Predictive Analytics (ISE529)

Resampling Methods

Dr. Tao Ma
ma.tao@usc.edu

Tue/Thu, May 22 - July 1, 2025, Summer

USC Viterbi
School of Engineering
Daniel J. Epstein Department of Industrial and Systems Engineering

USC University of Southern California

Outline

- Cross-Validation
- The Bootstrap

Resampling Methods

- Resampling methods are an indispensable tool in modern statistics.
- They involve repeatedly drawing samples from a training set and refitting a model of interest on each sample.
- Such an approach may allow us to obtain information that would not be available from fitting the model only once using the original training sample.
- Two of the most commonly used resampling methods are *cross-validation* and the *bootstrap*. For instance,
  - Cross-validation for : model assessment (MSE), model selection
  - Bootstrap for: confidence interval estimate

CROSS-VALIDATION

Training Error vs. Test Error

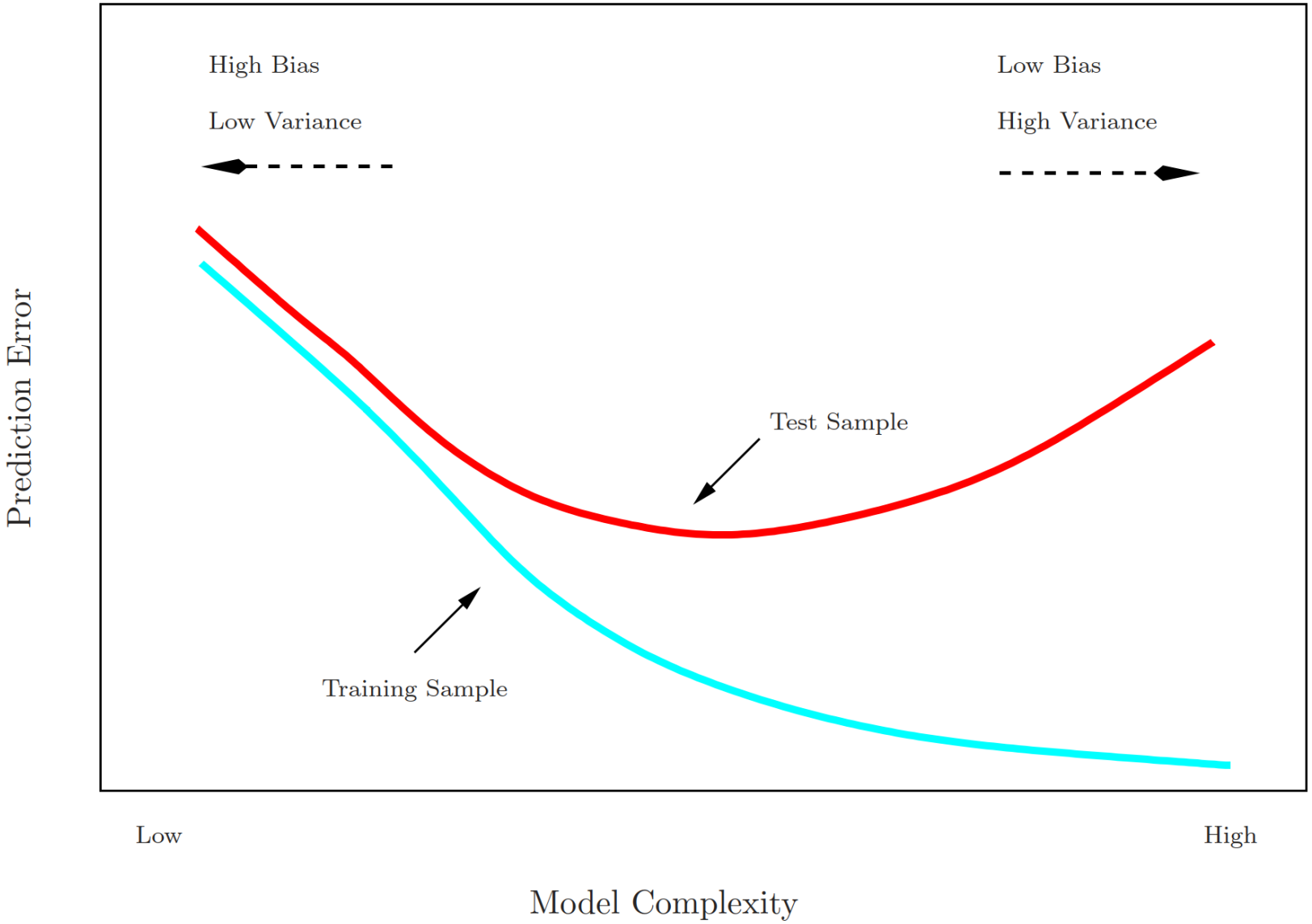
- The test error is the average error that results from using a statistical learning method to predict the response on a new observation, one that was not used in training the method.

