

DATASCI 347 Machine Learning

Lecture 6: Cross-Validation

Ruoxuan Xiong

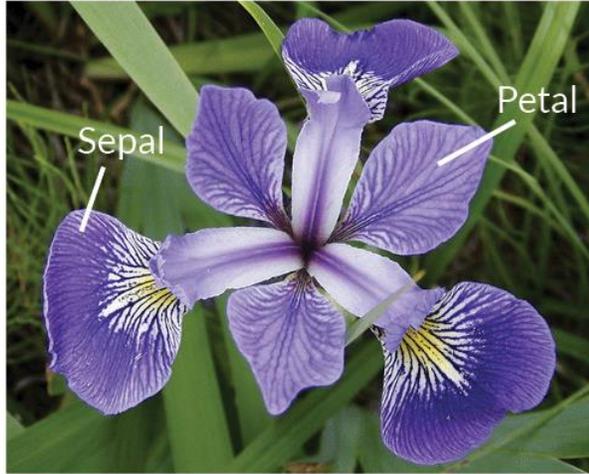
Suggested reading: ISL Chapter 5

Lecture plan

- Review of LDA and QDA
- Lab session of classification methods
- Cross validation



Example of LDA/QDA: An iris data set



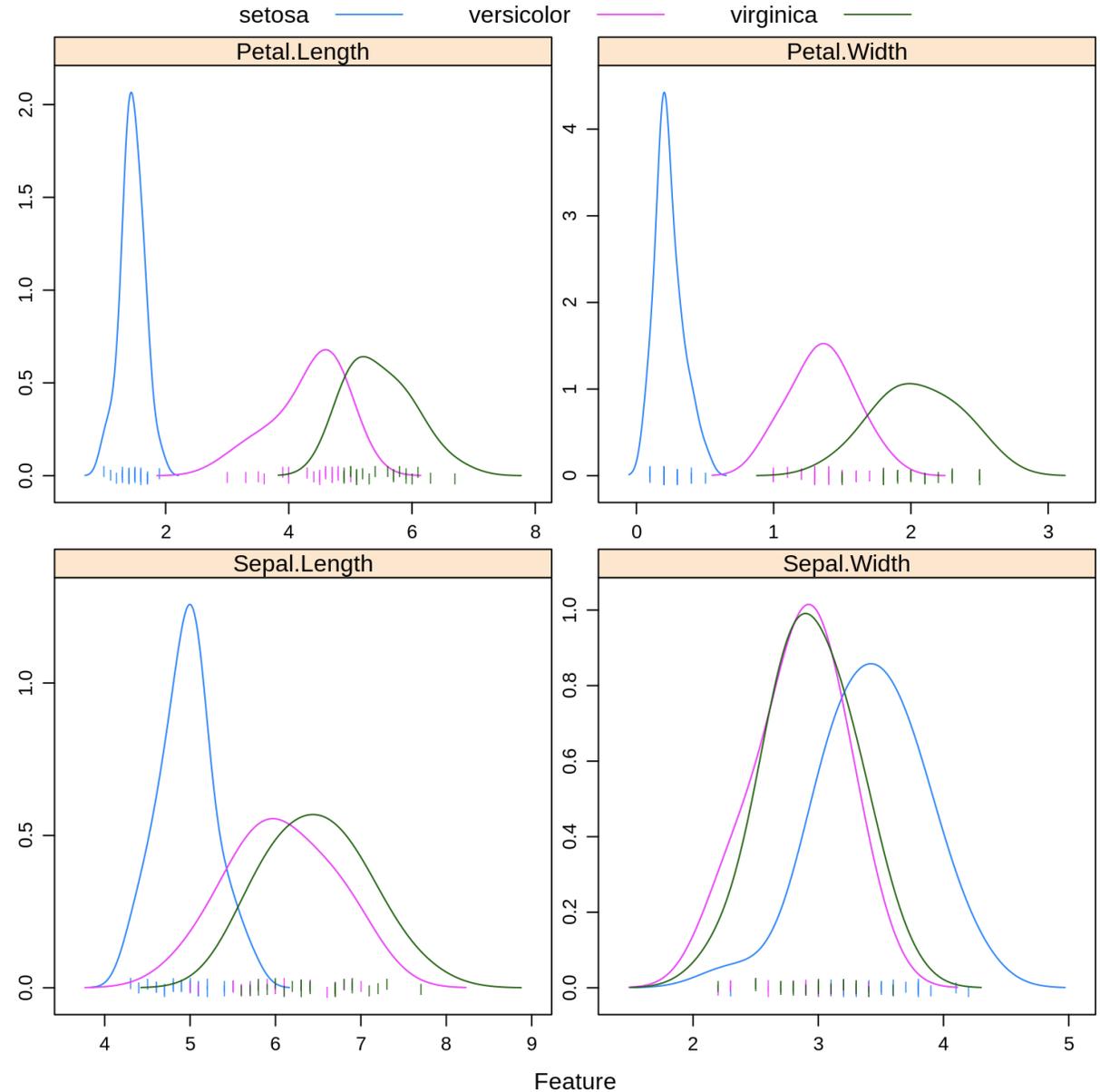
Iris Versicolor



Iris Setosa



Iris Virginica



LDA

- For each class k , we model $P(X = x|Y = k) = f_k(x)$ as a *Multivariate Normal Distribution* $N(\mu_k, \Sigma)$ with mean μ_k and covariance matrix Σ
- We estimate $\hat{P}(X = x|Y = k)$ as $N(\hat{\mu}_k, \hat{\Sigma})$ and $\hat{P}(Y = k) = \hat{\pi}_k$
- We apply to Bayes theorem to obtain $P(Y = k | X = x)$

