Overfitting and Cross validation Introduction

Model performance

How good is the regression model?

Model performance

How good is the regression model?

- *How well the model fits the data?*
- *How well the model predicts the data?*

Model performance

How good is the regression model?

- *How well the model fits the data?* SSE R^2
- *How well the model predicts new data?*

Model performance

How good is the regression model?

- *How well the model fits the data?* SSE R^2
- *How well the model predicts new data?* MSPE

Overfitting

Regression assumption:

Expected values of Y follow a regression function

Best model:

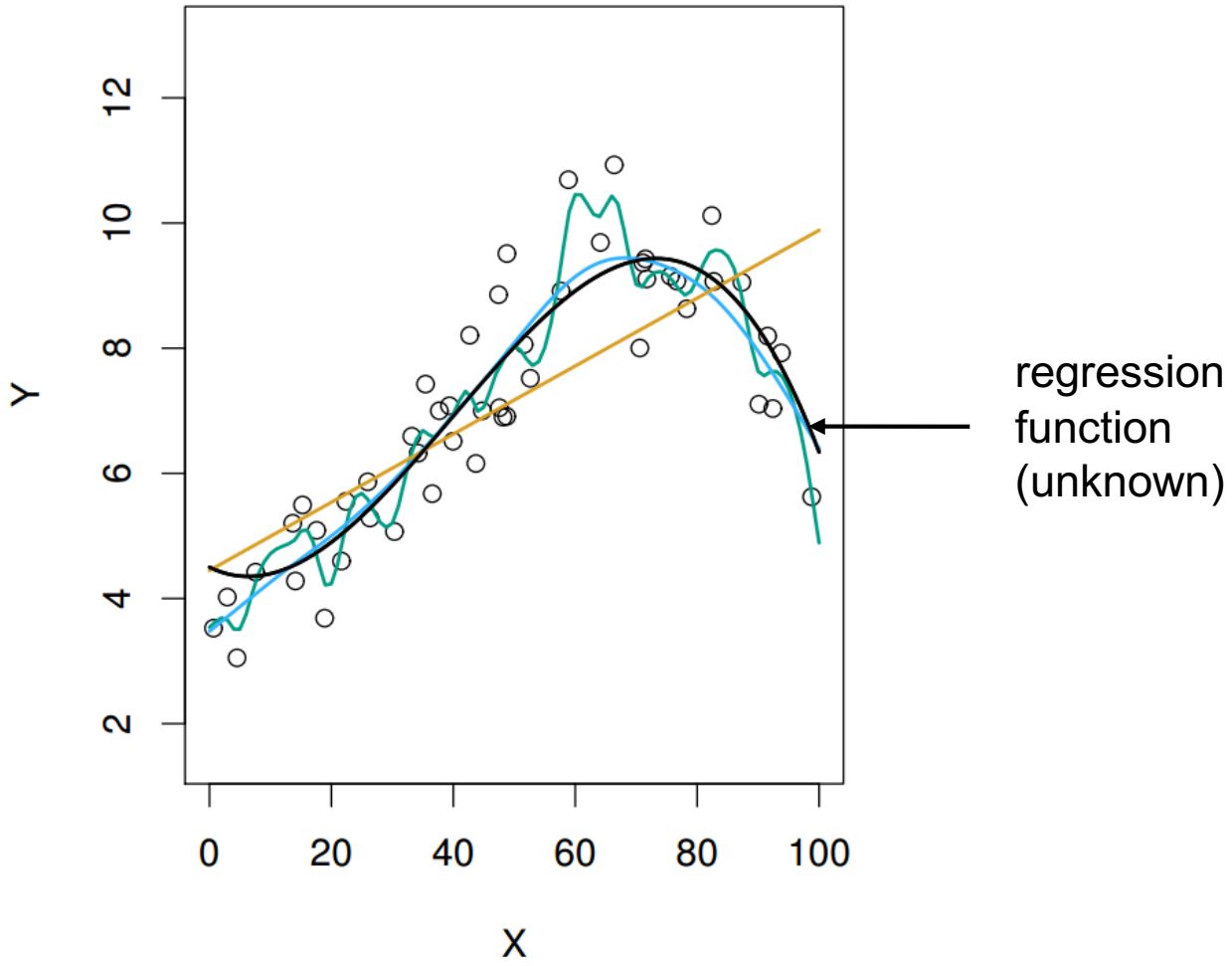
Closest model to the (unknown) regression function

Overfitting:

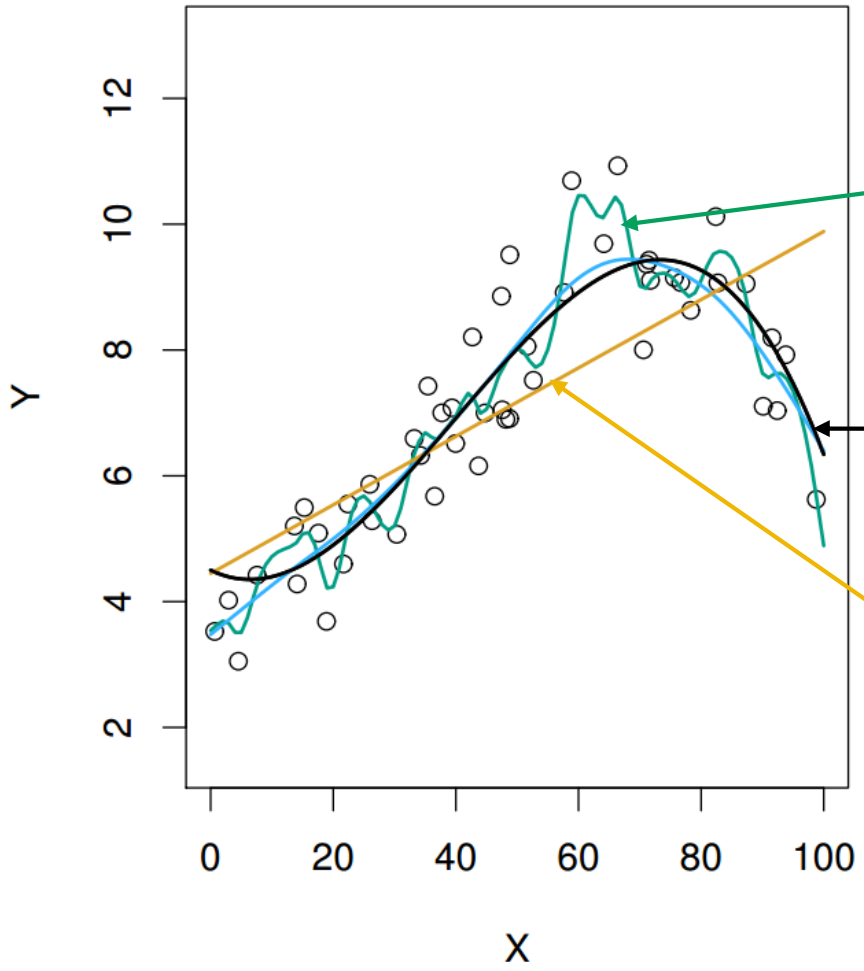
Model too close to data points

but far from the regression function

Overfitting - Example



Overfitting - Example



high variance model (overfitting)

