MATH 189
Resampling Methods

Wenxin Zhou
UC San Diego

Time: 2:00—3:20 & 3:30—4:50pm TueThur
Zoom ID: 985 7463 1067

Outline

- In the previous lecture, we introduced least squares method for linear regression.
  - Regression to the mean
  - Simple linear regression
  - Multivariate linear regression

- Today we will introduce some resampling methods and their applications in statistical learning.
  - Some questions in statistical learning
  - Cross-validation methods

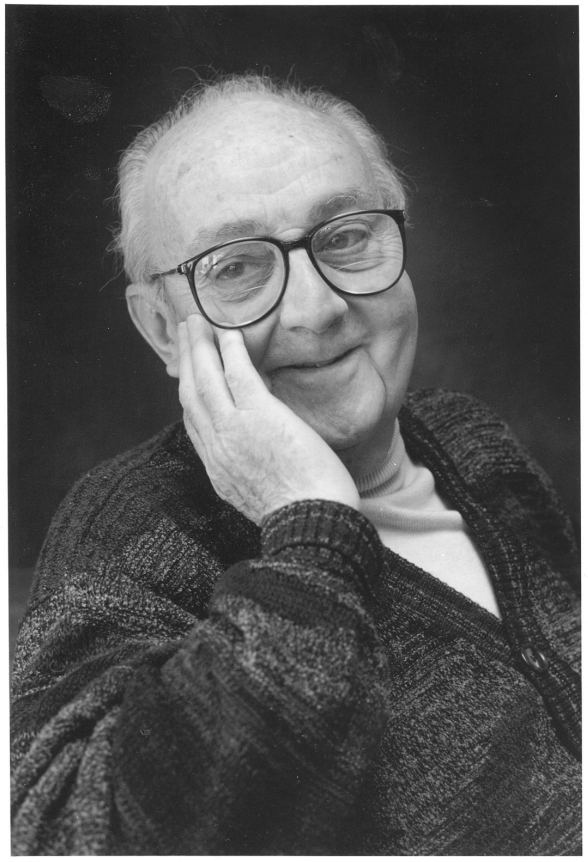
Resampling Method

- Resampling method is an indispensable tool in modern statistics.
- Resampling method involves repeatedly drawing samples from a training set and refitting a model of interest on each sample in order to obtain additional information about the fitted model.
- Resampling method is widely used in many statistical studies:
  1. Performance evaluation.
  2. Hyper-parameter selection.
  3. Inference of estimated parameters.
  4. Hypothesis testing.
  5. Model validation.

Question 1: How to Evaluate a Statistical Model?

"All models are wrong, but some are useful."

By George Box



- All statistical models depend on certain assumptions and hence are simplifications of reality.
- Some assumptions are imposed as we do not know the truth and have to "guess".
- Some simplifications are very useful:
  1. Ignore nuisance information.
  2. Explain complex phenomena with simple model.
  3. Make computation feasible.

How to Evaluate a Statistical Model?

- For a statistical problem, we can establish various statistical models based on different assumptions.
- A natural question: Which model should we use?
- A quick answer: Choose the model that fits our objective best.
- A follow-up question: For a given objective, how to assess various models?

- Suppose that we want to choose a model that has the best prediction performance. In other words, the smallest prediction error.

Training Error

- Let's consider a regression model with only one predictor:

$$y = f(x) + e,$$

  where y is the response variable and x is the covariate, e represents the proportion of y that can not be explained by x.

- Suppose we observe a sample $\{x_i, y_i\}$ for $i = 1, \dots, n$. We estimate the unknown function $f(\cdot)$ by some estimator $\hat{f}(\cdot)$.

- Then y_i will be estimated by $\hat{f}(x_i)$. The training error is defined as

$$Training\ Error = \frac{1}{n}\sum_{1=1}^{n}\left[y_i - \hat{f}(x_i)\right]^2.$$

- This training error is usually called mean squared error (MSE) or in-sample error.

Test Error

- Suppose we observe additional observations, $\{x_i^{test}\}$ for $i = 1, \dots, m$, for which we want to predict/classify the response.
- We can predict the response variable y_i^{test} corresponds to x_i^{test} as $\hat{f}(x_i^{test})$. The test error is

$$Test\ Error = \frac{1}{m}\sum_{i=1}^{m}\left[y_i^{test} - \hat{f}(x_i^{test})\right]^2.$$

- The test error is usually called mean squared prediction error (MSPE) or out-of-sample error

- Usually we don't observe y_i^{test}, and hence can not directly calculate the test error! Hence, we cannot directly select a model that minimizes the test error.

- The model that minimizes the training error may not be the one that minimizes the test error!

