DSC5103 Statistics

Session 5. Validation

Review of last session

- Logistic Regression
  - Y ~ Bernoulli(p)
  - Logistic function as a nonlinear mapping from η to p
- Classification in general
  - From p_hat to classes
- Other Generalized Linear Models: E[Y] ~ X
  - Poisson Regression
  - Survival Analysis

Plan for today

- Model selection in (generalized) linear models
  - The model selection workflow
  - The traditional vs. modern performance measures

- Validation methods: a tool for numerically estimating out-of-sample error
  - Validation set
  - Leave-One-Out Cross-Validation
  - K-fold Cross-Validation

Linear Model Selection

- To choose the optimal subset of predictors to be included in the model

- Workflow
  - Best subset: the best out of all possible combinations (2^p)

                                                            Null
  - Forward selection: start from none, iteratively add the variable that improve the performance measure the most

  - Backward selection: start from all, iteratively remove the least significant variable

- Performance measures
  - In-sample measures, such as RSS and R^2, are not enough (over-fitting!)
  - traditional Adj. R^2, AIC, BIC, and Mallow's Cp take model complexity (p) into account
  - modern Validation-based methods for approximating out-of-sample errors

Group # of variables	X_1	X_2	X_3	- - - -	X_p	FWD Selection	BWD Selection *	Best Subset	Best Model in Group
0						p		1	M_0
1		✓				$p-1$		$\binom{p}{1}=p$	M_1
2		✓	✓			$p-2$		$\binom{p}{2}=\frac{p\cdot(p-1)}{2}$	M_2
⋮			⋮			⋮		$\binom{p}{3}$	⋮
		✓	✓		✓	⋮	1	⋮	⋮
$p-1$	✓	✓ ✓	✓	✓ ✓	✓ ✓	1	1	$\binom{p}{p-1}=p$	M_{p-1}
p	✓ ✓	✓ ✓	✓ ✓	✓ ✓	✓ ✓		1	$\binom{p}{p}=1$	M_p

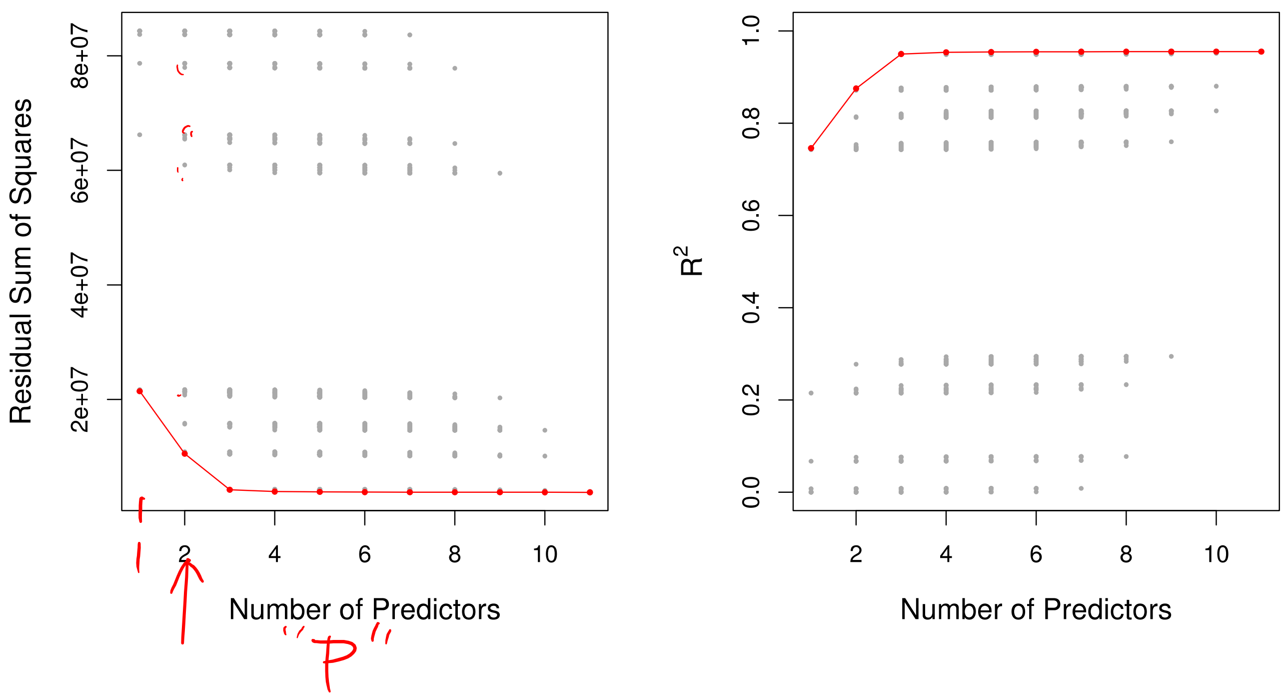
Within Group ⇒ same "p" ⇒ R^2 / RSS / deviance (In-Sample)

Across Group ⇒ Adj. R^2 / AIC / C_p / BIC

→ Validation Error

Credit Data: R^2 vs. Subset Size

- The RSS/R^2 will always decline/increase as the number of variables increase so they are not very useful
- The red line tracks the best model for a given number of predictors, according to RSS and R^2



Other Measures for Model Comparison

- *Indirectly* estimate test error by making an adjustment to the training error to account for the bias due to over-fitting
  - Adjusted R^2

$$\text{adjusted } R^2 = 1 - \frac{\text{RSS}/(n - p - 1)}{\text{TSS}/(n-1)}$$

  - AIC (Akaike information criterion)

(GLM) $$\text{AIC} = -2\text{Log-Likelihood} + 2p \qquad \text{AIC} = (\text{RSS} + 2p\hat{\sigma}^2)/(n\hat{\sigma}^2)$$ (Special: Linear / Gaussian)

