GENERAL ASSEMBLY

EVALUATING MODEL FIT

Tan Kwan Chong

Chief Data Scientist, Booz Allen Hamilton

LEARNING OBJECTIVES

- Define regularization, bias, and error metrics for regression problems
- Evaluate model fit using loss functions
- Select regression methods based on fit and complexity

INTRODUCTION

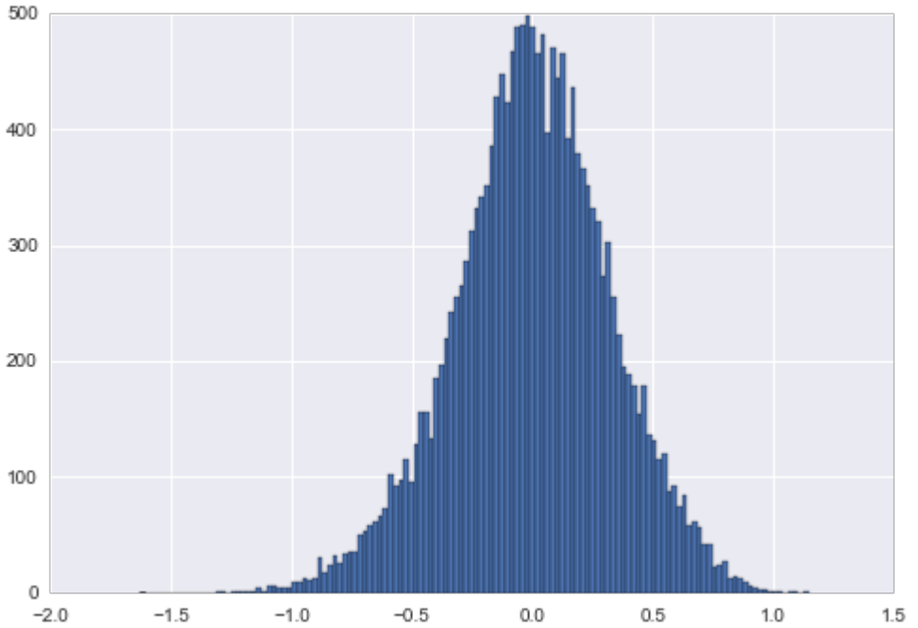
LINEAR MODELS AND ERROR

WHAT IS R-SQUARED? WHAT IS A RESIDUAL?

- R-squared, the central metric introduced for linear regression
- Which model performed better, one with an r-squared of 0.79 or 0.81?
- R-squared measures explain variance.
- But does it tell the magnitude or scale of error?
- We’ll explore loss functions and find ways to refine our model.

RECALL: WHAT'S RESIDUAL ERROR?

- In linear models, residual error must be normal with a median close to zero.
- Individual residuals are useful to see the error of specific points, but it doesn't provide an overall picture for optimization.
- We need a metric to summarize the error in our model into one value.



R-SQUARED

- Recall that R-squared or fraction of variance explained is

$$R^2 = \frac{TSS - RSS}{TSS} = 1 - \frac{RSS}{TSS}$$

- R-squared has very intuitive properties. When a model does no better than a null model then R-squared will be 0.
- When a model makes perfect predictions, R-squared will be 1.
- Commonly R-squared is only applied as a measure of training error and it is possible to overfit a model with a large number of parameters that are pure noise

ADJUSTED R-SQUARED

- Adjusted R-squared adjusts for this problem by penalizing additional complexity

$$Adjusted\ R^2 = 1 - \frac{(1 - R^2)(n - 1)}{n - p}$$

where n is the number of observations and p the number of parameters

- Thus the error predicted by Adjusted R-squared will start to increase as model complexity becomes very high
- However, Adjusted R-squared generally under-penalizes complexity
- R-squared values also don't tell us how far off our predictions are

MEAN ABSOLUTE ERROR (MAE)

- Mean Absolute Error is the mean of the absolute value of errors
- Calculate the difference between each target y and the model's predicted value y-hat (i.e. the residual)
- Apply the absolute function on the residual and take the mean of the resulting absolute error values
- MAE is easy to understand because it is the average error

$$MAE = \frac{1}{n}\sum_{1}^{n}|y_i - \hat{y}_i|$$

MEAN SQUARED ERROR (MSE)

