Guided Tour of Machine Learning in Finance

Overfitting and model capacity

Igor Halperin

NYU Tandon School of Engineering, 2017

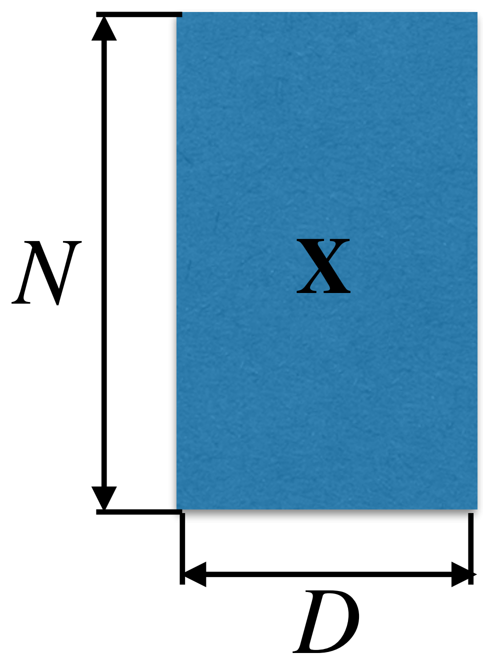
Generalization error in regression

$$\mathbb{E}\left[\left(y-\hat{f}(\boldsymbol{x})\right)^2\right]=(bias)^2+variance+noise$$

- A good measure of generalization error for regression is an expected squared loss
- The expectation is taken over all data, both seen and unseen.
- The bias-variance decomposition shows a general structure of the generalization error

Training set and test set

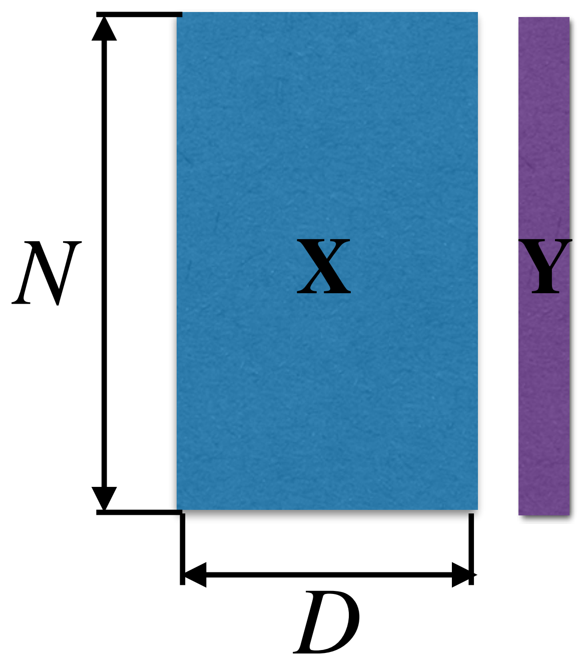
Design matrix (dimension $N \times D$):



