Cross validation

Readings for today

- Chapter 5: Resampling methods. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning: with applications in R (Vol. 6). New York: Springer.

Topics

1. Goals of cross validation

2. Leave one out cross validation

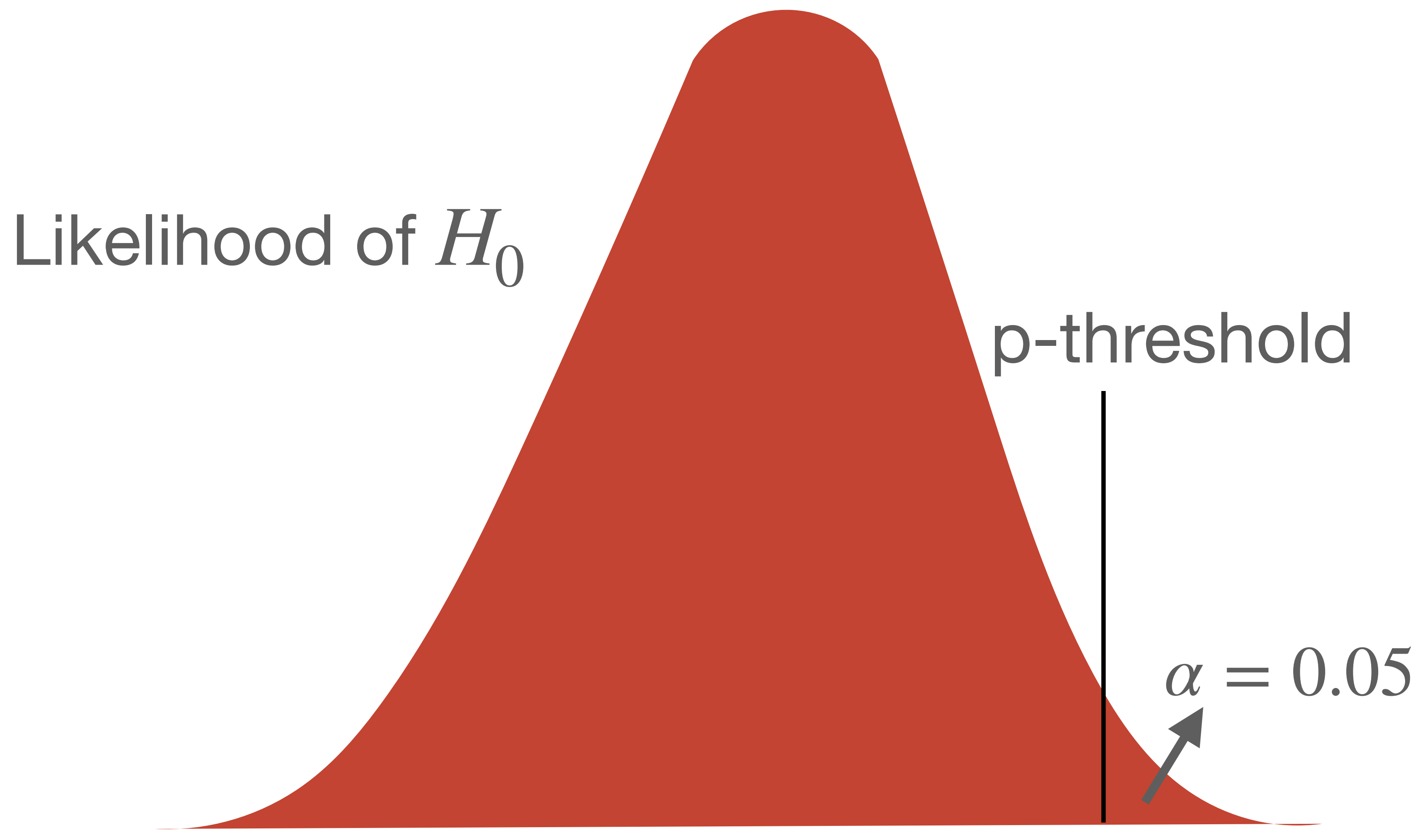
3. K-fold cross validation

Goals of cross validation

Inferring with prediction

Inference

Q: What is that the probability the null hypothesis (H_0) is true?



