Resampling Methods

- The generalization performance of a learning method relates to its prediction capability on independent test data

- Assessment of this performance is extremely important in practice, since it guides the choice of learning method or model. There are two types of assessment goals we typically consider

  - Model Assessment: estimate test error rates
  - Model Selection: select the appropriate level of model flexibility

- If we have sufficient data, the best approach for both problems is to randomly divide the dataset into three parts: a training set, a validation set, and a test set.

- The training set is used to fit the models; the validation set is used to estimate prediction error for model selection; the test set is used for assessment of the generalization error of the final chosen model

- A typical split might be 50% for training, and 25% each for validation and testing:

- But when data is not enough to “waist” on validation/testing we rely on resampling

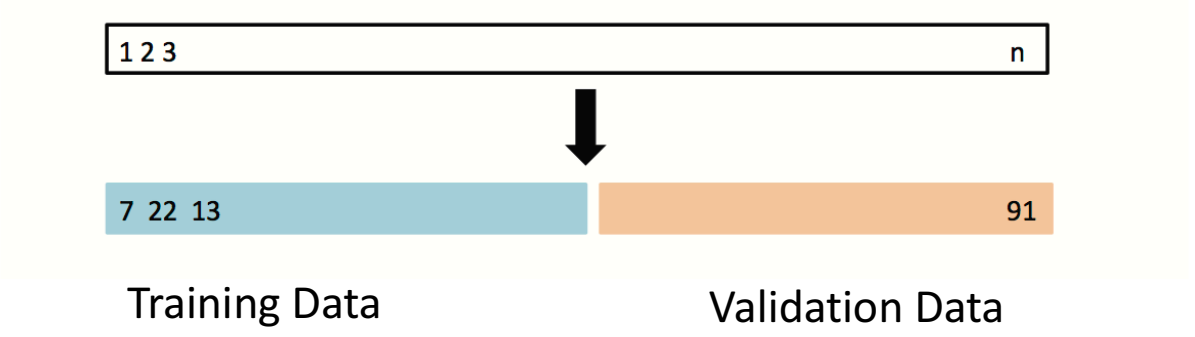
- A resampling method involves repeatedly drawing samples from a training set and refitting a model of interest on each sample in order to obtain more information about the fitted model

- The process is computationally expensive but this is less and less a problem

- We consider two resampling methods:
  - Cross Validation
  - Bootstrap

(5.1.1) The Validation Sample 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 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



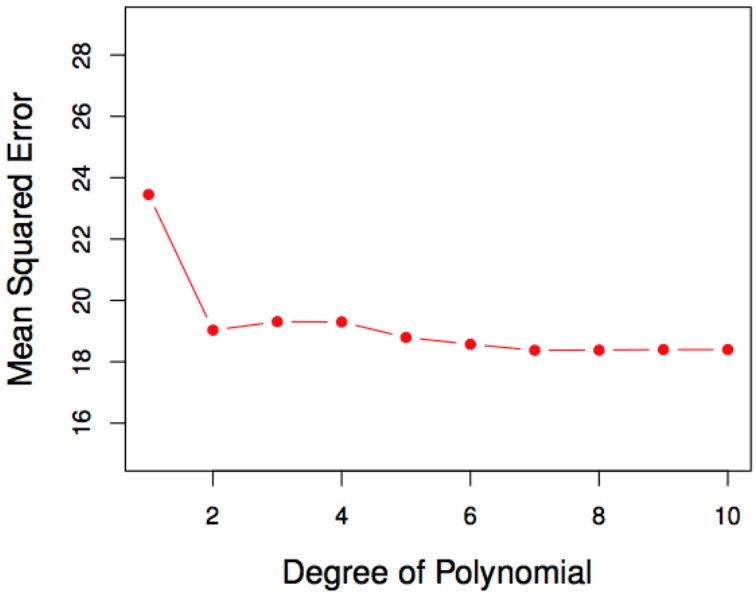
Example. The `Auto` data

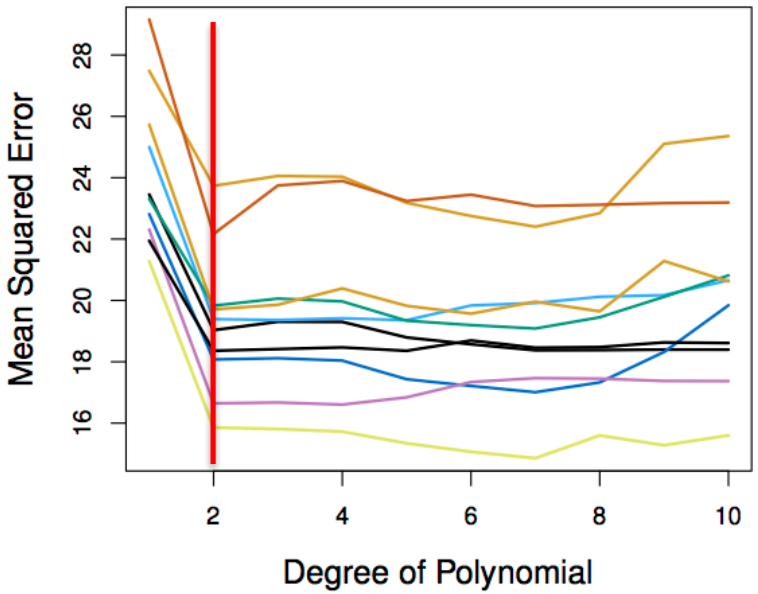
- Suppose that we want to predict mpg from horsepower
- Two models:
  - `mpg ~ horsepower`
  - `mpg ~ horsepower + horsepower`^{2}

- 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 MSE is the winner.