  - Mallow's C_p (equivalent to AIC for linear regression)

$$C_p = (\text{RSS} + 2p\hat{\sigma}^2)/n$$

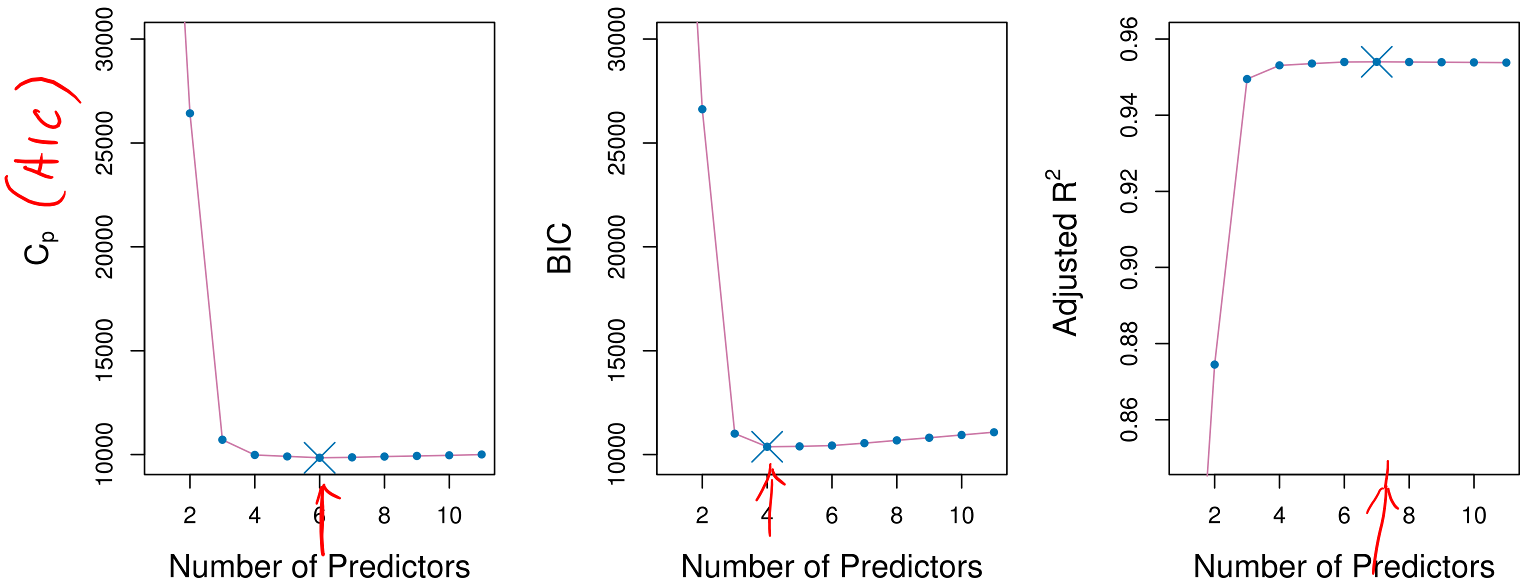
  - BIC (Bayesian information criterion)

$$\text{BIC} = (\text{RSS} + \log(n)p\hat{\sigma}^2)/n$$

- These methods add penalty to RSS for the number of variables (i.e. complexity) in the model, but none are perfect (e.g., how to estimate σ^2?)

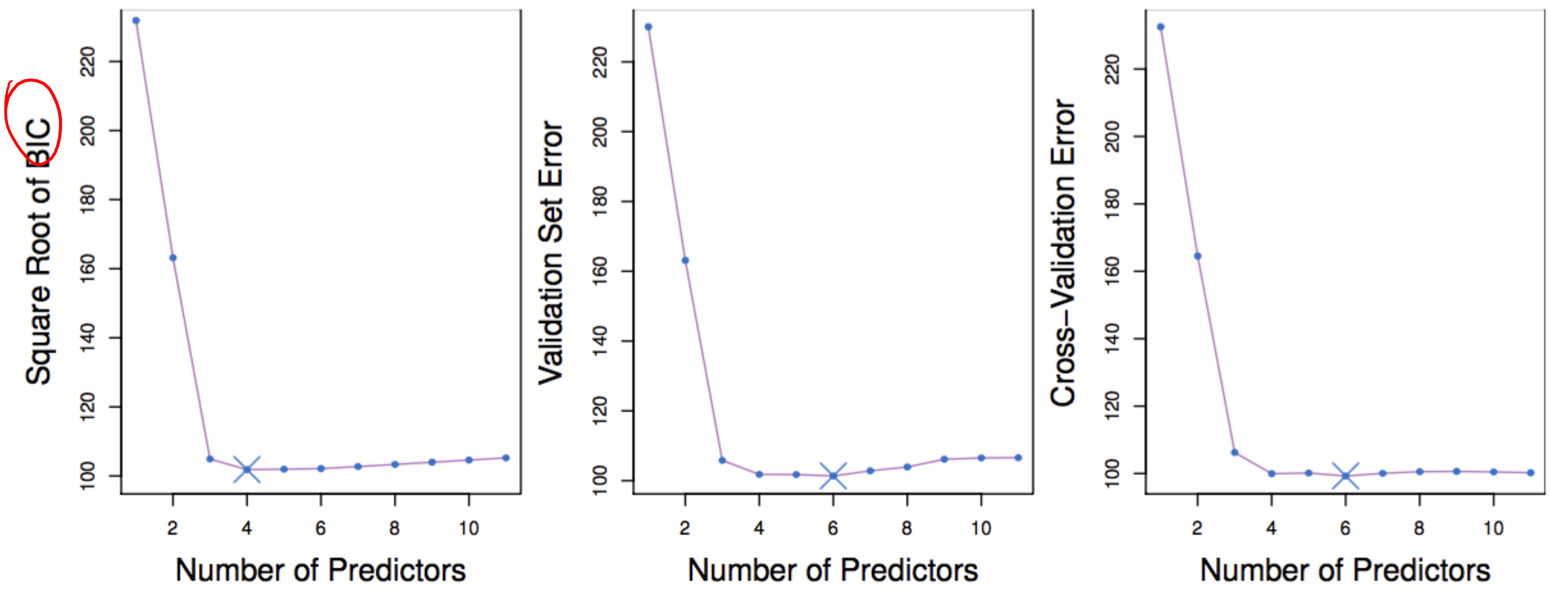
Credit Data: C_p, BIC, and Adjusted R^2

- A small value of C_p and BIC indicates a low error, and thus a better model
- A large value for the Adjusted R^2 indicates a better model



Model Comparison by Cross-Validation

- *Directly* estimate the out-of-sample error using validation/cross-validation



Model Selection

- Model selection with out-of-sample error in mind
  - CV is computationally intensive, it is not practical to do it for all possible models
  - A hybrid approach:
    - For each fixed model size k = 0, 1, …, p, select the best k predictors by RSS or R^2. We obtain the best model if we choose to have k predictors. Let's call it M_k.
    - Use Cp/AIC/BIC or cross-validation to compare M_k for $k = \underbrace{0, 1, \ldots, p}_{p+1}$, and choose the best k.