$p = P(Y \mid H_0 = \text{true})$

Prediction

Q: What is that the probability $\hat{y}$ predicts y better than chance (e.g., $\bar{y}$)?

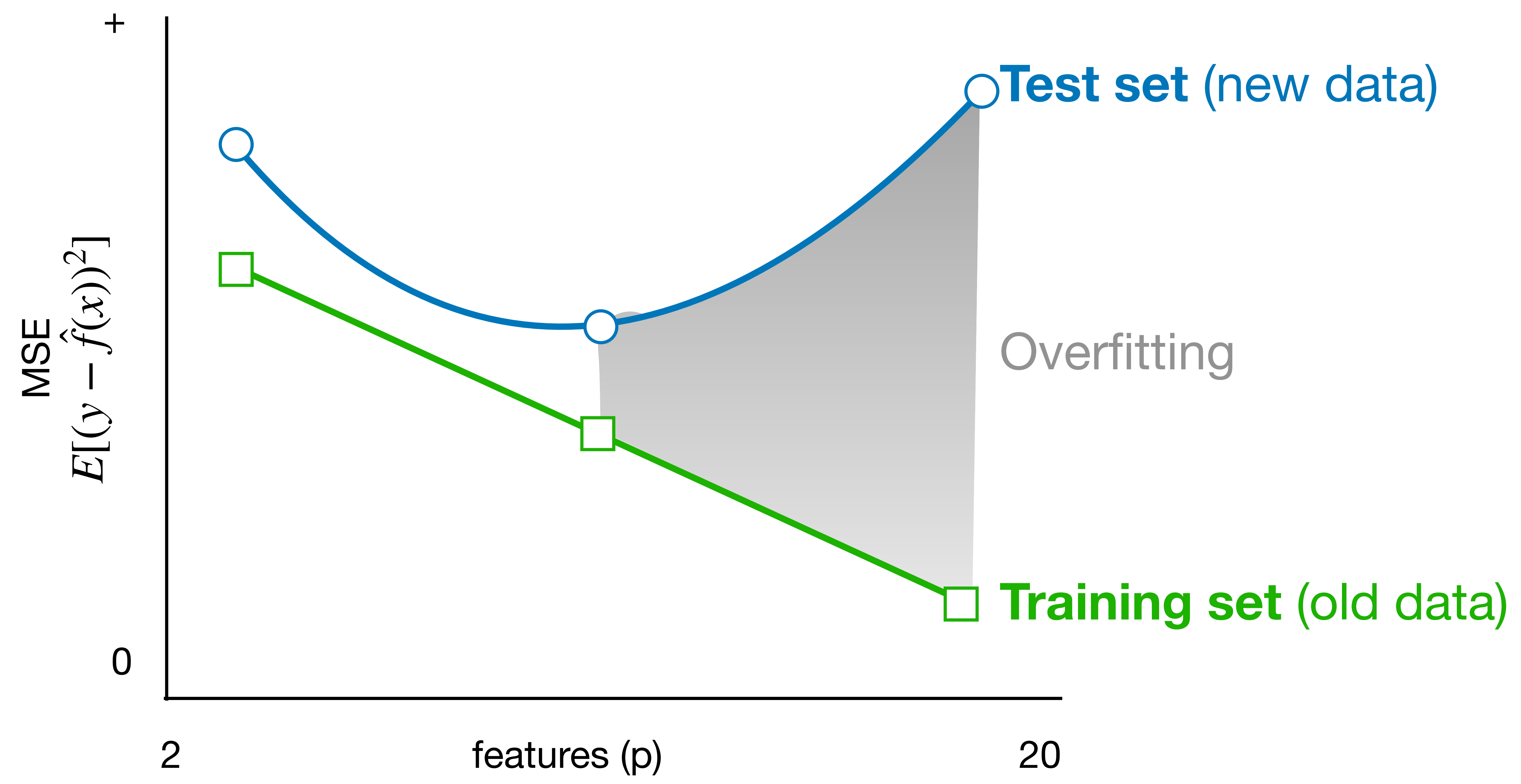
Learn: $\hat{Y} = \hat{f}_{train}(X_{train})$

