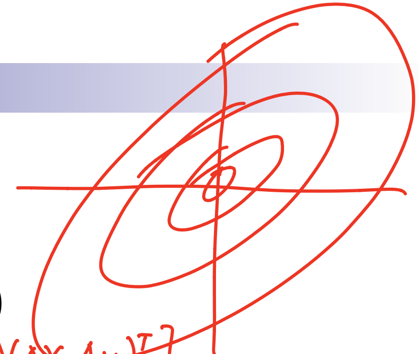
Warm up

Homework due tonight! 11:59 PM

B, $B^{1/2} := A : AA = B$



Let $X \sim \mathcal{N}(\mu, \Sigma)$ where $X \in \mathbb{R}^d$

1. Let $Y = AX + b$. For what $\widetilde{\mu}, \widetilde{\Sigma}$ is $Y \sim \mathcal{N}(\widetilde{\mu}, \widetilde{\Sigma})$

$\tilde{\mu} = \mathbb{E}[Y] = A\mathbb{E}[X] + b = A\mu + b$

$\tilde{\Sigma} = \mathbb{E}[(Y-\mathbb{E}[Y])(Y-\mathbb{E}[Y])^T] = \mathbb{E}[(AX-A\mu)(AX-A\mu)^T] = \mathbb{E}[A(X-\mu)(X-\mu)^T A^T] = A\Sigma A^T$

2. Suppose I can generate independent Gaussians $Z \sim \mathcal{N}(0,1)$ (e.g., `numpy.random.randn`). How can I use this to generate X?

$Z = \begin{bmatrix} z_1 \\ \vdots \\ z_d \end{bmatrix}$ $\quad \hat{X} = \mu + \Sigma^{1/2} Z$ $\quad \mathbb{E}[\hat{X}] = \mu$, $\mathbb{E}[(\hat{X}-\mathbb{E}\hat{X})(\hat{X}-\mathbb{E}\hat{X})^T]$

$= \mathbb{E}[\Sigma^{1/2} Z Z^T \Sigma^{1/2}]$

$= \Sigma$

3. What is $\mathbb{E}[X^T \Sigma^{-1} X]$? $= \mathbb{E}[X^T \Sigma^{-1/2} \Sigma^{-1/2} X]$

Assume $\mu = 0$

$= \mathbb{E}[(\Sigma^{-1/2} X)^T (\Sigma^{-1/2} X)]$

$= d$

$= \mathbb{E}[\text{Trace}(X^T \Sigma^{-1} X)] = \mathbb{E}[\text{Tr}(X X^T \Sigma^{-1})] = \text{Tr}(\Sigma \Sigma^{-1}) = \text{Tr}(I) = d$

$\text{Tr}(AB) = \text{Tr}(BA)$

©2017 Kevin Jamieson

Bias-Variance Tradeoff

Machine Learning – CSE546
Kevin Jamieson
University of Washington

Oct 4, 2018

©2018 Kevin Jamieson

Statistical Learning

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

Goal: Predict Y given X

Find function η that minimizes

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

©2018 Kevin Jamieson

Statistical Learning

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

Goal: Predict Y given X

Find function η that minimizes

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

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

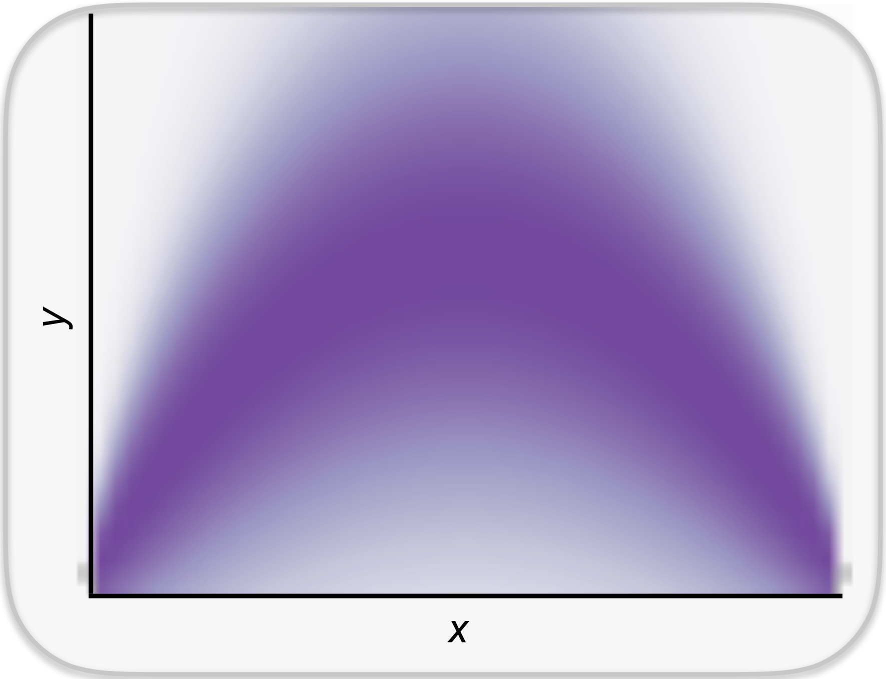
Under LS loss, optimal predictor: $\eta(x) = \underline{\mathbb{E}_{Y|X}[Y|X = x]}$