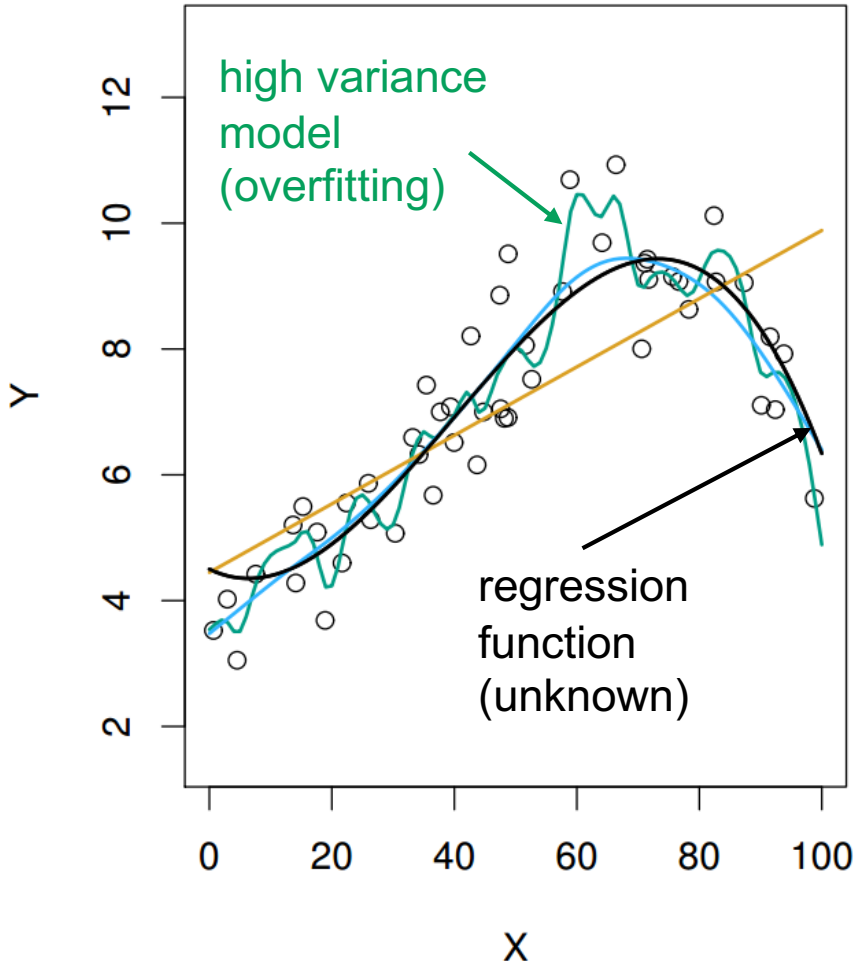
regression function (unknown)

biased model (underfitting)

Overfitting

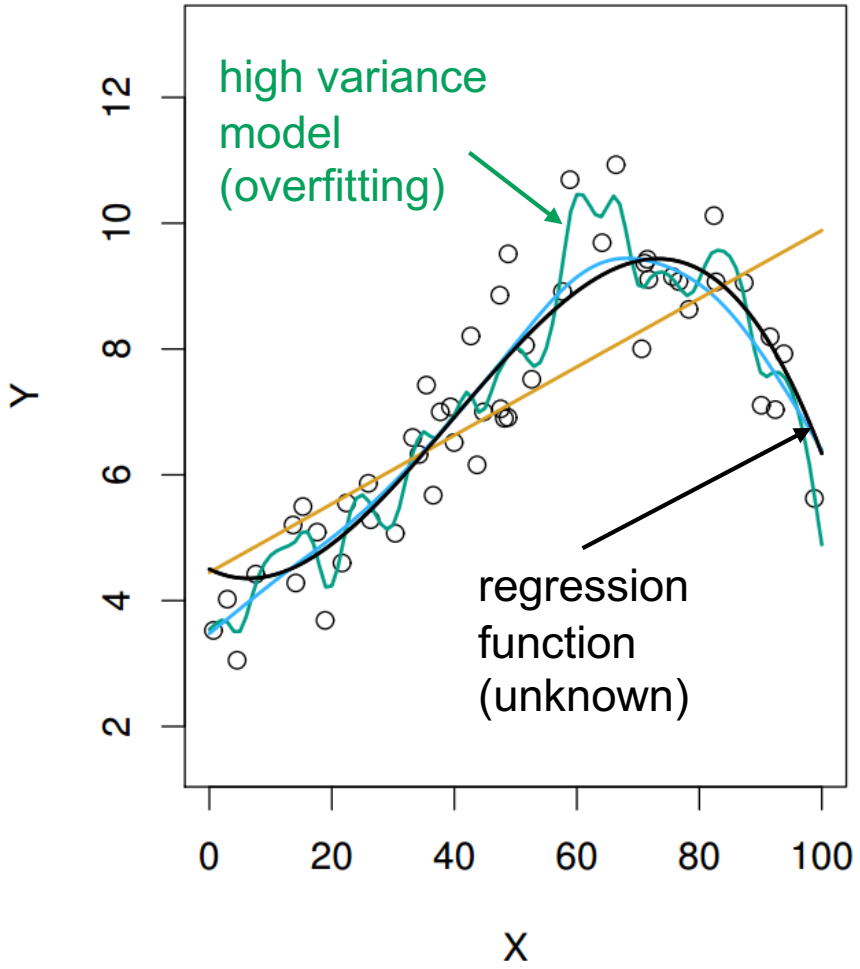
What is overfitting?

