Cross-Validation

Test error : an unbiased estimate of TRUE ERROR

BUT ONLY IF YOU DON'T TOUCH TEST DATA AT ALL in training

So, use your training data to "simulate" test data/error in model selection



How… How… How???????

- How do we pick the number of basis functions…

- We could use the test data, but…

How… How… How???????

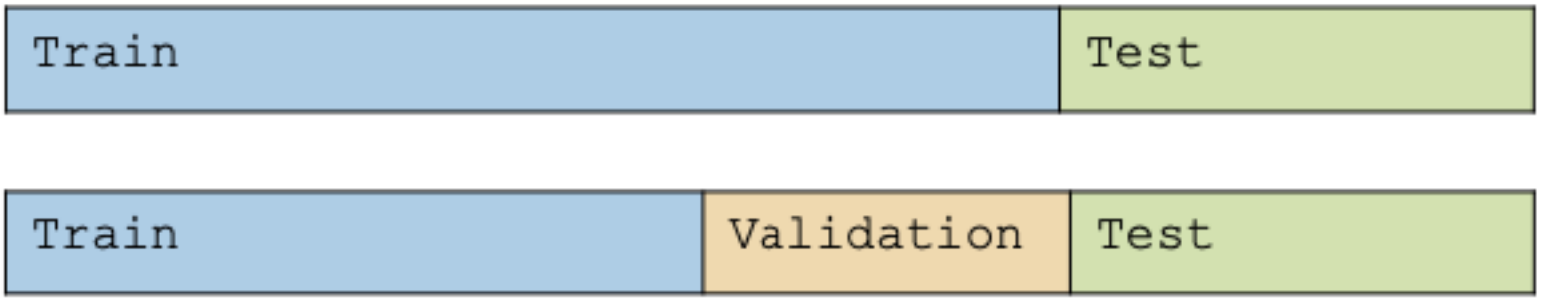
- How do we pick the number of basis functions…

- 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 ever ever ever train, or even look at the performance on the test data if you're making any decisions based on that performance.

Validation Set

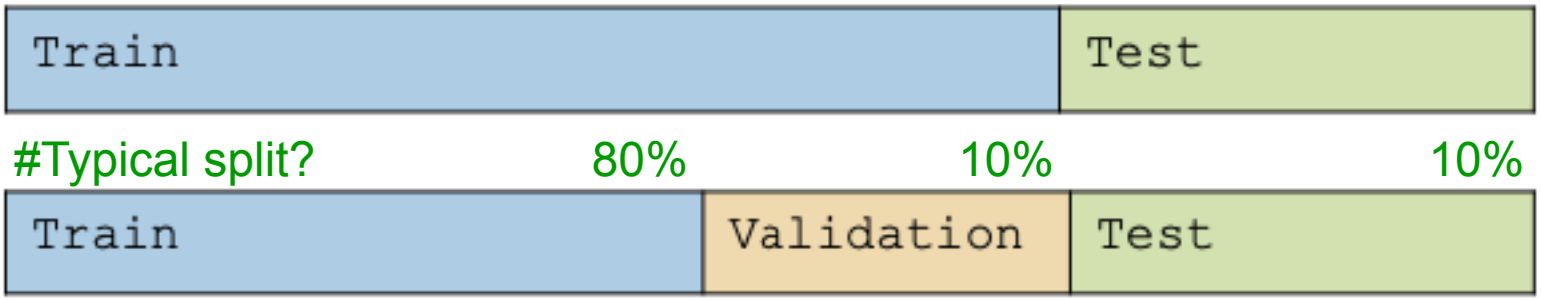
- Simplest approach is to add another dataset partition called the **validation set**



- Each set has its own role
  - **Train:** Used to train a model of each model complexity
  - **Validation:** Used as a hold-out to evaluate each model. Generally choose model complexity with lowest validation error.
  - **Test:** Once the model is chosen, can get an estimate of future error by test. This would be what you report. **Only do this once!**

Validation Set

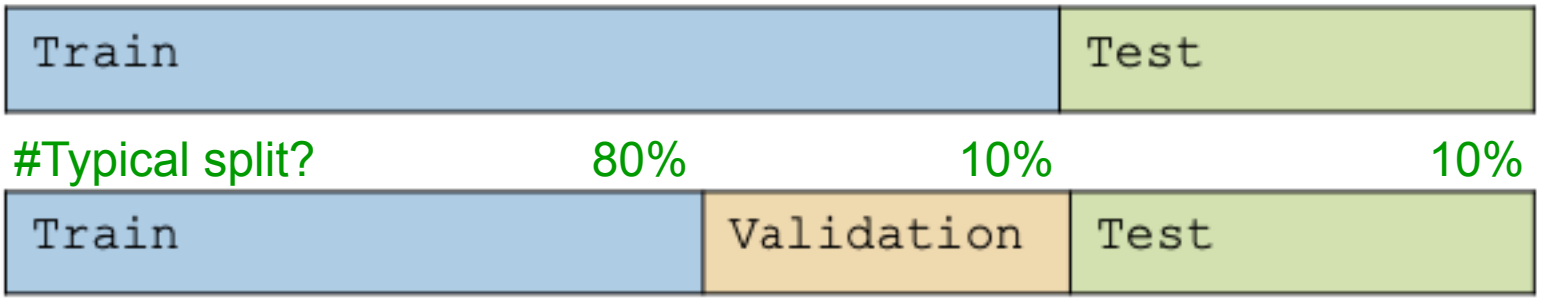
- Simplest approach is to add another dataset partition called the **validation set**



- Each set has its own role
  - **Train:** Used to train a model of each model complexity
  - **Validation:** Used as a hold-out to evaluate each model. Generally choose model complexity with lowest validation error.
  - **Test:** Once the model is chosen, can get an estimate of future error by test. This would be what you report. **Only do this once!**

Validation Set

- Simplest approach is to add another dataset partition called the **validation set**



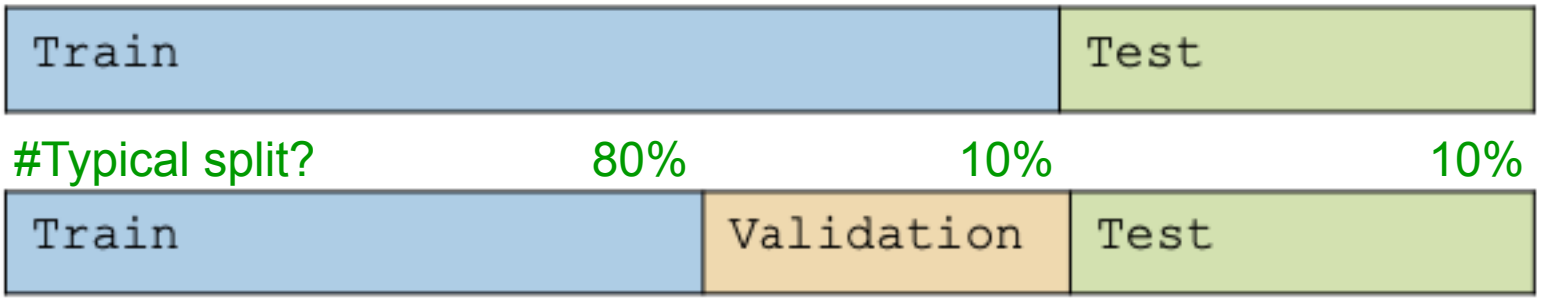
- Each set has its own role

Fit model params like w

  - **Train:** Used to train a model of each model complexity
  - **Validation:** Used as a hold-out to evaluate each model. Generally choose model complexity with lowest validation error.
  - **Test:** Once the model is chosen, can get an estimate of future error by test. This would be what you report. **Only do this once!**

Validation Set

- Simplest approach is to add another dataset partition called the **validation set**



- Each set has its own role

  # Fit model params like w

  - **Train:** Used to train a model of each model complexity
  - **Validation:** Used as a hold-out to evaluate each model. Generally choose model complexity with lowest validation error. # Choose *hyper*parameters, like polynomial degree p
  - **Test:** Once the model is chosen, can get an estimate of future error by test. This would be what you report. **Only do this once!**

(LOO) Leave-one-out cross validation

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

(LOO) Leave-one-out cross validation

- **Consider a validation set with 1 example:**
  - **D – training data** # n samples
  - **D\j – training data with j th data point (x_j ,y_j) moved to validation set**
- **Learn model $f_{D\setminus j}$ with $D\setminus j$ dataset**
- **Estimate true error** as squared error on predicting $\mathbf{y_j}$:
  - **Unbiased estimate of $\text{error}_{\text{true}}(f_{D\setminus j})$!**

(LOO) Leave-one-out cross validation

- **Consider a validation set with 1 example:**
  - **D – training data** # n samples
  - **D\j – training data with j th data point (x_j ,y_j) moved to validation set**
- **Learn model $f_{D\setminus j}$ with $D\setminus j$ dataset** # n-1 samples
- **Estimate true error** as squared error on predicting $\mathbf{y_j}$:
  - **Unbiased estimate of $\text{error}_{\text{true}}(f_{D\setminus j})$!**

(LOO) Leave-one-out cross validation

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

- **LOO cross validation**: Average over all data points *j*:
  - **For each data point you leave out, learn a new model $f_{D\setminus j}$**
  - **Estimate error** as:

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

LOO cross validation is (almost) unbiased estimate!

- **When computing LOOCV error, we only use *N-1* data points**
  - **So it's not quite an 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**! Why not **use LOO error for model selection!**
  - **E.g., picking degree**



LOO cross validation is (almost) unbiased estimate!

- **When computing LOOCV error, we only use *N-1* data points**
  - **So it's not quite an 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**! Why not **use LOO error for model selection!**
  - **E.g., picking degree**



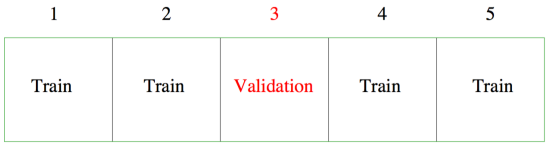
You just fit n models!!

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!!!
  -

Use *k*-fold cross validation

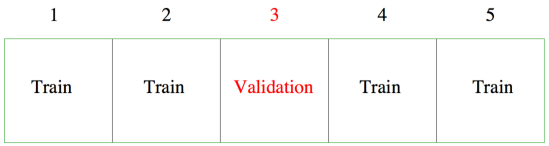
- Randomly divide training data into *k* equal parts
  - $D_1,\ldots,D_k$
- For each *i*
  - Learn model $f_{D\backslash Di}$ using data point not in D_i
  - Estimate error of $f_{D\backslash Di}$ on validation set D_i:



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

Use *k*-fold cross validation

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



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

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

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

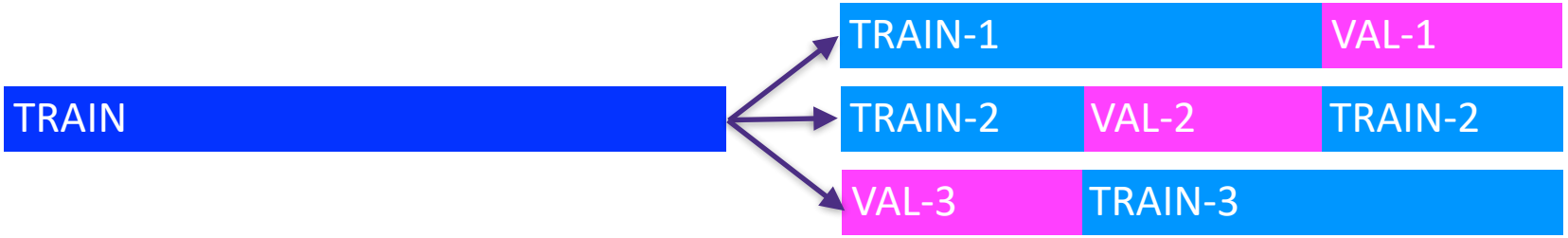
- k-fold cross validation properties:
  - Much faster to compute than LOO
  - More (pessimistically) biased – using much less data, only *n(k-1)/k*
  - Usually, k = 10

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 degree



- 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**

Example 1

- **You wish to predict the stock price of zoom.us given historical stock price data**
- **You use all daily stock price up to Jan 1, 2020 as TRAIN and Jan 2, 2020 - April 13, 2020 as TEST**
- **What's "wrong" with this procedure?**

Example 1

- **You wish to predict the stock price of zoom.us given historical stock price data**
- **You use all daily stock price up to Jan 1, 2020 as TRAIN and Jan 2, 2020 - April 13, 2020 as TEST**
- **What's "wrong" with this procedure?**

  - ML assumes data is i.i.d.; is this?

Example 1

- **You wish to predict the stock price of zoom.us given historical stock price data**
- **You use all daily stock price up to Jan 1, 2020 as TRAIN and Jan 2, 2020 - April 13, 2020 as TEST**
- **What's "wrong" with this procedure?**

  - ML assumes data is i.i.d.; is this?
    - Nope! Future data is o.o.d. (out of distribution) from the training data

Example 1

- **You wish to predict the stock price of zoom.us given historical stock price data**
- **You use all daily stock price up to Jan 1, 2020 as TRAIN and Jan 2, 2020 - April 13, 2020 as TEST**
- **What's "wrong" with this procedure?**

  - ML assumes data is i.i.d.; is this?
    - Nope! Future data is o.o.d. (out of distribution) from the training data

  - Why is "wrong" in square quotes?

Example 1

- **You wish to predict the stock price of zoom.us given historical stock price data**
- **You use all daily stock price up to Jan 1, 2020 as TRAIN and Jan 2, 2020 - April 13, 2020 as TEST**
- **What's "wrong" with this procedure?**

  - ML assumes data is i.i.d.; is this?
    - Nope! Future data is o.o.d. (out of distribution) from the training data

  - Why is "wrong" in square quotes?
    - You might still want to do this. It's economically valuable, if you could.

Example 2

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

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 break data into train and test dataset.**
- **We train linear regression on these selected features on the training set. We compute the test error and report it**
- **What's wrong with this procedure?**

Example 2

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

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 break data into train and test dataset.**
- **We train linear regression on these selected features on the training set. We compute the test error and report it**
- **What's wrong with this procedure?**
  - Need to hold out the test data first, otherwise you're contaminating your test data.

Example 2

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

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 break data into train and test dataset.**
- **We train linear regression on these selected features on the training set. We compute the test error and report it**
- **What's wrong with this procedure?**
  - Need to hold out the test data first, otherwise you're contaminating your test data.
    - This happened in my first ML project. Thought the accuracy was 90%. When I fixed, it went down to 60%.

Example 2

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

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 break data into train and test dataset.**
- **We train linear regression on these selected features on the training set. We compute the test error and report it**
- **What's wrong with this procedure?**
  - Need to hold out the test data first, otherwise you're contaminating your test data.
    - This happened in my first ML project. Thought the accuracy was 90%. When I fixed, it went down to 60%.
  - Top 50 features might be redundant with each other.

Recap

- Learning is…
  - Collect some data
    - E.g., potato delivery features and delivery time
  - 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 on TRAIN
  - Choose an optimization procedure
    - E.g., set derivative to zero to obtain estimator, cross-validation on VAL to pick num. features
- Justifying the accuracy of the estimate
    - E.g., report TEST error

Bias-Variance Tradeoff



Optimal Prediction

Goal: Predict $Y \in \mathbb{R}$ **given** $X \in \mathbb{R}^d$ **if** $(X, Y) \sim P_{XY}$

Find function η that minimizes

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

(Hint: for any x, $\eta(x) = c_x$ where c_x minimizes $\mathbb{E}_{Y|X}[(Y - c_x)^2 | X = x]$)

Optimal Prediction

True distribution, don't actually have access to it

Goal: Predict $Y \in \mathbb{R}$ **given** $X \in \mathbb{R}^d$ **if** $(X, Y) \sim P_{XY}$

Find function η that minimizes

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

(Hint: for any x, $\eta(x) = c_x$ where c_x minimizes $\mathbb{E}_{Y|X}[(Y - c_x)^2 | X = x]$)

Optimal Prediction

True distribution, don't actually have access to it

Goal: Predict $Y \in \mathbb{R}$ **given** $X \in \mathbb{R}^d$ **if** $(X, Y) \sim P_{XY}$

Find function η that minimizes

Tower rule: E[Y] = E$_X$[E[Y|X]]

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

(Hint: for any x, $\eta(x) = c_x$ where c_x minimizes $\mathbb{E}_{Y|X}[(Y - c_x)^2 | X = x]$)

Optimal Prediction

True distribution, don't actually have access to it

Goal: Predict $Y \in \mathbb{R}$ **given** $X \in \mathbb{R}^d$ **if** $(X, Y) \sim P_{XY}$

Find function η that minimizes

Tower rule: E[Y] = E$_X$[E[Y|X]]

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

(Hint: for any x, $\eta(x) = c_x$ where c_x minimizes $\mathbb{E}_{Y|X}[(Y - c_x)^2 | X = x]$)

$$\frac{d}{dc}\mathbb{E}_{Y|X}[(Y - c)^2 \,|\, X = x] = 0$$

Optimal Prediction

True distribution, don't actually have access to it

Goal: Predict $Y \in \mathbb{R}$ **given** $X \in \mathbb{R}^d$ **if** $(X, Y) \sim P_{XY}$

Find function η that minimizes

Tower rule: E[Y] = E$_X$[E[Y|X]]

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

(Hint: for any x, $\eta(x) = c_x$ where c_x minimizes $\mathbb{E}_{Y|X}[(Y - c_x)^2 | X = x]$)

$$\frac{d}{dc}\mathbb{E}_{Y|X}[(Y - c)^2 \,|\, X = x] = 0$$

$$= \mathbb{E}_{Y|X}[2(Y - c)(-1) \,|\, X = x] = 0$$

Optimal Prediction

True distribution, don't actually have access to it

Goal: Predict $Y \in \mathbb{R}$ **given** $X \in \mathbb{R}^d$ **if** $(X, Y) \sim P_{XY}$

Find function η that minimizes

Tower rule: E[Y] = E$_X$[E[Y|X]]

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

(Hint: for any x, $\eta(x) = c_x$ where c_x minimizes $\mathbb{E}_{Y|X}[(Y - c_x)^2 | X = x]$)

$$\frac{d}{dc}\mathbb{E}_{Y|X}[(Y - c)^2 \,|\, X = x] = 0$$

$$= \mathbb{E}_{Y|X}[2(Y - c)(-1) \,|\, X = x] = 0$$

$$= -2[\mathbb{E}_{Y|X}[Y \,|\, X = x] - c] = 0$$

Optimal Prediction

True distribution, don't actually have access to it

Goal: Predict $Y \in \mathbb{R}$ **given** $X \in \mathbb{R}^d$ **if** $(X, Y) \sim P_{XY}$

Find function η that minimizes

Tower rule: E[Y] = E$_X$[E[Y|X]]

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

(Hint: for any x, $\eta(x) = c_x$ where c_x minimizes $\mathbb{E}_{Y|X}[(Y - c_x)^2 | X = x]$)

$$\frac{d}{dc}\mathbb{E}_{Y|X}[(Y - c)^2 \,|\, X = x] = 0$$

$$= \mathbb{E}_{Y|X}[2(Y - c)(-1) \,|\, X = x] = 0$$

$$= -2[\mathbb{E}_{Y|X}[Y \,|\, X = x] - c] = 0$$

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

Optimal Prediction

True distribution, don't actually have access to it

Goal: Predict $Y \in \mathbb{R}$ **given** $X \in \mathbb{R}^d$ **if** $(X, Y) \sim P_{XY}$

Find function η that minimizes

Tower rule: E[Y] = E$_X$[E[Y|X]]

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

(Hint: for any x, $\eta(x) = c_x$ where c_x minimizes $\mathbb{E}_{Y|X}[(Y - c_x)^2 | X = x]$)

$$\frac{d}{dc}\mathbb{E}_{Y|X}[(Y - c)^2 \,|\, X = x] = 0$$

$$= \mathbb{E}_{Y|X}[2(Y - c)(-1) \,|\, X = x] = 0$$

$$= -2[\mathbb{E}_{Y|X}[Y \,|\, X = x] - c] = 0$$

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

Optimal Prediction

Goal: Predict $Y \in \mathbb{R}$ given $X \in \mathbb{R}^d$ if $(X, Y) \sim P_{XY}$

Find function η that minimizes

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

(Hint: for any x, $\eta(x) = c_x$ where c_x minimizes $\mathbb{E}_{Y|X}[(Y - c_x)^2 | X = x]$)

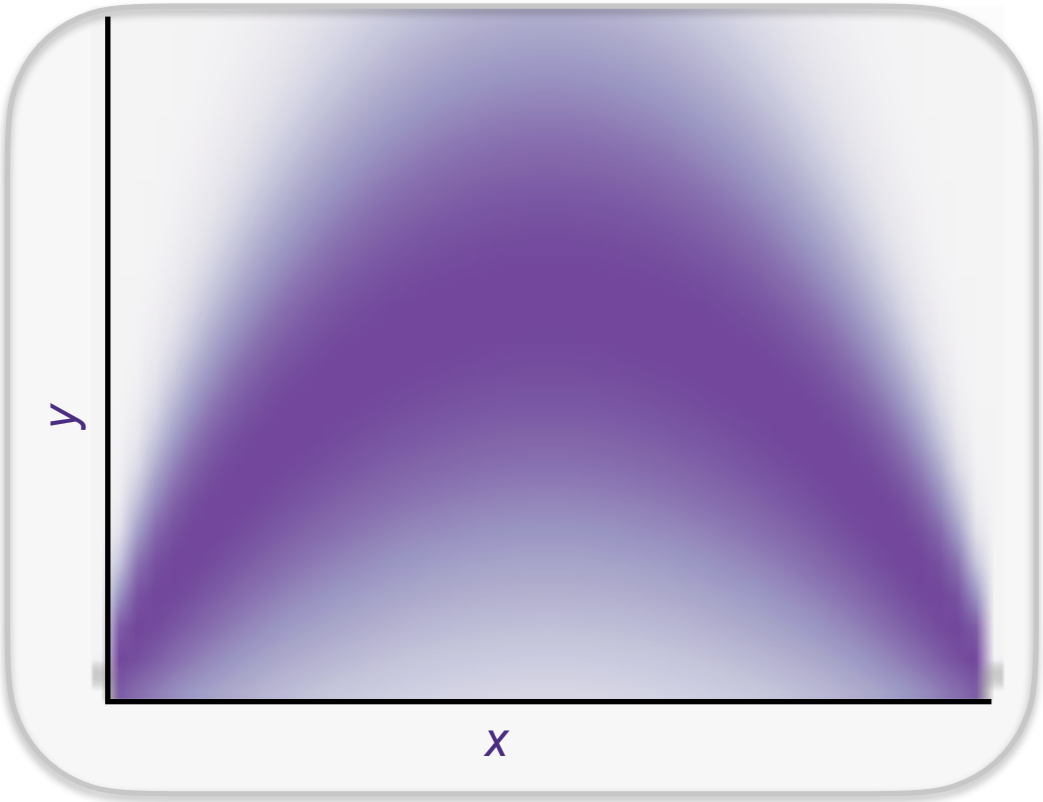
$$\begin{aligned} 0 &= \frac{d}{dc_x}\mathbb{E}_{Y|X}[(Y - c_x)^2 | X = x] \\ &= \mathbb{E}_{Y|X}[\frac{d}{dc_x}(Y - c_x)^2 | X = x] \\ &= \mathbb{E}_{Y|X}[-2(Y - c_x) | X = x] = -2\mathbb{E}_{Y|X}[Y | X = x] + 2c_x \end{aligned}$$

