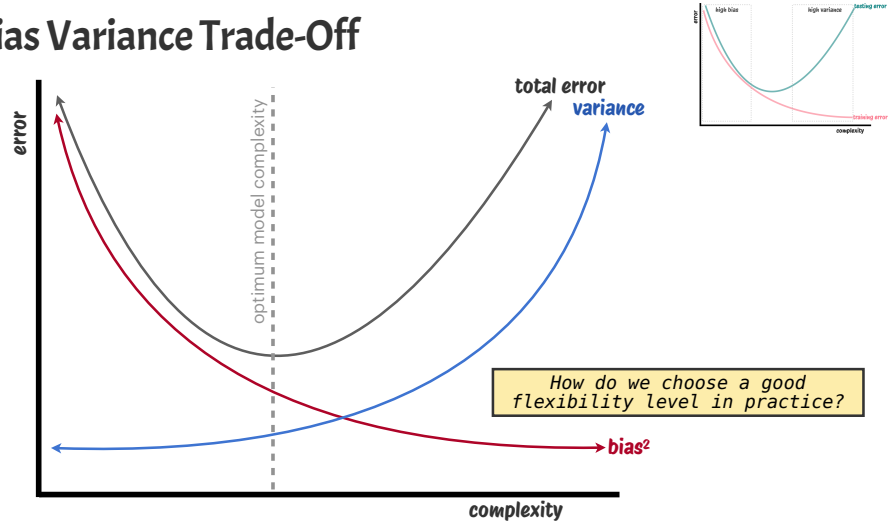
# Model Validation

Lecture 6

Termeh Shafie



## Bias Variance Trade-Off






## Validation Set Approach/Test-Train-Split (TTS)

Our validation approach so far...

1. Split data
2. Train model on training set
3. Evaluate train and test set


**The test error** is the average error resulting from using a method to predict a response on a new observation that was not used when training the method

**The training error**

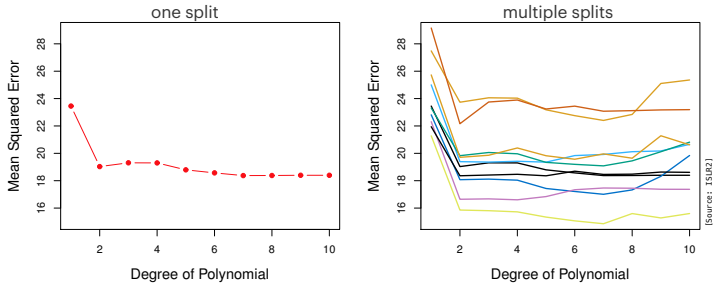
- is computed by applying SL method to observations when training the method
- often very different from test error rate and underestimates the test error rate



## Validation Set Approach/Test-Train-Split (TSS)

**Example:** Automobile Data (ISLR2)

- compare linear vs higher-order polynomial terms in a linear regression
- y = gas mileage in miles per gallon, x = horsepower
- randomly split the 392 observations in to two sets: training and validation set of 196 observations each



> "If model selection and true error estimates are to be computed simultaneously, the data should be divided into three disjoint sets"
>
> [Brian D. Ripley, 1996]

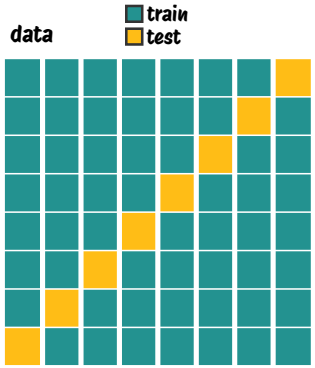
1. Training set (a.k.a. pseudo-training)
2. Validation set (a.k.a. pseudo-test)
3. Test set (*unseen!*)



# K-Fold Cross Validation

cross validation **simulates multiple train-test-splits** on the training data

1. Randomly divide the data into *K* equal-sized parts
2. Leave out one part *k* out
3. Fit model to *K-1* parts combined ("train")
4. Obtain predictions for the left out part *k* ("test")
5. Repeat for each part *k= 1,2,...,K*
6. Combine results to get error estimates