Question 2: How to Select Hyper-parameters

- The terminology hyper-parameter is first used in Bayes' statistics to denote the parameter of prior distributions. In Bayes statistical inference, the priors express one's beliefs about this quantity before some evidence is taken into account.
- Many statistical methods involved some parameters whose values should be set before the learning process. In a general sense, they can also be called hyper-parameters.
- Usually, we need to choose these hyper-parameters before we fit a statistical model.
- Different choices of hyper-parameters lead to different models.
- A natural question is how to choose these hyper-parameters?

Hyper-parameters Selection

- In practice, we want to choose a set of hyper-parameters that lead to a model with certain advantage:
  1. Model accuracy
  2. Computational efficiency
  3. Flexibility
  4. Robustness
- For each objective, we need to choose a set of hyper-parameters that minimizes some loss function:
  1. Prediction error.
  2. Computational cost.
  3. Loss when model is mis-specified.
  4. Loss when assumptions are violated.

The Validation Set Approach

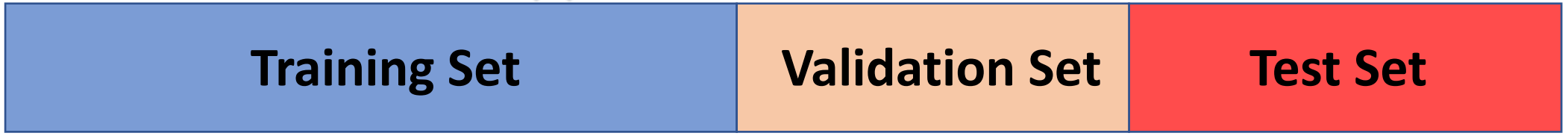
- Suppose we want to estimate the test error associated with fitting a particular statistical model on a sample.
- The true test/prediction set usually does not have the information of the response variable or true class labels. (Otherwise no need to predict.)
- We cannot directly assess the test error on the test set.
- The validation set approach involves randomly dividing the available set of observations into two parts, a *training set* and a *validation set*.
- We first fit the model on the training set. The fitted model is then used to predict the responses of the observations in the validation set.
- The resulting validation set error provides an estimate of the test error.

Normal Learning Process

- Training a statistical model using the observations in the training set.

Problem: Cannot assess the test error because we do not know the true responses of test set.

The Validation Set Approach



- Training a statistical model using the observations in the training set.
- Validate trained model by the "test error" in validation set.
- Use the trained model to predict the test set.

- Model accuracy is assessed by the test error in validation set.
- Select a model/parameter that minimizes the test error in validation set.

Example: Predict MPG with Horsepower

- We collected a dataset of 392 observations (automobile
For each automobile, we record the miles per gallon (MP
and horsepower.

- We are interested in predicting the MPG of an automobile with its horsepower:

$$MPG = f(horsepower) + error,$$

where $f(\cdot)$ is some regression function.

Question: Which model gives the best prediction performance?

Five Candidate Models

- Denote $y = MPG$ and $x = horsepower$:

$$y = f(x) + error.$$

- We consider to fit $f(\cdot)$ by a series of polynomial models with orders from 1 to 5:

  Model 1: $y = a + bx + \epsilon$

  Model 2: $y = a + bx + cx^2 + \epsilon$

  Model 3: $y = a + bx + cx^2 + dx^3 + \epsilon$

  Model 4: $y = a + bx + cx^2 + dx^3 + ex^4 + \epsilon$

  Model 5: $y = a + bx + cx^2 + dx^3 + ex^4 + fx^5 + \epsilon$

  Goal: Compare the model that minimizes the training error and the model that minimizes the test error.

Example: Validation Set Approach

- Denote $y = MPG$ and $x = horsepower$:

$$y = f(x) + error.$$

- We randomly divide the observed sample $\{x_i, y_i\}$ for $i = 1, \dots, 392$ into a training set $\{x_i^t, y_i^t\}$ for $i = 1, \dots, 196$ and a validation set $\{x_i^v, y_i^v\}$ for $i = 1, \dots, 196$.
- We use the training set to train an estimator $\hat{f}(\cdot)$. Then y_i is estimated by $\hat{f}(\mathrm{x_i})$.

- The training error is defined as

$$Training\ Error = \frac{1}{196}\sum_{i=1}^{196}\left[y_i^t - \hat{f}(x_i^t)\right]^2.$$

- The test error is estimated by the prediction error of the validation set

$$Test\ Error = \frac{1}{196}\sum_{i=i}^{196}\left[y_i^v - \hat{f}(x_i^v)\right]^2.$$