- The plot on left shows the test MSE for one random split graphed against the increasing degree of the polynomial of horsepower
- Assume we apply this procedure 10 times and for each split estimate the test MSE – the plot on right
- There is a lot of variability which could invalidate our choice of degree





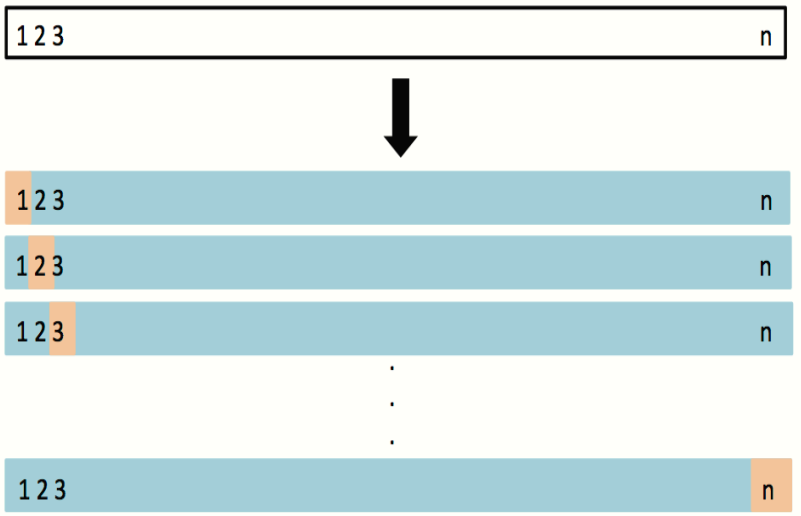
Characteristics of the validation set approach

- Advantages
  - Simple
  - Easy to implement
- Disadvantages
  - The validation/test MSE can be highly variable
  - Only a subset of observations is used to fit the model (training data). Statistical methods tend to perform worse when trained on fewer observations

Leave-One-Out Cross Validation (LOOCV)

- Similar to the Validation Set Approach, but addresses its disadvantages
- For each suggested model
  - Split the data set of size n into
    - Training data set (blue) size: $n - 1$
    - Validation data set (orange) size: 1
  - Fit the model using the training data
  - Validate model using the validation data, and compute the corresponding MSE_i
  - Repeat this process n times
  - The MSE for the model is computed as an average

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



Compare LOOCV to Validation Set

- LOOCV has less bias
  - We repeatedly fit the statistical learning method using training data that contains $n - 1$ observations, 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 observation each time
- LOOCV is computationally intensive (not good)
  - We fit each model n times
  - Can be avoided for linear models (and only them!) using a "shortcut"

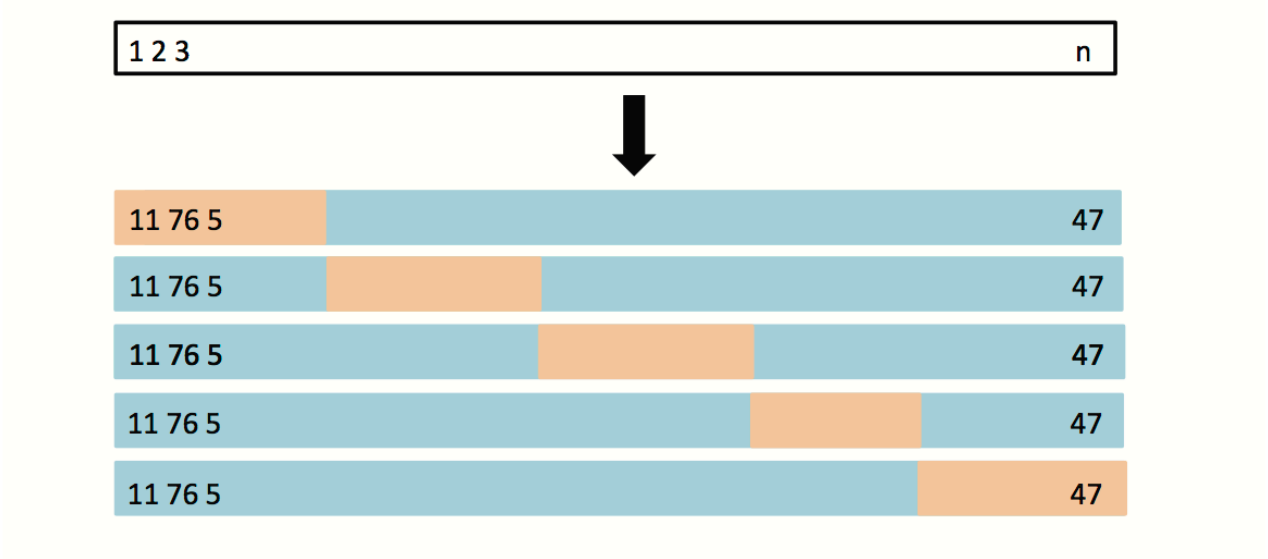
$$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1 - h_i}\right)^2$$

K-fold Cross Validation (5.1.3)

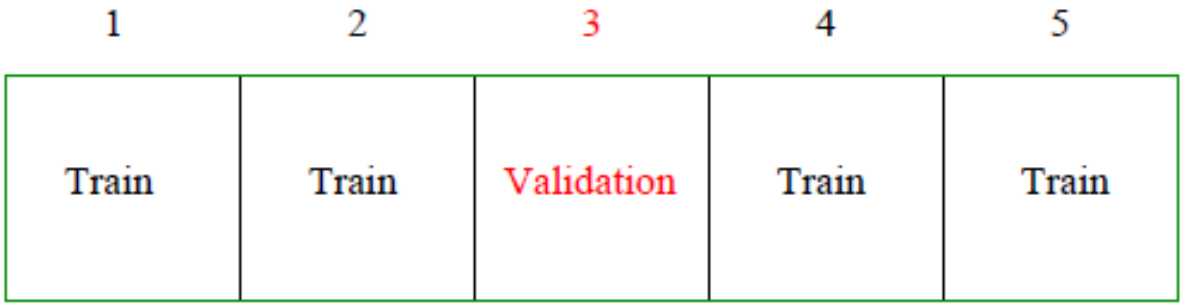
- LOOCV is computationally intensive, so we can run k-fold Cross Validation instead
- With k-fold Cross Validation, we divide the data set into K different parts (e.g. $K = 5$, or $K = 10$, etc.)
- We then remove the first part, fit the model on the remaining $K - 1$ parts, and see how good the predictions 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

