# Machine Learning for econometrics

## Flexible models for tabular data

Matthieu Doutreligne

February 18th, 2025

# Reminder from previous session

- Statistical learning 101: bias-variance trade-off
- Regularization for linear models: Lasso, Ridge, Elastic Net
- Transformation of variables: polynomial regression
-

# Reminder from previous session

- Statistical learning 101: bias-variance trade-off
- Regularization for linear models: Lasso, Ridge, Elastic Net
- Transformation of variables: polynomial regression
- 🤔 But… How to select the best model? the best hyper-parameters?

# Table of contents

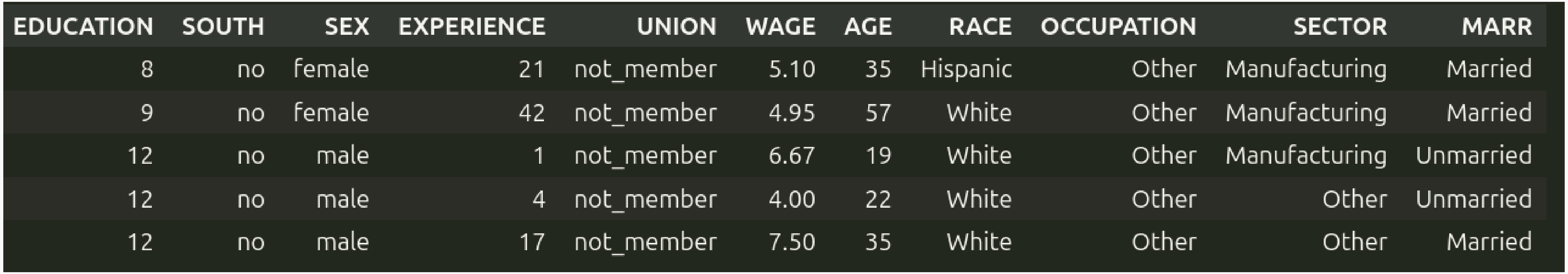
1. Model evaluation and selection with cross-validation
2. Tree, random forests and boosting
3. A word on other families of models

# Model evaluation and selection with cross-validation

# A closer look at model evaluation: Wage example

## Example with the Wage dataset

- Raw dataset: (N=534, p=11)

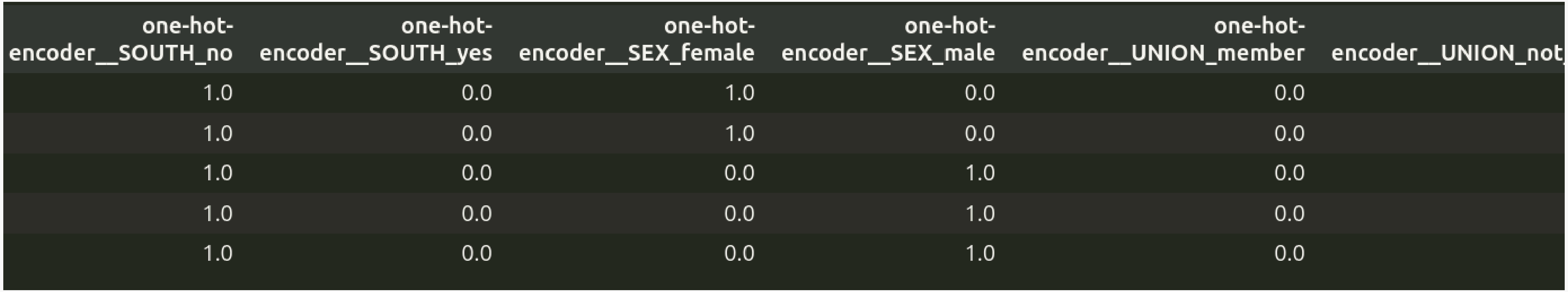


| EDUCATION | SOUTH | SEX | EXPERIENCE | UNION | WAGE | AGE | RACE | OCCUPATION | SECTOR | MARR |
|---:|---:|---:|---:|---:|---:|---:|---:|---:|---:|---:|
| 8 | no | female | 21 | not_member | 5.10 | 35 | Hispanic | Other | Manufacturing | Married |
| 9 | no | female | 42 | not_member | 4.95 | 57 | White | Other | Manufacturing | Married |
| 12 | no | male | 1 | not_member | 6.67 | 19 | White | Other | Manufacturing | Unmarried |
| 12 | no | male | 4 | not_member | 4.00 | 22 | White | Other | Other | Unmarried |
| 12 | no | male | 17 | not_member | 7.50 | 35 | White | Other | Other | Married |

