GENERAL ASSEMBLY

LINEAR REGRESSION PT 2

Brian Chung

CHI HACK NIGHTS

About

Join us every Tuesday from 6-10pm on the 8th floor of the Merchandise Mart to hear from amazing speakers, learn from each other and work on civic projects. **Everyone is welcome!**

We are a group of thousands of designers, academic researchers, data journalists, activists, policy wonks, web developers and curious citizens who want to make our city more just, equitable, transparent and delightful to live in through data, design and technology. More about us »

Pensions	Modelling Pension Reform in Illinois	 Denis Roarty  Ben Galewsky	Explore ways to use data and models to help pensioners and tax payers understand how reform proposals will impact them and each other.

Beach Water Quality	*E. coli* Predictions	 Tom Schenk Jr.	A statistical model is used to predict the *E. coli* levels at Chicago's beaches to determine whether a beach advisory is issued to warn swimmers of potentially high levels of bacteria. However, the actual levels of bacteria is not known until the next day when lab tests have been completed. This project has the goal of increasing the accuracy of these statistical predictions, avoiding unnecessary beach advisories and correctly issuing advisories when bacteria is present.

LINEAR REGRESSION AGENDA

I. Finish up Linear Regression

II. Regularization

III. Cross Validation

I. LINEAR REGRESSION

TYPES OF ML SOLUTIONS

	Continuous	Categorical
Supervised	Regression	Classification
Unsupervised	Dimension Reduction	Clustering

INTRO TO REGRESSION

Q: What is a **regression** model?

A: It is a **functional** relationship between **input** & **response** variables

$y \sim f(x, \beta)$

$ Advertising in:

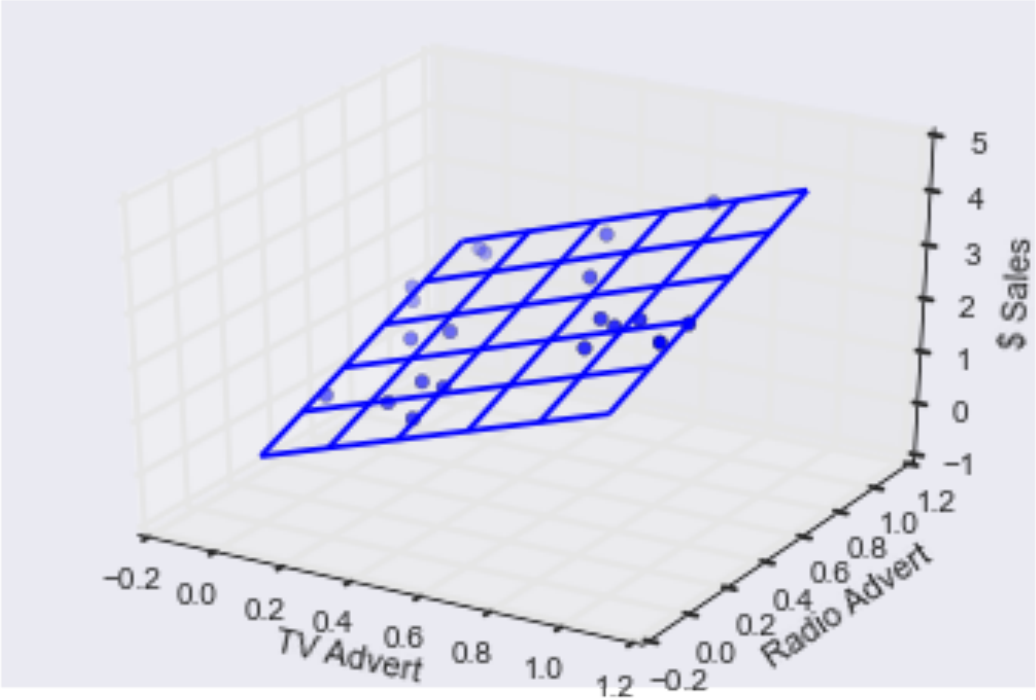
- TV
- Radio
- Newspaper

$ Product Sales

INTRO TO REGRESSION

Naturally, we can extend this to multiple input variables, giving us the **multiple linear regression** model