- Mean Squared Error is the mean of the squared errors
- Calculate the difference between each target y and the model's predicted value y-hat (i.e. the residual)
- Square each residual and take the mean of the squared errors
- MSE is more popular than MAE because MSE punishes large errors, which tends to be useful in the real world

$$MSE = \frac{1}{n}\sum_{1}^{n}(y_i - \hat{y}_i)^2$$

ROOT MEAN SQUARED ERROR (RMSE)

- Root Mean Squared Error is the square root of the mean of the squared errors
- Calculate the MSE and take the square root of it
- RMSE is even more popular than MSE because RMSE is interpretable in the “y” units

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n}\sum_{1}^{n}(y_i - \hat{y}_i)^2}$$

SKLEARN ERROR METRICS

- sklearn's metrics module includes mean_absolute_error and mean_squared_error functions.

```
from sklearn import metrics
import numpy as np
MAE = metrics.mean_absolute_error(y, model.predict(X))
MSE = metrics.mean_squared_error(y, model.predict(X))
RMSE = np.sqrt(MSE)
```

HOW DO WE MINIMIZE ERROR?

- The regression method we've used is called "Ordinary Least Squares".
- This means that given a matrix X, solve for the *least* amount of square error for y.
- However, this assumes that X is unbiased, that it is representative of the population.

BIAS VS. VARIANCE

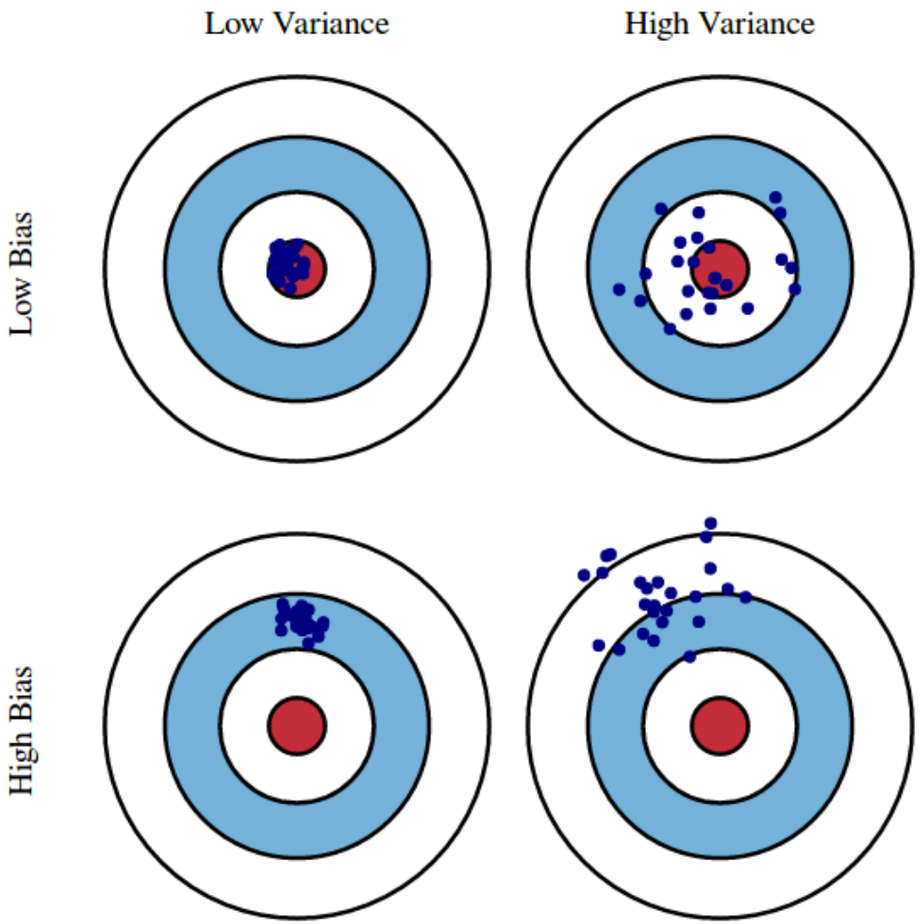


Fig. 1 Graphical illustration of bias and variance.