# Leave-One-Out Cross Validation (LOOCV)

cross validation **simulates multiple train-test-splits** on the training data

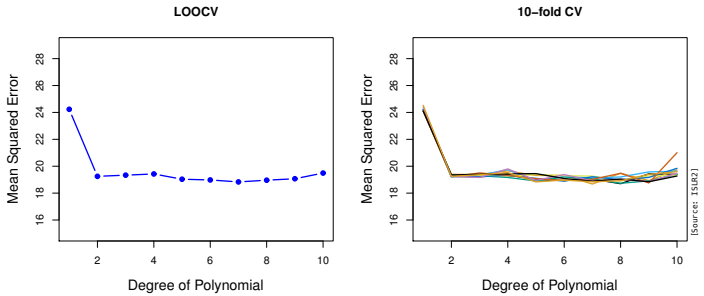
*K*-fold where every single data point is its own fold
*K = n*




# K-fold Cross Validation vs. LOOCV

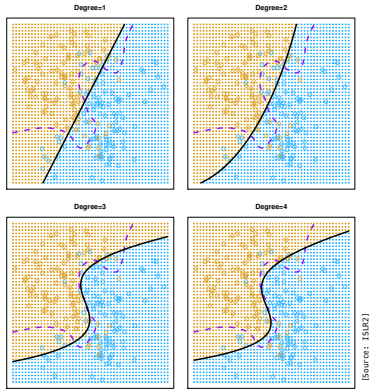
**Example:** Automobile Data (ISLR2)

- *K*-fold CV depends on the chosen split
- *K*-fold CV: we train the model on less data than what is available $\Longrightarrow$ bias in test error estimate
- LOOCV: training samples highly resemble each other $\Longrightarrow$ higher variance of test error estimate



# Cross Validation on Classification Problems

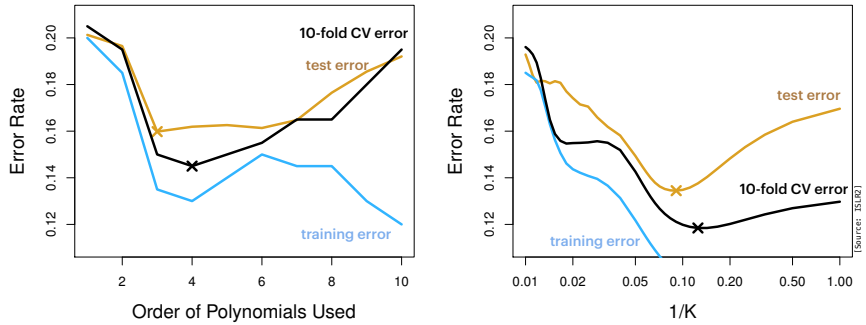
similar idea, but use misclassification error instead of MSE




9

# Cross Validation on Classification Problems

similar idea, but use misclassification error instead of MSE




10

# Cross Validation: Right or Wrong?

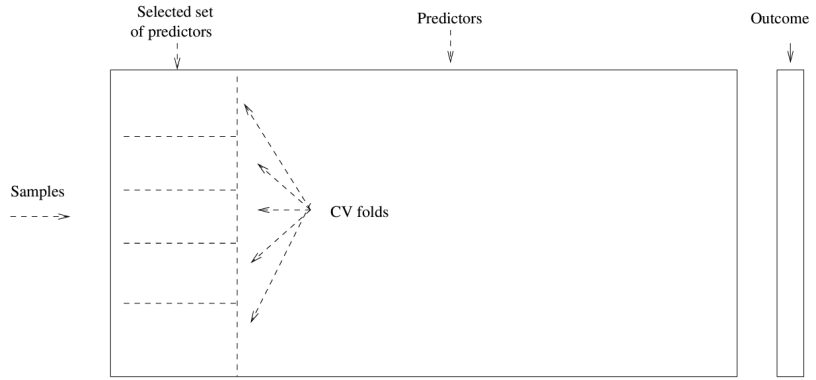
- We want to apply a classifier to a data set with two classes
- Proposed strategy:
  - Start with the 5000 available predictors and 50 samples
  - Select 100 predictors based on highest correlations with the class labels
  - Apply a classifier, e.g. logistic regression, using only these 100 predictors