Squared Error Optimal Predictor: $\eta(x) = \mathbb{E}_{Y|X}[Y | X = x]$

Statistical Learning

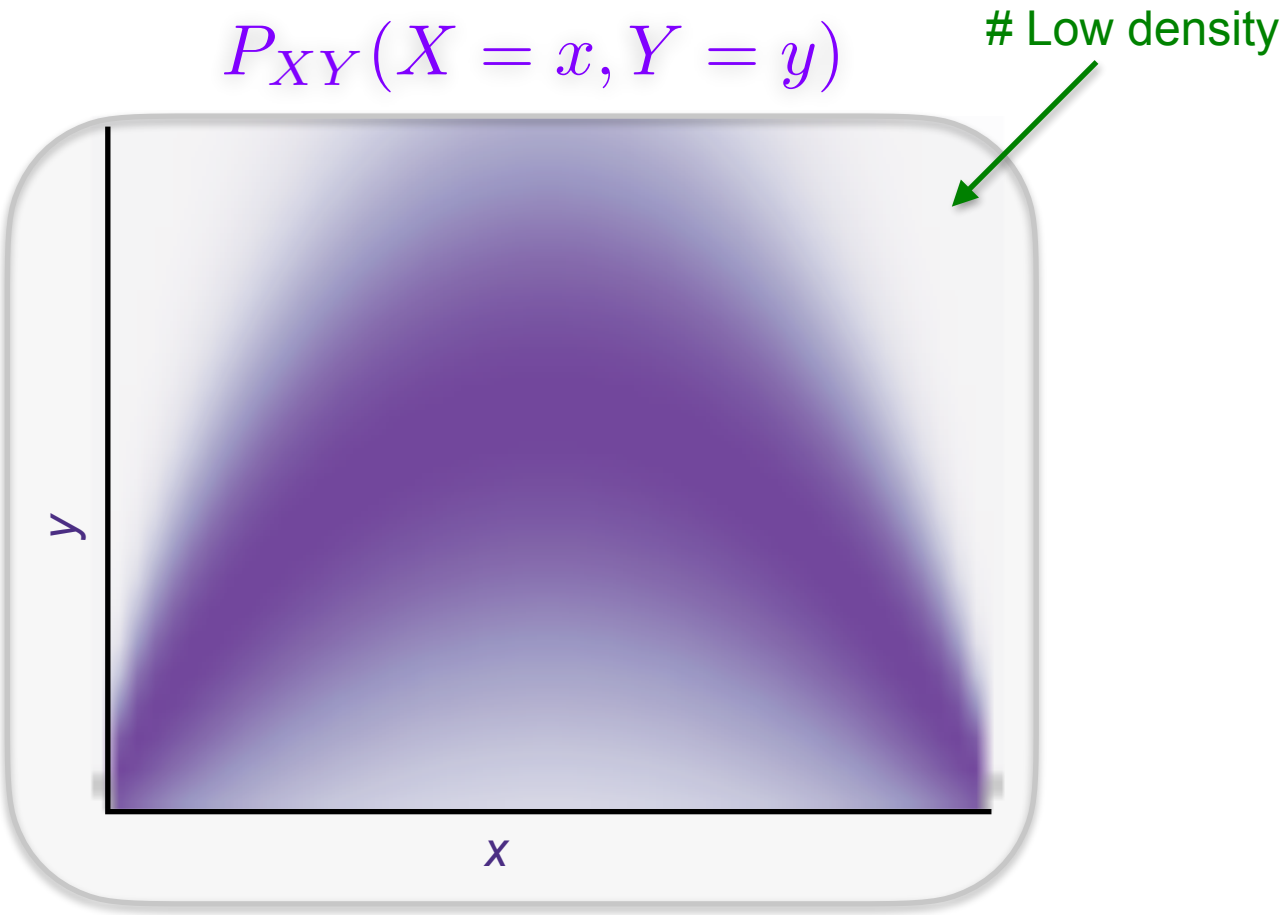
$$\mathbb{E}_{XY}[(Y - \eta(X))^2]$$

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



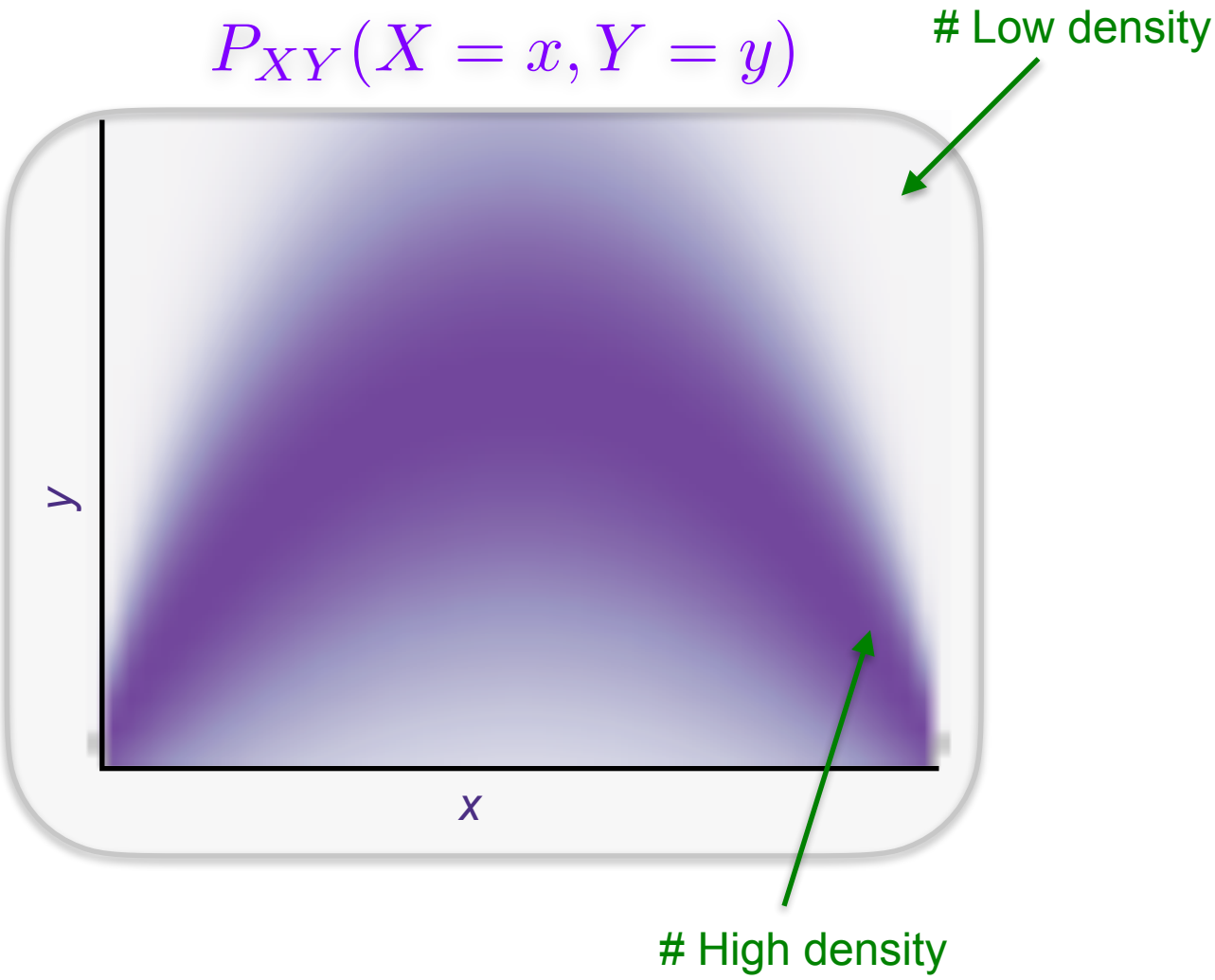
Statistical Learning

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



Statistical Learning

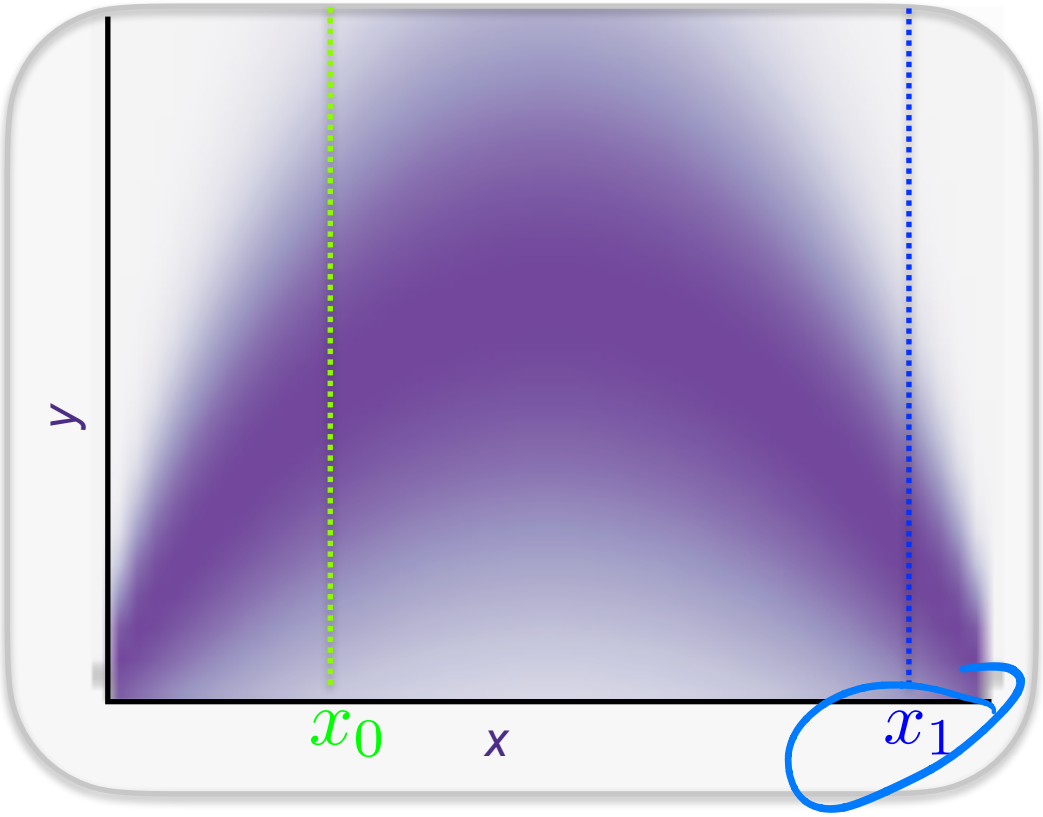
$$\mathbb{E}_{XY}[(Y - \eta(X))^2]$$



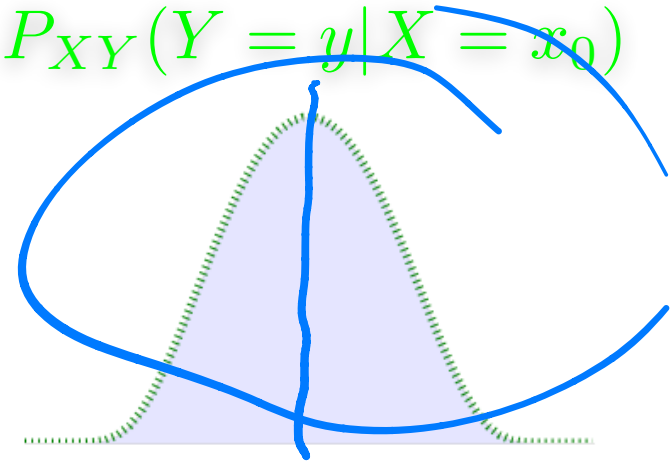
Statistical Learning

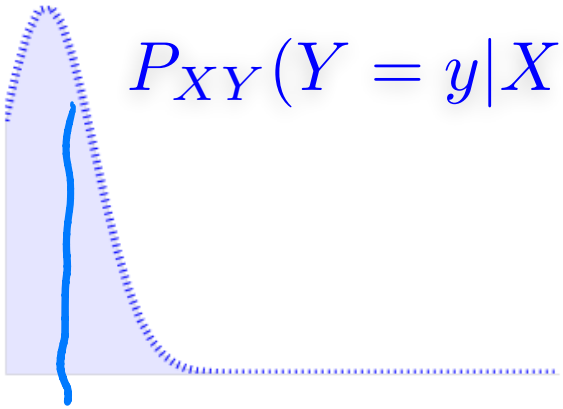
$$\mathbb{E}_{XY}[(Y - \eta(X))^2]$$

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



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

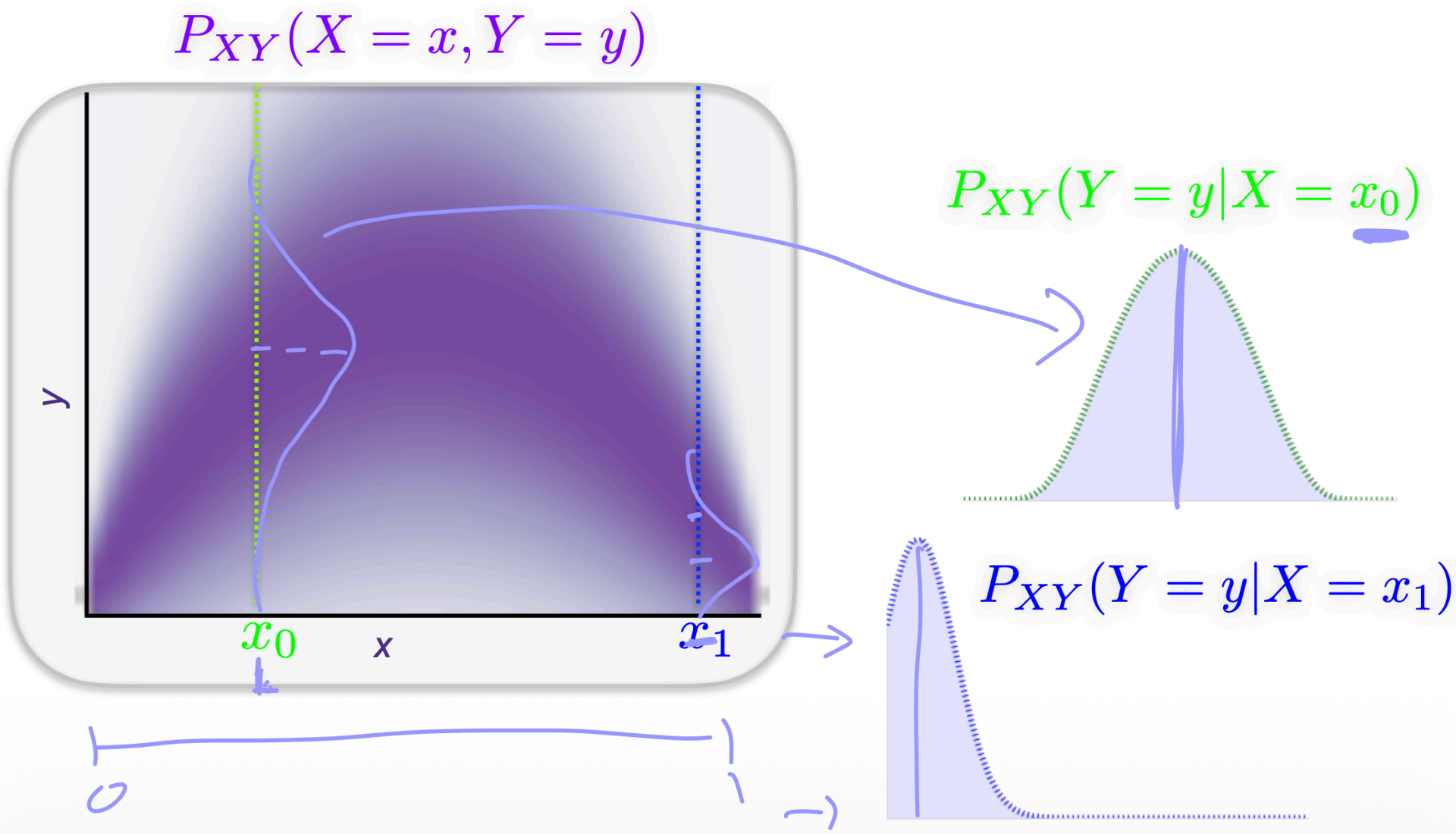




$P_{XY}(Y = y|X = x_1)$

Statistical Learning

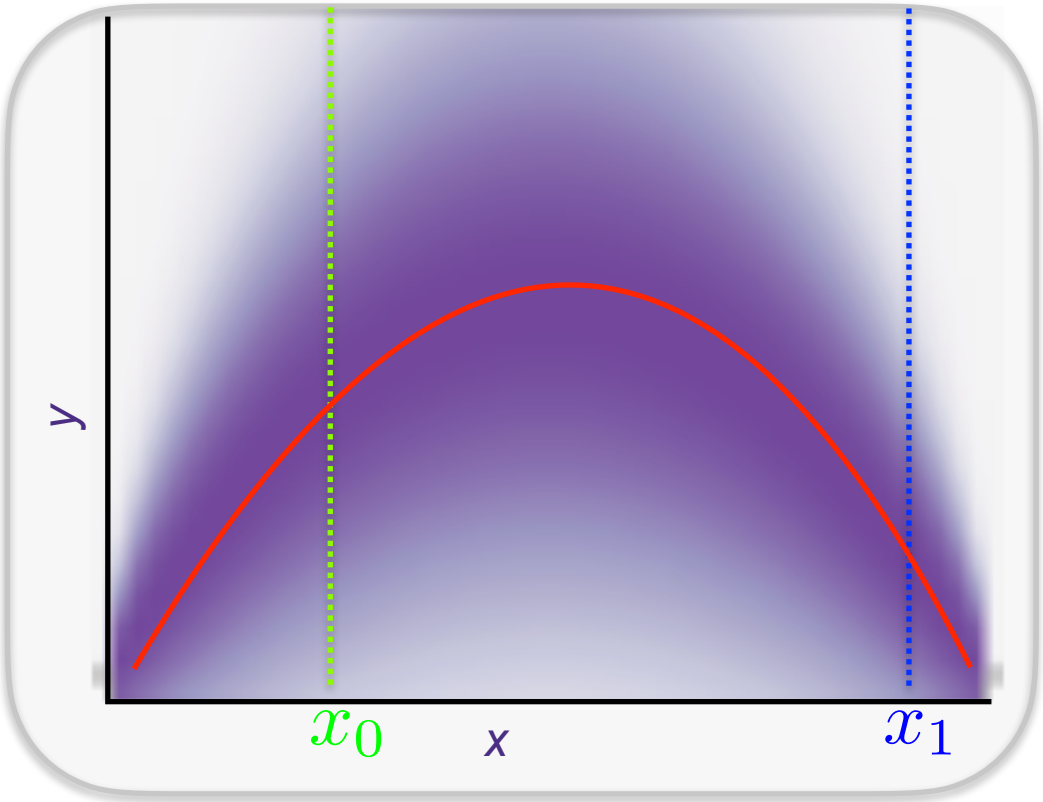
$$\mathbb{E}_{XY}[(Y - \eta(X))^2]$$



Statistical Learning

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

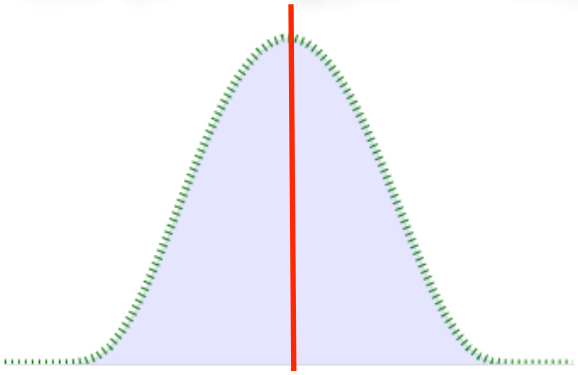
$P_{XY}(X = x, Y = y)$

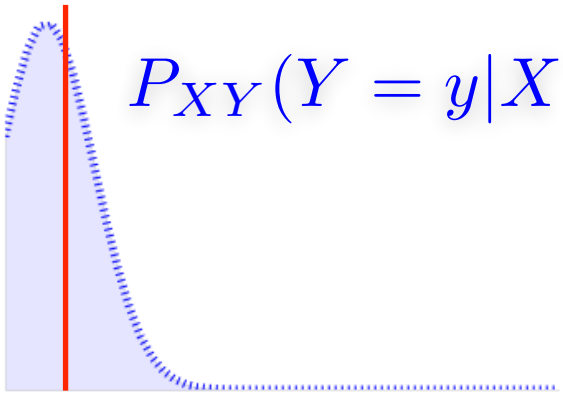


Ideally, we want to find:

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

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

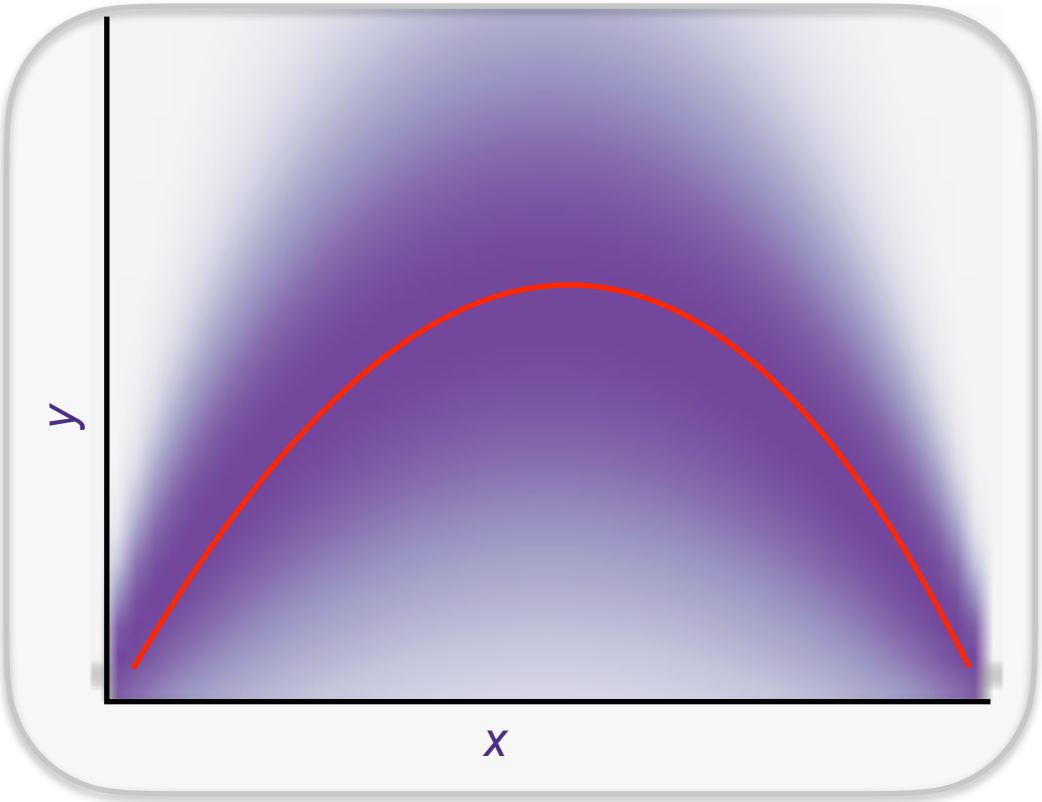




$P_{XY}(Y = y|X = x_1)$

Statistical Learning

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

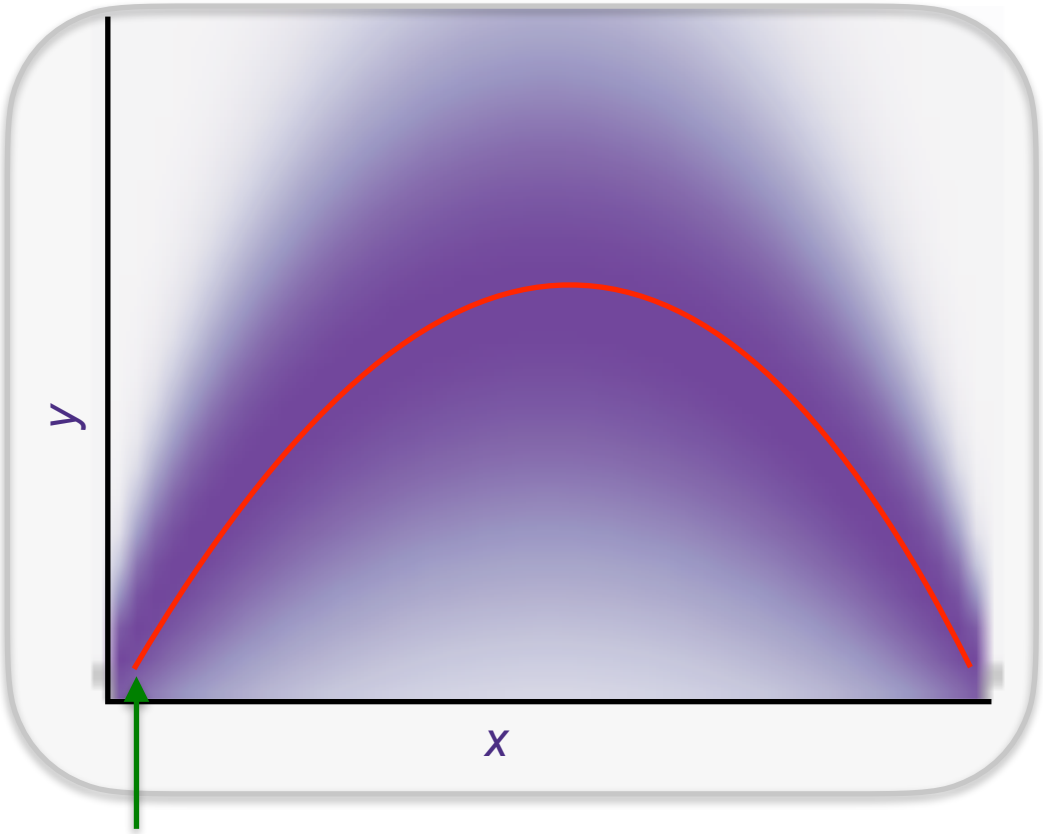


Ideally, we want to find:

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

Statistical Learning

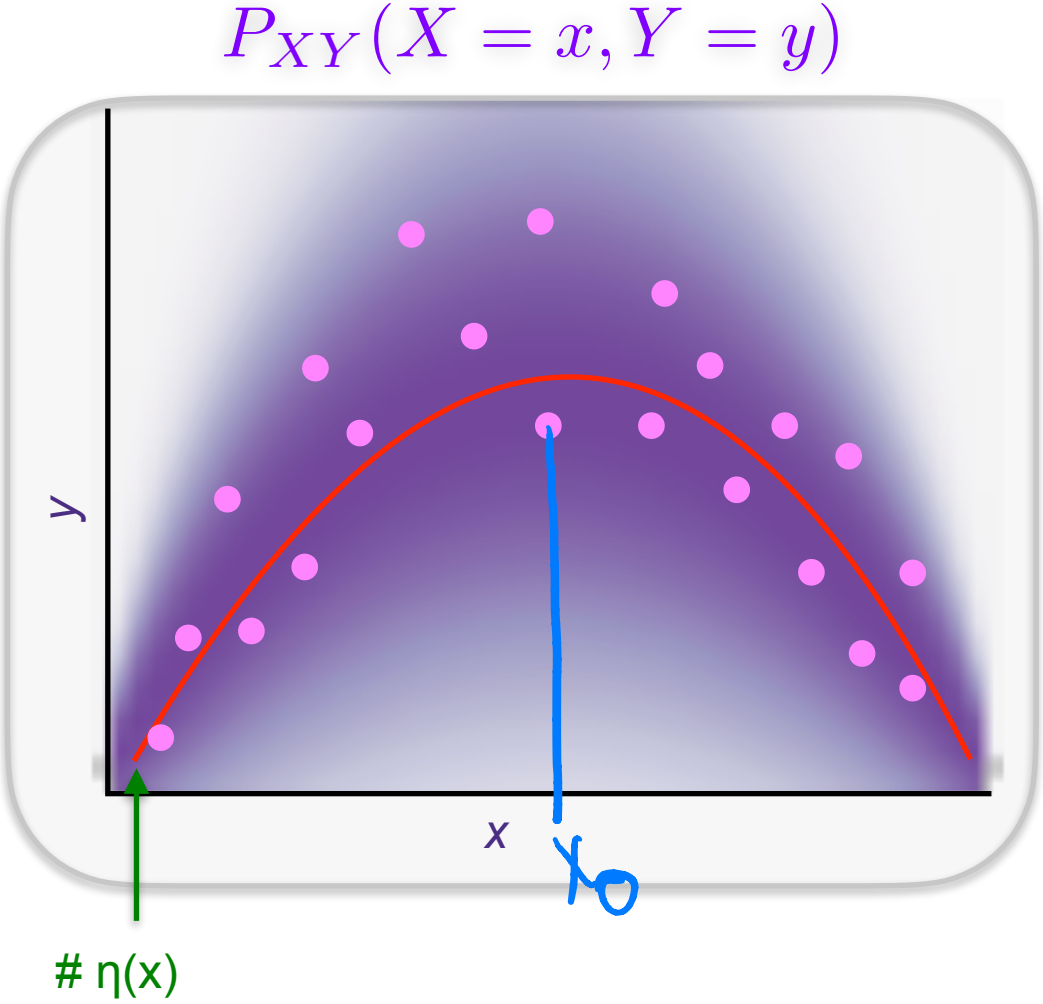
$P_{XY}(X = x, Y = y)$



Ideally, we want to find:

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

Statistical Learning



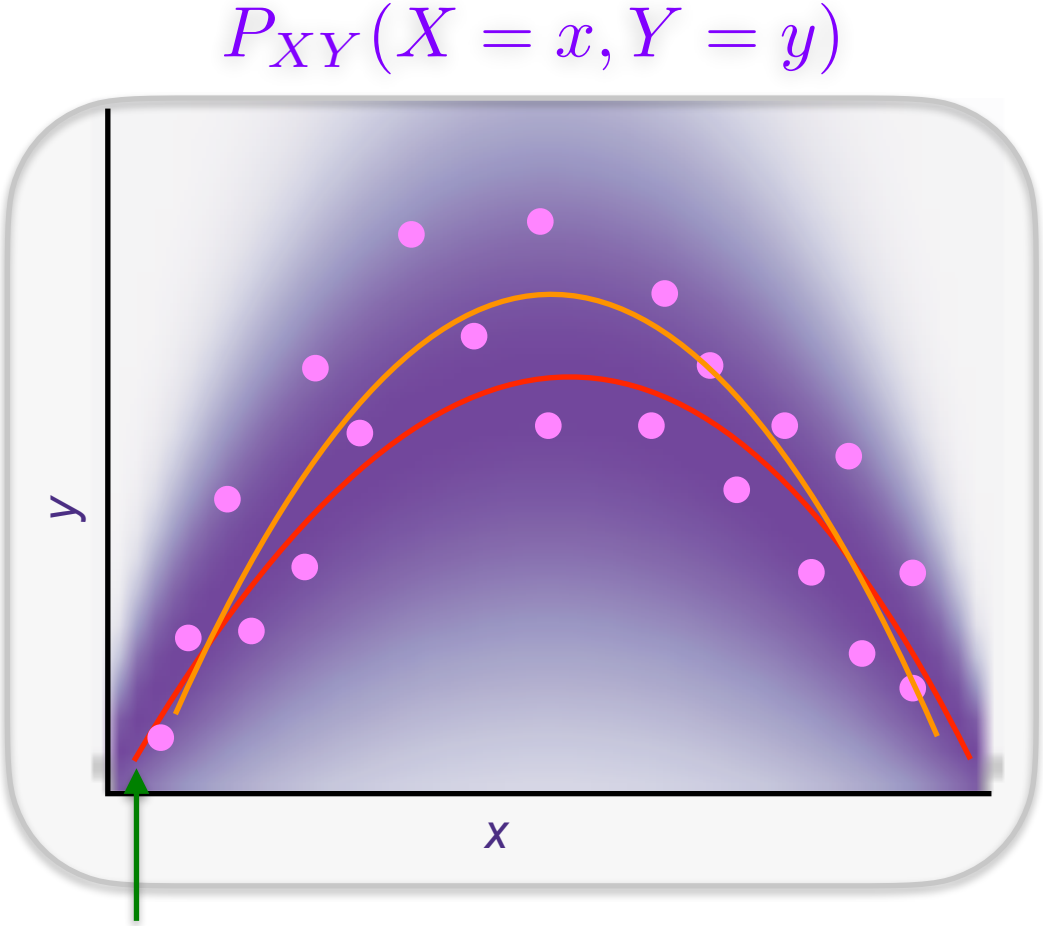
Ideally, we want to find:

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

But we only have samples:

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

Statistical Learning



Ideally, we want to find:

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

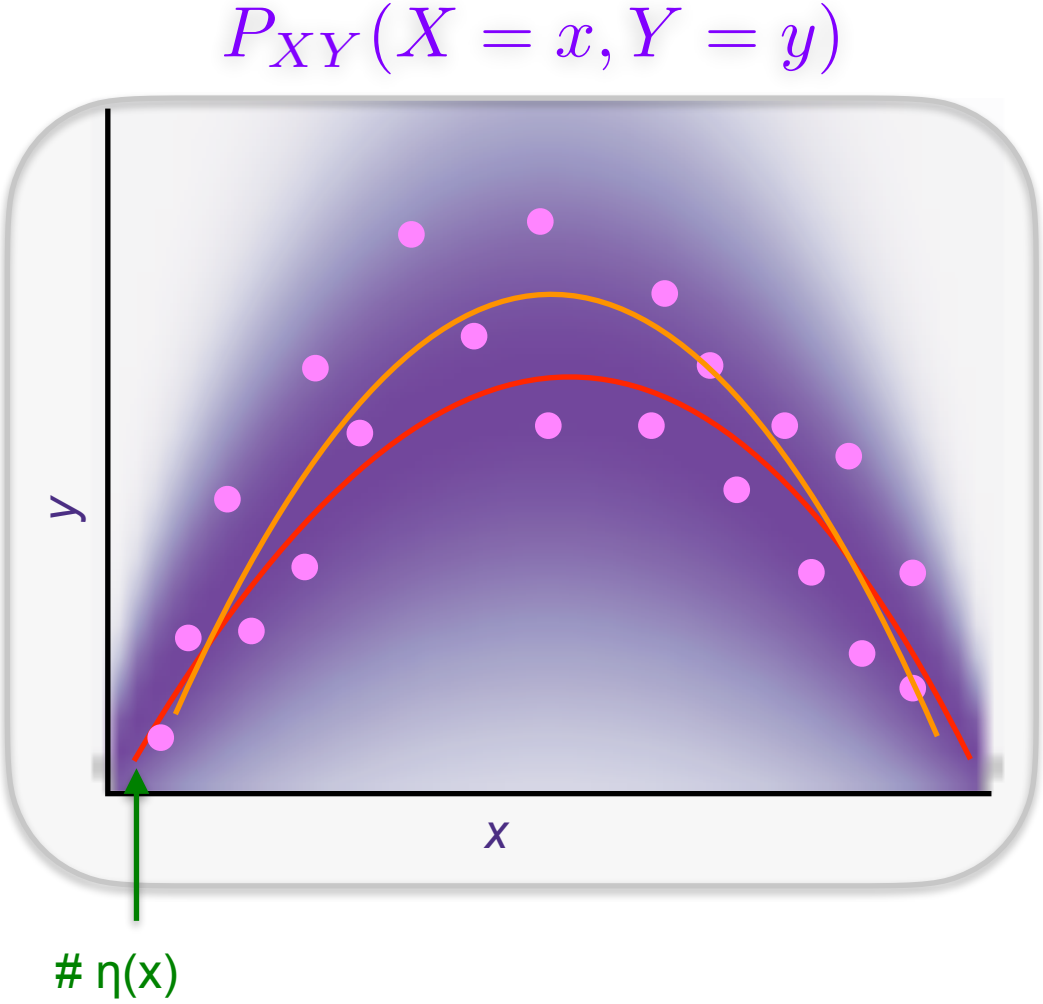
But we only have samples:

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

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:

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

But we only have samples:

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

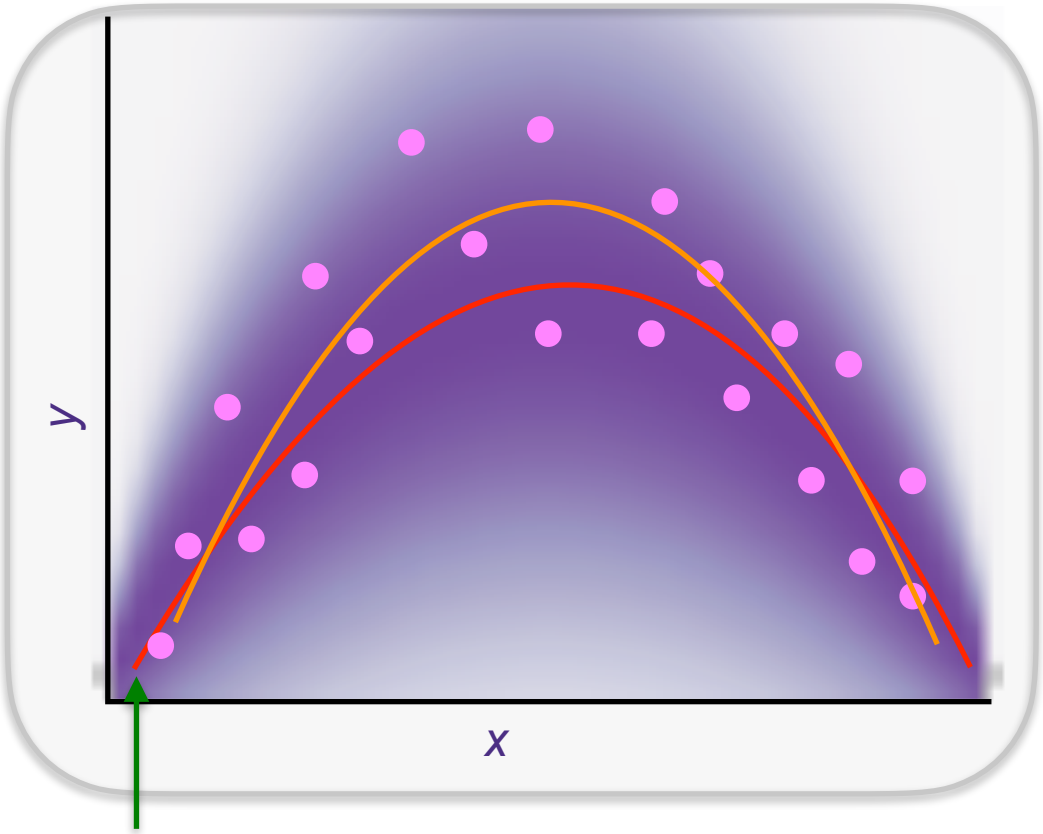
and are restricted to a function class (e.g., linear) so we compute:

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

$\hat{f} \neq \eta(x)$

Statistical Learning

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



η(x)

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 $i = 1, \ldots, n$

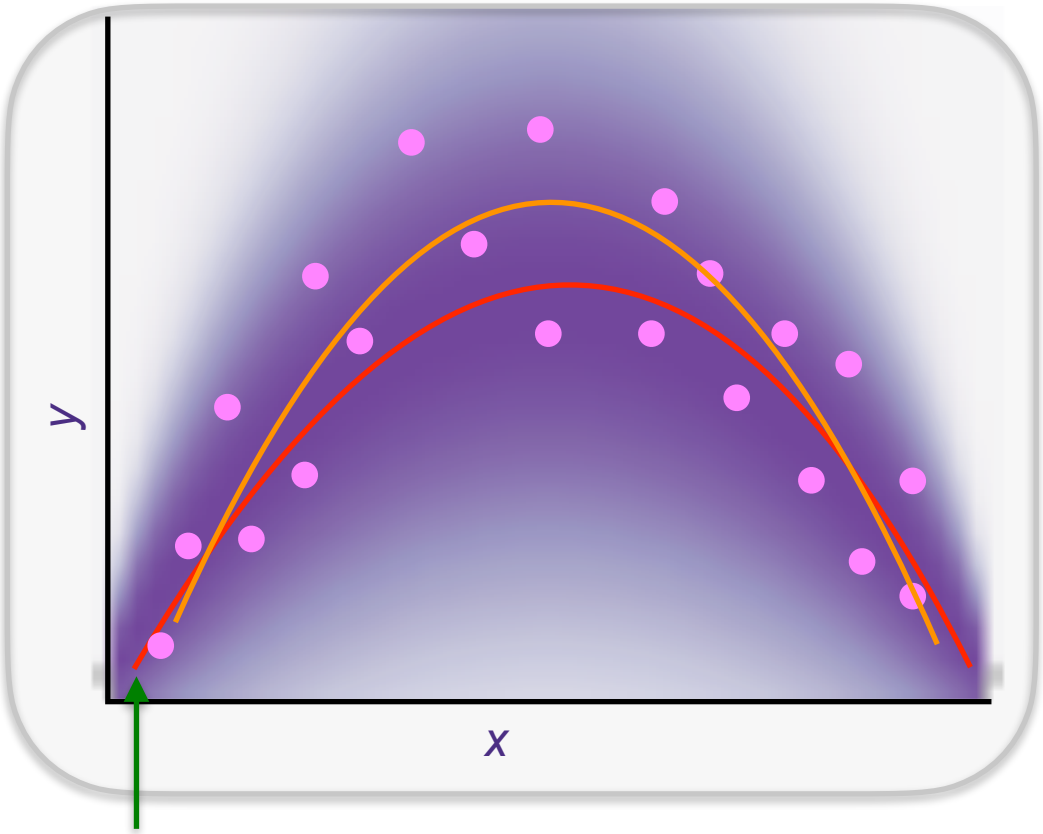
and are restricted to a function class (e.g., linear) so we compute:

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

$\hat{f} \neq \eta(x)$ Why?

Statistical Learning

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



η(x)

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 $i = 1, \ldots, n$

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

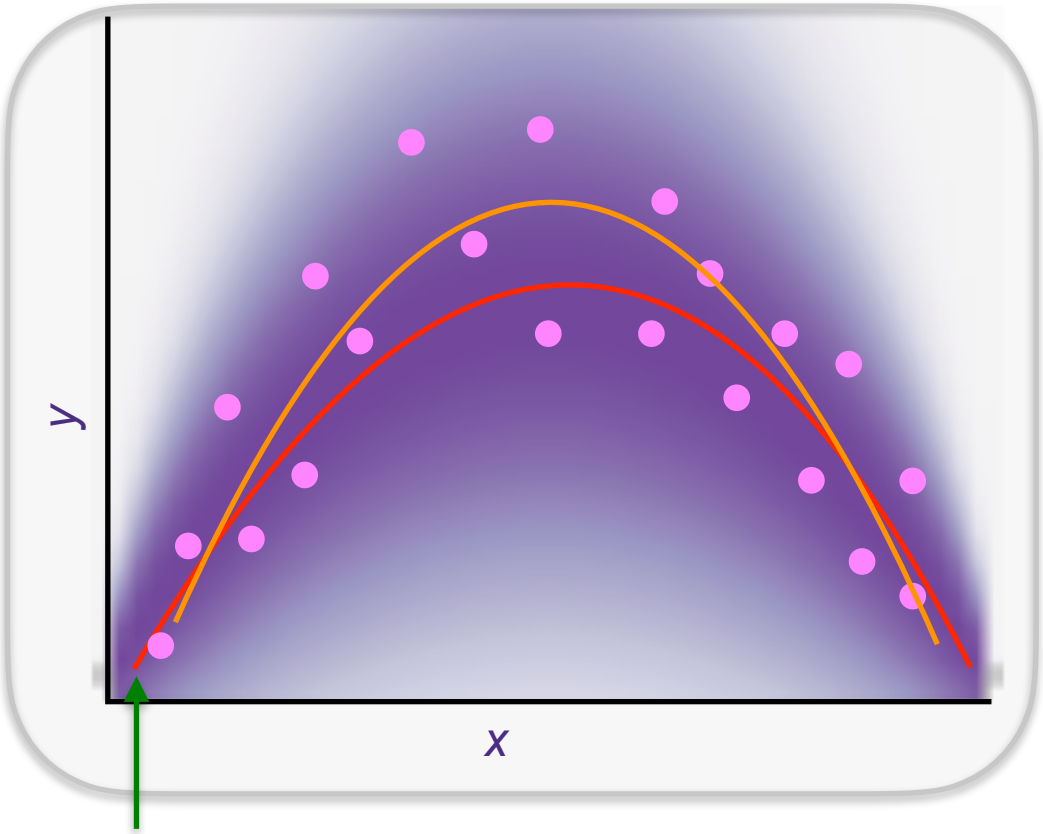
$$\widehat{f} = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2$$

$\hat{f} \neq \eta(x)$ Why?

n is limited

Statistical Learning

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



η(x)

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 $i = 1, \ldots, n$

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

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

$\hat{f} \neq \eta(x)$ Why?

n is limited

could have chosen wrong model class *F*

Empirical Reconstruction Error (ERE)

Empirical Reconstruction Error (ERE)

What is the $\mathbb{E}[\hat{f}]$?