  The test error can be easily calculated if a designated test set is available. Unfortunately, this is usually not the case.

- In contrast, the training error can be easily calculated by applying the statistical learning method to the observations used in its training.

  But the training error rate often is quite different from the test error rate, and in particular the former can dramatically underestimate the latter.

Training Error vs. Test Error

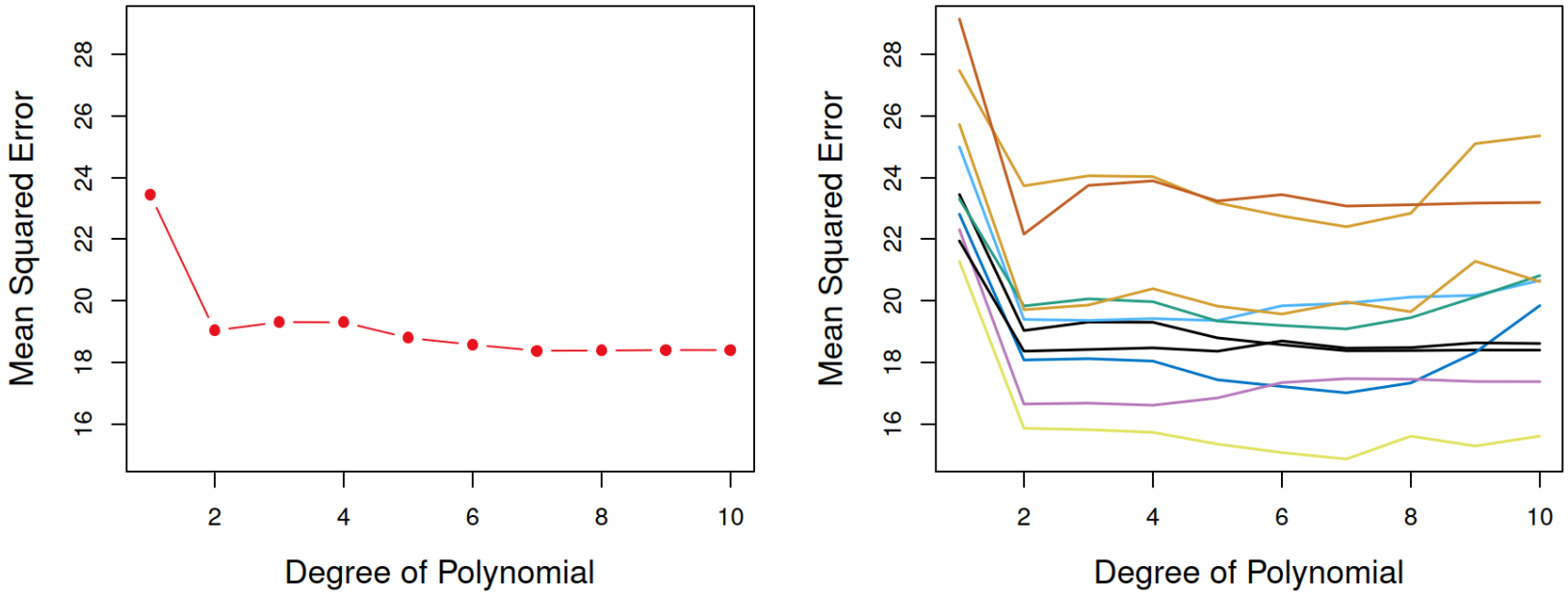


The Validation Set Approach

- Best solution: a large designated test set, but often not available.
- Hence, we consider a class of methods that **estimate the test error** by *holding out* a subset of the training observations from the fitting process. We randomly divide the available set of samples into two parts: a *training set* and a *validation* or *hold-out 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* or *hold-out set*.
- The resulting validation-set error provides an estimate of the test error. This is typically assessed using *MSE* in the case of a quantitative response and *misclassification rate* in the case of a qualitative (discrete) response.

Example: Auto data

We randomly split the 392 observations into two sets, a training set containing 196 of the data points, and a validation set containing the remaining 196 observations.



Left panel shows single split. Right panel shows multiple MSE curves, produced using ten different random splits of the observations into training and validation sets. There is **no consensus** among the curves as to which model results in the smallest validation set MSE.