Model Selection Algorithm

- Best Subset Selection

Best Subset Selection

1. Let $\mathcal{M}_0$ denote the *null model*, which contains no predictors. This model simply predicts the sample mean for each observation.
2. For $k = 1, 2, \ldots p$:
   (a) Fit all $\binom{p}{k}$ models that contain exactly k predictors.
   (b) Pick the best among these $\binom{p}{k}$ models, and call it $\mathcal{M}_k$. Here *best* is defined as having the smallest RSS, or equivalently largest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

Model Selection Algorithm

- Forward Stepwise Selection

Forward Stepwise Selection

1. Let $\mathcal{M}_0$ denote the *null* model, which contains no predictors.
2. For $k = 0, \ldots, p-1$:
   2.1 Consider all $p-k$ models that augment the predictors in $\mathcal{M}_k$ with one additional predictor.
   2.2 Choose the *best* among these $p-k$ models, and call it $\mathcal{M}_{k+1}$. Here *best* is defined as having smallest RSS or highest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

Model Selection Algorithm

- Backward Stepwise Selection

Backward Stepwise Selection

1. Let $\mathcal{M}_p$ denote the *full* model, which contains all p predictors.
2. For $k = p, p-1, \ldots, 1$:
   - 2.1 Consider all k models that contain all but one of the predictors in $\mathcal{M}_k$, for a total of $k-1$ predictors.
   - 2.2 Choose the *best* among these k models, and call it $\mathcal{M}_{k-1}$. Here *best* is defined as having smallest RSS or highest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

Evaluating Out-of-Sample Error

- Motivation
  - In-sample vs. Out-of-sample Error (Bias-Variance Trade-off)
  - How to estimate out-of-sample error (and then do model selection) **without** a test dataset?!
    - Theoretical adjustment (Adjusted R^2, AIC, BIC, Cp): penalize model complexity (p)
    - “Estimate” the out-of-sample error using validation data

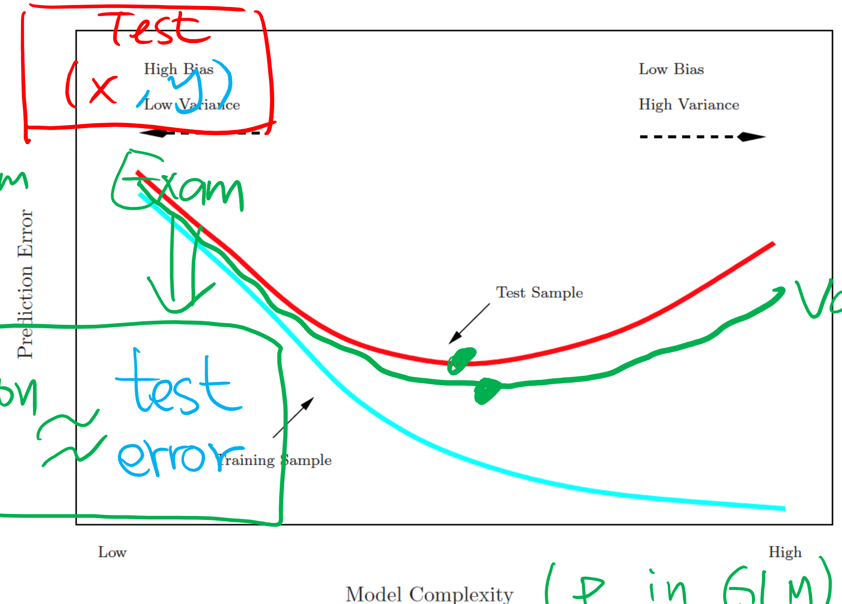
Training (x,y)	Validation (x,y)

Test (x, y)

Exercises ⇓ training error

Mock Exam ⇓ validation error ≈ test error

Exam ⇓



Validation error

(p in GLM)

(k in KNN)

Evaluating Out-of-Sample Error

- Three common approaches
  - The Validation Set Approach
  - Leave-One-Out Cross Validation
  - K-fold Cross Validation

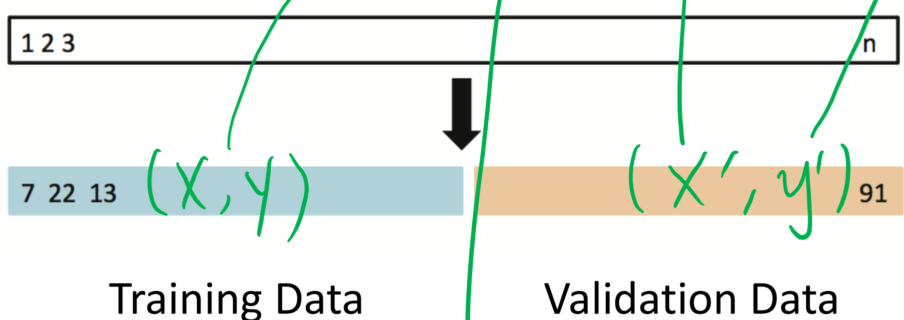
The Validation Set Approach

Stratified sampling

- The validation-set approach
  - *Randomly* divide the available data into **Training** and **Validation** set
  - Fit the model using the Training set, evaluate the prediction on the Validation set
  - Error in Validation set *approximates* the error out of sample

MSE, ...