Empirical Reconstruction Error (ERE)

What is the $\mathbb{E}[\hat{f}]$?

$$\mathbb{E}_{XY}[\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2]$$

Empirical Reconstruction Error (ERE)

What is the $\mathbb{E}[\hat{f}]$?

$$\mathbb{E}_{XY}[\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2]$$

By linearity of expectation

Empirical Reconstruction Error (ERE)

What is the $\mathbb{E}[\hat{f}]$?

$$\mathbb{E}_{XY}[\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2]$$

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(y_i - f(x_i))^2]$$ # By linearity of expectation

Empirical Reconstruction Error (ERE)

What is the $\mathbb{E}[\hat{f}]$?

$$\mathbb{E}_{XY}[\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2]$$

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(y_i - f(x_i))^2]$$ # By linearity of expectation

by IID: $\mathbb{E}[x_i]$ for any $i = \mathbb{E}[x] \forall x$

Empirical Reconstruction Error (ERE)

What is the $\mathbb{E}[\hat{f}]$?

$$\mathbb{E}_{XY}[\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2]$$

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(y_i - f(x_i))^2]$$ # By linearity of expectation

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(y - f(x))^2]$$ by IID: $\mathbb{E}[x_i]$ for any $i = \mathbb{E}[x] \forall x$

Empirical Reconstruction Error (ERE)

What is the $\mathbb{E}[\hat{f}]$?

$$\mathbb{E}_{XY}[\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2]$$

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(y_i - f(x_i))^2]$$ # By linearity of expectation

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(y - f(x))^2]$$ by IID: $\mathbb{E}[x_i]$ for any $i = \mathbb{E}[x] \forall x$

Empirical Reconstruction Error (ERE)

What is the $\mathbb{E}[\hat{f}]$?

$$\mathbb{E}_{XY}[\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2]$$

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(y_i - f(x_i))^2]$$ # By linearity of expectation

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(y - f(x))^2]$$ by IID: $\mathbb{E}[x_i]$ for any $i = \mathbb{E}[x] \forall x$

$$= \mathbb{E}[(y - f(x))^2]$$

Empirical Reconstruction Error (ERE)

What is the $\mathbb{E}[\hat{f}]$?

$$\mathbb{E}_{XY}[\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2]$$

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(y_i - f(x_i))^2]$$ # By linearity of expectation

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(y - f(x))^2]$$ by IID: $\mathbb{E}[x_i]$ for any $i = \mathbb{E}[x] \forall x$

$$= \mathbb{E}[(y - f(x))^2]$$ # Yet again, it works out to the mean squared error!

Empirical Reconstruction Error (ERE)

What is the $\mathbb{E}[\hat{f}]$?

$$\mathbb{E}_{XY}[\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2]$$

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(y_i - f(x_i))^2]$$ # By linearity of expectation

$$= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(y - f(x))^2]$$ by IID: $\mathbb{E}[x_i]$ for any $i = \mathbb{E}[x] \forall x$

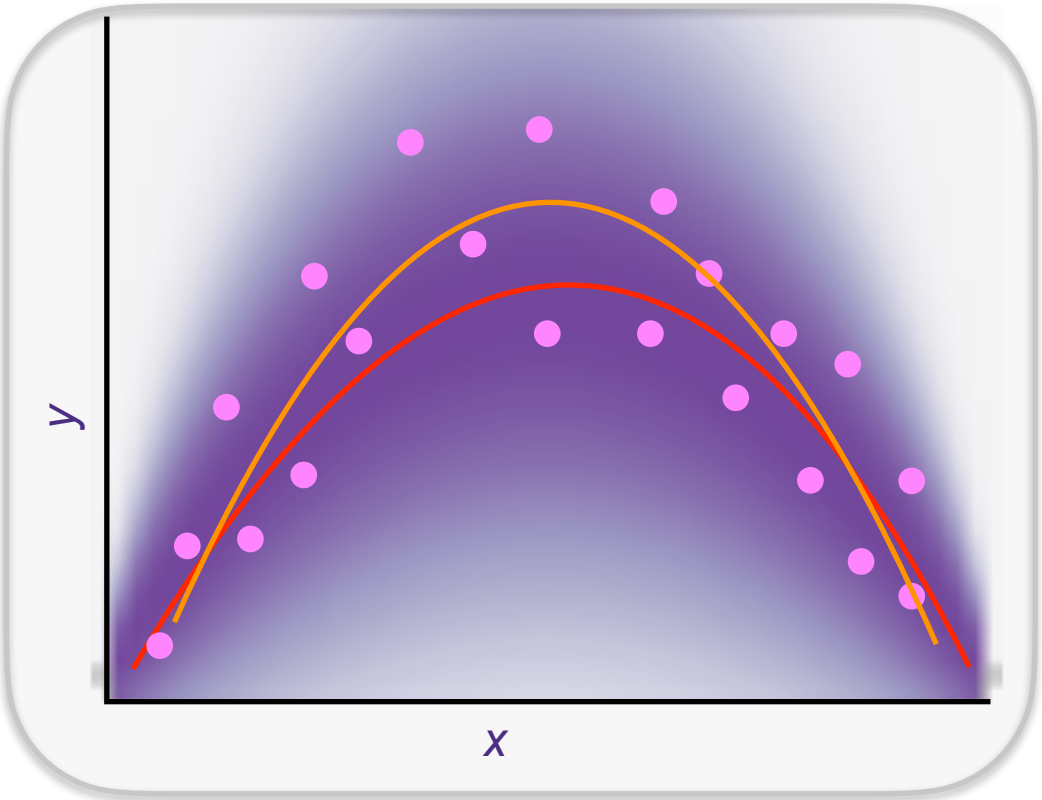
$$= \mathbb{E}[(y - f(x))^2]$$ # Yet again, it works out to the mean squared error!

Minimizing ERE → minimizing squared error loss in general

(in expectation….)

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 $i = 1, \ldots, n$

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:

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

But we only have samples:

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

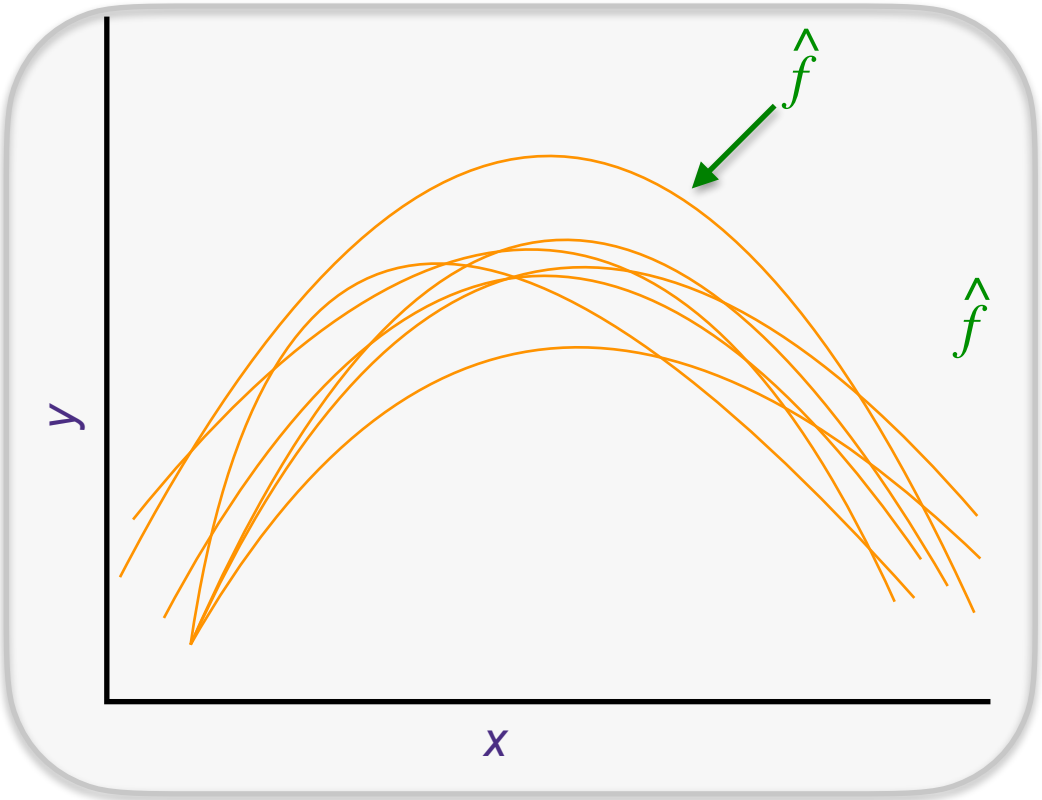
and are restricted to a function class (e.g., linear) so we compute:

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

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:

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

But we only have samples:

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

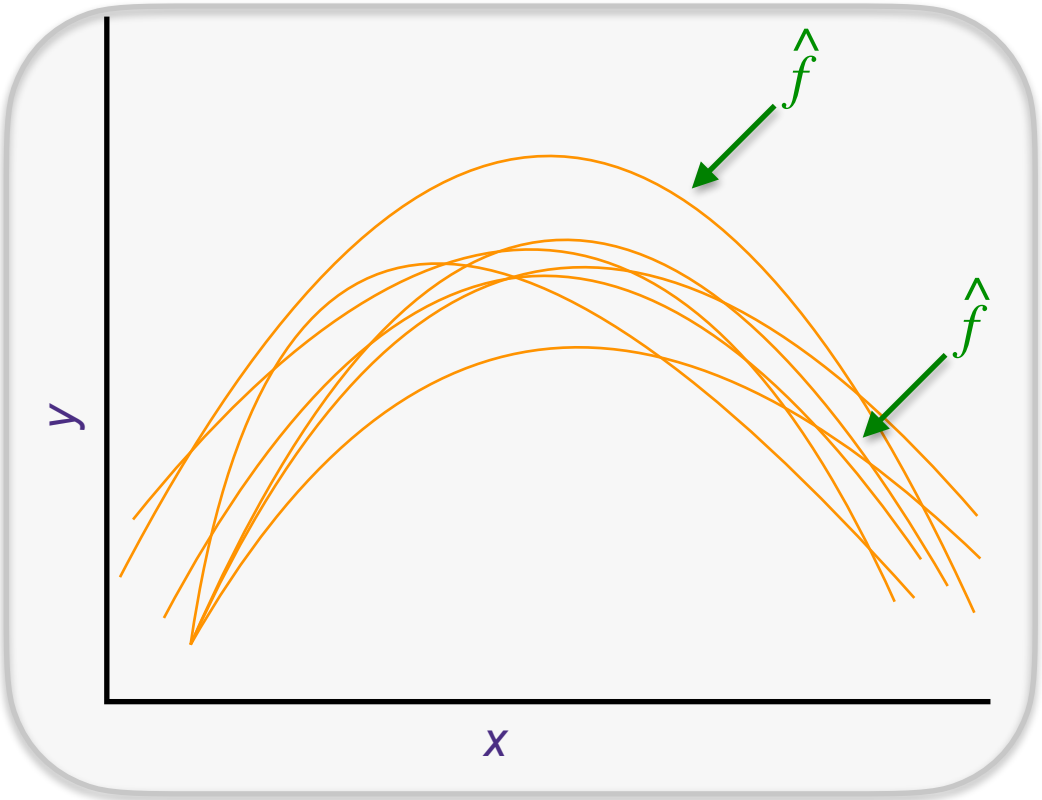
and are restricted to a function class (e.g., linear) so we compute:

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

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:

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

But we only have samples:

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

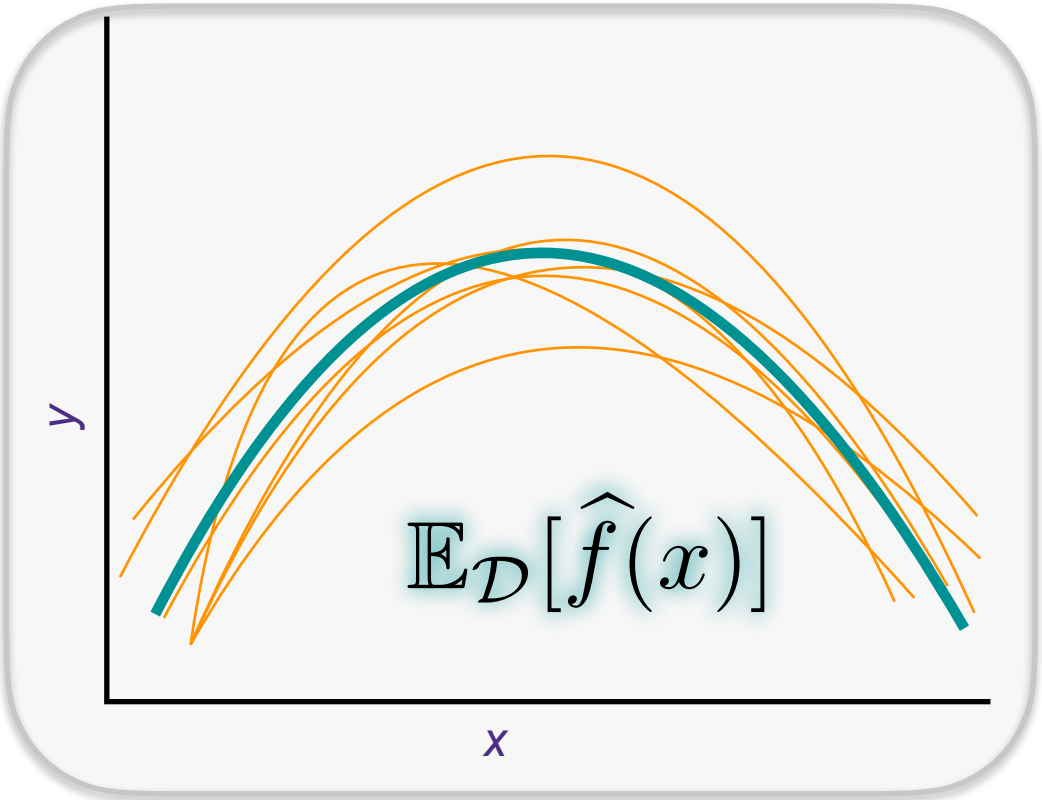
and are restricted to a function class (e.g., linear) so we compute:

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

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:

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

But we only have samples:

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

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

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

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

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

Bias-Variance Tradeoff

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

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

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

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

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

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

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

Bias-Variance Tradeoff

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

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

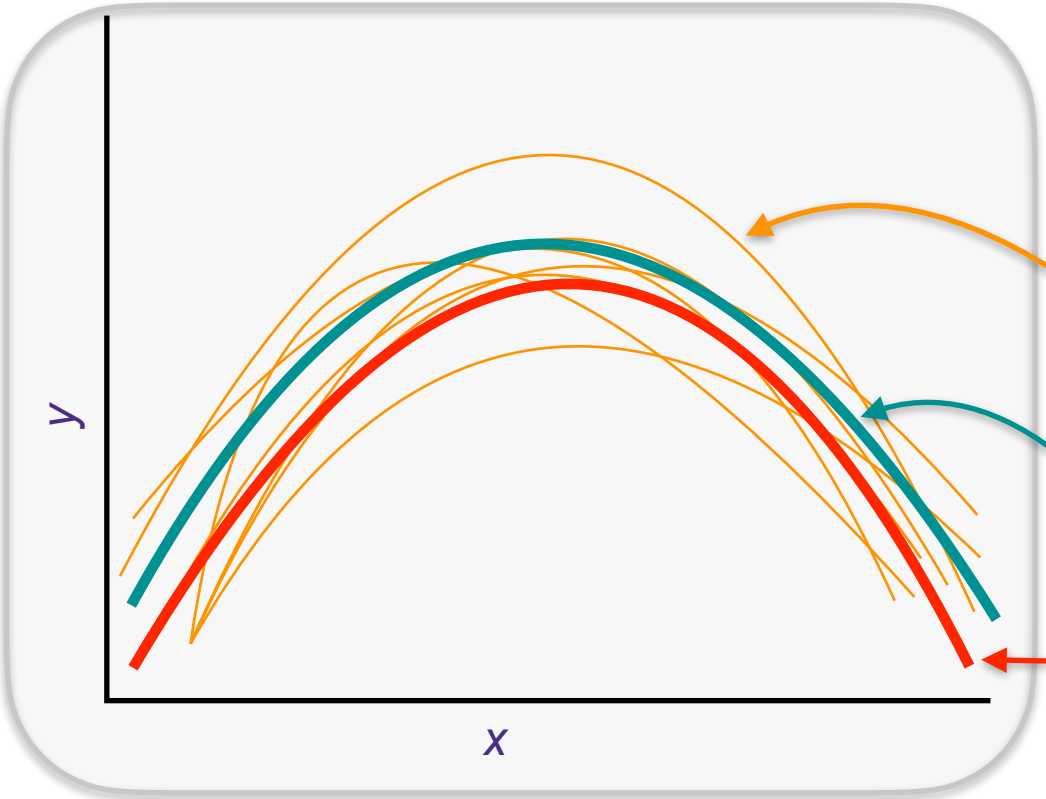
$$= \mathbb{E}_{\mathcal{D}}[(\eta(x) - \mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)])^2 + 2(\eta(x) - \mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)])(\mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)] - \widehat{f}_{\mathcal{D}}(x))$$
$$+ (\mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)]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

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

biased squared **variance**

Statistical Learning

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



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

$$\mathbb{E}_{\mathcal{D}}[\widehat{f}(x)]$$

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

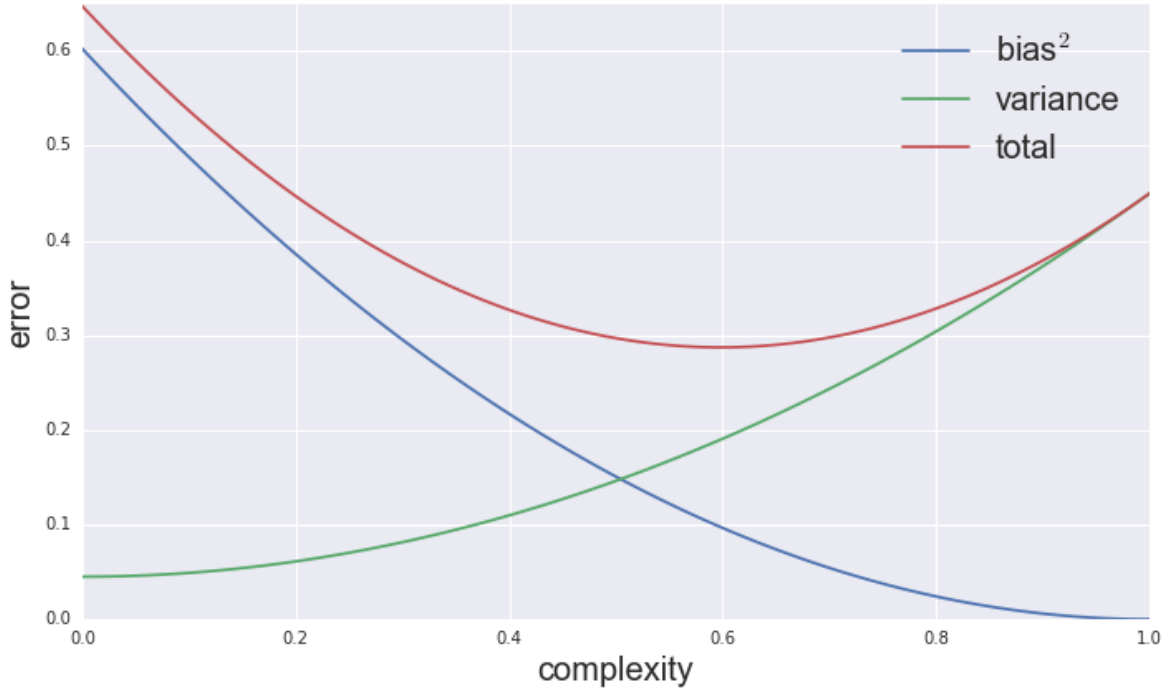
Bias-Variance Tradeoff

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

irreducible error

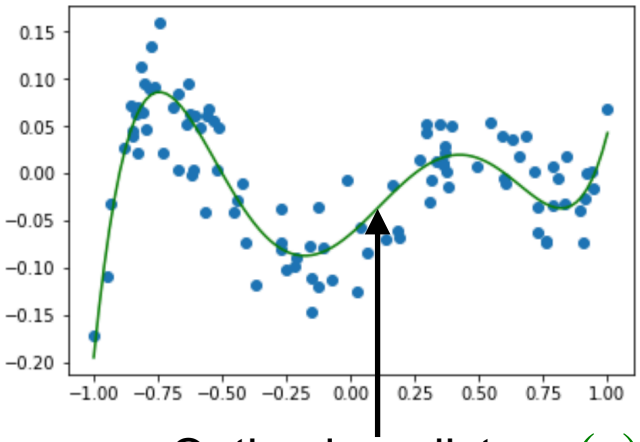
$$+(\eta(x) - \mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)])^2 + \mathbb{E}_{\mathcal{D}}[(\mathbb{E}_{\mathcal{D}}[\widehat{f}_{\mathcal{D}}(x)] - \widehat{f}_{\mathcal{D}}(x))^2]$$

biased squared **variance**



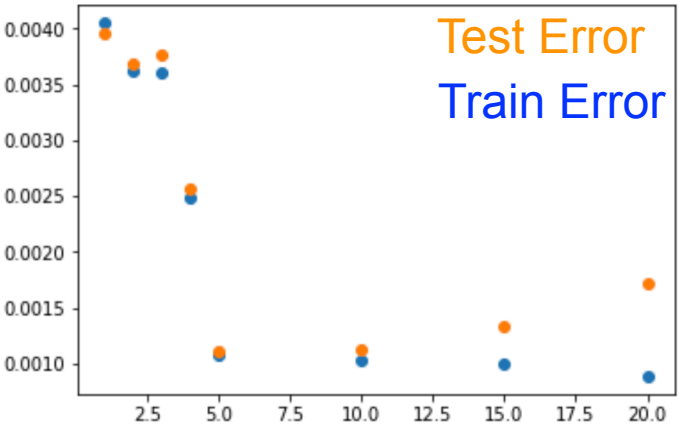
Overfitting

Test error vs. model complexity



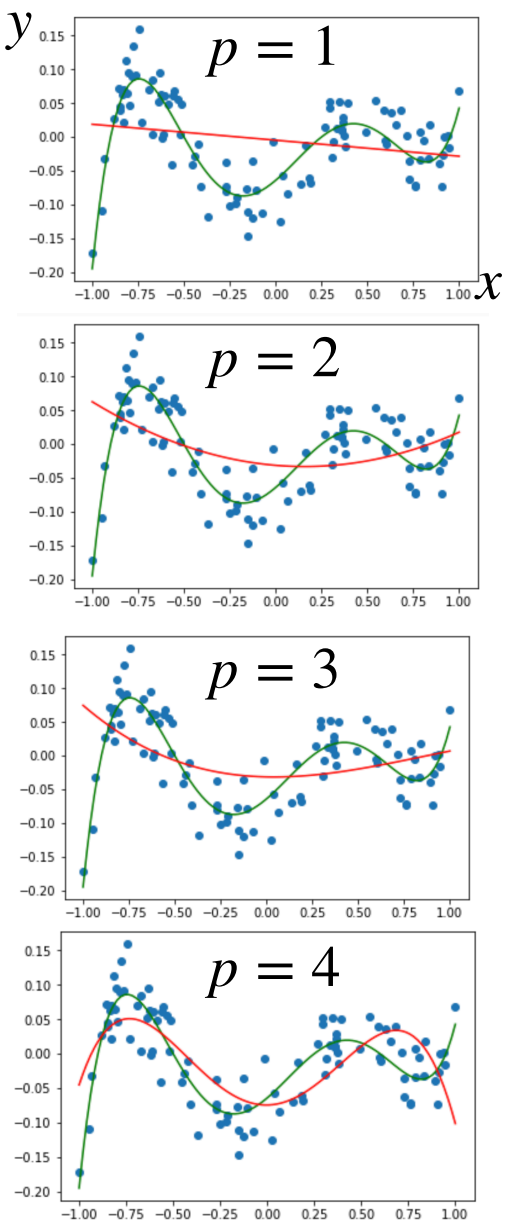
Optimal predictor $\eta(x)$ is degree-5 polynomial

Error

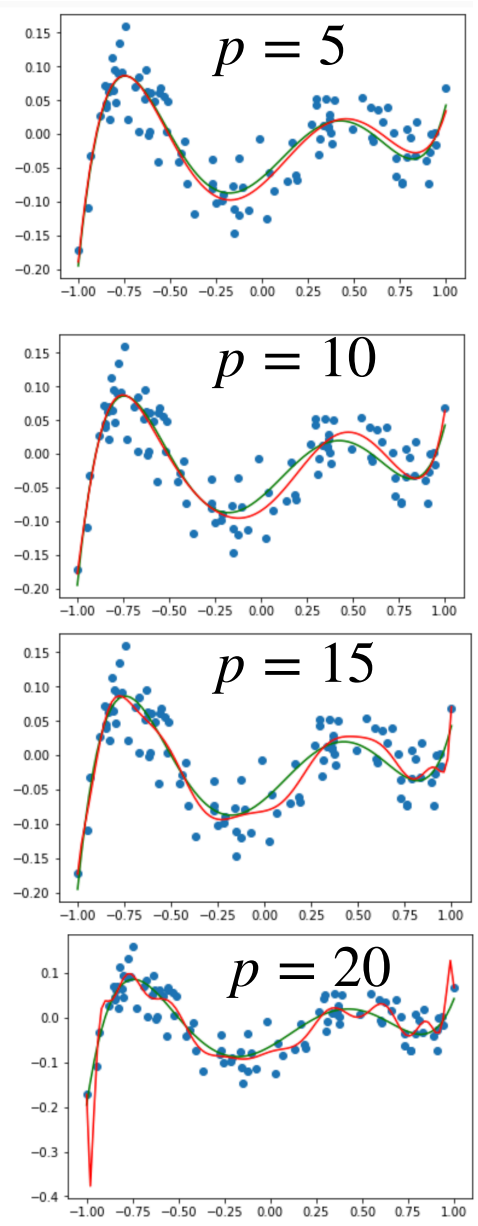


degree p of the polynomial regression

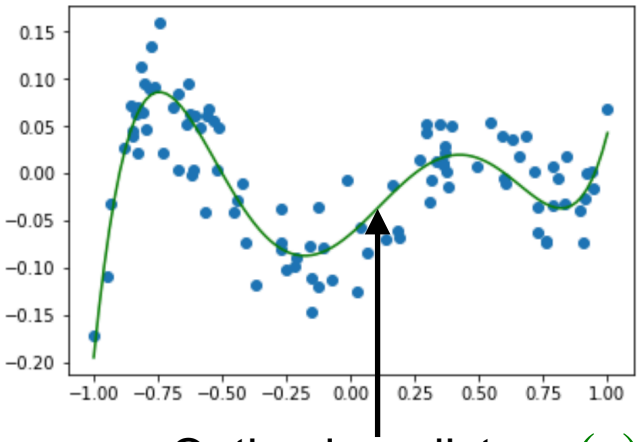
Simple model:
Model complexity is below the complexity of $\eta(x)$



Complex model:
Fits noise in train data, diverging from $\eta(x)$

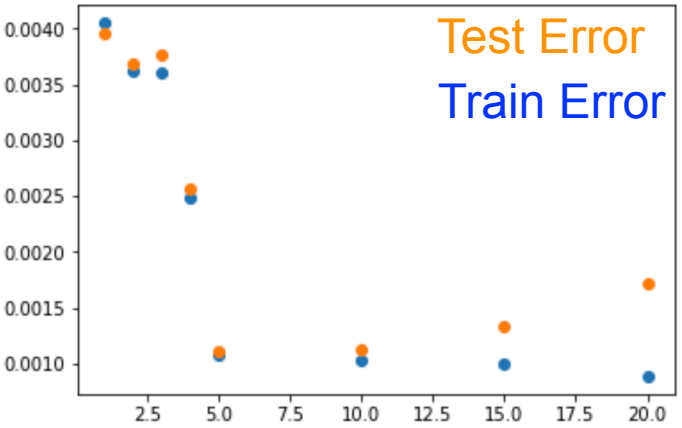


Test error vs. model complexity



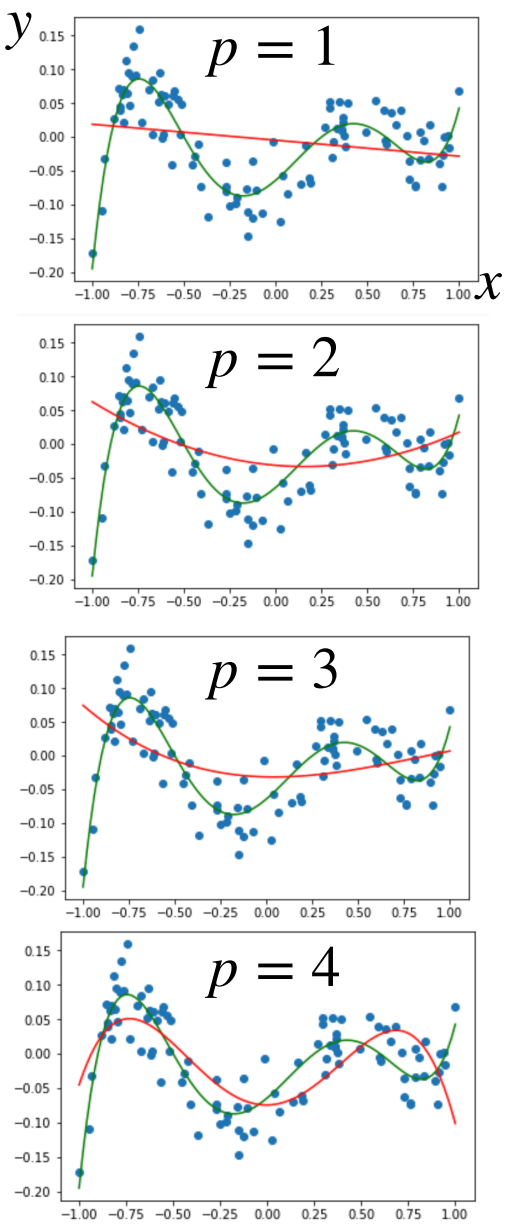
Optimal predictor $\eta(x)$ is degree-5 polynomial