Issues with Validation Set Approach

- The validation estimate of the test error can be highly variable, depending on precisely which observations are included in the training set and which observations are included in the validation set.
- In the validation approach, only a subset of the observations — those that are included in the training set rather than in the validation set — are used to fit the model.
- Since statistical methods tend to perform worse when trained on *fewer observations*. This suggests that the validation set error may tend to *overestimate* the test error for the model fit on the entire data set.

Cross-Validation

Cross-validation, a refinement of the validation set approach, addresses these two issues.

- Leave-One-Out Cross-Validation (LOOCV)
- *k*-Fold Cross-Validation

Leave-One-Out Cross-Validation

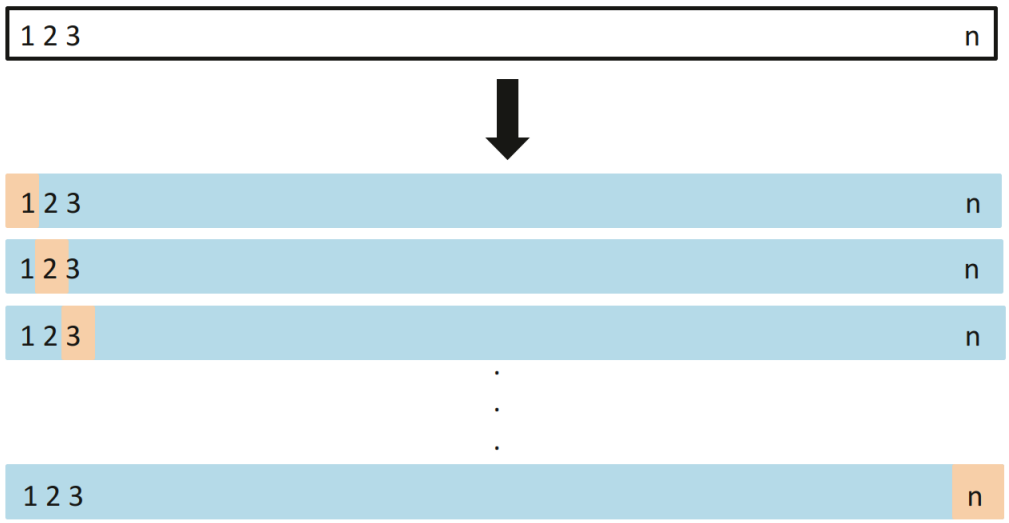
- LOOCV involves splitting the set of observations into two parts, leaves only a single observation (x_1, y_1) out for the validation set, and the remaining $(n - 1)$ observations make up the training set.
- A prediction is made for the excluded single observation to approximate unbiased estimate for the test error.
- Repeating this approach n times until every observation was the validation set once and produces n squared errors.
- The LOOCV estimate for the test MSE is the average of these n test error estimates:

$$\mathrm{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\mathrm{MSE}_i$$

where

$$MSE_i = \left(y_i - \hat{y}_i\right)^2$$

Leave-One-Out Cross-Validation



A schematic display of LOOCV. A set of *n* data points is repeatedly split into a training set (shown in blue) containing all but one observation, and a validation set that contains only that single observation (shown in beige). The test error is then estimated by averaging the *n* resulting *MSE*s.

Advantages of LOOCV

- It has far less bias. we repeatedly fit the model using training sets that contain $(n-1)$ observations, almost as same as the entire data set. Consequently, the LOOCV approach tends not to overestimate the test error rate
- LOOCV will always yield the same results: there is no randomness in the training/validation set splits.
- LOOCV has the potential to be expensive to implement, since the model must be fit n times. This can be very time consuming if n is large.
- In the context of **least squares regression (only)** as it does not hold in general, a shortcut formula makes the cost of LOOCV the same as that of a single model fit.

$$\mathrm{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1-h_i}\right)^2$$

- where $\hat{y}_i$ is the ith **fitted** value from the original least squares fit, h_i is diagonal elements of Hat Matrix, the variance of the fitted value.