$\hat{f}(x) \longrightarrow \hat{y}' \longrightarrow Err(y', \hat{y}')$

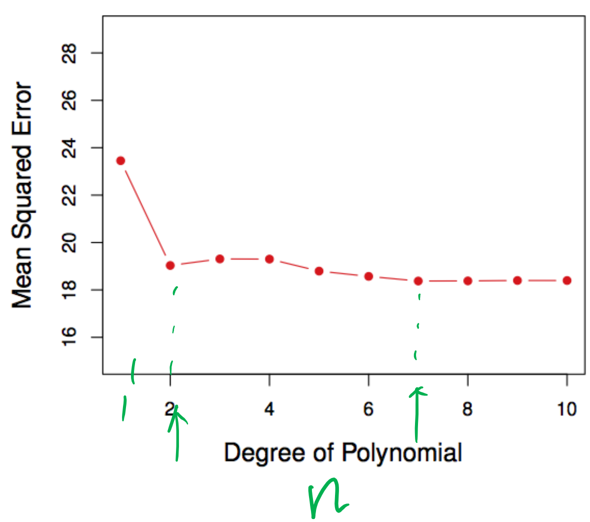


Example: Auto Data

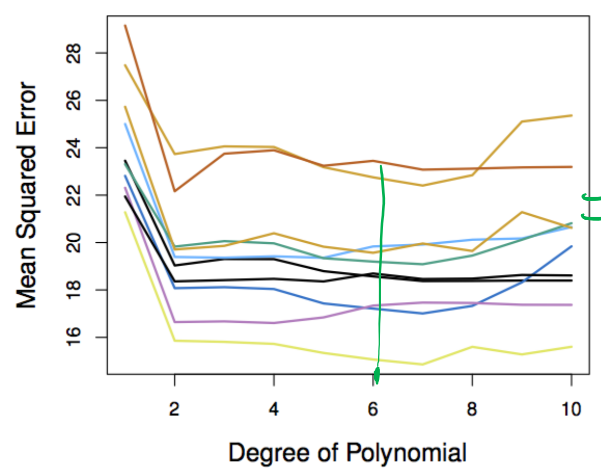
y x

- Suppose that we want to predict mpg from horsepower
- Compare models:
  - mpg ~ horsepower^n, n=1, 2, …, 10 ← 10 model, n^*
- Which model gives a better fit?
  - Randomly split Auto data set into training (196 obs.) and validation data (196 obs.)

Validation error for a single split



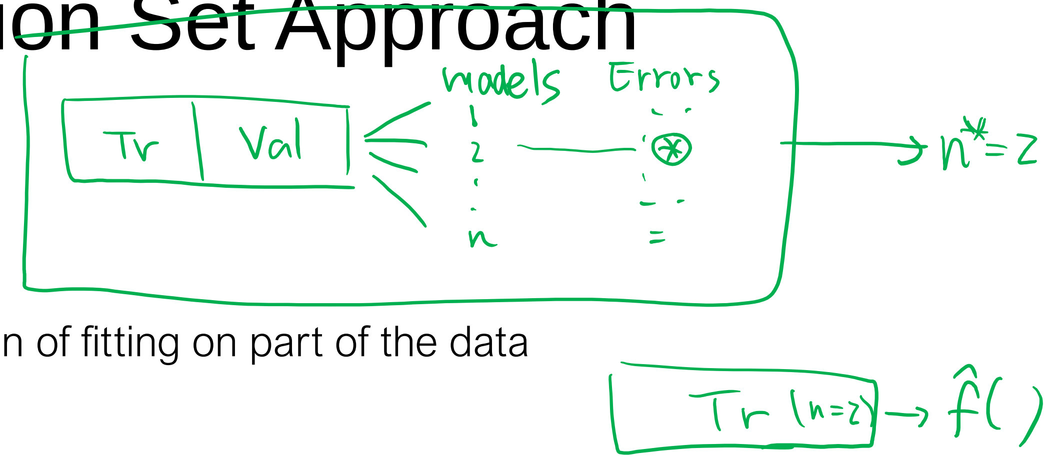
Validation error for 10 random splits



Avg

Model Selection

The Validation Set Approach



- Advantages:
  - Simple and easy to implement
  - Computationally efficient: one run of fitting on part of the data

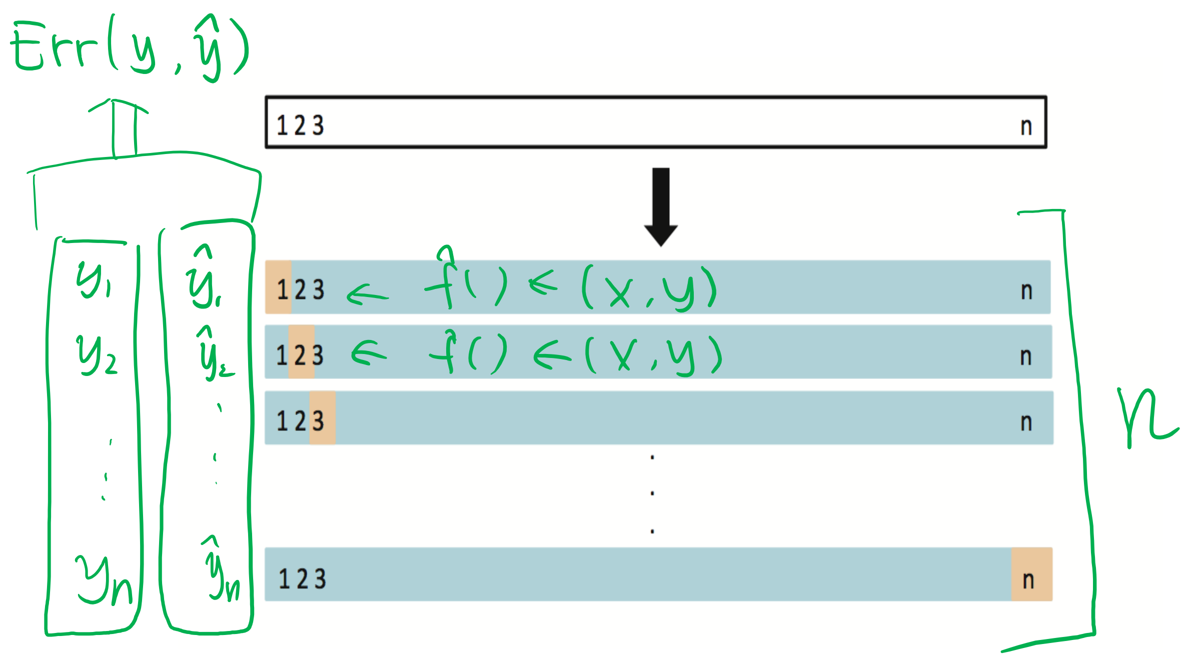
- Disadvantages:
  - Less data: only a subset of observations are used to fit the model (training data)
    - an overestimation of the out-of-sample error
  - Random splitting → Higher variance: the validation MSE can be highly variable because of the randomness in constructing Training and Validation datasets

→ average over multiple runs

Leave-One-Out Cross Validation (LOOCV)