Error

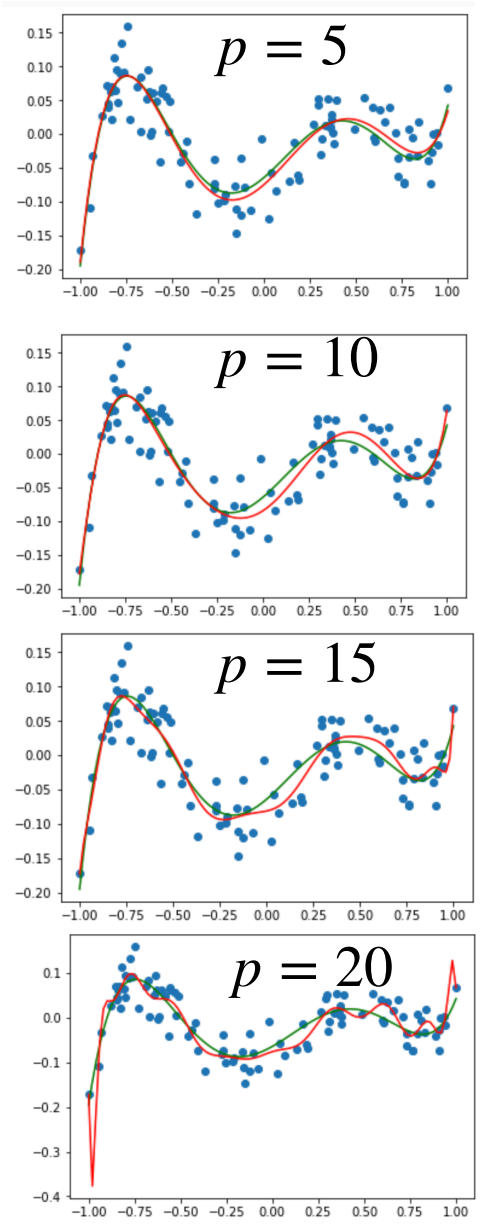


degree p of the polynomial regression

Simple model: # bias
Model complexity is below the complexity of $\eta(x)$

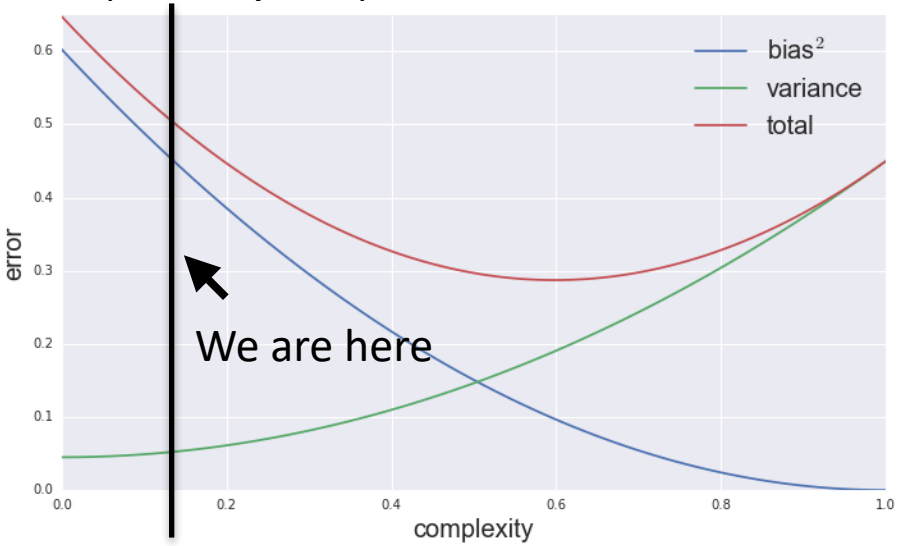


Complex model:
Fits noise in train data, diverging from $\eta(x)$

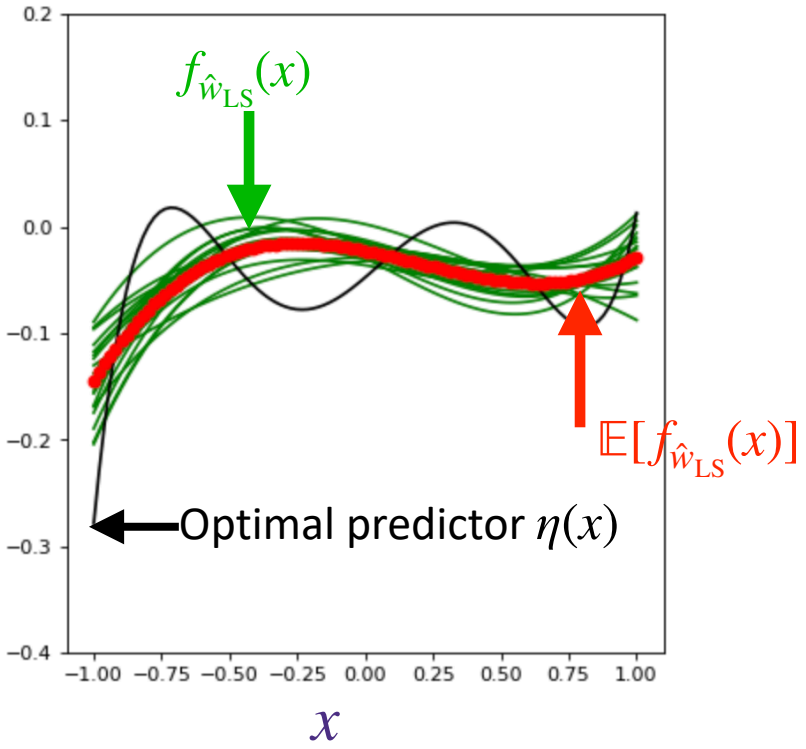


Recap: Bias-variance tradeoff with simple model

(Conceptual) bias variance tradeoff



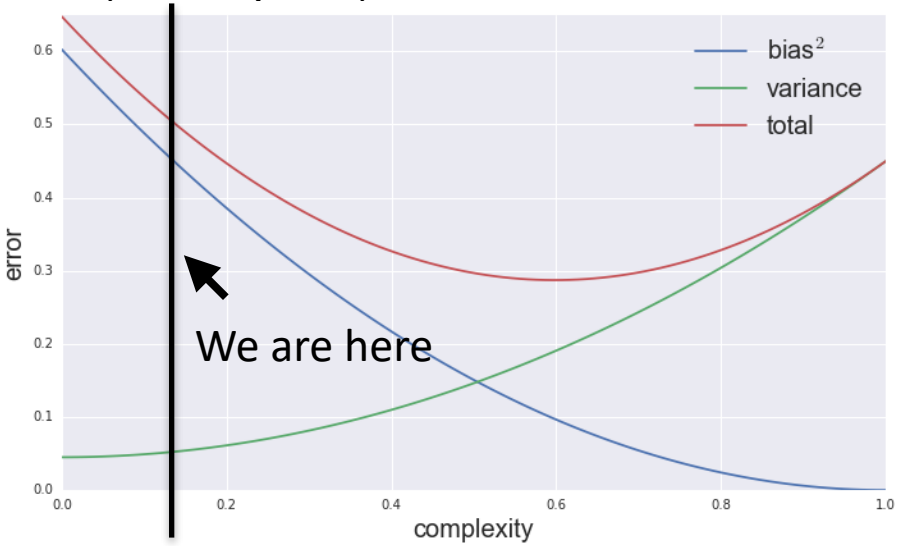
With degree-3 polynomials, we underfit



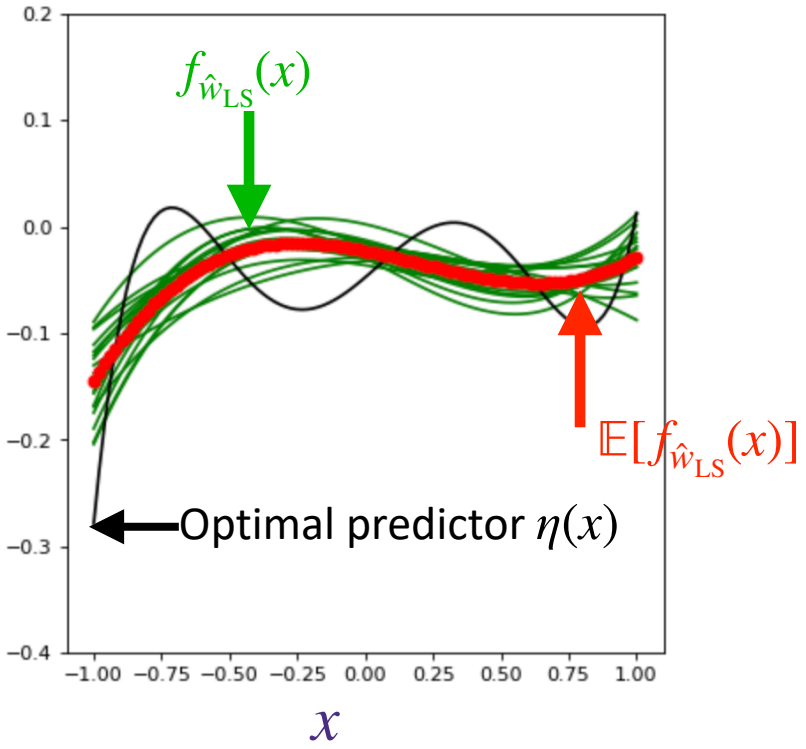
- When model **complexity is low** (lower than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor, $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$
  - Variance of our predictor, $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

Recap: Bias-variance tradeoff with simple model

(Conceptual) bias variance tradeoff



With degree-3 polynomials, we underfit

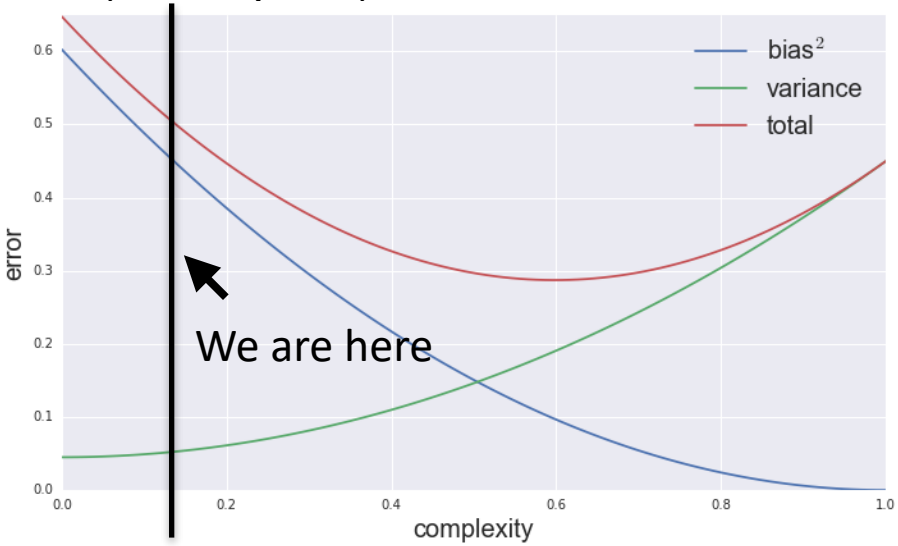


- When model **complexity is low** (lower than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

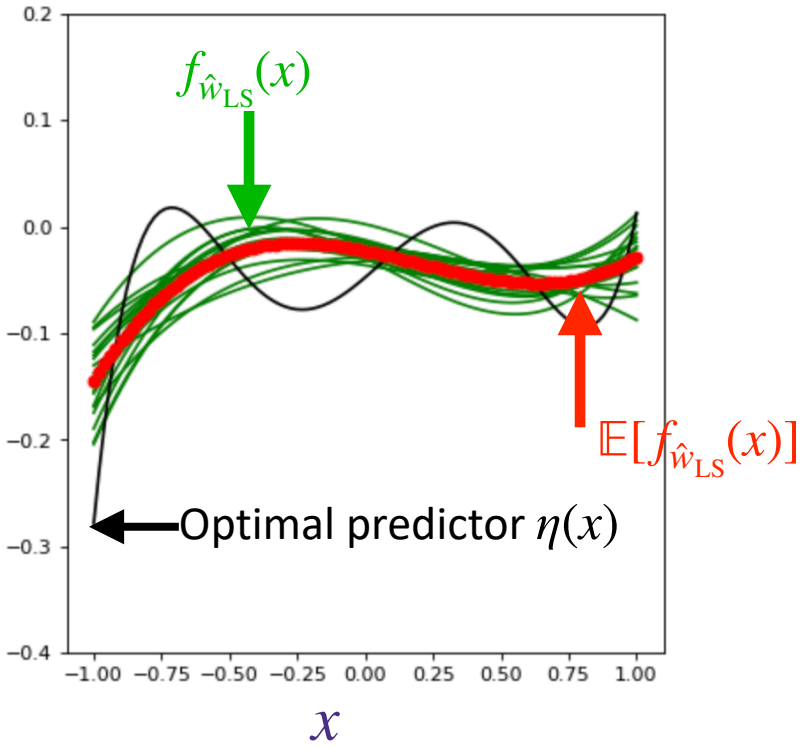
    # Large
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

Recap: Bias-variance tradeoff with simple model

(Conceptual) bias variance tradeoff



With degree-3 polynomials, we underfit



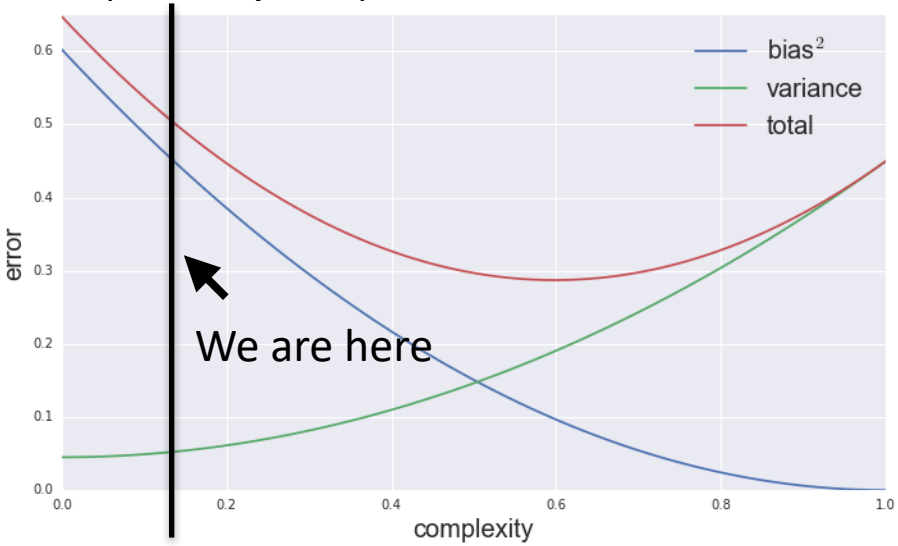
- When model **complexity is low** (lower than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

    # Large
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

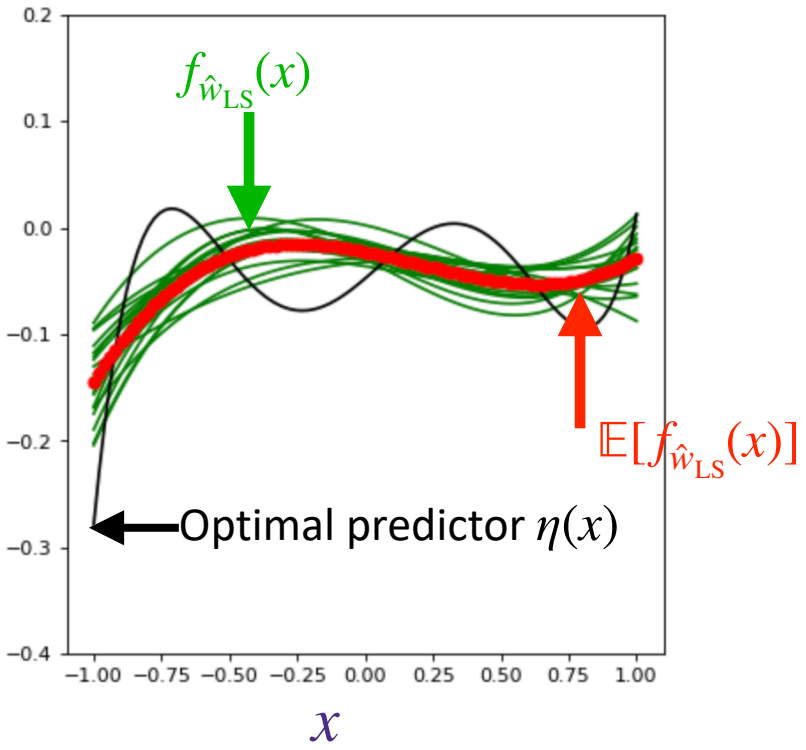
    # Small

Recap: Bias-variance tradeoff with simple model

(Conceptual) bias variance tradeoff



With degree-3 polynomials, we underfit



- When model **complexity is low** (lower than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

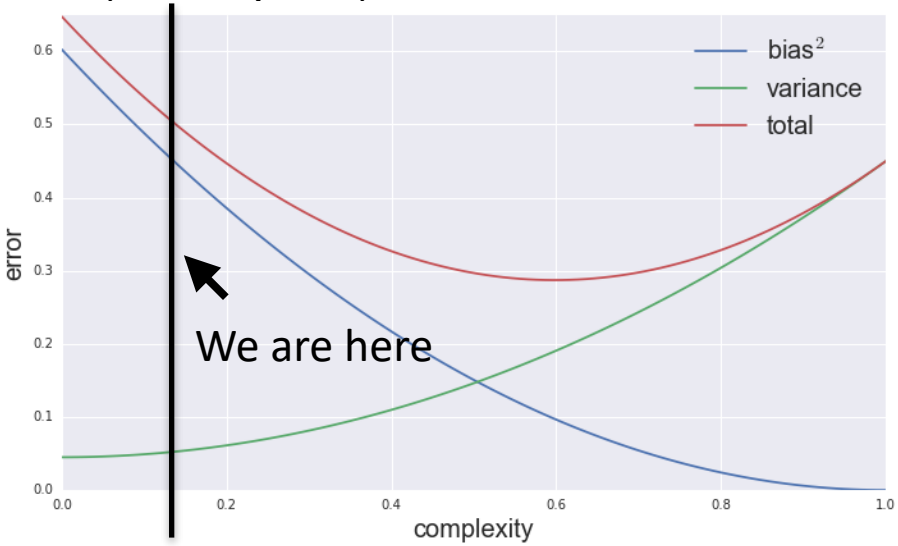
    # Large
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

    # Small

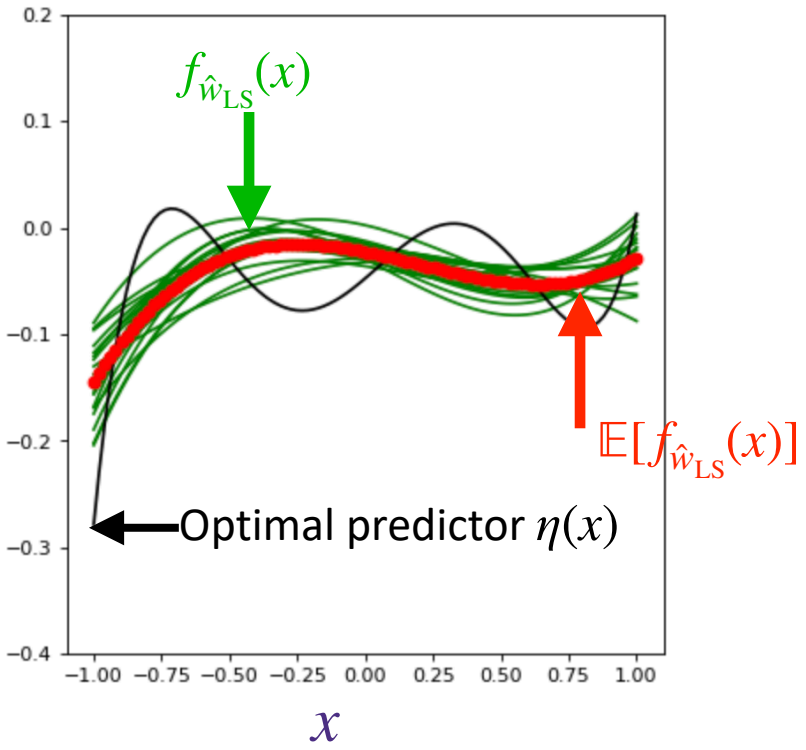
- **If we have more samples (larger n), then**

Recap: Bias-variance tradeoff with simple model

(Conceptual) bias variance tradeoff



With degree-3 polynomials, we underfit



- When model **complexity is low** (lower than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor, $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

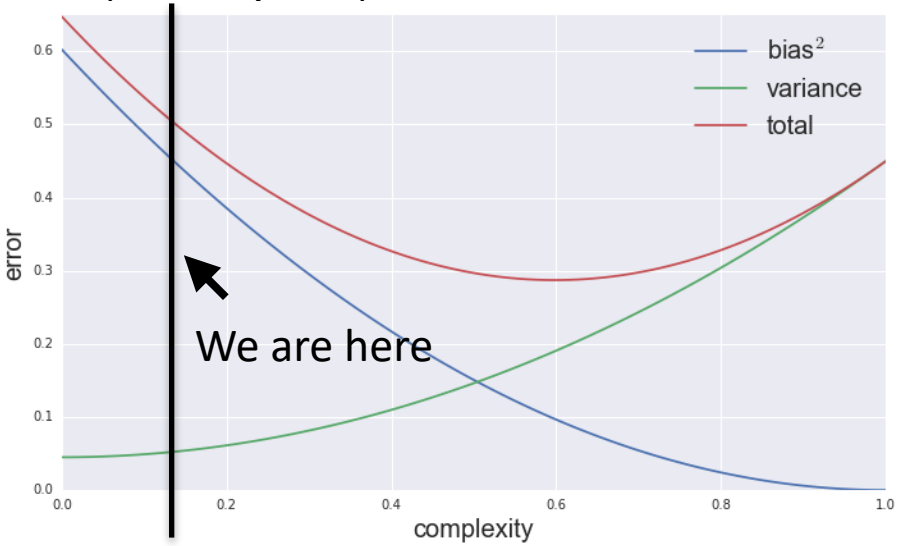
    # Large
  - Variance of our predictor, $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

    # Small

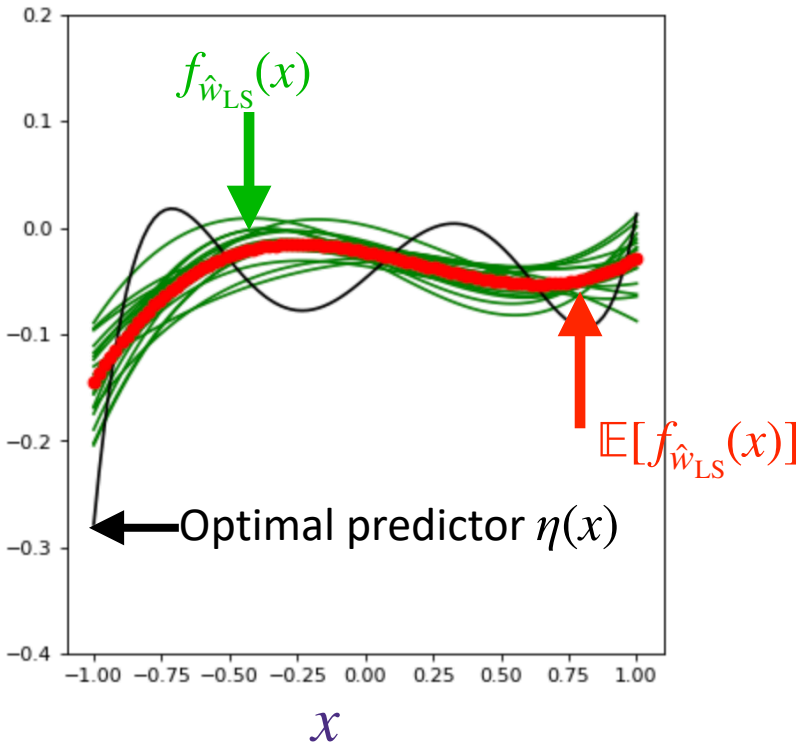
- **If we have more samples (larger n), then**
  - What happens to bias?

Recap: Bias-variance tradeoff with simple model

(Conceptual) bias variance tradeoff



With degree-3 polynomials, we underfit



- When model **complexity is low** (lower than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor, $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

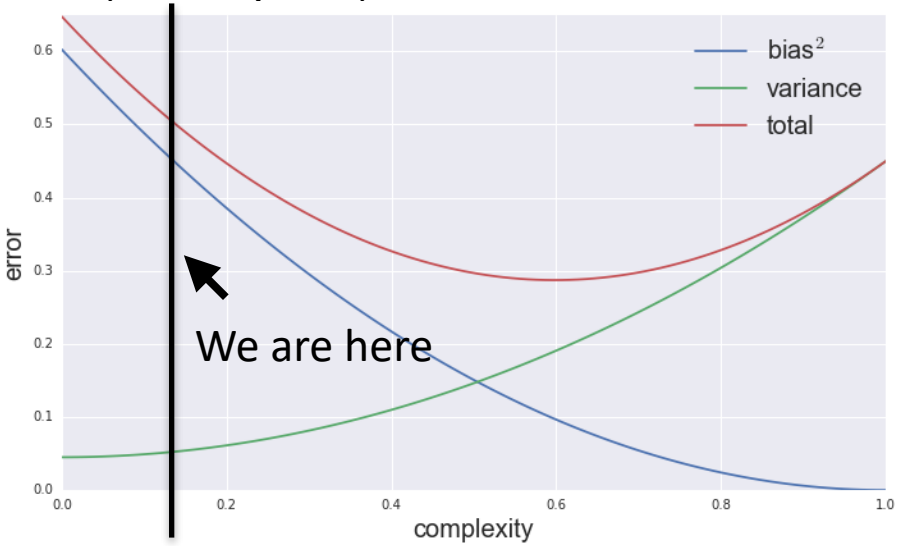
    # Large
  - Variance of our predictor, $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

    # Small

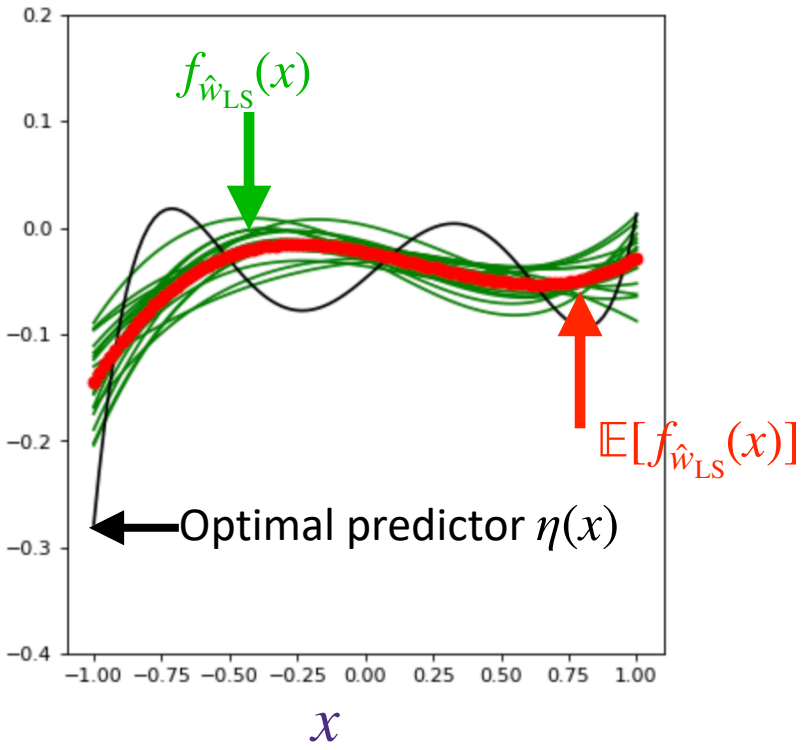
- **If we have more samples (larger n), then**
  - What happens to bias? # stays the same

Recap: Bias-variance tradeoff with simple model

(Conceptual) bias variance tradeoff



With degree-3 polynomials, we underfit



- When model **complexity is low** (lower than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

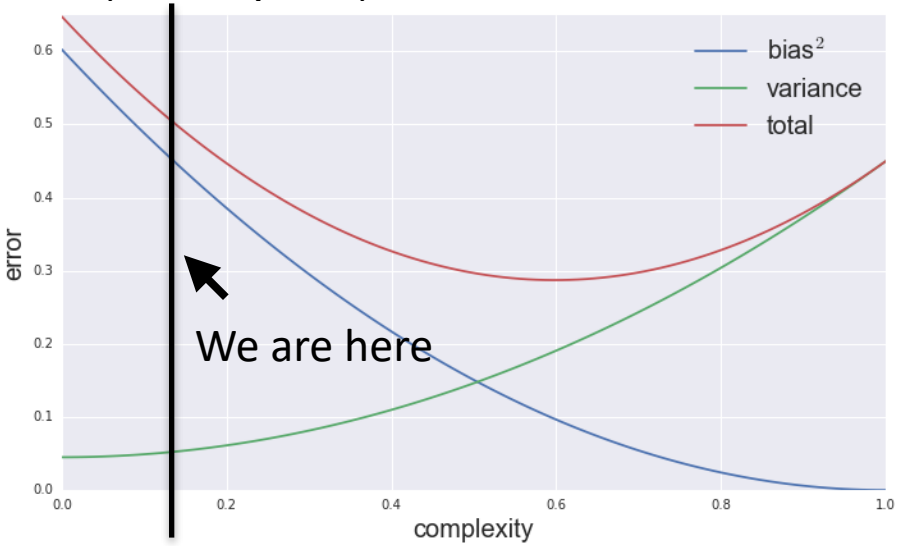
    # Large
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

    # Small

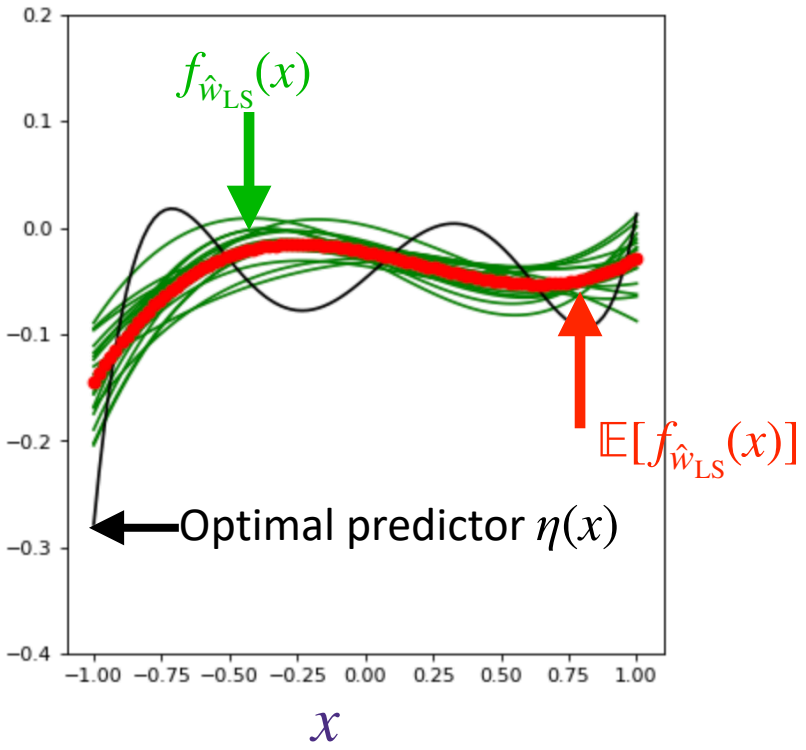
- **If we have more samples (larger n), then**
  - What happens to bias? # stays the same
  - What happens to variance?

Recap: Bias-variance tradeoff with simple model

(Conceptual) bias variance tradeoff



With degree-3 polynomials, we underfit



- When model **complexity is low** (lower than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

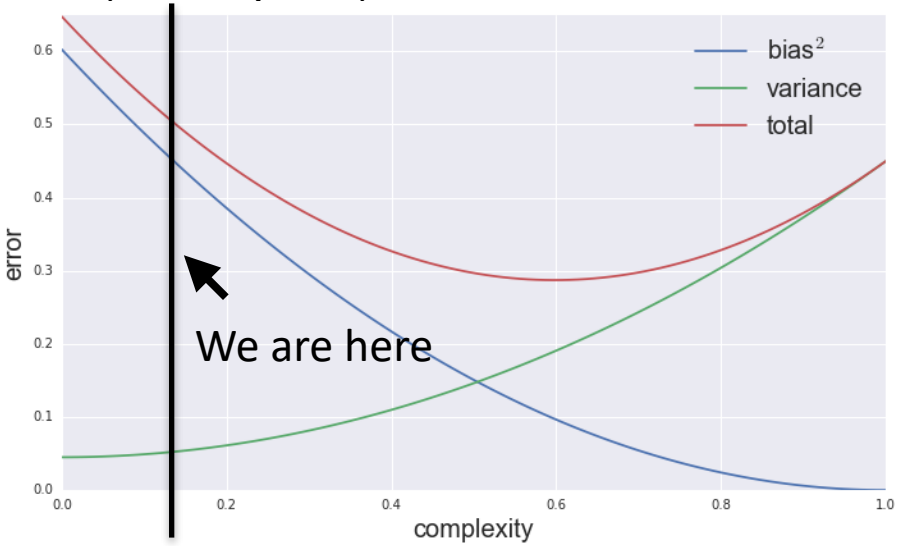
    # Large
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

    # Small

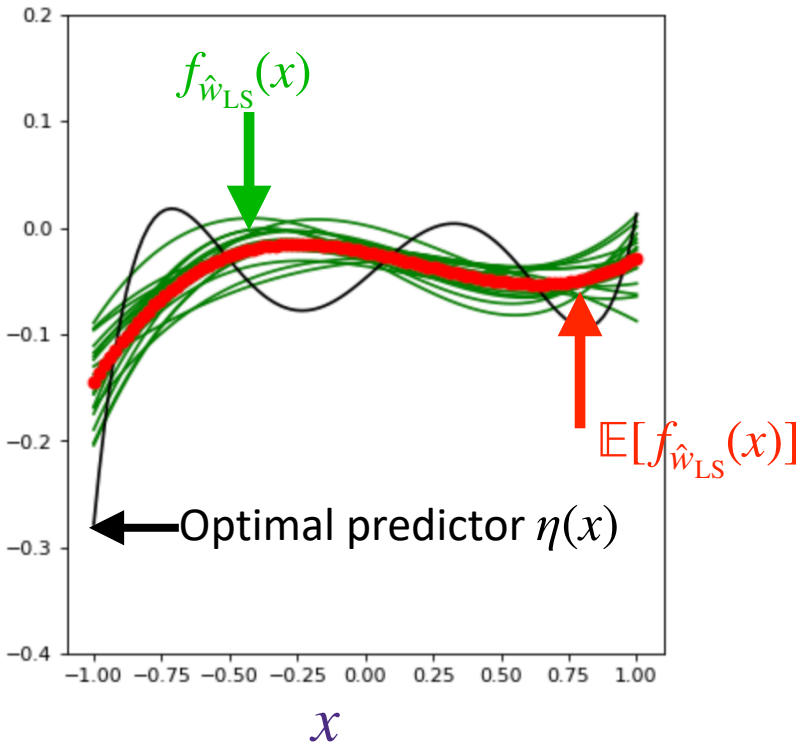
- **If we have more samples (larger n), then**
  - What happens to bias? # stays the same
  - What happens to variance? # decreases

Recap: Bias-variance tradeoff with simple model

(Conceptual) bias variance tradeoff



With degree-3 polynomials, we underfit



- When model **complexity is low** (lower than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

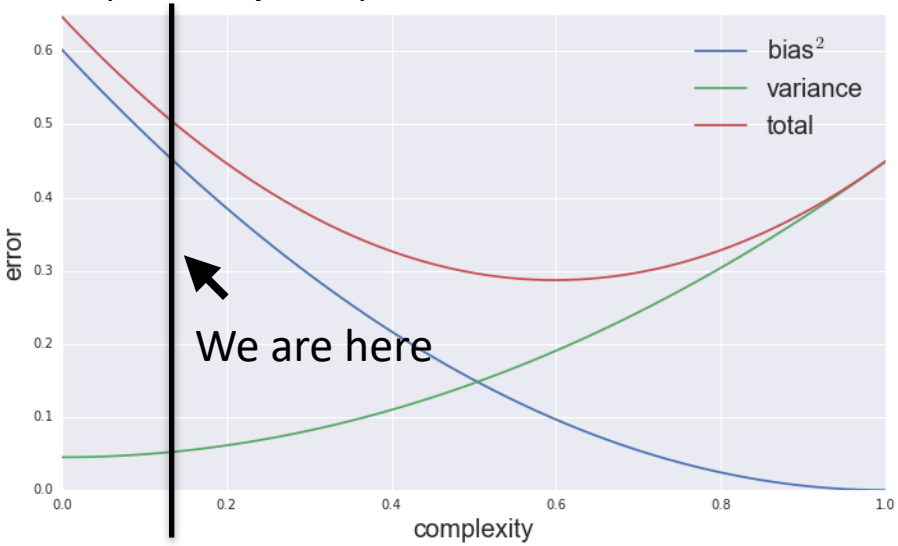
    # Large
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

    # Small

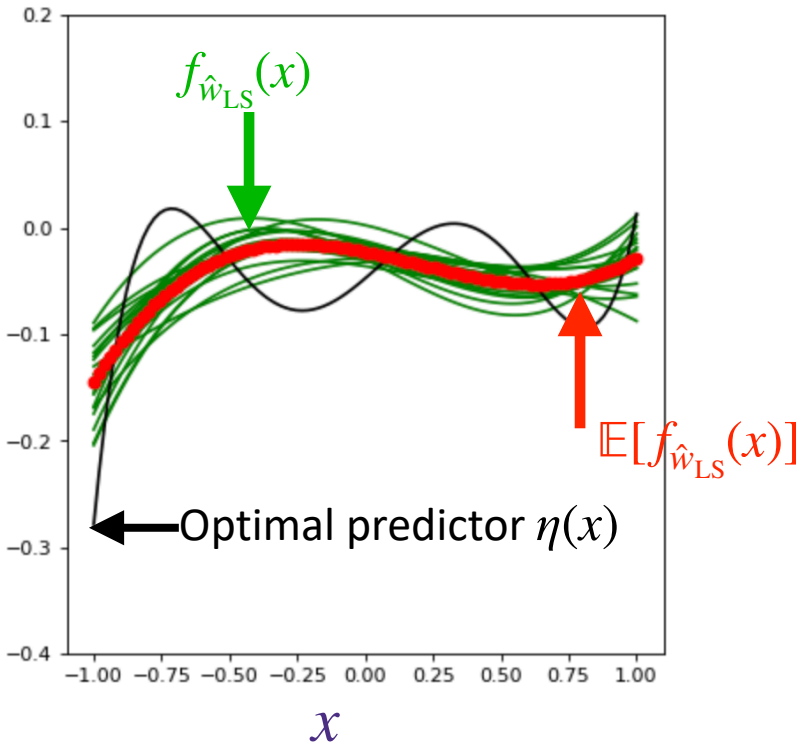
- **If we have more samples (larger n), then**
  - What happens to bias? # stays the same
  - What happens to variance? # decreases
  - What happens to overall test error?