k-Fold Cross-Validation

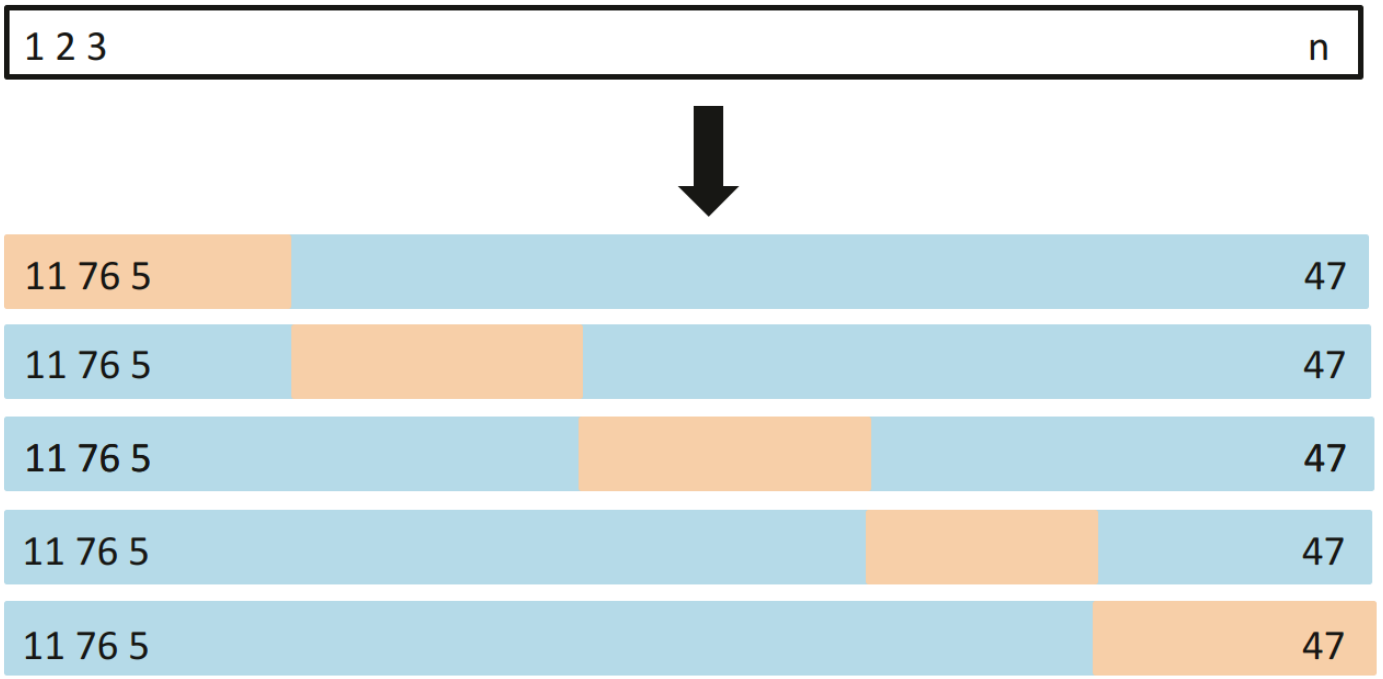
- An alternative to LOOCV is *k*-fold CV.
- Estimates can be used to select best model, and to give an idea of the test error of the final chosen model.
- This approach involves randomly dividing the set of observations into *k* groups, or folds, of approximately equal size.
- We leave out part *k* as a validation set, fit the model to the remaining $(K-1)$ parts (combined as the training set), and then obtain predictions and MSE for the left-out *k*th part.

k-Fold Cross-Validation

- This procedure is repeated *k* times; each time, a different group of observations is treated in turn for $k = 1, 2, \ldots K$ as a validation set.
- This process results in *k* estimates of the test error, MSE_1, $MSE_2, \ldots, MSE_k$. The *k*-fold CV estimate is computed by averaging these values,