$$\hat{P}(Y = k | X = x) = \frac{\hat{P}(Y = k, X = x)}{\hat{P}(X = x)} = \frac{\hat{P}(X = x | Y = k)\hat{P}(Y = k)}{\sum_j \hat{P}(X = x | Y = j)\hat{P}(Y = j)}$$

QDA

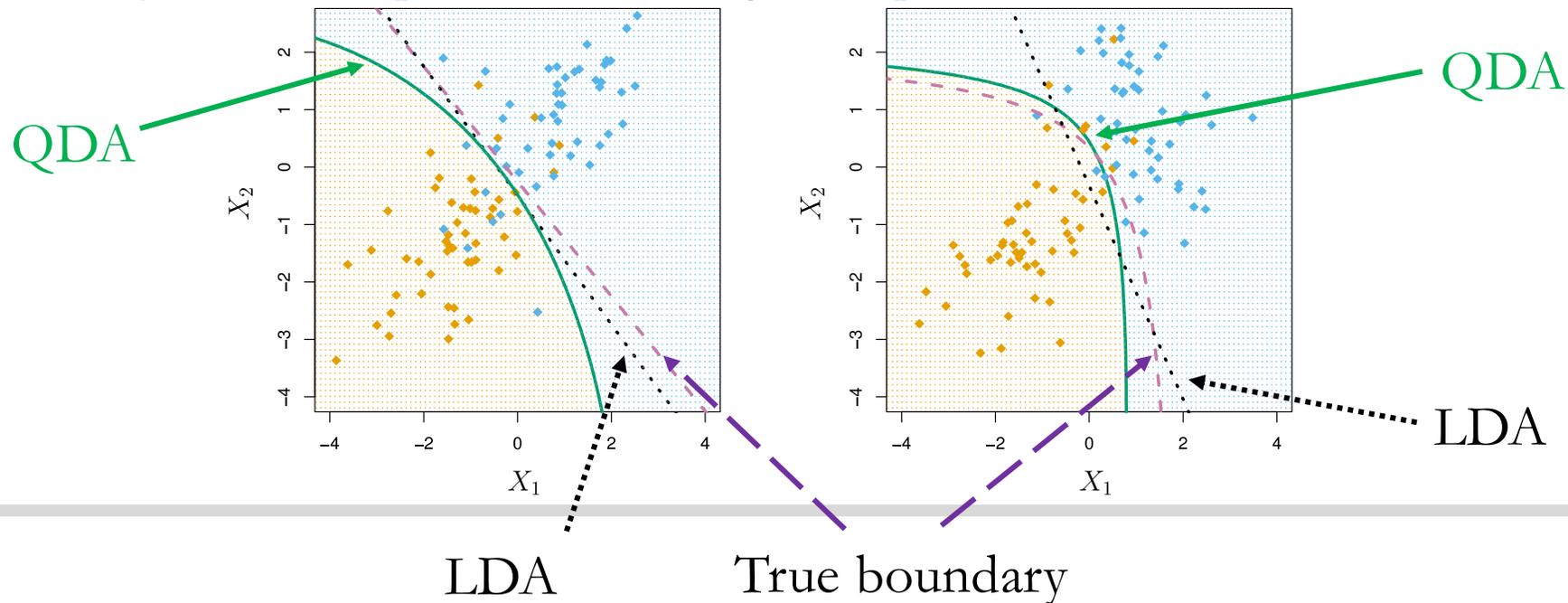
- For each class k , we model $P(X = x|Y = k) = f_k(x)$ as a *Multivariate Normal Distribution* $N(\mu_k, \Sigma_k)$ with mean μ_k and covariance matrix Σ_k
- We estimate $\hat{P}(X = x|Y = k)$ as $N(\hat{\mu}_k, \hat{\Sigma}_k)$ and $\hat{P}(Y = k) = \hat{\pi}_k$
- We apply to Bayes theorem to obtain $P(Y = k | X = x)$

$$\hat{P}(Y = k | X = x) = \frac{\hat{P}(Y = k, X = x)}{\hat{P}(X = x)} = \frac{\hat{P}(X = x | Y = k)\hat{P}(Y = k)}{\sum_j \hat{P}(X = x | Y = j)\hat{P}(Y = j)}$$