I.i.d. samples from a data-generating distribution p_{data}

Training set and test set

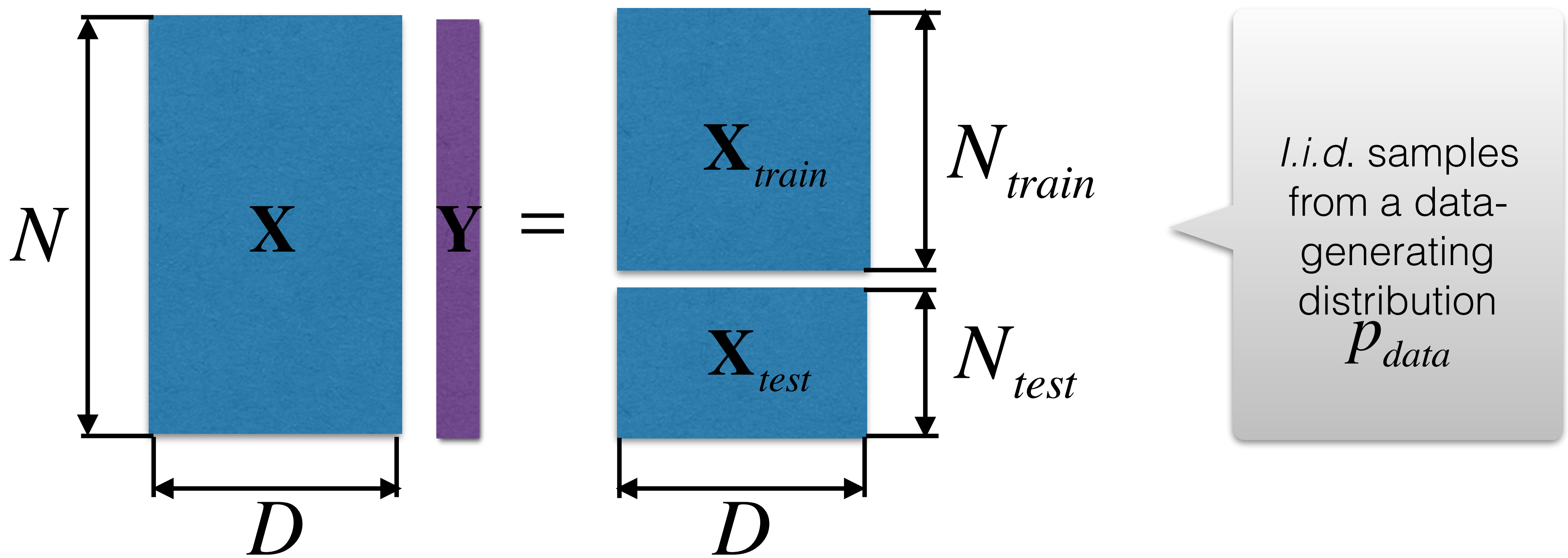
Design matrix (dimension $N \times D$):



I.i.d. samples from a data-generating distribution p_{data}

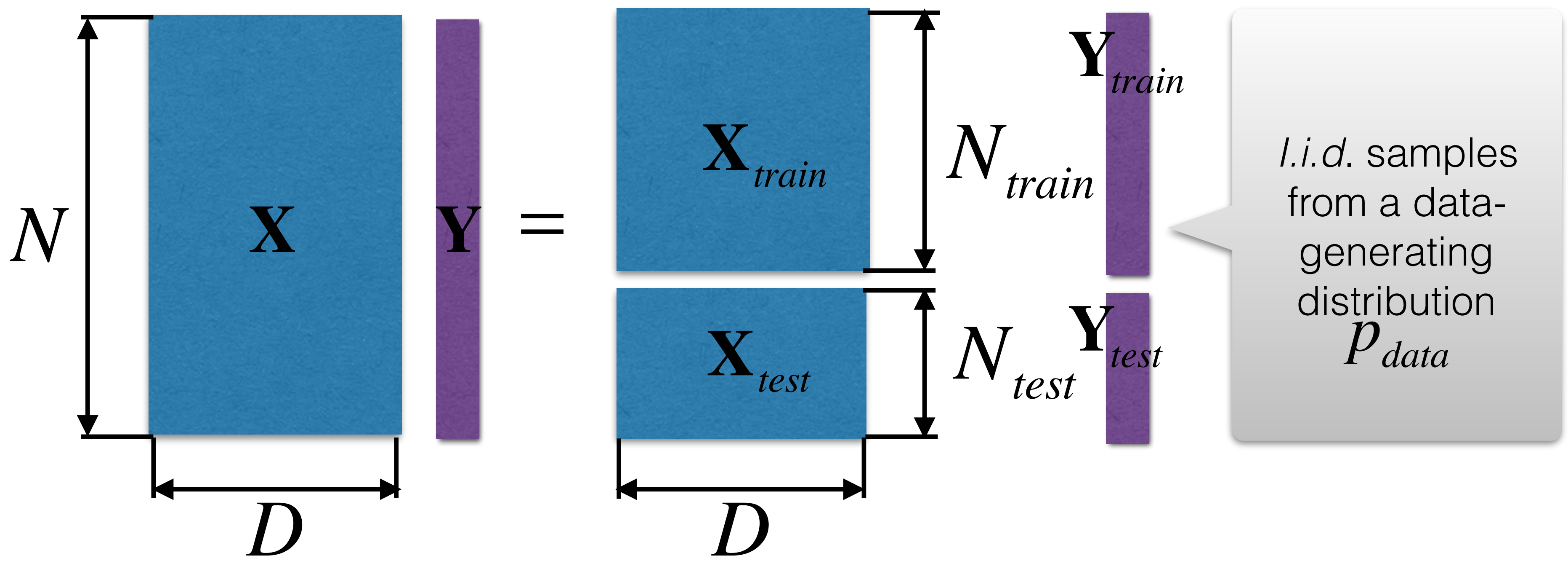
Training set and test set

Design matrix (dimension $N \times D$):