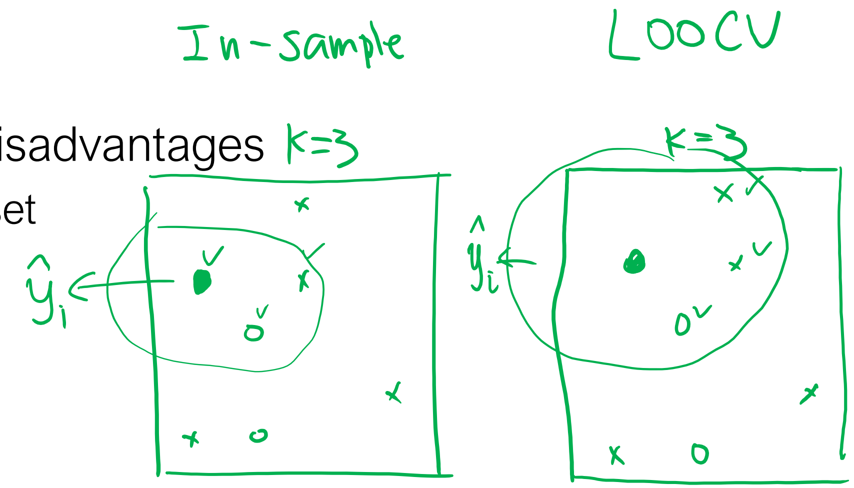
- For a dataset of size *n*, repeat the following *n* times
  - In iteration *i*, use the *i*-th data point for validation, the rest for training (*n* -*1*)
  - Fit the model with training data, and obtain validation error on point *i*
- The LOOCV error for the model is the average of the *n* validation errors:

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



LOOCV vs. the Validation Set Approach

- Designed to overcome the previous disadvantages
  - No randomness in sampling the dataset
  - Maximal utilization of data for training



- Disadvantages
  - LOOCV is computationally intensive (We fit the each model *n* times!)
    - Exception: least-squares linear or polynomial regression, KNN
  - High variance: each fold (iteration) is using almost the same data => high correlation

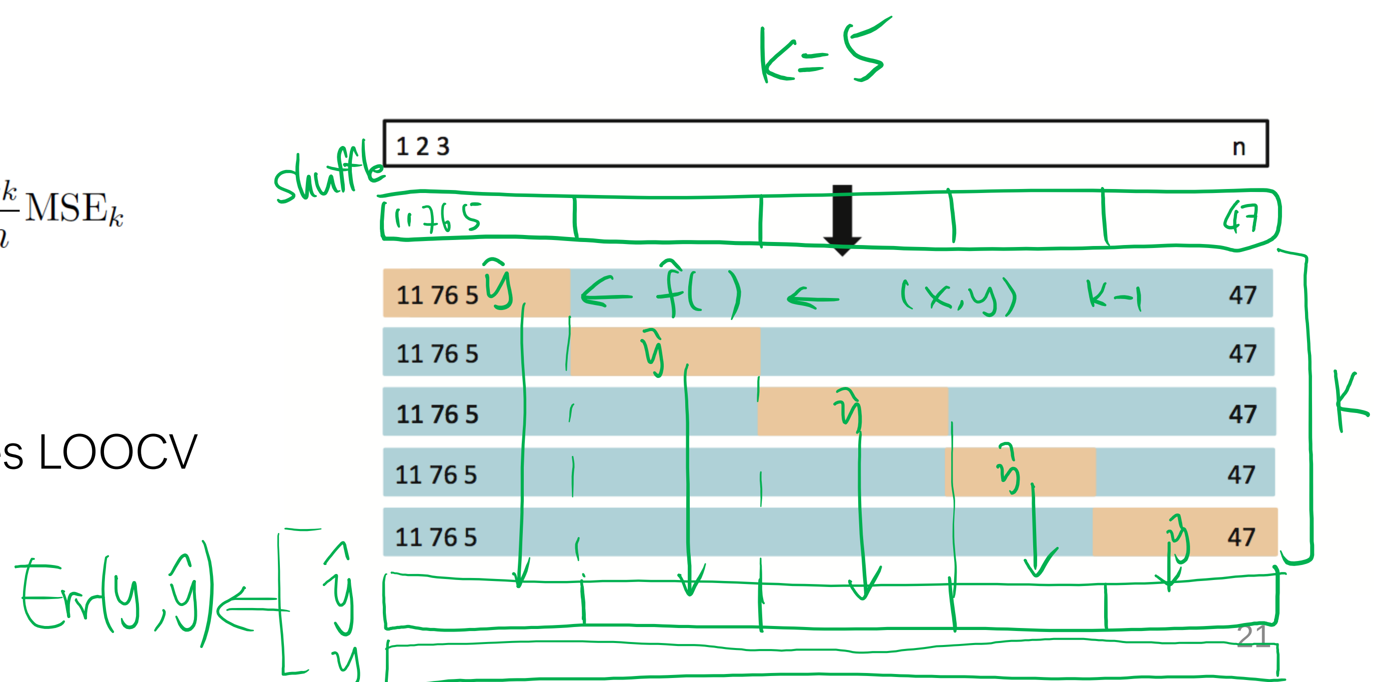
Randomness in the data sample

K-fold Cross Validation

- A trade-off between the validation-set approach and LOOCV
- Randomly divide the data into *K* different parts, repeat the following *K* times
  - Use the *i*-th part for validation, the remaining *K-1* parts for training
  - Fit the model with training data, and obtain validation error
- The K-fold cross validation error (parts may be of different size n_k)

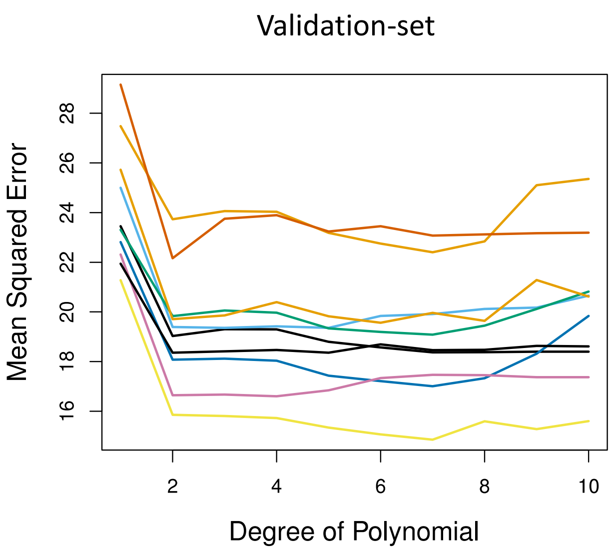
$$\mathrm{CV}_{(K)} = \sum_{k=1}^{K} \frac{n_k}{n} \mathrm{MSE}_k$$

