Resampling,
Cross-validation & Bootstrap

ISLR2 Chapter 05

Cross Validation

- The Validation Set Approach
- Leave-One-Out Cross Validation
- K-fold Cross Validation
- Bias-Variance Trade-off for k-fold Cross Validation
- Cross Validation on Classification Problems

What are resampling methods?

Tools that involves <u>repeatedly</u> drawing samples

- from a training set
- and refitting a model of interest on each sample
- in order to obtain more information about the fitted model

Model Assessment:

- estimate test error rates

Model Selection:

- select the appropriate level of model flexibility

They are computationally expensive!

- But these days we have powerful computers ☺

Two resampling methods:

- Cross Validation
- Bootstrapping

5.1.1 Typical Approach: The Validation Set Approach

Suppose that we would like to find a set of variables

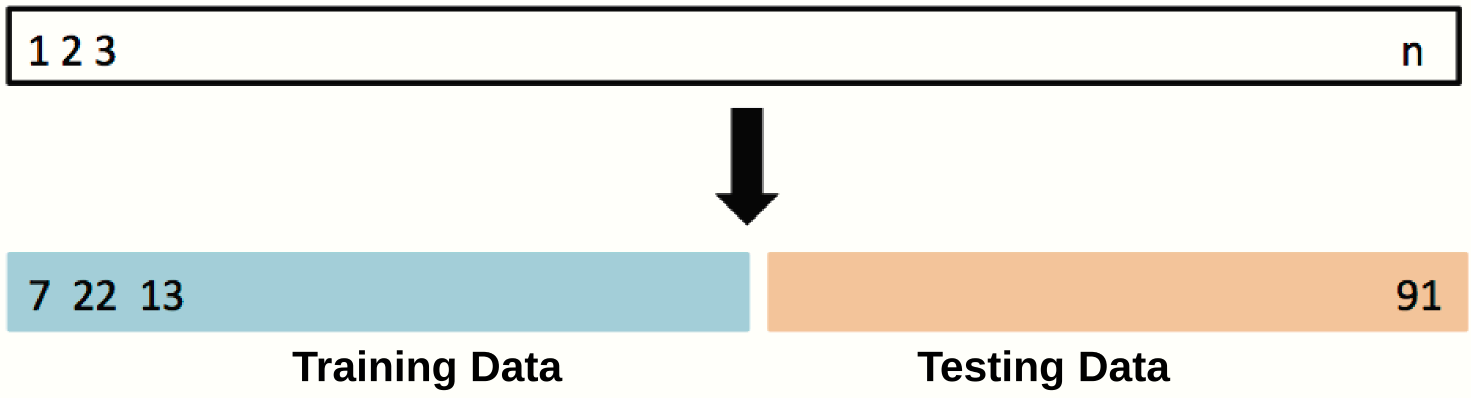
- that give the lowest test (not training) error rate

If we have a large data set,

- we can achieve this goal by randomly splitting the data
- into training and validation(testing) parts

We would then use the training part

- to build each possible model
- (i.e. the different combinations of variables)
- and choose the model that gave the lowest error rate
- when applied to the validation data



Example: Auto Data

Suppose that we want to predict mpg from horsepower

Two models:

- mpg ~ horsepower
- mpg ~ horsepower + horspower^2

Which model gives a better fit?

- Randomly split Auto data set
  into training (196 obs.) and validation data (196 obs.)
- Fit both models using the training data set
- Then, evaluate both models
  using the validation data set
- The model with the lowest validation (testing) MSE is the winner!

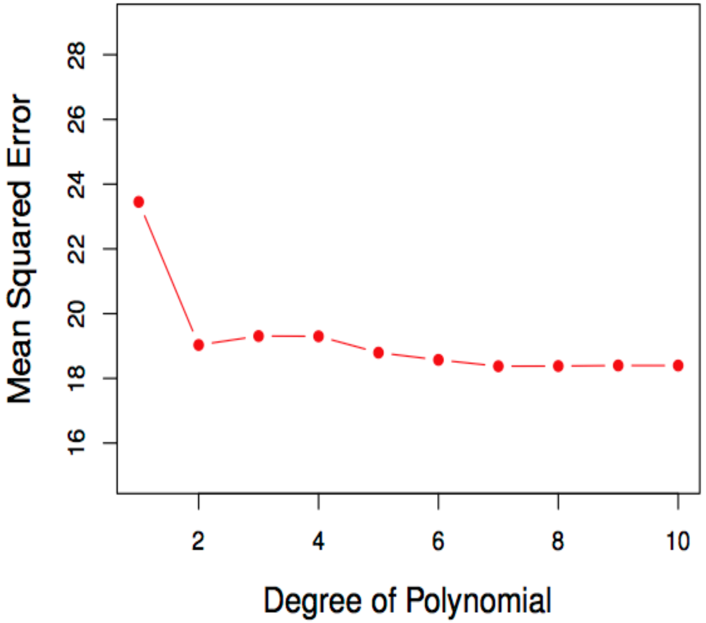
Left: Validation error rate for a single split

