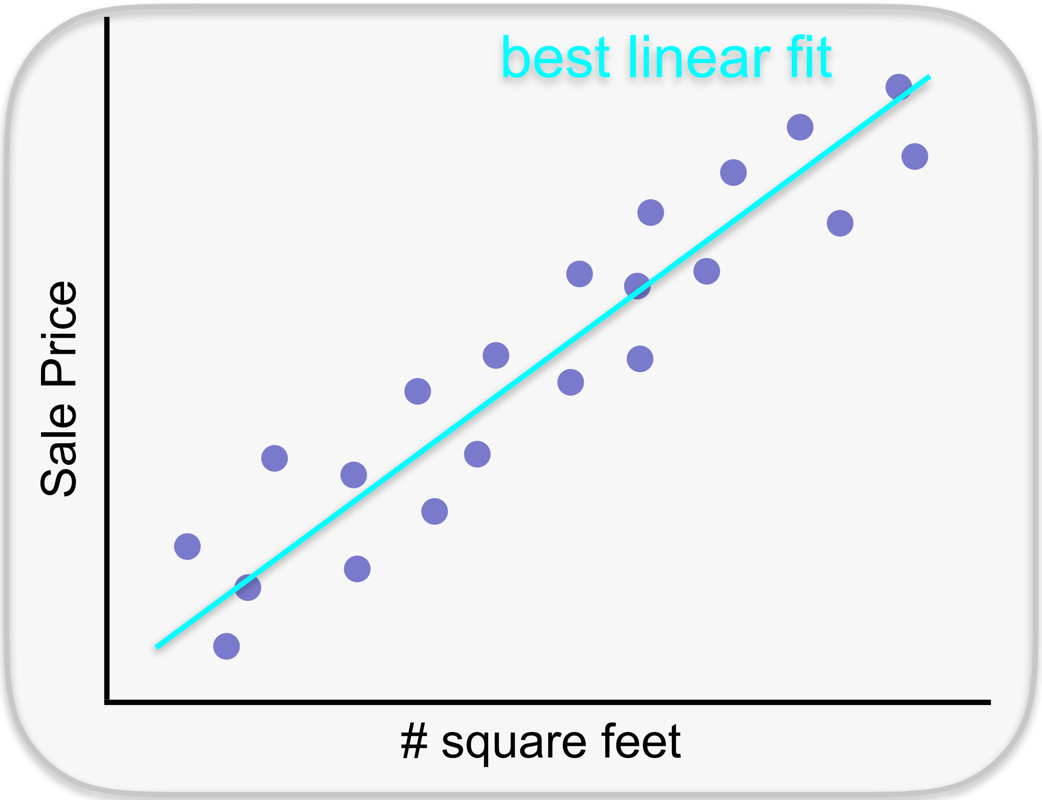
$$\boxed{\widehat{w}_{LS} = \widehat{w}_{MLE} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}}$$

The regression problem

Given past sales data on zillow.com, predict:

y = **House sale price** *from*

x = **{# sq. ft., zip code, date of sale, etc.}**



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

$x_i \in \mathbb{R}^d$

$y_i \in \mathbb{R}$

Hypothesis: linear

$$y_i \approx x_i^T w$$

Loss: least squares

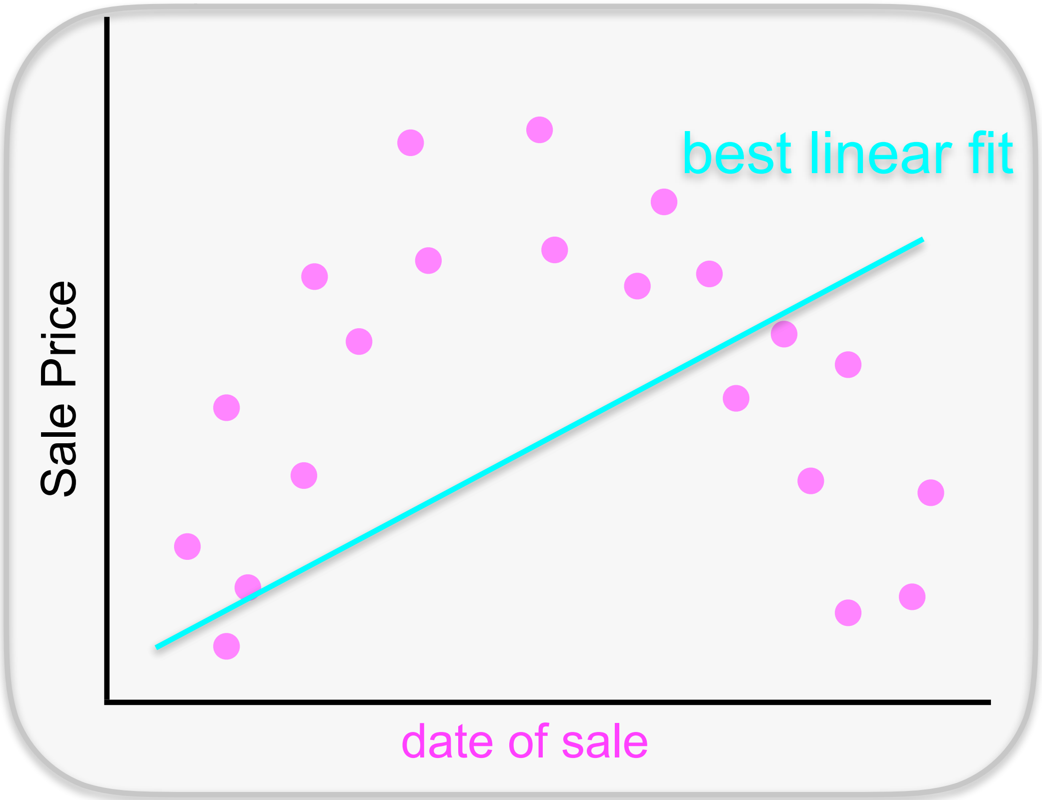
$$\min_w \sum_{i=1}^n \left(y_i - x_i^T w\right)^2$$

The regression problem

Given past sales data on zillow.com, predict:

y = **House sale price** *from*

x = **{# sq. ft., zip code, date of sale, etc.}**



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

$$x_i \in \mathbb{R}^d$$
$$y_i \in \mathbb{R}$$

Hypothesis: linear

$$y_i \approx x_i^T w$$

Loss: least squares

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

The regression problem

Training Data:

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

$$x_i \in \mathbb{R}^d$$
$$y_i \in \mathbb{R}$$

Transformed data:

Hypothesis: linear

$$y_i \approx x_i^T w$$

Loss: least squares

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

The regression problem

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

$$x_i \in \mathbb{R}^d$$
$$y_i \in \mathbb{R}$$

Hypothesis: linear

$$y_i \approx x_i^T w$$

Loss: least squares

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

Transformed data:

$h : \mathbb{R}^d \to \mathbb{R}^p$ maps original features to a rich, possibly high-dimensional space

in d=1: $h(x) = \begin{bmatrix} h_1(x) \\ h_2(x) \\ \vdots \\ h_p(x) \end{bmatrix} = \begin{bmatrix} x \\ x^2 \\ \vdots \\ x^p \end{bmatrix}$

for d>1, generate $\{u_j\}_{j=1}^p \subset \mathbb{R}^d$

$$h_j(x) = \frac{1}{1 + \exp(u_j^T x)}$$

$$h_j(x) = (u_j^T x)^2$$

$$h_j(x) = \cos(u_j^T x)$$

The regression problem

Training Data:

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

~~Hypothesis: linear~~

$$\cancel{y_i \approx x_i^T w}$$

~~Loss: least squares~~

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

Transformed data:

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

Hypothesis: linear

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

Loss: least squares

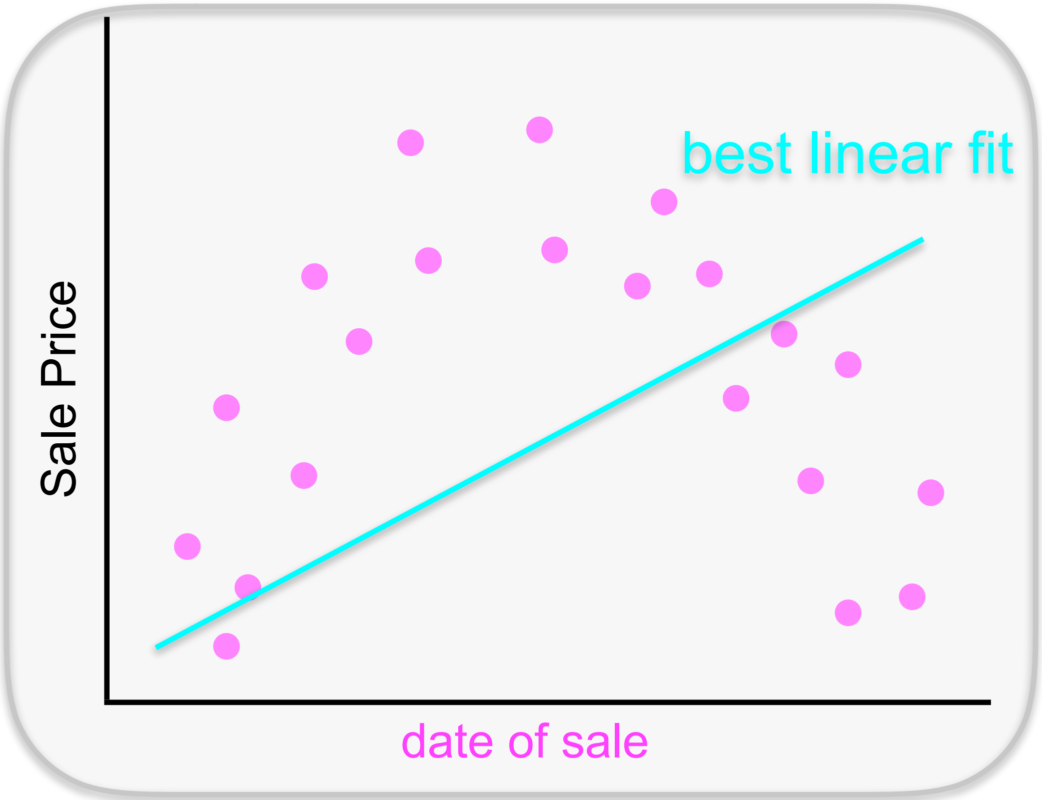
$$\min_w \sum_{i=1}^n \left(y_i - h(x_i)^T w\right)^2$$

The regression problem

Training Data:

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

$$x_i \in \mathbb{R}^d$$
$$y_i \in \mathbb{R}$$



Transformed data:

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

Hypothesis: linear

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

Loss: least squares

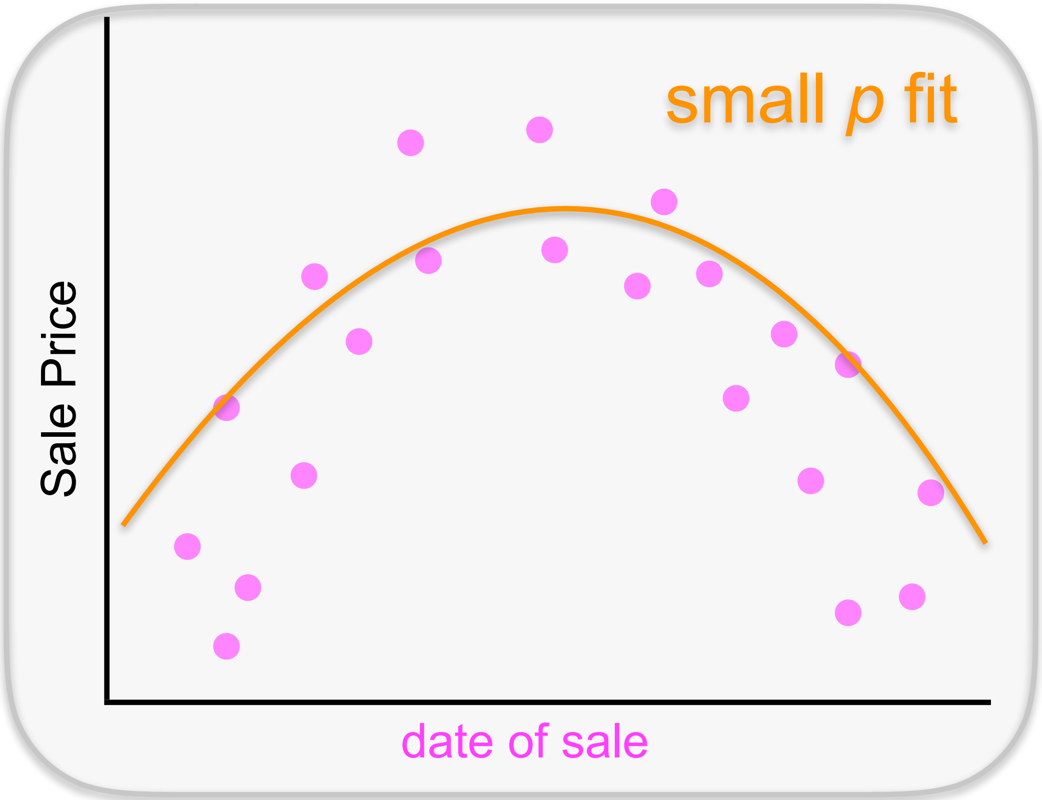
$$\min_w \sum_{i=1}^n \left(y_i - h(x_i)^T w\right)^2$$

The regression problem

Training Data:

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

$$x_i \in \mathbb{R}^d$$
$$y_i \in \mathbb{R}$$



Transformed data:

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

Hypothesis: linear

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

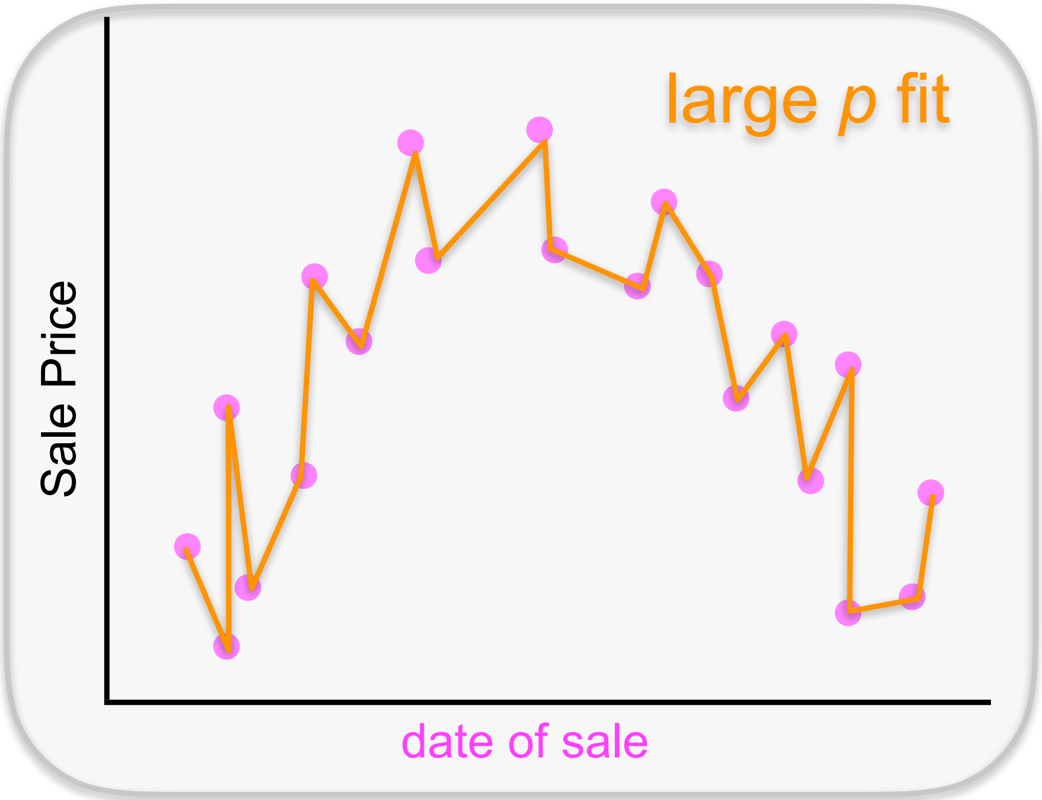
Loss: least squares

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

The regression problem

Training Data:

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



Transformed data:

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

Hypothesis: linear

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

Loss: least squares

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

What's going on here?