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \ldots + \beta_n x_n + \varepsilon$$



***y*: Predicted Sales**

***α*: Intercept Value**

***β_1*: Regression Coefficient (Beta1)**

***β_2*: Regression Coefficient (Beta2)**

…

***ε*: Residual (Error)**

SOLVING FOR REGRESSION COEFFICIENTS (ADVANCED)

In class solving for *"β"* that minimizes the cost function

$$Y = X\beta + \varepsilon$$

$$J(\beta) = \| (Y - X\beta) \|^2$$

SOLVING FOR REGRESSION COEFFICIENTS

The **ordinary least squares** solution for our coefficients

There is a **closed form solution** as you've seen, however, many machine learning problems do not necessarily have a closed form solution

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

FINISH UP LINEAR BASICS IN PYTHON NOTEBOOK

Let’s explore some more features of linear regression in the Python notebook…

I. LINEAR REGRESSION
II. REGULARIZATION

MODEL COMPLEXITY

Q: How do we define the **complexity** of a regression model?

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \varepsilon$$

MODEL COMPLEXITY

Q: How do we define the **complexity** of a regression model?

A: One method is to define complexity as a function of the size of the coefficients

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n + \varepsilon$$

MODEL COMPLEXITY

Q: How do we define the **complexity** of a regression model?

A: One method is to define complexity as a function of the size of the coefficients
****This means the features would have to be standardized****

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n + \varepsilon$$

MODEL COMPLEXITY

Q: How do we define the **complexity** of a regression model?

A: One method is to define complexity as a function of the size of the coefficients
This means the features would have to be standardized

L1 norm $$\sum \left| \beta_i \right|$$

L2 norm $$\sqrt{\sum \beta_i^{\,2}}$$

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \varepsilon$$

MODEL COMPLEXITY

These measures of magnitude lead to the following regularization techniques…

L1 norm $\sum \left| \beta_i \right|$

L2 norm $\sqrt{\sum \beta_i^{\,2}}$

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \varepsilon$$

COST FUNCTIONS

To solve for *"β", we chose the β that minimized the sum squared errors*

$$\min J(\beta) = \min \|(Y - X\beta)\|^2$$

COST FUNCTIONS

L1 Regularization chooses betas to minimize the sum squared errors as well as the sum of absolute values of beta

OLS:

$$\min J(\beta) = \min \|(Y - X\beta)\|^2$$

L1 Regularization

$$\min J(\beta) = \min(\|(Y - X\beta)\|^2 + \lambda\|\beta\|_1)$$

COST FUNCTIONS

L2 Regularization chooses betas to minimize the sum squared errors as well as the sum of squared values of beta

OLS: $$\min J(\beta) = \min ||(Y - X\beta)||^2$$

L1 Regularization $$\min J(\beta) = \min(||(Y - X\beta)||^2 + \lambda||\beta||_1)$$

L2 Regularization $$\min J(\beta) = \min(||(Y - X\beta)||^2 + \lambda||\beta||_2^2)$$

COST FUNCTIONS

Regularization refers to the method of preventing **overfitting** by explicitly controlling model **complexity**

OLS:

$$\min J(\beta) = \min \| (Y - X\beta) \|^2$$

LASSO

$$\min J(\beta) = \min(\| (Y - X\beta) \|^2 + \lambda \|\beta\|_1)$$

Ridge Regression

$$\min J(\beta) = \min(\| (Y - X\beta) \|^2 + \lambda \|\beta\|_2^2)$$

BIAS AND VARIANCE

Q: What are bias and variance?

BIAS AND VARIANCE

Q: What are bias and variance?

A: **Bias** refers to predictions that are systematically inaccurate

BIAS AND VARIANCE

Q: What are bias and variance?

A: **Bias** refers to predictions that are systematically inaccurate

Variance refers to predictions that are generally inaccurate

BIAS AND VARIANCE

Q: What are bias and variance?