BIAS VARIANCE TRADEOFF

- When our error is ***biased***, it means the model's prediction is consistently far away from the actual value.

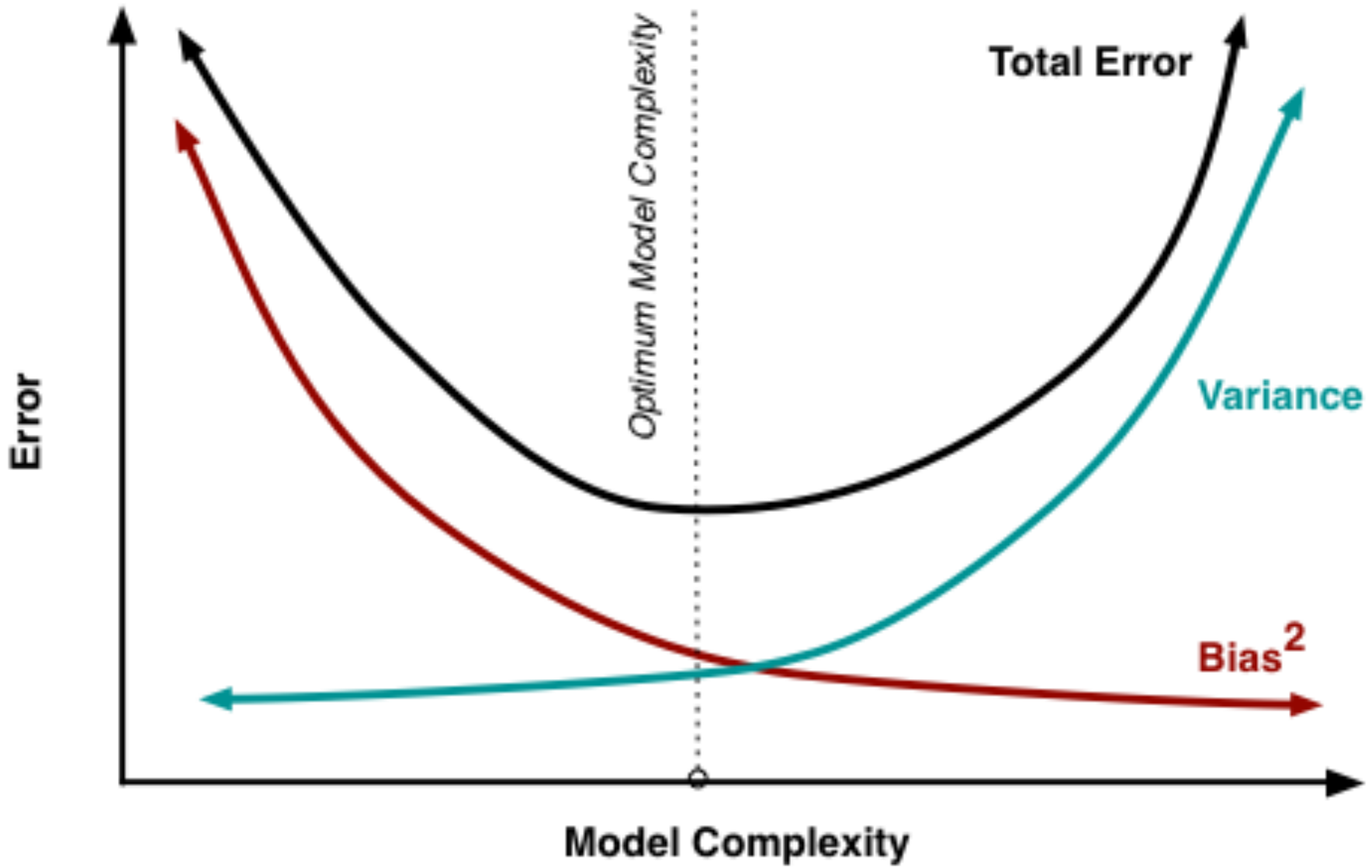
- This could be a sign of poor sampling and poor data or if the model is not a suitable representation of the true relationship.

