

# Machine Learning for econometrics

## Statistical learning and penalized regression

---

Matthieu Doutreligne

January 10, 2025

# Today's program

- Last session: importance of causal variable status
- Today: **predictive inference** in high dimensions
  - ▶ Statistical learning basics
  - ▶ Penalized linear regression for predictive inference
  - ▶ Hands-on with scikit-learn

# Today's program

- Last session: importance of causal variable status
- Today: **predictive inference** in high dimensions
  - ▶ Statistical learning basics
  - ▶ Penalized linear regression for predictive inference
  - ▶ Hands-on with scikit-learn
- Next session:
  - ▶ Flexible models: Trees, Random Forests, Gradient Boosting
  - ▶ Practical scikit-learn

# Table of contents

1. Statistical learning framework
2. Motivation: why prediction?
3. Statistical learning theory and intuitions
4. Lasso for predictive inference
5. A word on deep learning
6. Practical session

# Statistical learning framework

---

# Statistical learning, ie. predictive inference

## Goal

- Predict the value of an outcome based on one or more input variables.

## Setting

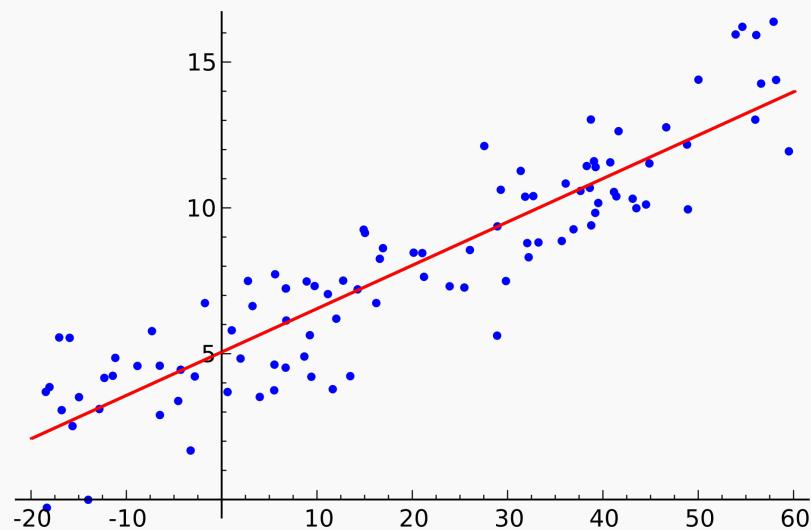
- Data: n pairs of (features, outcome),  $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$  identically and independently distributed (i.i.d.) from an unknown distribution  $P$ .
- Goal: find a function  $\hat{f} : \mathcal{X} \rightarrow \mathcal{Y}$  that approximates the true value of  $y$  ie. for a new pair  $(x, y)$ , we should have:

$$\hat{y} = \hat{f}(x) \approx y$$

## Vocabulary

Finding the appropriate model  $\hat{f}$  is called learning, training or fitting the model.

# Statistical learning, two classes of problems

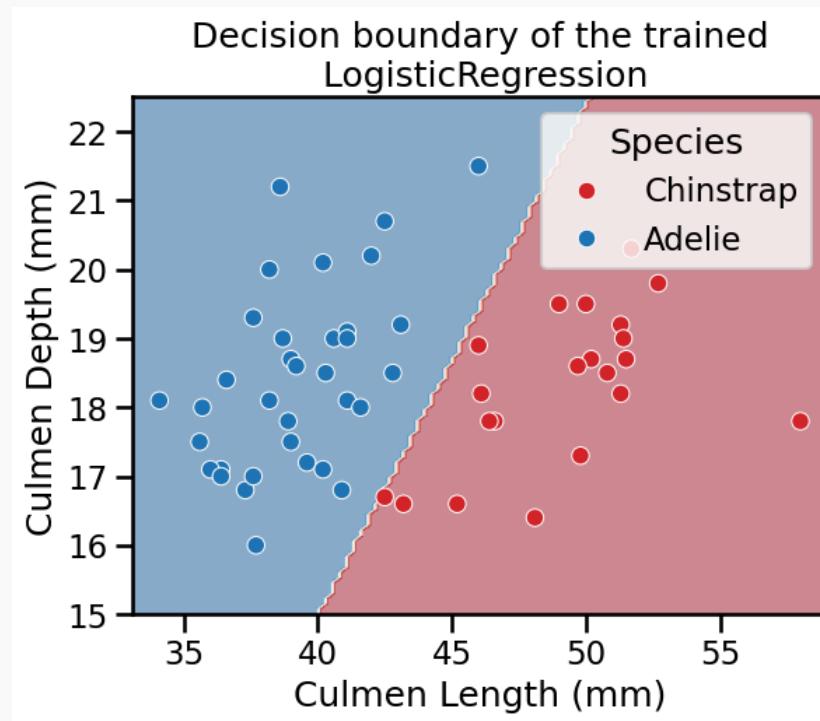


## Regression

- The outcome is continuous: eg. wage prediction
- The error is often measured by the mean squared error (MSE):

$$\text{MSE} = \mathbb{E} \left[ (Y - \hat{f}(X))^2 \right]$$

# Statistical learning, two classes of problems



## Classification

- Outcome is categorical: eg. diagnosis, loan default, ...
- Error is often measured with accuracy:

Misclassification rate =

$$\mathbb{E} \left[ \mathbb{1}(Y \neq \hat{f}(X)) \right]$$

with  $\hat{f} \in \{0, 1\}$  for binary classification

# Motivation: why prediction?

---

# Why do we need prediction for ?

## Statistical inference

- Goal: infer some intervention effect with a causal interpretation
- Require to regress “away” the relationship between the treatment or the outcome and the confounders -> more on this in sessions on Double machine learning.