Bias = *systematic error*
Variance = *general error*

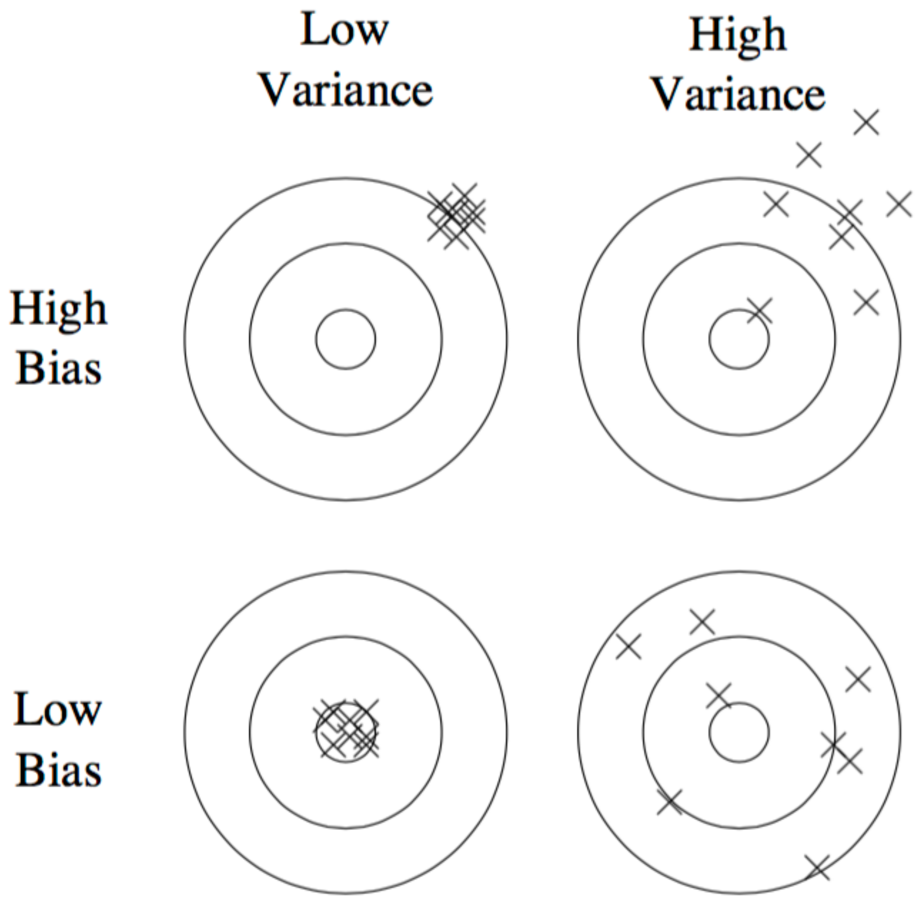


Figure 1: Bias and variance in dart-throwing.