- *A model that follows the data points too closely*
- *It does not follow the regression function*



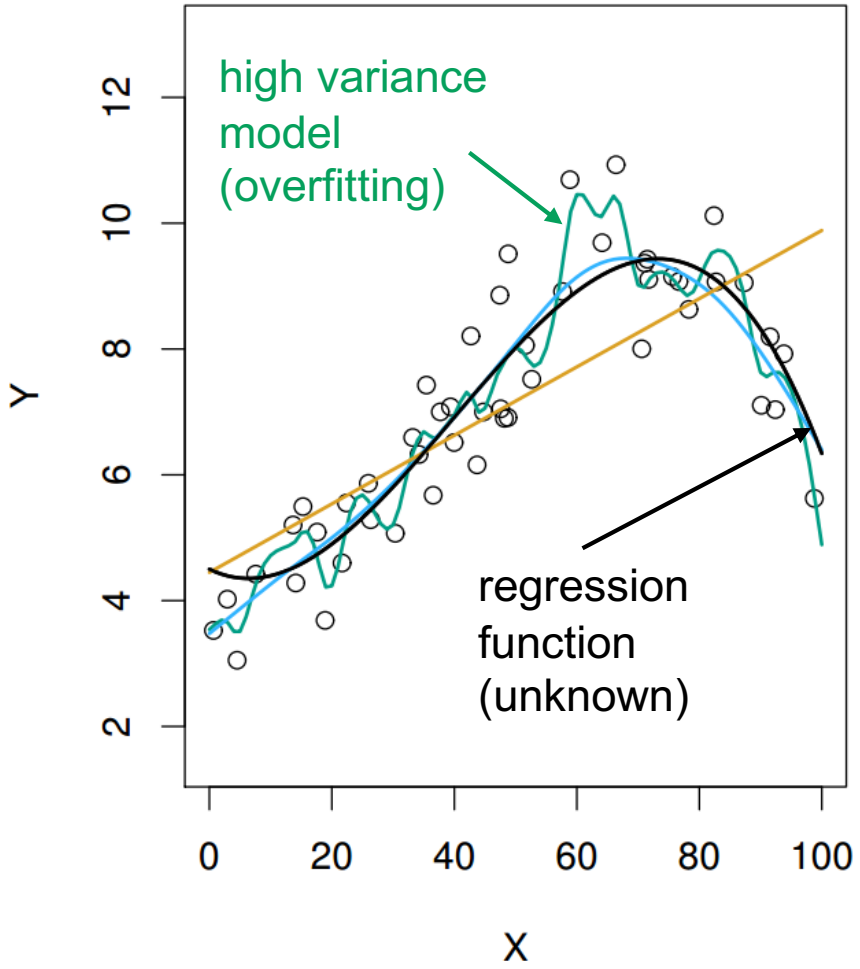
Overfitting

- *How to identify overfitting?*
- *How to avoid overfitting?*
- *Cross-validation, Ridge regression*



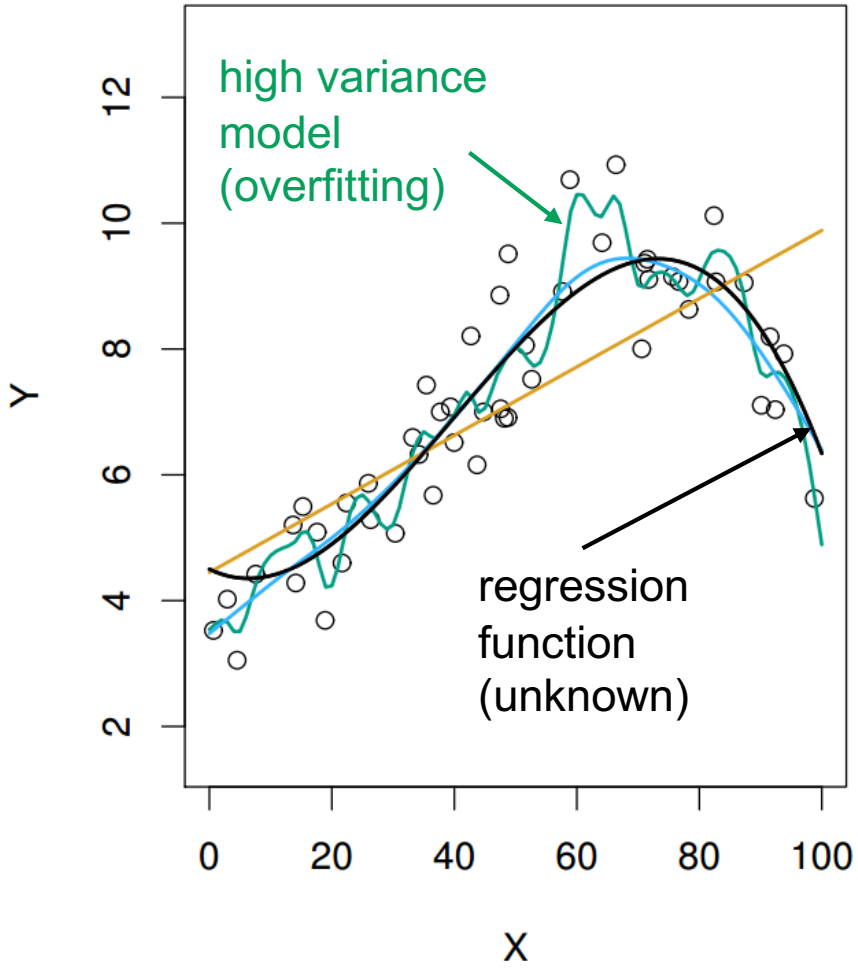
CROSS VALIDATION

- *Reserve part of the data to test the model (MSPE)*
- *Use the remaining data to build the model*
- *If the model fits the data (large R-square) but cannot predict well the test data (small MSPE), it is overfitting*



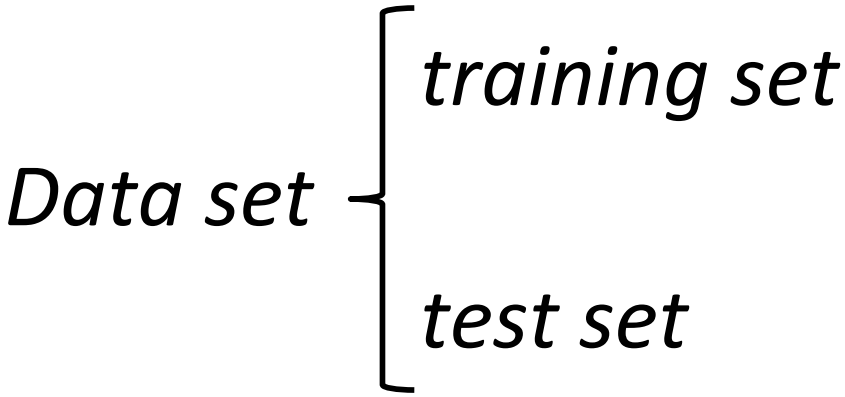
Cross Validation - Types

- *Holdout CV (validation Set approach)*
- *K-fold cross validation*
- *Leave-one-out cross validation (LOOCV)*