## Predictive inference

- Some problems in economics requires accurate prediction without a causal interpretation (Kleinberg et al., 2015)
- Eg. Stratifying on a risk score (loan, preventive care, ...)

# Do we need more than linear models?

Let:

- $p$  is the number of features
- $n$  is the number of observations

## Maybe no

- Low-dimensional data:  $n \gg p$
- No non-linearities, no or few interactions between features

## Maybe yes

- High-dimensional data: ie.  $p \gg n$
- Non-linearities, many interactions between features

# Do we need more than linear models?

## When do we have “high-dimension”?

- Is  $p \gg n$  a common setting in economics?
- Consider the wage dataset:
  - ▶  $n = 5150$  individuals
  - ▶  $d = 18$  variables

# Do we need more than linear models?

## When do we have “high-dimension”?

- Is  $p \gg n$  a common setting in economics?
- Consider the wage dataset:
  - ▶  $n = 5150$  individuals
  - ▶  $d = 18$  variables
  - ▶ But, categorical variables, non-linearities and interactions increase the real number of features:
    - non-linearities: add polynomials of degree 2:  $p = 2 \times 18 = 36$
    - interactions:
      - Of degree 2:  $\binom{d}{2} = \binom{18}{2} = 153$
      - All interactions:  $2^d = 2^{18} - 18 - 1 = 262125$

# Is this common?

## Yes

- Categorical or text data are increasingly common
- image data is high-dimensional by nature
- Automation of data collection and storage leads to more collections of variables

## Some examples:

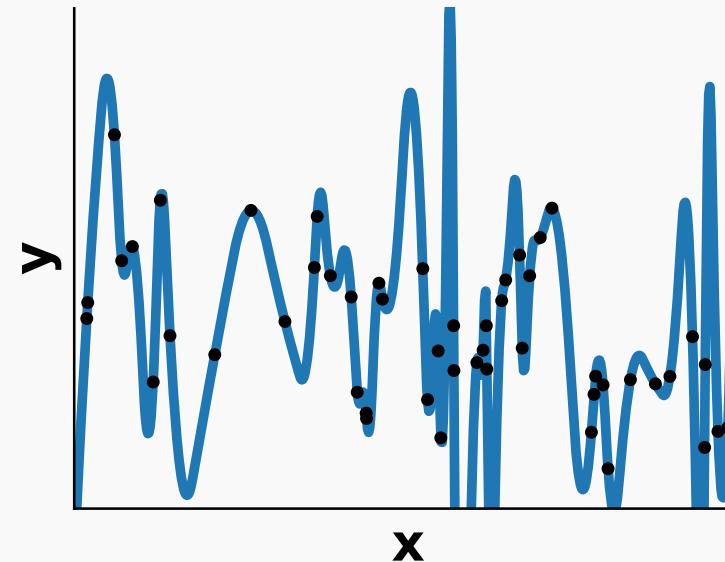
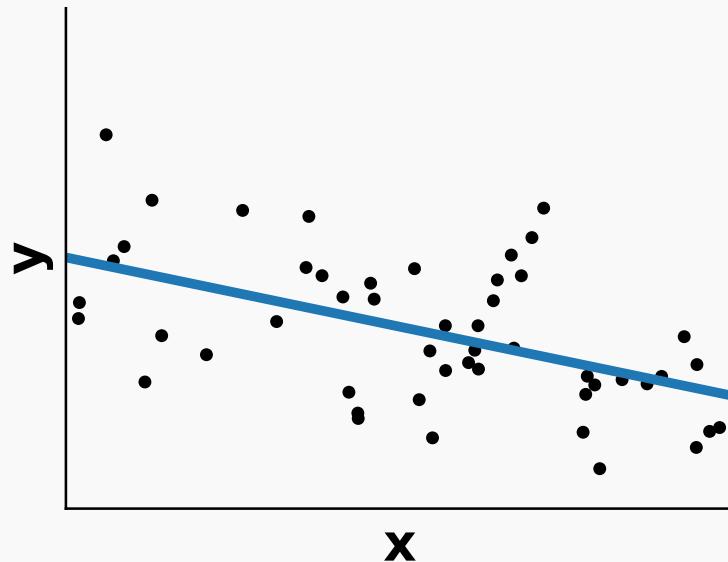
- The Current Population Survey (CPS) dataset has hundreds of variables, many of which are categorical
- The Système National des Données de Santé (SNDS) in France collects all reimbursements : many hundreds of variables, many of which are categorical
-