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

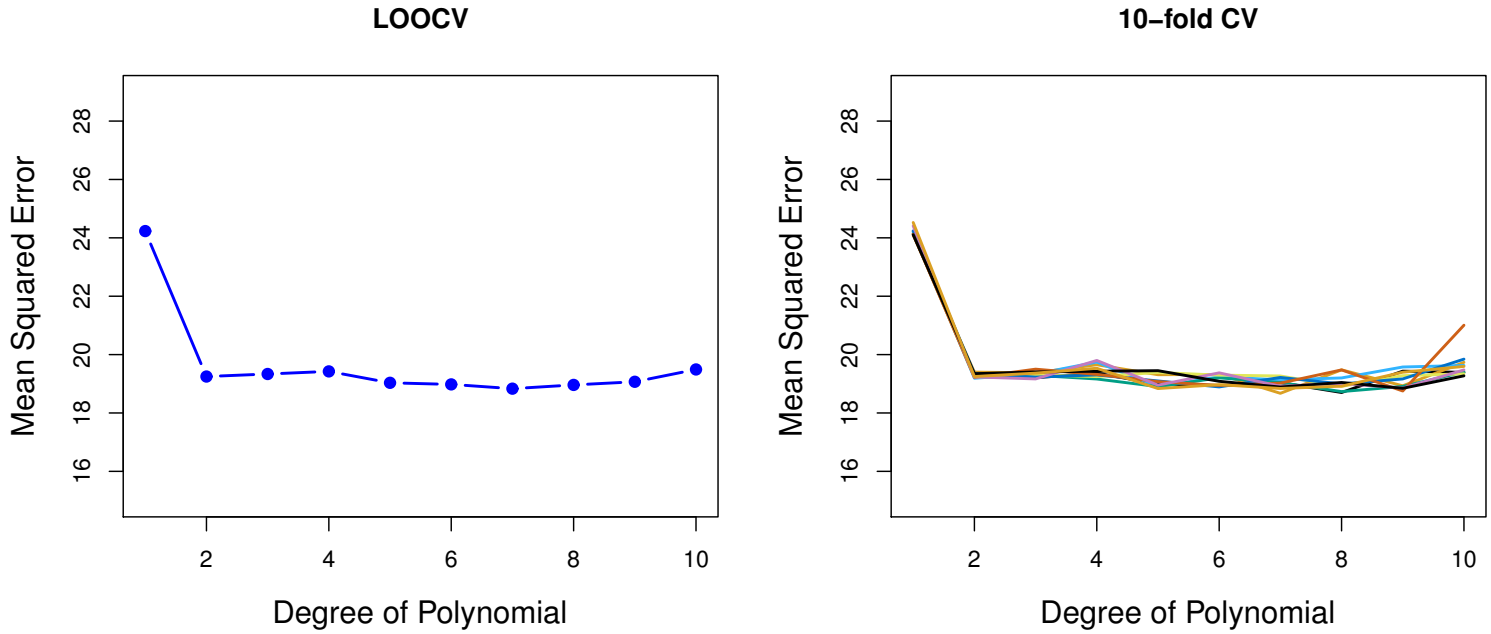
K-fold Cross Validation schematically shown:



- The set with n observations is randomly split into 5 parts (1st one on the left contains observations # 11, 76 and 5 of the original order). We could pick the 3rd part for validation



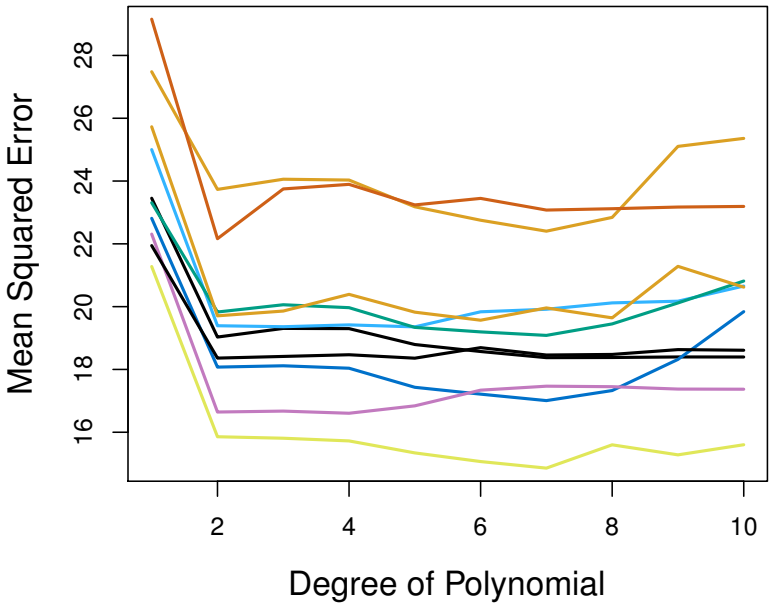
Data `Auto`: LOOCV vs. K-fold CV

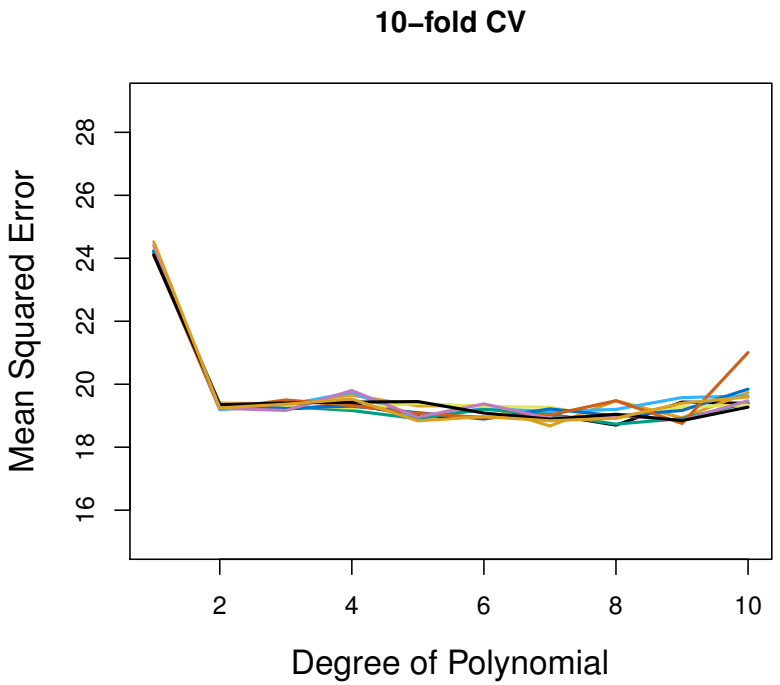


- Left plot: LOOCV error curve
- Right plot: 10-fold CV was run 9 times, and the figure shows the slightly different CV error rates
- LOOCV is a special case of k-fold, where $k = n$
- They are both stable, but LOOCV is more computationally intensive