Comparison between LDA and QDA

- **Decision boundary:** the set of points in which 2 classes do just as well
 - LDA has *linear* decision boundary
 - QDA has *quadratic* decision boundary
- **Bias-variance tradeoff**
 - LDA is less flexible but has a smaller variance. Small sample size n : LDA
 - QDA requires more parameters. Large sample size n : QDA



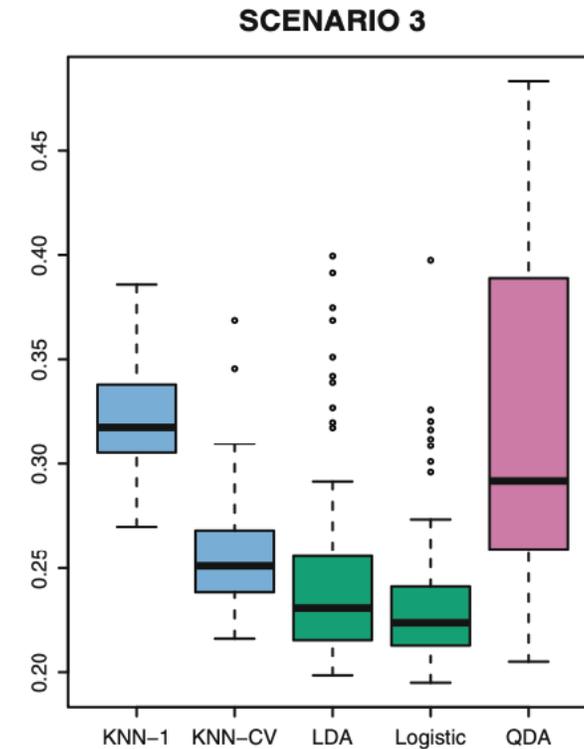
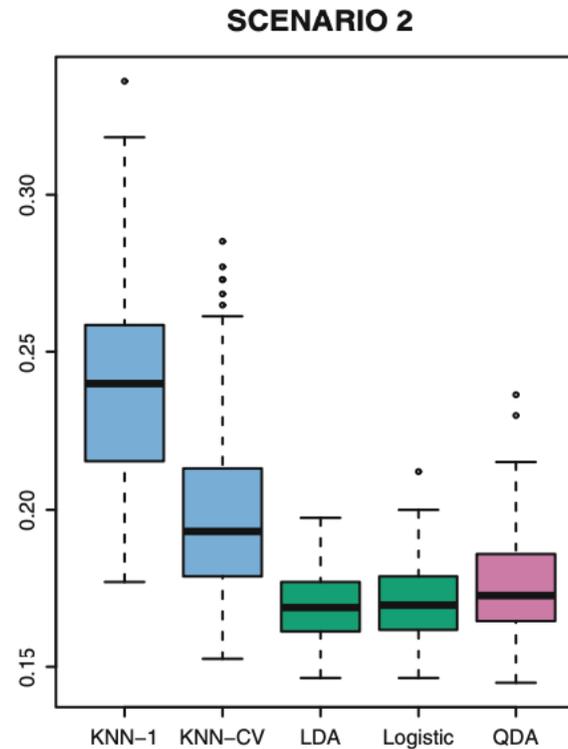
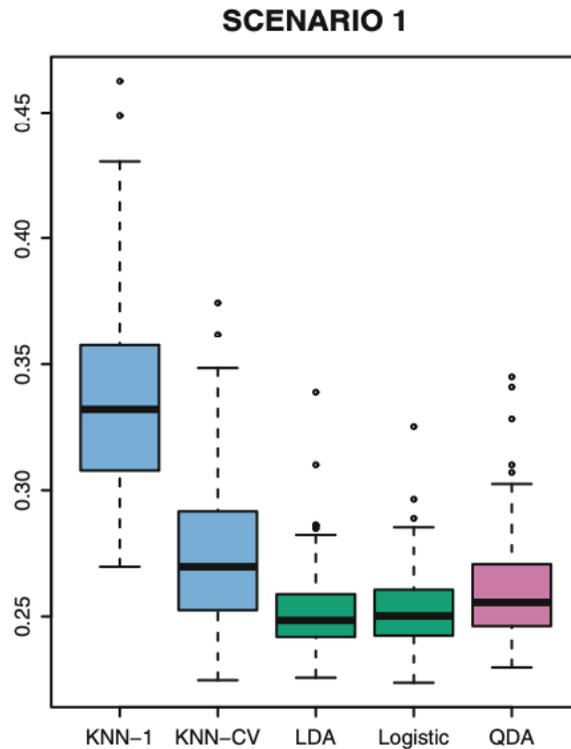
Examples: True decision boundaries are linear

- Data generating process: two predictors X_1 and X_2 , two classes in Y

X_1 and X_2 are drawn from **uncorrelated Normal distributions** with a different mean in each class