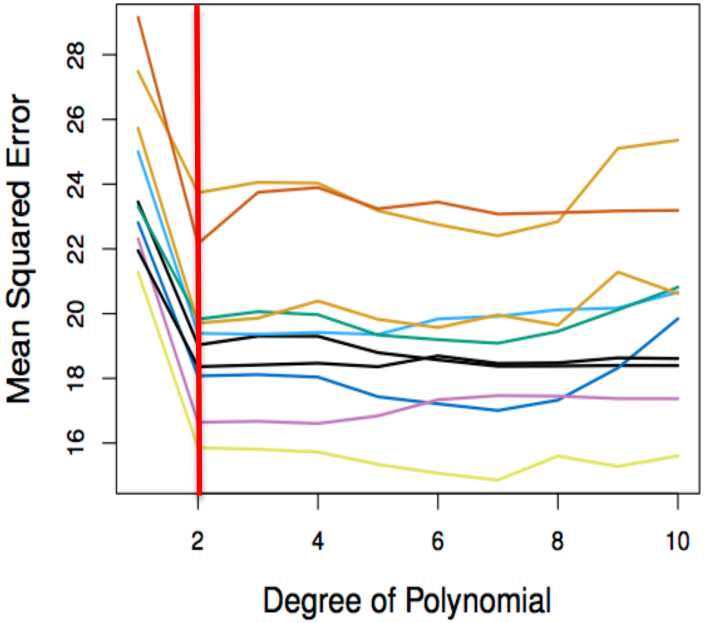
Right: Validation method repeated 10 times,

- **each time the split is done randomly!**

There is a lot of variability among the MSE's…

- **Not good! We need more stable methods!**





Advantages:

- Simple
- Easy to implement

Disadvantages:

- The validation MSE can be highly variable
- Only a subset of observations are used to fit the model (training data).

  Statistical methods tend to perform worse

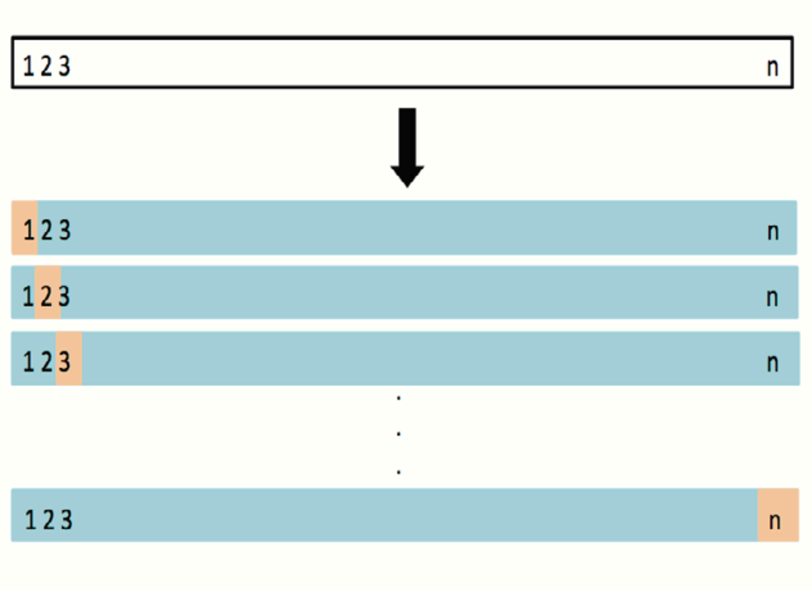
  when trained on fewer observations

5.1.2 Leave-One-Out Cross Validation (LOOCV)

This method is similar to the Validation Set Approach, but it tries to address the latter's disadvantages

For each suggested model, do:

- Split the data set of size n into
  Training data set (blue) size: n -1
  Validation data set (beige) size: 1
- Fit the model using the training data
- Validate model using the validation data, and compute the corresponding MSE
- Repeat this process n times
- The MSE for the model is computed as follows:



$$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n} MSE_i \,.$$

LOOCV vs. the Validation Set Approach

LOOCV has less bias

- We repeatedly fit the statistical learning method
  using training data that contains n-1 obs.,
  i.e. almost all the data set is used

LOOCV produces a less variable MSE

- The validation approach produces different MSE
  when applied repeatedly due to randomness in the splitting process,
- while performing LOOCV multiple times will always yield the same results,
  because we split based on 1 obs. each time

LOOCV is computationally intensive (disadvantage)

- We fit the each model n times!

5.1.3 k-fold Cross Validation

LOOCV is computationally intensive,

- so we can run k-fold Cross Validation instead

With k-fold Cross Validation,

- we divide the data set into K different parts
- (e.g. K = 5, or K = 10, etc.)

We then remove the first part,

- fit the model on the remaining K-1 parts,
- and see how good the predictions are on the left out part
- (i.e. compute the MSE on the first part)