BIAS AND VARIANCE

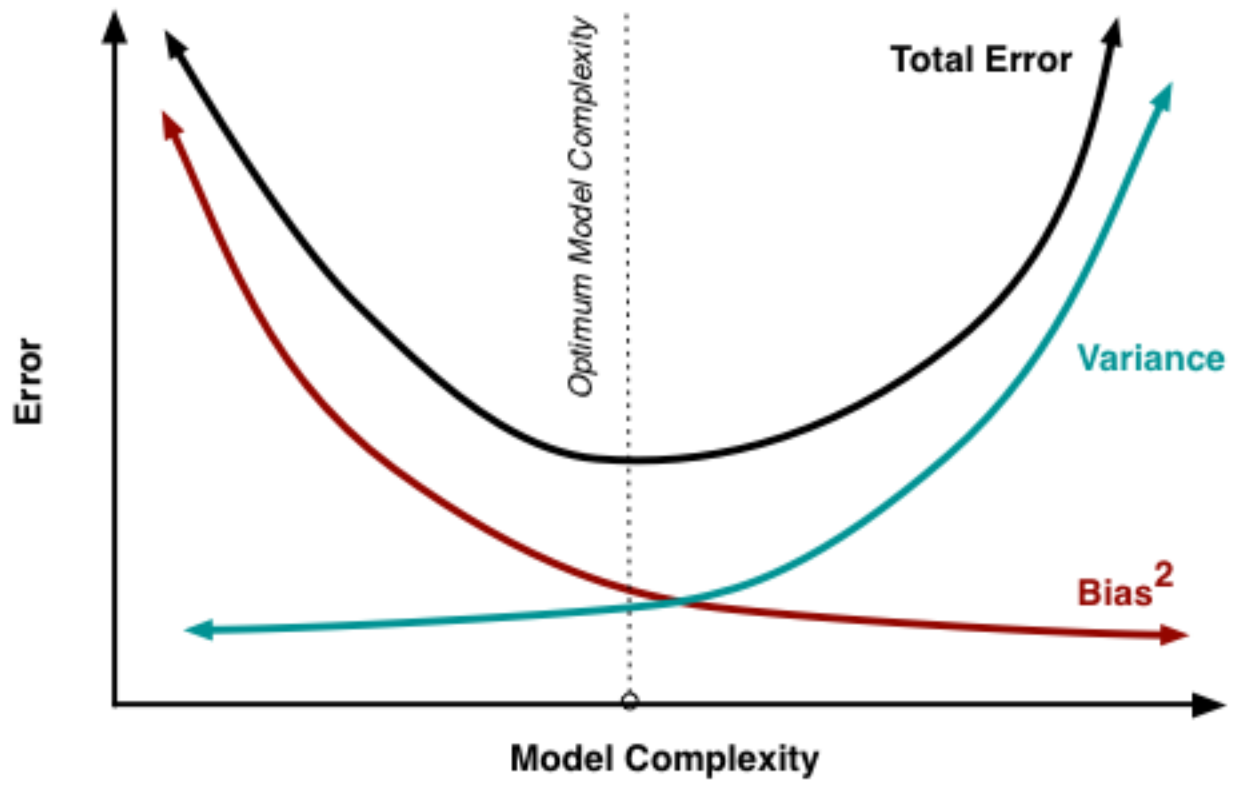
Q: What are bias and variance?

Bias = *systematic error*
Variance = *general error*

The generalization error (test error) in our model can be decomposed into a bias component and variance component (as well as an irreducible component)

BIAS AND VARIANCE

Q: What are bias and variance?