$$\mathrm{CV}_{(k)} = \frac{1}{k} \sum_{i=1}^{k} \mathrm{MSE}_i$$

k-Fold Cross-Validation

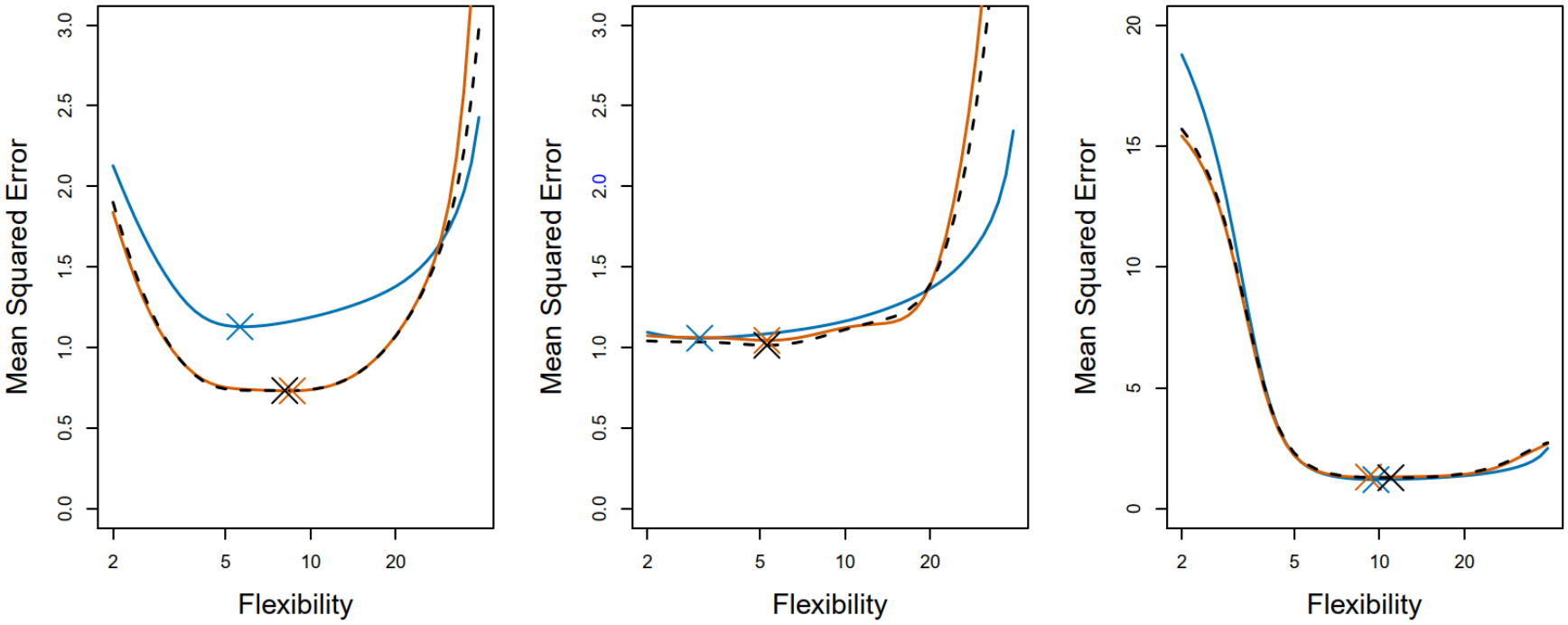


A schematic display of 5-fold CV. A set of *n* observations is randomly split into five non-overlapping groups. Each of these fifths acts as a validation set (shown in beige), and the remainder as a training set (shown in blue). The test error is estimated by averaging the five resulting MSE estimates.

k-Fold Cross-Validation

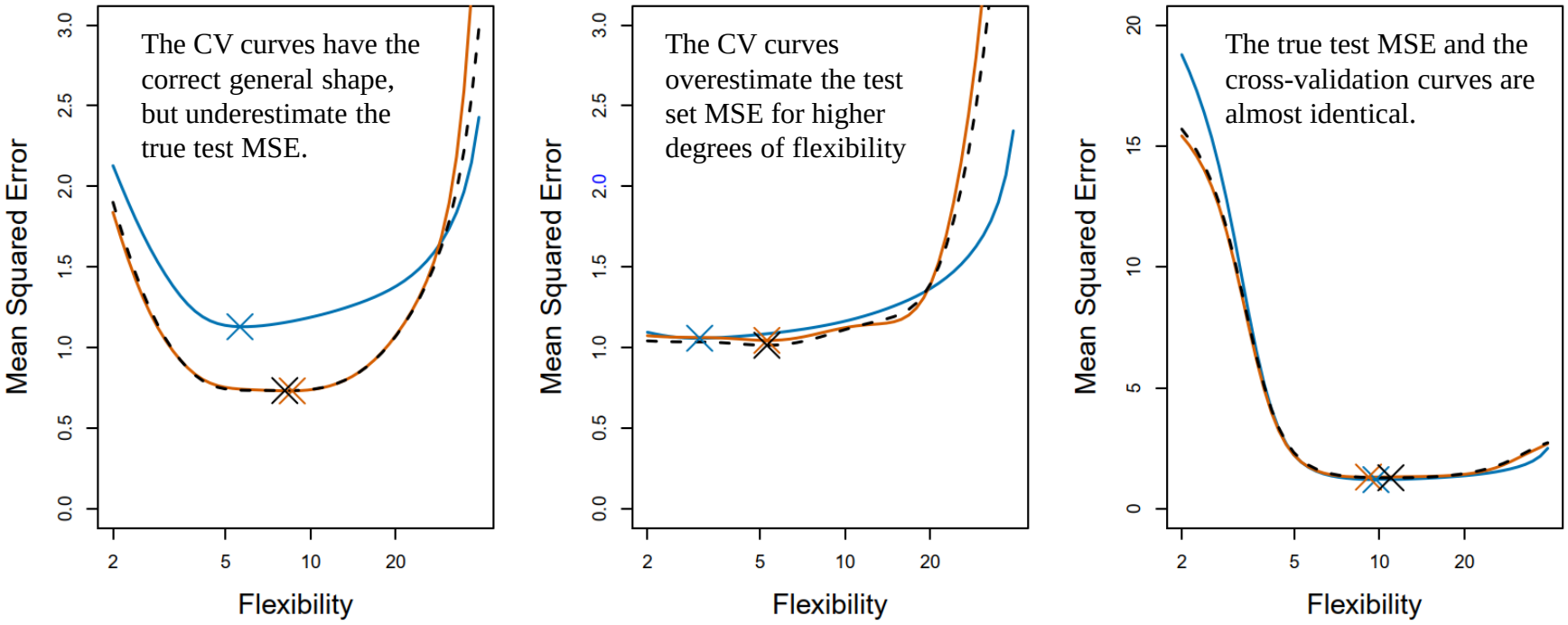
- It is not hard to see that LOOCV is a special case of *k*-fold CV in which *k* is set to equal *n*.
- In practice, one typically performs *k*-fold CV using $k = 5$ or $k = 10$. What is the advantage of using $k = 5$ or $k = 10$ rather than $k = n$?
- Cross-validation is a very general approach that can be applied to almost any statistical learning method. Some statistical learning methods have computationally intensive fitting procedures, and so performing LOOCV may pose computational problems, especially if *n* is extremely large.
- In contrast, performing 10-fold CV requires fitting the learning procedure only ten times, which may be much more feasible.