We then repeat this K different times

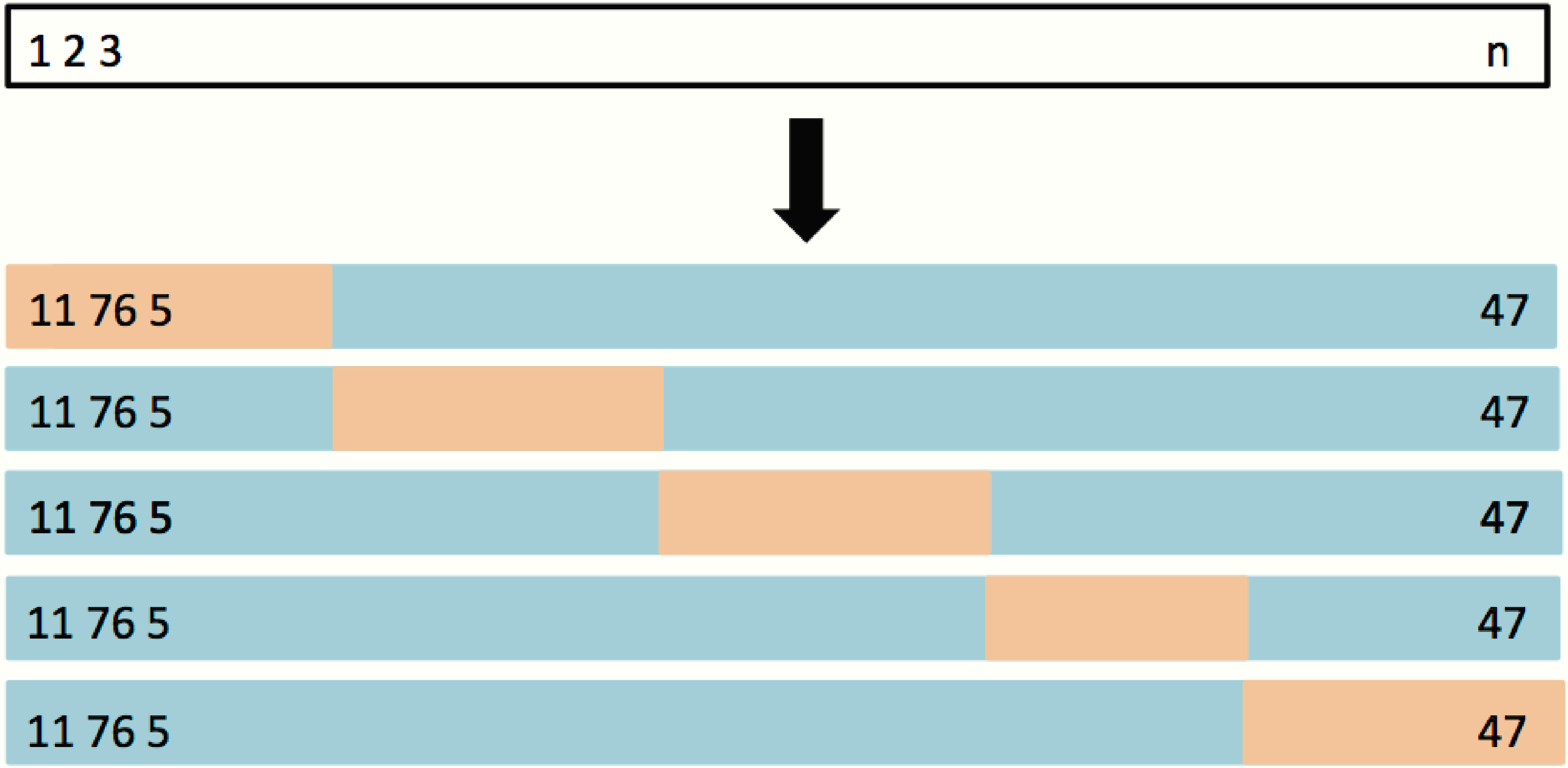
- taking out a different part each time

By averaging the K different MSE's

- **we get an estimated validation (test) error rate**
- **for new observations**

$$CV_{(k)} = \frac{1}{k}\sum_{i=1}^{k} MSE_i.$$

K-fold Cross Validation



Auto Data: LOOCV vs. K-fold CV

Left: LOOCV error curve

Right: 10-fold CV was run many times,

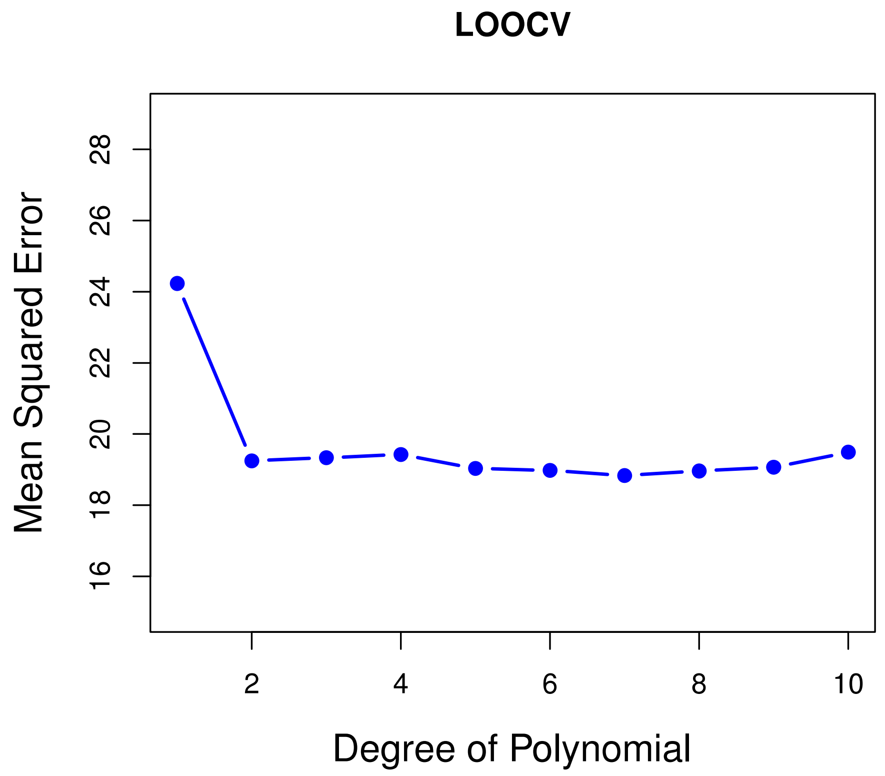
- and the figure shows the slightly different CV error rates