# A closer look at model evaluation: Wage example

## Example with the Wage dataset

- Raw dataset: (N=534, p=11)
- Transformation: encoding categorical data, scaling numerical data: (N=534, p=23)

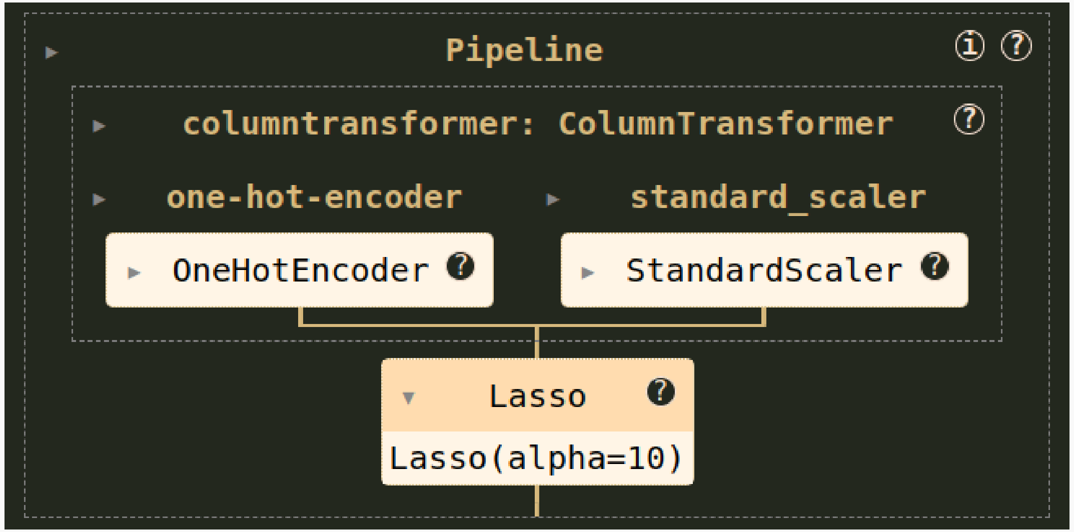


| one-hot-encoder__SOUTH_no | one-hot-encoder__SOUTH_yes | one-hot-encoder__SEX_female | one-hot-encoder__SEX_male | one-hot-encoder__UNION_member | encoder__UNION_not |
|---:|---:|---:|---:|---:|---:|
| 1.0 | 0.0 | 1.0 | 0.0 | 0.0 | |
| 1.0 | 0.0 | 1.0 | 0.0 | 0.0 | |
| 1.0 | 0.0 | 0.0 | 1.0 | 0.0 | |
| 1.0 | 0.0 | 0.0 | 1.0 | 0.0 | |
| 1.0 | 0.0 | 0.0 | 1.0 | 0.0 | |