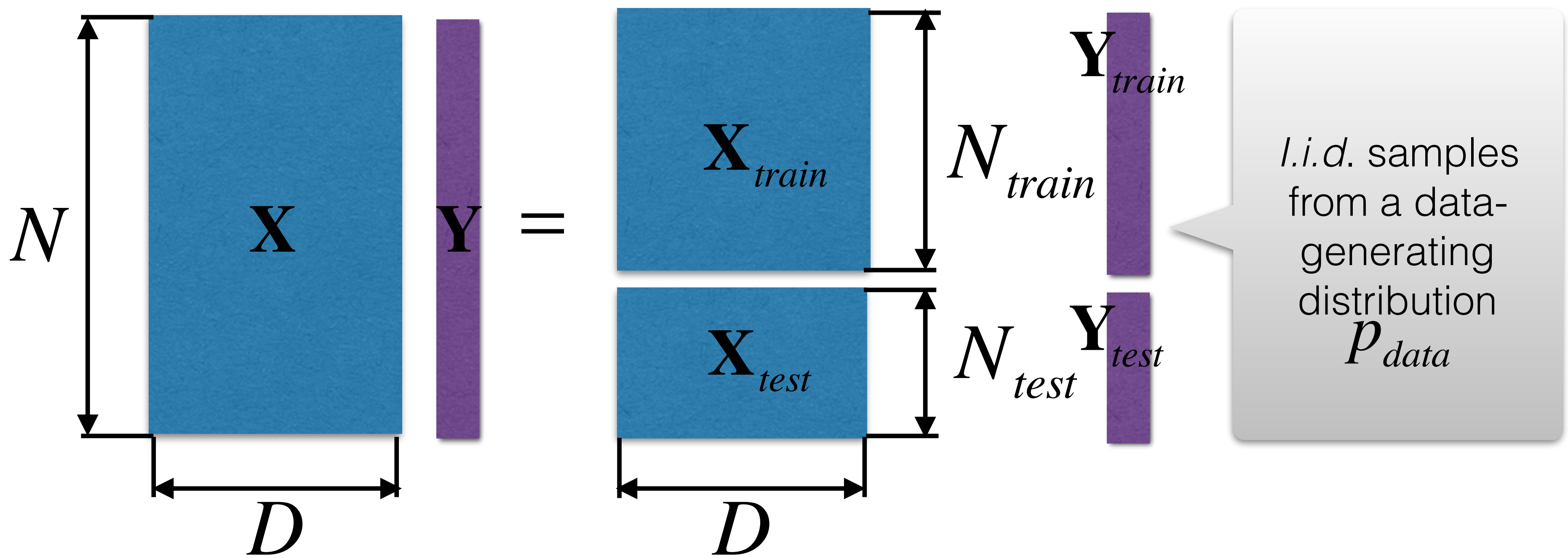
Training set and test set

Design matrix (dimension $N \times D$):



Training set and test set

Design matrix (dimension $N \times D$):