Holdout Cross validation

Holdout Cross Validation

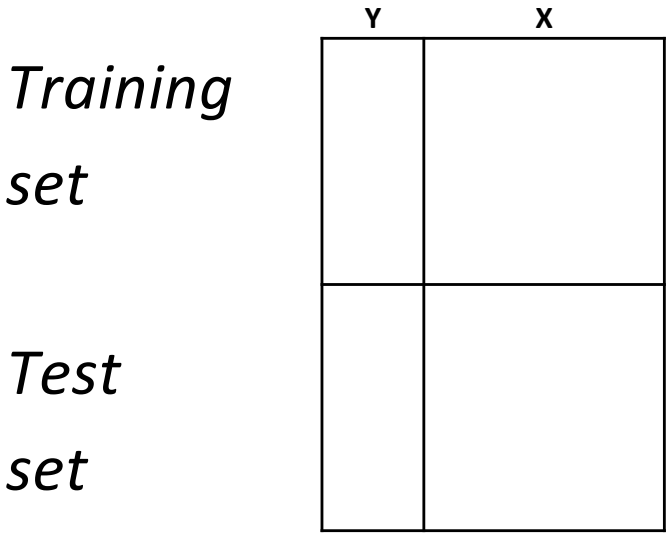


Holdout Cross Validation

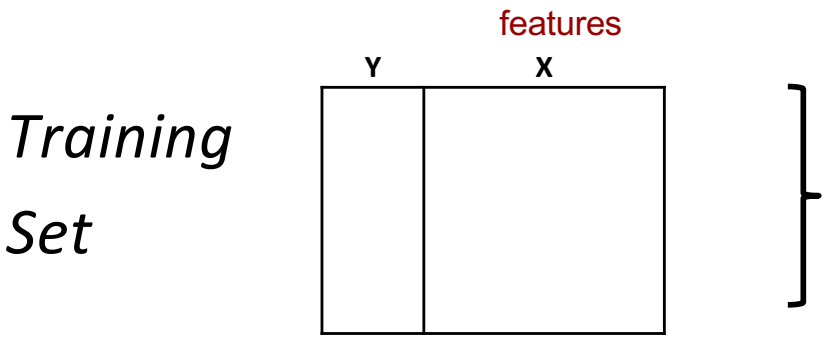
Data set → *training set* (*to build the model*)

Data set → *test set* (*to test model*)

Holdout Cross Validation

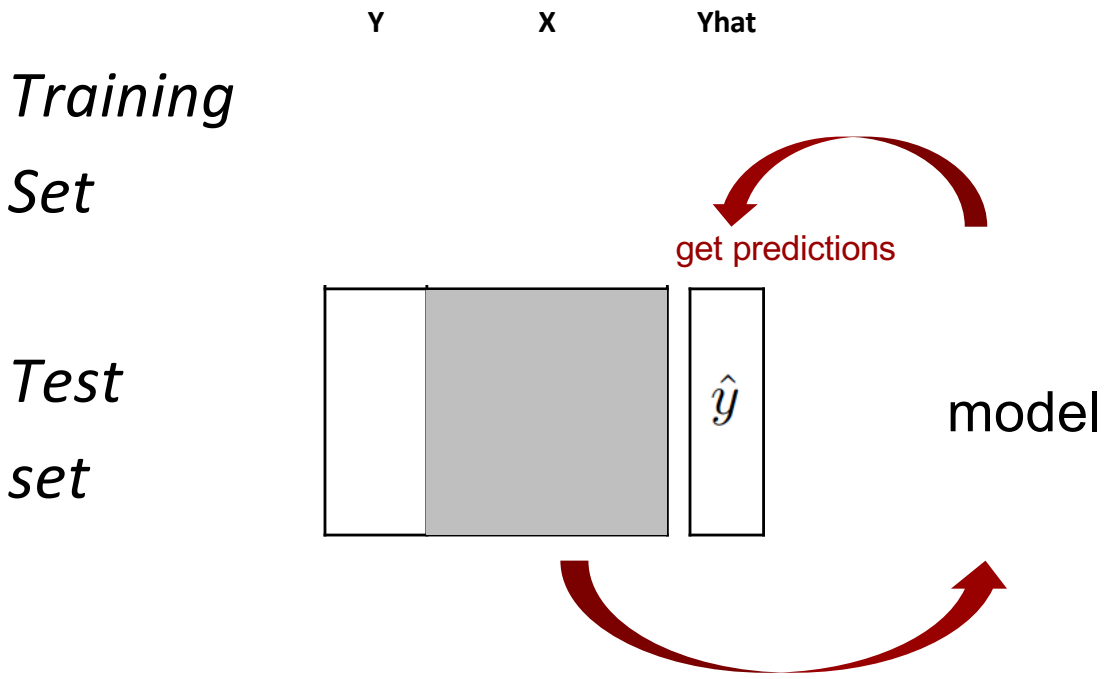


Holdout Cross Validation



- feature selection
- feature engineering
- R-squared
- Adj R-squared, AIC

Holdout Cross Validation

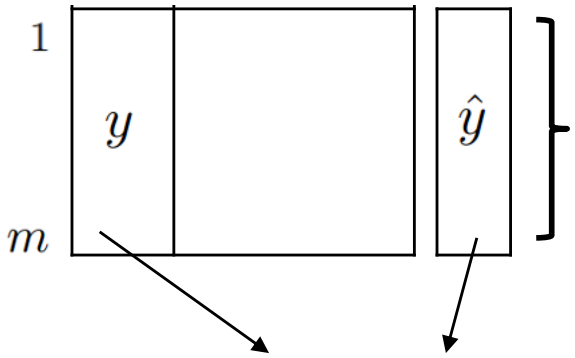


Holdout Cross Validation

Y X Yhat

Training Set

Test set

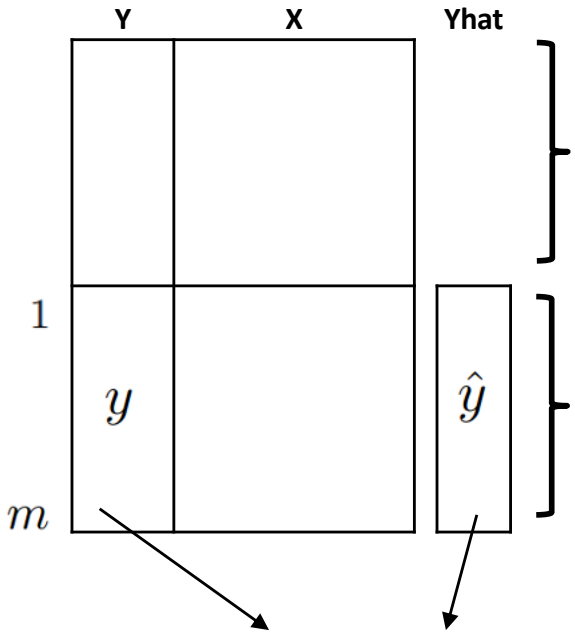


test model

Average of squared residuals on the test set

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

Holdout Cross Validation



build model

train MSE

test model

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

= test MSE

Prediction performance

- Compare models based on MSPE
- Model with the smallest MSPE
  is the best for prediction