©2018 Kevin Jamieson

Statistical Learning

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

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

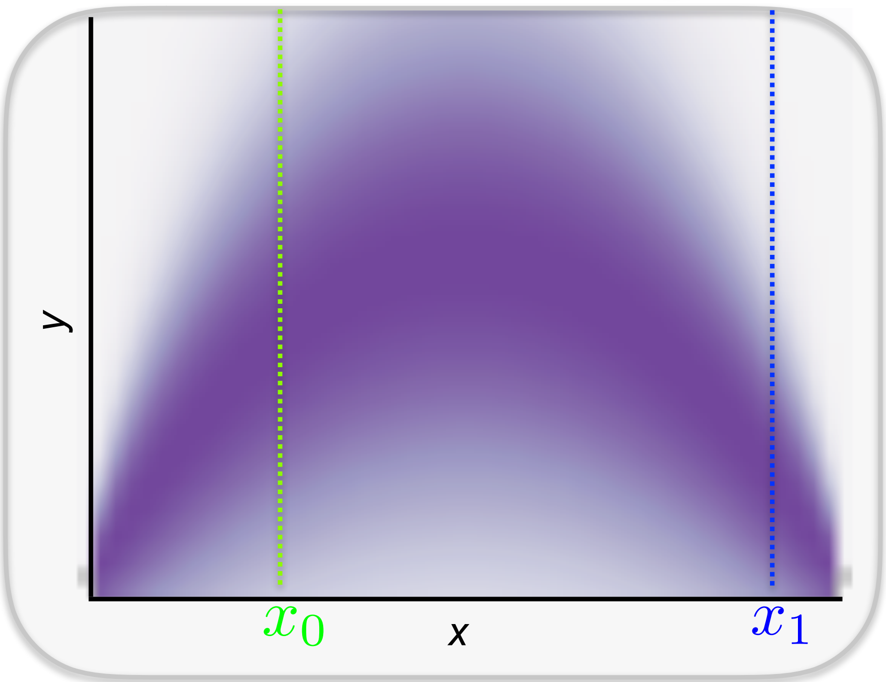


©2017 Kevin Jamieson

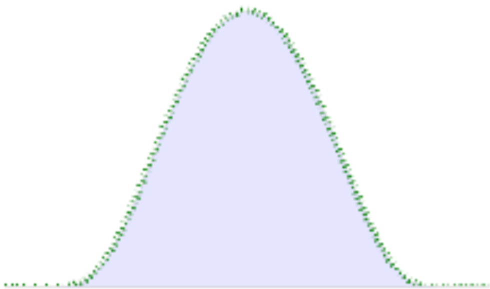
Statistical Learning

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

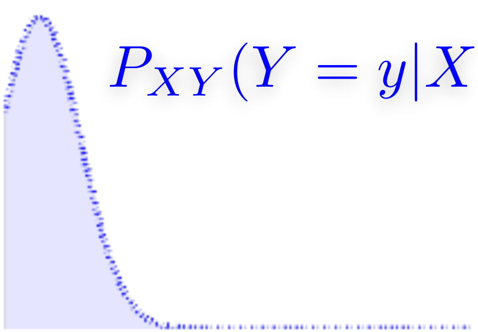
$$P_{XY}(X = x, Y = y)$$



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



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

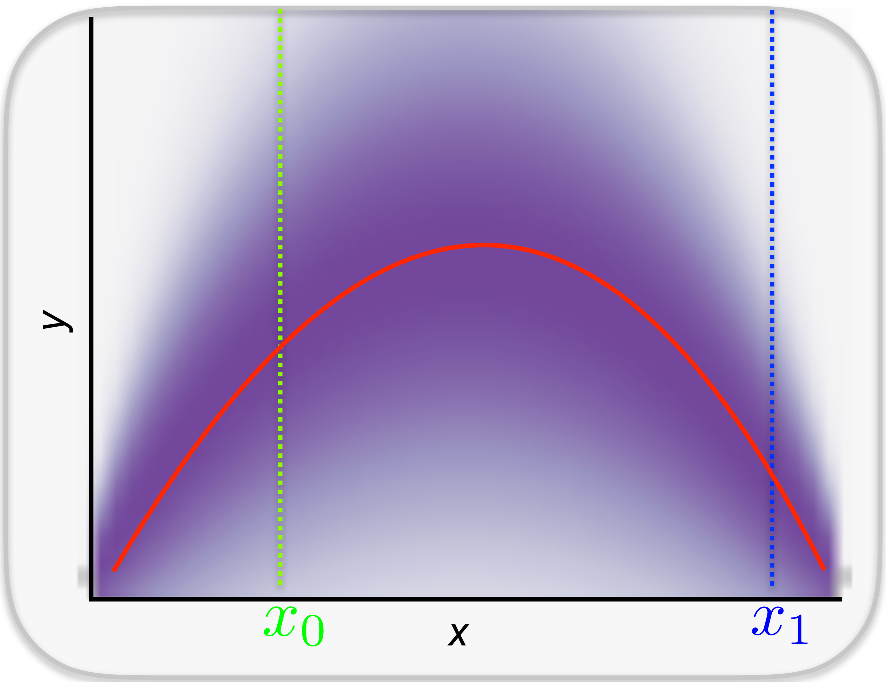


©2018 Kevin Jamieson

Statistical Learning

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

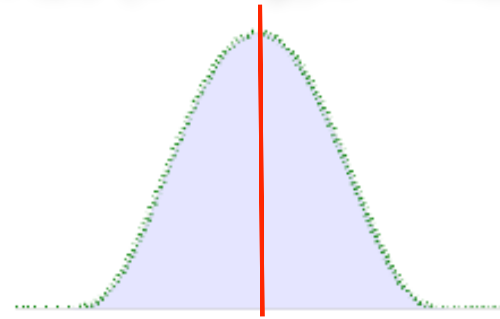
$$P_{XY}(X = x, Y = y)$$

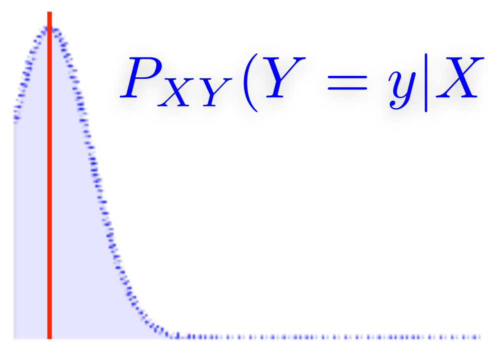


Ideally, we want to find:

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

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



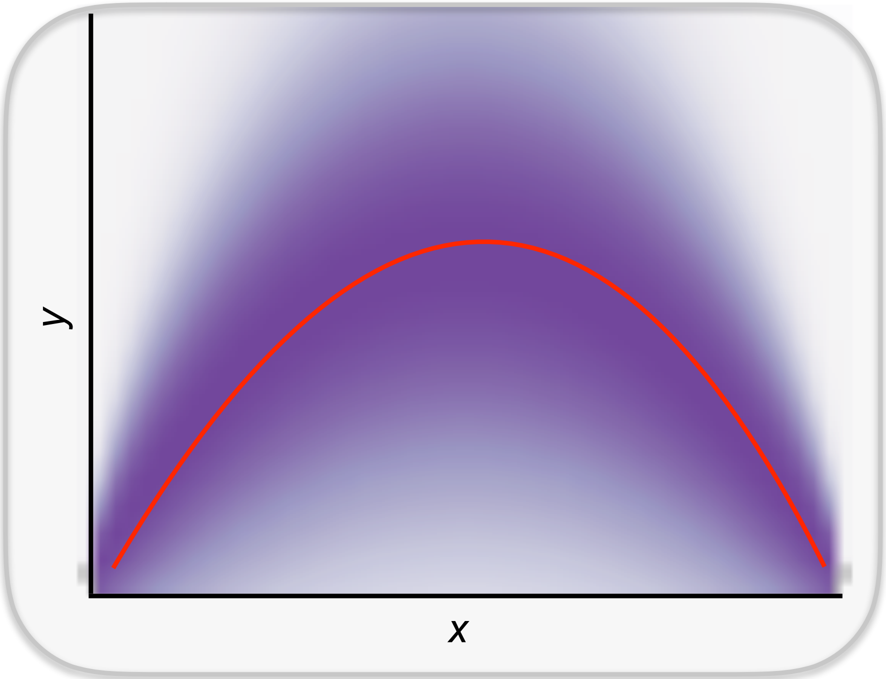


$$P_{XY}(Y = y|X = x_1)$$

©2018 Kevin Jamieson

Statistical Learning

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



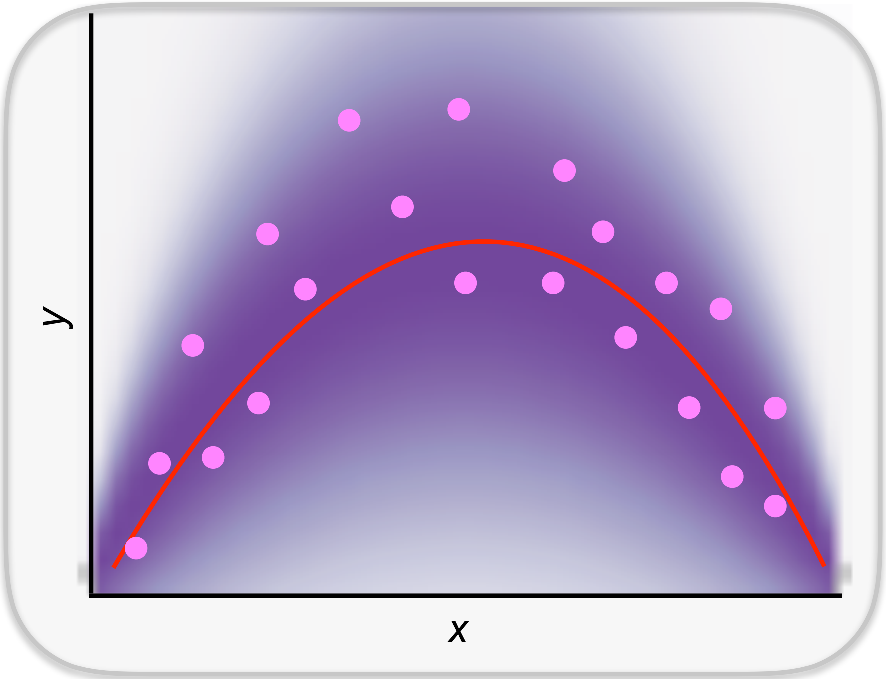
Ideally, we want to find:

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

©2018 Kevin Jamieson

Statistical Learning

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



Ideally, we want to find:

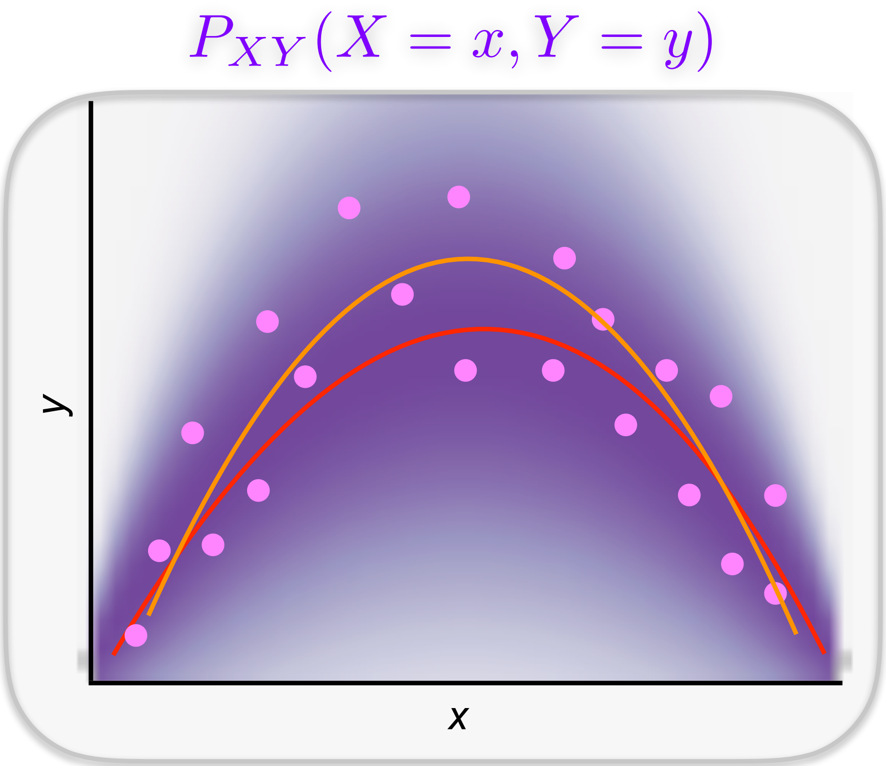
$$\eta(x) = \mathbb{E}_{Y|X}[Y|X = x]$$

But we only have samples:

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

©2018 Kevin Jamieson

Statistical Learning



Ideally, we want to find:

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

But we only have samples:

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

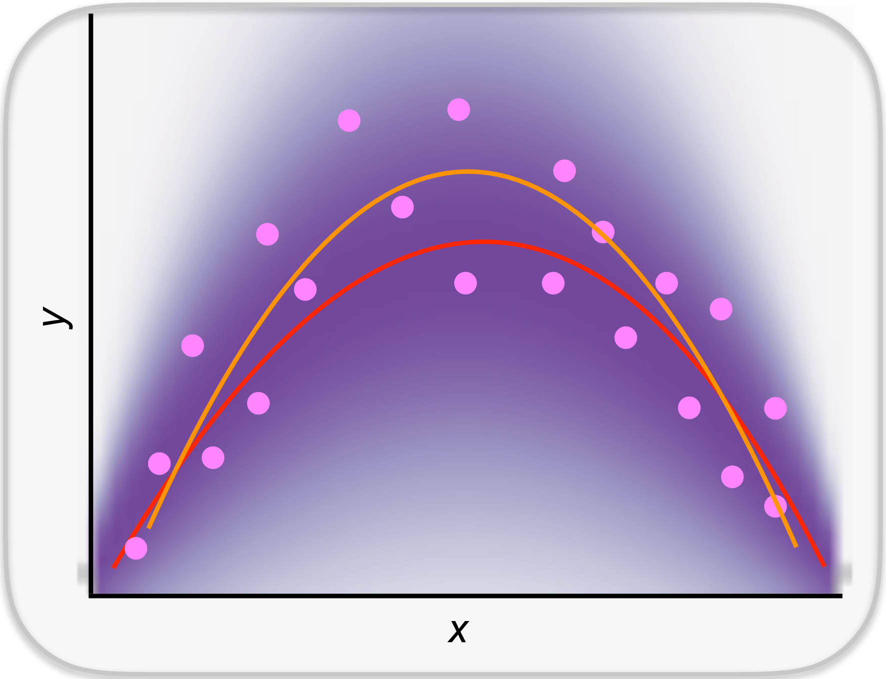
and are restricted to a function class (e.g., linear) so we compute:

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

©2018 Kevin Jamieson

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:

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

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

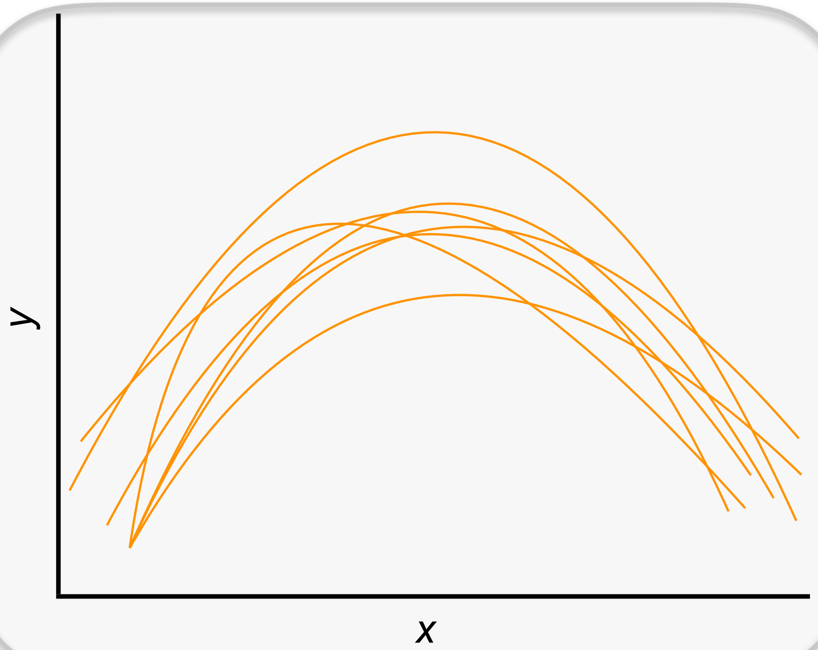
$$\widehat{f} = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2$$

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

©2018 Kevin Jamieson

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:

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

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

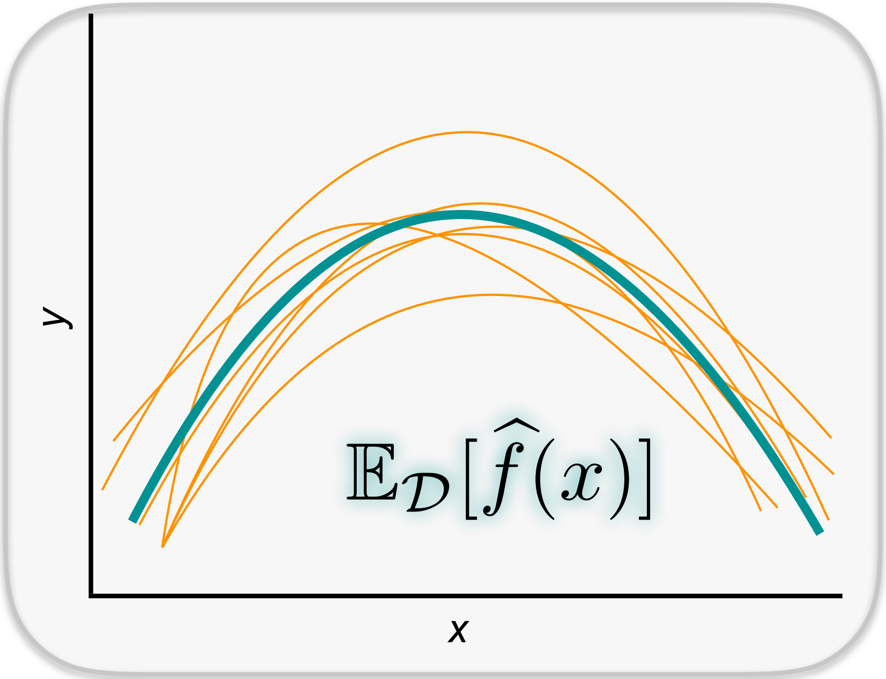
$$\widehat{f} = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2$$

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

©2018 Kevin Jamieson

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:

$(x_i, y_i) \overset{i.i.d.}{\sim} P_{XY}$ 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 $\widehat{f}$

©2018 Kevin Jamieson

Bias-Variance Tradeoff

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

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

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

©2018 Kevin Jamieson

Bias-Variance Tradeoff

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

$$\begin{aligned}
\underline{\mathbb{E}_{Y|X}[\mathbb{E}_{\mathcal{D}}[(Y - \widehat{f}_{\mathcal{D}}(x))^2] \big| X = x]} &= \mathbb{E}_{Y|X}[\mathbb{E}_{\mathcal{D}}[(Y - \eta(x) + \eta(x) - \widehat{f}_{\mathcal{D}}(x))^2] \big| X = x] \\
&= \mathbb{E}_{Y|X}\Big[\mathbb{E}_{\mathcal{D}}[(Y - \eta(x))^2 + 2(Y - \eta(x))(\eta(x) - \widehat{f}_{\mathcal{D}}(x)) \\
&\quad + (\eta(x) - \widehat{f}_{\mathcal{D}}(x))^2] \big| X = x\Big] \\
&= \underbrace{\mathbb{E}_{Y|X}[(Y - \eta(x))^2 \big| X = x]}_{\text{irreducible error}} + \underbrace{\mathbb{E}_{\mathcal{D}}[(\eta(x) - \widehat{f}_{\mathcal{D}}(x))^2]}_{\text{learning error}}
\end{aligned}$$

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

©2018 Kevin Jamieson

Bias-Variance Tradeoff

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

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

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

©2018 Kevin Jamieson

Bias-Variance Tradeoff

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

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

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

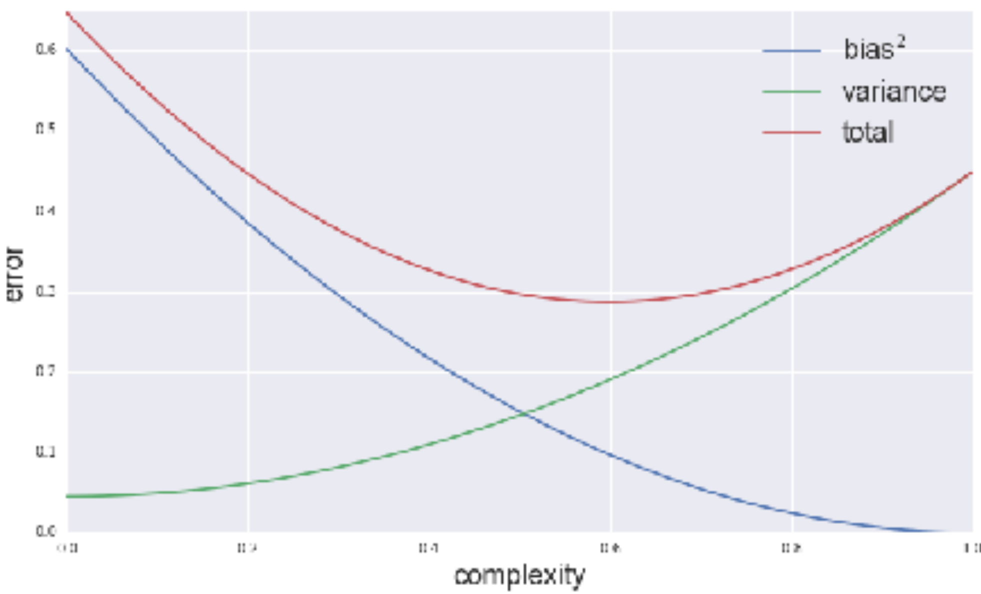
$$= \underbrace{(\eta(x) - \mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)])^2}_{\textbf{biased squared}} + \underbrace{\mathbb{E}_{\mathcal{D}}[(\mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)] - \widehat{f}_{\mathcal{D}}(x))^2]}_{\textbf{variance}}$$

©2018 Kevin Jamieson

Bias-Variance Tradeoff

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

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



Example: Linear LS $\quad \mathbf{Y} = \mathbf{X}w + \epsilon$

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

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

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

$$\widehat{f}_{\mathcal{D}}(x) = \hat{w}^T x$$

©2018 Kevin Jamieson

Example: Linear LS $\quad \mathbf{Y} = \mathbf{X}w + \epsilon$

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

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

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

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

$$\mathbb{E}_{XY}[\overbrace{(Y - \eta(x))}^{\varepsilon}{}^2 | X = x] = \sigma^2 \qquad (\eta(x) - \mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)])^2 = 0$$

irreducible error **biased squared**

$$\mathbb{E}_D[\hat{f}_D(x)] = w^T x$$

Example: Linear LS $\quad \mathbf{Y} = \mathbf{X}w + \epsilon$

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

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

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

$$\mathbb{E}_{\mathcal{D}}[(\overbrace{\mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)]}^{w^T x} - \widehat{f}_{\mathcal{D}}(x))^2] = \mathbb{E}_{D}\left[\left(\varepsilon^T X (X^T X)^{-1} x\right)^2\right]$$

variance

$$= \mathbb{E}_D\left[x^T (X^T X)^{-1} X^T \varepsilon \varepsilon^T X (X^T X)^{-1} x\right]$$

$$= \sigma^2 \mathbb{E}_D\left[x^T (X^T X)^{-1} \cancel{X^T X (X^T X)^{-1}} x\right] = \sigma^2 \mathbb{E}_D\left[\underline{x^T (X^T X)^{-1} x}\right]$$

$$= \sigma^2 \mathbb{E}_D\left[\mathrm{Trace}\left((X^T X)^{-1} x x^T\right)\right]$$

$$= \underline{\sigma^2 \, \mathrm{Trace}\left(\tfrac{1}{n} \Sigma^{-1} x x^T\right)}$$