Recap: Bias-variance tradeoff with simple model

(Conceptual) bias variance tradeoff



With degree-3 polynomials, we underfit



- When model **complexity is low** (lower than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor, $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

    # Large
  - Variance of our predictor, $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

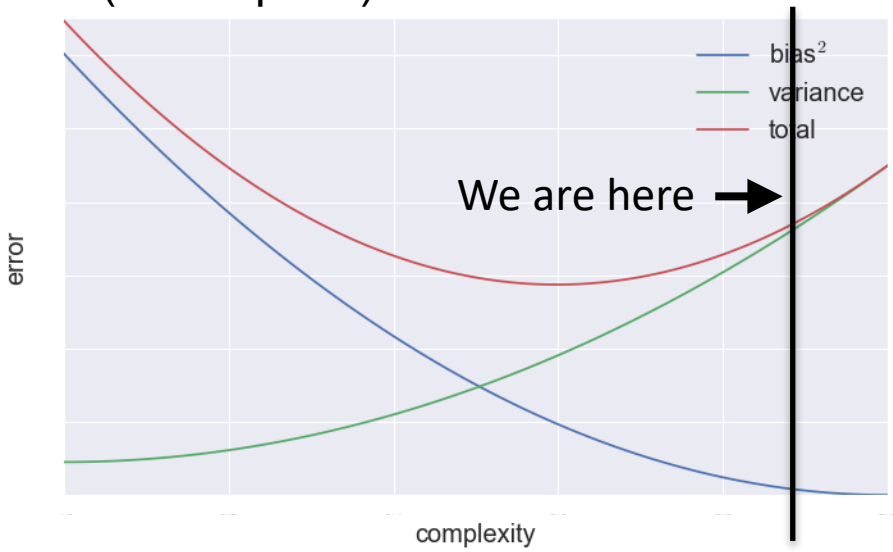
    # Small

- **If we have more samples (larger n), then**
  - What happens to bias? # stays the same
  - What happens to variance? # decreases
  - What happens to overall test error?

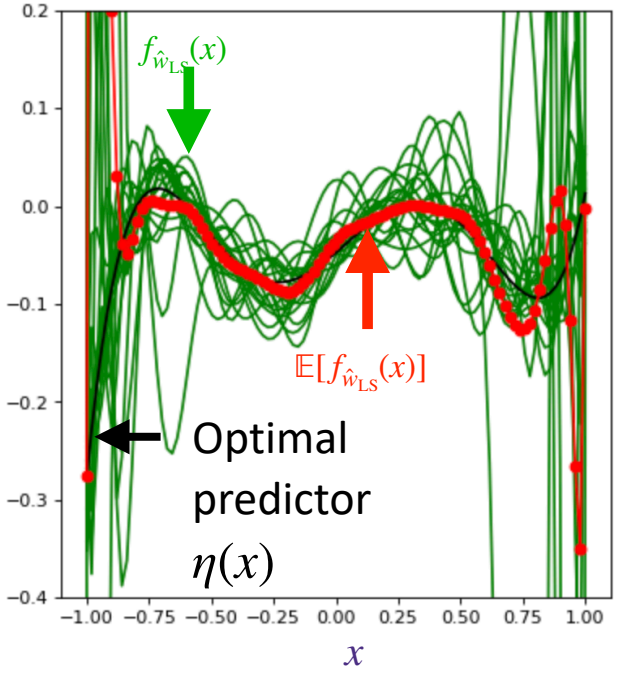
    # Stays high because bias was dominating our error term

Recap: Bias-variance tradeoff with complex model

(Conceptual) bias variance tradeoff



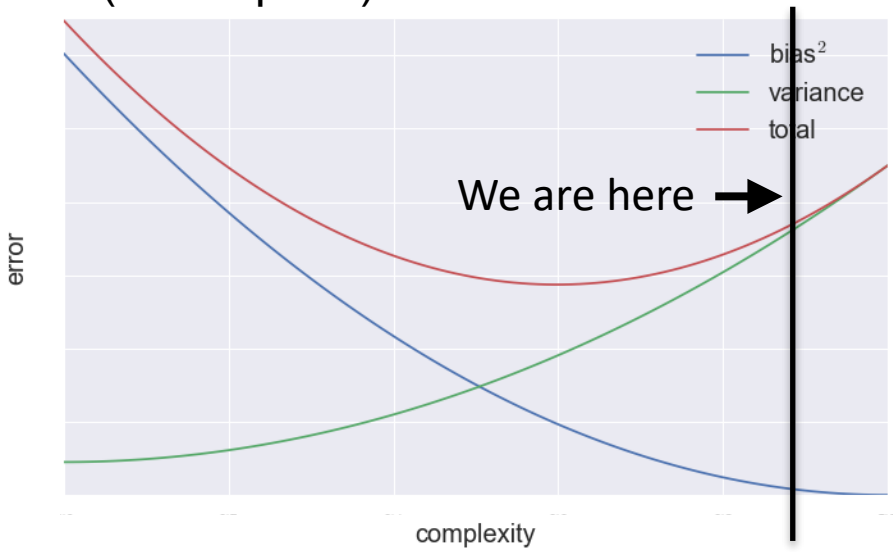
With degree-20 polynomials, we overfit



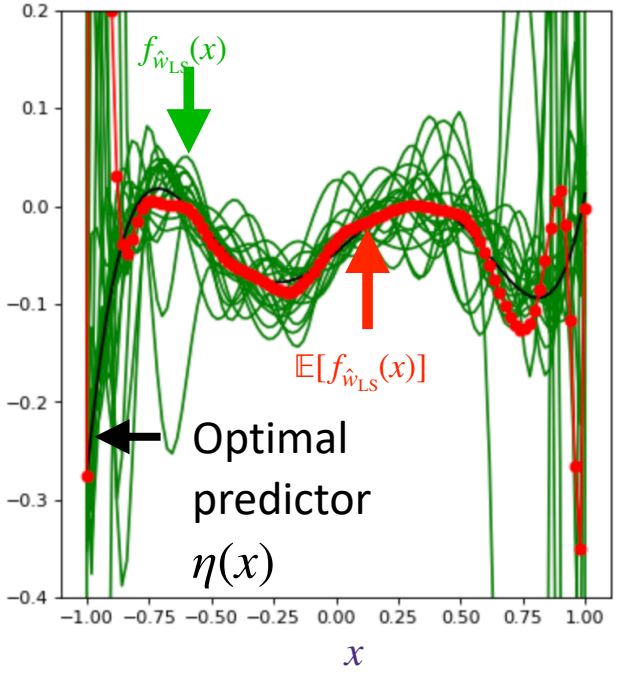
- When model **complexity is high** (higher than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

Recap: Bias-variance tradeoff with complex model

(Conceptual) bias variance tradeoff



With degree-20 polynomials, we overfit

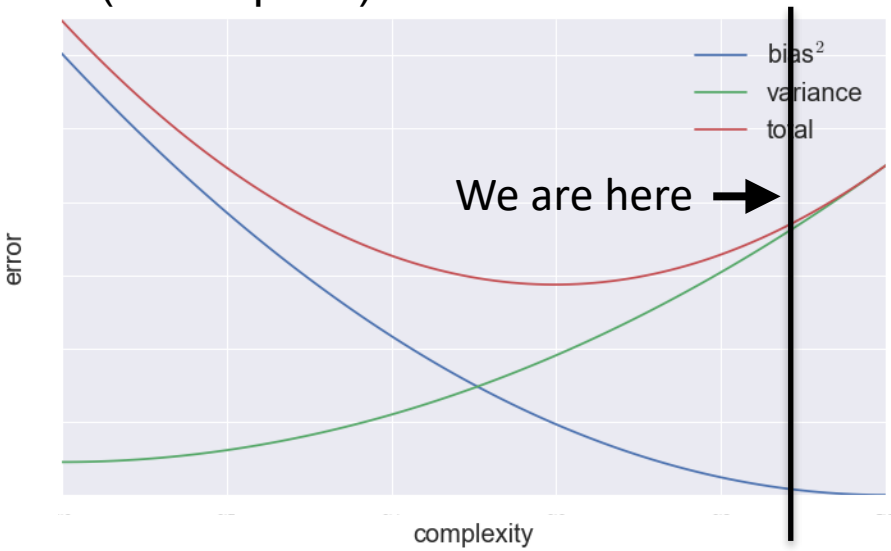


- When model **complexity is high** (higher than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

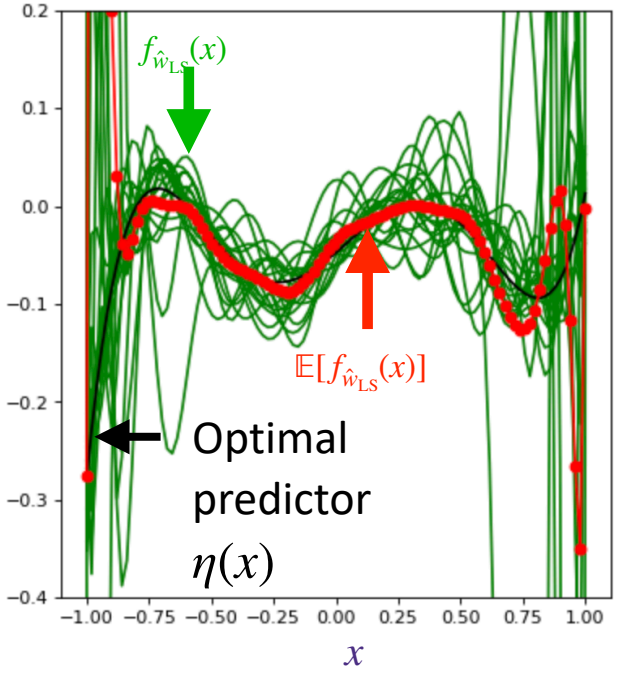
    # Low
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

Recap: Bias-variance tradeoff with complex model

(Conceptual) bias variance tradeoff



With degree-20 polynomials, we overfit



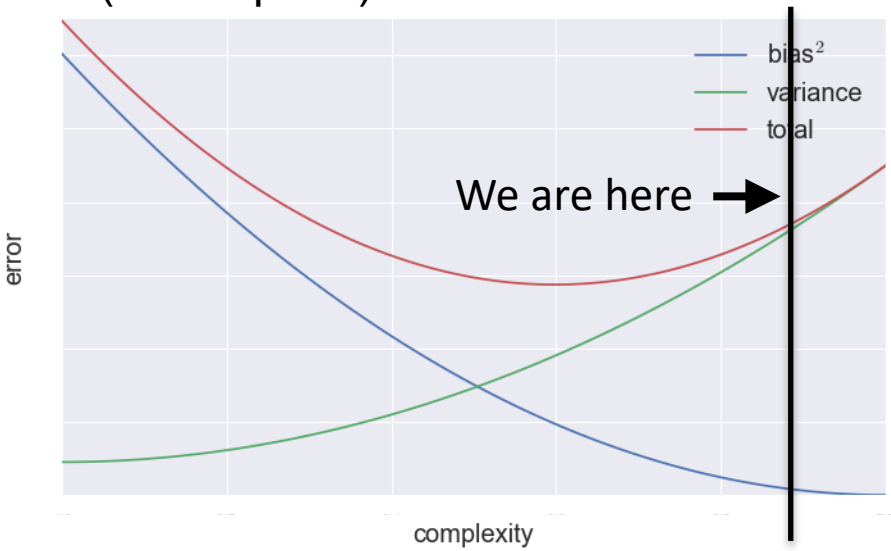
- When model **complexity is high** (higher than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

    # Low
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

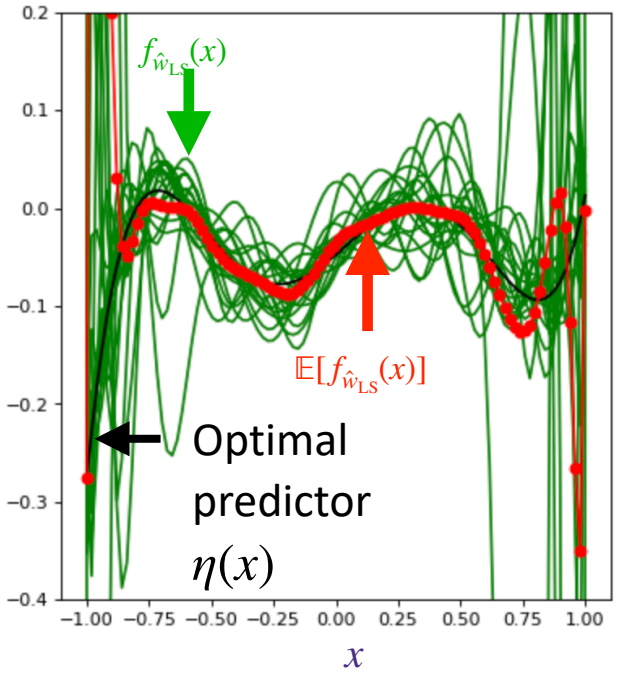
    # High

Recap: Bias-variance tradeoff with complex model

(Conceptual) bias variance tradeoff



With degree-20 polynomials, we overfit



- When model **complexity is high** (higher than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

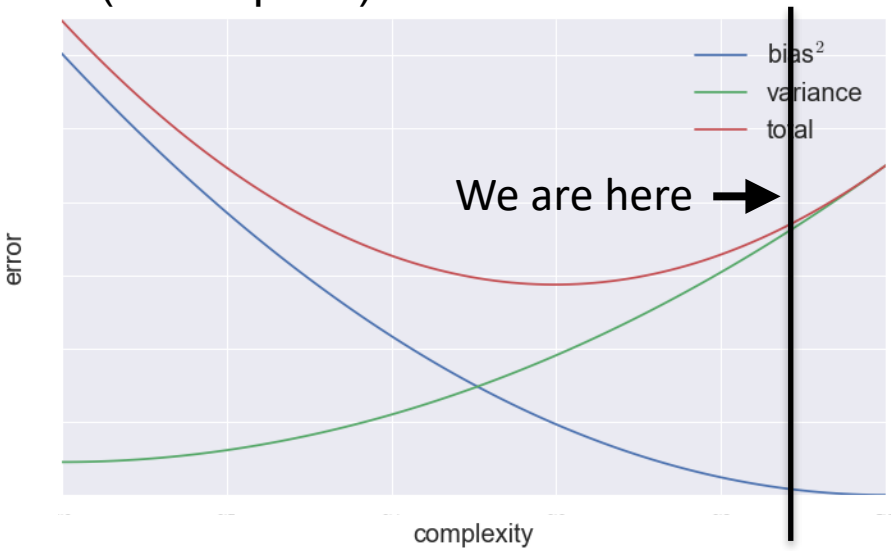
    # Low
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

    # High

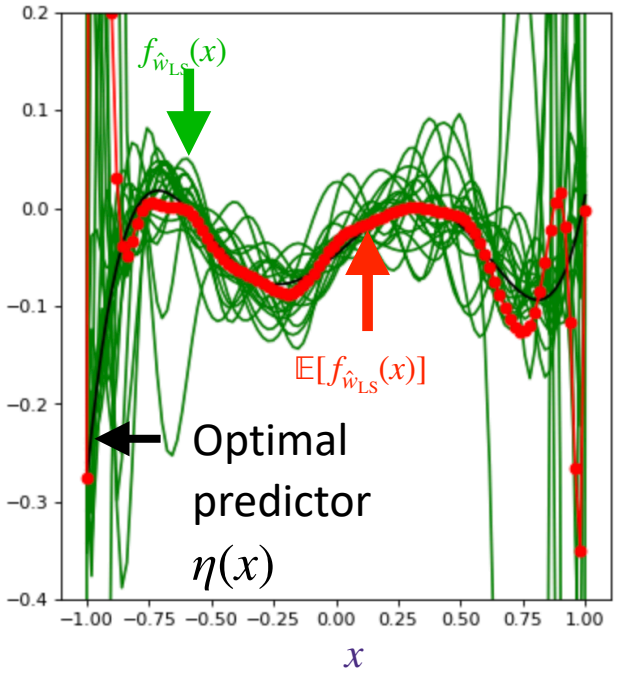
- **If we have more samples (larger n), then**

Recap: Bias-variance tradeoff with complex model

(Conceptual) bias variance tradeoff



With degree-20 polynomials, we overfit



- When model **complexity is high** (higher than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

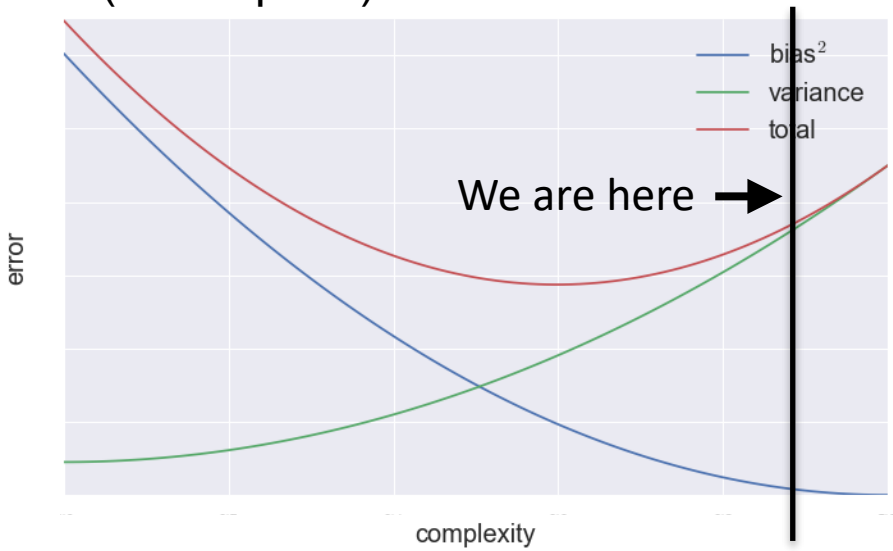
    # Low
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

    # High

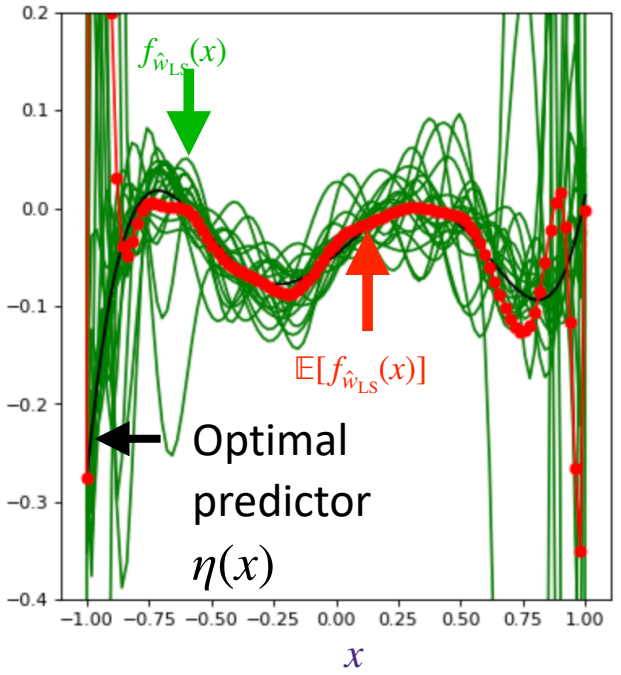
- **If we have more samples (larger n), then**
  - What happens to bias?

Recap: Bias-variance tradeoff with complex model

(Conceptual) bias variance tradeoff



With degree-20 polynomials, we overfit



- When model **complexity is high** (higher than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

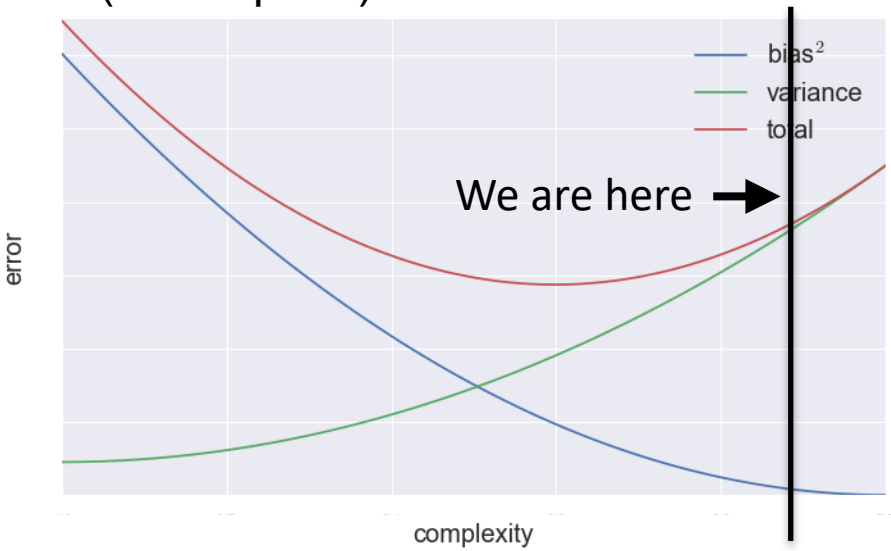
    # Low
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

    # High

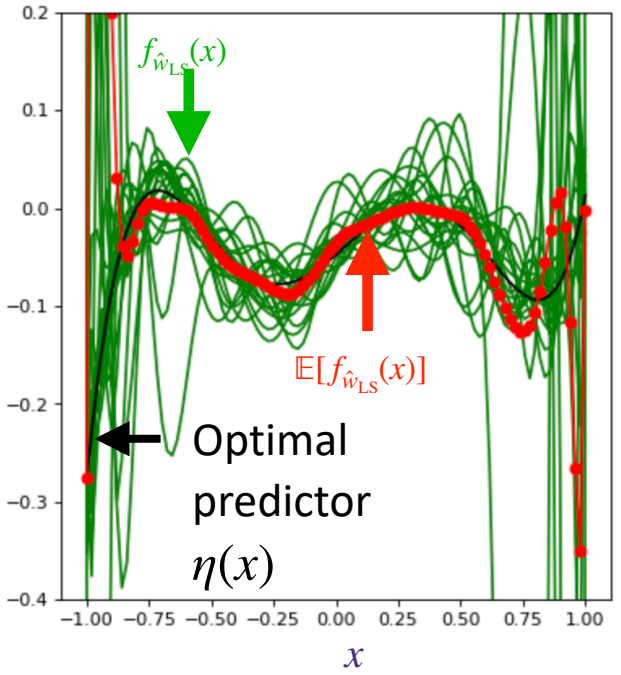
- **If we have more samples (larger n), then**
  - What happens to bias? # stays the same

Recap: Bias-variance tradeoff with complex model

(Conceptual) bias variance tradeoff



With degree-20 polynomials, we overfit



- When model **complexity is high** (higher than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

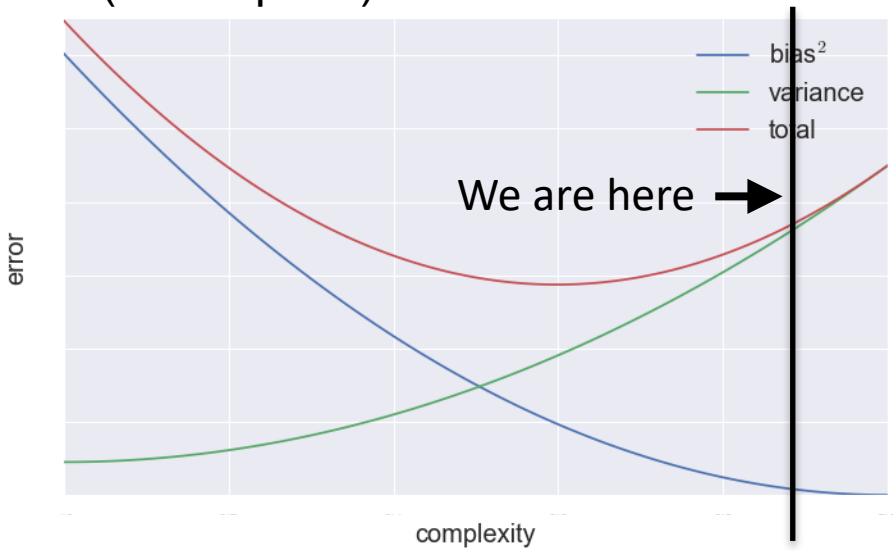
    # Low
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

    # High

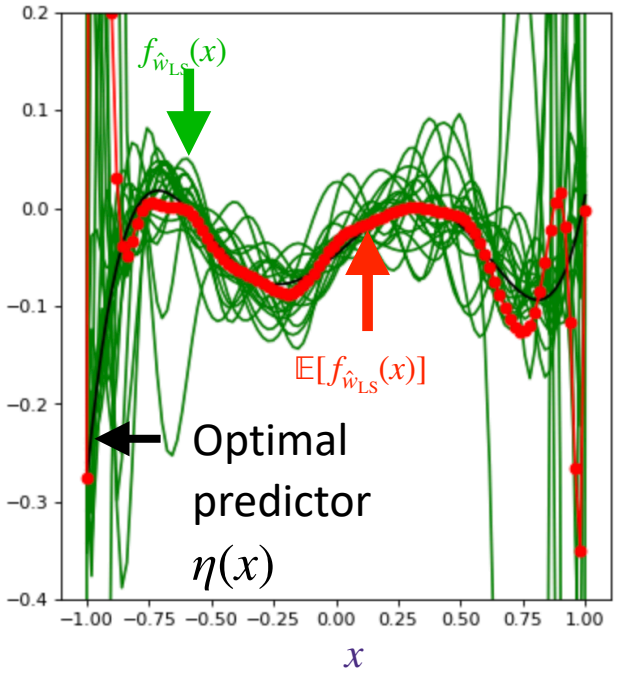
- **If we have more samples (larger n), then**
  - What happens to bias? # stays the same
  - What happens to variance?

Recap: Bias-variance tradeoff with complex model

(Conceptual) bias variance tradeoff



With degree-20 polynomials, we overfit



- When model **complexity is high** (higher than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

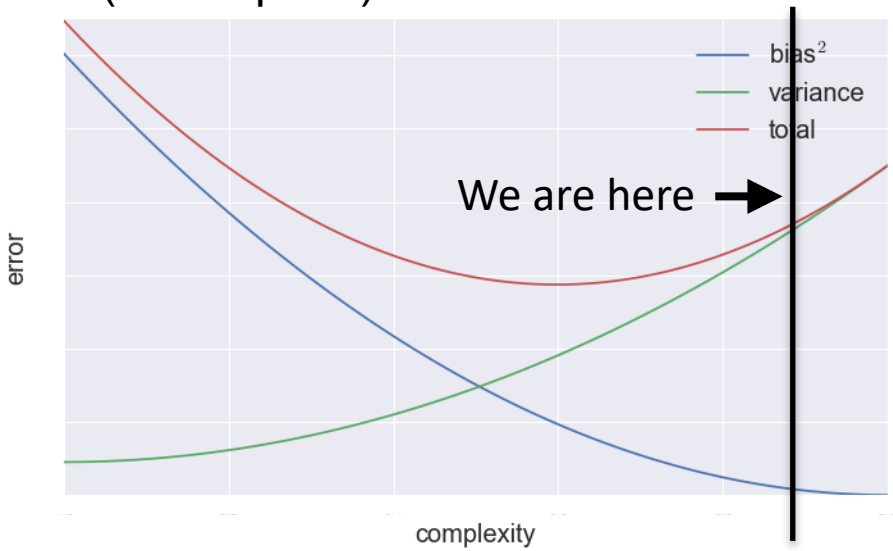
    # Low
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

    # High

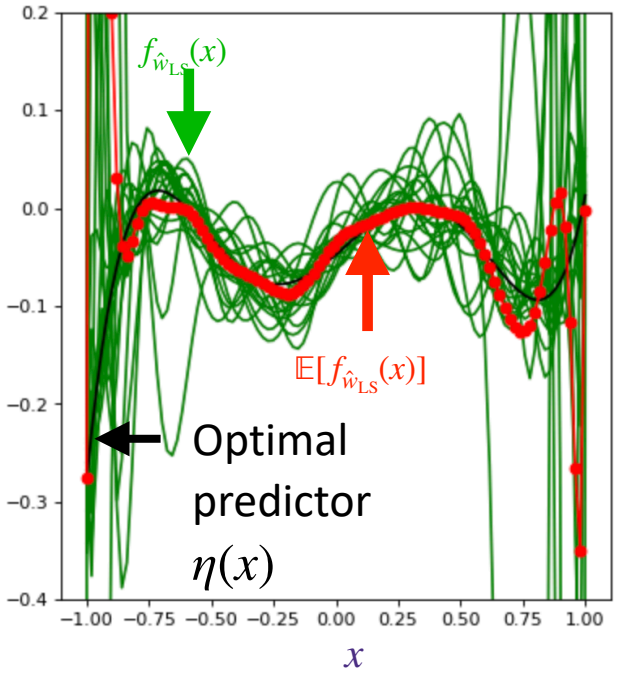
- **If we have more samples (larger n), then**
  - What happens to bias? # stays the same
  - What happens to variance? # decreases

Recap: Bias-variance tradeoff with complex model

(Conceptual) bias variance tradeoff



With degree-20 polynomials, we overfit



- When model **complexity is high** (higher than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

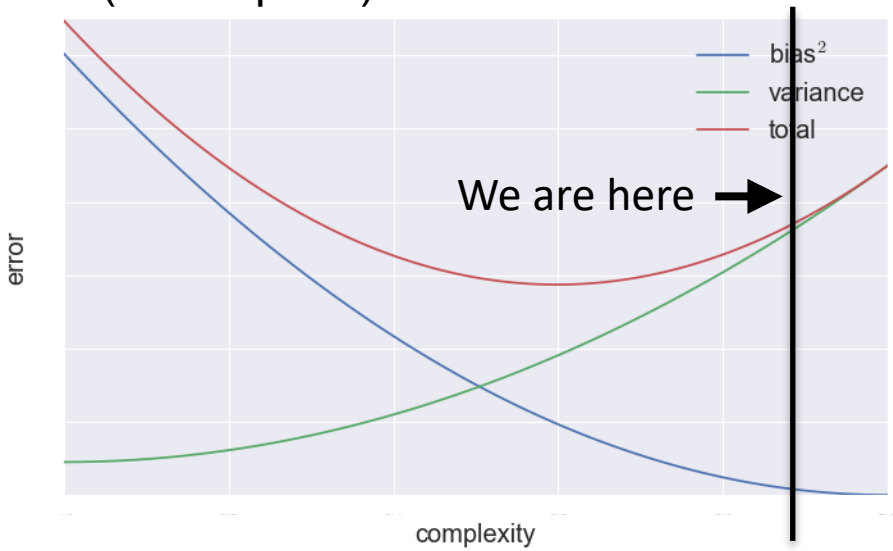
    # Low
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

    # High

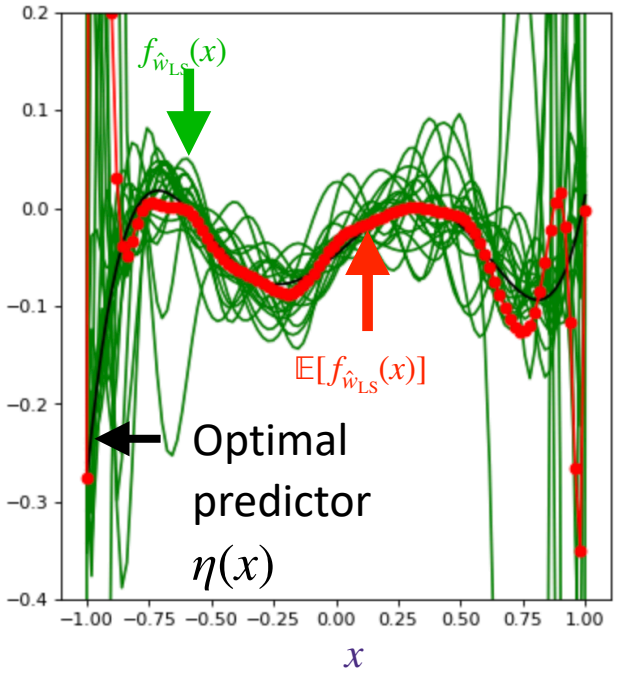
- **If we have more samples (larger n), then**
  - What happens to bias? # stays the same
  - What happens to variance? # decreases
  - What happens to overall test error?