Example: Validation Set Approach

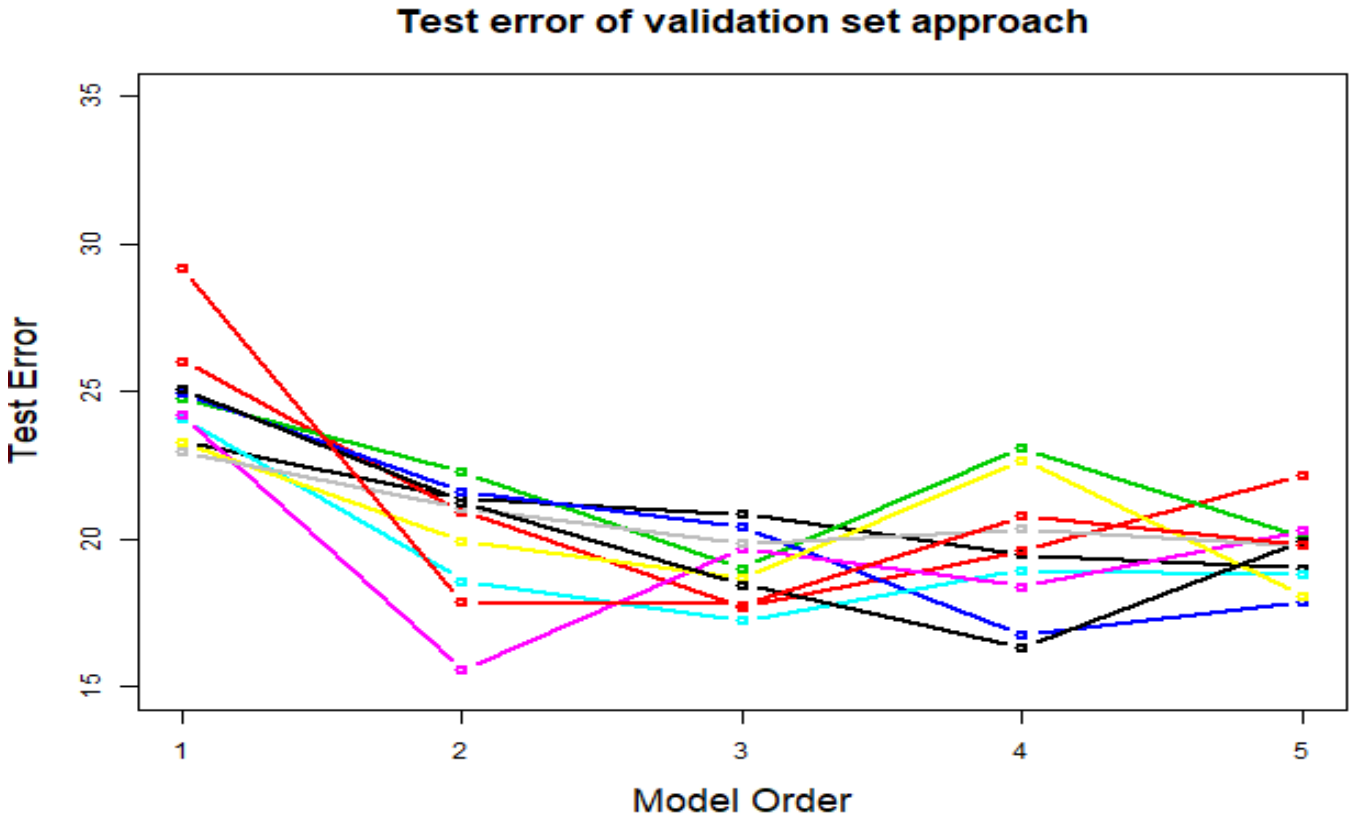
The training error and test error of each model are given in the following table.

Model	1	2	3	4	5
Training error	25.03	19.20	19.14	19.14	18.43
Test error	23.30	18.90	19.26	20.39	19.26

- Model 4 has smallest training error and Model 2 has smallest test error.
- Minimizing training error and minimizing test error leads to different models!
- The model that minimizes training error (5 parameters) is more complicated than then one that minimizes test error (3 parameters).
- This is a common phenomena in statistical learning.

Example: Validation Set Approach

Here we repeat the validation set approach for 10 times and plot the test errors in the following figure.



- The test error estimated by the validation set approach can be highly variable.
- It depends on the random partition of training and validation sets.