©2018 Kevin Jamieson

$$X^T X = n \frac{1}{n} \underbrace{\sum_{i=1}^{n} x_i x_i^T}$$

$$\xrightarrow{n \to \infty} n \Sigma$$

$$\boxed{\mathbb{E}[x x^T] =: \Sigma}$$

$$\boxed{\text{Assume } X^T X = n \Sigma}$$

$(X, Y) \sim P_{XY}$

$$\Rightarrow \mathbb{E}_D\left[\left(\mathbb{E}_D[\hat{f}_D(\cancel{X})] - \hat{f}_D(\cancel{X})\right)^2\right] = \frac{\sigma^2}{n} \operatorname{Tr}(\Sigma^{-1} \Sigma)$$

$$= \frac{\sigma^2}{n} \operatorname{Tr}(I)$$

$$= \frac{d\sigma^2}{n}$$

Example: Linear LS $\mathbf{Y} = \mathbf{X}w + \epsilon$

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

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

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

$$\underbrace{\mathbb{E}_{\mathcal{D}}[(\mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)] - \widehat{f}_{\mathcal{D}}(x))^2]}_{\text{variance}} = \mathbb{E}_{\underset{=(X,\varepsilon)}{\mathcal{D}}}[x^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T \underline{\epsilon\epsilon^T} \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}x]$$

$$= \mathbb{E}_{\underset{=(X,\varepsilon)}{\mathcal{D}}}[\sigma^2 x^T(\mathbf{X}^T\mathbf{X})^{-1}x]$$

$$= \sigma^2\mathbb{E}_{\mathcal{D}}[\text{Trace}((\mathbf{X}^T\mathbf{X})^{-1}xx^T)]$$

$$\mathbf{X}^T\mathbf{X} = \sum_{i=1}^n x_i x_i^T \overset{n \text{ large}}{\rightarrow} n\Sigma \qquad \Sigma = \mathbb{E}[XX^T], \quad X \sim P_X$$

$$\mathbb{E}_{X=x}\left[\mathbb{E}_{\mathcal{D}}[(\mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)] - \widehat{f}_{\mathcal{D}}(x))^2]\right] = \frac{\sigma^2}{n}\mathbb{E}_X[\text{Trace}(\Sigma^{-1}XX^T)] = \frac{d\sigma^2}{n}$$

©2018 Kevin Jamieson

Example: Linear LS $\mathbf{Y} = \mathbf{X}w + \epsilon$

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

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

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

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

$$\underbrace{\mathbb{E}_{XY}\left[(Y - \eta(x))^2 \middle| X = x\right]}_{\text{irreducible error}} = \sigma^2$$

$$\underbrace{(\eta(x) - \mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)])^2}_{\text{biased squared}} = 0$$

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

Overfitting

Machine Learning – CSE546

Kevin Jamieson
University of Washington

Oct 4, 2018

©2018 Kevin Jamieson

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:

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

©2018 Kevin Jamieson

Training set error as a function of model complexity

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

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

Complexity (k)

TRAIN error:

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

TRUE error:

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

TEST error:

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

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

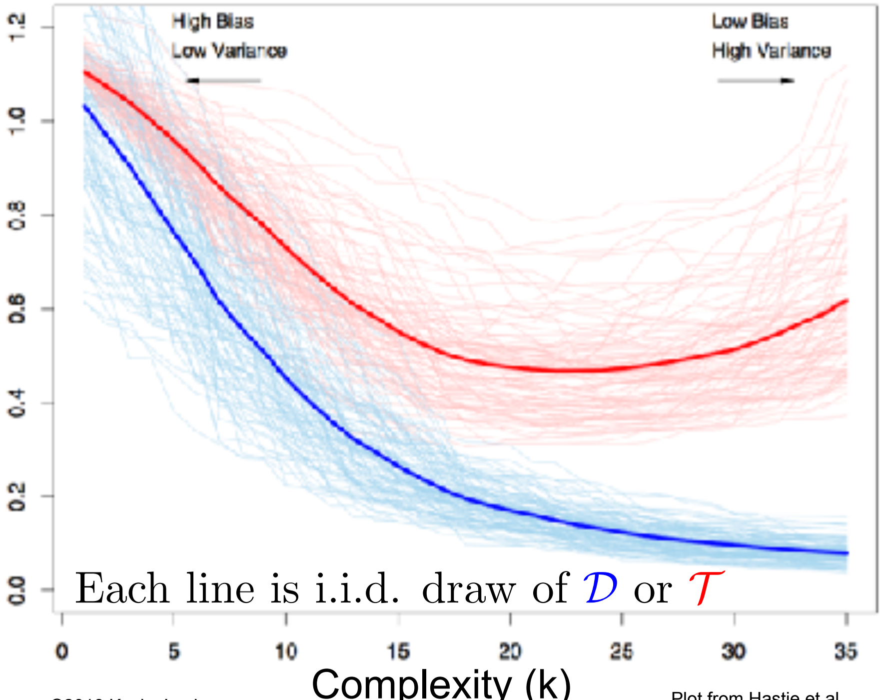
$$\boxed{\text{Important: } \mathcal{D} \cap \mathcal{T} = \emptyset}$$

©2018 Kevin Jamieson

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:

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

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

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

©2018 Kevin Jamieson

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:

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

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.

©2018 Kevin Jamieson

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
- Use test data to report predicted performance

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

©2018 Kevin Jamieson

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 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)

©2018 Kevin Jamieson

Regularization

Machine Learning – CSE546
Kevin Jamieson
University of Washington

October 4, 2016

©2017 Kevin Jamieson

Regularization in Linear Regression

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

$$= \arg\min_w (\mathbf{y} - \mathbf{X}w)^T(\mathbf{y} - \mathbf{X}w)$$

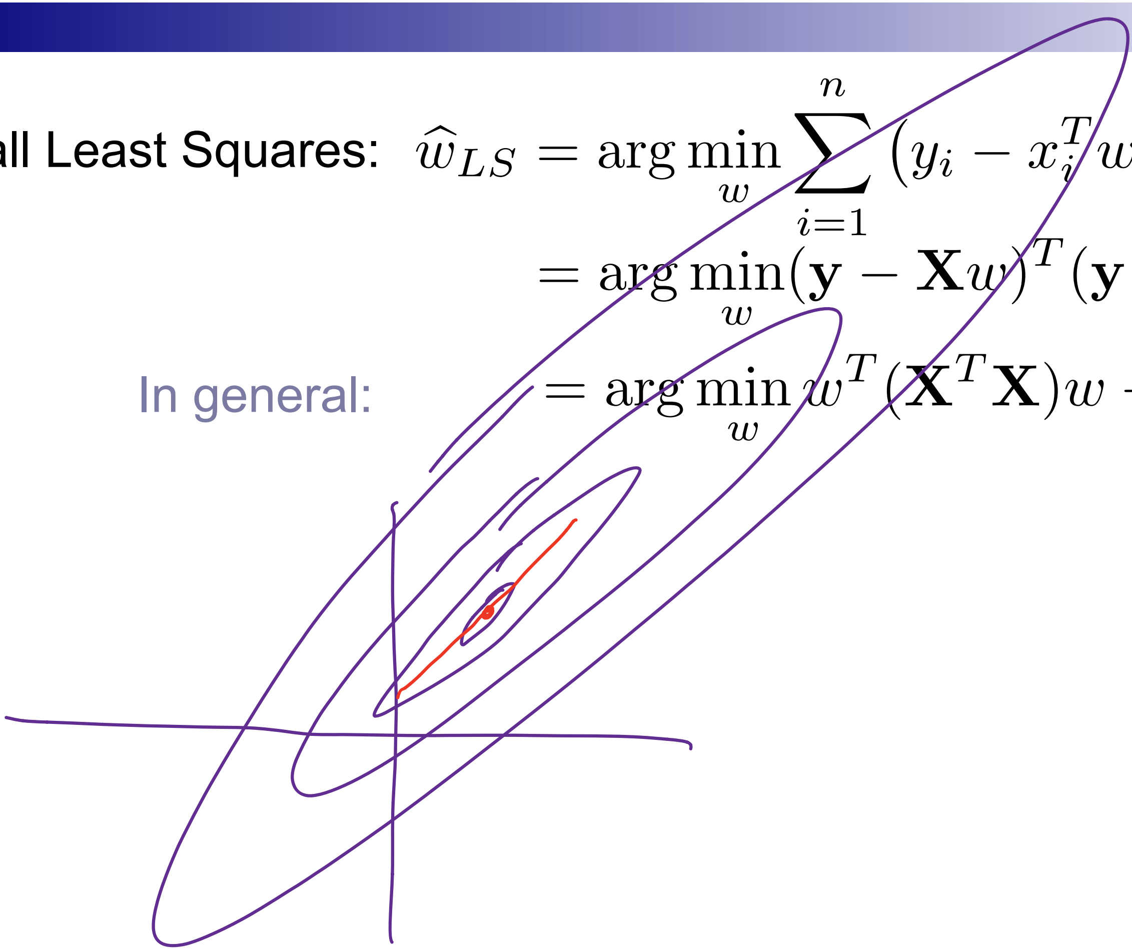
when $(\mathbf{X}^T\mathbf{X})^{-1}$ exists.... $= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$