# Statistical learning theory and intuitions

---

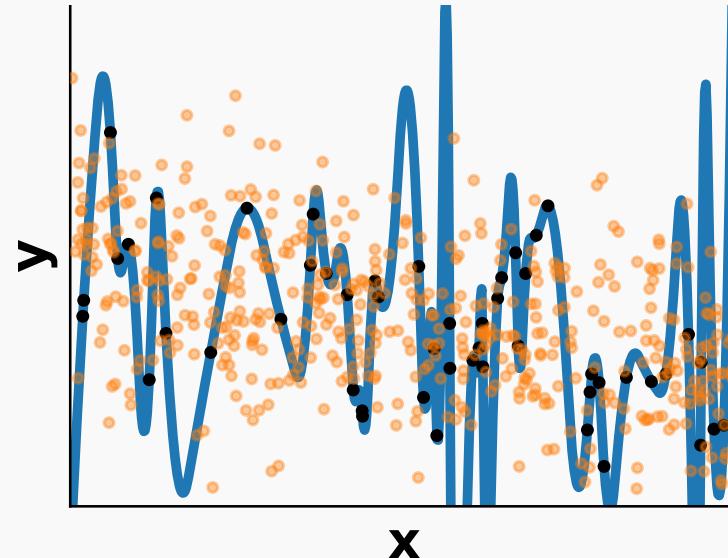
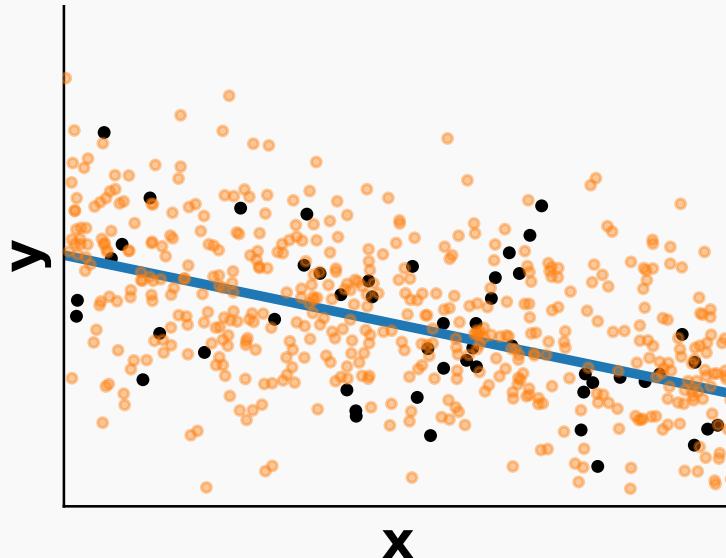
# Under vs. overfitting

Which data fit do you prefer?



# Under vs. overfitting

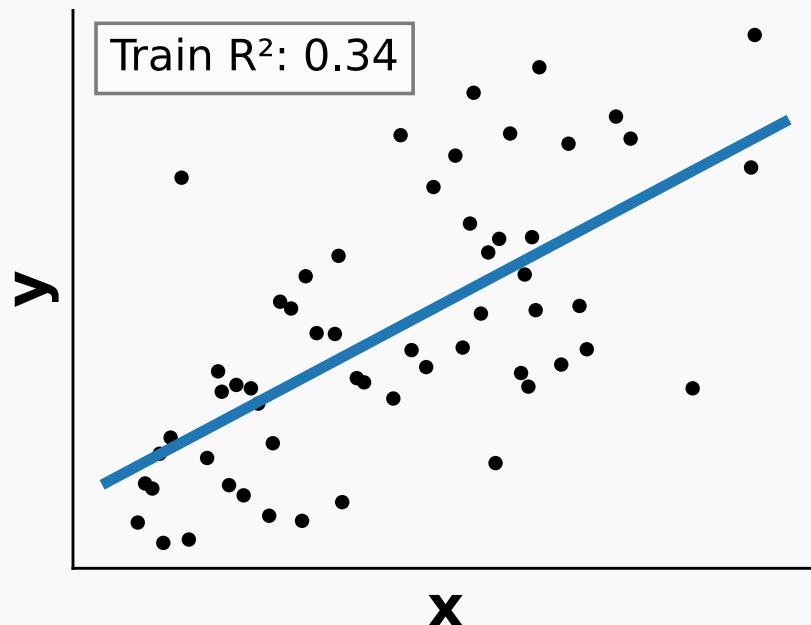
Which data fit do you prefer? (new data incoming)