k-Fold Cross validation

Holdout Cross Validation

dataset

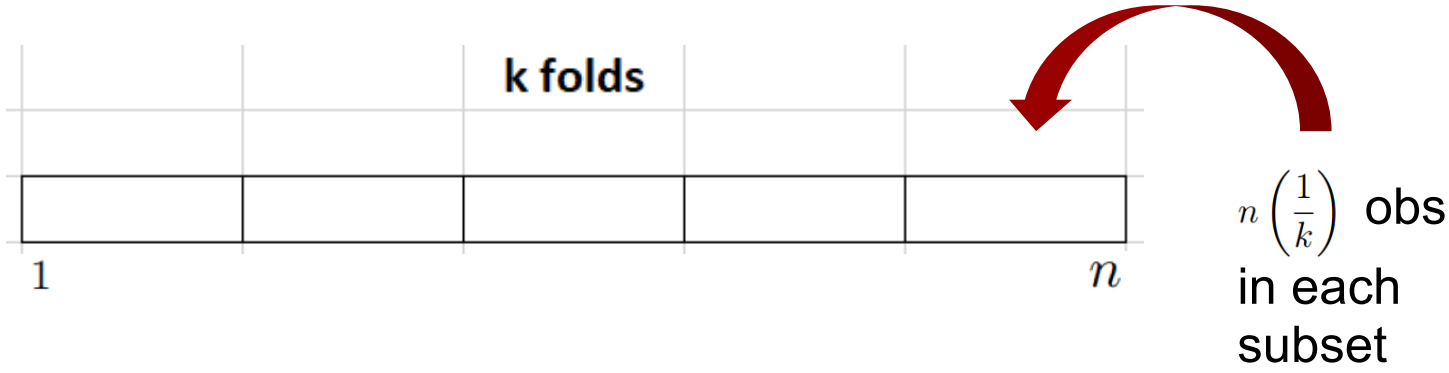
MSPE

k-Fold Cross Validation

dataset

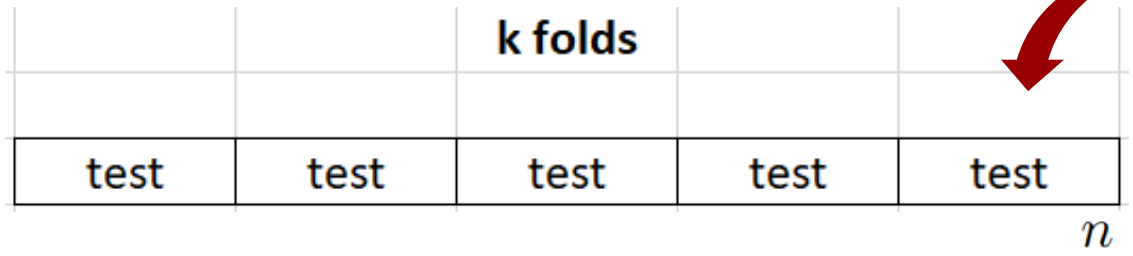
k-Fold Cross Validation

dataset



k-Fold Cross Validation

dataset

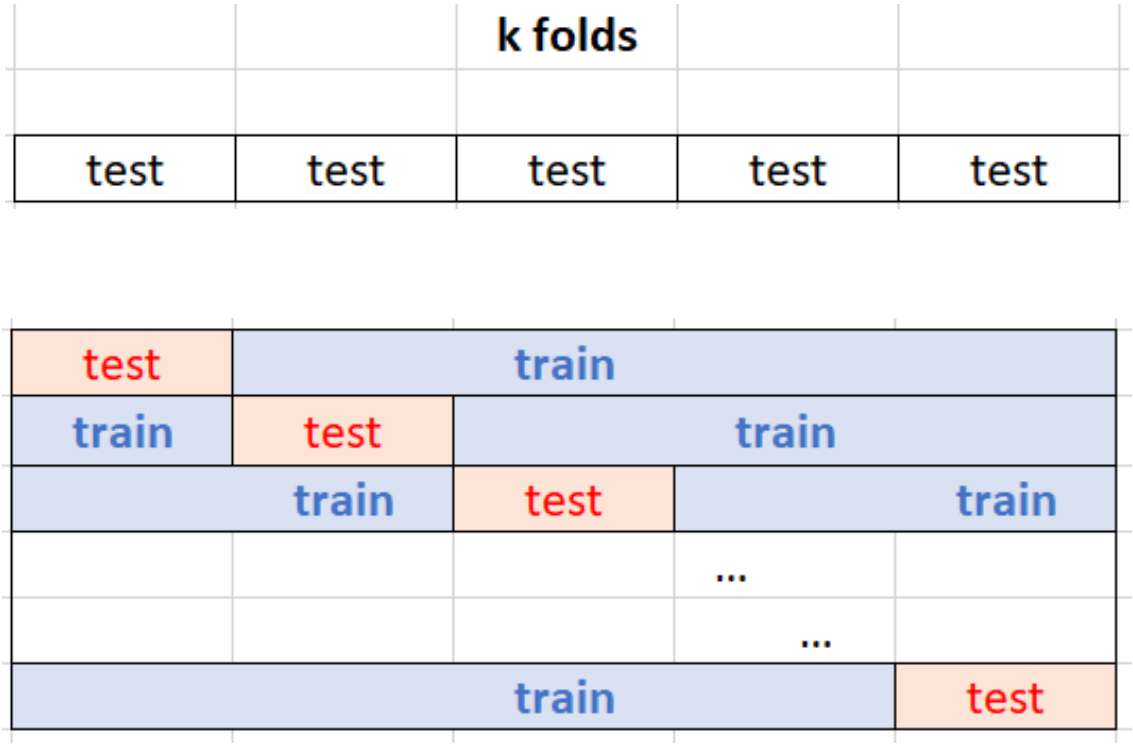


$n\left(\frac{1}{k}\right)$ obs in each subset

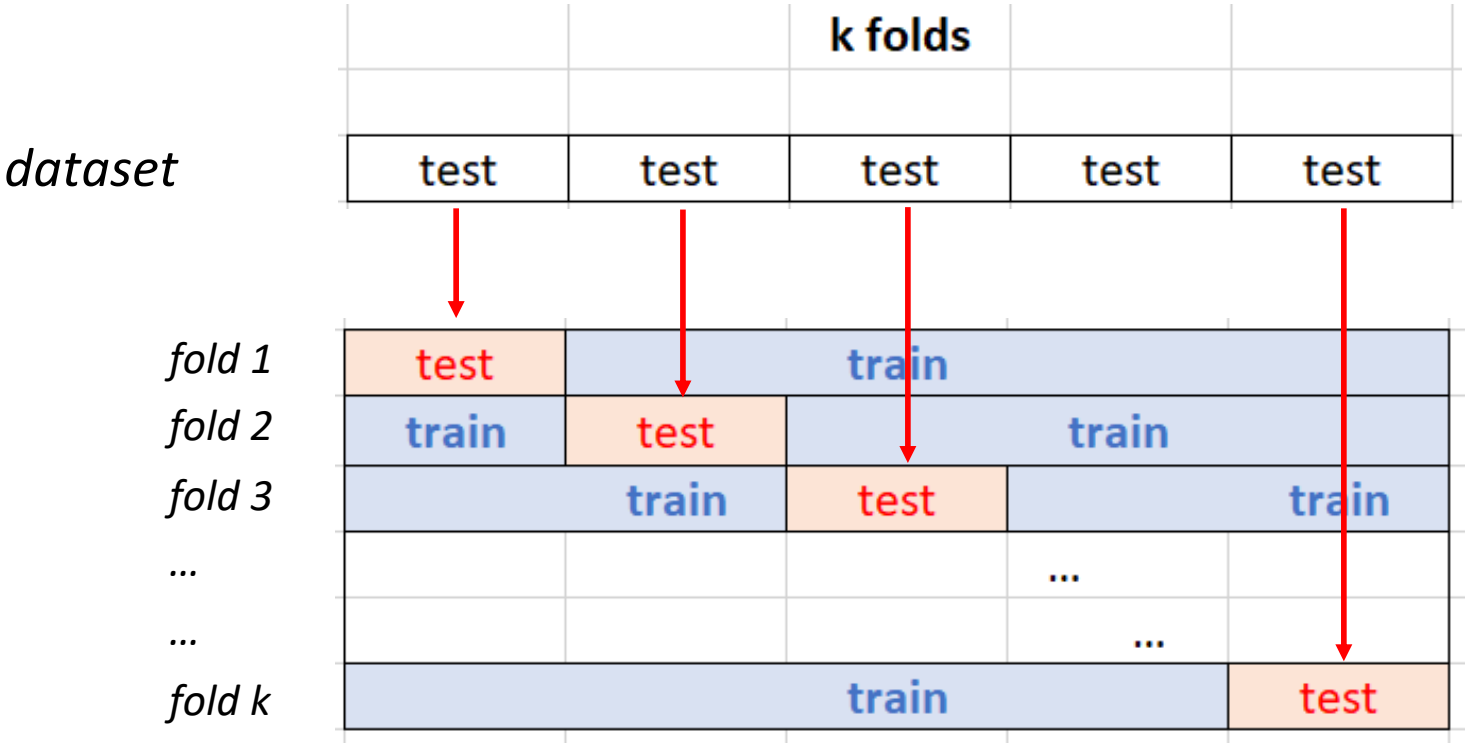
Eventually each subset becomes a test set