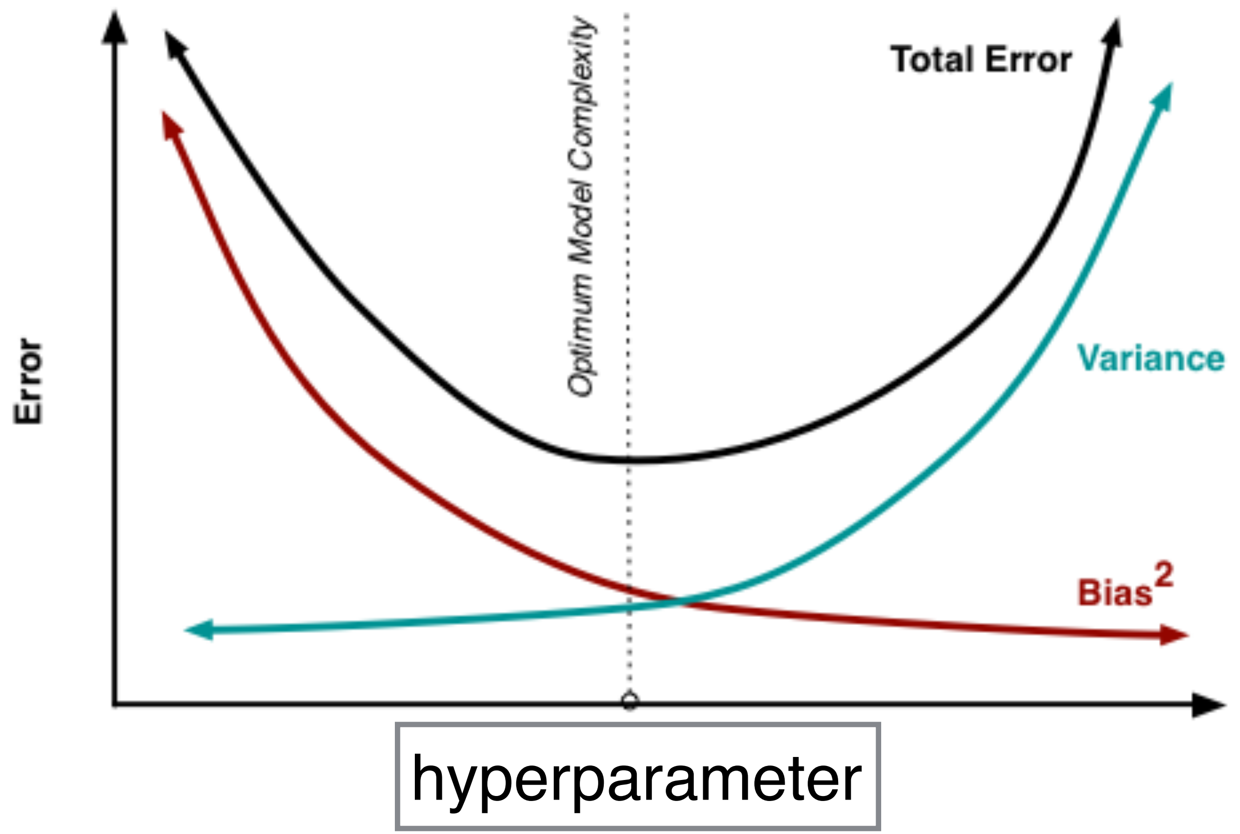


Bias = *systematic error*
Variance = *general error*

The generalization error (test error) in our model can be decomposed into a bias component and variance component (as well as an irreducible component)

BIAS AND VARIANCE

Q: What are bias and variance?



Bias = *systematic error*
Variance = *general error*

The generalization error (test error) in our model can be decomposed into a bias component and variance component (as well as an irreducible component)

BIAS AND VARIANCE

The tradeoff is regulated by the **hyperparameter lambda**

This is an example of **bias-variance** tradeoff

OLS: $$\min J(\beta) = \min ||(Y - X\beta)||^2$$

LASSO $$\min J(\beta) = \min(||(Y - X\beta)||^2 + \lambda||\beta||_1)$$

Ridge Regression $$\min J(\beta) = \min(||(Y - X\beta)||^2 + \lambda||\beta||_2^2)$$

BIAS AND VARIANCE

The tradeoff is regulated by the **hyperparameter lambda**

Regularization (by modulating the lambda), represents a method to trade away some variance for a little bias in our model, thus achieving a better overall fit

OLS:

$$\min J(\beta) = \min \|(Y - X\beta)\|^2$$

LASSO

$$\min J(\beta) = \min(\|(Y - X\beta)\|^2 + \lambda \|\beta\|_1)$$

Ridge Regression

$$\min J(\beta) = \min(\|(Y - X\beta)\|^2 + \lambda \|\beta\|_2^2)$$

HYPERPARAMETER SELECTION

Q: **Great!** So we can build a large linear model, and choose the **lambda** that most reduces our test error. But….how do we do that?

HYPERPARAMETER SELECTION

Q: **Great!** So we can build a large linear model, and choose the **lambda** that most reduces our test error. But….how do we do that?

A: Our dear old friend, **cross validation**

HYPERPARAMETER SELECTION

Q: **Great!** So we can build a large linear model, and choose the **lambda** that most reduces our test error. But….how do we do that?

A: Our dear old friend, **cross validation**

HYPERPARAMETER SELECTION

Q: **Great!** So we can build a large linear model, and choose the **lambda** that most reduces our test error. But….how do we do that?