# A closer look at model evaluation: Wage example

## Example with the Wage dataset

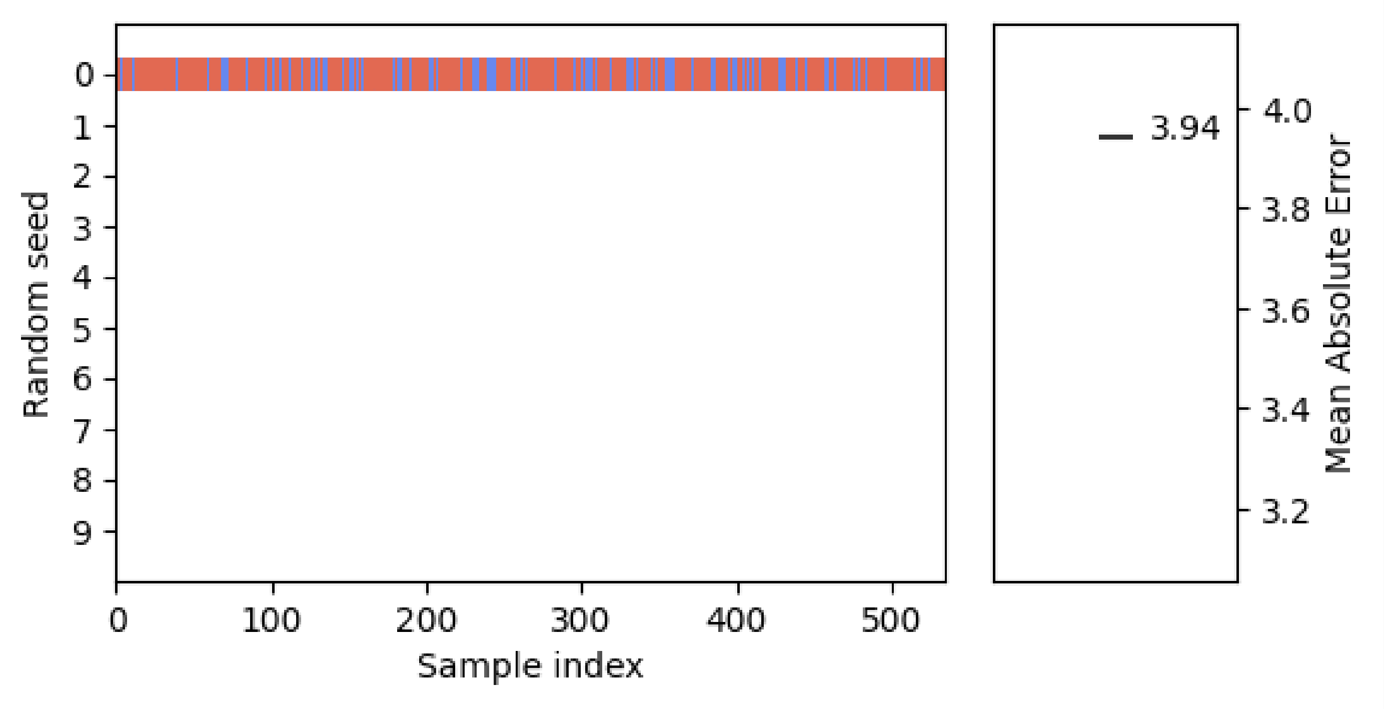
- Raw dataset: (N=534, p=11)
- Transformation: encoding categorical data, scaling numerical data: (N=534, p=23)
- Regressor: Lasso with regularization parameter ($\alpha = 10$)



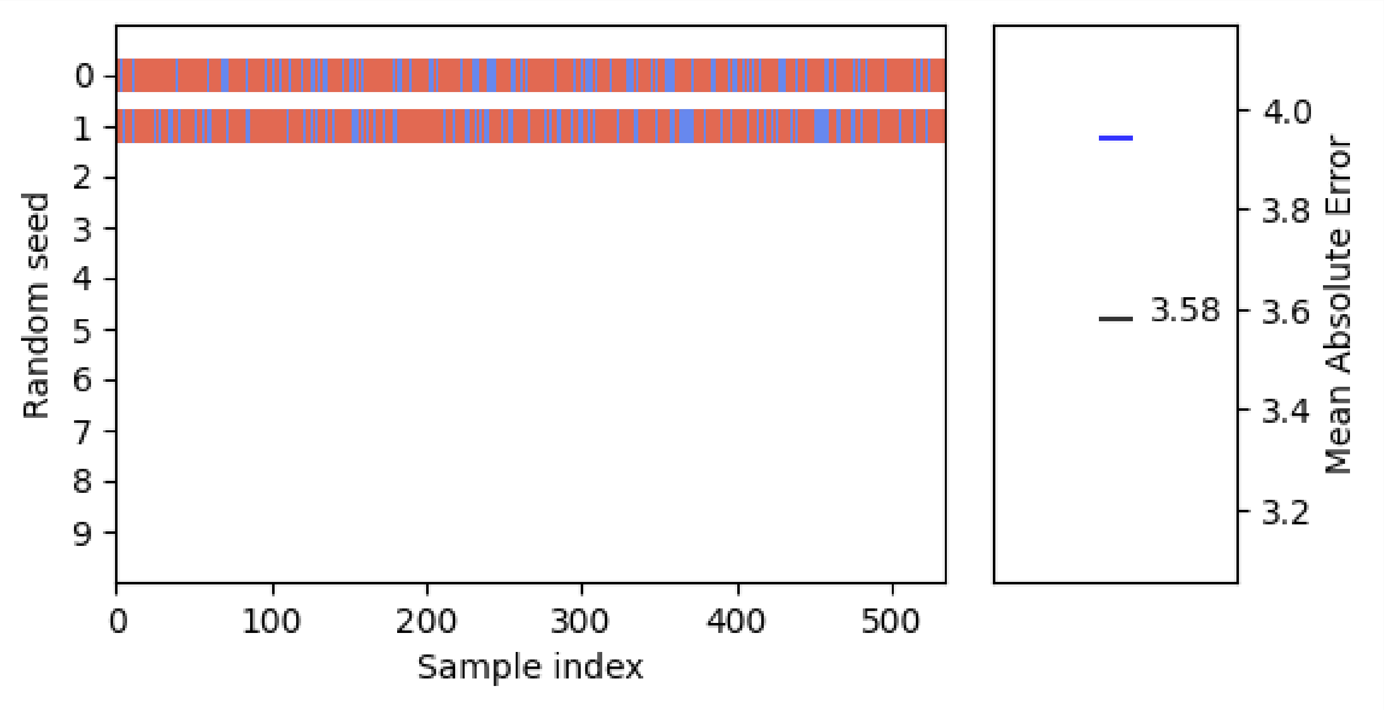
Lasso(alpha=10)