Test: $\hat{Y} = \hat{f}_{train}(X_{test})$

$$H_0: \underbrace{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}_{\text{RSS}} \approx \underbrace{\sum_{i=1}^{n} (y_i - \bar{y}_i)^2}_{\text{TSS}}$$

(James et al. 2013)

Training vs. Test error



Evaluate: $RSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$

Test Error: $\sum_{i=1}^{n} (y_{test,i} - \hat{y}_{test,i})^2,$

$$\hat{y}_{test,i} = \sum_{j=0}^{p} \hat{\beta}_j^{train} x_{test,i}$$

Training Error: $\sum_{i=1}^{n} (y_{train,i} - \hat{y}_{train,i})^2,$

$$\hat{y}_{train,i} = \sum_{j=0}^{p} \hat{\beta}_j^{train} x_{train,i}$$

Validation sets:

Subsets of the data divided into training & test sets.

(James et al. 2013)

Types of validation sets

- **Leave one out cross validation (LOOCV)**
- **K-fold cross validation**

(James et al. 2013)

Leave one out cross validation (LOOCV)

LOOCV

Full dataset

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = f\left(\begin{pmatrix} x_{1,1} & \cdots & x_{1,p} \\ x_{2,1} & \cdots & x_{2,p} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{pmatrix}\right)$$

Test set

$$\left(\hat{y}_1\right) = \hat{f}_{train}\left(\begin{pmatrix} x_{1,1} & \cdots & x_{1,p} \end{pmatrix}\right)$$

Training set

$$\begin{pmatrix} \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{pmatrix} = \hat{f}_{train}\left(\begin{pmatrix} x_{2,1} & \cdots & x_{2,p} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{pmatrix}\right)$$

Error:

regression: $(y_1 - \hat{y}_1)^2$

classification: $I(y_1 = \hat{y}_1)$

(James et al. 2013)

LOOCV

Full dataset

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = f\left(\begin{pmatrix} x_{1,1} & \cdots & x_{1,p} \\ x_{2,1} & \cdots & x_{2,p} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{pmatrix}\right)$$

Test set

$$(\hat{y}_2) = \hat{f}_{train}((x_{2,1} \quad \cdots \quad x_{2,p}))$$

Training set

$$\begin{pmatrix} \hat{y}_1 \\ \hat{y}_3 \\ \vdots \\ \hat{y}_n \end{pmatrix} = \hat{f}_{train}\left(\begin{pmatrix} x_{1,1} & \cdots & x_{1,p} \\ x_{3,1} & \cdots & x_{3,p} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{pmatrix}\right)$$

Error:

regression: $(y_1 - \hat{y}_1)^2, (y_2 - \hat{y}_2)^2$

classification: $I(y_1 = \hat{y}_1), I(y_2 = \hat{y}_2)$

(James et al. 2013)

LOOCV

Full dataset

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = f\left(\begin{pmatrix} x_{1,1} & \cdots & x_{1,p} \\ x_{2,1} & \cdots & x_{2,p} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{pmatrix}\right)$$

Test set

$$(\hat{y}_n) = \hat{f}_{train}((x_{n,1} \quad \cdots \quad x_{n,p}))$$

Training set

$$\begin{pmatrix} \hat{y}_1 \\ \vdots \\ \hat{y}_{n-1} \end{pmatrix} = \hat{f}_{train}\left(\begin{pmatrix} x_{1,1} & \cdots & x_{1,p} \\ \vdots & & \vdots \\ x_{n-1,1} & \cdots & x_{n-1,p} \end{pmatrix}\right)$$

Error:

regression: $(y_1 - \hat{y}_1)^2, (y_2 - \hat{y}_2)^2, \ldots, (y_n - \hat{y}_n)^2$

classification: $I(y_1 = \hat{y}_1), I(y_2 = \hat{y}_2), \ldots, I(y_n = \hat{y}_n)$

(James et al. 2013)

LOOCV algorithm

<u>Step 1</u>: Take all variables from a single observation, $\overrightarrow{x_i} = (x_{i,1} \quad \cdots \quad x_{i,p})$ and y_i. Set aside as a test set.