Some Issues with Validation Set Approach

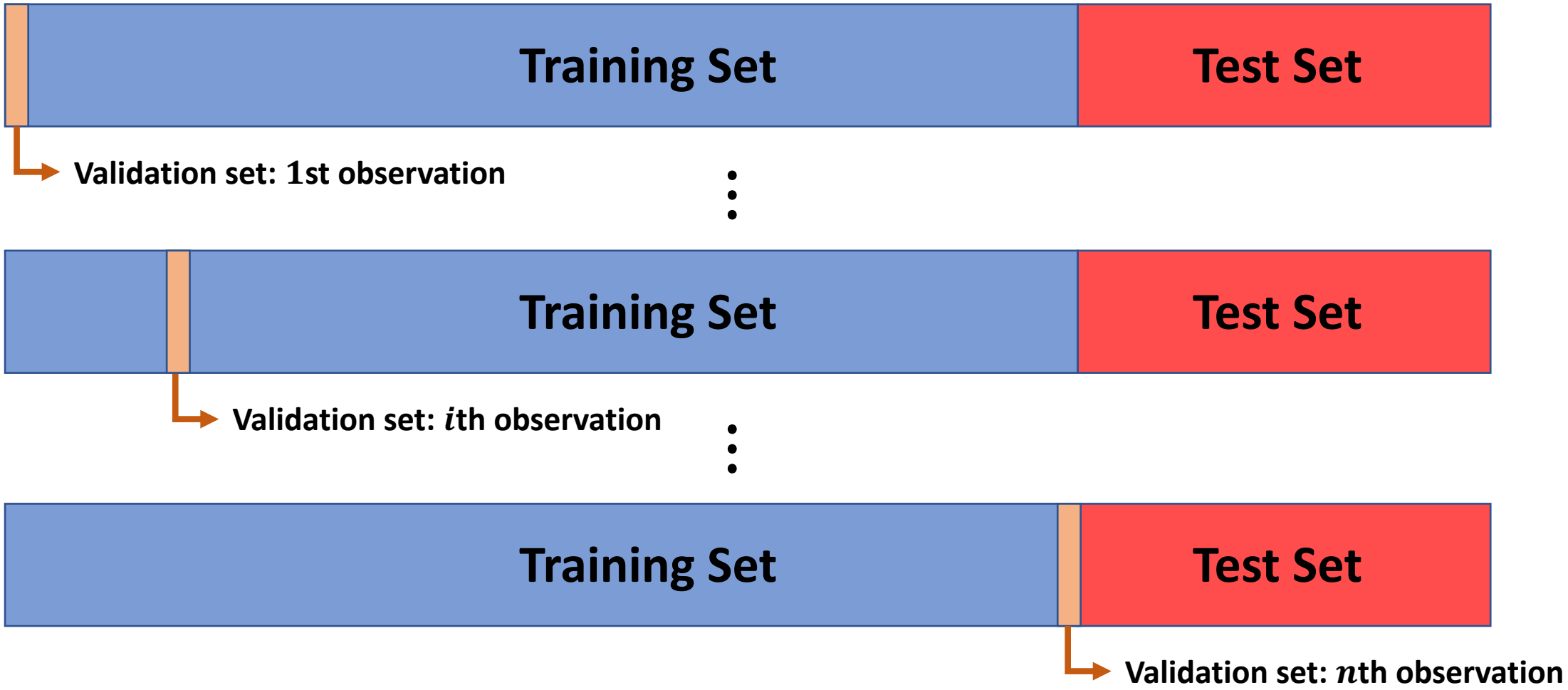
- The test error estimated by the validation set approach can be highly variable, depending on precisely which observations are included in the training set and which observations are included in the validation set.
- In the validation set approach, only a subset of the observations, which are included 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 fitted on the entire data set.

Leave-One-Out Cross Validation

- Leave-one-out cross-validation (LOOCV) is closely related to the validation set approach. It attempts to address the drawbacks of the latter.

- Instead of randomly dividing the sample into training set and validation set, LOOCV does it in an alternating way:
  1. Pick the first observation (x_1, y_1) as the validation set and use the rest observations $(x_2, y_2), \ldots, (x_n, y_n)$ as the training set. The test error is measured as $MSPE_1 = (y_1 - \hat{y}_1)^2$.
  2. Pick (x_2, y_2) as the validation set and obtain $MSPE_2 = (y_2 - \hat{y}_2)^2$.
  3. Repeat for each observation in the sample and obtain $MSPE_3, \ldots, MSPE_n$.
  4. $Test\ error = \frac{1}{n}\sum_{i=1}^{n} MSPE_i$.

Leave-One-Out Cross Validation Approach



Leave-one-out CV versus Validation Set

Leave-one-out CV

- Repeatedly fit the statistical learning method using training sets that contain n − 1 observations.

- Less bias as the training set is close to the whole sample.
- Avoid overestimating the test error.
- Yield same results when applied multiple times.
- Expensive computational cost as the model has to be fit n times.

Validation set

- Fit the statistical learning method using a random sub-sample.

- Large bias as we used smaller sample size to train the model.
- May overestimate the test error.
- Yield different results when applied repeatedly due to random training/validation splitting.
- Fast computation as only trained once.