k-Fold Cross Validation

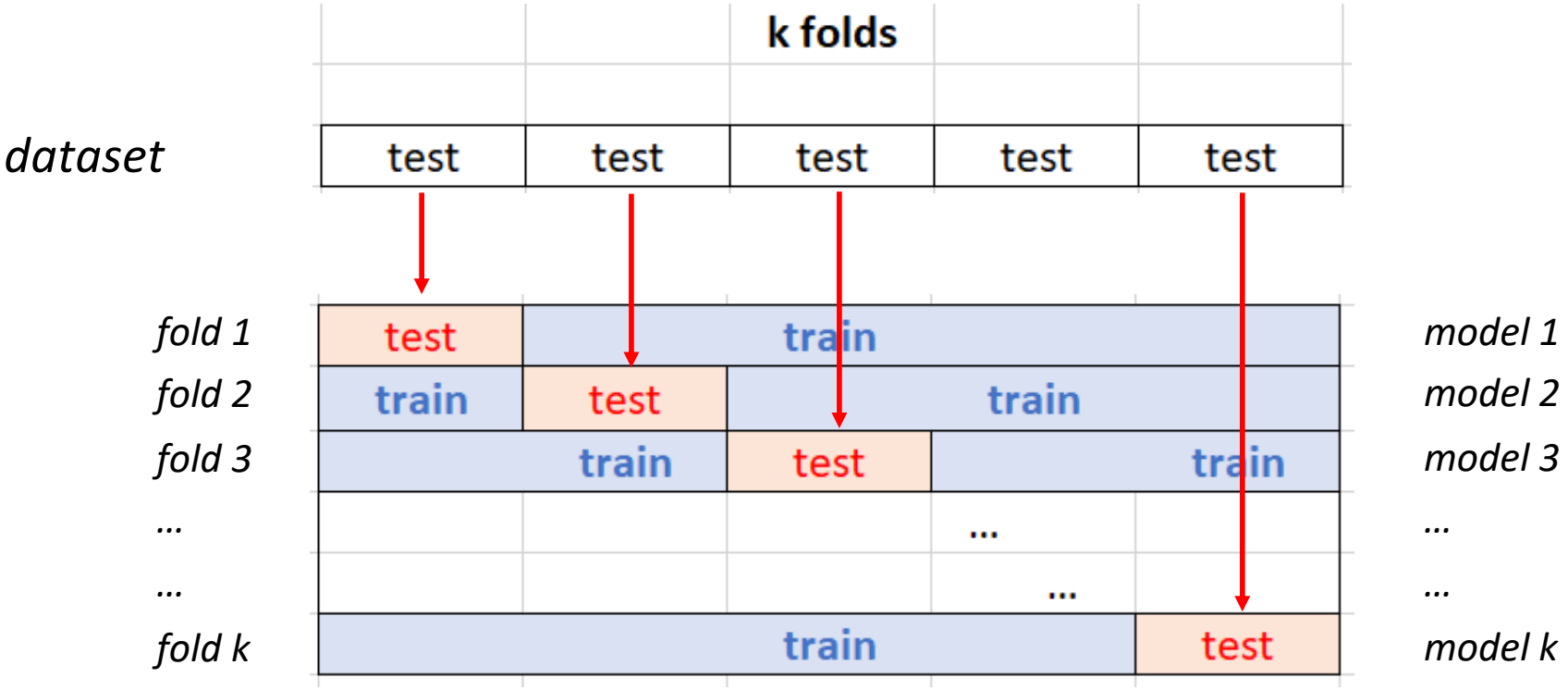
dataset



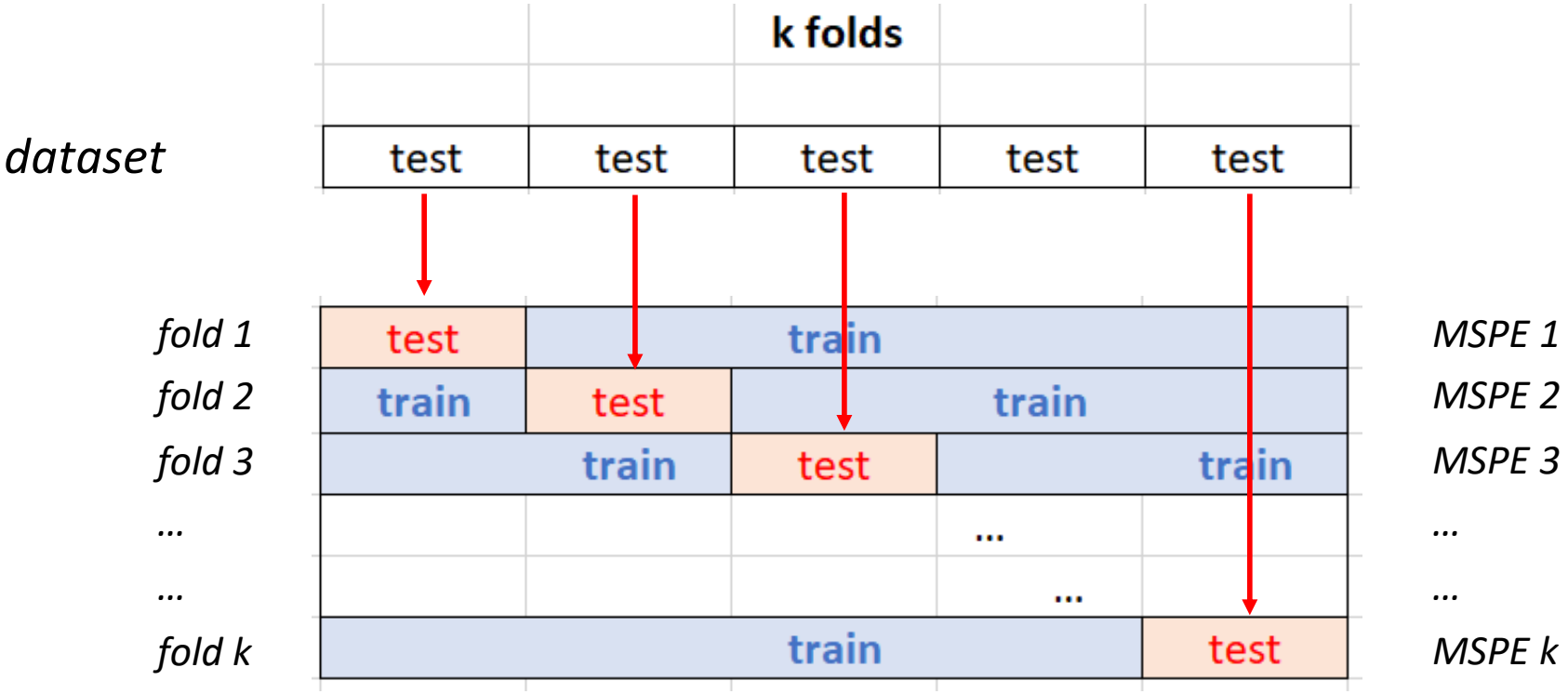
k-Fold Cross Validation



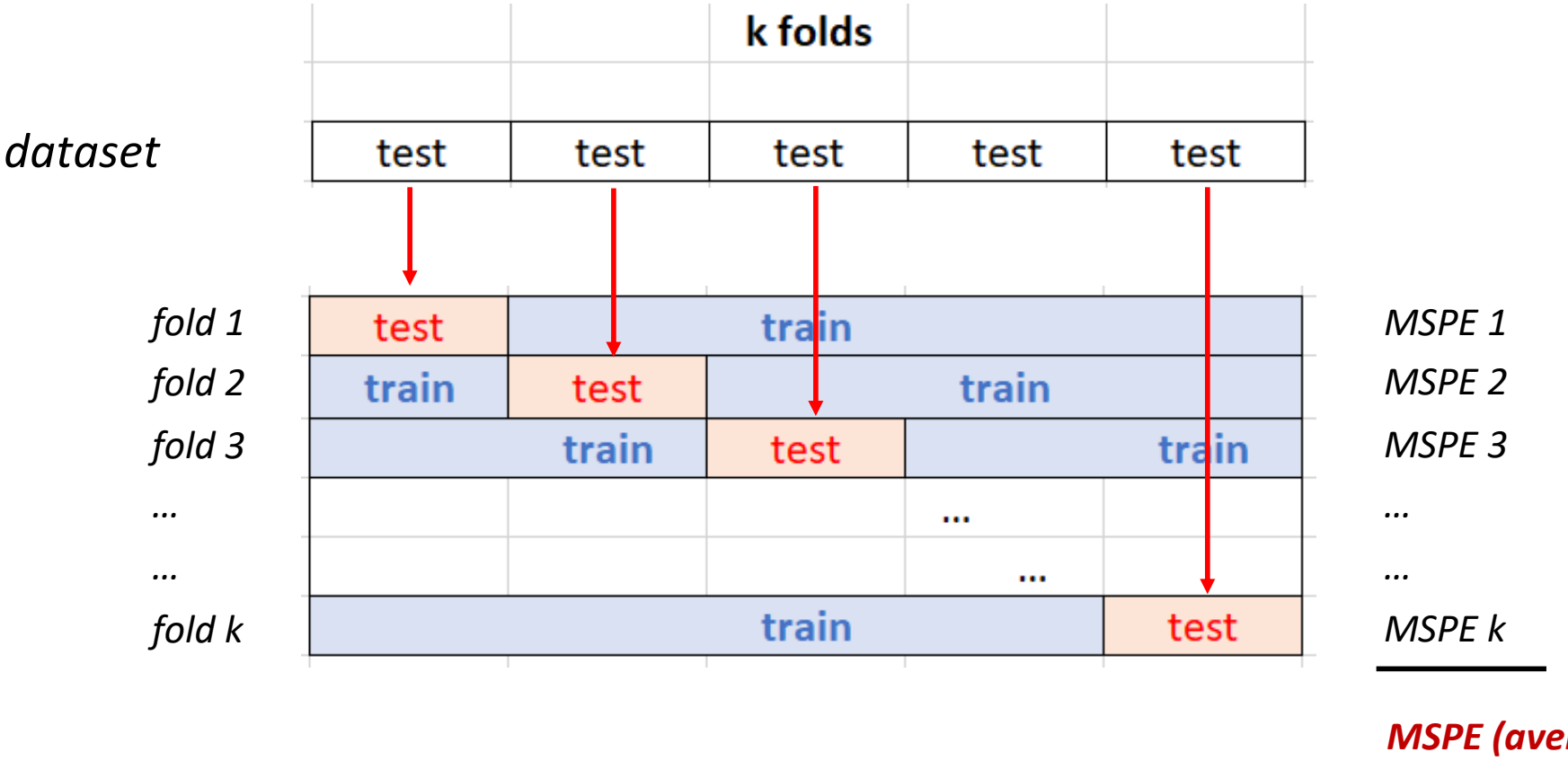
k-Fold Cross Validation



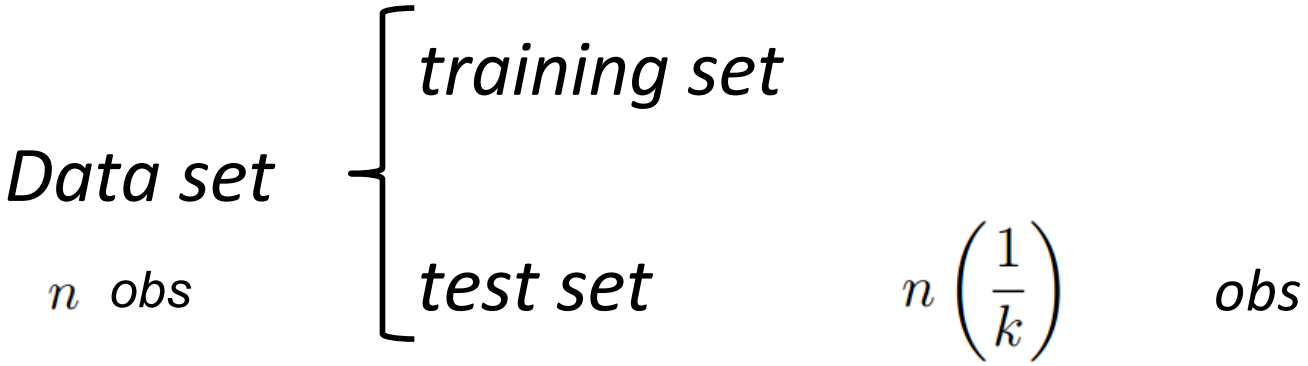
k-Fold Cross Validation



k-Fold Cross Validation



k-Fold Cross Validation



k-Fold Cross Validation

Data set

n *obs*

- *training set* $n\left(1-\frac{1}{k}\right)$ *obs*
- *test set* $n\left(\frac{1}{k}\right)$ *obs*

k-Fold Cross Validation

k=5 folds

Data set

- *training set* $n\left(1-\frac{1}{k}\right)$ *obs*
- *test set* $n\left(\frac{1}{k}\right)$ *obs*

k-Fold Cross Validation

k=5 folds

Data set

- *training set* $n\left(1-\frac{1}{k}\right)$ *obs* — *.80 n*
- *test set* $n\left(\frac{1}{k}\right)$ *obs* — *.20 n*

k-Fold Cross Validation

k=5 folds