- One objective of a biased model is to trade bias error for generalized error. We prefer the error to be more evenly distributed across the model.

- This is called error due to ***variance***.

- We want our model to *generalize* to data it hasn't seen even if doesn't perform as well on data it has already seen.

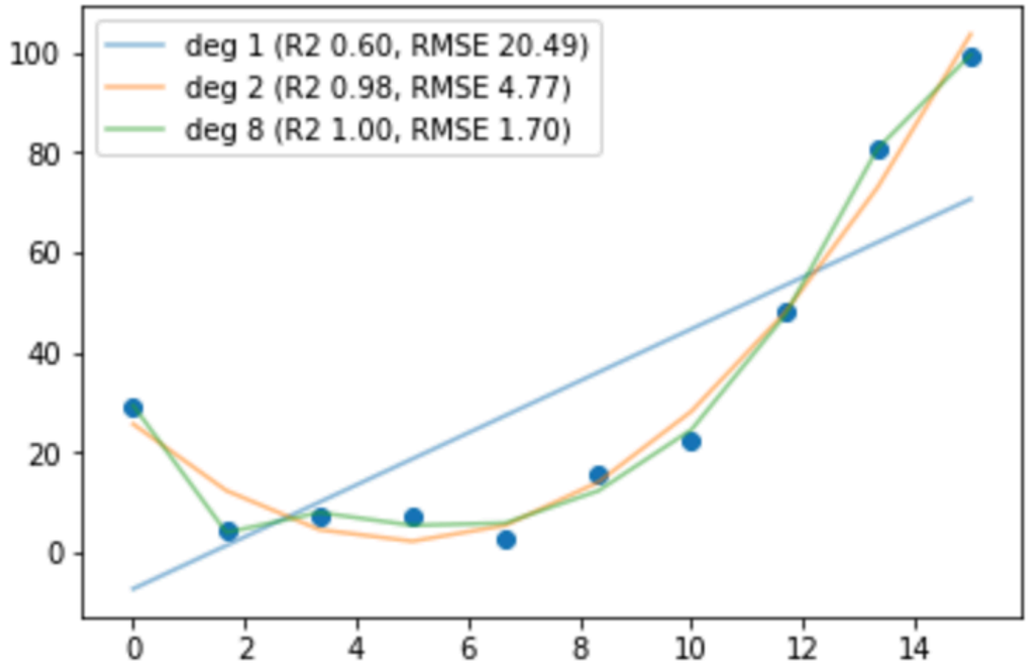
BIAS VARIANCE TRADEOFF

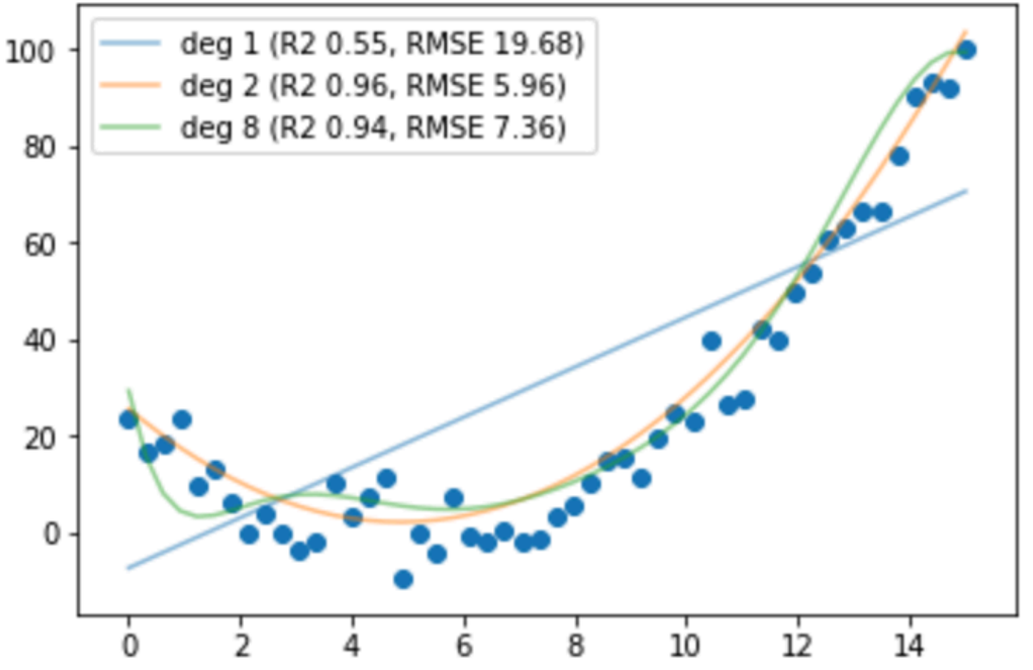


BIAS VARIANCE TRADEOFF

- As model flexibility increases, training MSE will decrease but the test MSE may not
- When a given method yields a small training MSE but a large test MSE, we are said to be overfitting the data
- This happens when the model is working too hard to find patterns in the training data and may be picking up patterns that are just caused by random chance rather than true properties of the unknown function