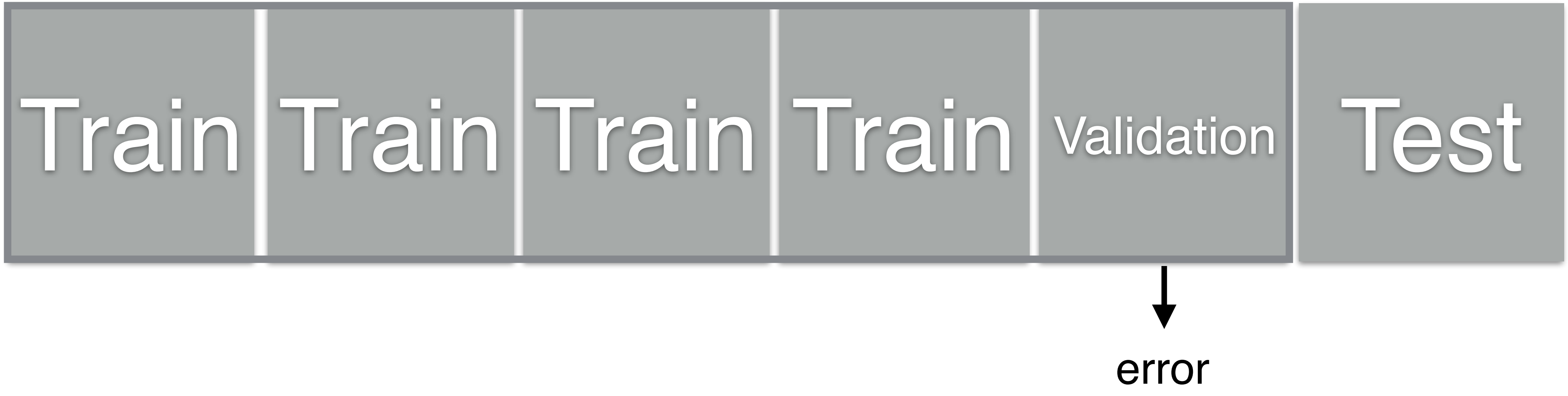
A: Our dear old friend, **cross validation**

HYPERPARAMETER SELECTION

Q: **Great!** So we can build a large linear model, and choose the **lambda** that most reduces our test error. But….how do we do that?

A: Our dear old friend, **cross validation**

Split into train and test sets. Within the training set, use cross validation to find the lambda the results in the *simplest model* with lowest avg error

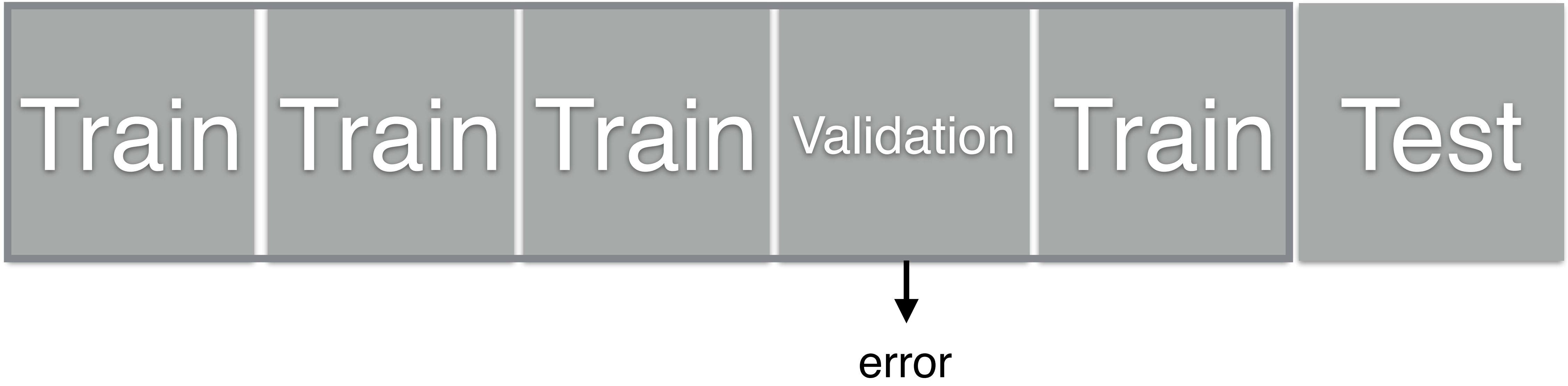


HYPERPARAMETER SELECTION

Q: **Great!** So we can build a large linear model, and choose the **lambda** that most reduces our test error. But….how do we do that?