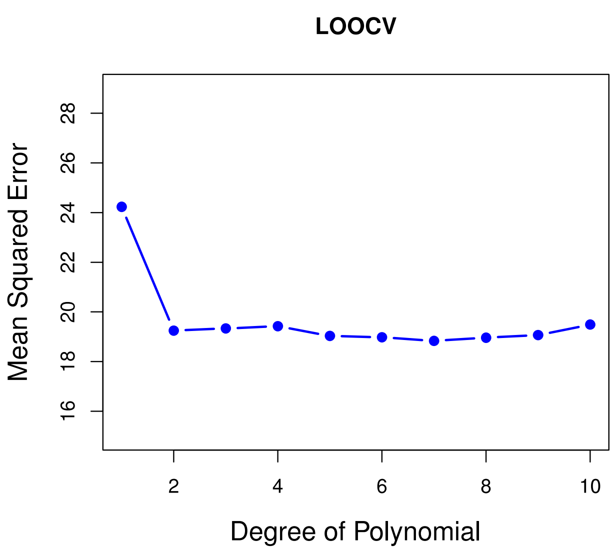
- If *K*=*n*, it becomes LOOCV

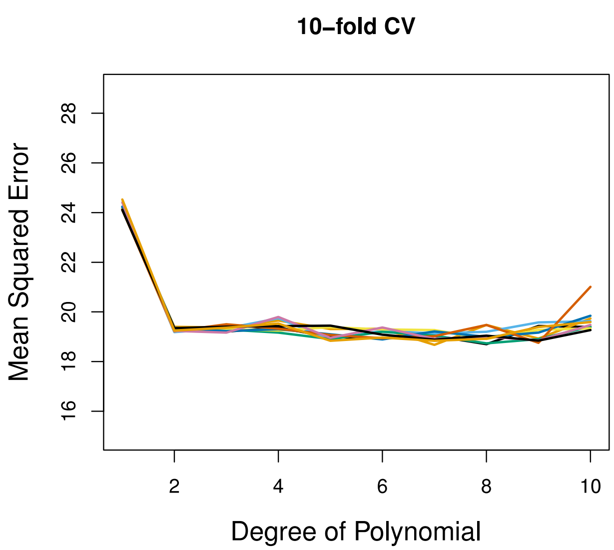


Auto Data: LOOCV vs. K-fold CV

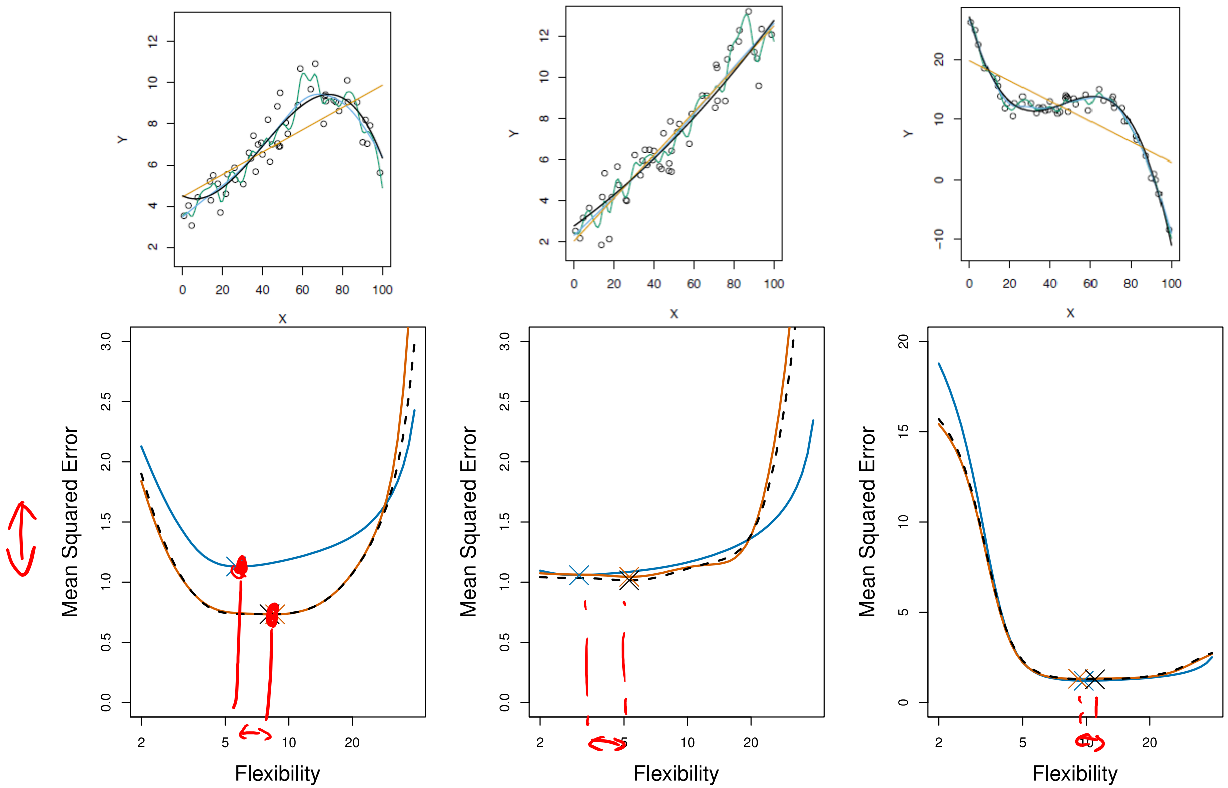
- Left: Validation-set, repeated many times
- Middle: LOOCV
- Right: 10-fold CV, repeated many times
  - K-fold CV is still random, but variability is small







K-fold Cross Validation on Simulated Data



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

Bias-Variance Trade-off for K-fold CV

- How to choose K?

$K=2$ $k=n$

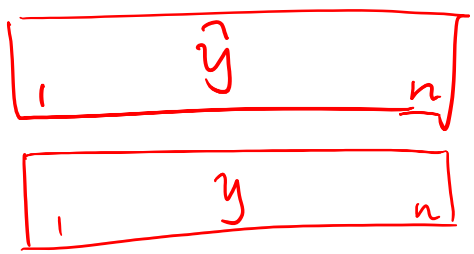
K-fold CV

Validation-set ⟷ LOOCV

- Recommendation
  - We tend to use K-fold CV with K = 5 or K = 10
  - It has been empirically shown that they yield test error rate estimates that suffer neither from excessively high bias, nor from very high variance

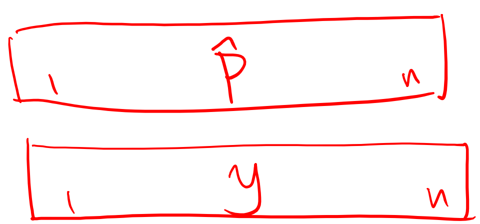
Cross Validation on Classification Problems

- Cross validation on classification problems in a similar manner
  - Replace MSE with error rate or other performance measures for classification
  - ROC / AUC on the validation dataset



$\Rightarrow \underline{\mathrm{Err}}(y, \hat{y})$

↳ mse . . . .