<u>Step 2</u>: Make new X_{train} with all $\neg i$ observations.

<u>Step 3</u>: Fit $\hat{f}_{train}(X_{train})$.

<u>Step 4</u>: Evaluate test error on y_i using $\hat{f}_{train}$ from Step 3.

<u>Step 5</u>: Repeat Steps 1 - 4 for all n observations.

(James et al. 2013)

The good and the bad of LOOCV

Pros:

- High power on training set.
- Stable estimate of $\hat{f}_{train}(X_{train})$.
- Direct measure of generalizability.

Cons:

- Computationally expensive.
- Sensitive to high leverage points.
- Negative bias (regression).

(James et al. 2013)

K-fold cross validation

K-fold CV

Full dataset

$$\begin{pmatrix} y_1 \\ \vdots \\ y_m \\ y_{m+1} \\ \vdots \\ y_n \end{pmatrix} = f\left(\begin{pmatrix} x_{1,1} & \dots & x_{1,p} \\ \vdots & & \vdots \\ x_{m,1} & \dots & x_{m,p} \\ x_{m+1,1} & \dots & x_{m+1,p} \\ \vdots & & \vdots \\ x_{n,1} & \dots & x_{n,p} \end{pmatrix}\right)$$

Test set

$$\begin{pmatrix} \hat{y}_1 \\ \vdots \\ \hat{y}_m \end{pmatrix} = \hat{f}_{train}\left(\begin{pmatrix} x_{1,1} & \dots & x_{1,p} \\ \vdots & & \vdots \\ x_{m,1} & \dots & x_{m,p} \end{pmatrix}\right)$$

Training set

$$\begin{pmatrix} \hat{y}_{m+1} \\ \vdots \\ \hat{y}_n \end{pmatrix} = \hat{f}_{train}\left(\begin{pmatrix} x_{m+1,1} & \dots & x_{m+1,p} \\ \vdots & & \vdots \\ x_{n,1} & \dots & x_{n,p} \end{pmatrix}\right)$$

Error:

regression: $e_1 = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$

classification: $e_1 = \sum_{i=1}^{m} I(y_i = \hat{y}_i)$

(James et al. 2013)

K-fold CV

Full dataset

$$\begin{pmatrix} y_1 \\ \vdots \\ y_m \\ y_{m+1} \\ \vdots \\ y_n \end{pmatrix} = f\left(\begin{pmatrix} x_{1,1} & \cdots & x_{1,p} \\ \vdots & & \vdots \\ x_{m,1} & \cdots & x_{m,p} \\ x_{m+1,1} & \cdots & x_{m+1,p} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{pmatrix}\right)$$

Test set

$$\begin{pmatrix} \hat{y}_{m+1} \\ \vdots \\ \hat{y}_n \end{pmatrix} = \hat{f}_{train}\left(\begin{pmatrix} x_{m+1,1} & \cdots & x_{m+1,p} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{pmatrix}\right)$$

Training set

$$\begin{pmatrix} \hat{y}_1 \\ \vdots \\ \hat{y}_m \end{pmatrix} = \hat{f}_{train}\left(\begin{pmatrix} x_{1,1} & \cdots & x_{1,p} \\ \vdots & & \vdots \\ x_{m,1} & \cdots & x_{m,p} \end{pmatrix}\right)$$

Error:

regression: $e_1 = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2, e_2 = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$

classification: $e_1 = \sum_{i=1}^{m} I(y_i = \hat{y}_i), e_2 = \sum_{i=1}^{m} I(y_i = \hat{y}_i)$

(James et al. 2013)

4-fold CV example

Full dataset

$$\begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = f\left(\begin{pmatrix} x_{1,1} & \cdots & x_{1,p} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{pmatrix}\right)$$