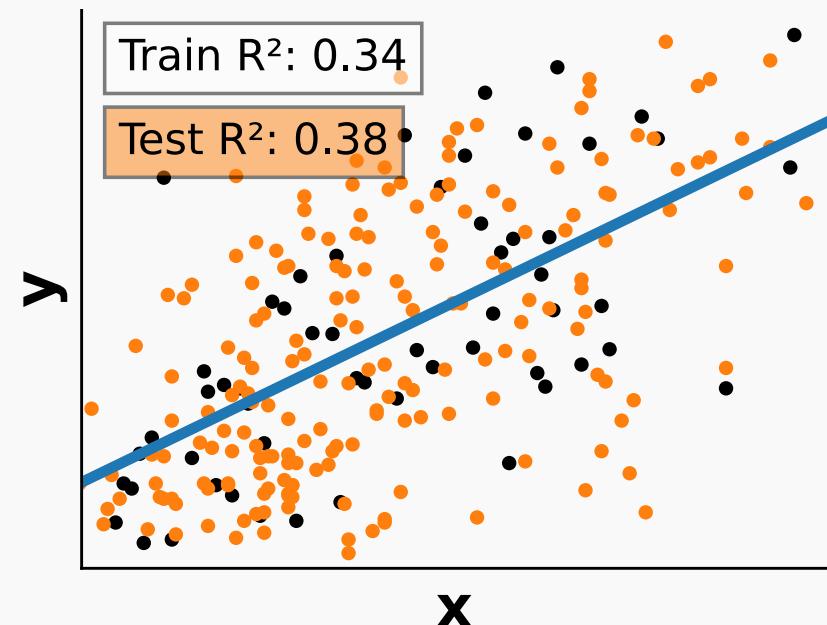
- Answering this question might be hard.
- Goal: create models that generalize.
- The good way of framing the question is: **how will the model perform on new data?**

# Train vs test error: simple models

**Measure the errors on the training data**  
= fitting



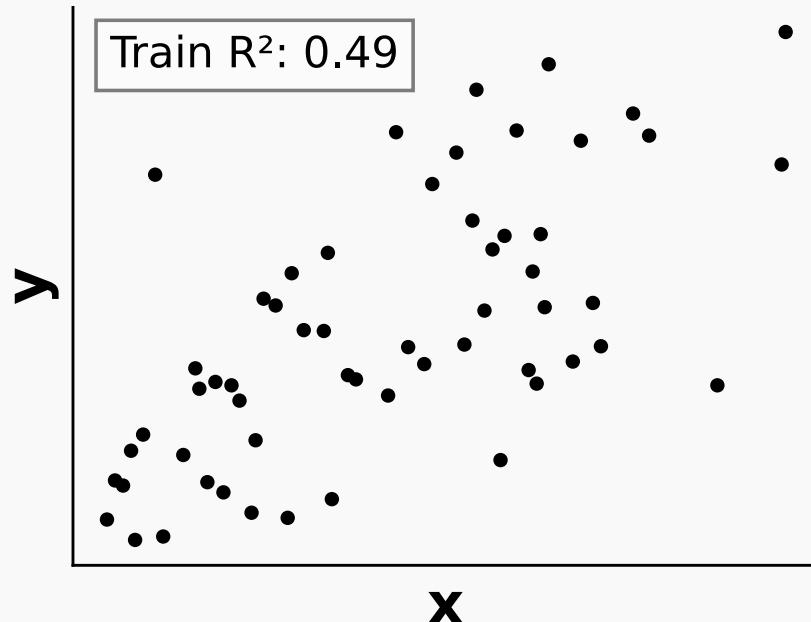
**Measure the performances on test data**  
= generalization



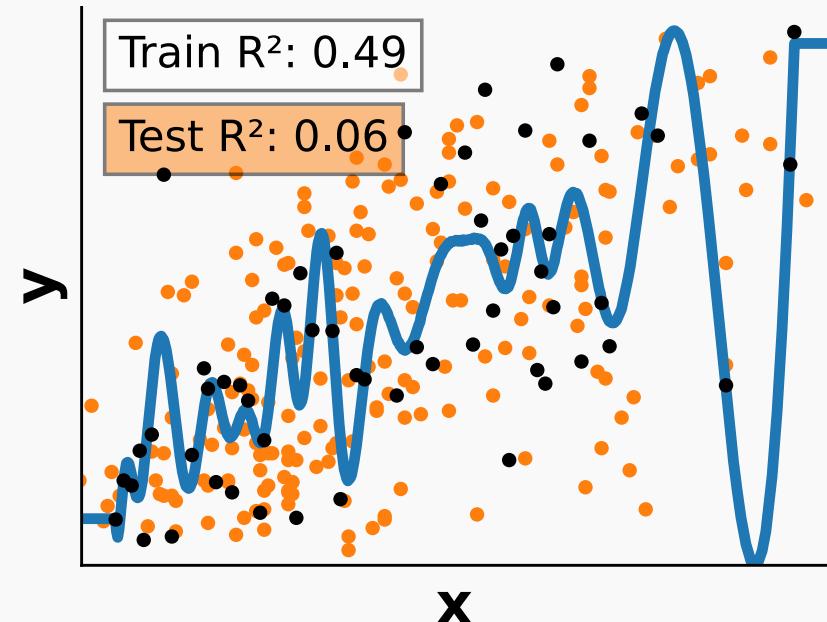
Here, no problem of overfitting: train vs test error are similar.