$\Rightarrow \mathrm{Err}(y, \hat{p})$

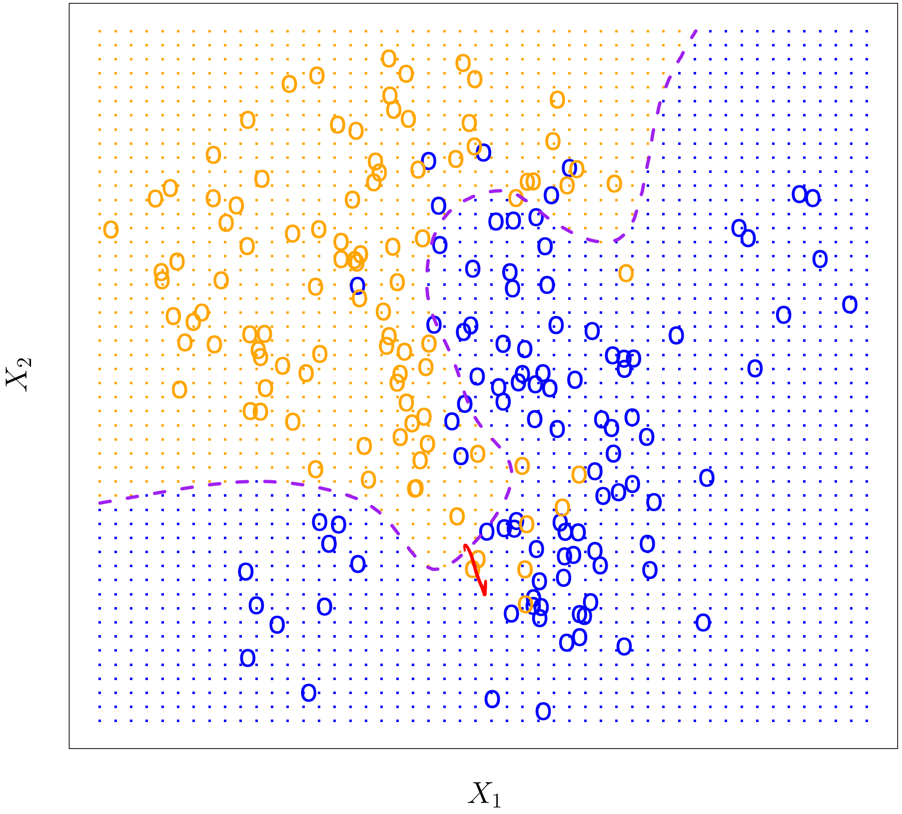
↳ misclassification

↳ AUC

↳ . . . .

CV to Choose Order of Polynomial

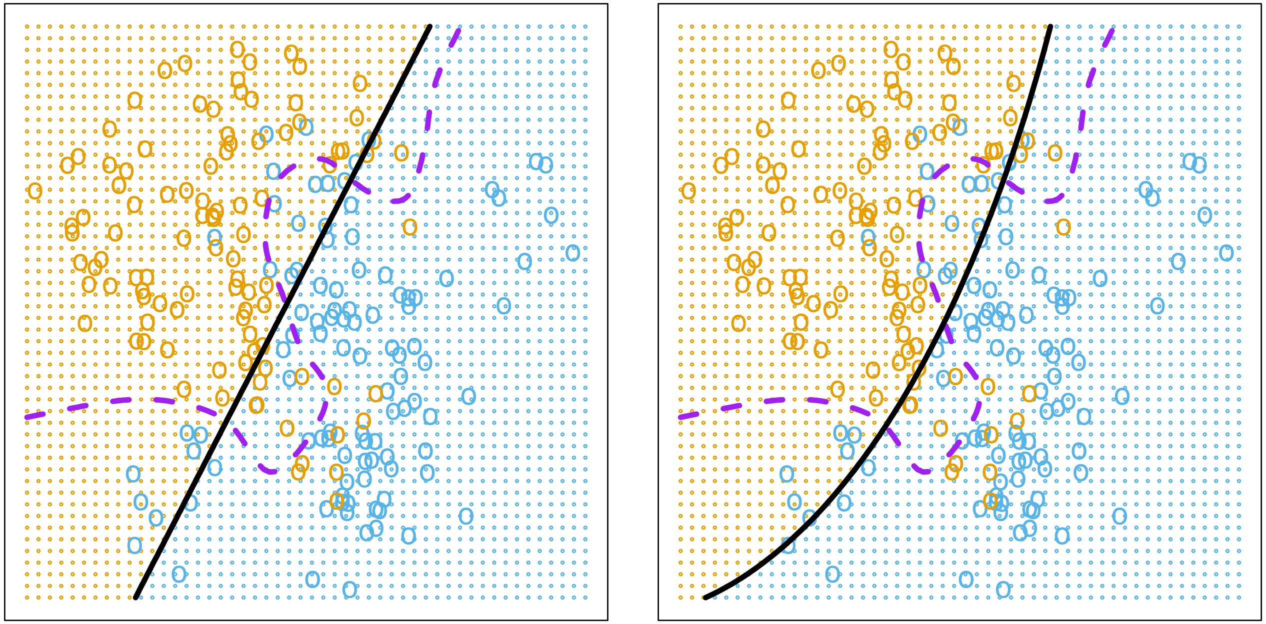
- The data set used is simulated (refer to Fig 2.13)
- The purple dashed line is the Bayes' boundary