Recap: Bias-variance tradeoff with complex model

(Conceptual) bias variance tradeoff



With degree-20 polynomials, we overfit



- When model **complexity is high** (higher than the optimal predictor $\eta(x)$)
  - Bias^2 of our predictor,
    $\left(\eta(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)]\right)^2$

    # Low
  - Variance of our predictor,
    $\mathbb{E}_{\mathcal{D}}\left[\left(\mathbb{E}_{\mathcal{D}}[\hat{f}_{\mathcal{D}}(x)] - \hat{f}_{\mathcal{D}}(x)\right)^2\right]$

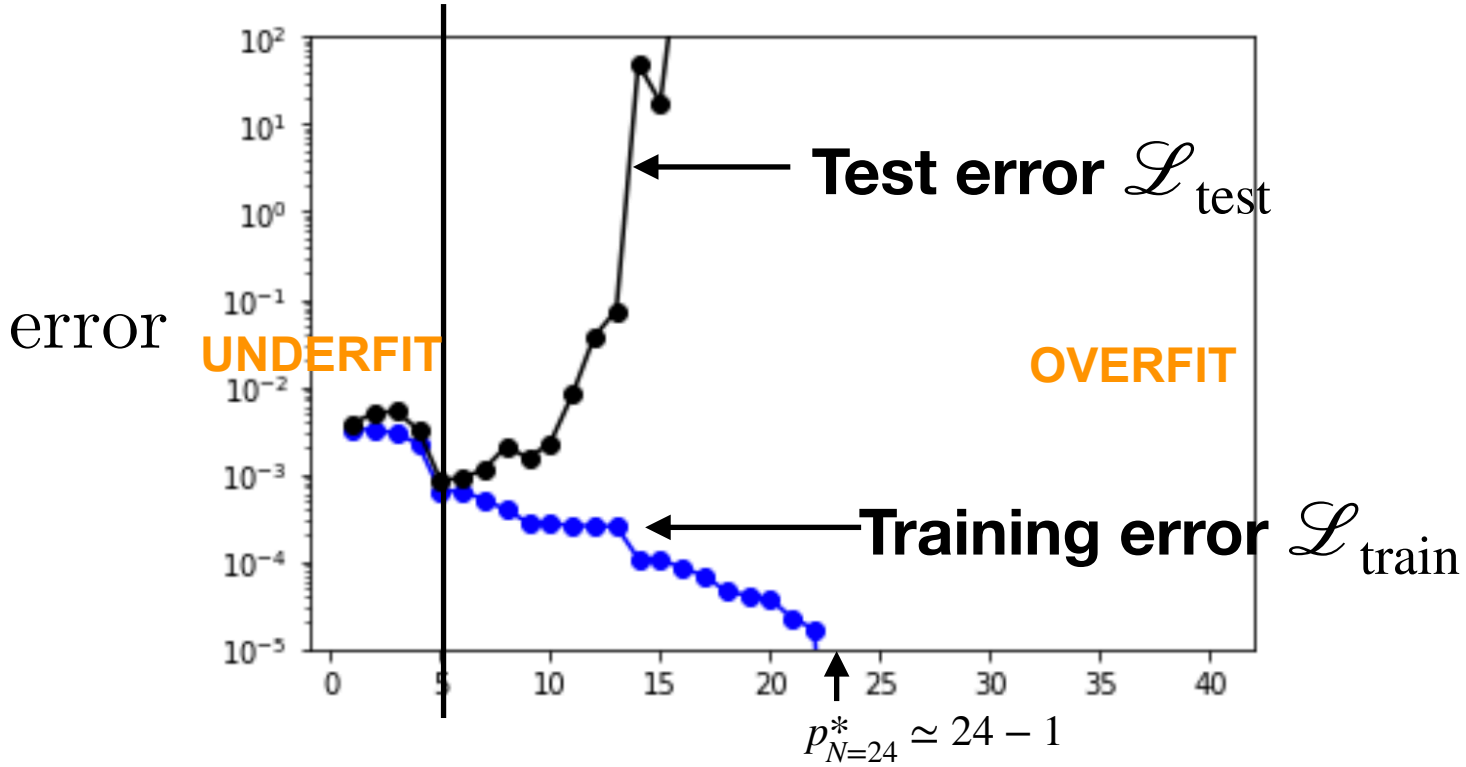
    # High

- **If we have more samples (larger n), then**
  - What happens to bias? # stays the same
  - What happens to variance? # decreases
  - What happens to overall test error?

    # decreases

Optimal model complexity depends on dataset size

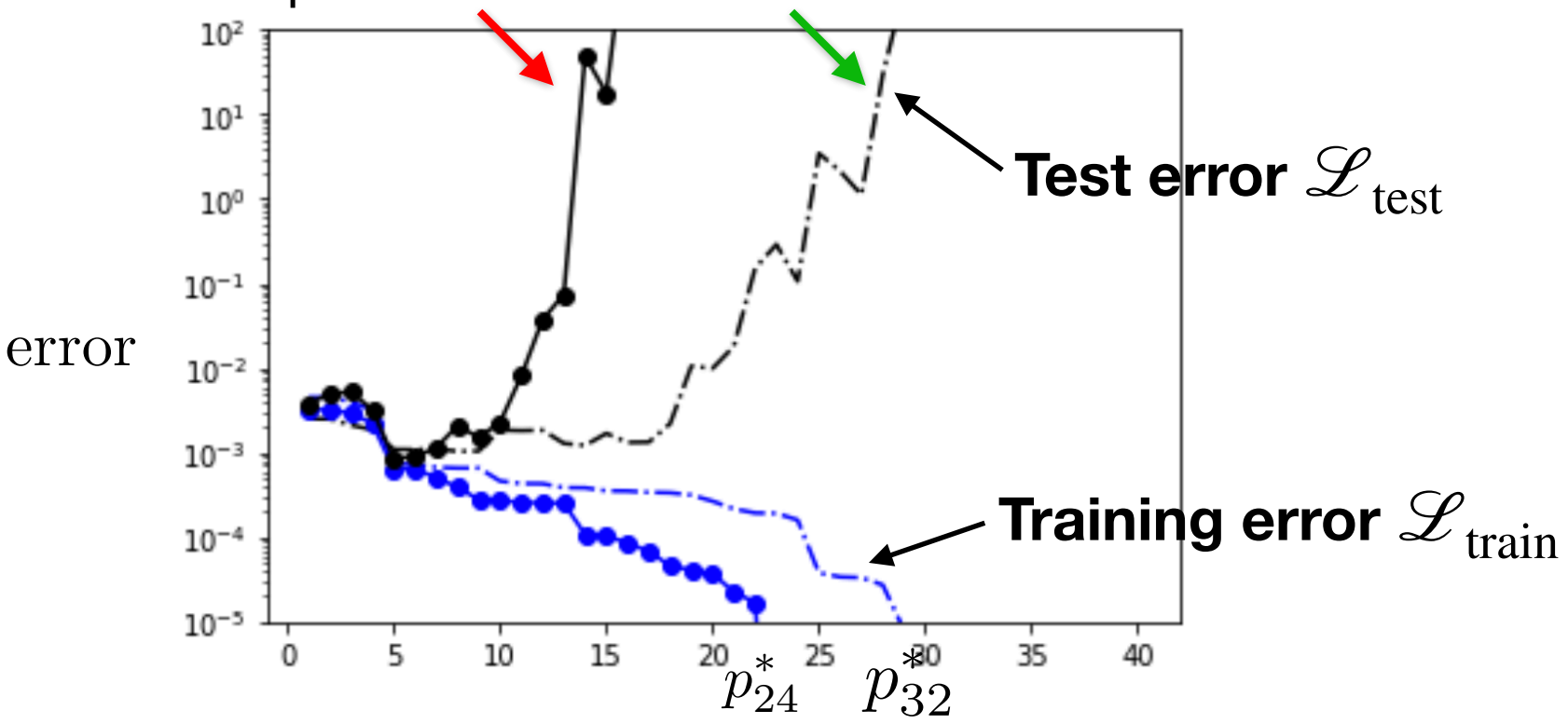
- Assume ***N=30***



- Given sample size N there is a threshold, p^*_N, where training error is zero
- Training error is **always** monotonically non-increasing
- Test error has a trend of going down and then up, but fluctuates

Variance decreases with more data, letting you fit more complex models

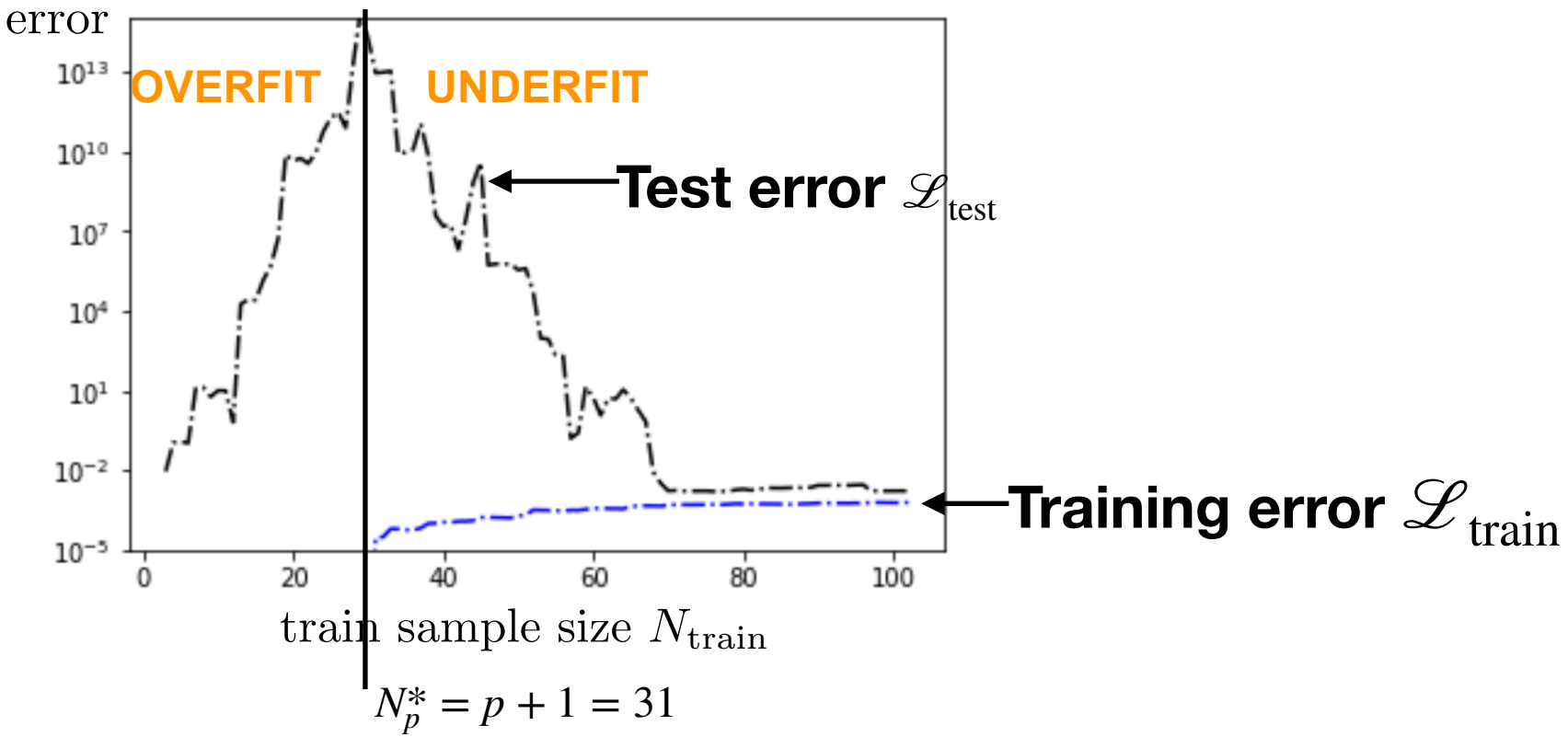
- Now compare ***N=30*** *case to* ***N=40***



- **The threshold, p_N^*, moves right as dataset size increases**
- Training error tends to increase, because more points need to fit
- Test error tends to decrease, because Variance decreases

Variance decreases with more data, letting you fit more complex models

- Choose model complexity ***p=30***, **vary dataset size n**



- There is a threshold, N_p^*, below which training error is zero (extreme overfit)
- Above the threshold, test error tends to decrease
- Training error tends to increase (harder to fit so much data with simple model)

Ridge Regression



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….

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

Regularization in Linear Regression

Recall Least Squares:

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

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

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

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:



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 *not unique* 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

Sensitivity increases overfitting

- For a linear model,
  $$y \simeq b + w_1 x_1 + w_2 x_2 + \cdots + w_d x_d$$
  if $|w_j|$ is large then the prediction is **sensitive** to small changes in x_j
- Large ***sensitivity*** leads to overfitting and poor generalization, and equivalently models that overfit tend to have large weights
- Note that b is a constant and hence there is no sensitivity for the offset b
- In ***Ridge Regression***, we use a regularizer $\|w\|_2^2$ to measure and control the sensitivity of the predictor
- And optimize for small loss and small sensitivity, by adding a ***regularizer*** in the objective (assume no offset for ~~now~~)

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

Sensitivity increases overfitting

- For a linear model,
  $$y \simeq b + w_1 x_1 + w_2 x_2 + \cdots + w_d x_d$$
  if $|w_j|$ is large then the prediction is **sensitive** to small changes in x_j
- Large ***sensitivity*** leads to overfitting and poor generalization, and equivalently models that overfit tend to have large weights
- Note that b is a constant and hence there is no sensitivity for the offset b

```
# Never regularize b
```

- In ***Ridge Regression***, we use a regularizer $\|w\|_2^2$ to measure and control the sensitivity of the predictor
- And optimize for small loss and small sensitivity, by adding a ***regularizer*** in the objective (assume no offset for now)

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

Sensitivity increases overfitting

- For a linear model,
  $$y \simeq b + w_1 x_1 + w_2 x_2 + \cdots + w_d x_d$$
  if $|w_j|$ is large then the prediction is **sensitive** to small changes in x_j
- Large ***sensitivity*** leads to overfitting and poor generalization, and equivalently models that overfit tend to have large weights
- Note that b is a constant and hence there is no sensitivity for the offset b

Never regularize b

- In ***Ridge Regression***, we use a regularizer $\|w\|_2^2$ to measure and control the sensitivity of the predictor
- And optimize for small loss and small sensitivity, by adding a ***regularizer*** in the objective (assume no offset for now)

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

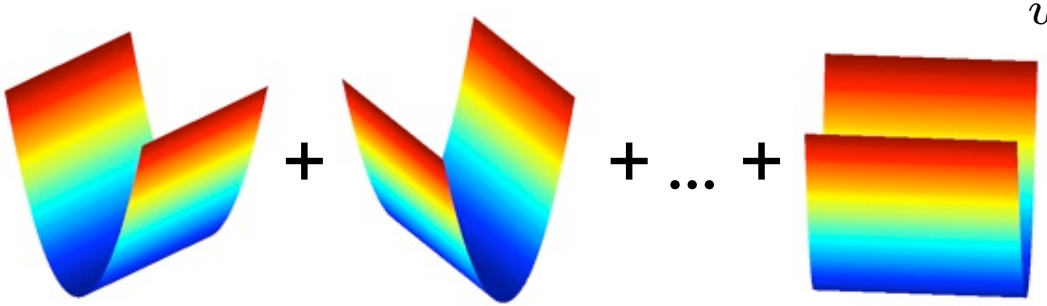
Regularize to make weights smaller / less sensitive. Multiple ways to do that. We'll start with L2 norm

Ridge Regression

if d > n

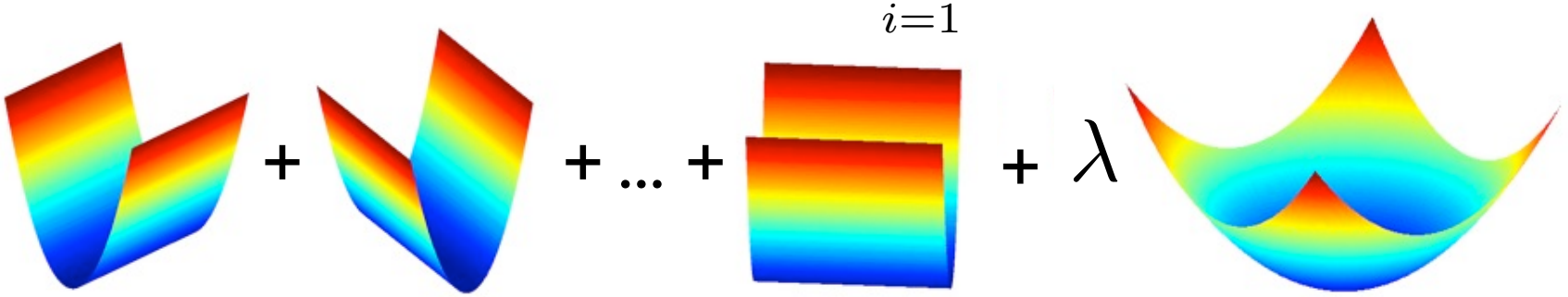
- Old Least squares objective:

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



- Ridge Regression objective:

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



Minimizing the Ridge Regression Objective

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

More identities

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{\mathrm{d}f}{\mathrm{d}x}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{\mathrm{d}f}{\mathrm{d}\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

Minimizing the Ridge Regression Objective

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

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

Minimizing the Ridge Regression Objective

$w \in \mathbb{R}^{d \times 1}$

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

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

Minimizing the Ridge Regression Objective

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

Minimizing the Ridge Regression Objective

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

More identities

$$\arg\min_{w} \left\|y - Xw\right\|_2^2 + \lambda \|w\|_2^2$$

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

Minimizing the Ridge Regression Objective

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

$$\arg\min_{w} \left\|y - Xw\right\|_2^2 + \lambda \|w\|_2^2$$

$$\arg\min_{w} (Xw - y)^T (Xw - y) + \lambda w^T w$$

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

Minimizing the Ridge Regression Objective

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

$$\arg\min_{w} \left\|y - Xw\right\|_2^2 + \lambda \|w\|_2^2$$

$$\arg\min_{w} (Xw - y)^T (Xw - y) + \lambda w^T w$$

From previous derivation

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{\mathrm{d}f}{\mathrm{d}x}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{\mathrm{d}f}{\mathrm{d}\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

Minimizing the Ridge Regression Objective

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

$$\arg\min_{w} \left\|y - Xw\right\|_2^2 + \lambda \|w\|_2^2$$

$$\arg\min_{w} (Xw - y)^T (Xw - y) + \lambda w^T w$$

From previous derivation

$$\nabla_w [-2y^T Xw + w^T X^T Xw + \lambda w^T w] = 0$$

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

Minimizing the Ridge Regression Objective

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

$$\arg\min_{w} \left\|y - Xw\right\|_2^2 + \lambda \|w\|_2^2$$

$$\arg\min_{w} (Xw - y)^T (Xw - y) + \lambda w^T w$$

From previous derivation

$$\nabla_w [-2y^T Xw + w^T X^T Xw + \lambda w^T w] = 0$$

$$-2X^T y + 2X^T Xw + 2\lambda w = 0$$

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

Minimizing the Ridge Regression Objective

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

$$\arg\min_{w} \left\|y - Xw\right\|_2^2 + \lambda \|w\|_2^2$$

$$\arg\min_{w} (Xw - y)^T (Xw - y) + \lambda w^T w$$

From previous derivation

$$\nabla_w [-2y^T Xw + w^T X^T Xw + \lambda w^T w] = 0$$

$$-2X^T y + 2X^T Xw + 2\lambda w = 0$$

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{Bx}$	$\rightarrow$	$2\mathbf{Bx}$

Basic identities from calculus/Matrix Cookbook

Minimizing the Ridge Regression Objective

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

$$\arg\min_{w} \left\|y - Xw\right\|_2^2 + \lambda \|w\|_2^2$$

$$\arg\min_{w} (Xw - y)^T (Xw - y) + \lambda w^T w$$

From previous derivation

$$\nabla_w [-2y^T Xw + w^T X^T Xw + \lambda w^T w] = 0$$

$$-2X^T y + 2X^T Xw + 2\lambda w = 0$$

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

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

Minimizing the Ridge Regression Objective

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

$$I^d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \ddots & \\ & & 1 \end{bmatrix}$$

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

$$\arg\min_{w} \left\|y - Xw\right\|_2^2 + \lambda \|w\|_2^2$$

$$\arg\min_{w} (Xw - y)^T (Xw - y) + \lambda w^T w$$

From previous derivation

$$\nabla_w [-2y^T Xw + w^T X^T Xw + \lambda w^T w] = 0$$

$$-2X^T y + 2X^T Xw + 2\lambda w = 0$$

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

Minimizing the Ridge Regression Objective

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

$$I^d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \ddots & \\ & & 1 \end{bmatrix}$$

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

$$\arg\min_{w} \left\|y - Xw\right\|_2^2 + \lambda \|w\|_2^2$$

$$\arg\min_{w} (Xw - y)^T (Xw - y) + \lambda w^T w$$

From previous derivation

$$\nabla_w [-2y^T Xw + w^T X^T Xw + \lambda w^T w] = 0$$

$$-2X^T y + 2X^T Xw + 2\lambda w = 0$$

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

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

Minimizing the Ridge Regression Objective

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

$$I^d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \ddots & \\ & & 1 \end{bmatrix}$$

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

$$\arg\min_{w} \left\|y - Xw\right\|_2^2 + \lambda \|w\|_2^2$$

$$\arg\min_{w} (Xw - y)^T (Xw - y) + \lambda w^T w$$

From previous derivation

$$\nabla_w [-2y^T Xw + w^T X^T Xw + \lambda w^T w] = 0$$

$$-2X^T y + 2X^T Xw + 2\lambda w = 0$$

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

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

$$\hat{w}_{\text{ridge}} = (X^T X + \lambda I)^{-1} \cdot X^T y$$

Minimizing the Ridge Regression Objective

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

More identities

$$\|w\|_p = (|w_1|^p + \cdots + |w_d|^p)^{\frac{1}{p}}$$

$$I^d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \ddots & \\ & & 1 \end{bmatrix}$$

Scalar derivative			Vector derivative		
$f(x)$	$\rightarrow$	$\frac{df}{dx}$	$f(\mathbf{x})$	$\rightarrow$	$\frac{df}{d\mathbf{x}}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{B}$	$\rightarrow$	$\mathbf{B}$
bx	$\rightarrow$	b	$\mathbf{x}^T\mathbf{b}$	$\rightarrow$	$\mathbf{b}$
x^2	$\rightarrow$	$2x$	$\mathbf{x}^T\mathbf{x}$	$\rightarrow$	$2\mathbf{x}$
bx^2	$\rightarrow$	$2bx$	$\mathbf{x}^T\mathbf{B}\mathbf{x}$	$\rightarrow$	$2\mathbf{B}\mathbf{x}$

Basic identities from calculus/Matrix Cookbook

$$\arg\min_{w} \left\|y - Xw\right\|_2^2 + \lambda \|w\|_2^2$$

$$\arg\min_{w} (Xw - y)^T (Xw - y) + \lambda w^T w$$

From previous derivation

$$\nabla_w [-2y^T Xw + w^T X^T Xw + \lambda w^T w] = 0$$

$$-2X^T y + 2X^T Xw + 2\lambda w = 0$$

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

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

$$\hat{w}_{\text{ridge}} = (X^T X + \lambda I)^{-1} \cdot X^T y$$

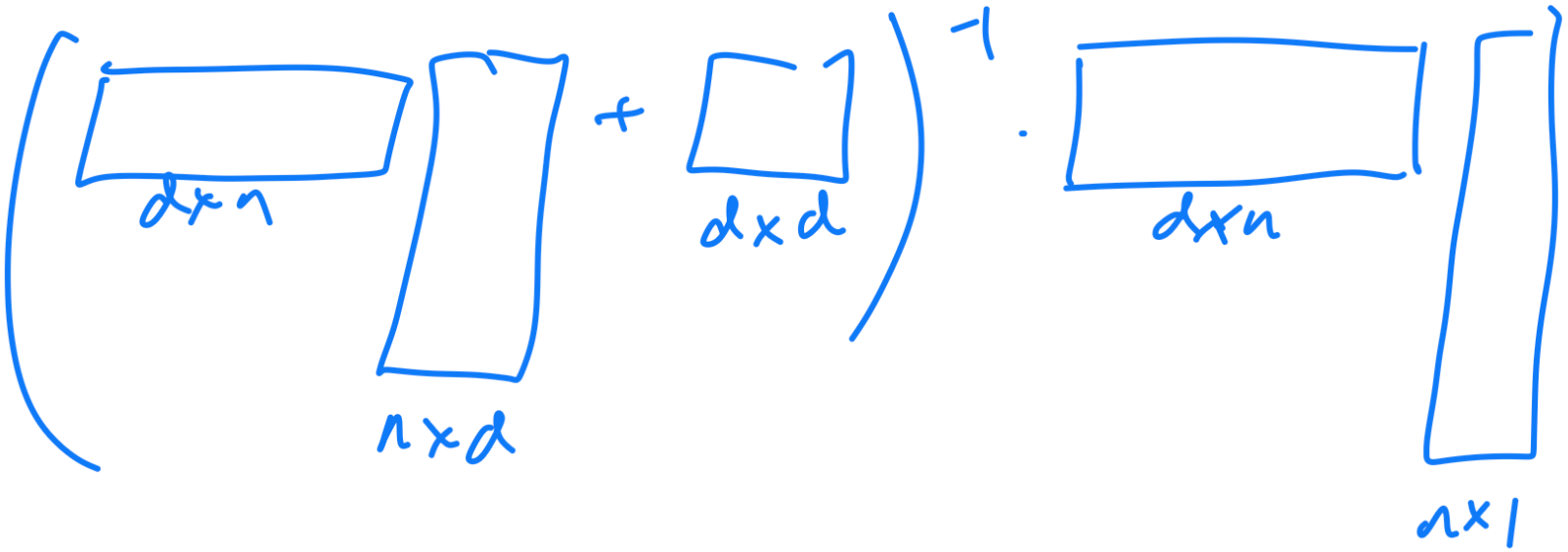
Shrinkage Properties

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

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

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



- When $\lambda = 0$, this gives the least squares model
- This defines a family of models hyper-parametrized by λ
- Large λ means more regularization and simpler model
- Small λ means less regularization and more complex model

Shrinkage Properties

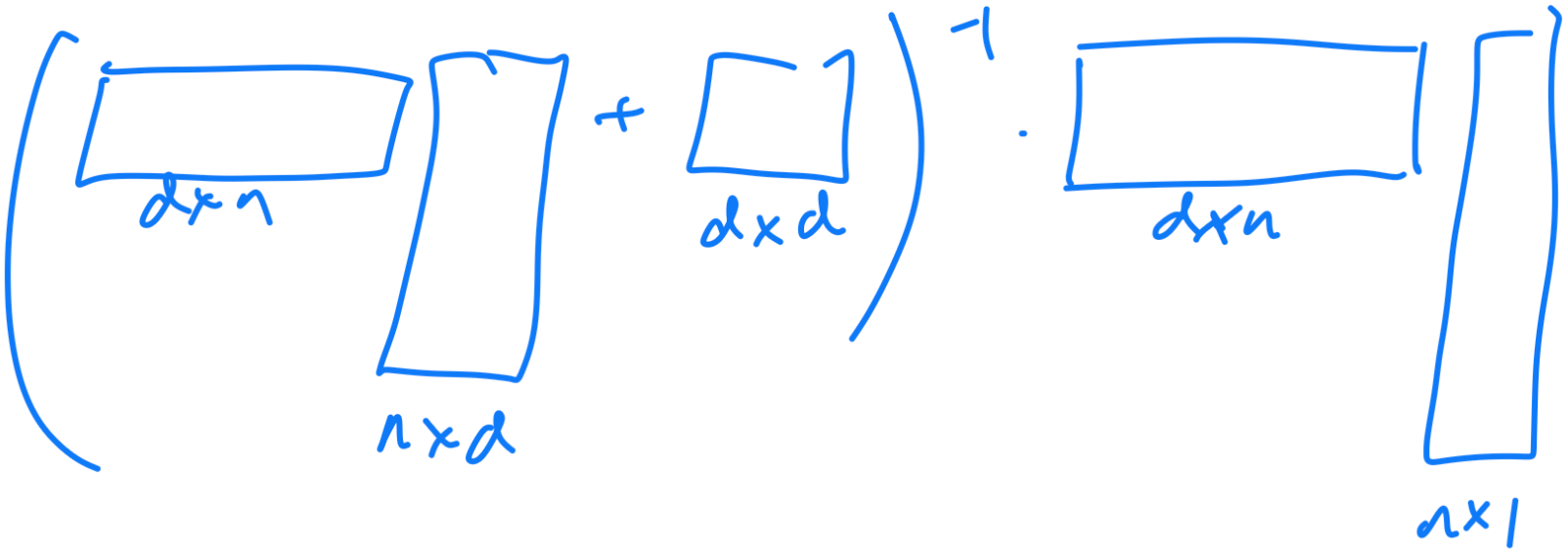
$$\widehat{w}_{ridge} = \arg\min_{w} \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 + \lambda ||w||_2^2$$

$w \in \mathbb{R}^{d \times 1}$

$X \in \mathbb{R}^{n \times d}$

What problem does this solve?

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



- When $\lambda = 0$, this gives the least squares model
- This defines a family of models hyper-parametrized by λ
- Large λ means more regularization and simpler model
- Small λ means less regularization and more complex model

Shrinkage Properties

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

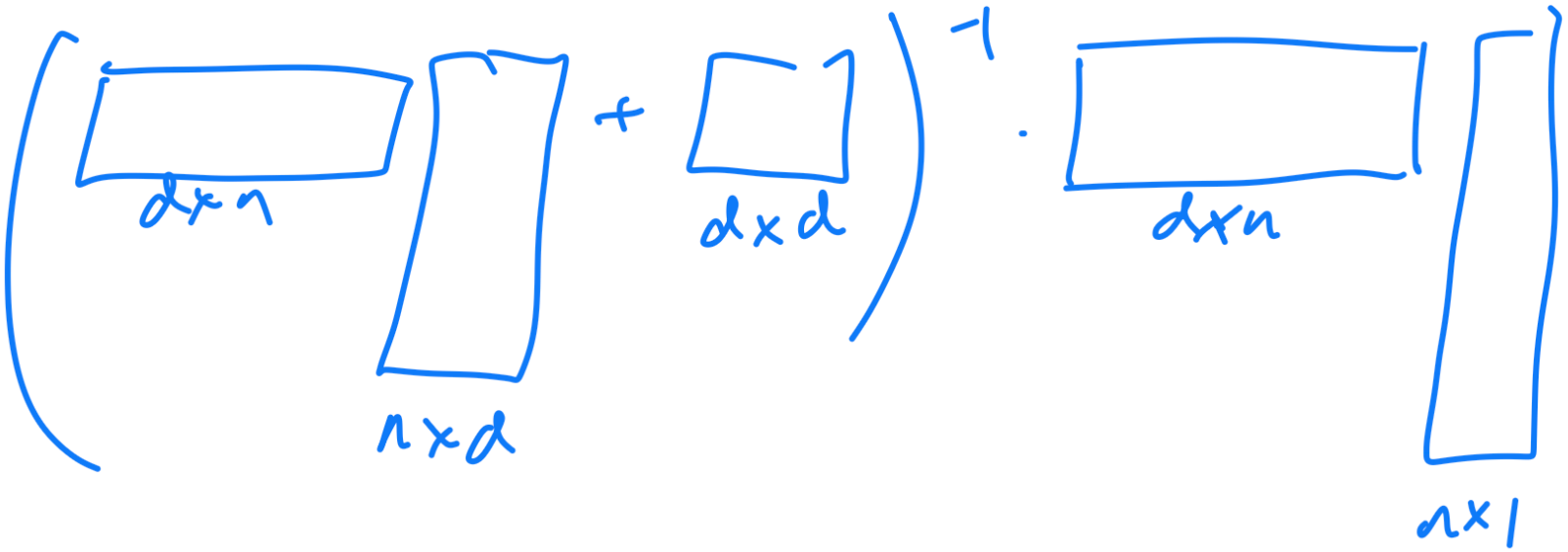
$$= (\mathbf{X}^T\mathbf{X} + \lambda I)^{-1}\mathbf{X}^T\mathbf{y}$$

$w \in \mathbb{R}^{d\times 1}$

$X \in \mathbb{R}^{n\times d}$

What problem does this solve?

Matrix is always invertible



- When $\lambda = 0$, this gives the least squares model
- This defines a family of models hyper-parametrized by λ
- Large λ means more regularization and simpler model
- Small λ means less regularization and more complex model

Shrinkage Properties

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

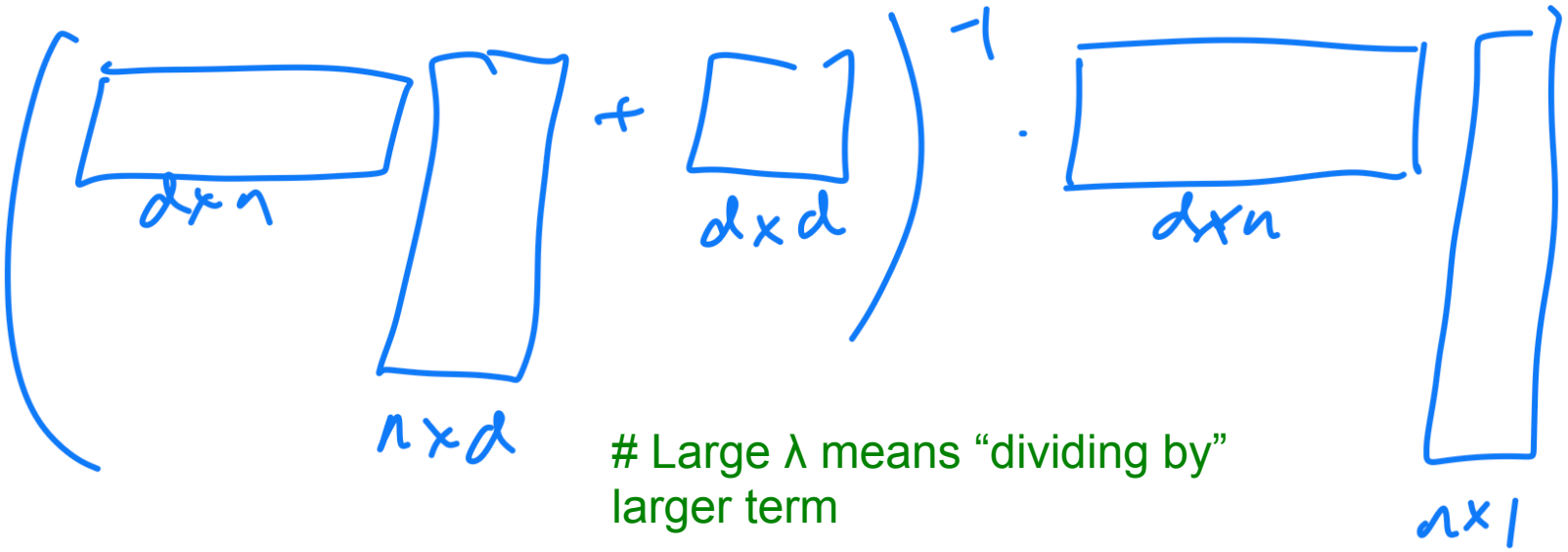
$$= (\mathbf{X}^T\mathbf{X} + \lambda I)^{-1}\mathbf{X}^T\mathbf{y}$$

$w \in \mathbb{R}^{d\times 1}$

$X \in \mathbb{R}^{n\times d}$

What problem does this solve?