Accuracy of the CV estimate



When we examine real data, we do not know the true test MSE, so it is difficult to determine the accuracy of the cross-validation estimate. However, if we examine simulated data, then we can compute the *true* test MSE, and can thereby evaluate the accuracy of our cross-validation results.

Accuracy of the CV estimate



The true test MSE is shown in blue, the LOOCV estimate is shown as a black dashed line, and the 10-fold CV estimate is shown in orange. In all three plots, the two cross-validation estimates are very similar. The crosses indicate the minimum of each of the MSE curves.

Two Goals of the CV estimate

- Model assessment (MSE)

  When we perform cross-validation, our goal might be to determine how well a given statistical learning procedure can be expected to perform on independent data; in this case, the actual estimate of the test MSE is of interest.

- Model selection

  But at other times we are interested only in the location of the minimum point in the estimated test MSE curve.

  This is because we might be performing cross-validation on a number of statistical learning methods, or on a single method using different levels of flexibility, in order to identify the method that results in the lowest test error.

  For this purpose, the location of the minimum point in the estimated test MSE curve is important, but the actual value of the estimated test MSE is not.

Bias-Variance Trade-Off for *k*-Fold CV

- *k*-fold CV gives more accurate estimates of the test error rate than does LOOCV due to a bias-variance trade-off.
- Empirically, using $k = 5$ or 10 for *k*-fold cross-validation yields test error rate estimates that suffer **neither** from excessively high bias **nor** from very high variance.

From the perspective of bias reduction,

- LOOCV gives approximately **unbiased** estimates of the test error, since each training set contains $(n-1)$ observations, which is almost as many as the number of observations in the full data set.
- *k*-fold CV for, say, k = 5 or k = 10 will lead to an **intermediate level of bias**, since each training set contains approximately $(k-1)n/k$ observations — fewer than in the LOOCV approach.

Bias-Variance Trade-Off for *k*-Fold CV

- LOOCV has **higher variance** than does *k*-fold CV with $k < n$.

From the perspective of variance

- When we perform LOOCV, we are in effect averaging the outputs of *n* fitted models, each of which is trained on an **almost identical** set of observations; therefore, these outputs are highly (positively) **correlated** with each other.
- In contrast, when we perform *k*-fold CV with $k < n$, we are averaging the outputs of *k* fitted models that are somewhat **less correlated** with each other, since the overlap between the training sets in each model is smaller.
- Since the mean of many highly **correlated** quantities has higher variance than does the mean of many quantities that are not as highly correlated,
- Hence, the test error estimate resulting from LOOCV tends to have higher variance than does the test error estimate resulting from *k*-fold CV.