*How do we estimate the test performance?*

- The wrong way: apply cross validation for the classifier only
- The right way: apply cross validation in both steps

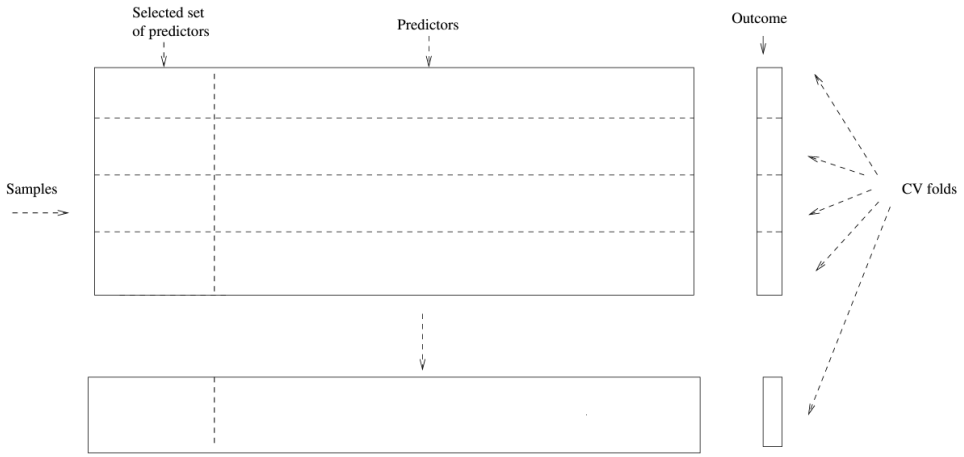
11

# Cross Validation: Right or Wrong?




12

# Cross Validation: Right or Wrong?






# Cross Validation: Right or Wrong?

**Example:**

- Divide the data into 10 folds
- For $i = 1,\ldots,10$
  - Using every fold except $i$, perform variable selection and fit the model with the selected variables
  - Compute the error on fold $i$
- Average the 10 test errors obtained

**Moral of the story:**
You can and often should validate your entire data processing & learning pipeline, even variable selection!



# Cross Validation: Right or Wrong?

❌ WRONG

```
Entire dataset (X, y)
        ↓
❌ Select variables using ALL data
        ↓
Split into CV folds (leakage already happened!)
        ↓
Train & evaluate model in CV folds
```

✅ RIGHT

```
Split the data into K folds
For each fold i = 1 … K:
    Training folds (all except fold i)
        ↓
    Perform feature selection   ← done ONLY on training data
    Train the model
        ↓
    Test on fold i (untouched data)
Average the K test errors
```

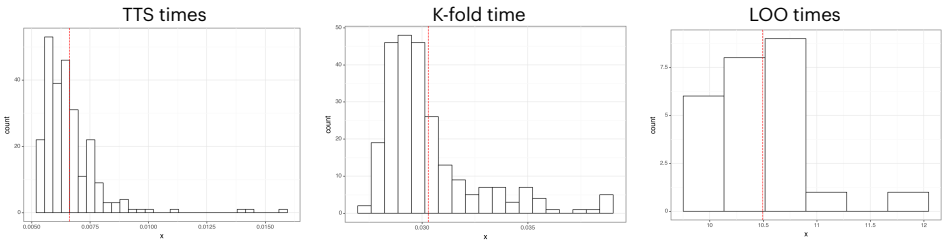


# How many folds?

- With large number of folds:
  - \+ The bias of the true error rate estimator will be small (estimator very accurate)
  - − The variance of the true error rate estimator will be large
  - − The computational time will be very large as well (many experiments)
- With small number of folds:
  - \+ The number of experiments and, therefore, computation time are reduced
  - \+ The variance of the estimator will be small
  - − The bias of the estimator will be large (smaller than the true error rate)
- In practice, the choice of the number of folds depends on the size of the dataset
  - For large data: even 3-fold cross validation will be quite accurate
  - For sparse data: leave-one-out in order to train on as many examples as possible