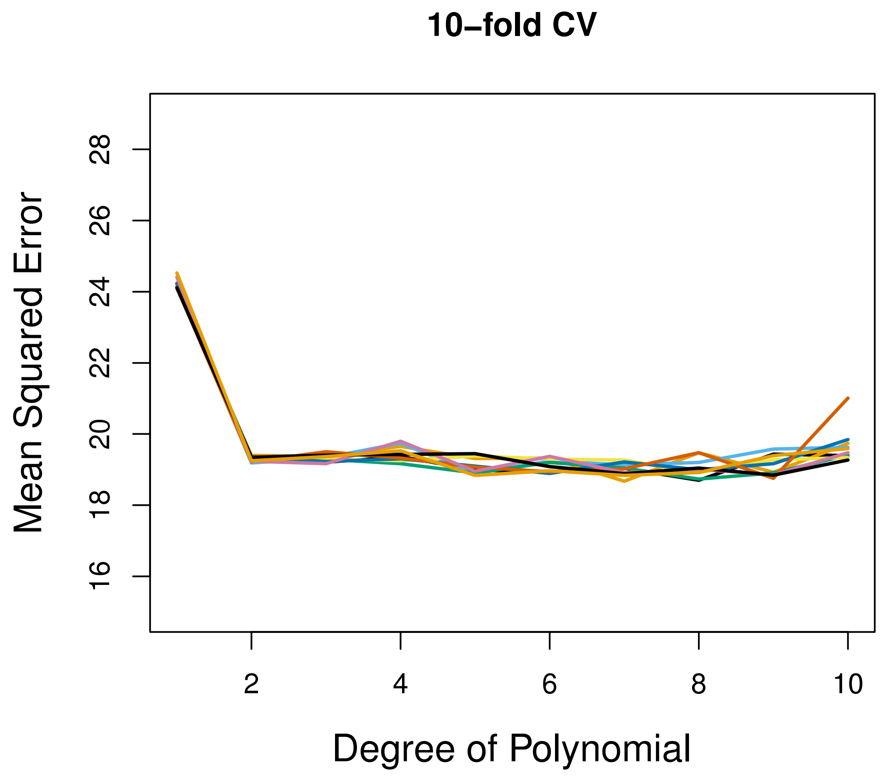
LOOCV is a special case of k-fold,

- where k = n

They are both stable,

- but LOOCV is more computationally intensive!



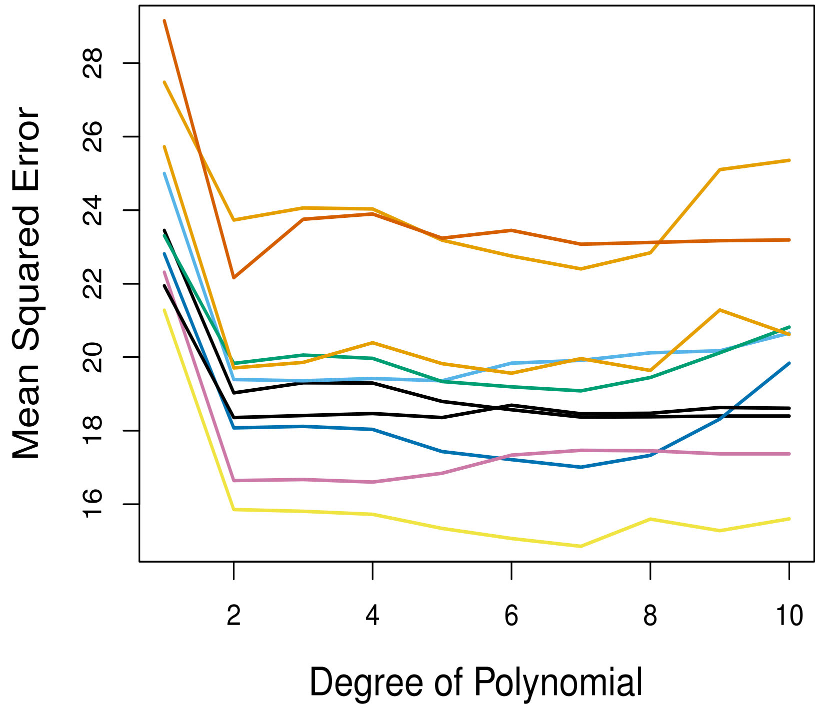


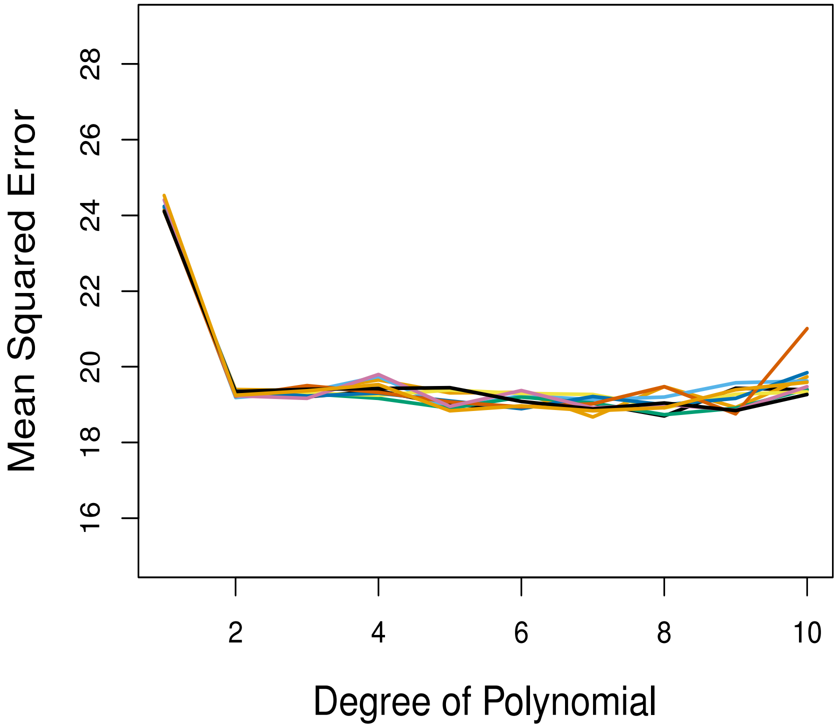
Auto Data: Validation Set Approach vs. K-fold CV Approach