Same as Scenario 1, but correlation between X_1 and X_2 is -0.5

X_1 and X_2 are sampled from t -distribution

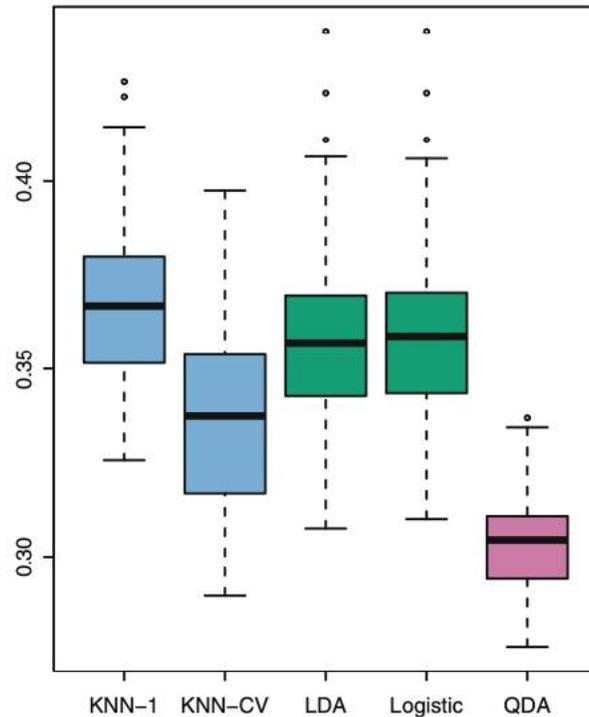


Examples: True decision boundaries are nonlinear

- Data generating process: two predictors X_1 and X_2 , two classes in Y

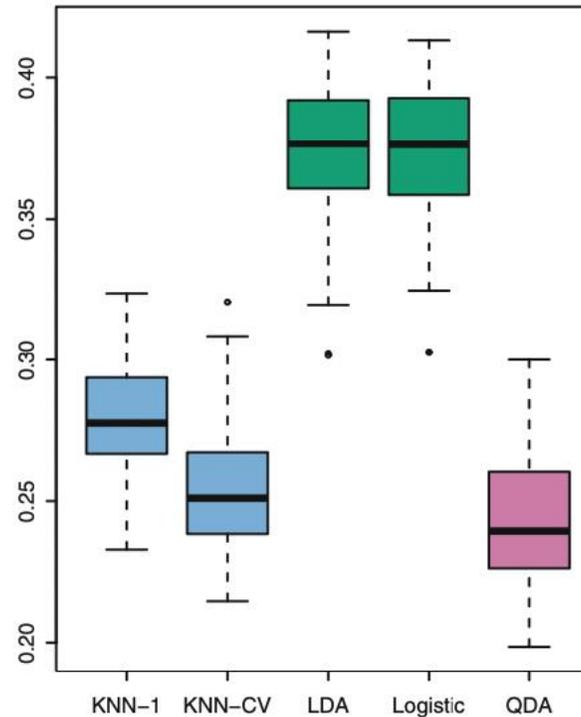
X_1 and X_2 are drawn from Normal distributions. First class: correlation between X_1 and X_2 is -0.5 . Second class: correlation is 0.5

SCENARIO 4



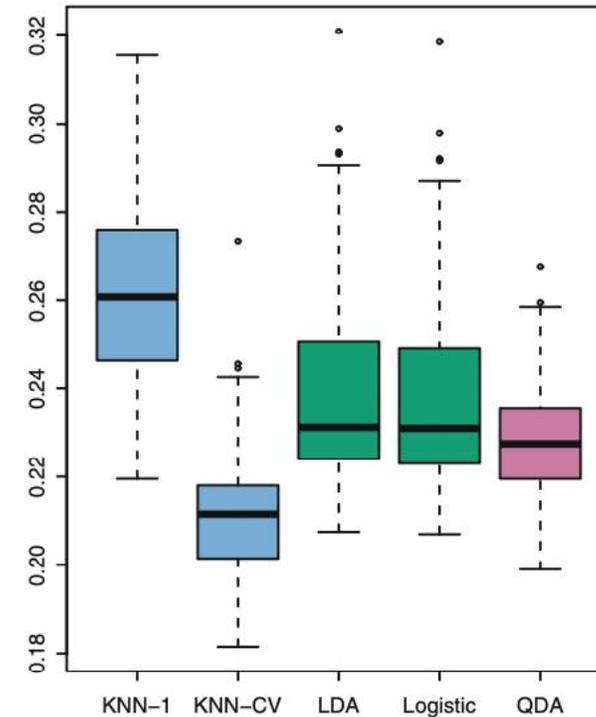
X_1 and X_2 are drawn from **uncorrelated Normal distributions**. Y is sampled from logic model using X_1^2 , X_2^2 and X_1X_2