# Cross Validation and Computational Time

two major things to think about when choosing a method of model validation are **computational expense of the model** and **the size of your dataset**




excepts for ***very*** large data sets or incredibly complex models (which we won't really touch until the very end)
computational expense between TTS and KF is often negligible and a weak argument for justifying TTS over KF/LOO



# Hyperparameter Tuning

**Hyperparameters**

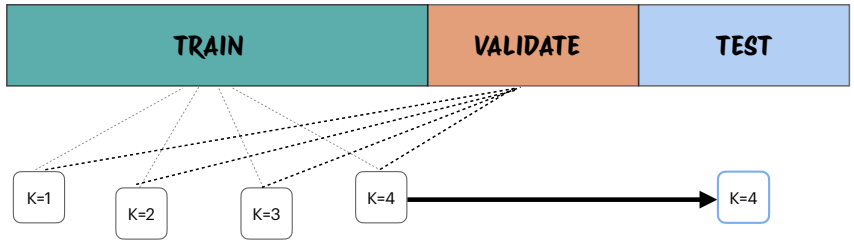
- parameters of SL model that are not learned during training
- values are set before training begins
- Examples:
  - *K* in KNN
  - the learning rate of a neural network
  - the depth and width of a decision tree
  - the regularization strength of a linear regression model
- uses performance metric (e.g. confusion matrix) obtained from cross-validation



# Hyperparameter Tuning

**KNN Example**

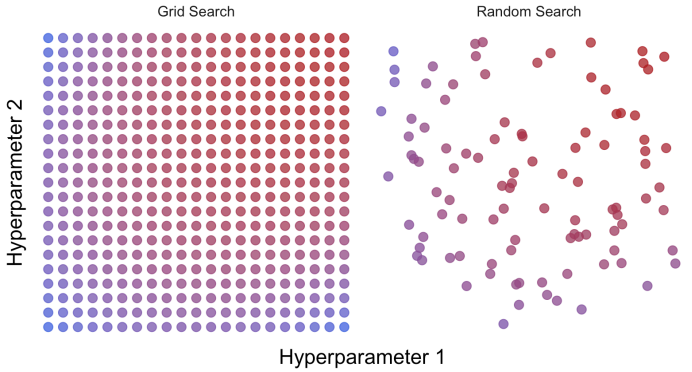
Which K to choose?






# Hyperparameter Tuning

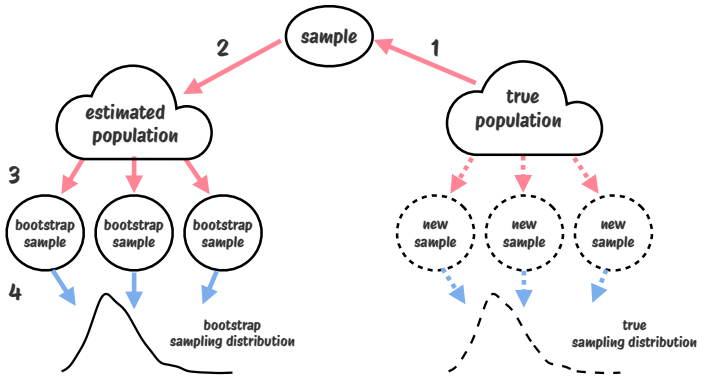
1. Select grid or random hyperparameter combinations
2. For each combination, use cross-validation to estimate model performance
3. The combination with best performance is chosen



# Bootstrapping: resampling with replacement

*"The Baron had fallen to the bottom of a deep lake. Just when it looked like all was lost, he thought to pick himself up by his own bootstraps."*