$\hat{f}_{train}(X_{train})$

$$e_1 = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$$

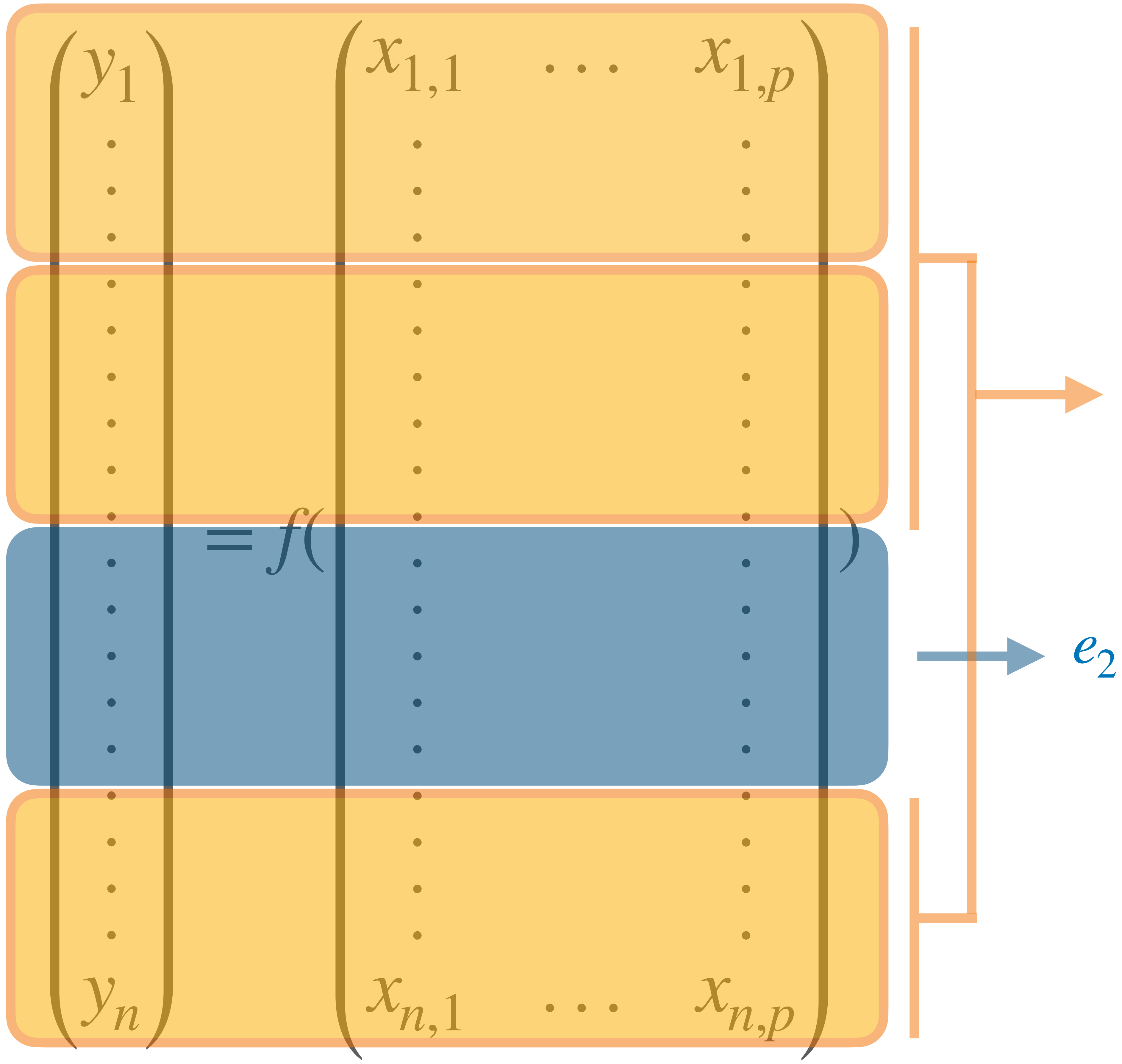
Error vector

$$(e_1, \qquad)$$

(James et al. 2013)