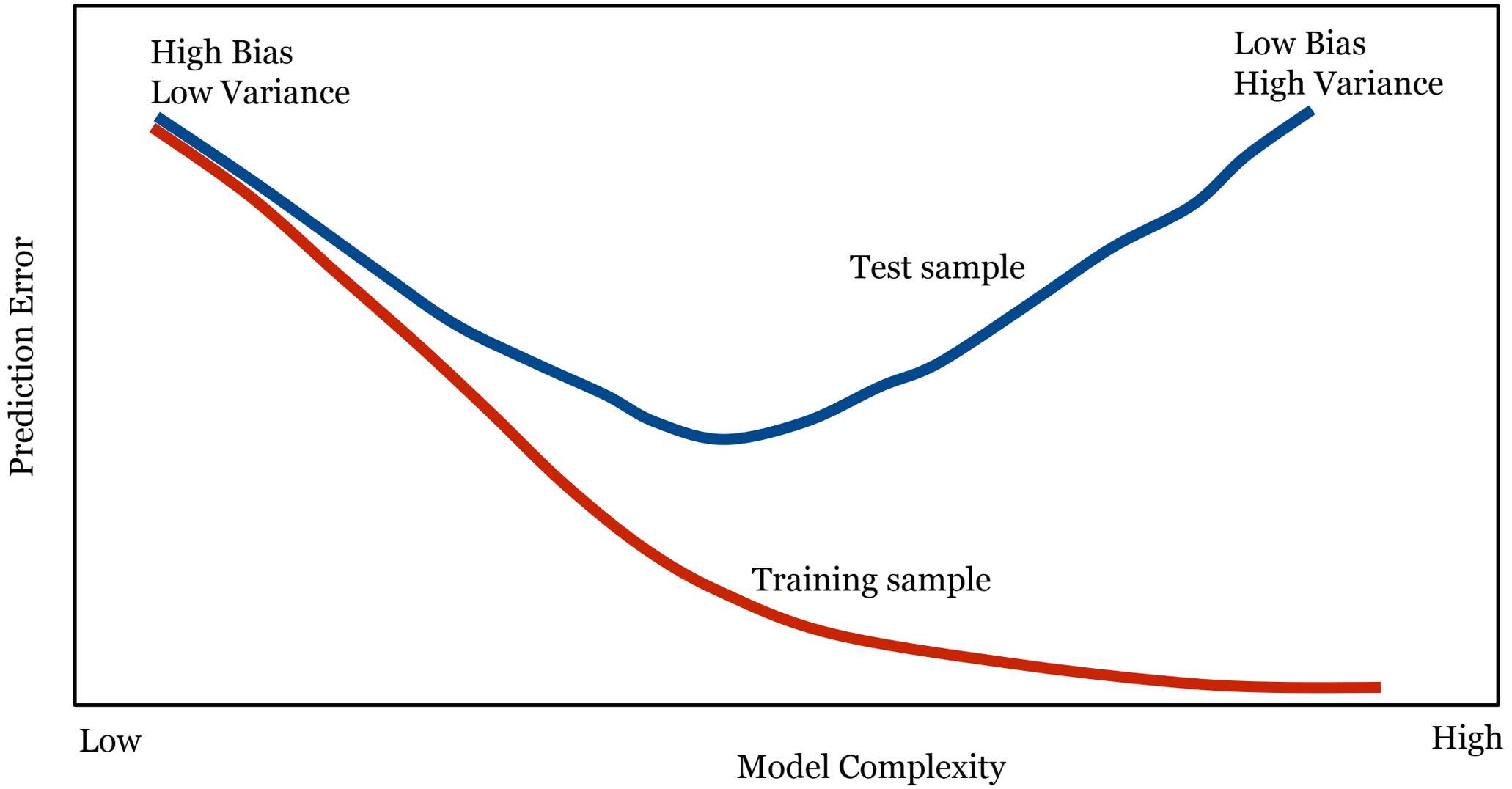
BIAS VARIANCE TRADEOFF



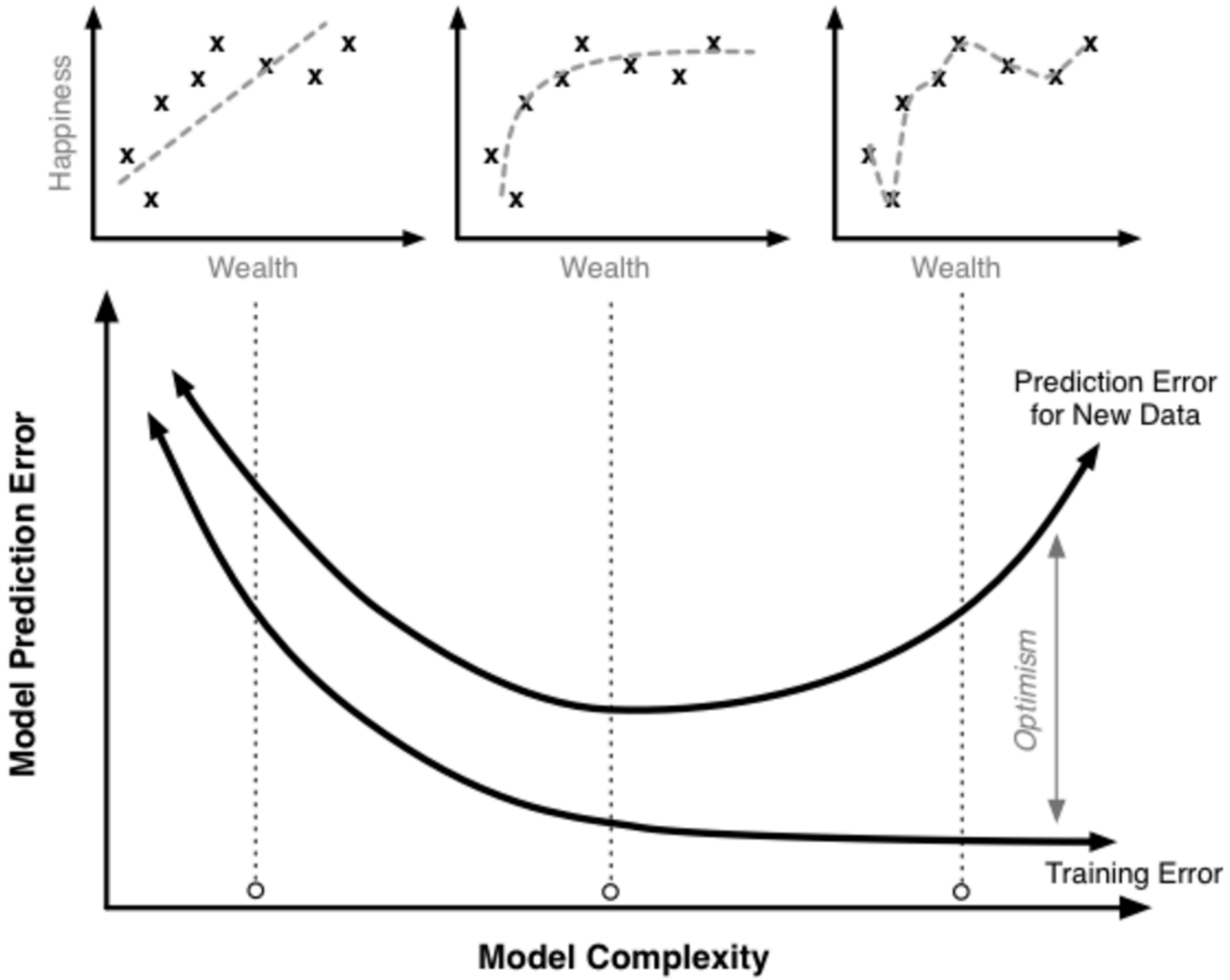


- A model using a high number of polynomials can easily overfit a small sample of training data and perform poorly on unseen or test data

BIAS VARIANCE TRADEOFF



BIAS VARIANCE TRADEOFF



MODEL VALIDATION

MODEL VALIDATION

Validation Set

- We divide the available data into two parts: a training set and a validation set
- The model is fit on the training set, and the fitted model is used to predict the responses for the observations in the validation set