SCENARIO 5



Same as Scenario 5. Y is sampled from a more complicated nonlinear function

SCENARIO 6



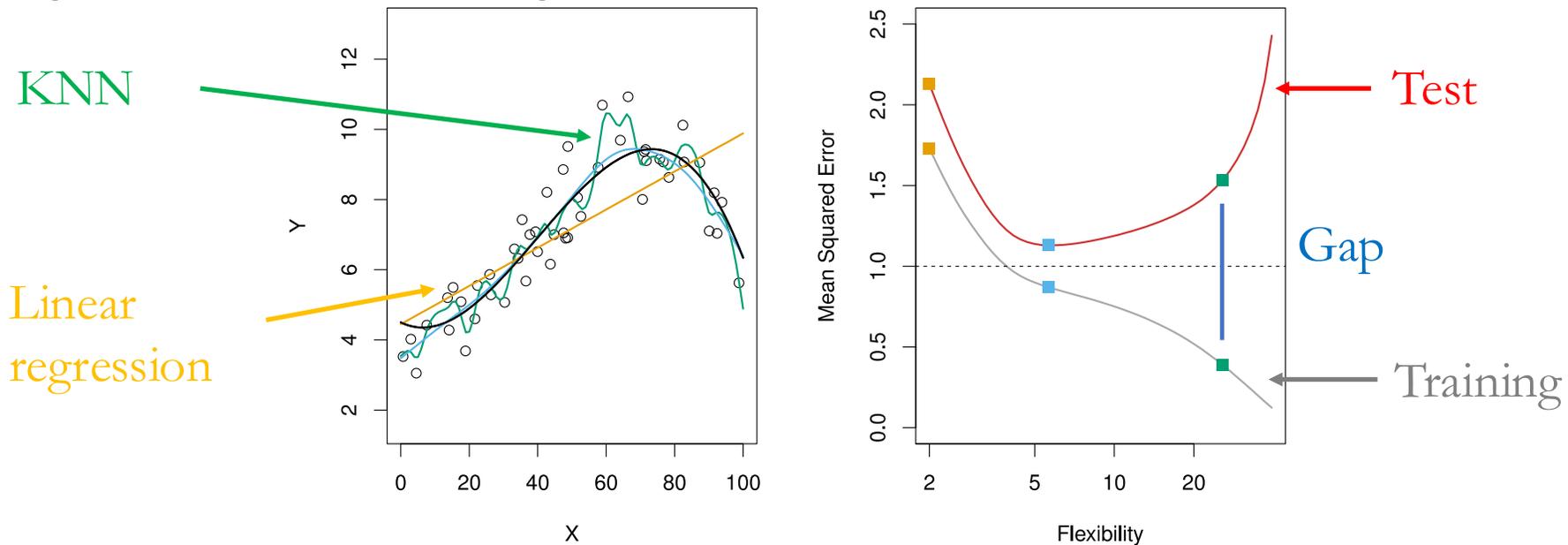
Lecture plan

- Review of LDA and QDA
- Lab session of classification methods
- Cross validation



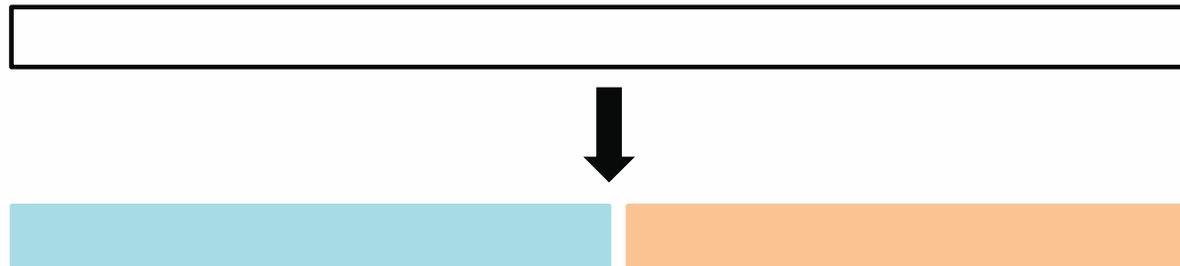
Motivation

- **Supervised learning:** Minimize test error
 - However, we only have access to the training error
 - There is often a gap between them
- **Illustration:** Suppose we know what f is (the black curve)
 - We generate data according to f as simulated data (in circles)



Validation set approach

- **Goal of validation set approach:** Using the training data set alone, find out the test error as closely as possible
- **A first attempt:**
 - Randomly split the data in two parts
 - Train the method in the first part
 - Compute the error on the second part