Matrix is always invertible



Large λ means "dividing by" larger term

- When $\lambda = 0$, this gives the least squares model
- This defines a family of models hyper-parametrized by λ
- Large λ means more regularization and simpler model
- Small λ means less regularization and more complex model

Shrinkage Properties

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

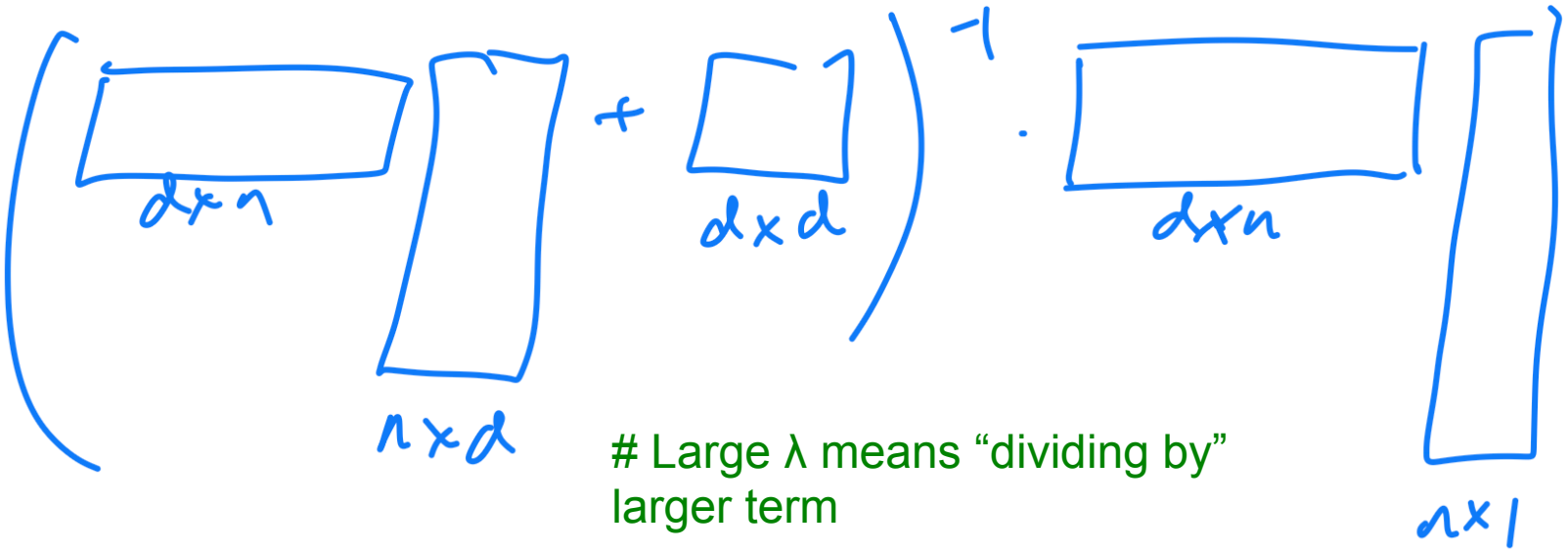
$$= (\mathbf{X}^T\mathbf{X} + \lambda I)^{-1}\mathbf{X}^T\mathbf{y}$$

$w \in \mathbb{R}^{d\times 1}$

$X \in \mathbb{R}^{n\times d}$

What problem does this solve?

Matrix is always invertible



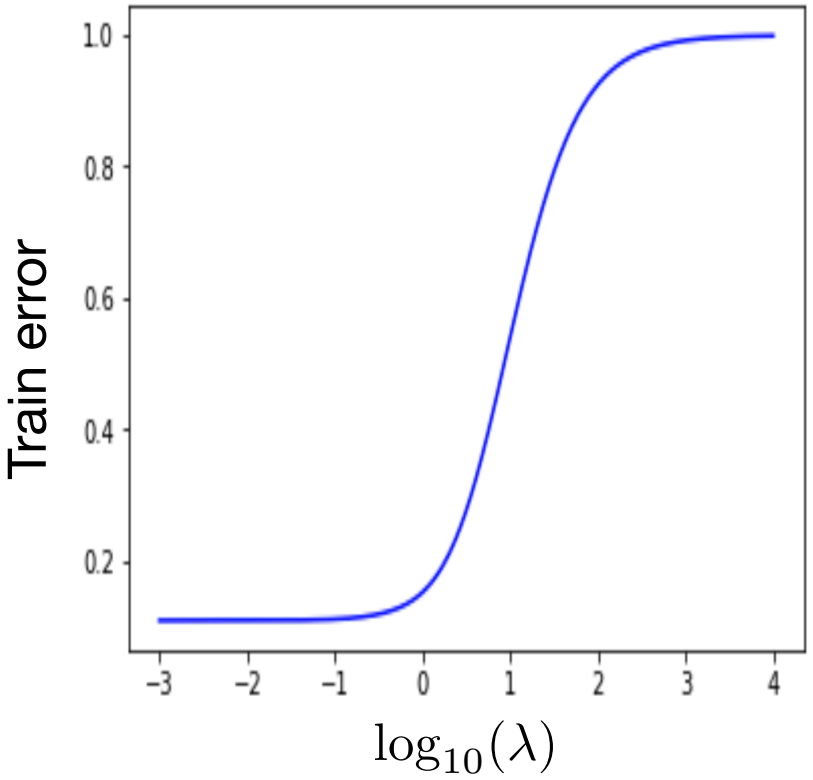
Large λ means "dividing by" larger term

- When $\lambda = 0$, this gives the least squares model
- This defines a family of models hyper-parametrized by λ
- Large λ means more regularization and simpler model
- Small λ means less regularization and more complex model

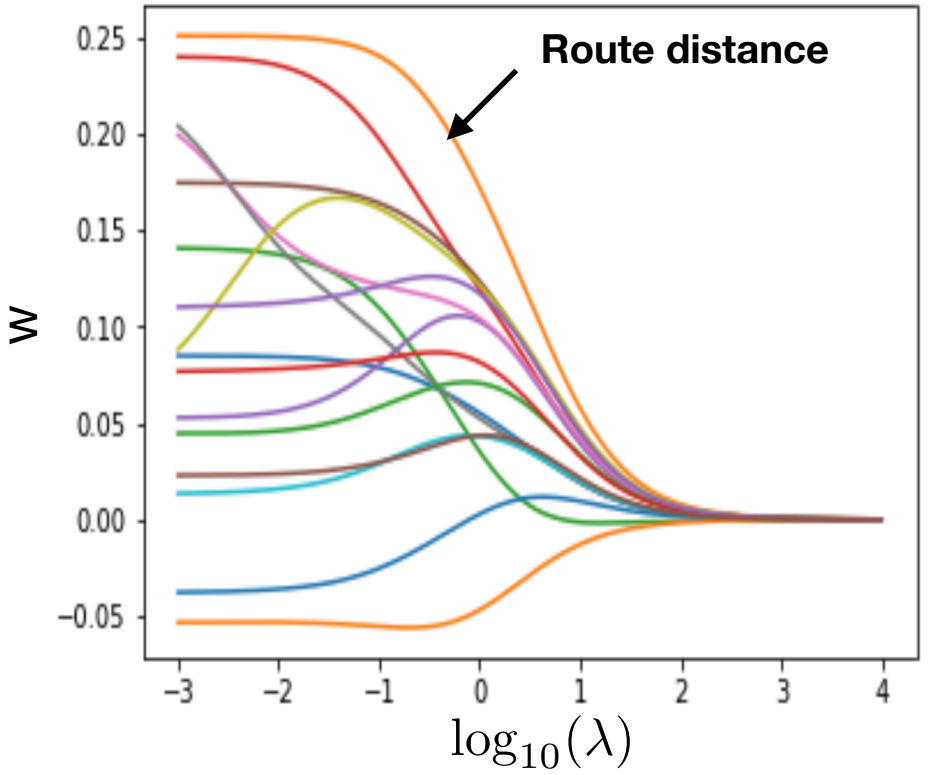
Optimal λ depends on magnitude of features

Ridge regression: minimize $\sum_{i=1}^{n} (w^T x_i - y_i)^2 + \lambda \|w\|_2^2$

training MSE $\frac{1}{n} \sum_{i=1}^{n} (y_i - x_i^T \hat{w}_{\text{ridge}}^{(\lambda)})^2$



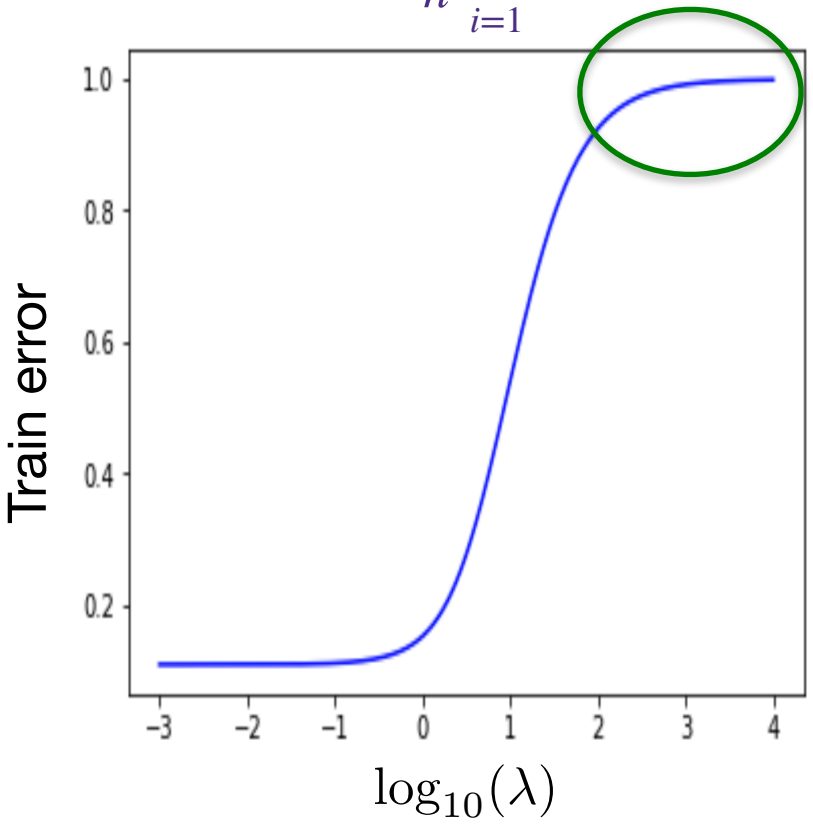
Illustrative potato delivery: w_i



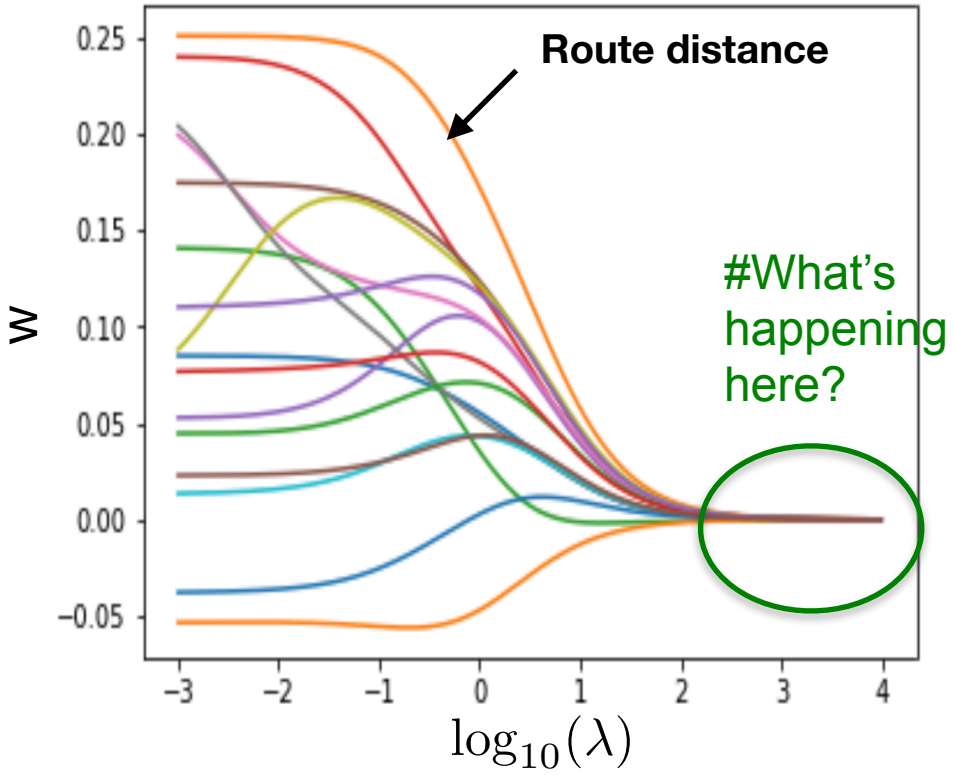
Right plot: called **regularization path**

Ridge regression: minimize $\sum_{i=1}^{n} (w^T x_i - y_i)^2 + \lambda \|w\|_2^2$

training MSE $\frac{1}{n} \sum_{i=1}^{n} (y_i - x_i^T \hat{w}_{\text{ridge}}^{(\lambda)})^2$



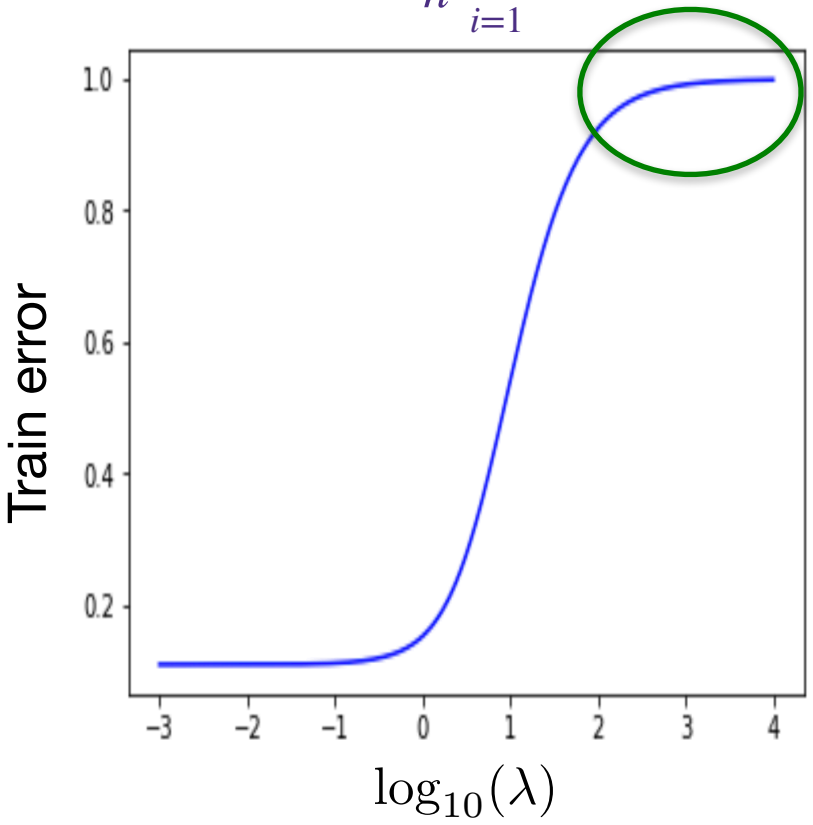
Illustrative potato delivery: w_i



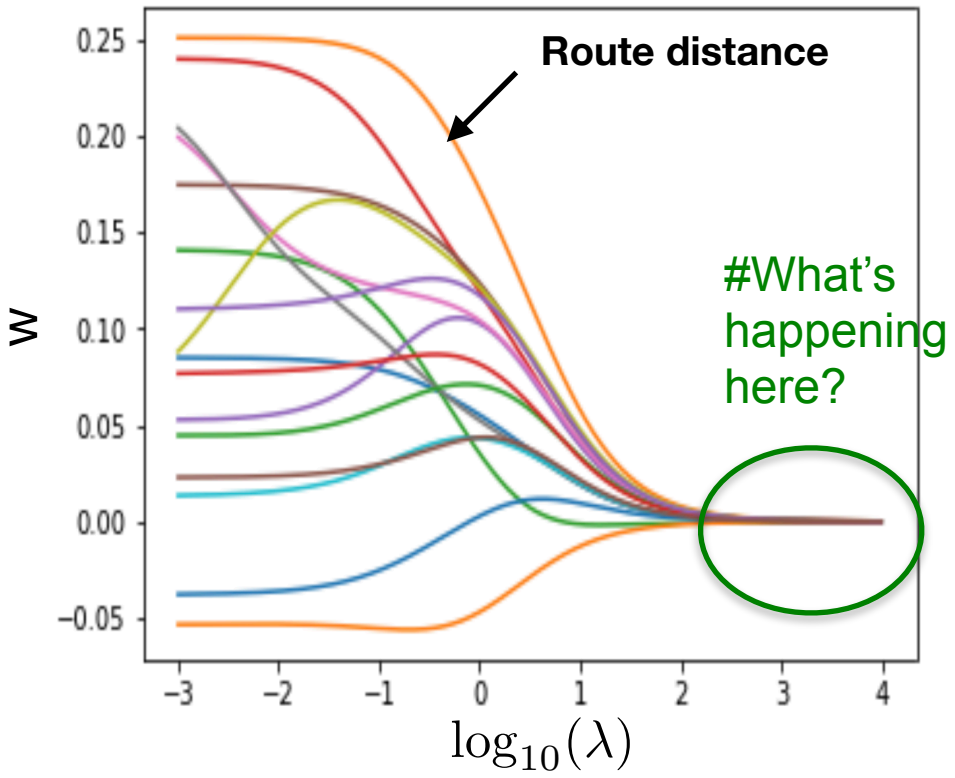
Right plot: called **regularization path**

Ridge regression: minimize $\sum_{i=1}^{n} (w^T x_i - y_i)^2 + \lambda \|w\|_2^2$

training MSE $\frac{1}{n} \sum_{i=1}^{n} (y_i - x_i^T \hat{w}_{\text{ridge}}^{(\lambda)})^2$



Illustrative potato delivery: w_i

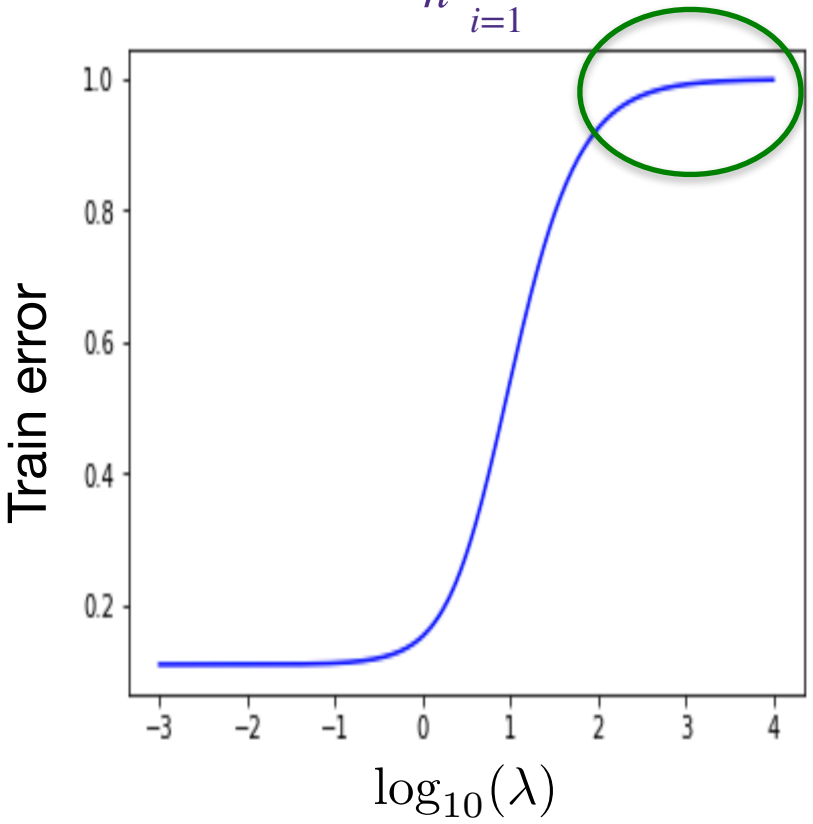


Right plot: called **regularization path**

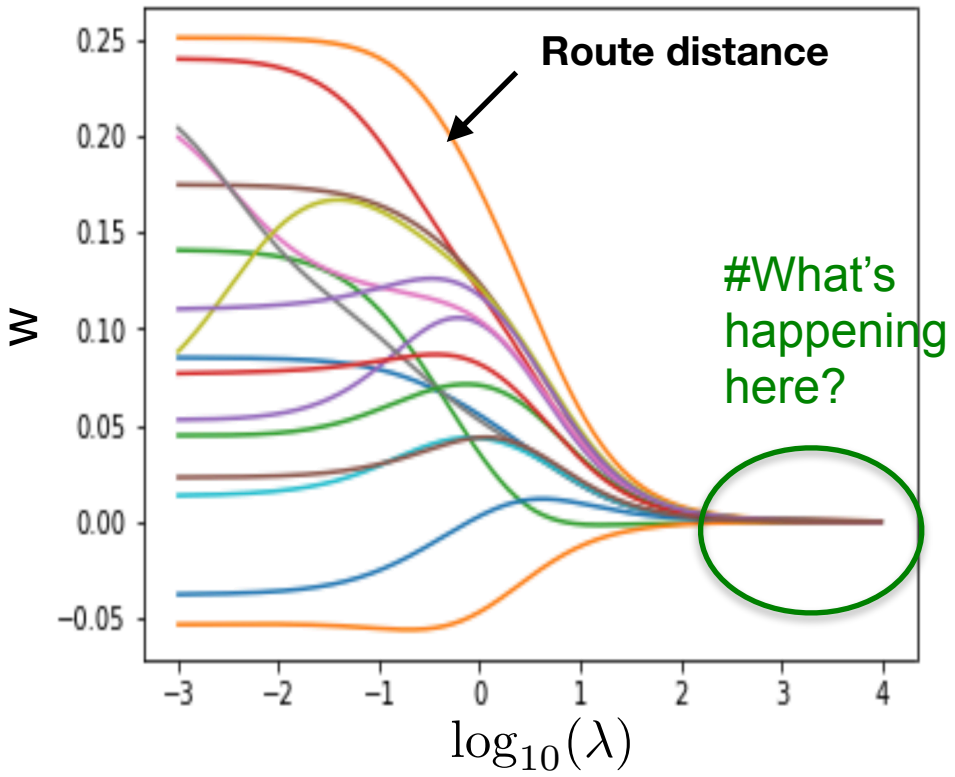
Too much regularization introduces bias / underfitting

Ridge regression: minimize $\sum_{i=1}^{n} (w^T x_i - y_i)^2 + \lambda \|w\|_2^2$

training MSE $\frac{1}{n} \sum_{i=1}^{n} (y_i - x_i^T \hat{w}_{\text{ridge}}^{(\lambda)})^2$



Illustrative potato delivery: w_i

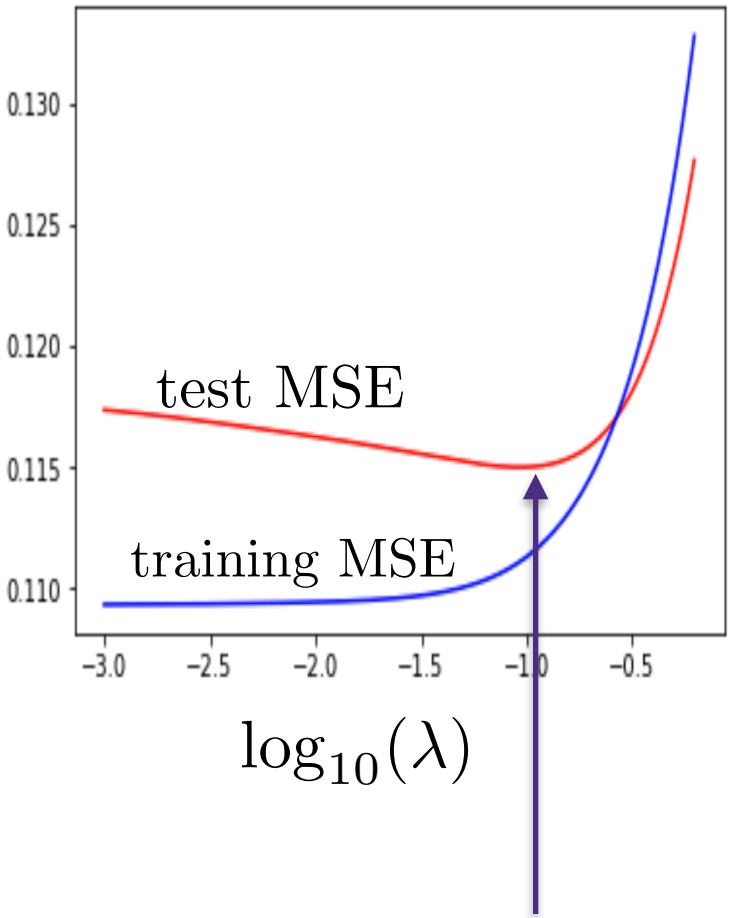


Right plot: called **regularization path**

Too much regularization introduces bias / underfitting

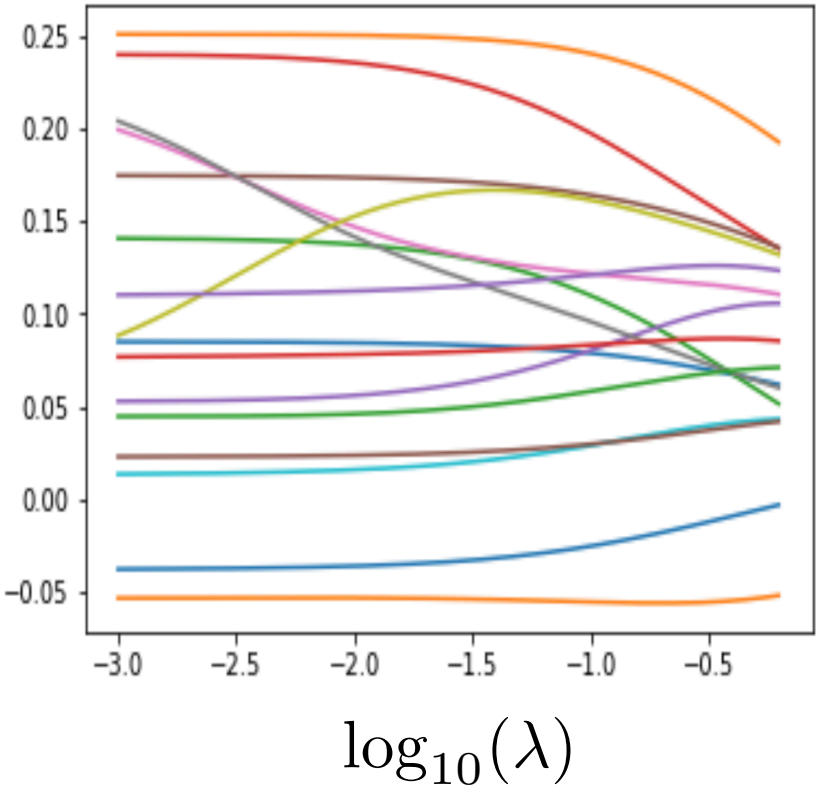
Route-distance coefficient shrinks slowly in this example

Ridge regression: $\text{minimize} \sum_{i=1}^{n} (w^T x_i - y_i)^2 + \lambda \|w\|_2^2$



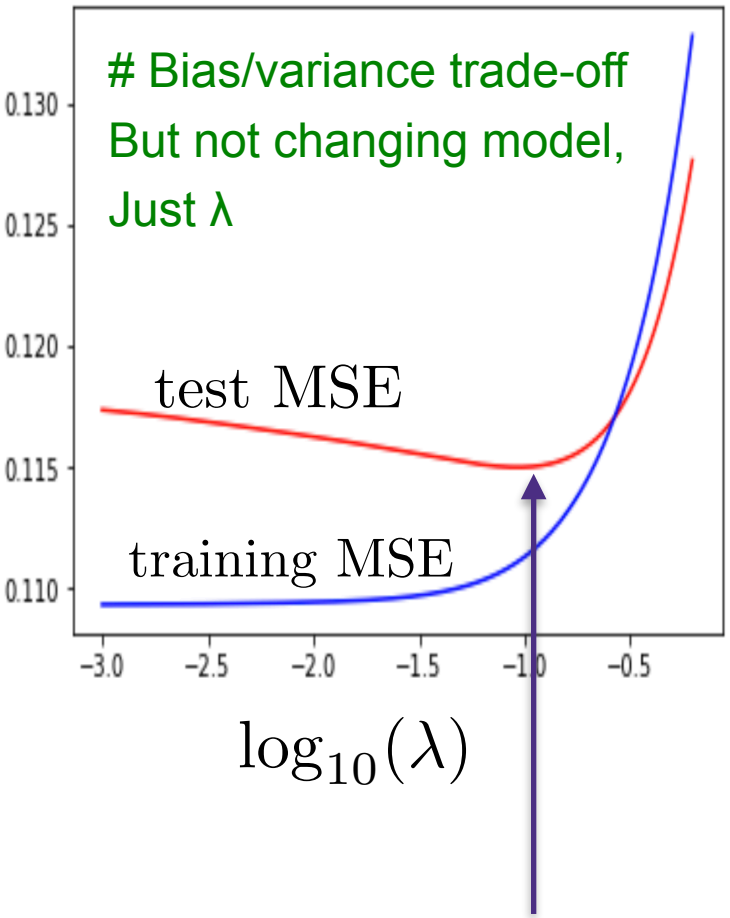
- this gain in test MSE comes from shrinking w's to get a less sensitive predictor
(which in turn reduces the variance)

Illustrative potato delivery: w_i

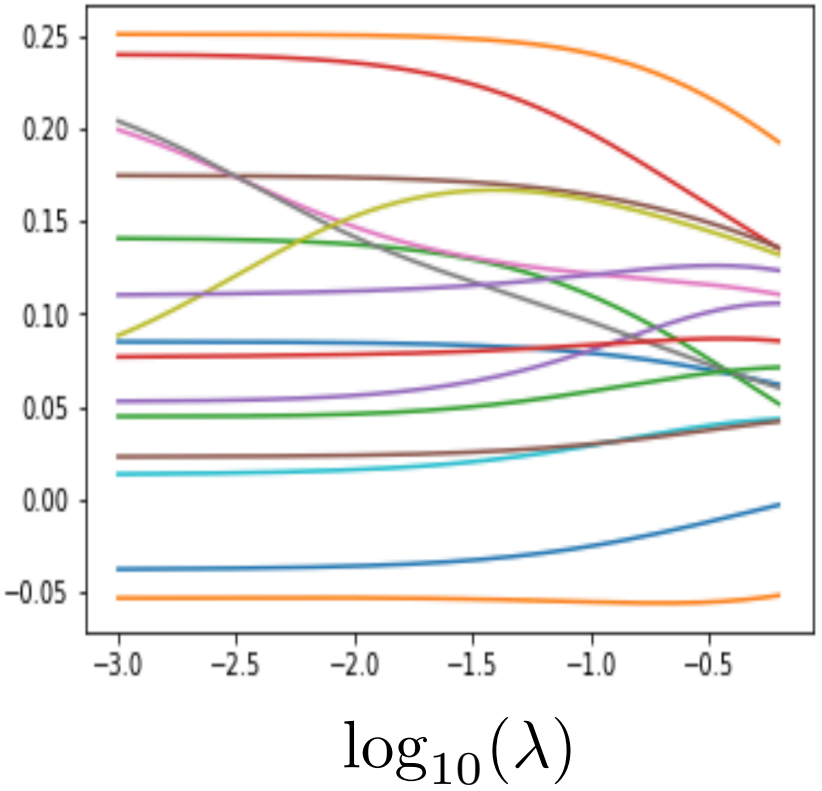


- this is the role of regularizer

Ridge regression: $\text{minimize} \sum_{i=1}^{n} (w^T x_i - y_i)^2 + \lambda \|w\|_2^2$



Illustrative potato delivery: w_i



- this gain in test MSE comes from shrinking w's to get a less sensitive predictor
  (which in turn reduces the variance)

- this is the role of regularizer

Bias-Variance Properties

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \;\; \mathbf{y} = \mathbf{X}w + \epsilon, \;\; \epsilon \sim \mathcal{N}(0,\sigma^2\mathbf{I})$ for some ground truth model parameter w
- The true error at a sample with feature x is $\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$

Bias-Variance Properties

Plan: work through an example where we can directly put bias/ variance in terms of λ

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \ \mathbf{y} = \mathbf{X}w + \epsilon, \ \epsilon \sim \mathcal{N}(0,\sigma^2\mathbf{I})$ for some ground truth model parameter w
- The true error at a sample with feature x is $\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$

Bias-Variance Properties

Plan: work through an example where we can directly put bias/ variance in terms of λ

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \ \mathbf{y} = \mathbf{X}w + \epsilon, \ \epsilon \sim \mathcal{N}(0,\sigma^2\mathbf{I})$ for some ground truth model parameter w
- The true error at a sample with feature x is $\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$

x is a test sample, $\hat{w}_{\text{ridge}}$ is fit to a dataset

Bias-Variance Properties

Plan: work through an example where we can directly put bias/ variance in terms of λ