Data set

- *training set* $n\left(1-\frac{1}{k}\right)$ *obs* — *80%*
- *test set* $n\left(\frac{1}{k}\right)$ *obs* — *20%*

Leave-one-out Cross Validation (LOOCV)

k=n folds

Data set { *training set*, *test set*

$$n\left(\frac{1}{k}\right) = 1 \quad \textit{observation}$$

Leave-one-out Cross Validation (LOOCV)

k=n folds

Data set

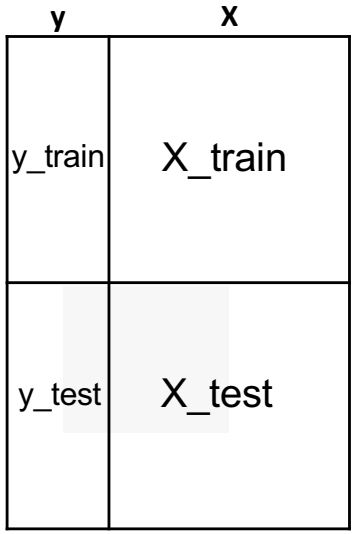
- *training set* $(n-1)$ *observations*
- *test set* 1 *observation*

LOOCV is a K-Fold Cross validation when k = n

Holdout Cross Validation - sklearn

HOLDOUT Cross Validation

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split( X, y,
                                   test_size=0.5,
                                   random_state=1)
```



Holdout Cross Validation - sklearn

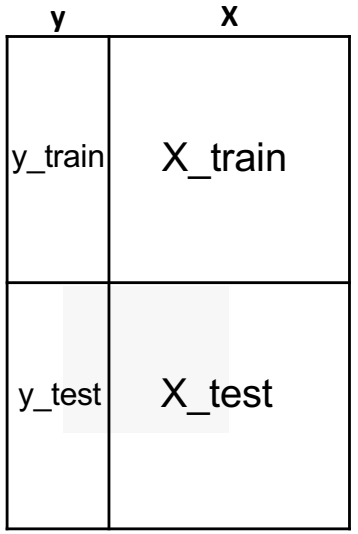
HOLDOUT Cross Validation

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split( X, y,
                                                     test_size=0.5,
                                                     random_state=1)
```

```
m2 = LinearRegression().fit( X_train, y_train )
yhat2 = m2.predict( X_test )
```

```
# mspe
res2 = (yhat2 -     y_test) **2
mspe2 = np.mean(res2)
mspe2
```

```
20.005851783316732
```



K-Fold Cross Validation - sklearn

```
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import KFold
```

```
measure = 'neg_mean_squared_error'
```

linear model

```
mspe1 = cross_val_score(LinearRegression(), X, y,
                        cv = KFold(n_splits = 5),
                        scoring = measure)
```

```
cvmspe1 = mspe1.mean()
-cvmspe1
```

mspe1 is an array with
5 mspe values,
one from each fold

MSPE