# Train vs test error: flexible models

**Measure the errors on the training data**  
= fitting



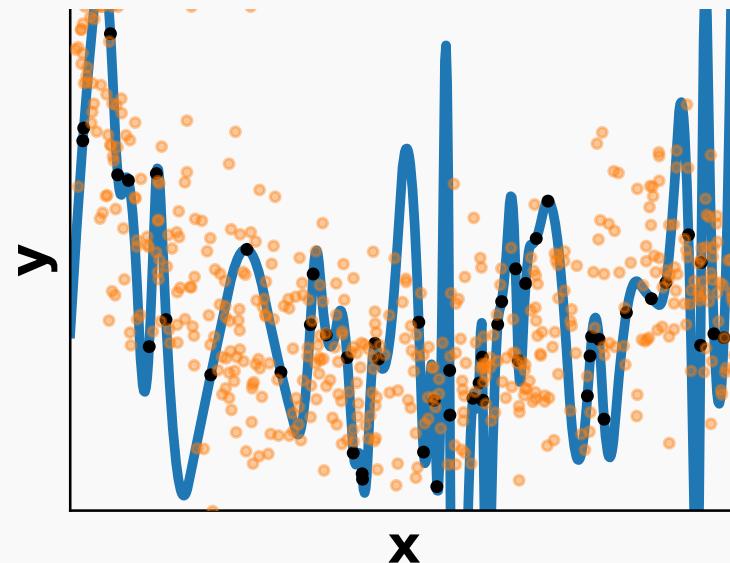
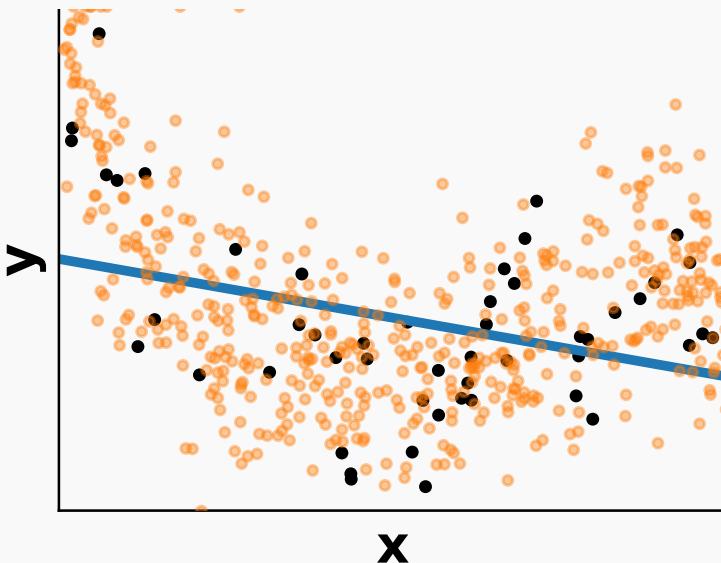
**Measure the performances on test data**  
= generalization



Overfitting: the model is too complex and captures noise.

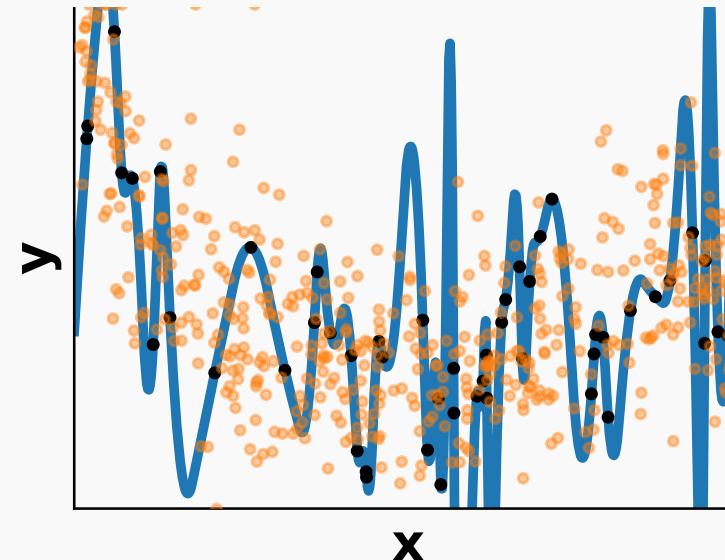
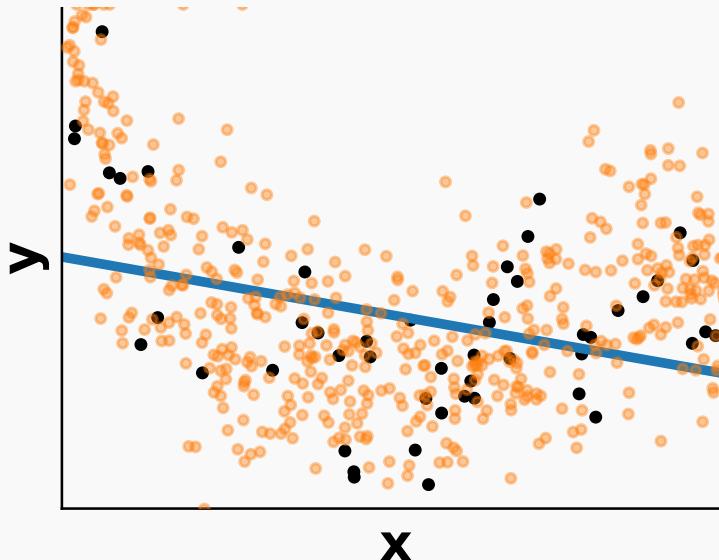
# Bias-variance trade-off: Under vs. overfitting

## How to choose the complexity of the model?



# Bias-variance trade-off: Under vs. overfitting

## How to choose the complexity of the model?



This trade-off is called **Bias variance trade-off**.