©2017 Kevin Jamieson

Regularization in Linear Regression

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

In general:
$$= \arg\min_w w^T(\mathbf{X}^T\mathbf{X})w - 2y^T\mathbf{X}w$$



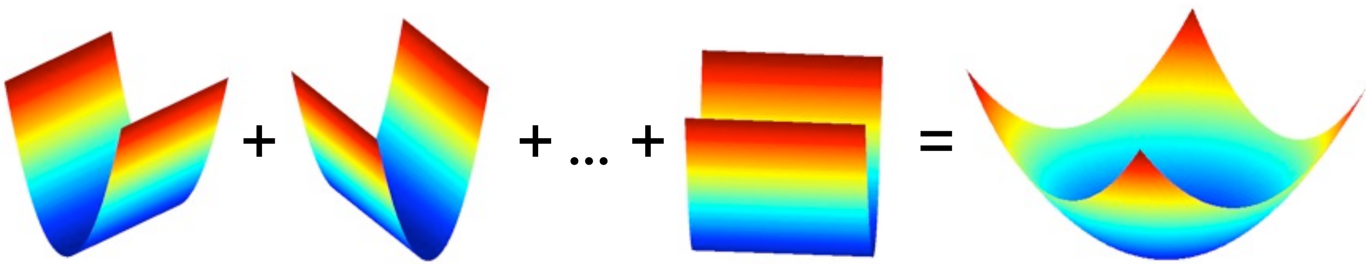
©2017 Kevin Jamieson

Regularization in Linear Regression

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

$$= \arg\min_w (\mathbf{y} - \mathbf{X}w)^T(\mathbf{y} - \mathbf{X}w)$$

In general: $$= \arg\min_w w^T(\mathbf{X}^T\mathbf{X})w - 2y^T\mathbf{X}w$$



$$(y_1 - x_1^T w)^2 + (y_2 - x_2^T w)^2 + \cdots + (y_n - x_n^T w)^2 = \sum_{i=1}^n (y_i - x_i^T w)^2$$

What if $x_i \in \mathbb{R}^d$ and $d > n$?

©2017 Kevin Jamieson

Regularization in Linear Regression

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

When $x_i \in \mathbb{R}^d$ and $d > n$ the objective function is flat in some directions:

©2017 Kevin Jamieson

Regularization in Linear Regression

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

When $x_i \in \mathbb{R}^d$ and $d > n$ the objective function is flat in some directions:

Implies optimal solution is *underconstrained* and unstable due to lack of curvature:

- small changes in training data result in large changes in solution
- often the *magnitudes* of *w* are "very large"

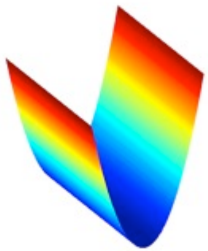
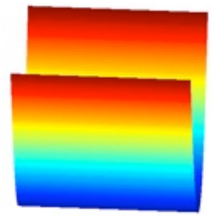
Regularization imposes "simpler" solutions by a "complexity" penalty

©2017 Kevin Jamieson

Ridge Regression

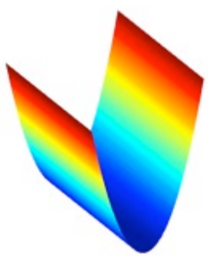
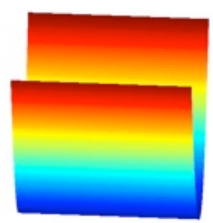
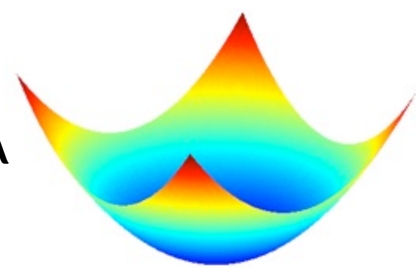
- Old Least squares objective:

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

 +  + … + 

- Ridge Regression objective:

$$\widehat{w}_{ridge} = \arg\min_w \sum_{i=1}^n \left(y_i - x_i^T w\right)^2 + \lambda ||w||_2^2$$

 +  + … +  + λ 

Minimizing the Ridge Regression Objective

$$\widehat{w}_{ridge} = \arg\min_w \sum_{i=1}^n \left(y_i - x_i^T w\right)^2 + \lambda ||w||_2^2$$

$$||z||_2^2 = z^T z$$

$$= \arg\min_w ||Xw - y||_2^2 + \lambda ||w||_2^2$$

$$\nabla_w = \cancel{2} X^T(Xw - y) + \cancel{2}\lambda w = 0$$

$$X^T X w + \lambda w = X^T y$$

$$(X^T X + \lambda I) w = X^T y$$

$$\widehat{w}_{Ridge} = (X^T X + \lambda I)^{-1} X^T y$$

©2017 Kevin Jamieson

Shrinkage Properties

$\epsilon \sim \mathcal{N}(0, \sigma^2 I)$

$$\widehat{w}_{ridge} = (\mathbf{X}^T\mathbf{X} + \lambda I)^{-1}\mathbf{X}^T\mathbf{y}$$