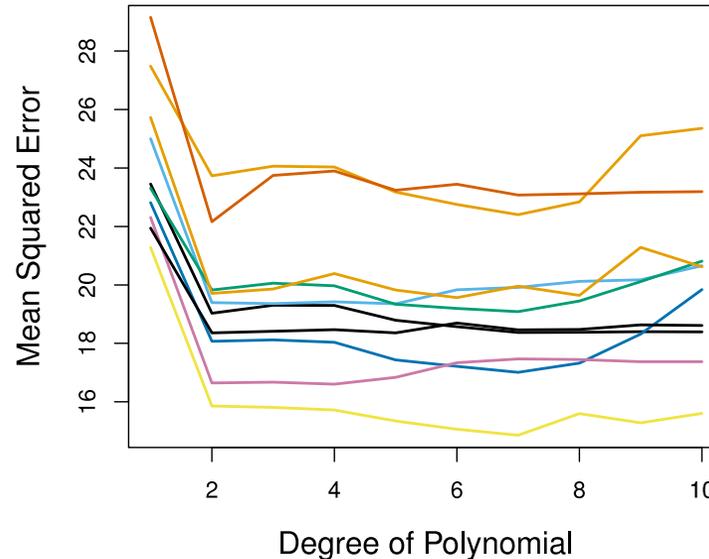


Example

- Estimate **miles per gallon (mpg)** from engine **horsepower**
 - Auto data: **horsepower**, gas mileage, and other information for 392 vehicles
- **Simple linear regression**
 - $\text{mpg} = \beta_0 + \beta_1 \text{horsepower}$
- **Multiple linear regression with polynomial features**
 - $\text{mpg} = \beta_0 + \beta_1 \text{horsepower} + \beta_2 \text{horsepower}^2$
 - $\text{mpg} = \beta_0 + \beta_1 \text{horsepower} + \beta_2 \text{horsepower}^2 + \beta_3 \text{horsepower}^3$
- **Which polynomial is the right relationship?**
 - **Resampling**
 - Partition 392 samples into two sets with equal size
 - One is the training set and the other one is the validation set

Example

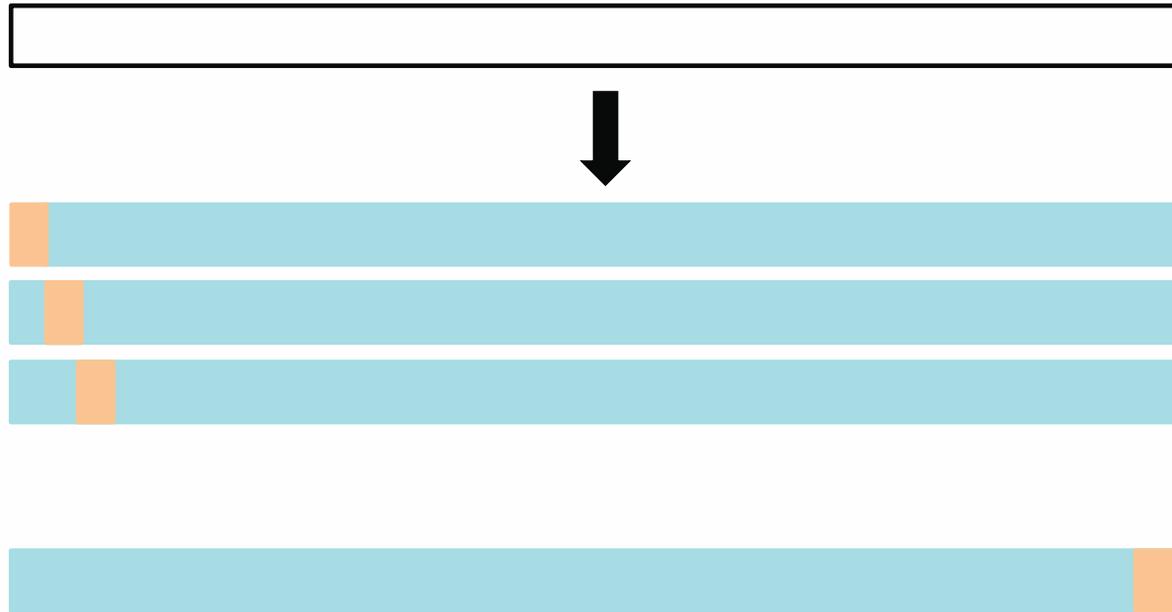
- Estimate miles per gallon (mpg) from engine horsepower



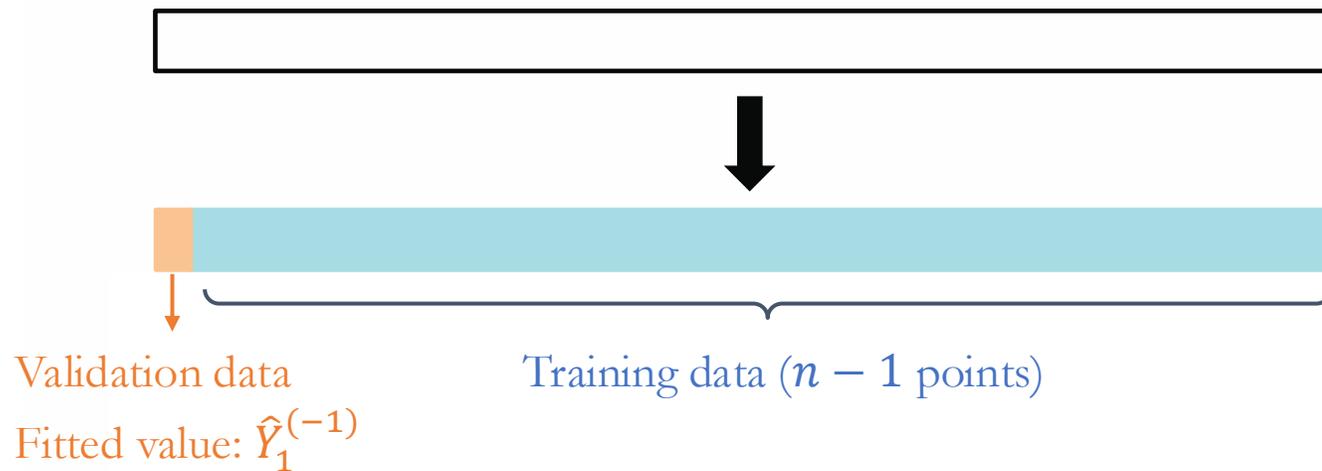
- Each line is the result with a different random split of the data into two parts
- Every split yields a different estimate of the error 😞