- Let's recover this trade-off in the context of statistical learning theory.

# Empirical Risk Minimization

- Define a loss function  $\ell$  that defines proximity between the predicted value  $\hat{y} = f(x)$  and the true value  $y$ :  $\ell(f(x), y)$
- Usually, for continuous outcomes, the squared loss is used:  $\ell(f(x), y) = (f(x) - y)^2$
- We choose among a (finite) family of functions  $f \in \mathcal{F}$ , the best possible function  $f^*$  minimizes the **risk or expected loss**  $\mathcal{E}(f) = \mathbb{E}\left[(f(x) - y)^2\right]$ :

$$f^* = \operatorname{argmin}_{f \in \mathcal{F}} \mathbb{E}\left[(f(x) - y)^2\right]$$

## Empirical risk minimization: estimation error

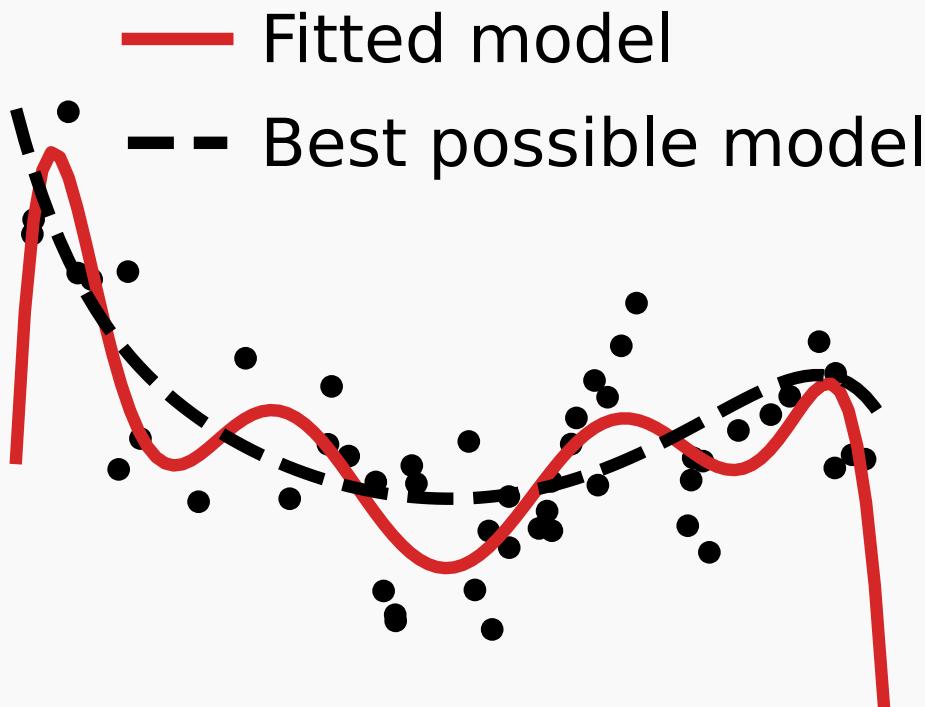
- In finite sample regimes, the expectation is not accessible since we only have access to a finite number of data pairs
- In practice, we minimize the **empirical risk** or average loss  $R_{\text{emp}} = \sum_{i=1}^n (f(x_i) - y_i)^2$ :

$$\hat{f} = \operatorname{argmin}_{f \in \mathcal{F}} \sum_{i=1}^n (f(x_i) - y_i)^2$$

- This creates the **estimation error**, related to sampling noise:

$$\mathcal{E}(\hat{f}) - \mathcal{E}(f^\star) = \mathbb{E}\left[\left(\hat{f}(x) - y\right)^2\right] - \mathbb{E}\left[\left(f^\star(x) - y\right)^2\right] \geq 0$$

## High estimation error means overfit



### Model is too complex

- The model is able to recover the true generative process
- But its flexibility captures noise

### Too much noise

### Not enough data

# Bayes error rate: Randomness of the problem

- Interesting problems exhibit randomness

$$y = g(x) + e \text{ with } E(e|x) = 0 \text{ and } \text{Var}(e|x) = \sigma^2$$

- The best possible estimator is  $g(\cdot)$ , yielding the Bayes error, the unavoidable error:

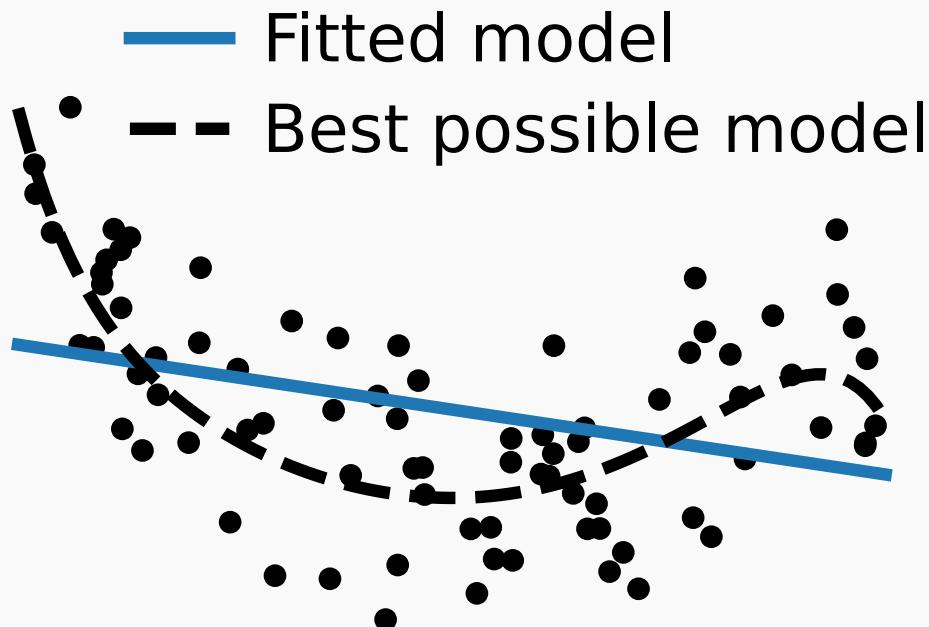
$$\mathcal{E}(g) = \mathbb{E}[(g(x) + e - g(x))^2] = \mathbb{E}[e^2]$$

## Empirical risk minimization: approximation error

- In practice you don't know the class of function in which the true function lies :  $y \approx g(x)$  : Every model is wrong !
- You are choosing the best possible function in the class of functions you have access to:  
 $f^* \in \mathcal{F}$  eg. linear models, polynomials, trees, ...
- This creates the **approximation error**:

$$\mathcal{E}(f(\star)) - \mathcal{E}(g) = \mathbb{E}\left[(f^*(x) - y)^2\right] - \mathbb{E}\left[(g(x) - y)^2\right] \geq 0$$

## High approximation error means underfit



**Model is too simple for the data**

- its best fit does not approximate the true generative process
- Yet it captures little noise

**Low noise**

**Rapidly enough data to fit the model**

# Bias variance trade-off: Putting the pieces together

**Decomposition of the empirical risk of a fitted model  $\hat{f}$**

$$\mathcal{E}(\hat{f}) = \underbrace{\mathcal{E}(g)}_{\text{Bayes error}} + \underbrace{\mathcal{E}(f^*) - \mathcal{E}(g)}_{\text{approximation error}} + \underbrace{\mathcal{E}(\hat{f}) - \mathcal{E}(f^*)}_{\text{estimation error}}$$

# Lasso for predictive inference

---

## **High bias == underfitting**

- systematic prediction errors
- the model prefers to ignore some aspects of the data
- misspecified models

## **High variance == overfitting:**

- prediction errors without obvious structure
- small change in the training set, large change in model
- unstable models









# A word on deep learning

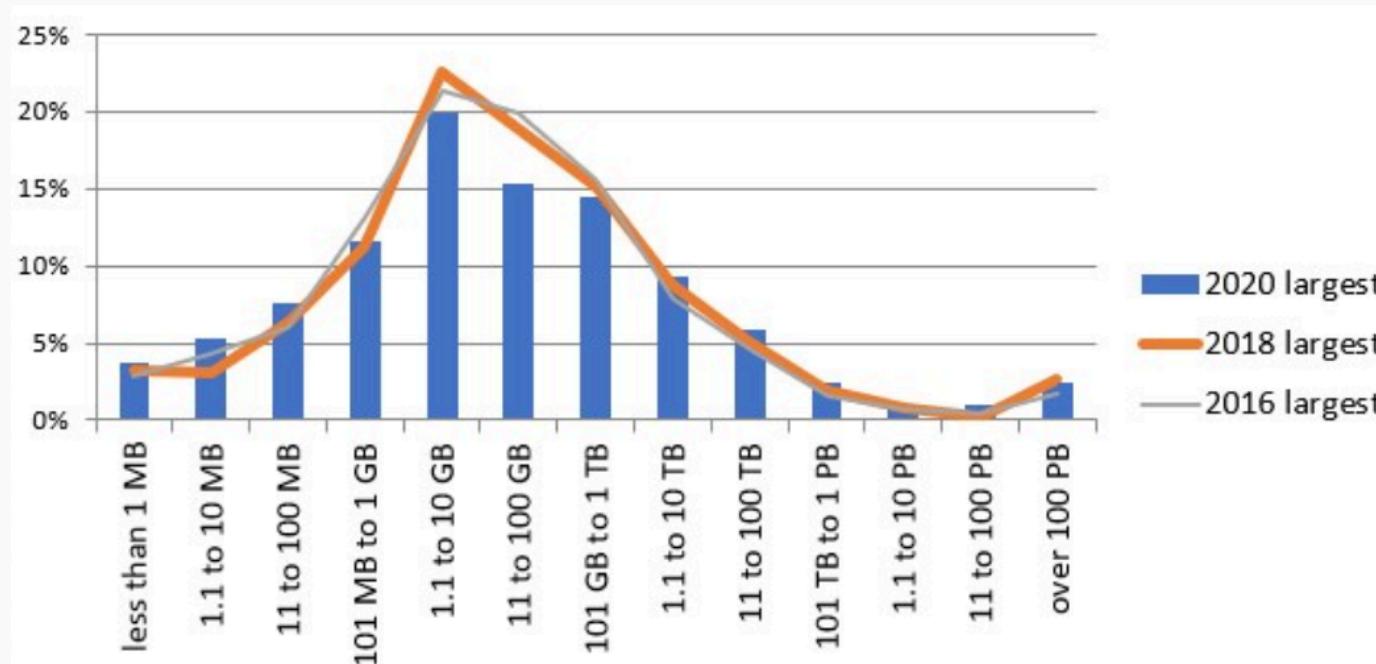
---

# Why not use deep learning everywhere?

- Success of deep learning in image, speech recognition and text
- Why not so used in economics?