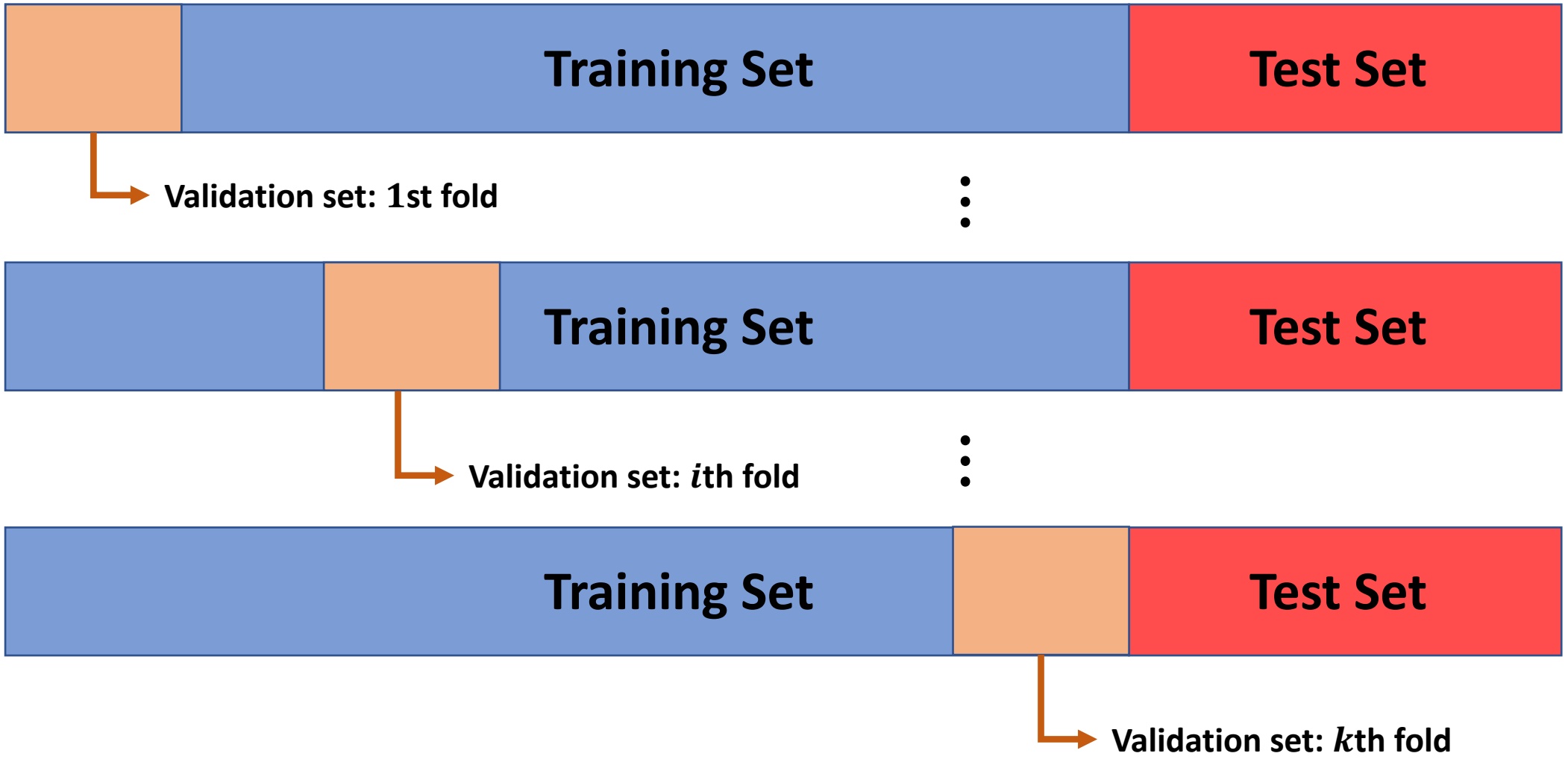
K-fold Cross Validation

- An alternative to LOOCV is k-fold CV. This approach involves randomly dividing the set of observations into k groups, or folds, of approximately equal size.

- k-fold CV also works in a repeatedly way:
  1. Pick the first fold as the validation set and use the rest $k-1$ folds as the training set. The test error, $MSPE_1$, is computed on the observations in first fold.
  2. Repeat step 1 by using each fold as the validation set and use the rest -1 folds as the training set. The test errors, $MSPE_2$, … , $MSPE_k$, are computed on the observations in the held-out fold.
  3. $Test\ error = \frac{1}{K}\sum_{j=1}^{K} MSPE_j$.

- LOOCV can be considered as a special k-fold CV with $k = n$.
- In practice, we usually perform k-fold CV using $k = 5$ or $k = 10$.

K-fold Cross Validation Approach



Advantages of K-fold Cross Validation

- Both LOOCV and k-fold CV are very general resampling methods and can be used to any kind of predictive learning process.
- In general, k-fold CV (with $k = 5$ or 10) is more popular than LOOCV.
- What are the advantages of k-fold CV?

1. The most obvious advantage is computational feasibility
   - LOOCV requires fitting the statistical learning method *n* times. This has the potential to be computationally expensive when n is large or fitting is computationally intensive.
   - k-fold CV only requires fitting the statistical learning method *k* times.
2. A less obvious but potentially more important advantage of k-fold CV is that it often gives more accurate estimates of the test error rate than LOOCV. This has to do with a bias-variance trade-off.