Leave-one-out cross-validation

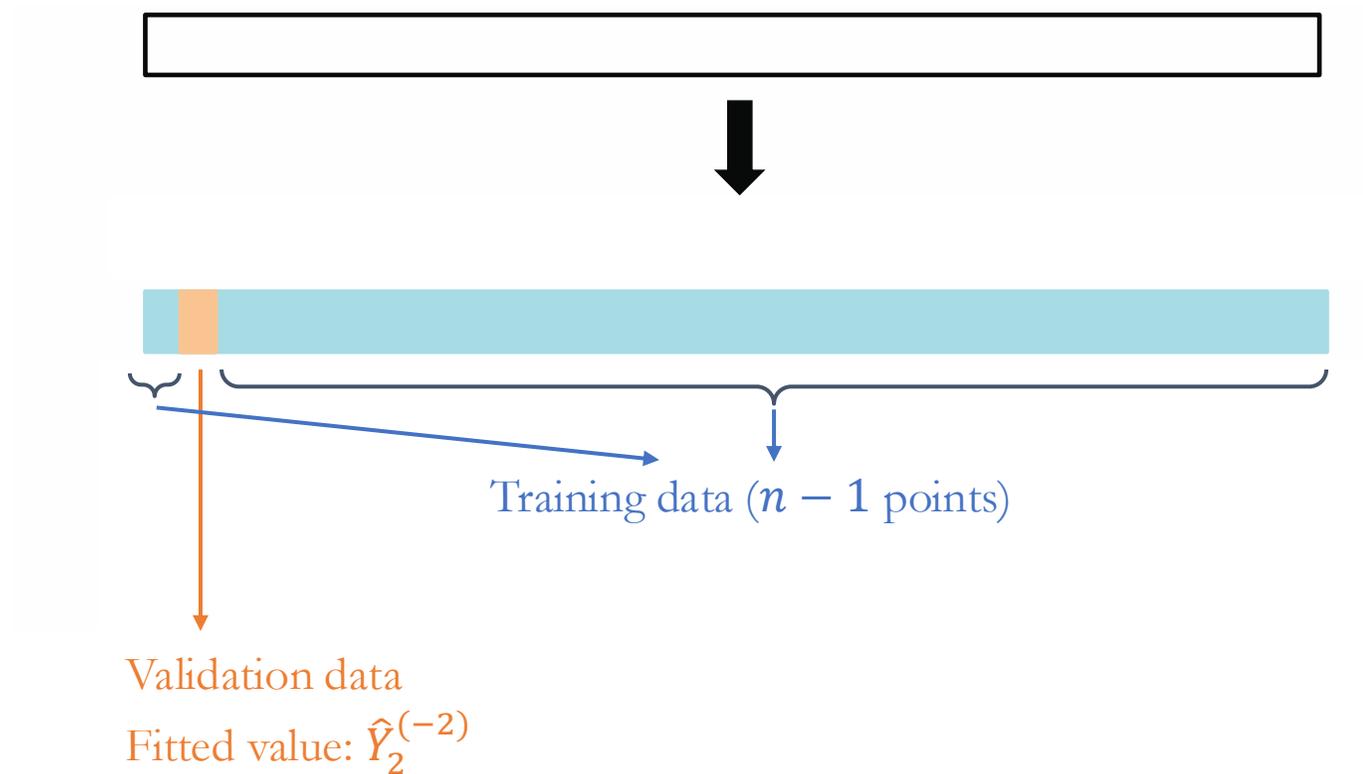
- **Leave-one-out cross-validation**
- For every $i = 1, \dots, n$,
 - Train the model on every point except i ;
 - Compute the test error on the hold-out point;
 - Average over all n points.