- ***Assume:*** $\mathbf{X}^T\mathbf{X} = nI$ ***and*** $\mathbf{y} = \mathbf{X}w + \boldsymbol{\epsilon}$

$$\hat{w} = (X^TX+\lambda I)^{-1}X^TXw + (X^TX+\lambda I)^{-1}X^T\varepsilon$$

$$= (X^TX+\lambda I)^{-1}(X^TX+\lambda I - \lambda I)w + (X^TX+\lambda I)^{-1}X^T\varepsilon$$

$$= w - \lambda(X^TX+\lambda I)^{-1}w + (X^TX+\lambda I)^{-1}X^T\varepsilon$$

$$= w - \lambda(nI+\lambda I)^{-1}w + (nI+\lambda I)^{-1}X^T\varepsilon$$

$$= w - \frac{\lambda}{n+\lambda}w + \frac{1}{n+\lambda}X^T\varepsilon$$

©2017 Kevin Jamieson

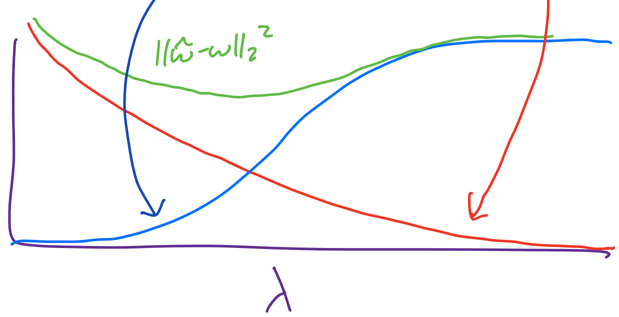
$$\mathbb{E}\|\hat{w}-w\|_2^2 = \left\|\frac{\lambda}{n+\lambda}w\right\|_2^2 + \cancel{2\left(\frac{\lambda}{n+\lambda}w\right)^T\mathbb{E}\left[\frac{1}{n+\lambda}X^T\varepsilon\right]} = 0$$

$$+\,\mathbb{E}\left[\frac{1}{(n+\lambda)^2}\underline{\varepsilon^T X X^T \varepsilon}\right]$$

$$= \frac{\lambda^2}{(n+\lambda)^2}\|w\|_2^2 + \frac{1}{(n+\lambda)^2}\mathbb{E}\left[Tr\left(X^T\varepsilon\varepsilon^T X\right)\right]$$

$$= \frac{\lambda^2}{(n+\lambda)^2}\|w\|_2^2 + \frac{\sigma^2}{(n+\lambda)^2}Tr\left(X^TX\right), \quad Tr(nI) = nd$$

$$= \underbrace{\frac{\lambda^2}{(n+\lambda)^2}\|w\|_2^2}_{\text{bias}} + \underbrace{\frac{nd\sigma^2}{(n+\lambda)^2}}_{\text{variance}}$$



Shrinkage Properties

$$\epsilon \sim \mathcal{N}(0, \sigma^2 I)$$

$$\widehat{w}_{ridge} = (\mathbf{X}^T\mathbf{X} + \lambda I)^{-1}\mathbf{X}^T\mathbf{y}$$

- **Assume:** $\mathbf{X}^T\mathbf{X} = nI$ **and** $\mathbf{y} = \mathbf{X}w + \boldsymbol{\epsilon}$

$$\begin{aligned}\widehat{w}_{ridge} &= (\mathbf{X}^T\mathbf{X} + \lambda I)^{-1}\mathbf{X}^T(\mathbf{X}w + \boldsymbol{\epsilon}) \\ &= \frac{n}{n+\lambda}w + \frac{1}{n+\lambda}\mathbf{X}^T\boldsymbol{\epsilon}\end{aligned}$$

$$\mathbb{E}\|\widehat{w}_{ridge} - w\|^2 = \frac{\lambda^2}{(n+\lambda)^2}\|w\|^2 + \frac{dn\sigma^2}{(n+\lambda)^2}$$

$$\lambda^* = \frac{d\sigma^2}{\|w\|^2}$$

©2017 Kevin Jamieson

Ridge Regression: Effect of Regularization

$$\widehat{w}_{ridge} = \arg\min_{w} \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 + \lambda ||w||_2^2$$

- Solution is indexed by the regularization parameter λ
- Larger λ
- Smaller λ
- As λ → 0
- As λ → ∞

©2017 Kevin Jamieson

Ridge Regression: Effect of Regularization

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

$$\widehat{w}_{\mathcal{D},ridge}^{(\lambda)} = \arg\min_w \frac{1}{|\mathcal{D}|} \sum_{(x_i,y_i)\in\mathcal{D}} (y_i - x_i^T w)^2 + \lambda ||w||_2^2$$

TRAIN error:

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

TRUE error:

$$\mathbb{E}[(Y - X^T \widehat{w}_{\mathcal{D},ridge}^{(\lambda)})^2]$$

TEST error:

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

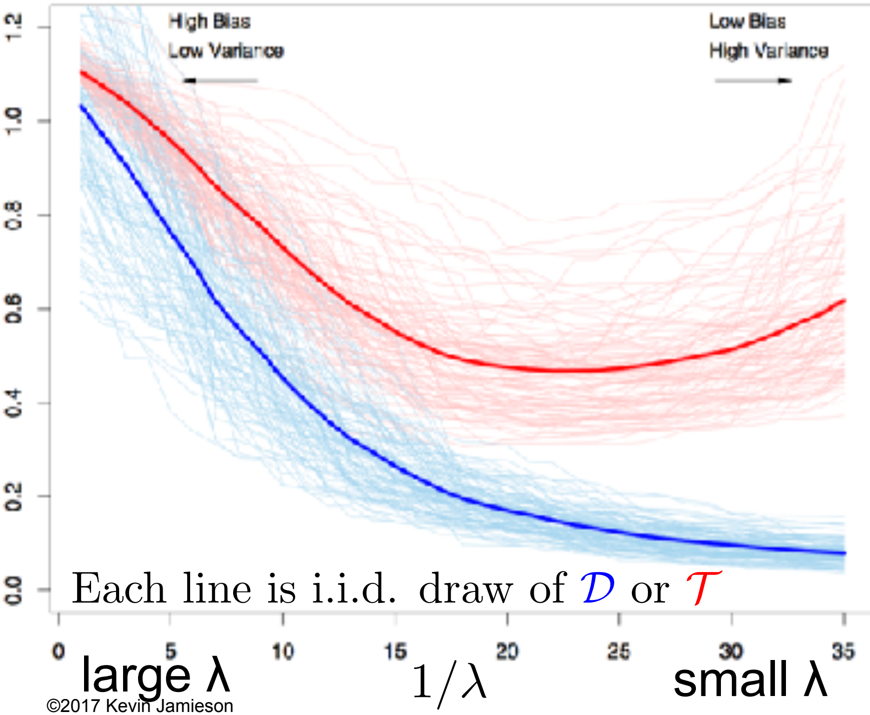
$$\frac{1}{|\mathcal{T}|} \sum_{(x_i,y_i)\in\mathcal{D}} (y_i - x_i^T \widehat{w}_{\mathcal{D},ridge}^{(\lambda)})^2$$

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

Ridge Regression: Effect of Regularization

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

$$\widehat{w}_{\mathcal{D},ridge}^{(\lambda)} = \arg\min_w \frac{1}{|\mathcal{D}|} \sum_{(x_i,y_i)\in\mathcal{D}} (y_i - x_i^T w)^2 + \lambda ||w||_2^2$$



TRAIN error:

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

TRUE error:

$$\mathbb{E}[(Y - X^T \widehat{w}_{\mathcal{D},ridge}^{(\lambda)})^2]$$

TEST error:

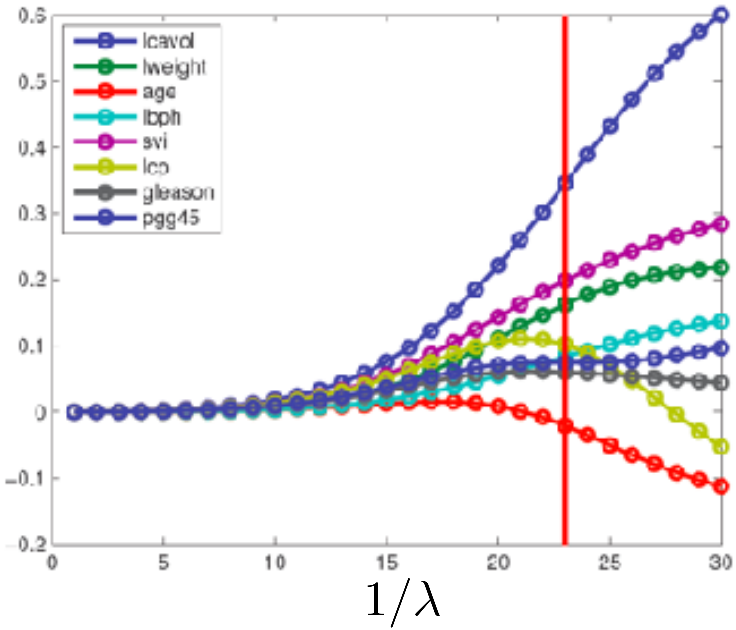
$$\mathcal{T} \overset{i.i.d.}{\sim} P_{XY}$$