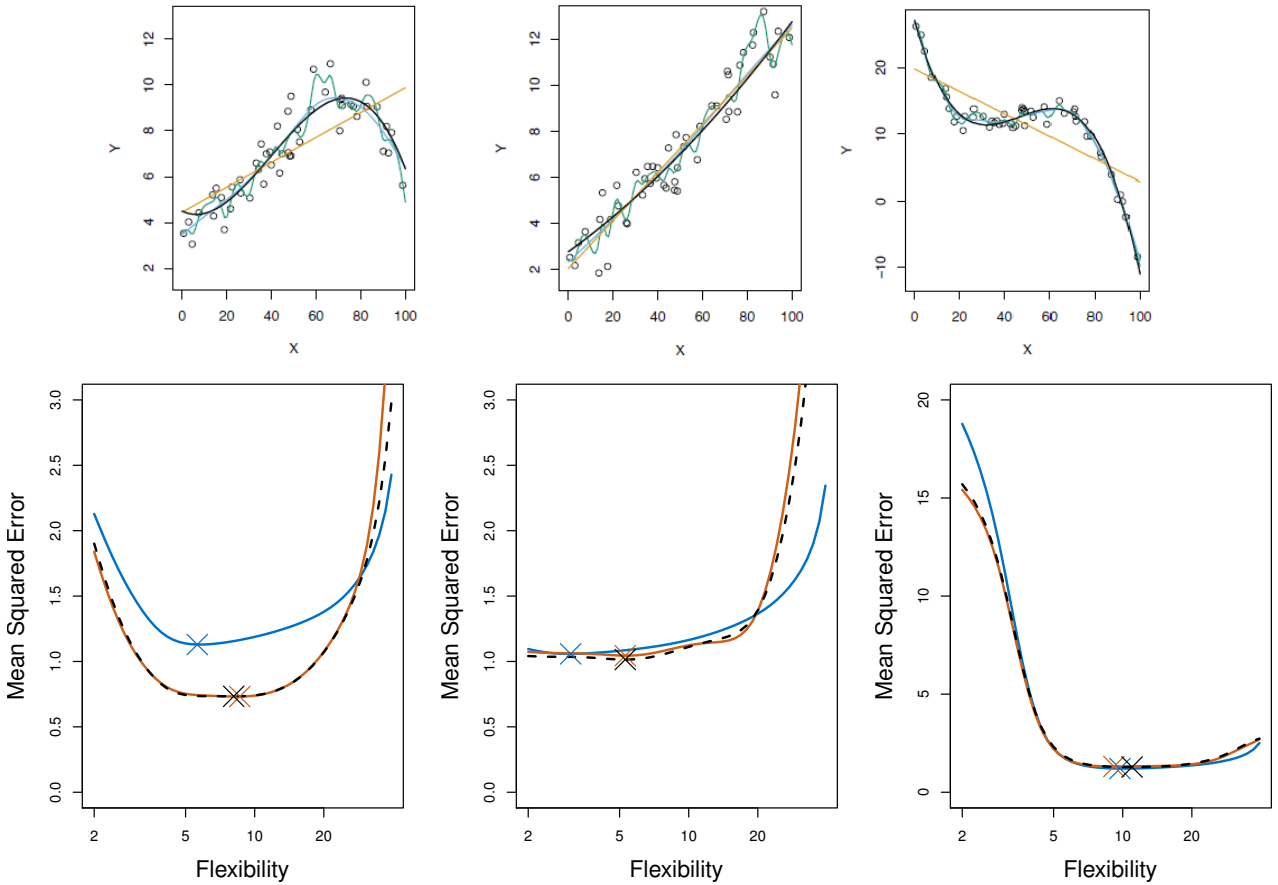
Data `Auto`: Validation Set Approach vs. K-fold CV Approach

- Left: Validation Set approach
- Right: 10-fold Cross Validation approach
- The 10-fold CV is more stable than the Validation Set method and is close (previous slide) to LOOCV





K-fold Cross Validation on Three Simulated Data Sets



- Blue: True Test MSE, **Black**: LOOCV MSE, Orange: 10-fold MSE
  (The top 3 graphs, are Fig 2.9, 2.10, and 2.11 from ISLR, Chap 2, pp.31-34)
- All of the CV curves come close to identifying the correct level of flexibility

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

- LOOCV is more computationally intensive than k-fold CV but which is better LOOCV or K-fold CV?

- Since there is no randomness in the training/validation set splits for LOOCV ($k = n$) its estimation of the test error is less biased than the k-fold CV ($k < n$) estimation

- But, LOOCV has higher variance than k-fold CV (when $k < n$)

- The reason is that each of the n models trained in the case of LOOCV are trained on almost the same dataset and thus the results are highly positively correlated. Thus, the variance is larger than for k-fold CV where the correlation is much lower

- There is a trade-off associated with the choice of K but empirically it's shown that $K = 5$ and $K = 10$ are the best choices