Leave-one-out cross-validation



Leave-one-out cross-validation



Leave-one-out cross-validation

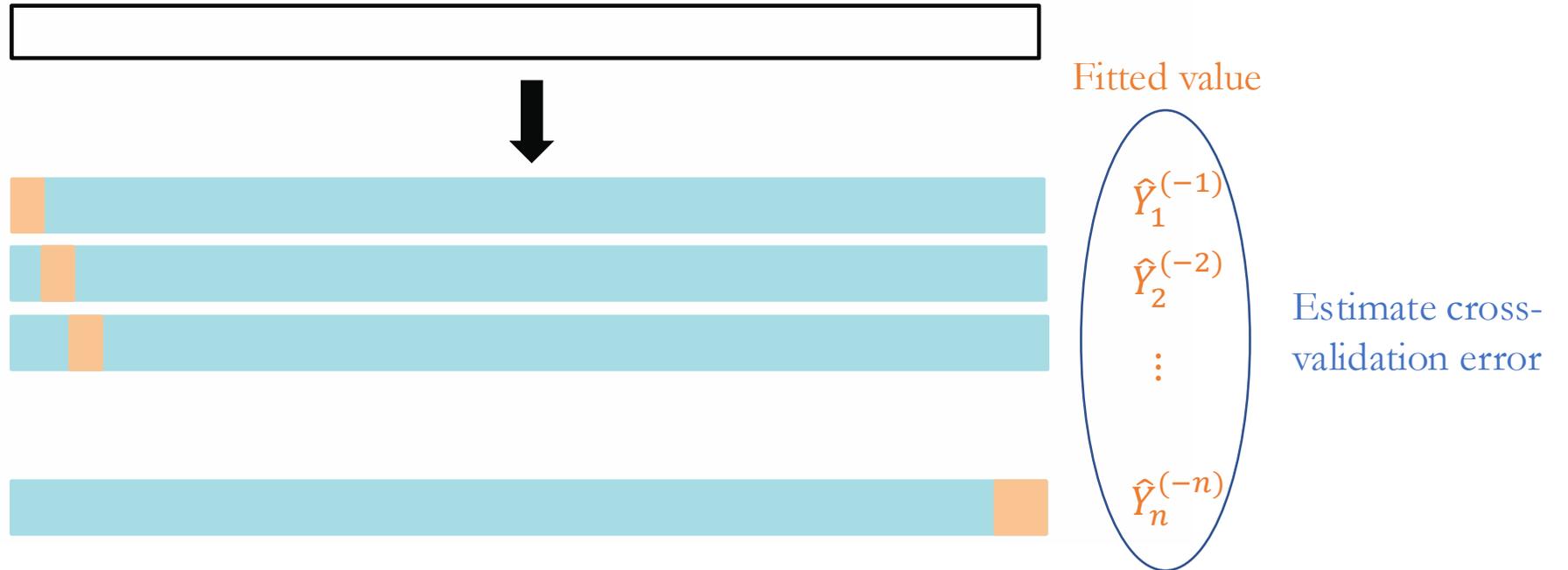


Training data ($n - 1$ points)

Validation data

Fitted value: $\hat{Y}_n^{(-n)}$

Leave-one-out cross-validation



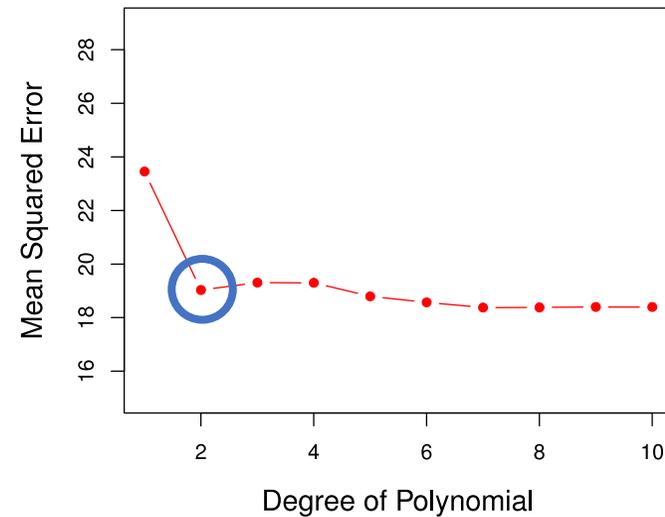
Cross-validation error

- **Regression** with mean squared loss
 - $\hat{Y}_i^{(-i)}$: Prediction for the i th sample without using the i th sample
 - $CV_{(n)} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i^{(-i)})^2$

- **Classification** with zero-one loss
 - $\hat{Y}_i^{(-i)}$: Prediction for the i th sample without using the i th sample
 - $CV_{(n)} = \frac{1}{n} \sum_{i=1}^n 1 \left[Y_i \neq \hat{Y}_i^{(-i)} \right]$

Example

- Estimate **miles per gallon (mpg)** from engine **horsepower**
- The LOOCV error curve



LOOCV has low bias and no randomness

- Each training set in LOOCV has $n - 1$ observations, almost as many as are in the entire data set
 - LOOCV tends not to overestimate the test error rate by too much (**low bias**)
 - There is **no randomness** in the training/validation set splits

Computational concerns

- Computing $CV_{(n)}$ can be computationally expensive, since it involves fitting the model n times
- What if we use a model other than linear or polynomial regression?
- **k -fold cross-validation**: Split the data into k equal sized subsets
 - Only requires fitting the model k times
 - $\frac{n}{k}$ times speed up over leave one out cross-validation