CV to Choose Order of Polynomial

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

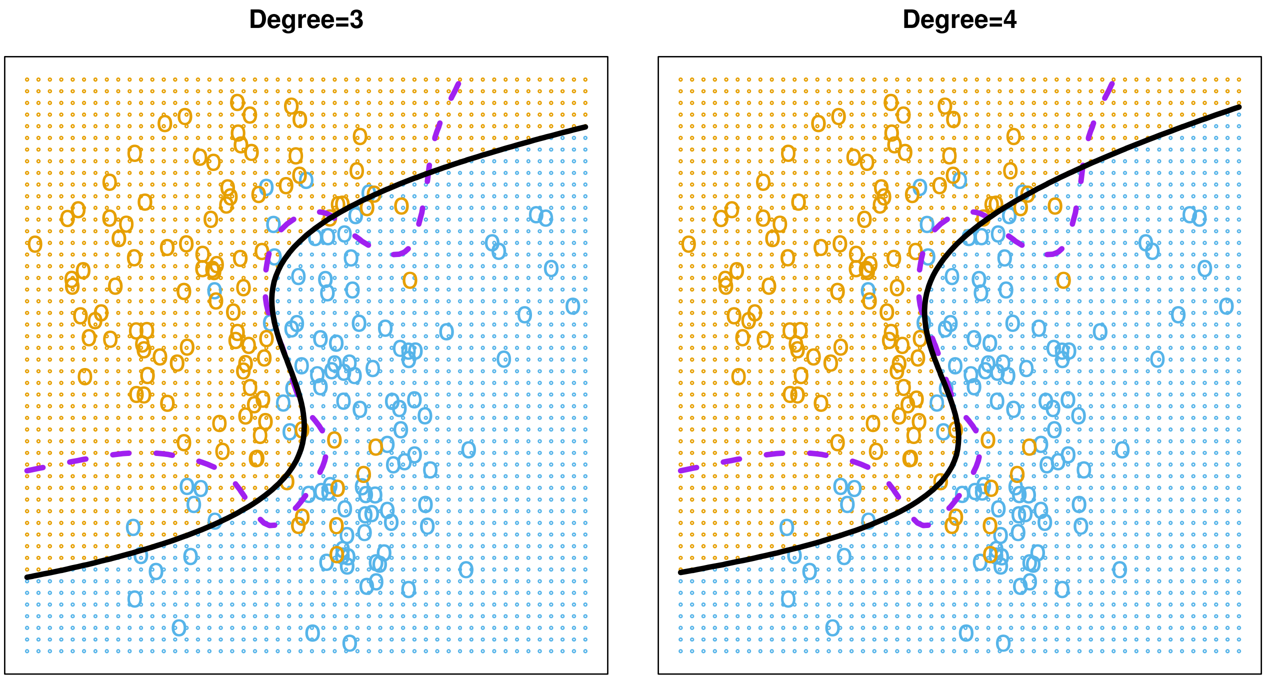
Degree=1 | Degree=2



Error Rate: 0.201 | Error Rate: 0.197

CV to Choose Order of Polynomial

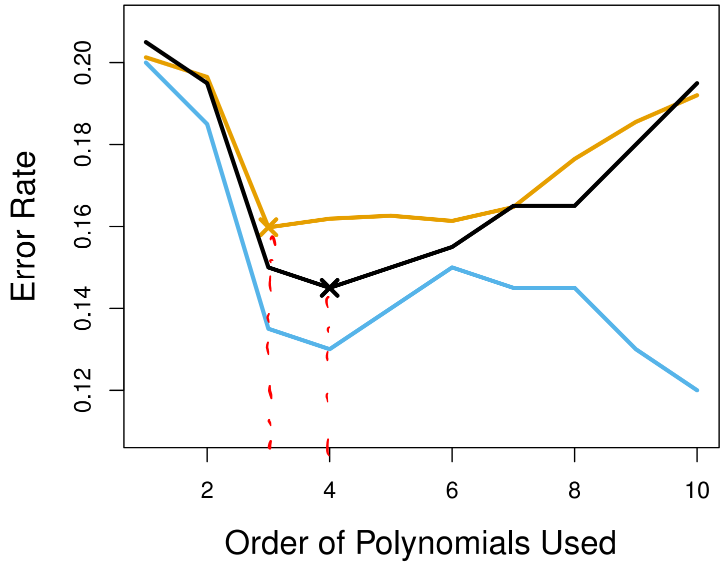
- Using cubic and quartic predictors, the accuracy of the model improves



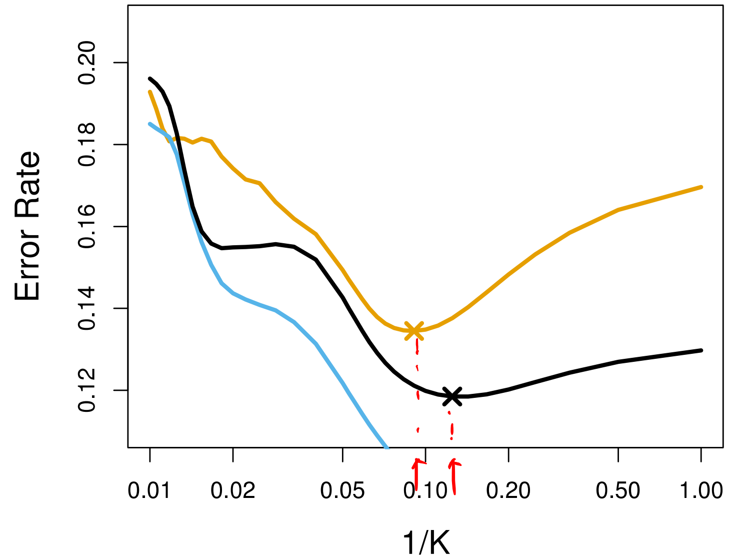
Error Rate: 0.160 Error Rate: 0.162

CV to Choose the Order

Logistic Regression



KNN



- Brown: Test Error
- Blue: Training Error
- Black: 10-fold CV Error

Even when the CV error and Test error are quite different, CV error serves as a good measure for model selection!

Cross-Validation in R

- Demo of the three approaches on GLM (using cv.glm())
  - 5-glm_validation.R

- Demo of LOOCV in kNN (using knn.cv())
  - 5-Mixture_knn_cv.R

- Demo of k-fold CV in logistic regression (using cv.glm() and cost)
- Demo of manual k-fold CV in logistic regression
  - 5-Mixture_LogisticRegression.R

cv.glm( )

- Set.seed( )

run = 1, . . , RUN

model i = 1, . . , I

cv.glm( )

- split data into K ~~fold~~ fold

k = 1, . . . , K

- fit model with K-1
- predict on 1 fold
- Error measure

manual

- set.seed( )

run = 1, . . . , RUN

✱ split data into K folds

model family I

i = 1, . . . , I

k = 1, . . . , K

- fit
- predict
- error

model family II

j = 1, . . . , J

k = 1, . . . , K

- fit
- predict
- error