A: Our dear old friend, **cross validation**

Split into train and test sets. Within the training set, use cross validation to find the lambda the results in the *simplest model* with lowest avg error

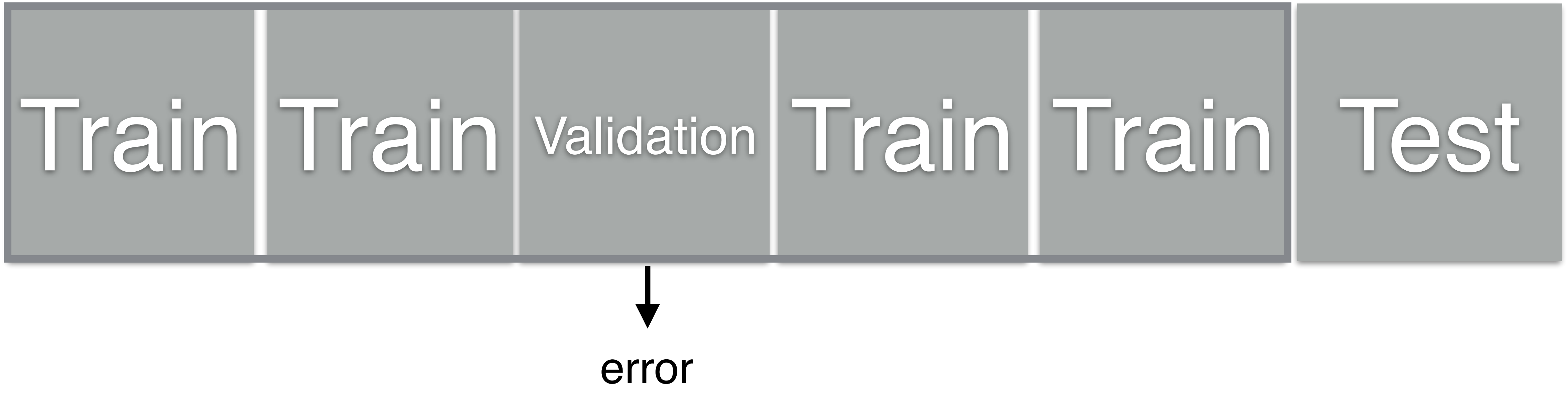


HYPERPARAMETER SELECTION

Q: **Great!** So we can build a large linear model, and choose the **lambda** that most reduces our test error. But….how do we do that?

A: Our dear old friend, **cross validation**

Split into train and test sets. Within the training set, use cross validation to find the lambda the results in the *simplest model* with lowest avg error



HYPERPARAMETER SELECTION

Q: **Great!** So we can build a large linear model, and choose the **lambda** that most reduces our test error. But….how do we do that?

A: Our dear old friend, **cross validation**

Split into train and test sets. Within the training set, use cross validation to find the lambda the results in the *simplest model* with lowest avg error

HYPERPARAMETER SELECTION

Q: **Great!** So we can build a large linear model, and choose the **lambda** that most reduces our test error. But….how do we do that?

A: Our dear old friend, **cross validation**

Split into train and test sets. Within the training set, use cross validation to find the lambda the results in the *simplest model* with lowest avg error