k-Fold CV on Classification

In classification setting, we use the number of misclassified observations. For instance, the LOOCV error rate takes the form

$$\mathrm{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\mathrm{Err}_i$$

where

$$\mathrm{Err}_i = I(y_i \neq \hat{y}_i)$$

The *k*-fold CV error rate takes the form

$$\mathrm{CV}_K = \sum_{k=1}^{K}\frac{n_k}{n}\mathrm{Err}_k$$

where

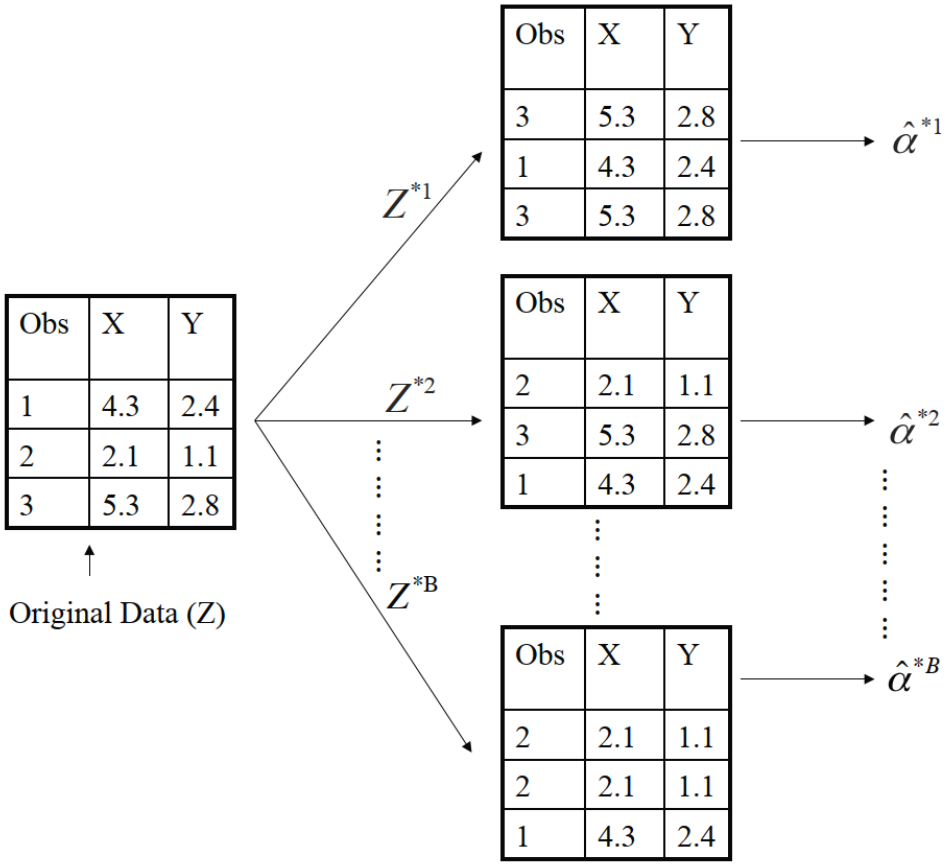
$$\mathrm{Err}_k = \sum_{i\in C_k} I(y_i \neq \hat{y}_i)/n_k$$

THE BOOTSTRAP

The Bootstrap

- The bootstrap procedure is to obtain distinct data sets by repeatedly sampling observations from the original data set with replacement.
- Each of these "bootstrap data sets" is created by sampling with replacement and is the same size as our original dataset. As a result, some observations may appear more than once in a given bootstrap data set and some not at all.
- The bootstrap is a flexible and powerful statistical tool that can provide an estimate of the standard error of a coefficient, or a confidence interval for that coefficient.

Example



A graphical illustration of the bootstrap approach on a small sample containing $n = 3$ observations. Each bootstrap data set contains n observations, sampled with replacement from the original data set. Each bootstrap data set is used to obtain an estimate of α.

Example

- We wish to determine the best investment allocation for an investment.
- Suppose that we wish to invest a fixed sum of money in two financial assets that yield returns of X and Y , respectively, where X and Y are random quantities. We will invest a fraction α of our money in X, and will invest the remaining 1 − α in Y.
- Since there is variability associated with the returns on these two assets, we wish to choose α to minimize the total risk, or variance, of our investment. In other words, we want to minimize Var($\alpha X + (1 - \alpha)Y$). The value that minimizes the risk is given by

$$\alpha = \frac{\sigma_Y^2 - \sigma_{XY}}{\sigma_X^2 + \sigma_Y^2 - 2\sigma_{XY}}$$

- where $\sigma_X^2 = \text{Var}(X)$, $\sigma_Y^2 = \text{Var}(Y)$, and $\sigma_{XY} = \text{Cov}(X, Y)$.