- The resulting validation set error provides an estimate of the test error

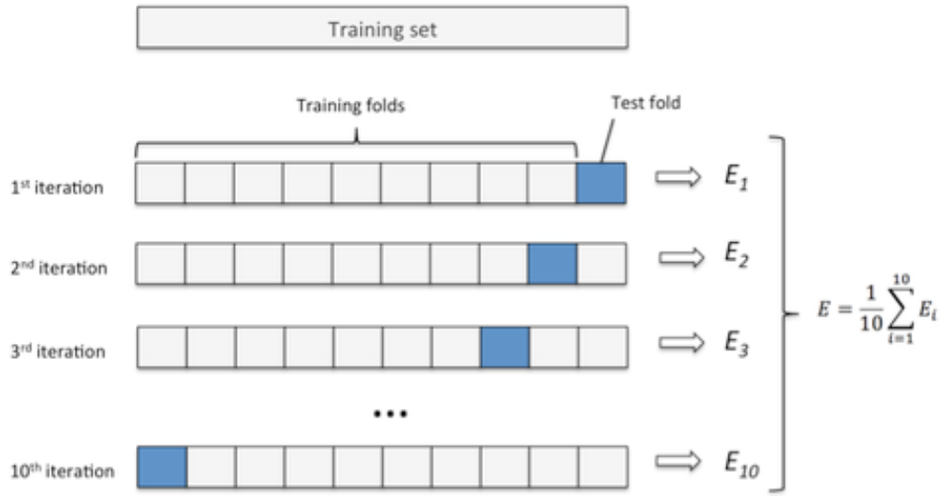
Limitations

- The validation estimate can be highly variable, depending on which observations are included in the training and validation sets
- Only a subset of the observations are used to train the model

MODEL VALIDATION

K-fold Cross-Validation

- Randomly divide the data into K equal-sized parts
- We leave out part k, fit the model to the other K-1 parts (combined) and then obtain predictions for the left out part
- This is done in turn for each part k = 1, 2, … K and the results combined



K = 5, 10 are typically used

MODEL VALIDATION

K-fold Cross-Validation

- The average performance of the K models will be poorer than a model built on all of the data.

- But this technique swaps bias error for generalized error, describing previous trends accurately enough to extend to future trends.

REGULARIZATION

WHAT IS REGULARIZATION? AND WHY DO WE USE IT?