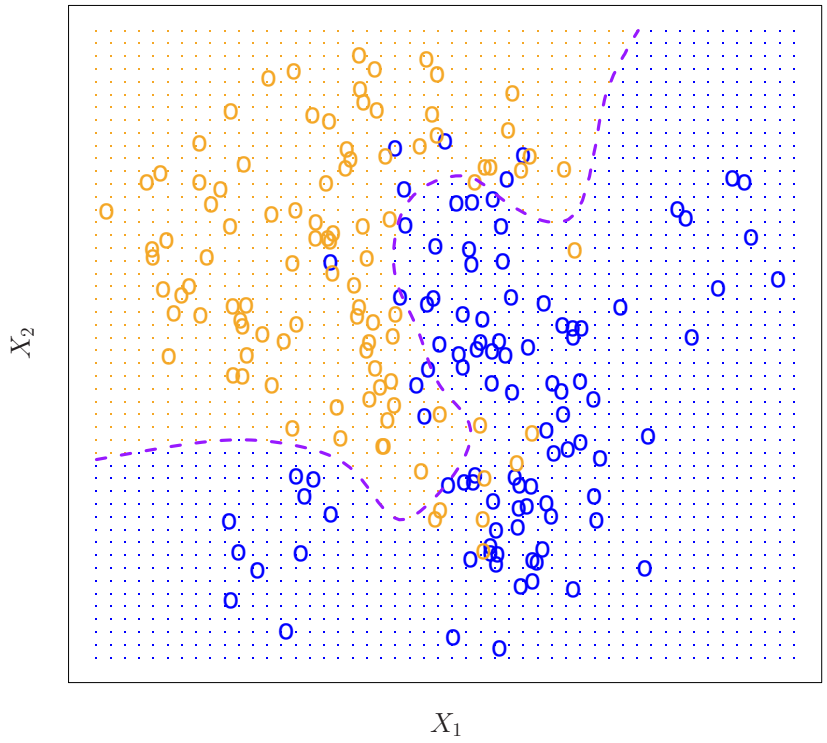
Cross Validation on Classification Problems (5.1.5)

- 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 Err_i on the hold out data (part i)
  - Repeat K times
  - CV error rate is the average over the K errors we have computed

$$CV_{(k)} = \frac{1}{K}\sum_{i=1}^{K} Err_i = \frac{1}{K}\sum_{i=1}^{K}\sum_{\substack{all\ m\ in \\ part\ i}} I(y_m \neq \hat{y}_m)$$

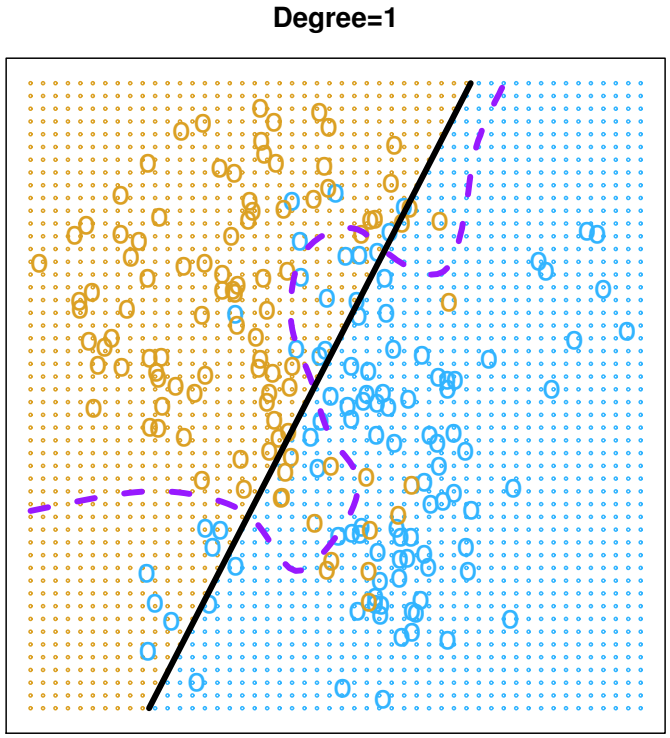
CV to Choose Order of Polynomial

- The data set is simulated (ESLII: Fig 2.13)
- The purple dashed line is the Bayes' boundary

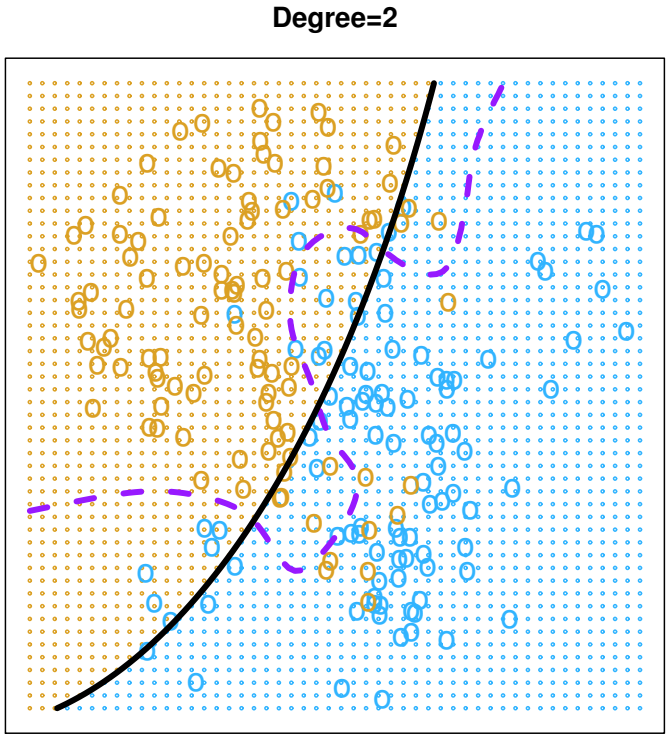


Bayes' Error Rate: 0.133

- Linear Logistic regression (Degree 1) is not able to fit the Bayes' decision boundary
- Quadratic Logistic regression does better than linear