Left: Validation Set Approach

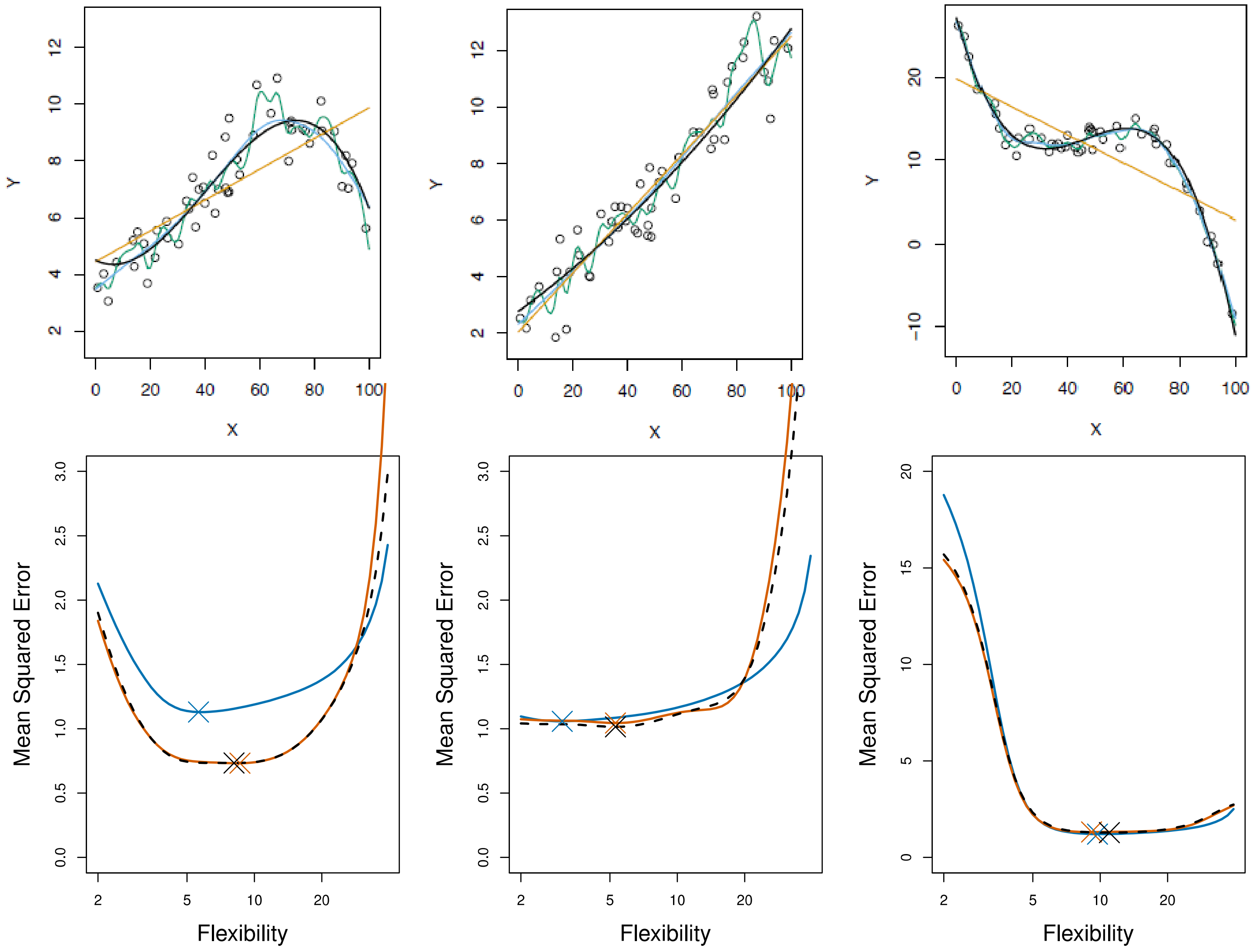
Right: 10-fold Cross Validation Approach

Indeed, 10-fold CV is more stable!





K-fold Cross Validation on Three Simulated Date



Blue: True Test MSE
Black: LOOCV MSE
Orange: 10-fold MSE
Refer to chapter 2 for the top graphs, Fig 2.9, 2.10, and 2.11

5.1.4 Bias- Variance Trade-off for k-fold CV

Putting aside that LOOCV is more computationally intensive than k-fold CV…

Which is better LOOCV or K-fold CV?