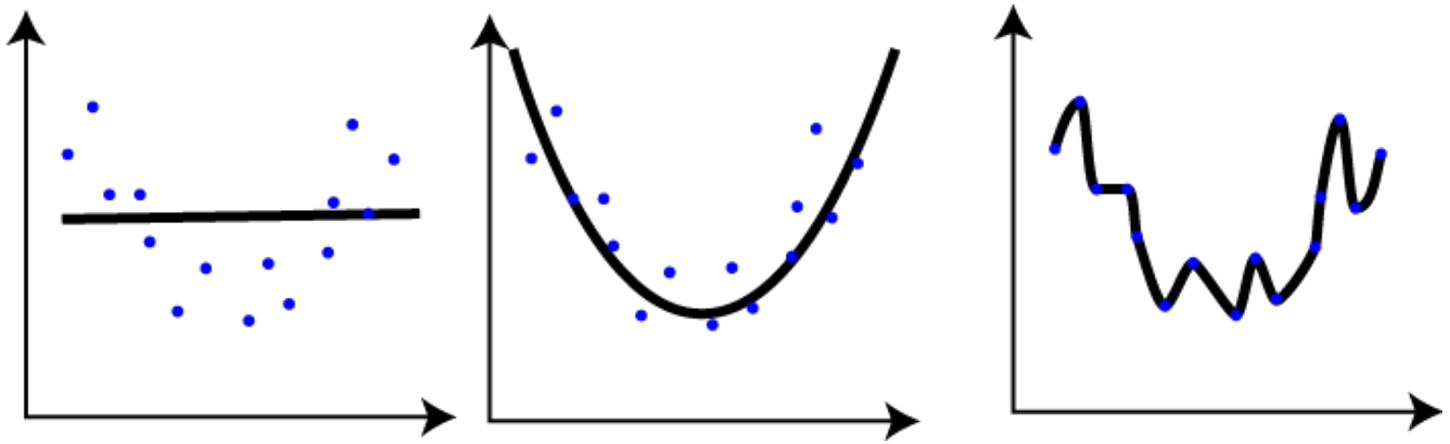
- Regularization is an additive approach to protect models against overfitting (being potentially biased and overconfident, not generalizing well).

- Regularization becomes an additional weight to coefficients, shrinking them closer to zero.

- L1 (Lasso Regression) adds the extra weight to coefficients.

- L2 (Ridge Regression) adds the square of the extra weight to coefficients.

WHAT IS OVERFITTING?



- The first model poorly explains the data.
- The second model explains the general curve of the data.
- The third model drastically overfits the model, bending to every point.
- Regularization helps prevent the third model.

RIDGE REGRESSION

- Recall that the least squares fitting procedure estimates $\beta_0, \beta_1, \ldots, \beta_p$ using values that minimize

$$RSS = \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2$$

- While the ridge regression selects coefficients that minimize

$$\sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 = RSS + \lambda \sum_{j=1}^{p} \beta_j^2$$

- Where $\lambda \geq 0$ is a tuning parameter, to be determined separately

RIDGE REGRESSION

- As with least squares, ridge regression seeks coefficient estimates that fit the data well, minimizing RSS
- The second term is small where $\beta_0, \beta_1, \ldots, \beta_p$ are close to zero and generates a shrinking penalty, driving the estimates towards zero
- The tuning parameter λ serves to control the relative impact of the two terms on the regression coefficient estimates
- Selecting a good value of λ is therefore critical and cross-validation is often used for this

RIDGE REGRESSION

- Unlike the least squares method which is scale equivalent, the ridge regression coefficient estimates can change substantially when multiplying a given predictor by a constant
- Therefore it is best to apply ridge regression after standardizing the predictors

LASSO

- One limitation of ridge regression is that it will include all p predictors into the final model
- The Lasso is an alternative to ridge regression that overcomes this disadvantage. The Lasso coefficients minimize the quantity:

$$\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j| = RSS + \lambda\sum_{j=1}^{p}|\beta_j|$$

LASSO

- As with ridge regression, the Lasso shrinks the coefficient estimates towards zero
- However, the l1 penalty has the effect of forcing some of the coefficient estimates to be exactly equal to zero when the tuning parameter λ is sufficiently large
- Therefore the Lasso yields sparse models that involve only a subset of the variables

CONCLUSION

TOPIC REVIEW

LESSON REVIEW

- What's the (typical) range of r-squared?
- What's the range of mean squared error?
- How would changing the scale or interpretation of y (your target variable) effect mean squared error?
- What's cross validation, and why do we use it in machine learning?
- What is error due to bias? What is error due to variance? Which is better for a model to have, if it had to have one?

COURSE

BEFORE NEXT CLASS

LESSON

Q & A

LESSON

EXIT TICKET

DON’T FORGET TO FILL OUT YOUR EXIT TICKET