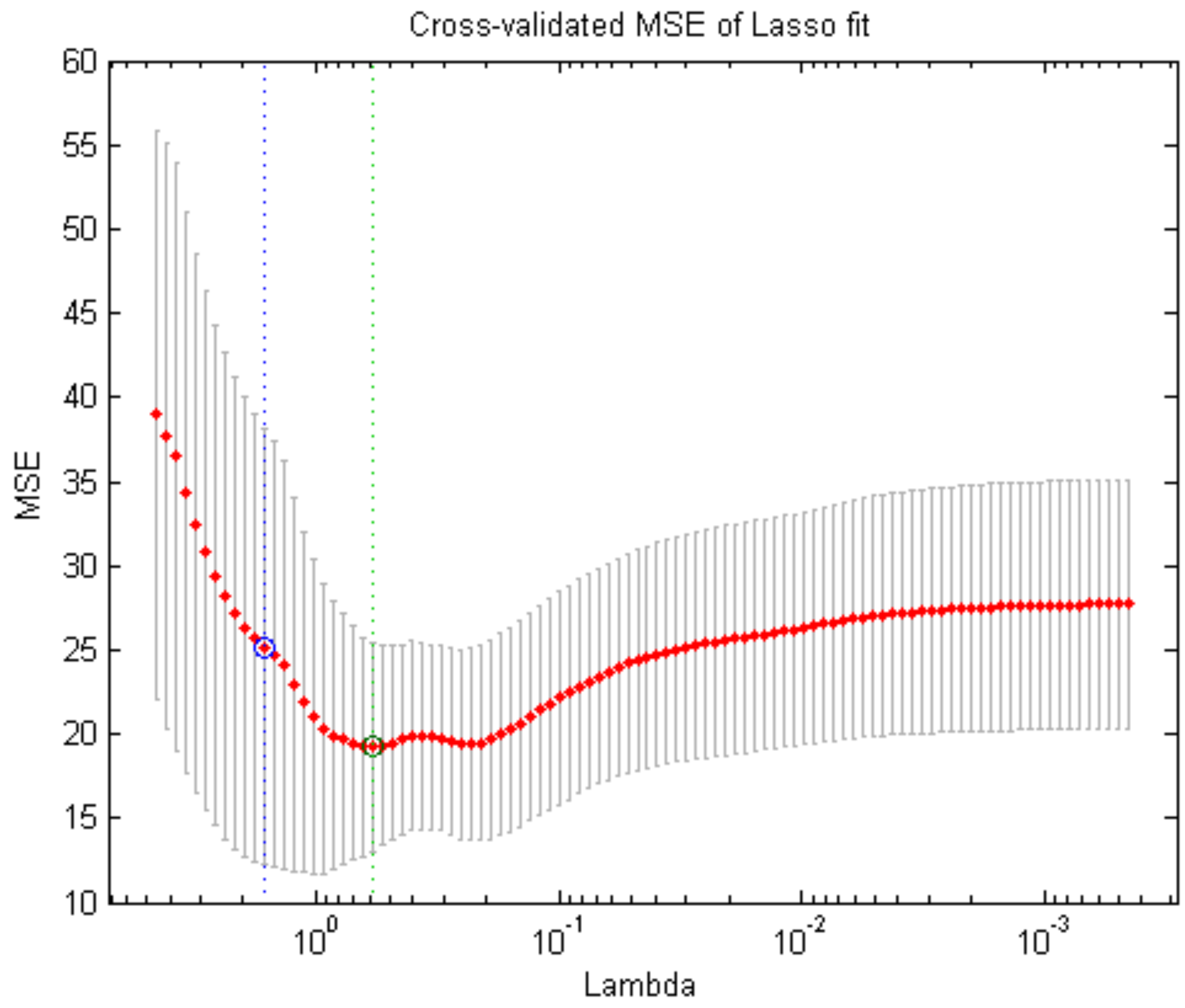
HYPERPARAMETER SELECTION

Algorithm

- Split data into train and test sets
- On train data:

  For lambda 1000 to .0001

    Generate avg of cross validated MSE with that particular lambda on the K folds
- Choose the simplest lambda that results in lowest error
- Another choice is the 1 std error rule. Choose the simplest lambda that is within 1 SE of the lowest error lambda



RIDGE (L2) VS LASSO (L1)

Ridge Regression

Pros:

- Easier to implement and compute
- There's a closed form solution
- Also solves the issue of singularities!

Cons:

- No feature selection. Either keep every feature, or no features
- Need to standardize each feature

LASSO

Pros:

- Solves issues of singularities
- Also performs feature selection!
- dfdfd

Cons:

- Need to standardize each feature
- More complex to compute

RIDGE (L2) VS LASSO (L1)

Both solve issues of:

* **Categorical data with lots of levels**
* **Too many factors for the amount of data**
* **Collinear factors**

More information:

http://www.machinelearning.org/proceedings/icml2004/papers/354.pdf

THAT'S IT!

- Exit Tickets: DAT1 - Lesson 6 - Regularization
- Homework 4 is due Jan 11, 2016