- LOOCV is less biased than k-fold CV (when k < n)
- But, LOOCV has higher variance than k-fold CV (when k < n)
- Thus, <u>there is a trade-off between what to use</u>

Conclusion:

- We tend to use k-fold CV with (K = 5 and K = 10)
  These are the magical K's
- It has been empirically shown that they yield test error rate estimates
  that suffer neither from excessively high bias,
  nor from very high variance

5.1.5 Cross Validation on Classification Problems

So far, we have been dealing with CV on regression problems

We can use cross validation in a classification situation in a similar manner

- Divide data into K parts
- Hold out one part,

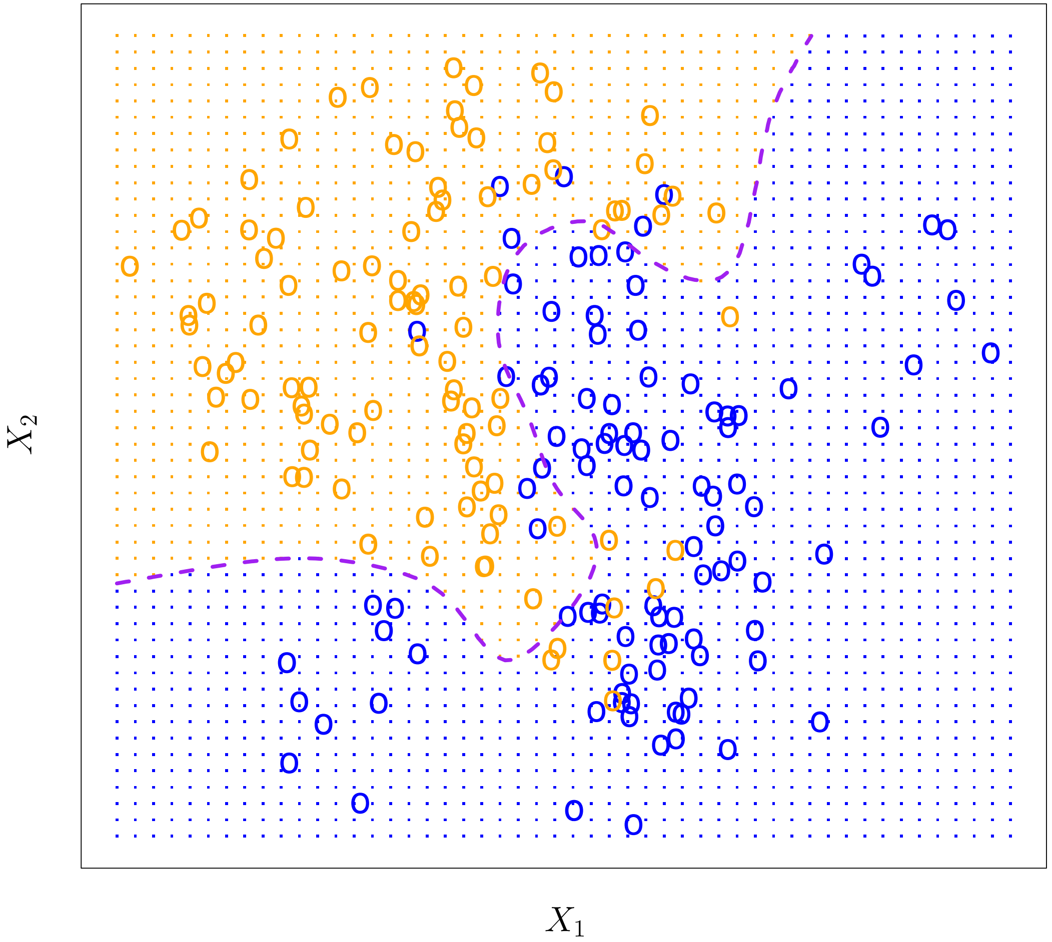
  fit using the remaining data

  and compute the error rate on the hold out data
- Repeat K times
- CV error rate is the average over the K errors we have computed

CV to Choose Order of Polynomial

The data set used is simulated (refer to Fig 2.13)

The purple dashed line is the Bayes' boundary



Bayes' Error Rate: 0.133

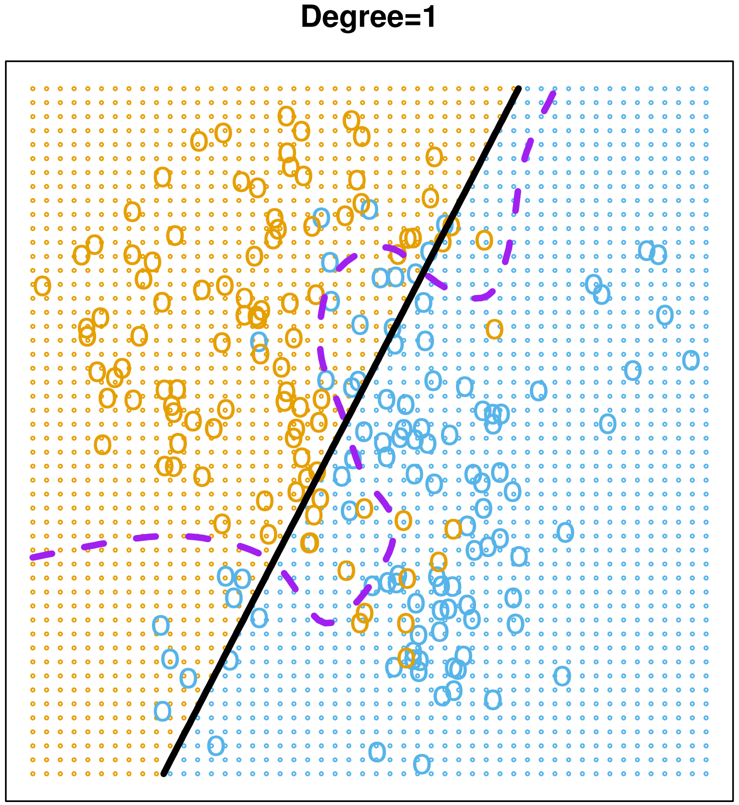
Naive Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naïve) independence assumptions between the features (see Bayes classifier). They are among the simplest Bayesian network models, but coupled with kernel density estimation, they can achieve higher accuracy levels.