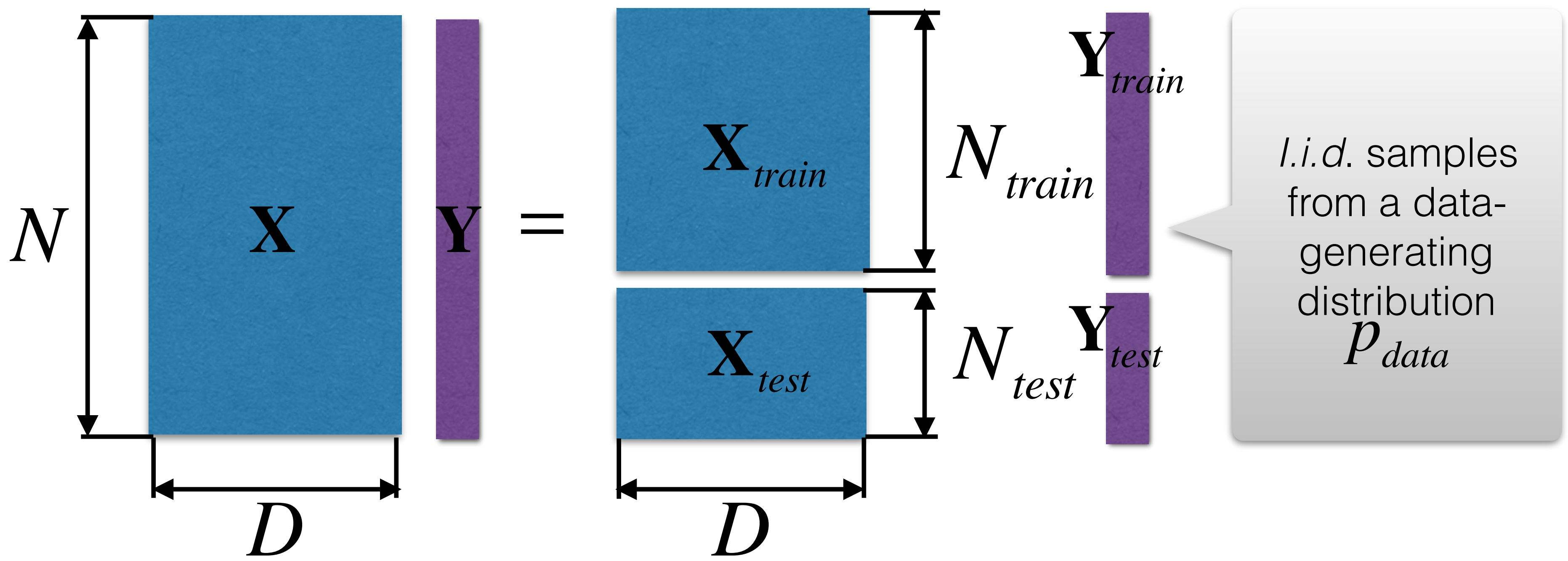


A model is trained using only a **training set** $(\mathbf{X}_{train}, y_{train}) \sim p_{data}$

A **test set** $(\mathbf{X}_{test}, y_{test}) \sim p_{data}$ is used to estimate algorithm's ability to **generalize**, i.e. perform well on unseen data.

Training set and test set

Design matrix (dimension $N \times D$):



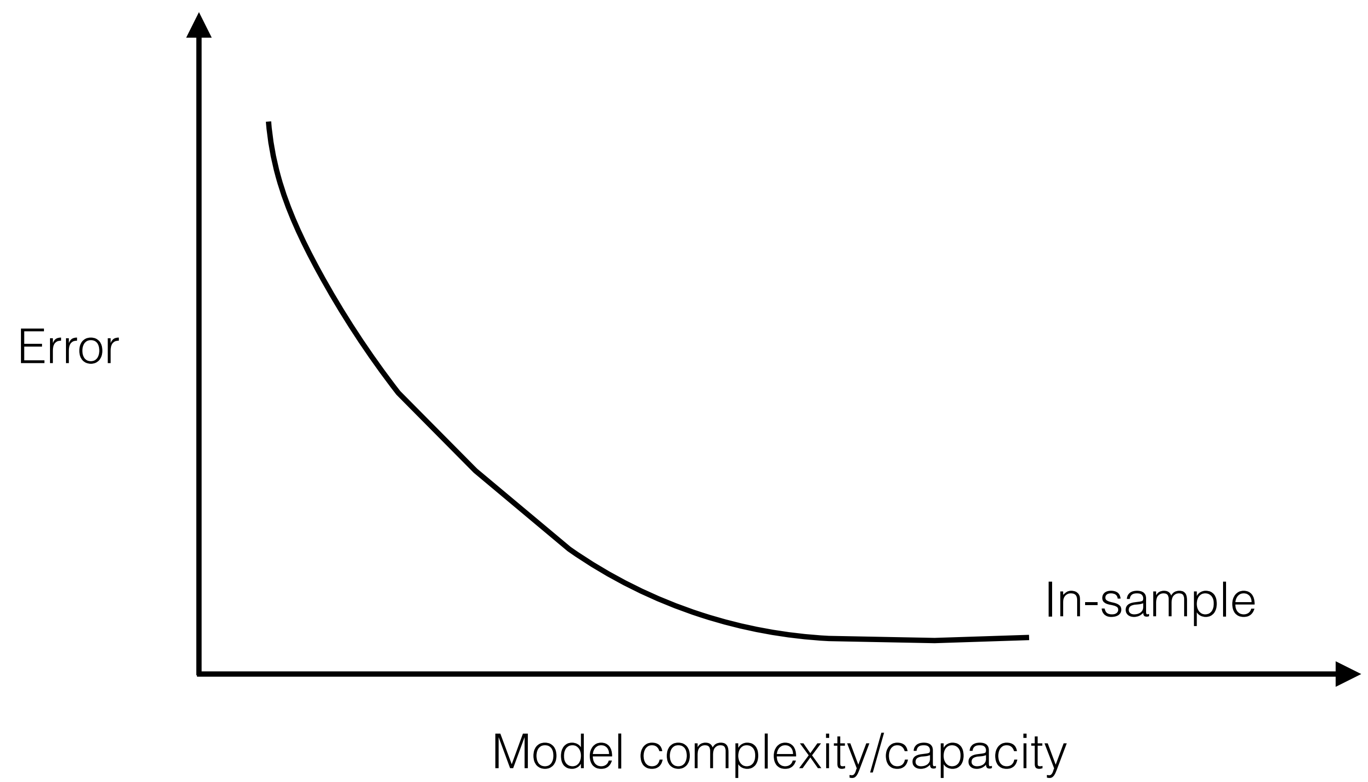
A model is trained using only a **training set** $(\mathbf{X}_{train}, y_{train}) \sim p_{data}$