"The Surprising Adventures of Baron Munchausen" (1785) by Rudolph Erich Raspe






# Bootstrapping

From a dataset with $N$ observations

1. Randomly select (with replacement) $N$ examples and use this set for training
   - remaining examples not selected for training are used for testing
2. Repeat this process for a specified number of folds $K$
3. The true error is estimated as the average error rate on test examples




# Comparison CV and Bootstrapping

- The bootstrap increases the variance that can occur in each fold (desirable)
- Example: a classification problem
  - Assume $C$ classes, $N$ observations, $N_C$ observations for each class
  - The a *priori* probability of choosing an example from class $C$ is $N_C/C$
  - Without replacement, once class is chosen, probability has changed to $(N_C\text{-}1)/C$
- Sampling with replacement preserves a priori probabilities of random selection



# Cross Validation and Panel Data

- Autocorrelation issues: assumption of i.i.d. observations won't hold over time
- The CV methods presented will ignore the sequential nature of time

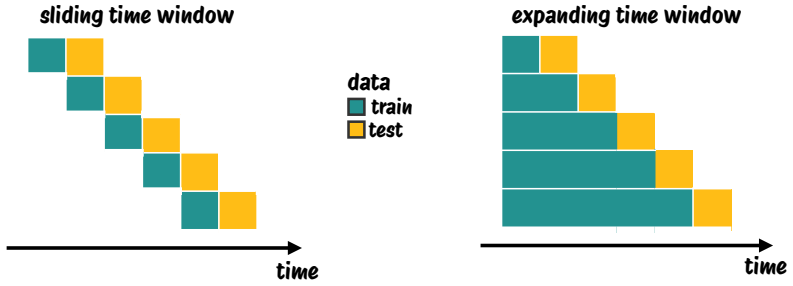
$\Longrightarrow$ **Nested Cross Validation**

- a nested loop of cross-validation:
  - inner loop used for parameter tuning:
    $K$-fold CV is used to pick parameter values
  - outer loop is used for performance evaluation
    another $K$-fold CV used to evaluate model with selected parameters
- The CV methods presented will ignore the sequential nature of time

# Cross Validation and Panel Data

- Autocorrelation issues: assumption of i.i.d. observations won't hold over time
- The CV methods presented will ignore the sequential nature of time

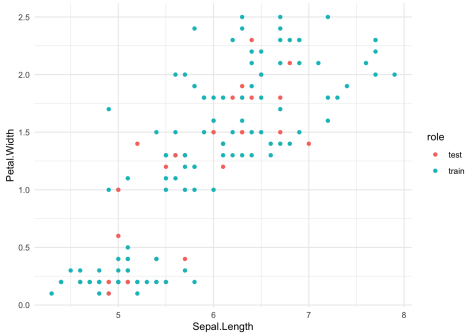
$\Longrightarrow$ **Nested Cross Validation**






# Today's R Practical Part I: Iris Data

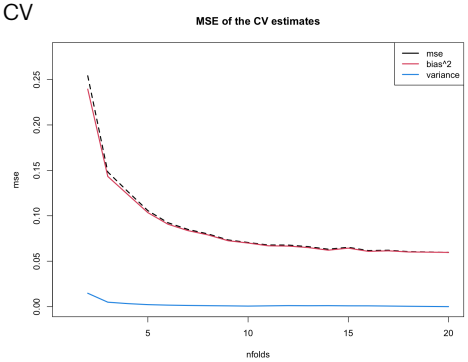
- Data: 50 flowers in 3 different species and 4 different attributes
- Use some useful packages such as `caret` (Classification And Regression Training)
- Apply linear model and use following performance metrics: *$R^2$, RMSE, MAE*
- Validation approaches:
  - Validation set approach
  - LOOCV
  - K-fold CV
  - Repeated K-fold CV






# Today's R Practical Part II: KNN Simulation

- Use cross validation to estimate test performance of KNN regression
  - Choosing optimal number of numbers *K* (using MSE and RMSE)
  - Choosing number of folds in *K*-fold CV






# Today's R Practical Part III: Amazon Books

Use cross validation to estimate test performance of the previous linear regression