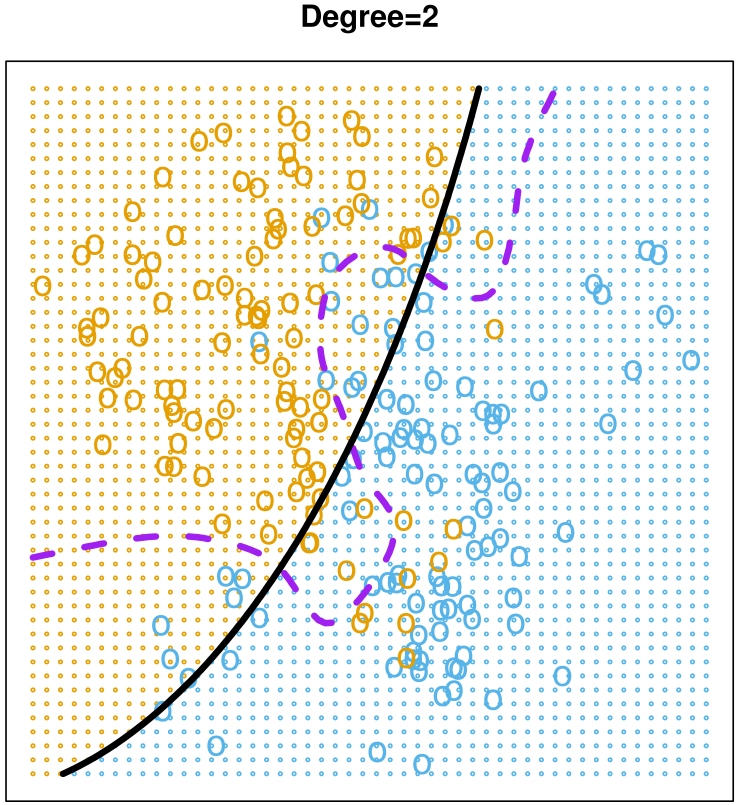
Naïve Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem. Maximum-likelihood training can be done by evaluating a closed-form expression, which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers.

Linear Logistic regression (Degree 1)