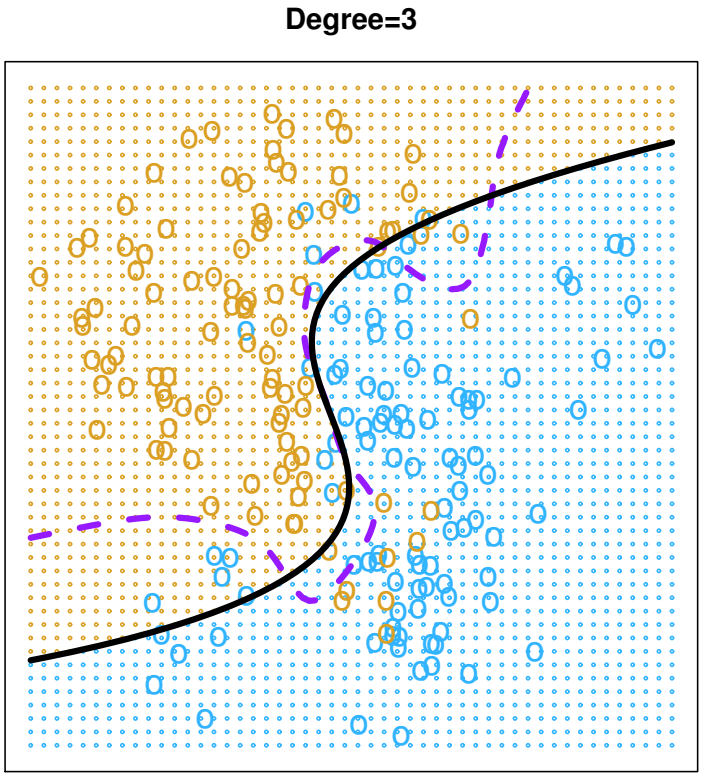


Error Rate: 0.201

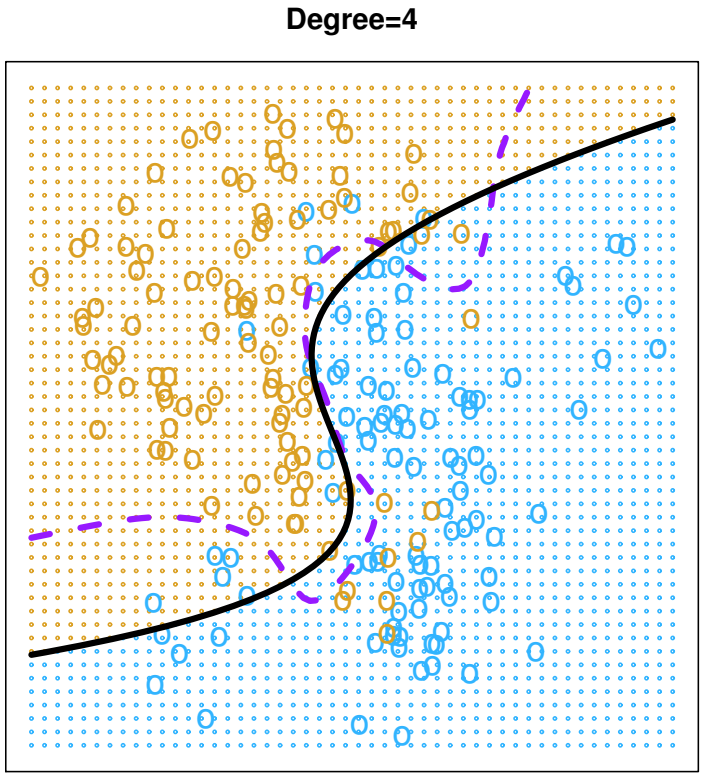


Error Rate: 0.197

- Using cubic and quartic predictors, the accuracy of the model improves

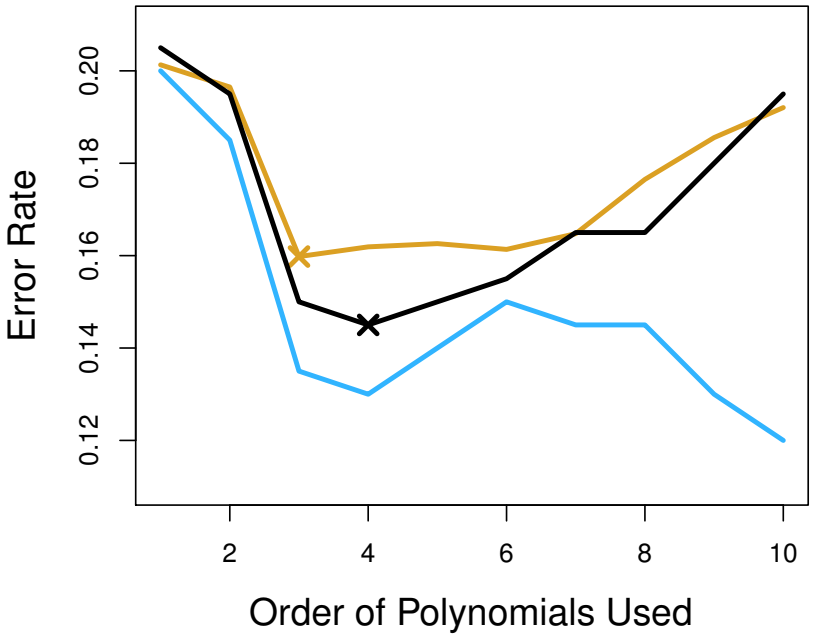


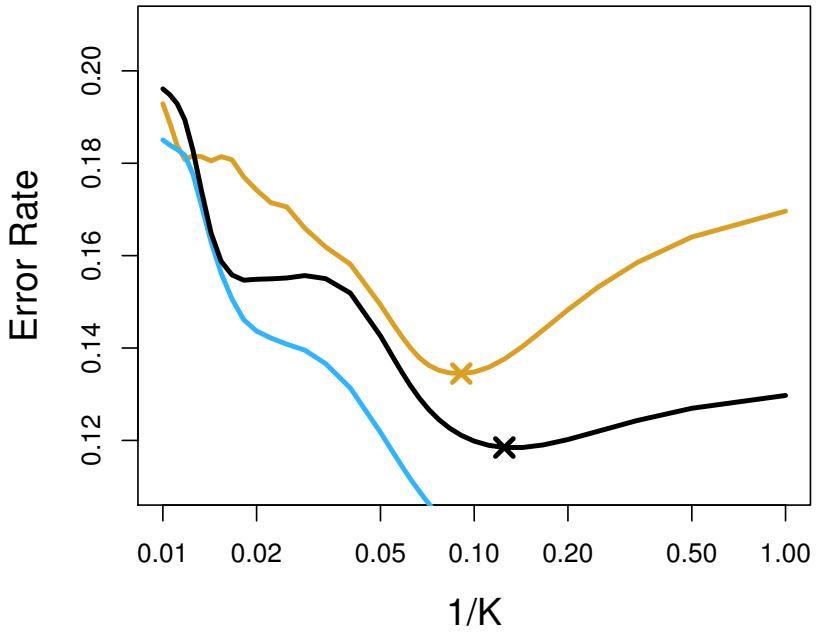
Error Rate: 0.160



Error Rate: 0.162

CV to Choose the Order





- Brown: Test Error, Blue: Training Error, **Black**: 10-fold CV Error
- The 10-fold CV underestimates the test error but reaches min close to the best value of K