A **test set** $(\mathbf{X}_{test}, y_{test}) \sim p_{data}$ is used to estimate algorithm's ability to **generalize**, i.e. perform well on unseen data.

More specifically, a test set is used to detect when a ML algorithm starts to **overfit** data, by estimating a generalization error by a test error.

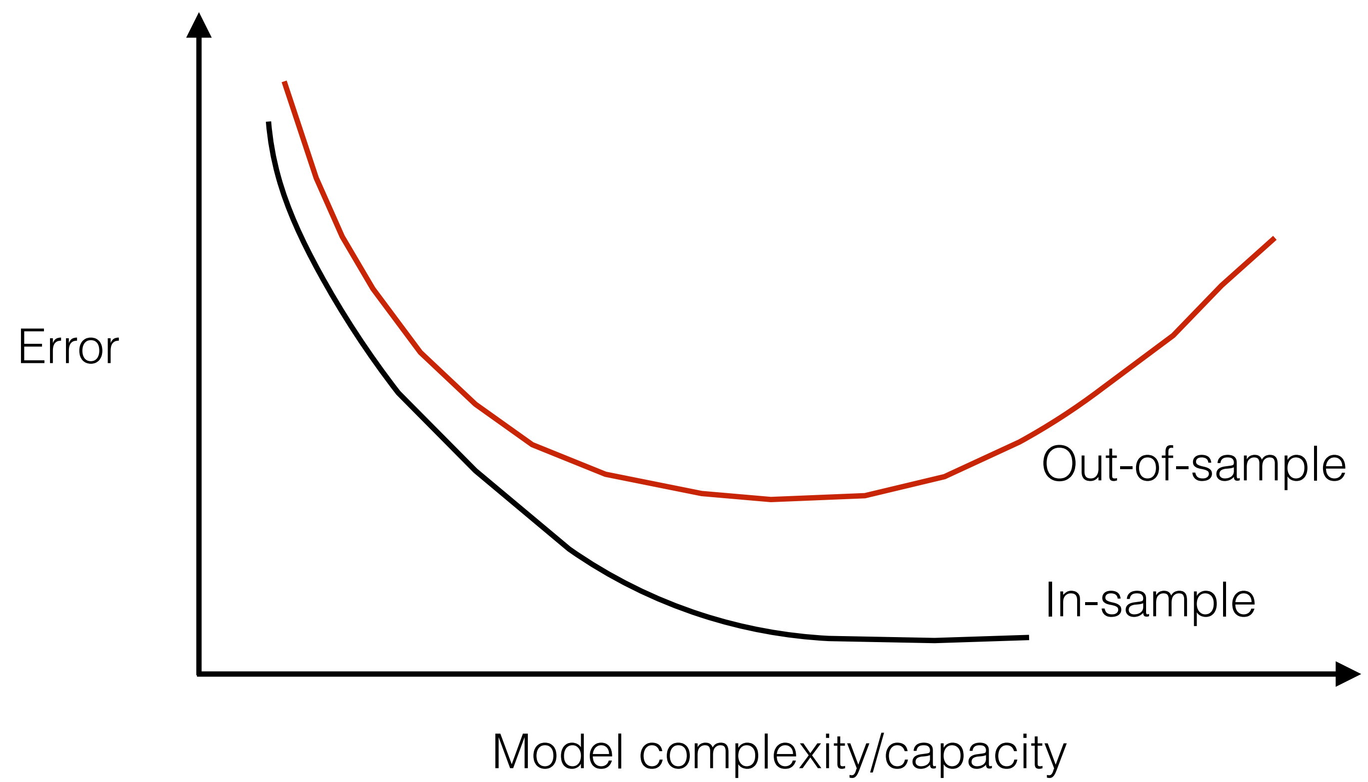
Overfitting

Trying to exactly match all available data is almost always a bad idea



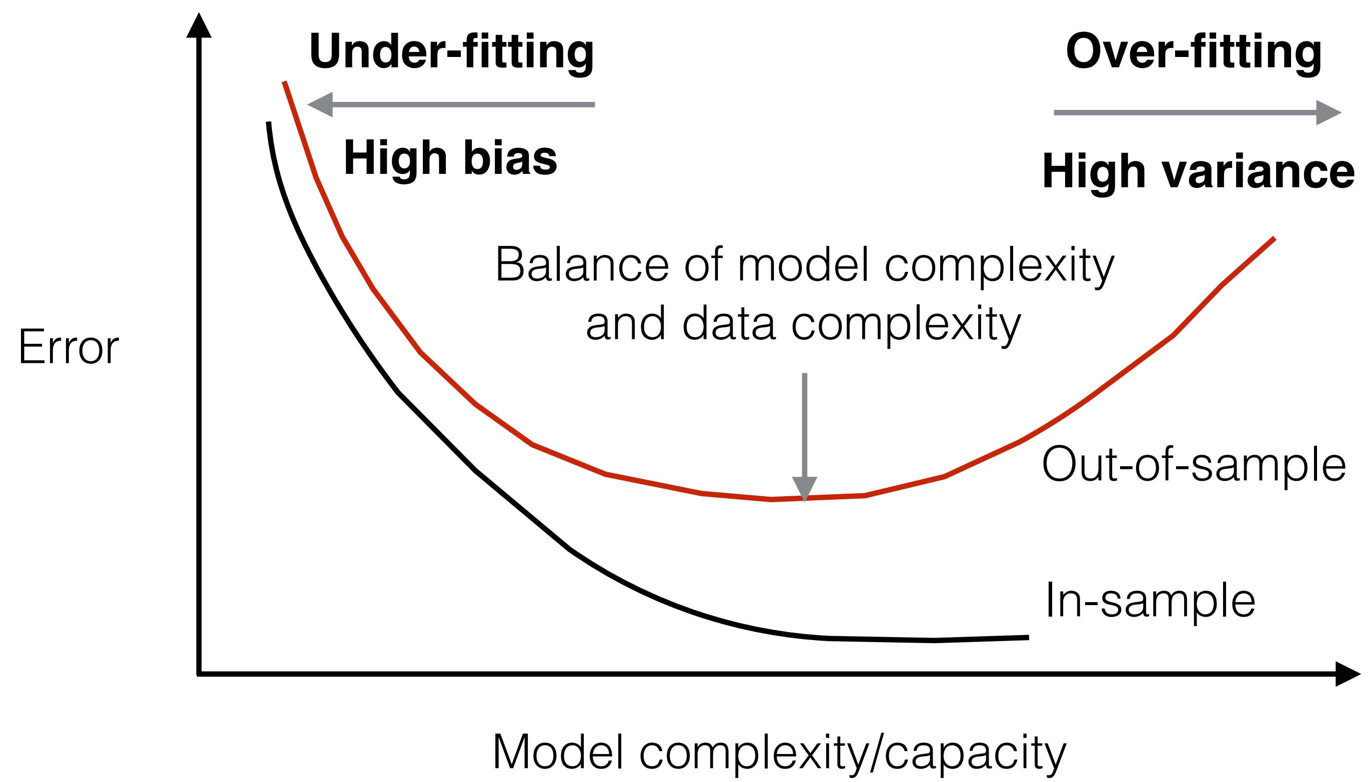