- is not able to fit the Bayes' decision boundary

Quadratic Logistic regression does better than linear



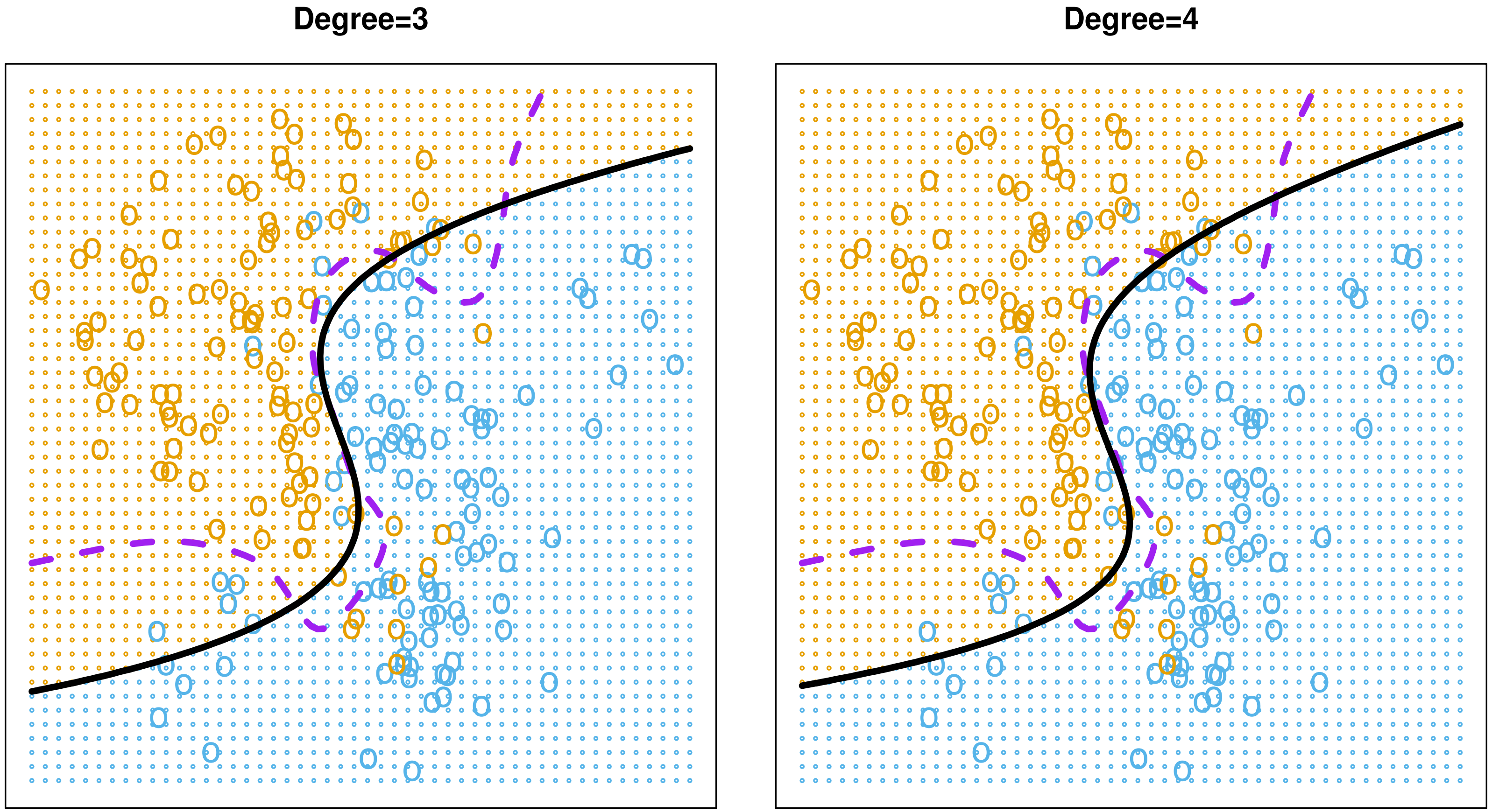


Error Rate: 0.201

Error Rate: 0.197

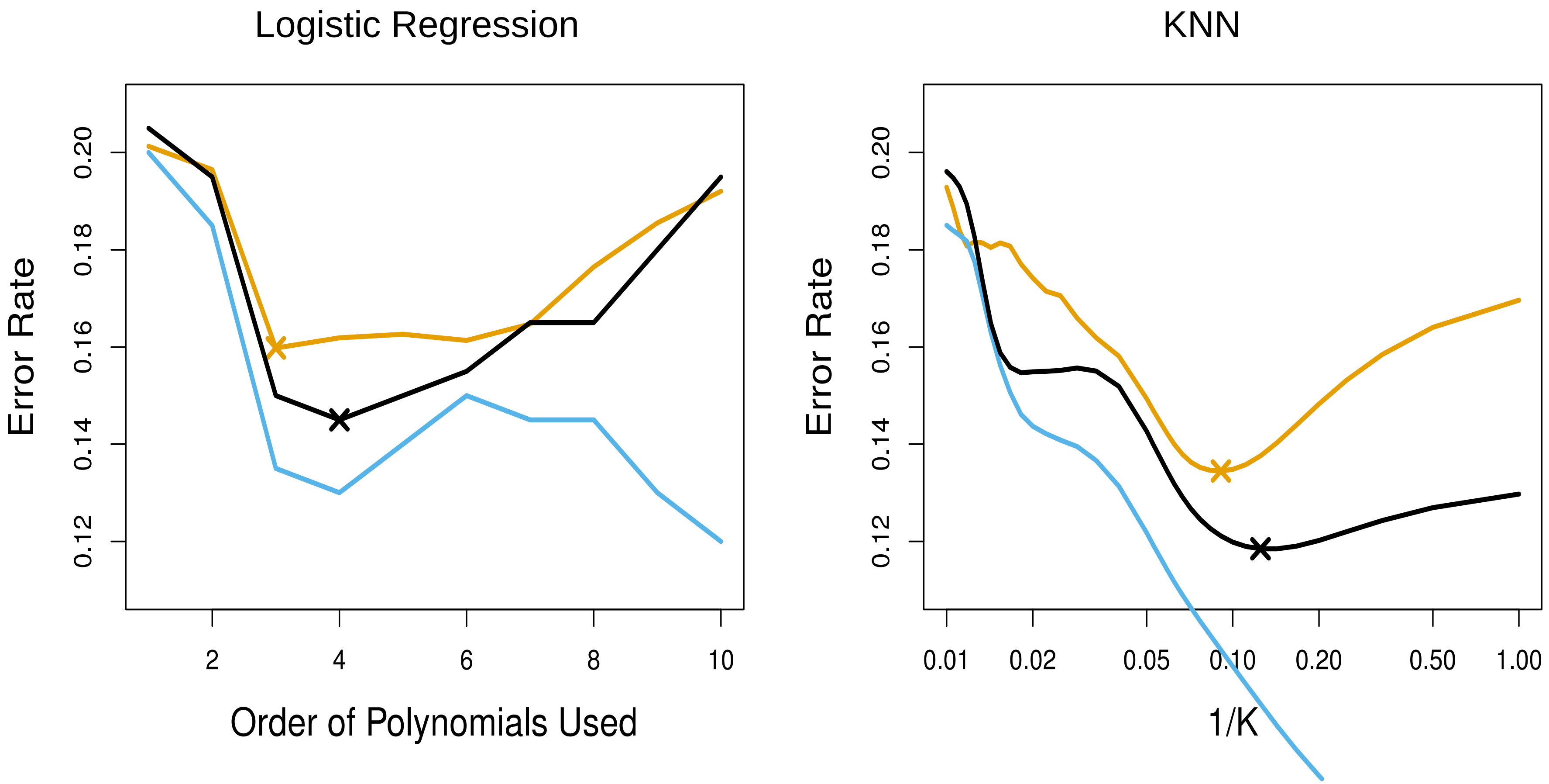
Using cubic and quartic predictors,

- the accuracy of the model improves



Error Rate: 0.160 Error Rate: 0.162

CV to Choose the Order



Brown: Test Error

Blue: Training Error

Black: 10-fold CV Error