# Limited data settings

- Typically in economics everywhere, we have a limited number of observations

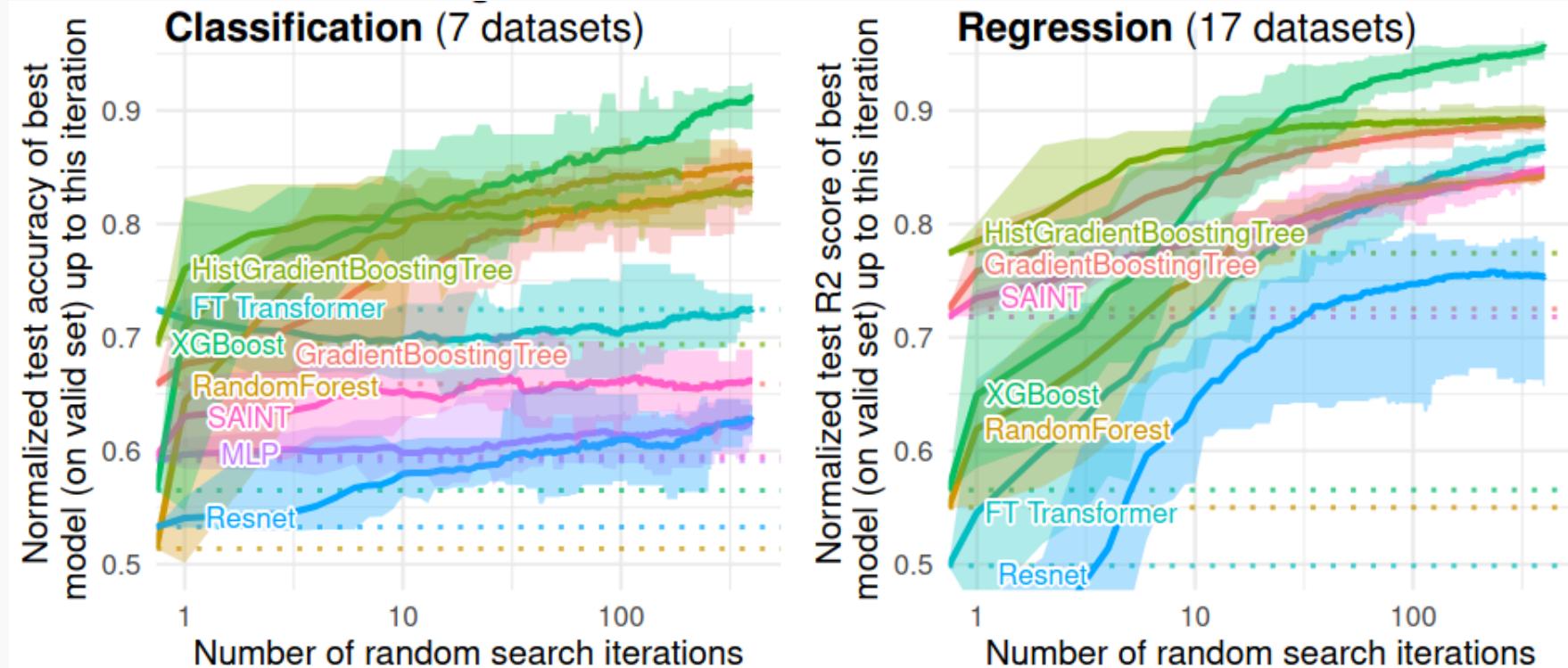


Typical dataset are mid-sized. This does not change with time.<sup>1</sup>

<sup>1</sup><https://www.kdnuggets.com/2020/07/poll-largest-dataset-analyzed-results.html>

# Deep learning underperforms on data tables

**Tree-based methods outperform tailored deep learning architectures (Grinsztajn et al., 2022)**



DAG for a RCT: the treatment is independent of the confounders

# Practical session

---

# Resources

- <https://web.stanford.edu/~swager/stats361.pdf>
- <https://www.mixtapesessions.io/>
- <https://alejandroschuler.github.io/mci/>
- <https://theeffectbook.net/index.html>

## *Bibliography*

*Grinsztajn, L., Oyallon, E., & Varoquaux, G. (2022). Why do tree-based models still outperform deep learning on typical tabular data?. Advances in Neural Information Processing Systems, 35, 507–520.*

*Kleinberg, J., Ludwig, J., Mullainathan, S., & Obermeyer, Z. (2015). Prediction policy problems. American Economic Review, 105(5), 491–495.*