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \ \mathbf{y} = \mathbf{X}w + \epsilon, \ \epsilon \sim \mathcal{N}(0,\sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

x is a test sample, $\hat{w}_{\text{ridge}}$ is fit to a dataset

$$= \underbrace{\mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2 \,|\, x]}_{\text{Irreducible Error}} + \underbrace{\mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]}_{\text{Learning Error}}$$

Bias-Variance Properties

Plan: work through an example where we can directly put bias/ variance in terms of λ

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X,\ \ \mathbf{y} = \mathbf{X}w + \epsilon,\ \ \epsilon \sim \mathcal{N}(0,\sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

x is a test sample, $\hat{w}_{\text{ridge}}$ is fit to a dataset

$$= \mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

$$= \mathbb{E}_{y|x}[(y - x^Tw)^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(x^Tw - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

Bias-Variance Properties

Plan: work through an example where we can directly put bias/ variance in terms of λ

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X,\ \ \mathbf{y} = \mathbf{X}w + \epsilon,\ \ \epsilon \sim \mathcal{N}(0,\sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2\,|\,x]$$ # x is a test sample, $\hat{w}_{\text{ridge}}$ is fit to a dataset

$$= \mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2\,|\,x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2\,|\,x]$$

$$= \mathbb{E}_{y|x}[(y - x^Tw)^2\,|\,x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(x^Tw - x^T\hat{w}_{\text{ridge}})^2\,|\,x]$$ # true w

Bias-Variance Properties

Plan: work through an example where we can directly put bias/ variance in terms of λ

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \;\; \mathbf{y} = \mathbf{X}w + \epsilon, \;\; \epsilon \sim \mathcal{N}(0,\sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$ # x is a test sample, $\hat{w}_{\text{ridge}}$ is fit to a dataset

$$= \mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

$$= \mathbb{E}_{y|x}[(y - x^Tw)^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(x^Tw - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$ # true w

$$= \sigma^2 + (x^Tw - \mathbb{E}_{\tilde{\mathcal{D}}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x])^2 + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}_{\tilde{\mathcal{D}}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

Irreduc. Error — Bias-squared — Variance

Bias-Variance Properties

Plan: work through an example where we can directly put bias/ variance in terms of λ

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \ \mathbf{y} = \mathbf{X}w + \epsilon, \ \epsilon \sim \mathcal{N}(0,\sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$ # x is a test sample, $\hat{w}_{\text{ridge}}$ is fit to a dataset

$$= \mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

$$= \mathbb{E}_{y|x}[(y - x^Tw)^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(x^Tw - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$ # true w

$$= \sigma^2 + (x^Tw - \mathbb{E}_{\mathcal{D}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x])^2 + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}_{\tilde{\mathcal{D}}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

Irreduc. Error — Bias-squared — Variance

Suppose $\mathbf{X}^T\mathbf{X} = n\mathbf{I}$, then $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T(\mathbf{X}w + \epsilon)$

$$= \frac{n}{n+\lambda}w + \frac{1}{n+\lambda}\mathbf{X}^T\epsilon$$

Bias-Variance Properties

Plan: work through an example where we can directly put bias/ variance in terms of λ

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \;\; \mathbf{y} = \mathbf{X}w + \epsilon, \;\; \epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$ # x is a test sample, $\hat{w}_{\text{ridge}}$ is fit to a dataset

$$= \mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

$$= \mathbb{E}_{y|x}[(y - x^Tw)^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(x^Tw - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$ # true w

$$= \sigma^2 + (x^Tw - \mathbb{E}_{\mathcal{D}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x])^2 + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}_{\tilde{\mathcal{D}}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

Irreduc. Error — Bias-squared — Variance

Suppose $\mathbf{X}^T\mathbf{X} = n\mathbf{I}$, then $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T(\mathbf{X}w + \epsilon)$

Not realistic but pretend we're sampling independent Gaussians

$$= \frac{n}{n+\lambda}w + \frac{1}{n+\lambda}\mathbf{X}^T\epsilon$$

Bias-Variance Properties

Plan: work through an example where we can directly put bias/ variance in terms of λ

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \;\; \mathbf{y} = \mathbf{X}w + \epsilon, \;\; \epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

x is a test sample, $\hat{w}_{\text{ridge}}$ is fit to a dataset

$$= \mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

$$= \mathbb{E}_{y|x}[(y - x^Tw)^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(x^Tw - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

true w

$$= \sigma^2 + (x^Tw - \mathbb{E}_{\mathcal{D}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x])^2 + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}_{\tilde{\mathcal{D}}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

Irreduc. Error — Bias-squared — Variance

nI

Suppose $\mathbf{X}^T\mathbf{X} = n\mathbf{I}$, then $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T(\mathbf{X}w + \epsilon)$

Not realistic but pretend we're sampling independent Gaussians

$$= \frac{n}{n+\lambda}w + \frac{1}{n+\lambda}\mathbf{X}^T\epsilon$$

Bias-Variance Properties

Plan: work through an example where we can directly put bias/ variance in terms of λ

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \;\; \mathbf{y} = \mathbf{X}w + \epsilon, \;\; \epsilon \sim \mathcal{N}(0,\sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

x is a test sample, $\hat{w}_{\text{ridge}}$ is fit to a dataset

$$= \mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

$$= \mathbb{E}_{y|x}[(y - x^Tw)^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(x^Tw - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

true w

$$= \sigma^2 + (x^Tw - \mathbb{E}_{\mathcal{D}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x])^2 + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}_{\tilde{\mathcal{D}}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

Irreduc. Error — Bias-squared — Variance

nI

Suppose $\mathbf{X}^T\mathbf{X} = n\mathbf{I}$, then $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T(\mathbf{X}w + \epsilon)$

$$= \frac{n}{n+\lambda}w + \frac{1}{n+\lambda}\mathbf{X}^T\epsilon$$

Not realistic but pretend we're sampling independent Gaussians

λ trades off weight on data vs. noise

Bias-Variance Properties

Suppose $\mathbf{X}^T\mathbf{X} = n\mathbf{I}$, then

$$\hat{w}_{\text{ridge}} = \frac{n}{n+\lambda} w + \frac{1}{n+\lambda}\mathbf{X}^T\epsilon$$

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \;\; \mathbf{y} = \mathbf{X}w + \epsilon, \;\; \epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\begin{aligned}
&\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x] \\
&\quad = \mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x] \\
&\quad = \mathbb{E}_{y|x}[(y - x^Tw)^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(x^Tw - x^T\hat{w}_{\text{ridge}})^2 \,|\, x] \\
&= \sigma^2 + (x^Tw - \mathbb{E}_{\mathcal{D}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x])^2 + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}_{\tilde{\mathcal{D}}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]
\end{aligned}$$

(verify at home)

$$= \sigma^2 + \frac{\lambda^2}{(n+\lambda)^2}(w^Tx)^2 + \frac{\sigma^2 n}{(n+\lambda)^2}\|x\|_2^2$$

Irreduc. Error | Bias-squared | Variance

Bias-Variance Properties

Suppose $\mathbf{X}^T\mathbf{X} = n\mathbf{I}$, then

$$\hat{w}_{\text{ridge}} = \frac{n}{n+\lambda} w + \frac{1}{n+\lambda}\mathbf{X}^T\epsilon$$

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \;\; \mathbf{y} = \mathbf{X}w + \epsilon, \;\; \epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\begin{aligned}
&\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x] \\
&\quad = \mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x] \\
&\quad = \mathbb{E}_{y|x}[(y - x^Tw)^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(x^Tw - x^T\hat{w}_{\text{ridge}})^2 \,|\, x] \\
&= \sigma^2 + (x^Tw - \mathbb{E}_{\mathcal{D}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x])^2 + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}_{\tilde{\mathcal{D}}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]
\end{aligned}$$

(verify at home)

$$= \sigma^2 + \frac{\lambda^2}{(n+\lambda)^2}(w^Tx)^2 + \frac{\sigma^2 n}{(n+\lambda)^2}\|x\|_2^2$$

Irreduc. Error | Bias-squared | Variance

- λ trades off bias + variance

Bias-Variance Properties

Suppose $\mathbf{X}^T\mathbf{X} = n\mathbf{I}$, then

$$\hat{w}_{\text{ridge}} = \frac{n}{n+\lambda} w + \frac{1}{n+\lambda} \mathbf{X}^T \epsilon$$

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \;\; \mathbf{y} = \mathbf{X}w + \epsilon, \;\; \epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

$$= \mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

$$= \mathbb{E}_{y|x}[(y - x^Tw)^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(x^Tw - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

$$= \sigma^2 + (x^Tw - \mathbb{E}_{\mathcal{D}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x])^2 + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}_{\tilde{\mathcal{D}}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

(verify at home)

$$= \sigma^2 + \frac{\lambda^2}{(n+\lambda)^2}(w^Tx)^2 + \frac{\sigma^2 n}{(n+\lambda)^2}\|x\|_2^2$$

Irreduc. Error | Bias-squared | Variance

- λ trades off bias + variance
- Larger λ →

Bias-Variance Properties

Suppose $\mathbf{X}^T\mathbf{X} = n\mathbf{I}$, then

$$\hat{w}_{\text{ridge}} = \frac{n}{n+\lambda} w + \frac{1}{n+\lambda} \mathbf{X}^T \epsilon$$

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \ \mathbf{y} = \mathbf{X}w + \epsilon, \ \epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\begin{aligned}
&\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x] \\
&\quad = \mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x] \\
&\quad = \mathbb{E}_{y|x}[(y - x^Tw)^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(x^Tw - x^T\hat{w}_{\text{ridge}})^2 \,|\, x] \\
&= \sigma^2 + (x^Tw - \mathbb{E}_{\mathcal{D}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x])^2 + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}_{\tilde{\mathcal{D}}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]
\end{aligned}$$

(verify at home)

$$= \sigma^2 + \frac{\lambda^2}{(n+\lambda)^2}(w^Tx)^2 + \frac{\sigma^2 n}{(n+\lambda)^2}\|x\|_2^2$$

Irreduc. Error | Bias-squared | Variance

- λ trades off bias + variance
- Larger λ →
  - Bigger bias

Bias-Variance Properties

Suppose $\mathbf{X}^T\mathbf{X} = n\mathbf{I}$, then

$$\hat{w}_{\text{ridge}} = \frac{n}{n+\lambda} w + \frac{1}{n+\lambda}\mathbf{X}^T\epsilon$$

- Recall: $\hat{w}_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
- To analyze bias-variance tradeoff, we need to assume probabilistic generative model: $x_i \sim P_X, \quad \mathbf{y} = \mathbf{X}w + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$
- The true error at a sample with feature x is

$$\begin{aligned}
&\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x] \\
&\quad = \mathbb{E}_{y|x}[(y - \mathbb{E}[y\,|\,x])^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}[y\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x] \\
&\quad = \mathbb{E}_{y|x}[(y - x^Tw)^2 \,|\, x] + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(x^Tw - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]
\end{aligned}$$

$$= \sigma^2 + (x^Tw - \mathbb{E}_{\mathcal{D}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x])^2 + \mathbb{E}_{\mathcal{D}_{\text{train}}}[(\mathbb{E}_{\tilde{\mathcal{D}}_{\text{train}}}[x^T\hat{w}_{\text{ridge}}\,|\,x] - x^T\hat{w}_{\text{ridge}})^2 \,|\, x]$$

(verify at home)

$$= \sigma^2 + \frac{\lambda^2}{(n+\lambda)^2}(w^Tx)^2 + \frac{\sigma^2 n}{(n+\lambda)^2}\|x\|_2^2$$

Irreduc. Error | Bias-squared | Variance

- λ trades off bias + variance
- Larger λ →
  - Bigger bias
  - Smaller variance

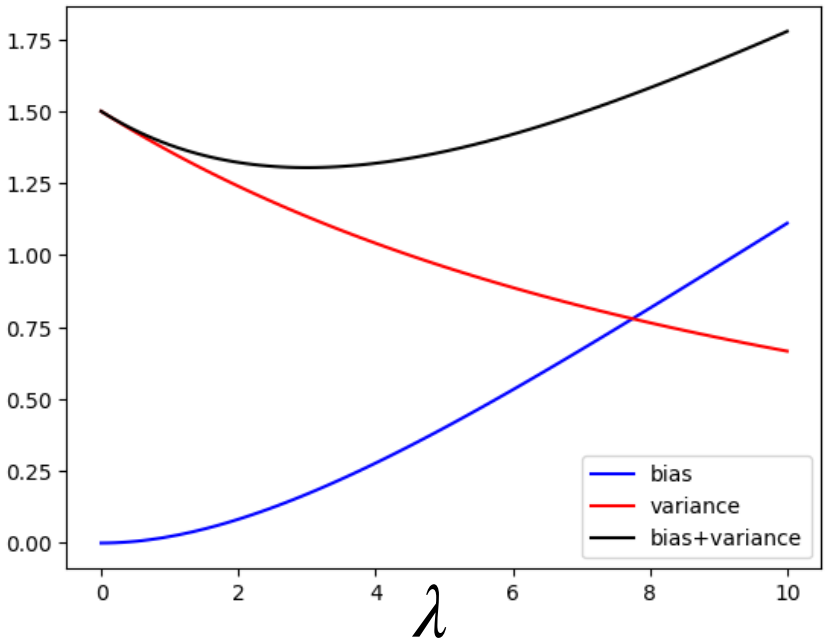
Bias-Variance Properties

Suppose $\mathbf{X}^T\mathbf{X} = n\mathbf{I}$,

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

$$\mathbb{E}_{y,\mathcal{D}_{\text{train}}|x}[(y - x^T\hat{w}_{\text{ridge}})^2 \,|\, x] = \sigma^2 + \underbrace{\frac{\lambda^2}{(n+\lambda)^2}(w^T x)^2}_{\text{Bias-squared}} + \underbrace{\frac{\sigma^2 n}{(n+\lambda)^2}\|x\|_2^2}_{\text{Variance}}$$

d=10, n=20, $\sigma^2 = 3.0, \|w\|_2^2 = 10$



as λ →0,

$\hat{w}_{\text{ridge}} \to \hat{w}_{\text{LS}}$

as λ →∞

$\hat{w}_{\text{ridge}} \to 0$

What you need to know…

- Regularization
  - Penalizes complex models towards preferred, simpler models
- Ridge regression
  - L_2 penalized least-squares regression
  - Regularization parameter trades off model complexity with training error
  - Never regularize the offset!

What you need to know…

- Regularization
  - Penalizes complex models towards preferred, simpler models
- Ridge regression
  - L_2 penalized least-squares regression $\lambda \|w\|_2^2$
  - Regularization parameter trades off model complexity with training error
  - Never regularize the offset!

Simple Variable Selection
LASSO: Sparse Regression



Sparsity

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

- **Vector w is sparse, if many entries are zero**

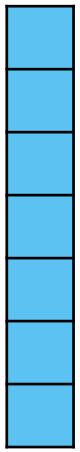
Sparsity

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

- **Vector w is sparse, if many entries are zero**
  - **Efficiency**: If size(**w**) = 100 Billion, each prediction is expensive:
    - **If w is sparse, prediction computation only depends on number of non-zeros**



$$\widehat{y}_i = \widehat{w}_{LS}^{\top} x_i = \sum_{j=1}^{d} x_i[j]\widehat{w}_{LS}[j]$$



Sparsity

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

- **Vector w is sparse, if many entries are zero**
  - **Interpretability**: What are the relevant dimension to make a prediction?

y = delivery time (s)



Candidate features x

Known at prediction time

Route distance (km)	Traffic forecast
Direct distance (km)	Forecast wind speed
Potato weight (lbs)	Forecast tailwind
Battery charge (%)	Forecast rainfall
Package size (L)	Forecast temperature
Planned stops	Launch queue

Illustrative potato delivery example

- How do we find “best” subset among all possible?

Finding best subset: Exhaustive

- Try all subsets of size 1, 2, 3, … and one that minimizes validation error
- Problem?

Finding best subset: Greedy

Forward stepwise:
Starting from simple model and iteratively add features most useful to fit

Backward stepwise:
Start with full model and iteratively remove features least useful to fit

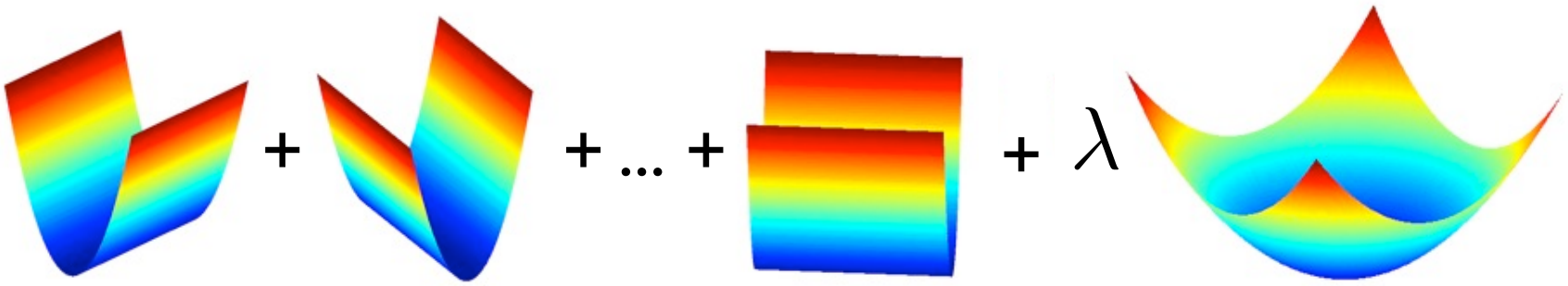
Combining forward and backward steps:
In forward algorithm, insert steps to remove features no longer as important

Lots of other variants, too.

Finding best subset: Regularize

Ridge regression makes coefficients small

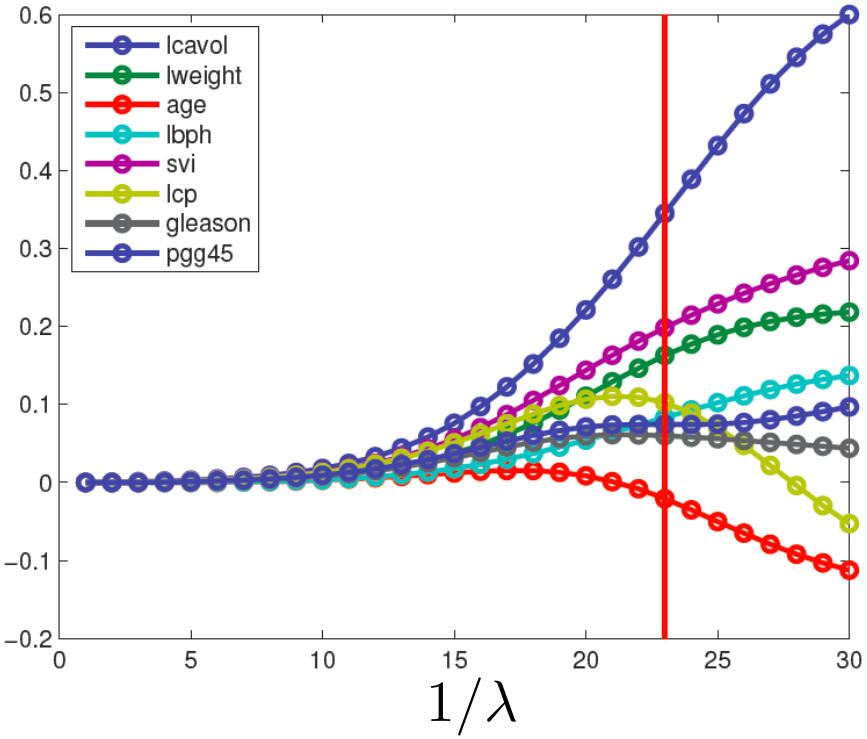
$$\widehat{w}_{ridge} = \arg\min_{w} \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 + \lambda ||w||_2^2$$



Finding best subset: Regularize

Ridge regression makes coefficients small

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

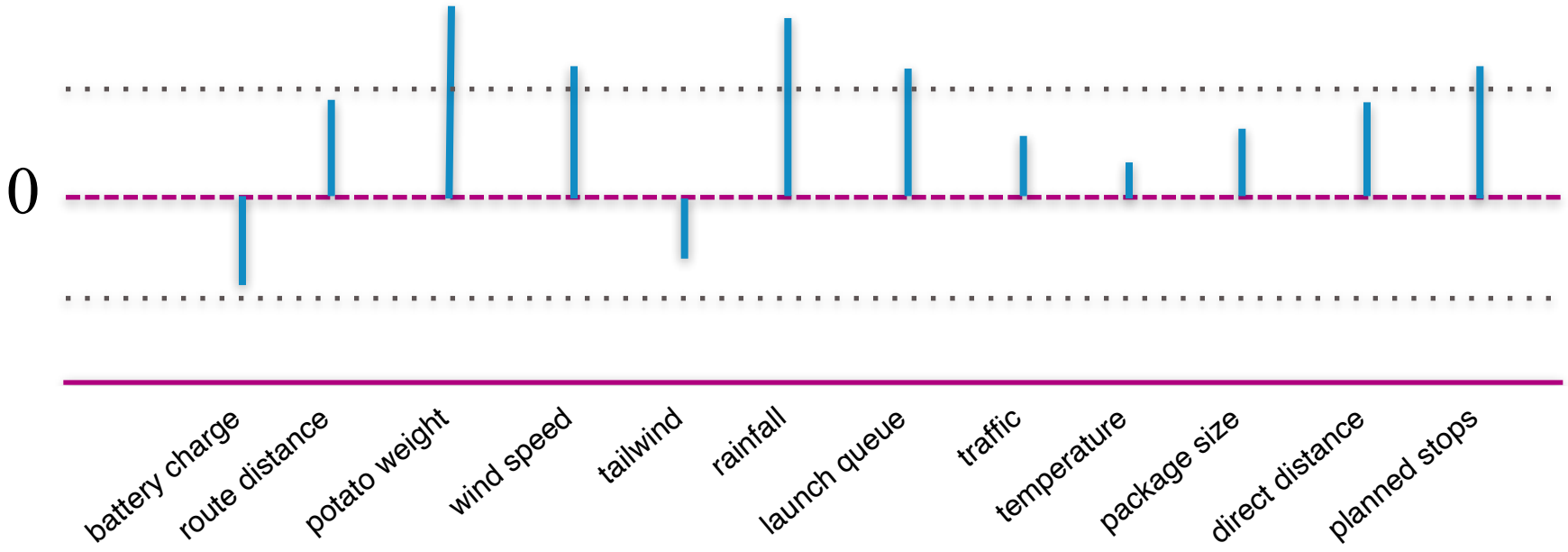


From
Kevin Murphy
textbook

Thresholded Ridge Regression

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

Why don't we just set **small** ridge coefficients to 0?



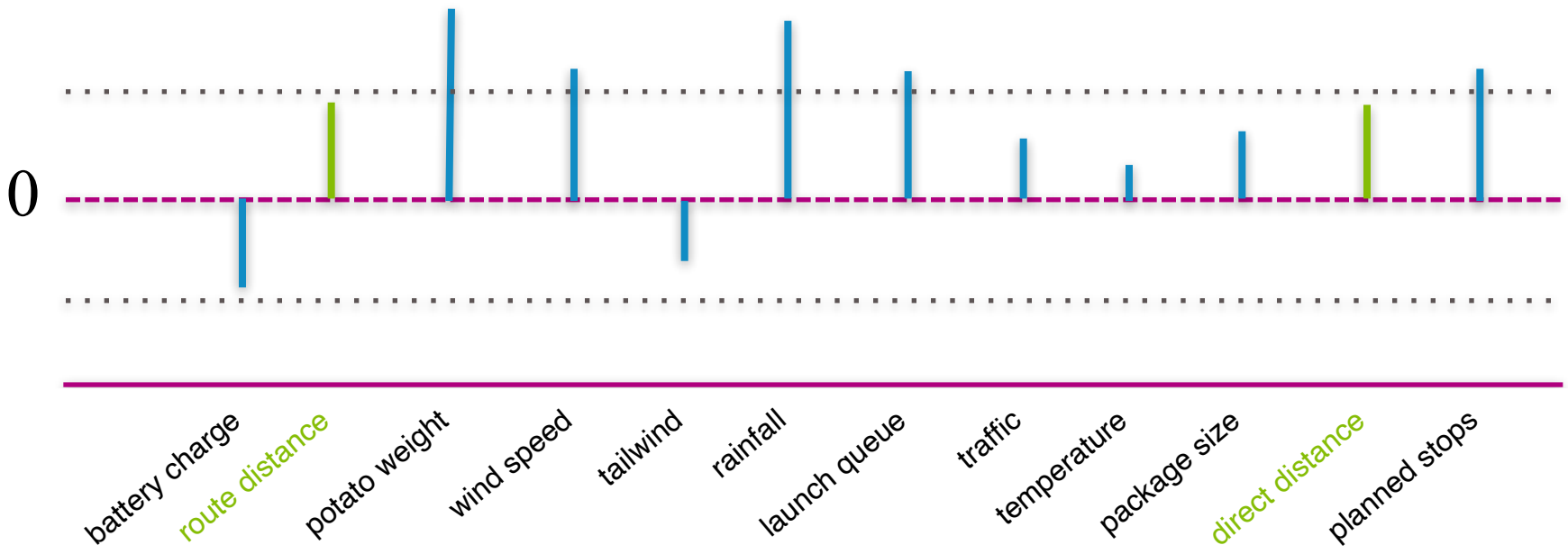
Standardized delivery features

Illustrative

Thresholded Ridge Regression

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

Related features: route distance and direct distance



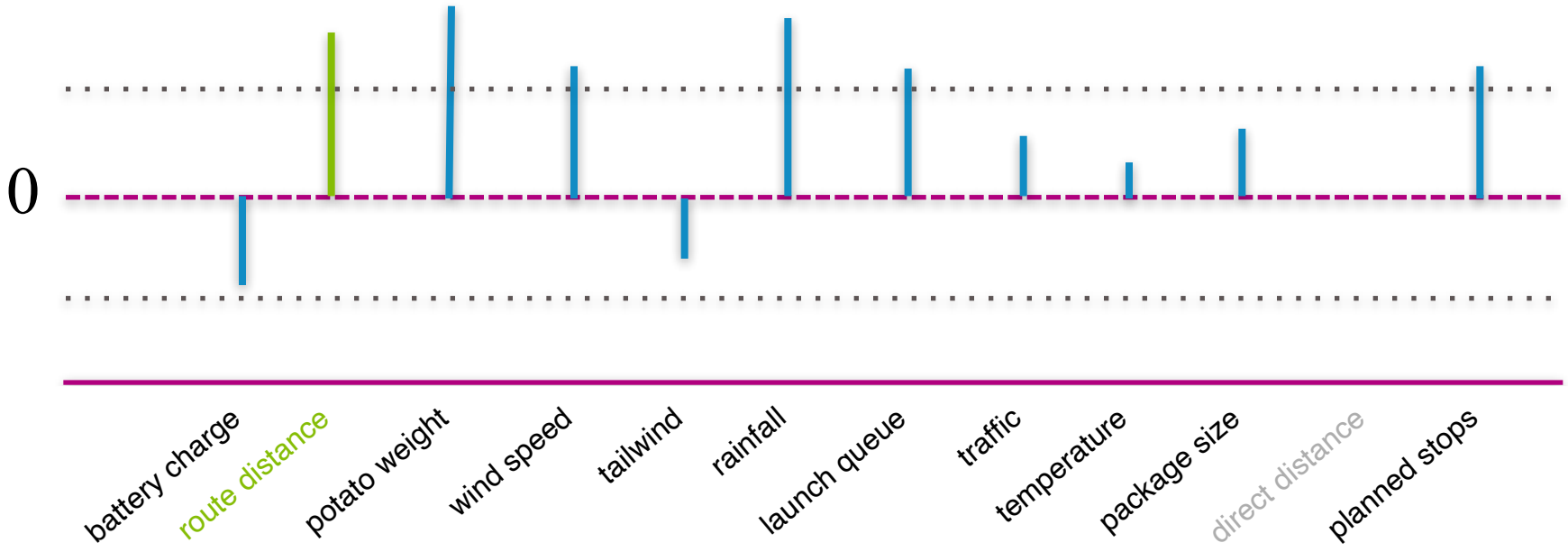
Standardized delivery features

Illustrative

Thresholded Ridge Regression

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

Omit direct distance and refit: route-distance weight may rise.



Standardized delivery features

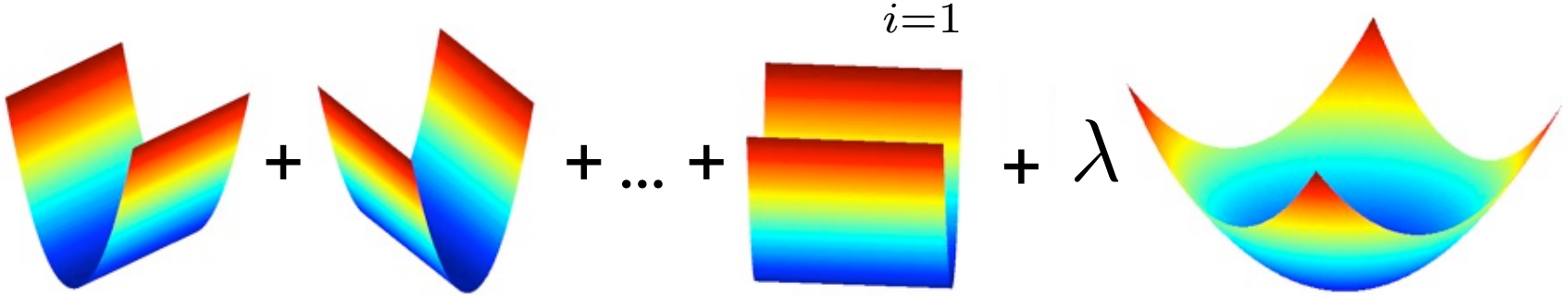
Illustrative

Can another regularizer perform selection automatically?

Recall Ridge Regression

- Ridge Regression objective:

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

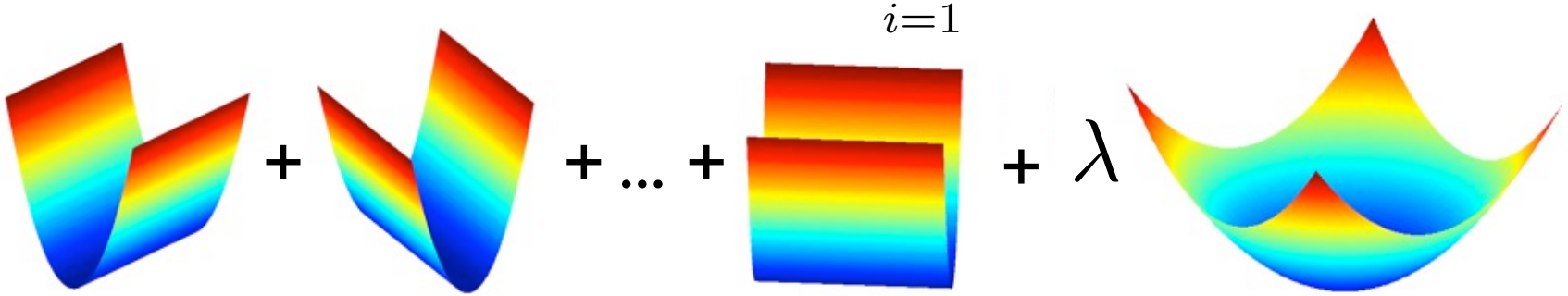


$$\|w\|_p = \left(\sum_{i=1}^{d} |w|^p\right)^{1/p}$$

Ridge vs. Lasso Regression

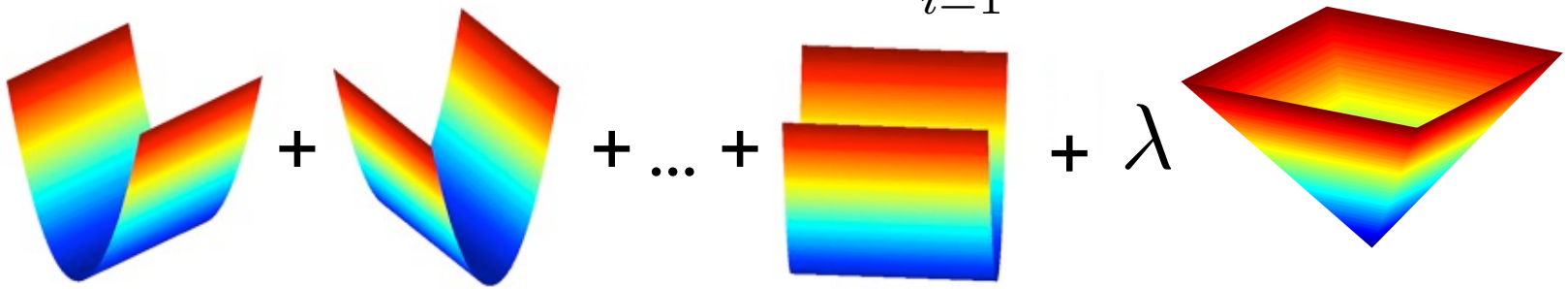
- Ridge Regression objective:

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



- Lasso objective:

$$\widehat{w}_{lasso} = \arg\min_{w} \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 + \lambda ||w||_1$$



Penalized Least Squares

$$\text{Ridge} : r(w) = ||w||_2^2 \qquad \text{Lasso} : r(w) = ||w||_1$$

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

Penalized Least Squares

$$\text{Ridge} : r(w) = ||w||_2^2 \qquad \text{Lasso} : r(w) = ||w||_1$$

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

For any $\lambda \geq 0$ for which $\widehat{w}_r$ achieves the minimum, there exists a $\nu \geq 0$ such that

$$\widehat{w}_r = \arg\min_w \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 \qquad \text{subject to } r(w) \leq \nu$$

Penalized Least Squares

$$\text{Ridge} : r(w) = ||w||_2^2 \qquad \text{Lasso} : r(w) = ||w||_1$$

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

For any $\lambda \geq 0$ for which $\widehat{w}_r$ achieves the minimum, there exists a $\nu \geq 0$ such that

$$\widehat{w}_r = \arg\min_w \sum_{i=1}^{n} \left(y_i - x_i^T w\right)^2 \qquad \text{subject to } r(w) \leq \nu$$