Example: Simulation vs. Bootstrap

- For comparison, we use two methods to generate samples to estimate α.
- We repeatedly **simulate** 100 paired observations of X and Y for 1000 times and estimate α 1,000 times. For these simulations, the parameters were set to $\sigma_X^2 = 1$, $\sigma_Y^2 = 1.25$, and $\sigma_{XY} = 0.5$, and so we know that the true value of α is 0.6 (in the original population).
- The mean over all 1,000 estimates for α is

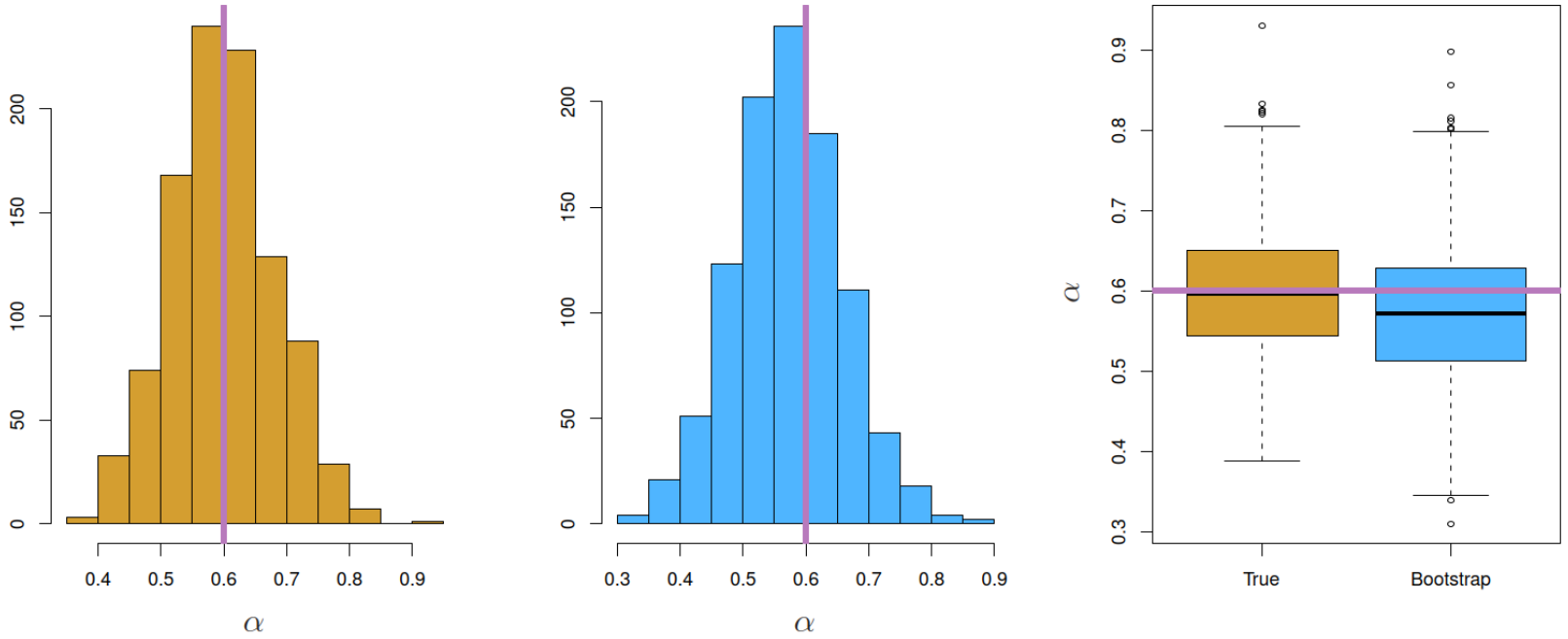
$$\bar{\alpha} = \frac{1}{1000} \sum_{r=1}^{1000} \hat{\alpha}_r = 0.5996.$$

- Very close to α = 0.6, and the standard deviation of the estimates is

$$\sqrt{\frac{1}{1000-1} \sum_{r=1}^{1000} (\hat{\alpha}_r - \bar{\alpha})^2} = 0.083.$$

- Use the **bootstrap** approach to obtain 1000 new data sets from a single original data set.

Example



Left: A histogram of the estimates of α obtained by generating 1,000 simulated data sets from the true population. **Center**: A histogram of the estimates of α obtained from 1,000 bootstrap samples from a single data set. **Right**: The estimates of α displayed in the left and center panels are shown as boxplots. In each panel, the pink line indicates the true value of α.