4-fold CV example

Full dataset



$\hat{f}_{train}(X_{train})$

$e_2 = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$

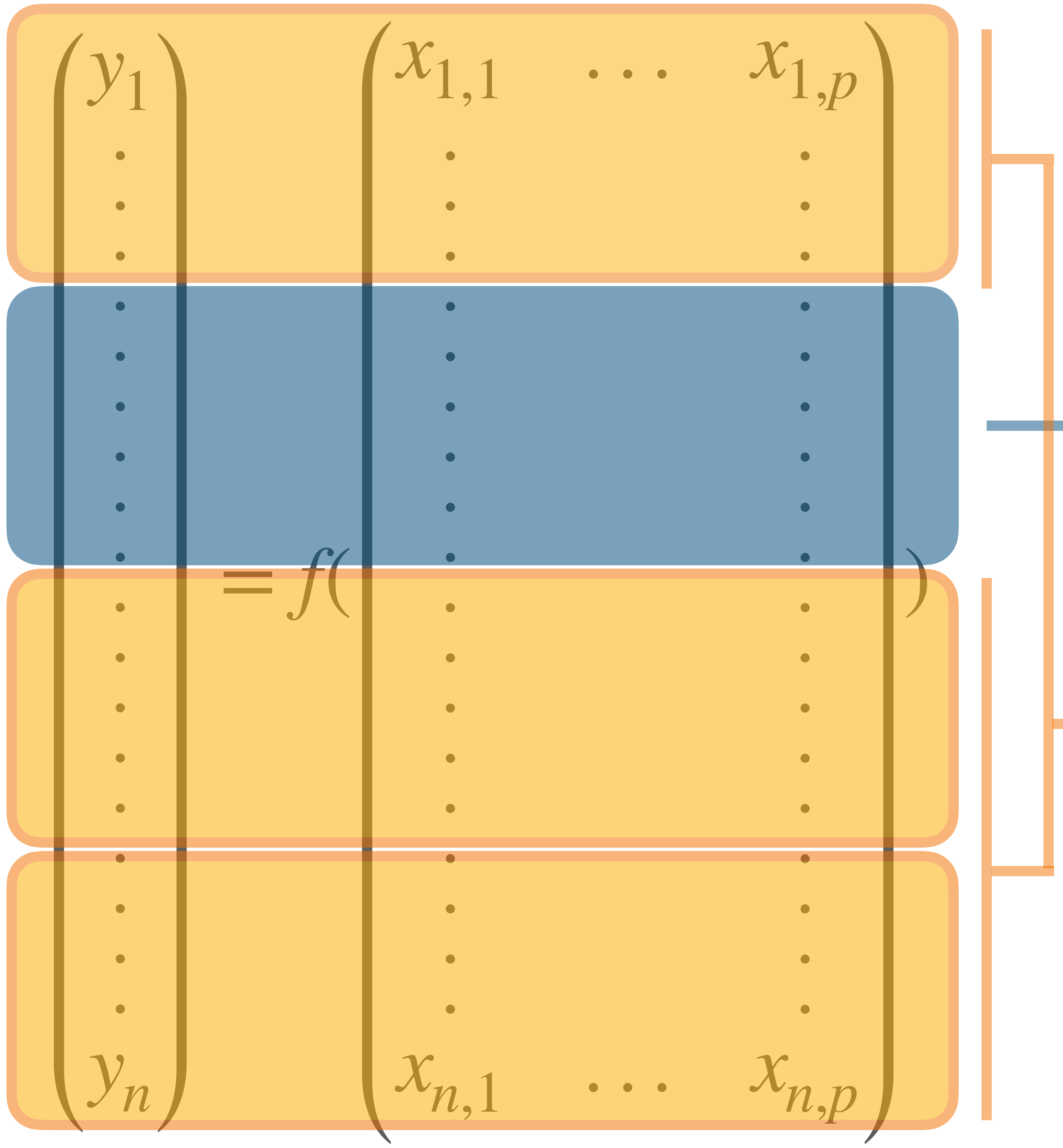
Error vector

$(e_1, \quad e_2, \qquad)$

(James et al. 2013)