Overfitting

Trying to exactly match all available data is almost always a bad idea



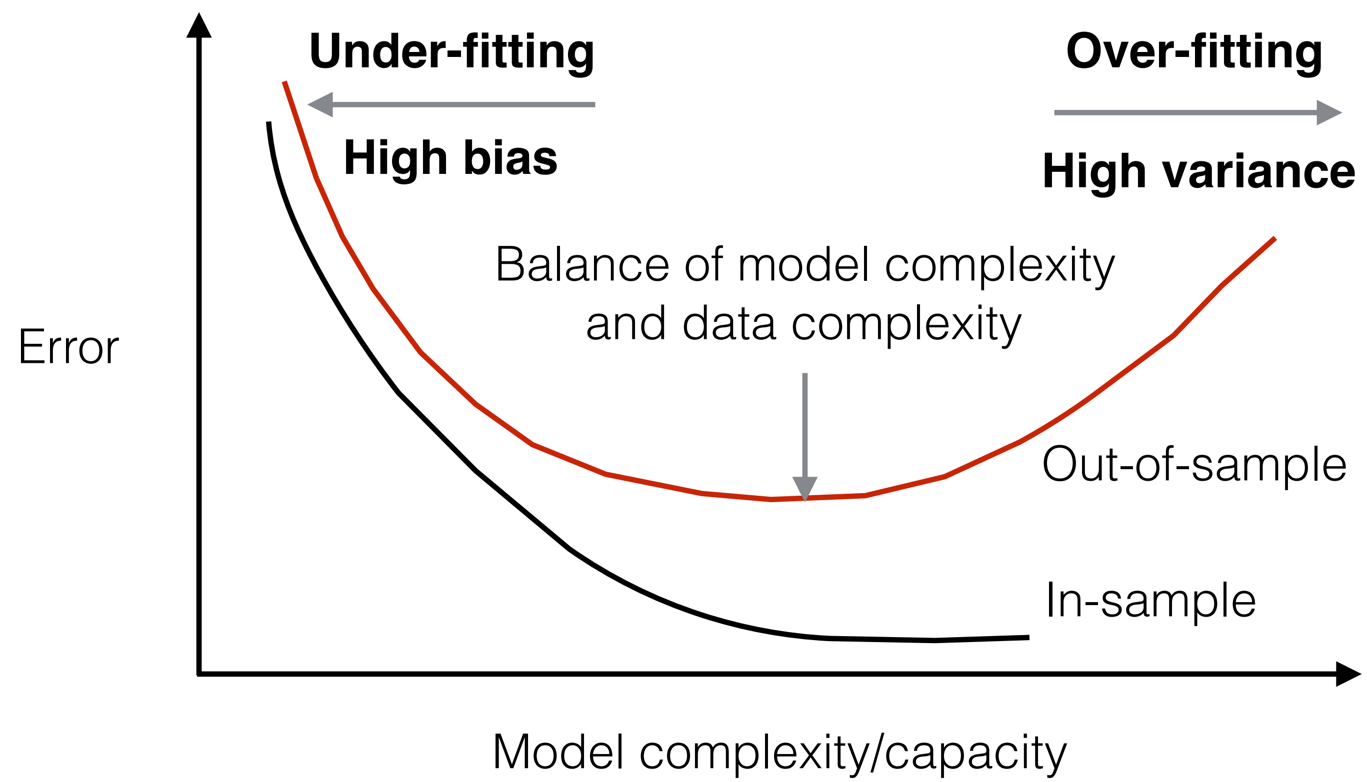
Overfitting

Trying to exactly match all available data is almost always a bad idea



Overfitting

Trying to exactly match all available data is almost always a bad idea



A good ML algorithm should achieve **two goals**:

1. Make the **training error** small (avoid under-fitting)
2. Make the gap between training and **test errors** small (avoid over-fitting)

Key ingredients: 1) data is i.i.d. $\sim p_{data}$, and 2) model **capacity control**

Model capacity and overfitting

- Model **capacity** controls model's ability to fit a wide variety of functions.
- Models with low capacity can **under-fit**, but models with high capacity can **over-fit**!
- Capacity is controlled by the choice of a **hypothesis space** (architecture), and other techniques