Bias-Variance Tradeoff in Cross Validation

Bias:

- The validation set approach tends to have large bias.

  The model is trained with a smaller sample size. The validation set approach tends to overestimate the test error.

- Using this logic, the LOOCV will give approximately unbiased estimates of the test error.

  Each training set contains $n - 1$ observations, which is almost as many as the number of observations in the full dataset.

- And performing k-fold CV will lead to an intermediate level of bias.

  Each training set contains $(k - 1)n/k$ observations—fewer than in the LOOCV approach, but substantially more than in the validation set approach.

Bias-Variance Tradeoff in Cross Validation

Variance:

- On the other hand, LOOCV used to has high level of variance.
    1. 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.
    2. Mean of many highly correlated quantities has higher variance than the mean of many variables that are less correlated.
- In contrast, k-fold CV with $k < n$ used to have smaller variance.

  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.

- To summarize, there is a bias-variance trade-off associated with the choice of k in k-fold CV.
- Typically, given these considerations, popular choices $k = 5$ or $k = 10$ have been shown empirically to yield test error estimates that suffer neither from excessively high bias nor from very high variance.

Example: Predict MPG with Horsepower

- Denote $y = MPG$ and $x = horsepower$:

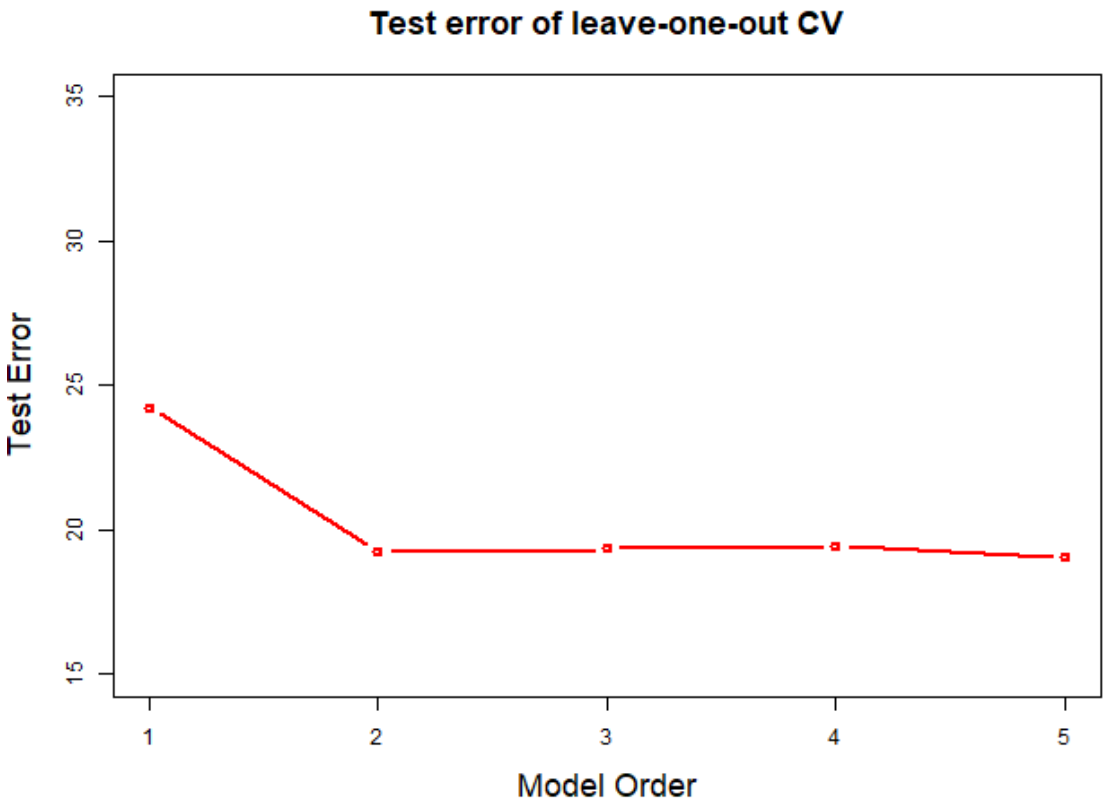
$$y = f(x) + error.$$

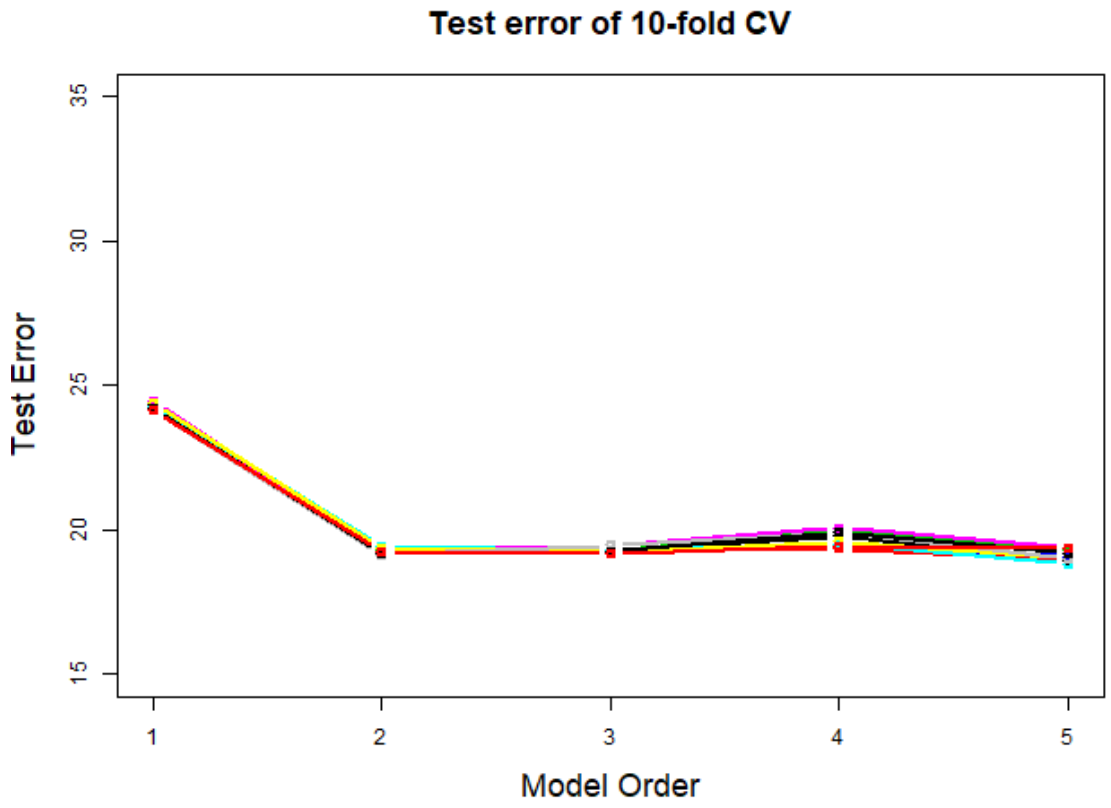
- We consider to fit $f(\cdot)$ by a series of polynomial function of x with orders equal 1 to 5:

  Model 1: $y = a + bx + \epsilon$
  Model 2: $y = a + bx + cx^2 + \epsilon$
  Model 3: $y = a + bx + cx^2 + dx^3 + \epsilon$
  Model 4: $y = a + bx + cx^2 + dx^3 + ex^4 + \epsilon$
  Model 5: $y = a + bx + cx^2 + dx^3 + ex^4 + fx^5 + \epsilon$

Now we compare the test errors estimated by LOOCV and 10-fold CV (repeat 10 times).