4-fold CV example

<u>Full dataset</u>



$$e_3 = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$$

$$\hat{f}_{train}(X_{train})$$

<u>Error vector</u>

$$(e_1, \quad e_2, \quad e_3, \quad)$$

(James et al. 2013)

4-fold CV example

Full dataset

$$\begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = f\left(\begin{pmatrix} x_{1,1} & \cdots & x_{1,p} \\ \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,p} \end{pmatrix}\right)$$

$$\rightarrow e_4 = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$$

$$\rightarrow \hat{f}_{train}(X_{train})$$

Error vector

$$(e_1, \quad e_2, \quad e_3, \quad e_4)$$

CV error:

$$CV_k = \sum_{i=1}^{k} e_i = \sum_{i=1}^{k} \sum_{j=1}^{m} (y_j - \hat{y}_j)^2$$

(James et al. 2013)

K-fold CV algorithm

<u>Step 1</u>: Take unique subsample of m observations, where $m = \frac{n}{k}$. Set aside as a test set.

<u>Step 2</u>: Make new X_{train} with all $\neg m$ observations.

<u>Step 3</u>: Fit $\hat{f}_{train}(X_{train})$.

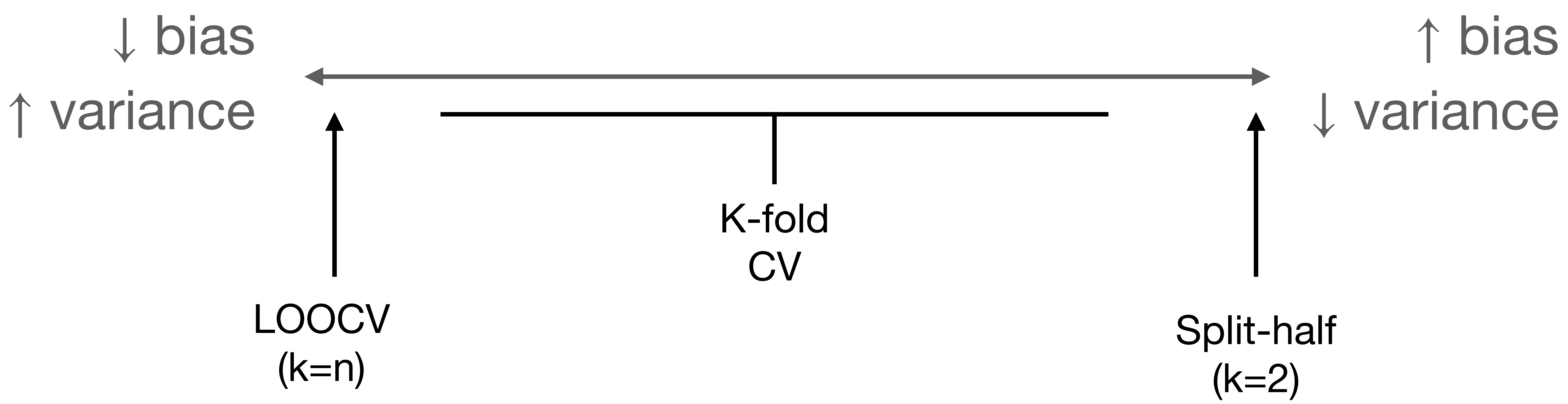
<u>Step 4</u>: Evaluate test error on y_i using $\hat{f}_{train}$ from Step 3.

<u>Step 5</u>: Repeat steps 1 - 4 for all k folds.

<u>Step 6</u>: Evaluate full set performance $CV_k = \sum_{i=1}^{k} \sum_{j=1}^{m} (y_j - \hat{y}_j)^2$.

(James et al. 2013)

Bias-variance tradeoff



(James et al. 2013)

Things to consider

- Choose k to have enough power in the training set to fit a reliable $\hat{f}(X)$.
- Have enough folds (k) to reduce model variance.
- Balance your subsampling to watch for high-leverage points.
- Watch the dimensionality in the training set.

(James et al. 2013)

Take home message

- Cross validation allows for you to directly test the generalizability of your effects without having to collect additional data.