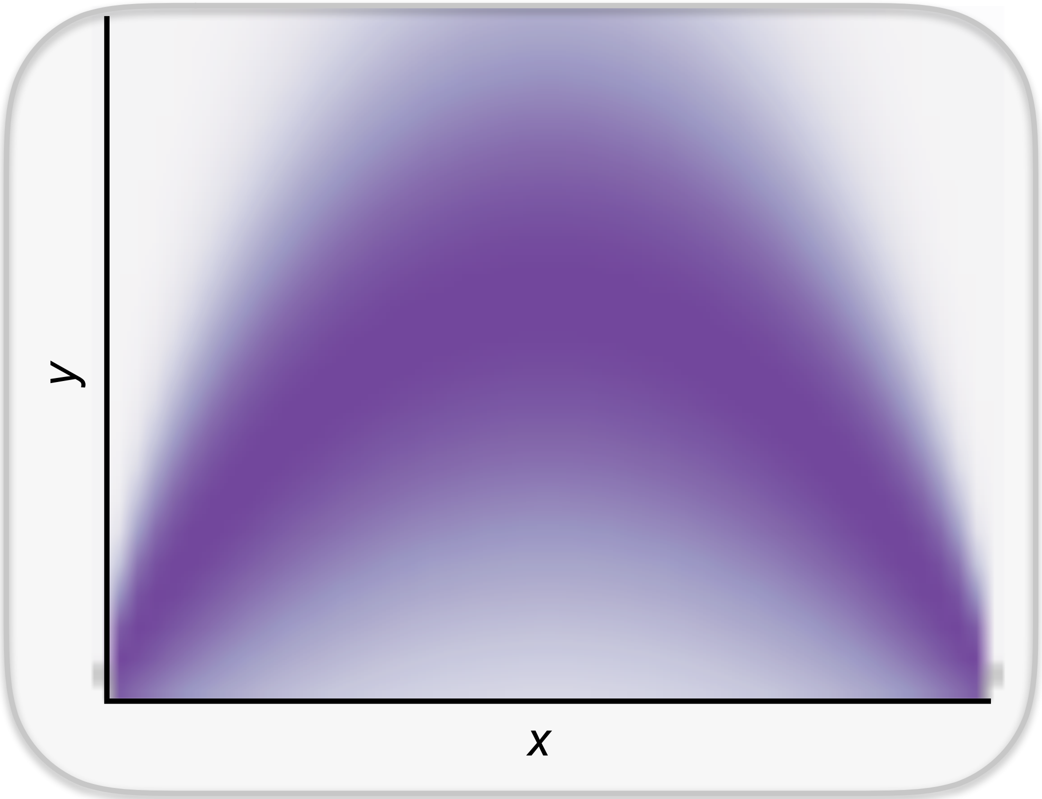
Bias-Variance Tradeoff

Machine Learning – CSE546
Kevin Jamieson
University of Washington

Oct 5, 2017

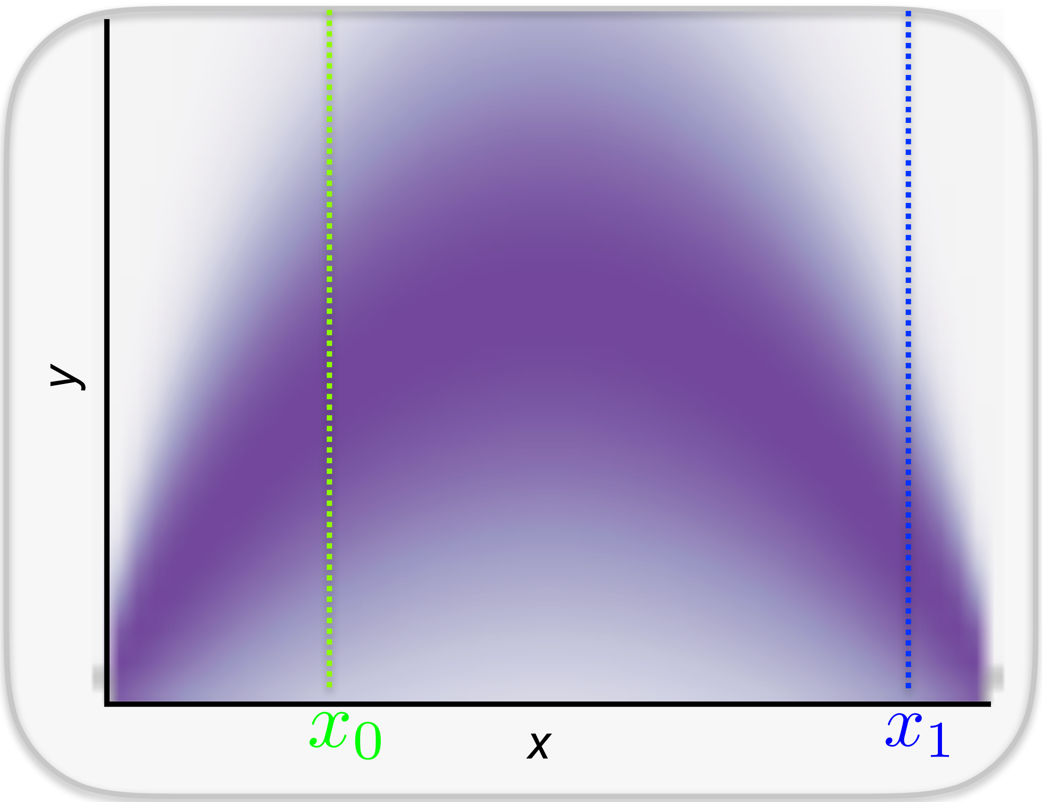
Statistical Learning

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

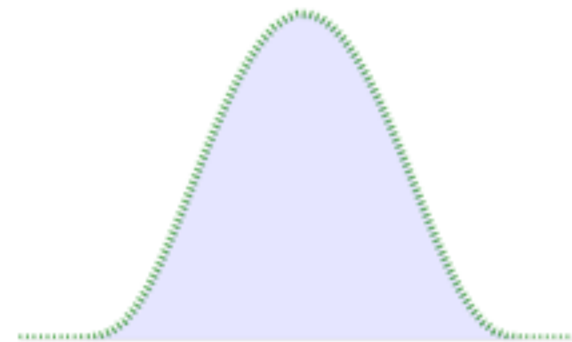


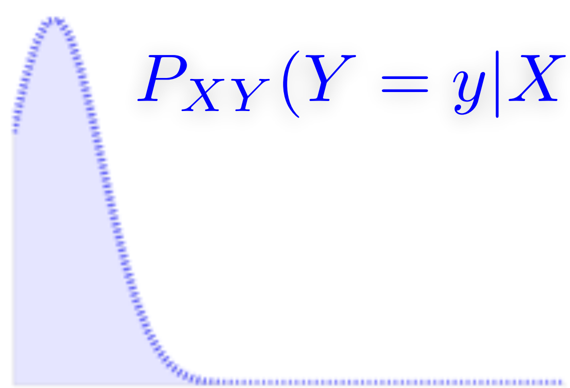
Statistical Learning

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



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

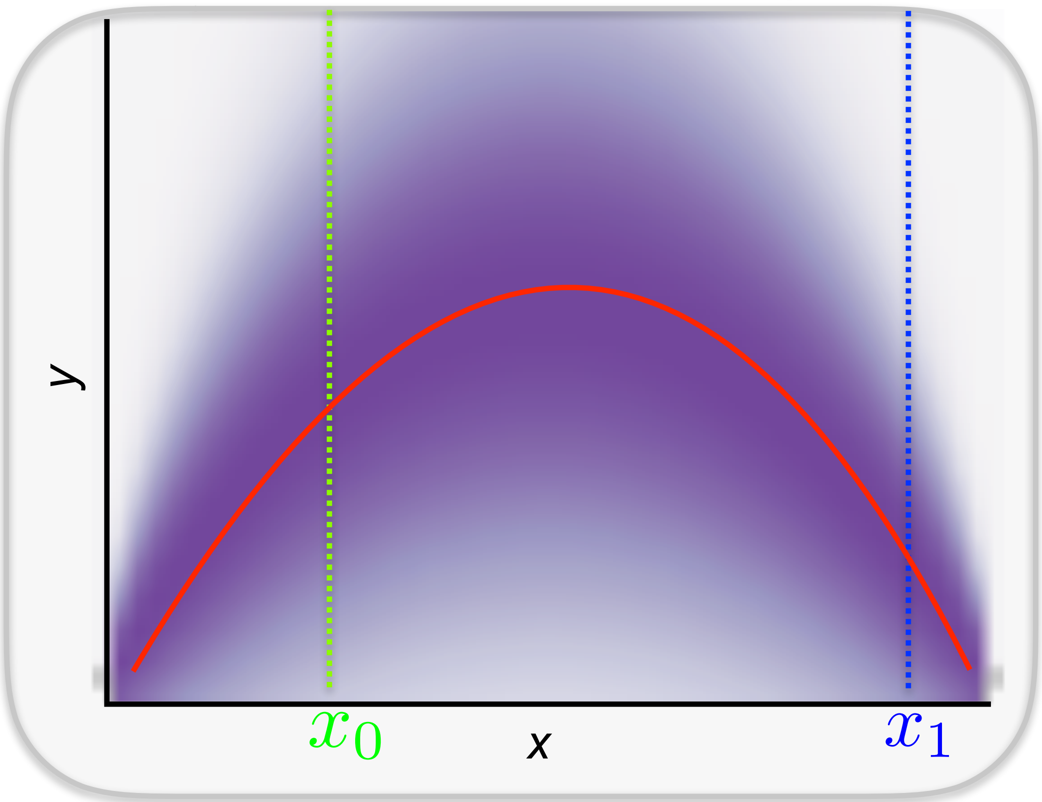




$P_{XY}(Y = y | X = x_1)$

Statistical Learning

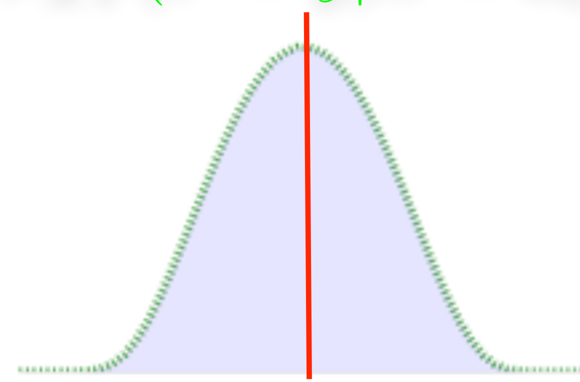
$$P_{XY}(X = x, Y = y)$$

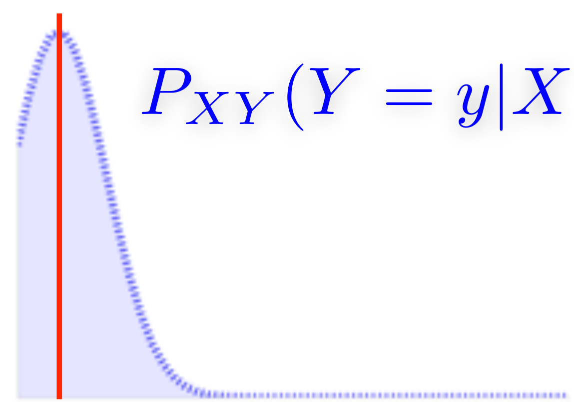


Ideally, we want to find:

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

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

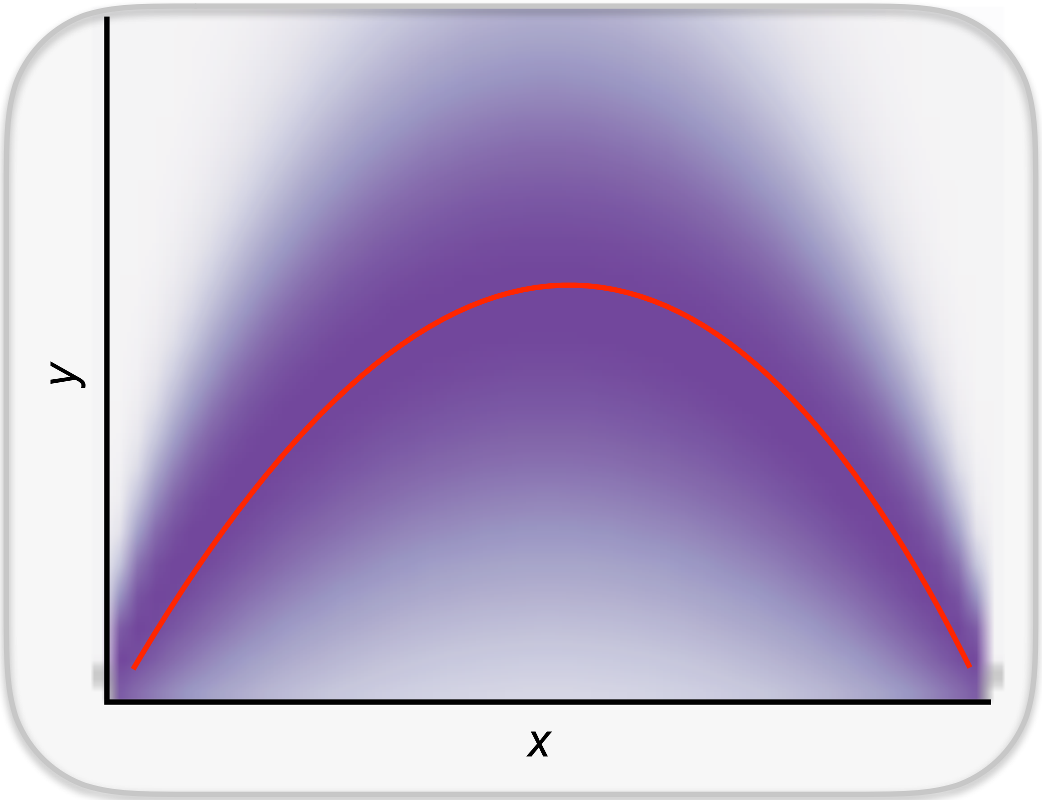




$$P_{XY}(Y = y | X = x_1)$$

Statistical Learning

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

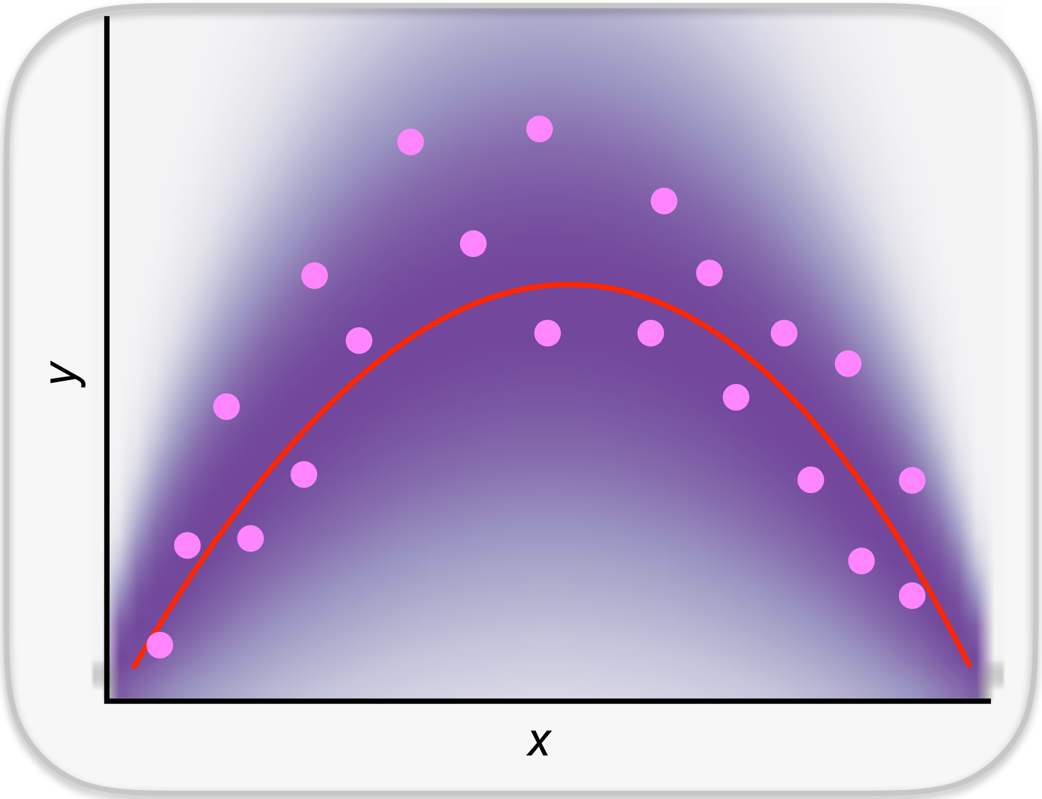


Ideally, we want to find:

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

Statistical Learning

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



Ideally, we want to find:

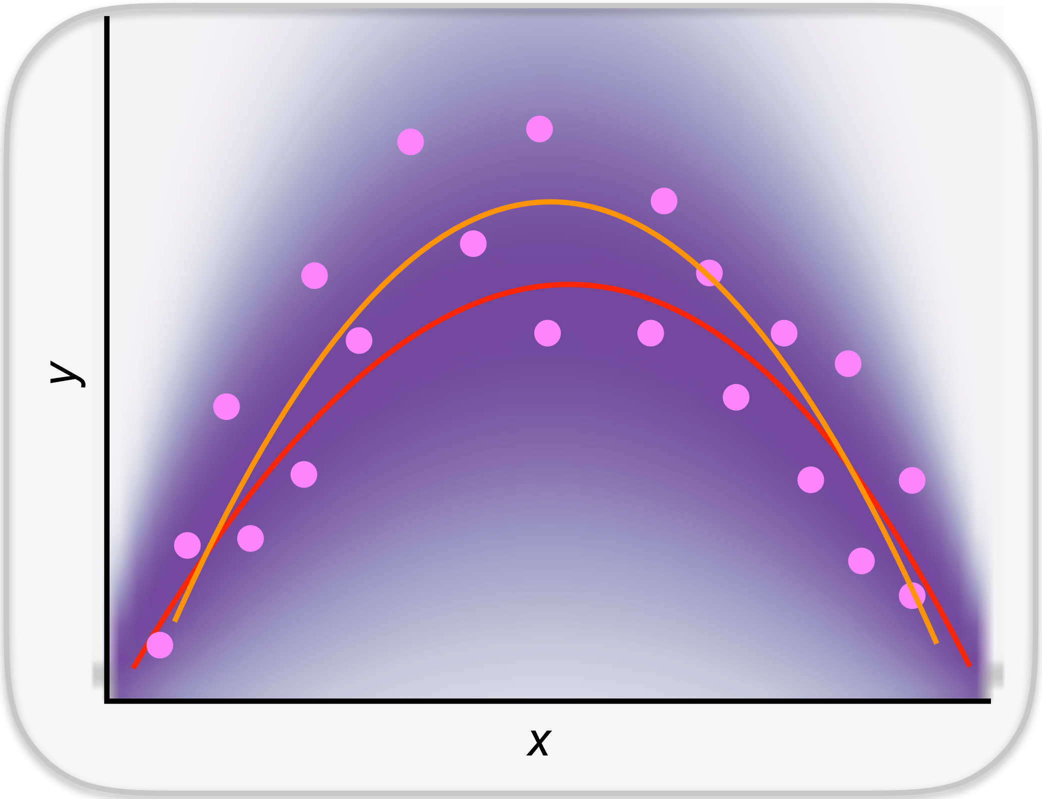
$$\eta(x) = \mathbb{E}_{XY}[Y|X = x]$$

But we only have samples:

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

Statistical Learning

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



Ideally, we want to find:

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

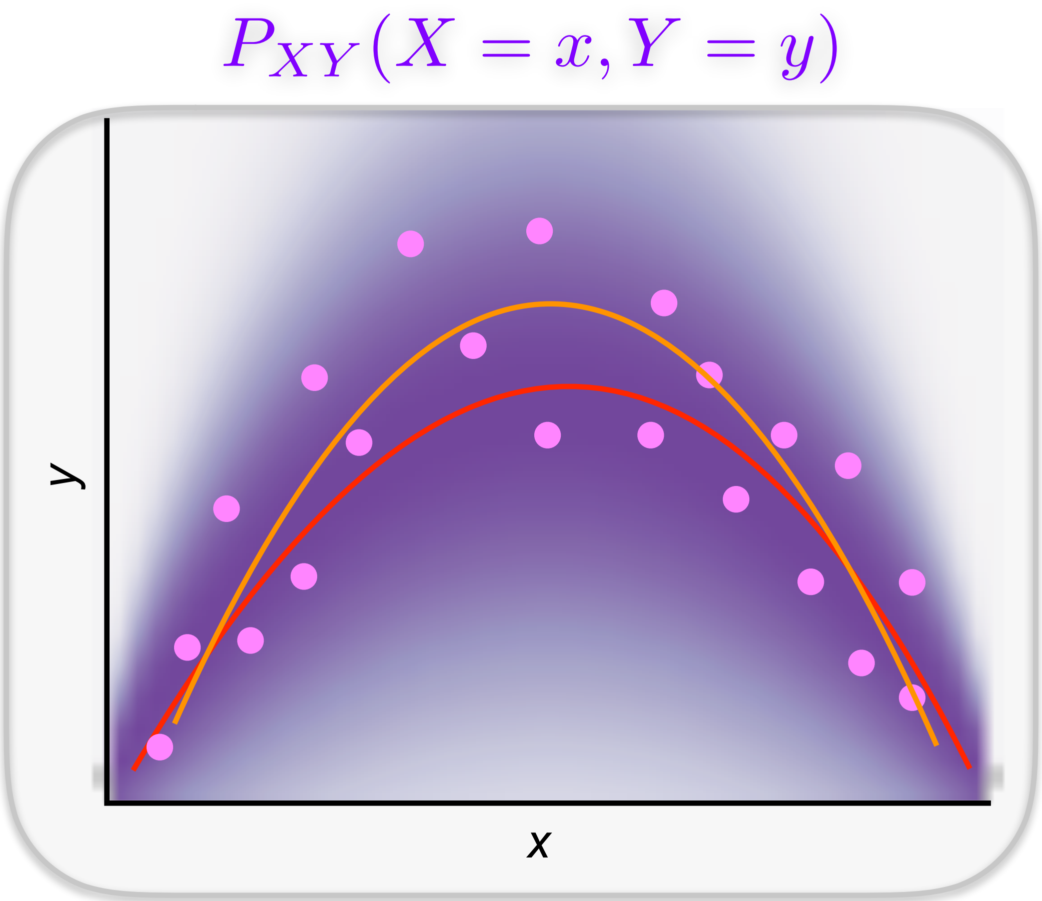
But we only have samples:

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

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

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

Statistical Learning



Ideally, we want to find:

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

But we only have samples:

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

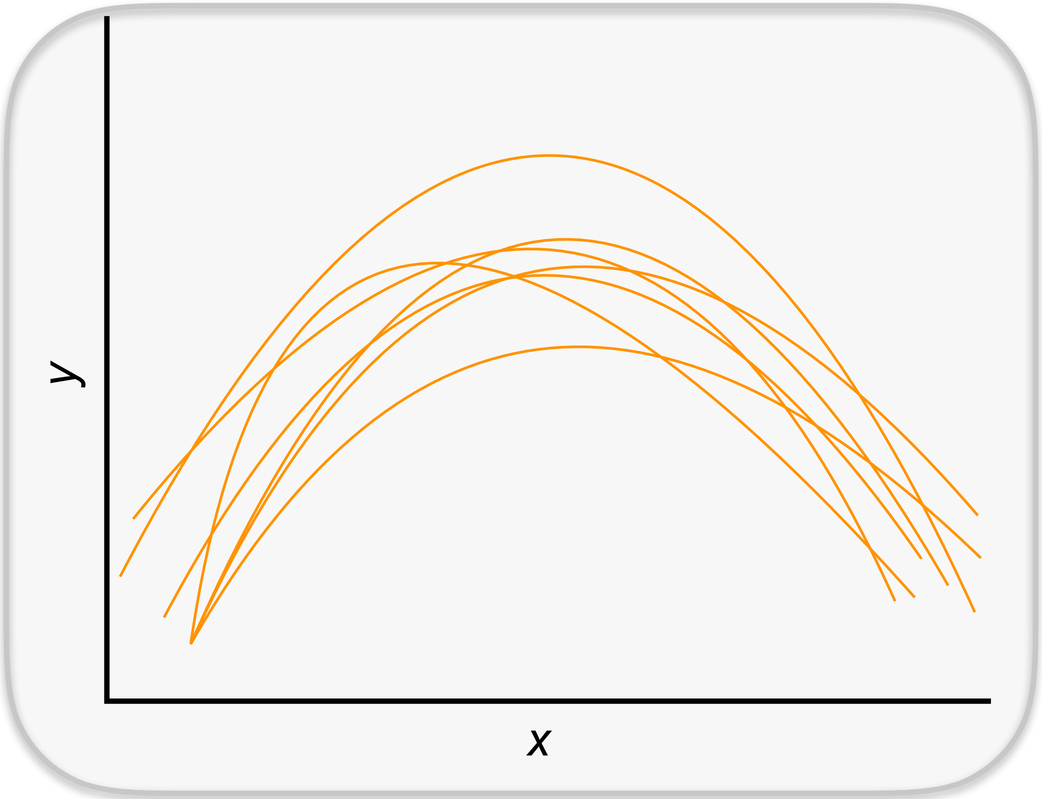
and are restricted to a function class (e.g., linear) so we compute:

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

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

Statistical Learning

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



Ideally, we want to find:

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

But we only have samples:

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

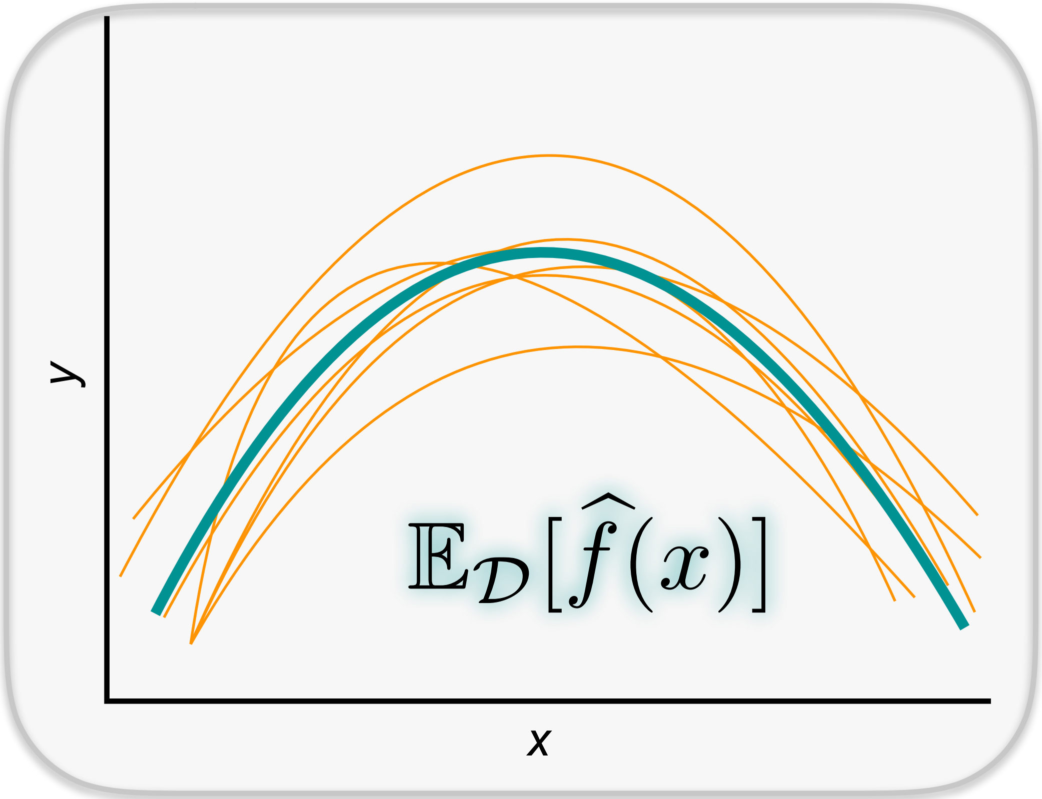
and are restricted to a function class (e.g., linear) so we compute:

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

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

Statistical Learning

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



Ideally, we want to find:

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

But we only have samples:

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

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

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

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

Bias-Variance Tradeoff

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

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

Bias-Variance Tradeoff

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

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

$$\mathbb{E}_{XY}[\mathbb{E}_{\mathcal{D}}[(Y - \widehat{f}_{\mathcal{D}}(x))^2] \big| X = x] = \mathbb{E}_{XY}[\mathbb{E}_{\mathcal{D}}[(Y - \eta(x) + \eta(x) - \widehat{f}_{\mathcal{D}}(x))^2] \big| X = x]$$

$$= \mathbb{E}_{XY}[\mathbb{E}_{\mathcal{D}}[(Y - \eta(x))^2 + 2(Y - \eta(x))(\eta(x) - \widehat{f}_{\mathcal{D}}(x)) + (\eta(x) - \widehat{f}_{\mathcal{D}}(x))^2] \big| X = x]$$

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

irreducible error
Caused by stochastic label noise

learning error
Caused by either using too "simple" of a model or not enough data to learn the model accurately

Bias-Variance Tradeoff

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

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

Bias-Variance Tradeoff

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

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

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

$$= \underbrace{(\eta(x) - \mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)])^2}_{\textbf{biased squared}} + \underbrace{\mathbb{E}_{\mathcal{D}}[(\mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)] - \widehat{f}_{\mathcal{D}}(x))^2]}_{\textbf{variance}}$$

Bias-Variance Tradeoff

$$\mathbb{E}_{XY}[\mathbb{E}_{\mathcal{D}}[(Y-\widehat{f}_{\mathcal{D}}(x))^2]\big|X=x] = \underbrace{\mathbb{E}_{XY}[(Y-\eta(x))^2\big|X=x]}_{\textbf{irreducible error}}$$

$$+\underbrace{(\eta(x)-\mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)])^2}_{\textbf{biased squared}} + \underbrace{\mathbb{E}_{\mathcal{D}}[(\mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)]-\widehat{f}_{\mathcal{D}}(x))^2]}_{\textbf{variance}}$$

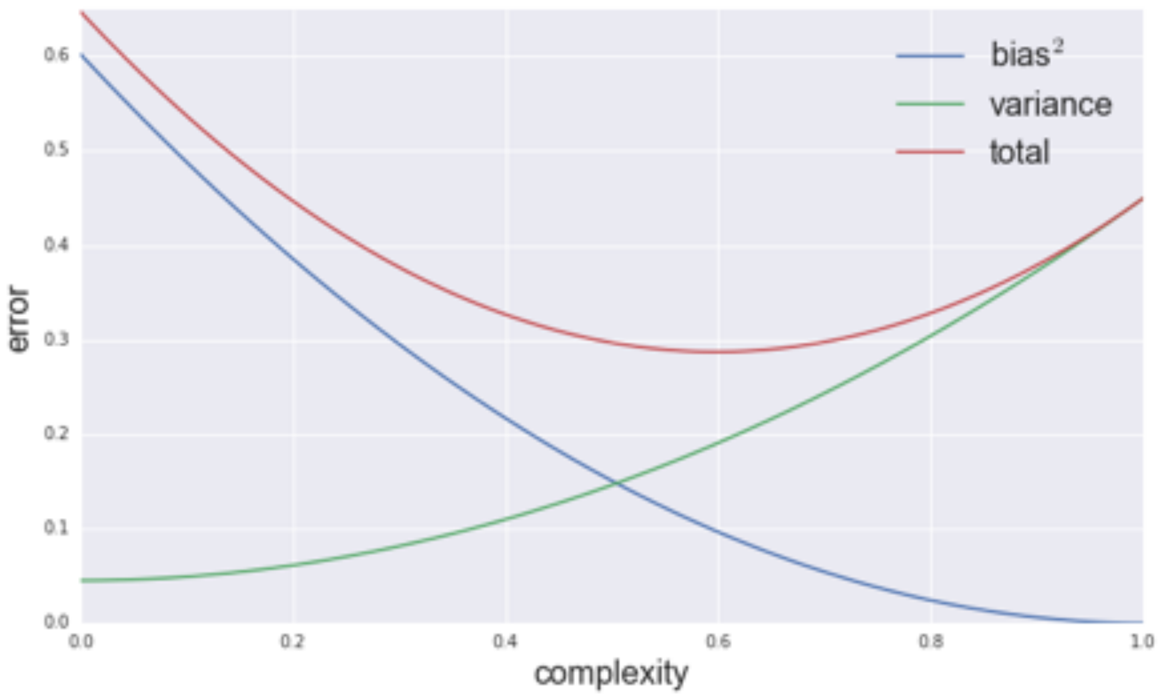
Model too simple ➔ high bias, cannot fit well to data

Model too complex ➔ high variance, small changes in data change learned function a lot

Bias-Variance Tradeoff

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

$$+ \underbrace{(\eta(x) - \mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)])^2}_{\textbf{biased squared}} + \underbrace{\mathbb{E}_{\mathcal{D}}[(\mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)] - \widehat{f}_{\mathcal{D}}(x))^2]}_{\textbf{variance}}$$



Overfitting

Machine Learning – CSE546
Kevin Jamieson
University of Washington

Oct 5, 2017

Bias-Variance Tradeoff

- Choice of hypothesis class introduces learning bias
  - More complex class → less bias
  - More complex class → 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:

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

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

Complexity grows as k grows

Training set error as a function of model complexity

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \mathcal{F}_3 \subset \dots \quad \mathcal{D} \overset{i.i.d.}{\sim} P_{XY}$$

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

TRAIN error:

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

TRUE error:

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

Training set error as a function of model complexity

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \mathcal{F}_3 \subset \ldots \qquad \mathcal{D} \overset{i.i.d.}{\sim} P_{XY}$$

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

TRAIN error:

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

TRUE error:

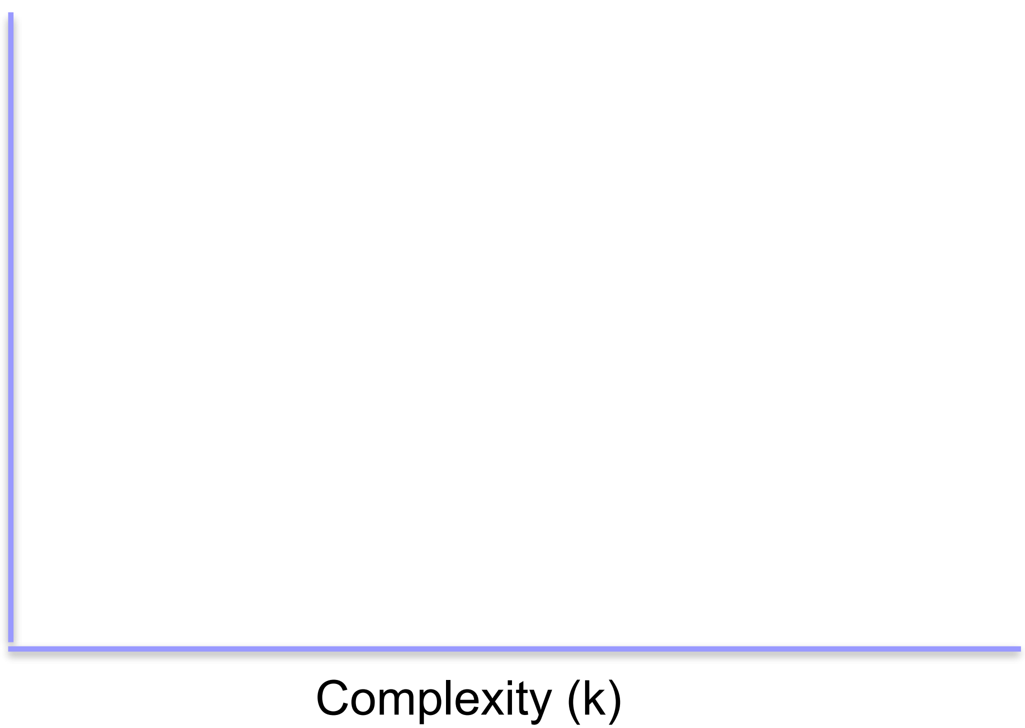
$$\mathbb{E}_{XY}[(Y - \widehat{f}^{(k)}_{\mathcal{D}}(X))^2]$$

TEST error:

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

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

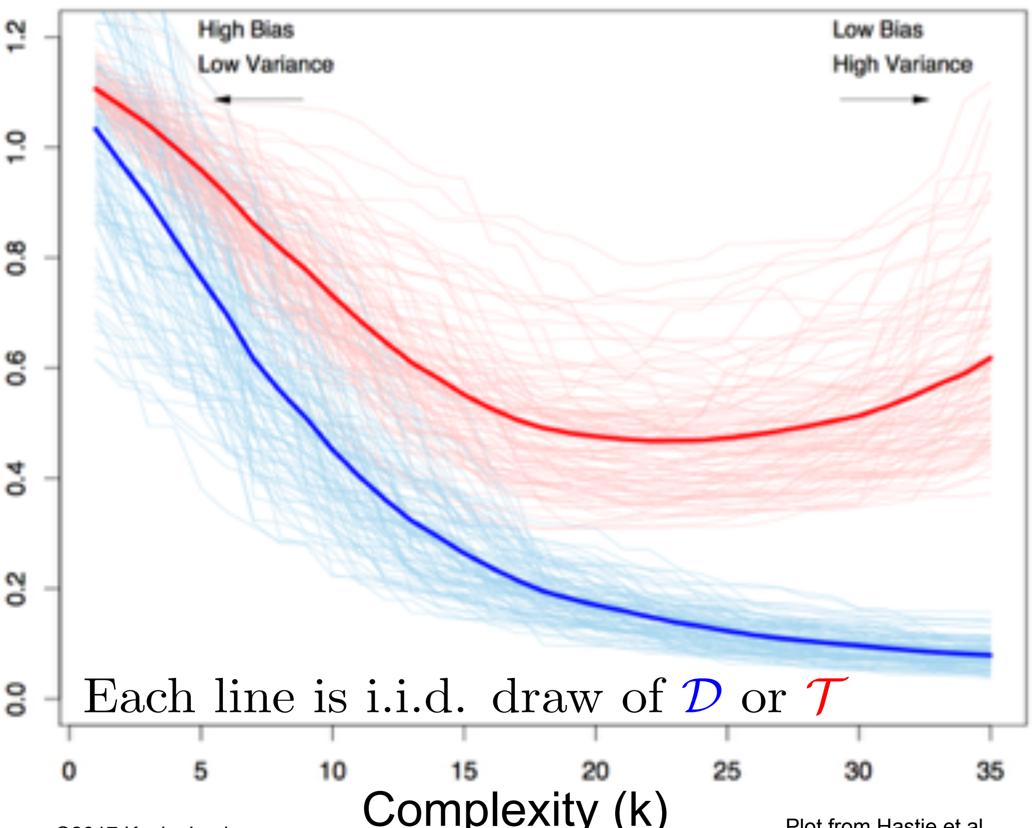
$$\boxed{\text{Important: } \mathcal{D} \cap \mathcal{T} = \emptyset}$$



Training set error as a function of model complexity

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \mathcal{F}_3 \subset \dots \qquad \mathcal{D} \overset{i.i.d.}{\sim} P_{XY}$$

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



Plot from Hastie et al

TRAIN error:

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

TRUE error:

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

TEST error:

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

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

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

Training set error as a function of model complexity

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \mathcal{F}_3 \subset \ldots \quad \mathcal{D} \overset{i.i.d.}{\sim} P_{XY}$$

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

TRAIN error:

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

TRUE error:

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

TEST error:

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

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

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

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

Test set error

- Given a dataset, **randomly** split it into two parts:
  - Training data: $\mathcal{D}$
  - Test data: $\mathcal{T}$

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

- Use training data to learn predictor
  - e.g., $\frac{1}{|\mathcal{D}|} \sum_{(x_i,y_i) \in \mathcal{D}} (y_i - \widehat{f}_{\mathcal{D}}^{(k)}(x_i))^2$
  - use training data to pick complexity k (next lecture)

- Use test data to report predicted performance

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

Overfitting

- **Overfitting:** a learning algorithm overfits the training data if it outputs a solution **w** when there exists another solution **w'** such that:

$$[error_{train}(\mathbf{w}) < error_{train}(\mathbf{w}')] \wedge [error_{true}(\mathbf{w}') < error_{true}(\mathbf{w})]$$

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
- Bounds, such as Hoeffding's inequality can help:

$$P(|\,\widehat{\theta} - \theta^*\,| \geq \epsilon) \;\leq\; 2e^{-2N\epsilon^2}$$

- More on this later this 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)

Recap

- Learning is…
  - Collect some data
    - E.g., housing info and sale price
  - Randomly split dataset into TRAIN and TEST
    - E.g., 80% and 20%, respectively
  - Choose a hypothesis class or model
    - E.g., linear
  - Choose a loss function
    - E.g., least squares
  - Choose an optimization procedure
    - E.g., set derivative to zero to obtain estimator
  - Justifying the accuracy of the estimate
    - E.g., report TEST error