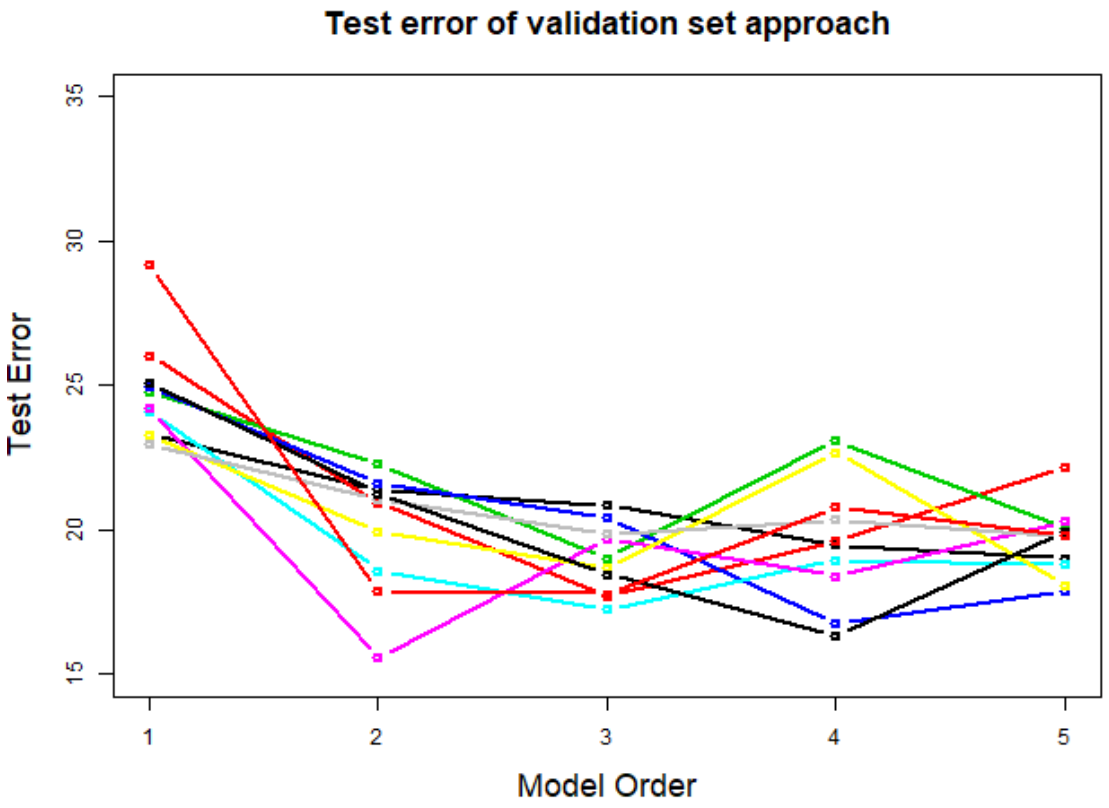
Example: LOOCV versus 10-fold CV

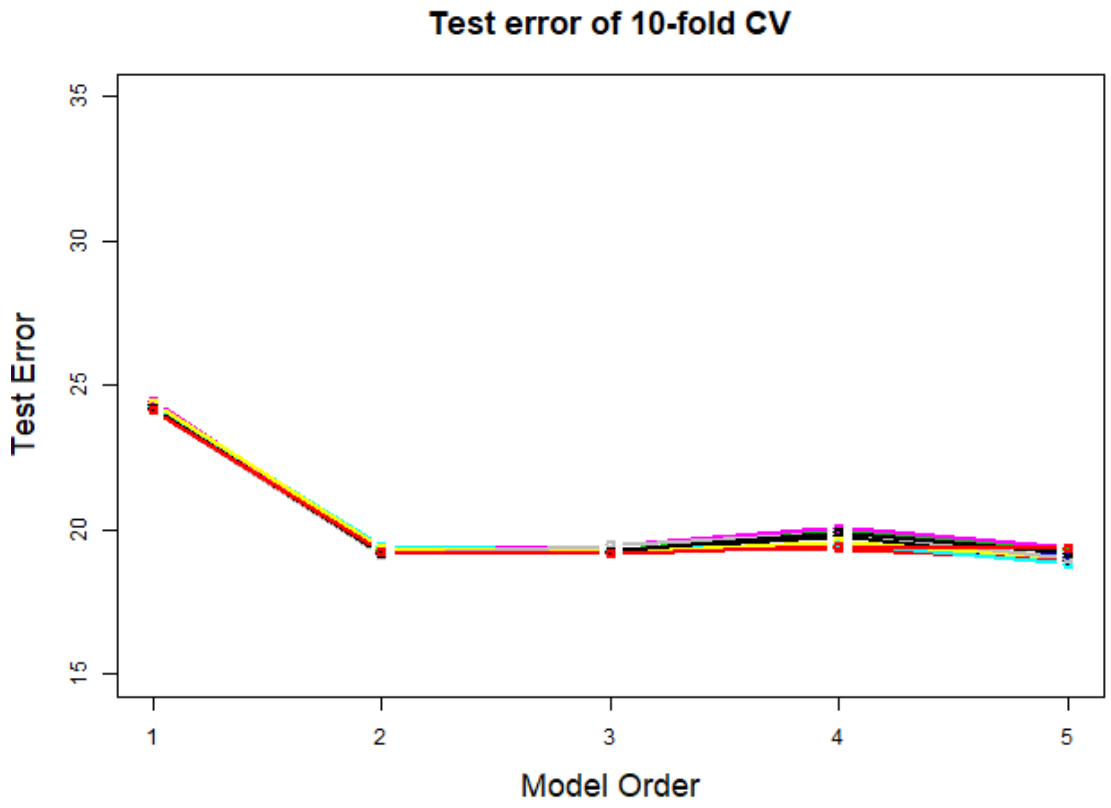




The empirical performance of these two methods are very close !

Example: Validation Set Approach vs 10-fold CV





10-fold CV is more stable than the validation set approach!

Cross Validation on Classification Problems

- So far, we have illustrated the use of cross-validation in the regression setting where the outcome Y is quantitative. The test error is quantified by the $MSPE$.
- Cross-validation can also be a very useful approach in the classification setting when Y is qualitative.
- In classification problems, we can quantify test error by the number of misclassified observations.

- For instance, in the classification setting, the LOOCV error takes the form

$$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$$, where $I(y_i \neq \hat{y}_i) = 1$ when $y_i \neq \hat{y}_i$ and 0 otherwise.

- The K-fold CV error takes the form

$$CV_{(n)} = \frac{1}{K}\sum_{j=1}^{K}\sum_{i=1}^{n_j} I(y_i \neq \hat{y}_i)$$, where n_j is the sample size in jth fold.