The Bootstrap (5.2)

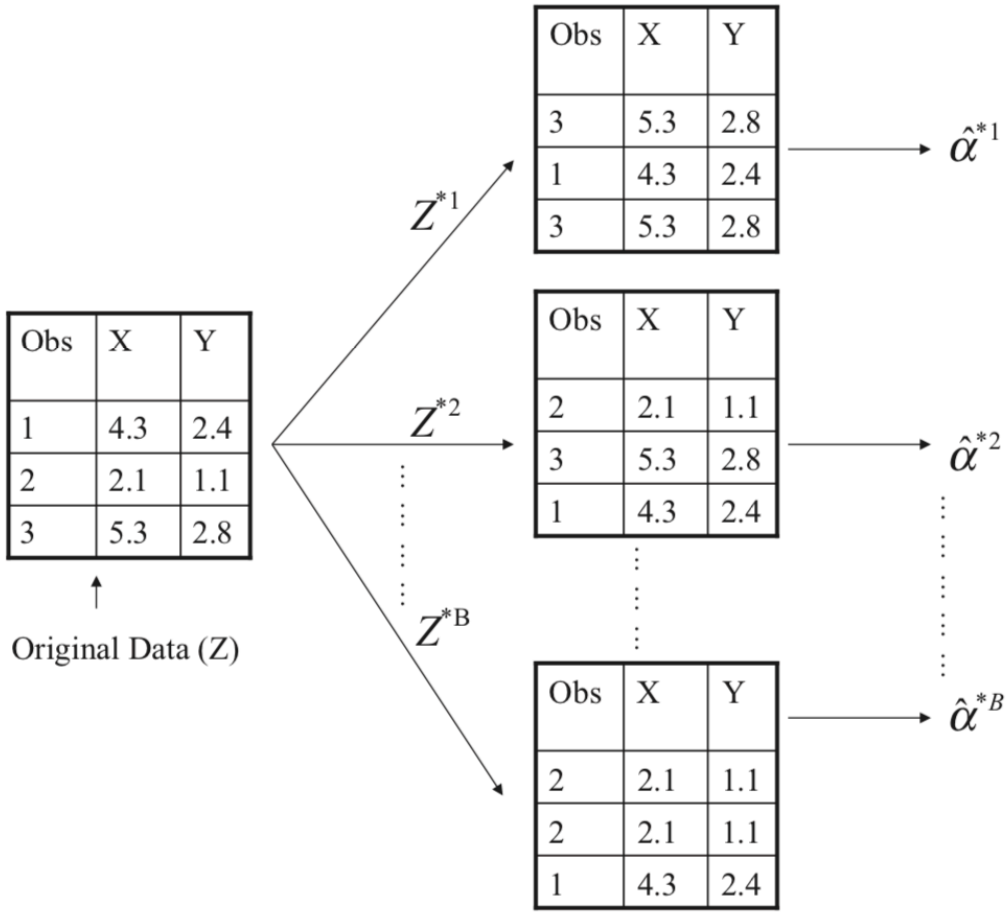
- Powerful approach used to quantify the uncertainty associated with a given estimator or statistical learning method
- For example, we could estimate the standard errors of the coefficients of a linear regression fit
- The real value of the bootstrap is that it's easy to apply to many statistical methods where there is no other way to estimate variability
- It's also a flexible alternative for getting Confidence Intervals (CI) for complex functions of parameters being estimated

- Suppose we know the true distributions of the variables involved in estimation of (the function of) the parameters. Then we could simulate samples many times, estimate the parameters each time and create an empirical distribution on the desired function of the parameters. Then use percentiles to determine CI.

- For real data we cannot generate new samples from the original population or distribution (by simulation). However, the bootstrap approach allows us to emulate the process of obtaining new sample sets

- We illustrate this approach on a simple data set Z that contains only $n = 3$ observations

- We randomly select n observations from the data set in order to produce a bootstrap data set, Z^{*1}. The sampling is performed with replacement



- In the example, Z^{*1} contains the third observation twice, the first observation once, and no instances of the second observation.

- This procedure is repeated B times for some large value of B, in order to produce B different bootstrap data sets,

$$Z^{*1}, Z^{*2}, \ldots, Z^{*B}$$

and B corresponding α estimates,

$$\hat{\alpha}^{*1}, \hat{\alpha}^{*2}, \ldots, \hat{\alpha}^{*B}$$

- We can compute the standard error of these bootstrap estimates

$$SE_B(\hat{\alpha}) = \sqrt{\frac{1}{B-1}\sum_{r=1}^{B}\left(\hat{\alpha}^{*r} - \frac{1}{B}\sum_{i=1}^{B}\hat{\alpha}^{*i}\right)^2}$$

- This serves as an estimate of the standard error of $\hat{\alpha}$ estimated from the original data set

Comparison of the estimates in the example in ISLR, Ch 5.2:

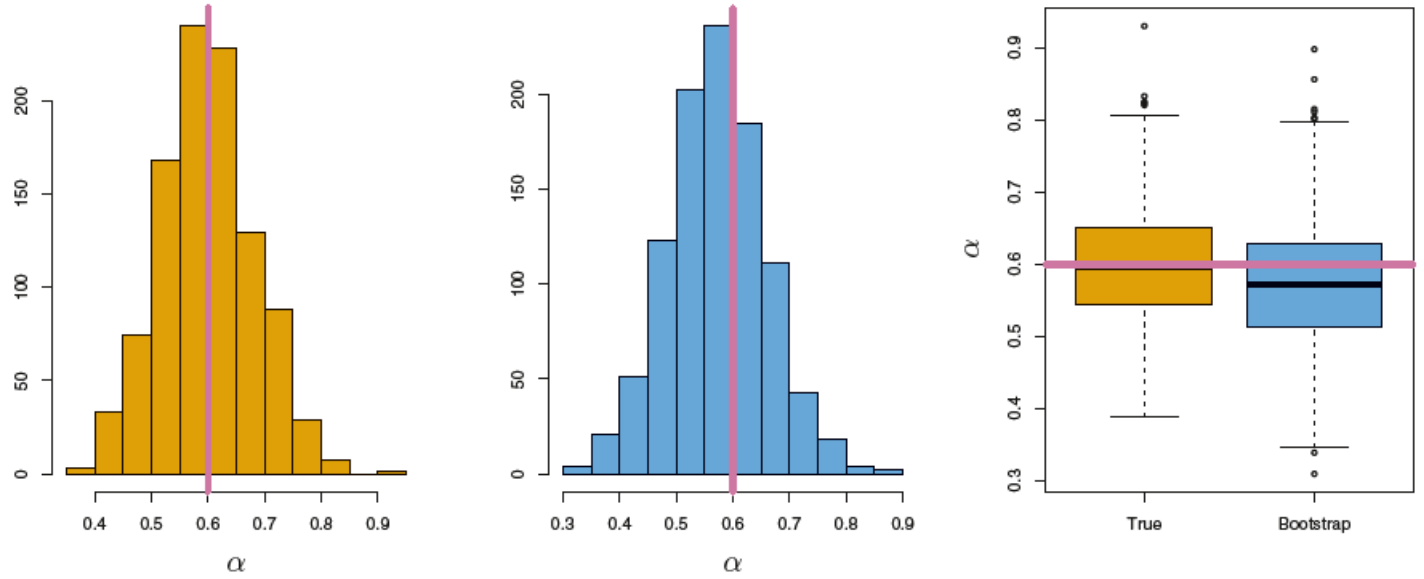


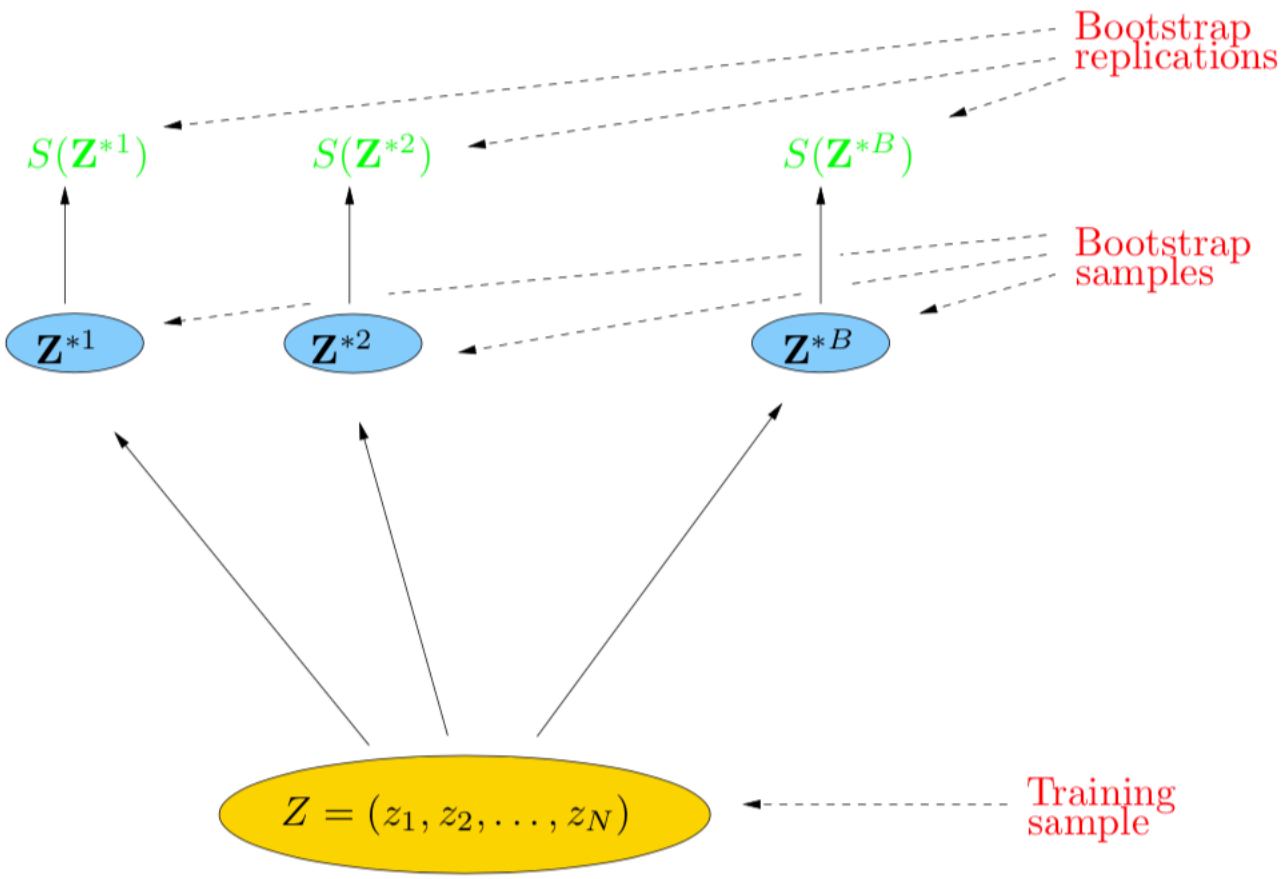
FIGURE 5.10. Left: *A histogram of the estimates of α obtained by generating 1,000 simulated data sets from the true population.* Center: *A histogram of the estimates of α obtained from 1,000 bootstrap samples from a single data set.* Right: *The estimates of α displayed in the left and center panels are shown as boxplots. In each panel, the pink line indicates the true value of α.*

- The parameter estimates by simulation and by bootstrap are both close to the true parameter value

- Code for `Auto` data

- Reading:

  ISLR: Chapter 5
  ESLII: Chapter 7.2, 7.10, 7.11


Bootstrap replications
$S(\mathbf{Z}^{*1})$
$S(\mathbf{Z}^{*2})$
$S(\mathbf{Z}^{*B})$
Bootstrap samples
$\mathbf{Z}^{*1}$
$\mathbf{Z}^{*2}$
$\mathbf{Z}^{*B}$
$Z = (z_1, z_2, \ldots, z_N)$
Training sample