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

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

©2017 Kevin Jamieson

Ridge Coefficient Path



From
Kevin Murphy
textbook

- Typical approach: select λ using cross validation, up next

©2017 Kevin Jamieson

What you need to know…

- Regularization
  - Penalizes for complex models
- Ridge regression
  - L_2 penalized least-squares regression
  - Regularization parameter trades off model complexity with training error

©2017 Kevin Jamieson

Cross-Validation

Machine Learning – CSE546
Kevin Jamieson
University of Washington

October 4, 2016

©2017 Kevin Jamieson

How… How… How????????

- *How do we pick the regularization constant λ…*
- *How do we pick the number of basis functions…*

- We could use the test data, but…

©2017 Kevin Jamieson

How… How… How????????

- *How do we pick the regularization constant λ…*
- *How do we pick the number of basis functions…*

- We could use the test data, but…

- Never ever ever ever ever ever ever ever ever ever ever ever ever ever ever ever ever ever ever ever ever ever ever ever ever ever ever train on the test data

(LOO) Leave-one-out cross validation

- Consider a **validation set with 1 example**:
  - D – training data
  - $D\backslash j$ – training data with j th data point $(\mathbf{x}_j, \mathbf{y}_j)$ moved to validation set
- **Learn classifier $f_{D\backslash j}$ with $D\backslash j$ dataset**
- **Estimate true error** as squared error on predicting $\mathbf{y}_j$:
  - Unbiased estimate of $\text{error}_{\text{true}}(f_{D\backslash j})$!

(LOO) Leave-one-out cross validation

- Consider a **validation set with 1 example**:
  - D – training data
  - $D\backslash j$ – training data with j th data point ($\mathbf{x}_j$,$\mathbf{y}_j$) moved to validation set
- **Learn classifier $f_{D\backslash j}$ with $D\backslash j$ dataset**
- **Estimate true error** as squared error on predicting $\mathbf{y_j}$:
  - Unbiased estimate of $\text{error}_{\text{true}}(f_{D\backslash j})$!
- **LOO cross validation**: Average over all data points j:
  - **For each data point you leave out, learn a new classifier $f_{D\backslash j}$**
  - **Estimate error** as:

$$\text{error}_{LOO} = \frac{1}{n}\sum_{j=1}^{n}(y_j - f_{\mathcal{D}\backslash j}(x_j))^2$$

©2017 Kevin Jamieson

LOO cross validation is (almost) unbiased estimate of true error of h_D!

- When computing **LOOCV error, we only use *N-1* data points**
  - So it's not estimate of true error of learning with N data points
  - Usually pessimistic, though – learning with less data typically gives worse answer
- **LOO is almost unbiased**! **Use LOO error for model selection!!!**
  - **E.g., picking λ**

Computational cost of LOO

- Suppose you have 100,000 data points
- You implemented a great version of your learning algorithm
  - Learns in only 1 second
- Computing LOO will take about 1 day!!!
  -

©2017 Kevin Jamieson

Use *k*-fold cross validation

- Randomly **divide training data into *k* equal parts**
  - $D_1,\ldots,D_k$
- For each i
  - **Learn classifier $f_{D\backslash D_i}$ using data point not in D_i**
  - **Estimate error of $f_{D\backslash D_i}$ on validation set D_i**:

- $$\text{error}_{\mathcal{D}_i} = \frac{1}{|\mathcal{D}_i|} \sum_{(x_j, y_j) \in \mathcal{D}_i} (y_j - f_{\mathcal{D}\backslash\mathcal{D}_i}(x_j))^2$$

©2017 Kevin Jamieson

Use *k*-fold cross validation

- Randomly **divide training data into *k* equal parts**
  - $D_1,\ldots,D_k$
- For each *i*
  - **Learn classifier $f_{D\backslash D_i}$ using data point not in D_i**
  - **Estimate error of $f_{D\backslash D_i}$ on validation set D_i**:

$$\text{error}_{\mathcal{D}_i} = \frac{1}{|\mathcal{D}_i|} \sum_{(x_j,y_j)\in\mathcal{D}_i} (y_j - f_{\mathcal{D}\backslash\mathcal{D}_i}(x_j))^2$$

- ***k*-fold cross validation error is average** over data splits:

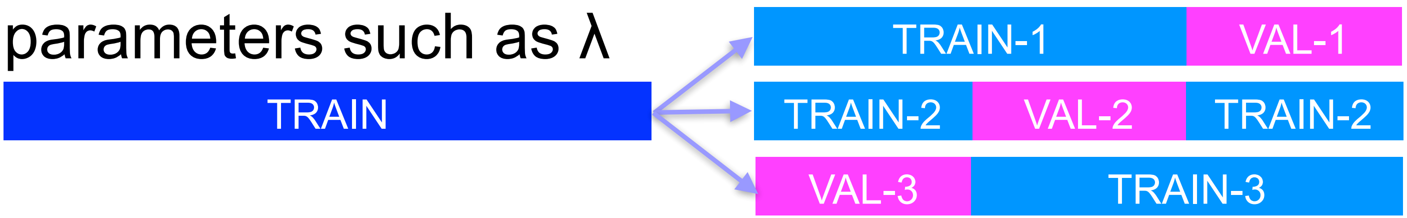
$$error_{k-fold} = \frac{1}{k}\sum_{i=1}^{k} error_{\mathcal{D}_i}$$

- *k*-fold cross validation properties:
  - **Much faster to compute** than LOO
  - **More (pessimistically) biased** – using much less data, only *n(k-1)/k*
  - Usually, **k = 10**

Recap

- Given a dataset, begin by splitting into

- **Model selection**: Use k-fold cross-validation on TRAIN to train predictor and choose magic parameters such as λ



- **Model assessment**: Use TEST to assess the accuracy of the model you output
  - Never ever ever ever ever train or choose parameters based on the test data

©2017 Kevin Jamieson

Example

- Given 10,000-dimensional data and n examples, we pick a subset of 50 dimensions that have the highest correlation with labels in the training set:

50 indices j that have largest $$\frac{|\sum_{i=1}^n x_{i,j} y_i|}{\sqrt{\sum_{i=1}^n x_{i,j}^2}}$$

- After picking our 50 features, we then use CV to train ridge regression with regularization λ
- What's wrong with this procedure?

©2017 Kevin Jamieson

Recap

- Learning is…
  - Collect some data
    - E.g., housing info and sale price
  - Randomly split dataset into TRAIN, VAL, and TEST
    - E.g., 80%, 10%, and 10%, respectively
  - Choose a hypothesis class or model
    - E.g., linear with non-linear transformations
  - Choose a loss function
    - E.g., least squares with ridge regression penalty on TRAIN
  - Choose an optimization procedure
    - E.g., set derivative to zero to obtain estimator, cross-validation on VAL to pick num. features and amount of regularization
  - Justifying the accuracy of the estimate
    - E.g., report TEST error with Bootstrap confidence interval

©2017 Kevin Jamieson