# Repeated train/test splits

## Splitting once: In red, the training set, in blue, the test set

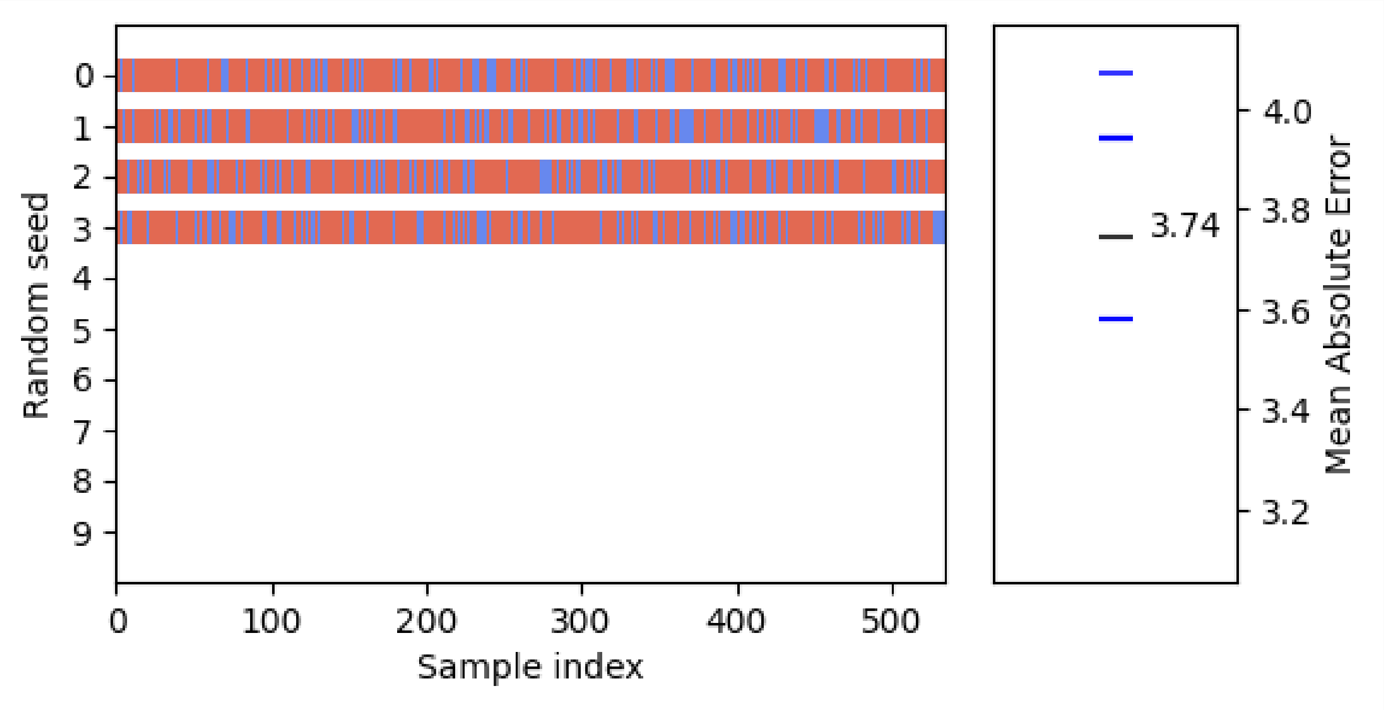


## But we could have chosen another split ! Yielding a different MAE



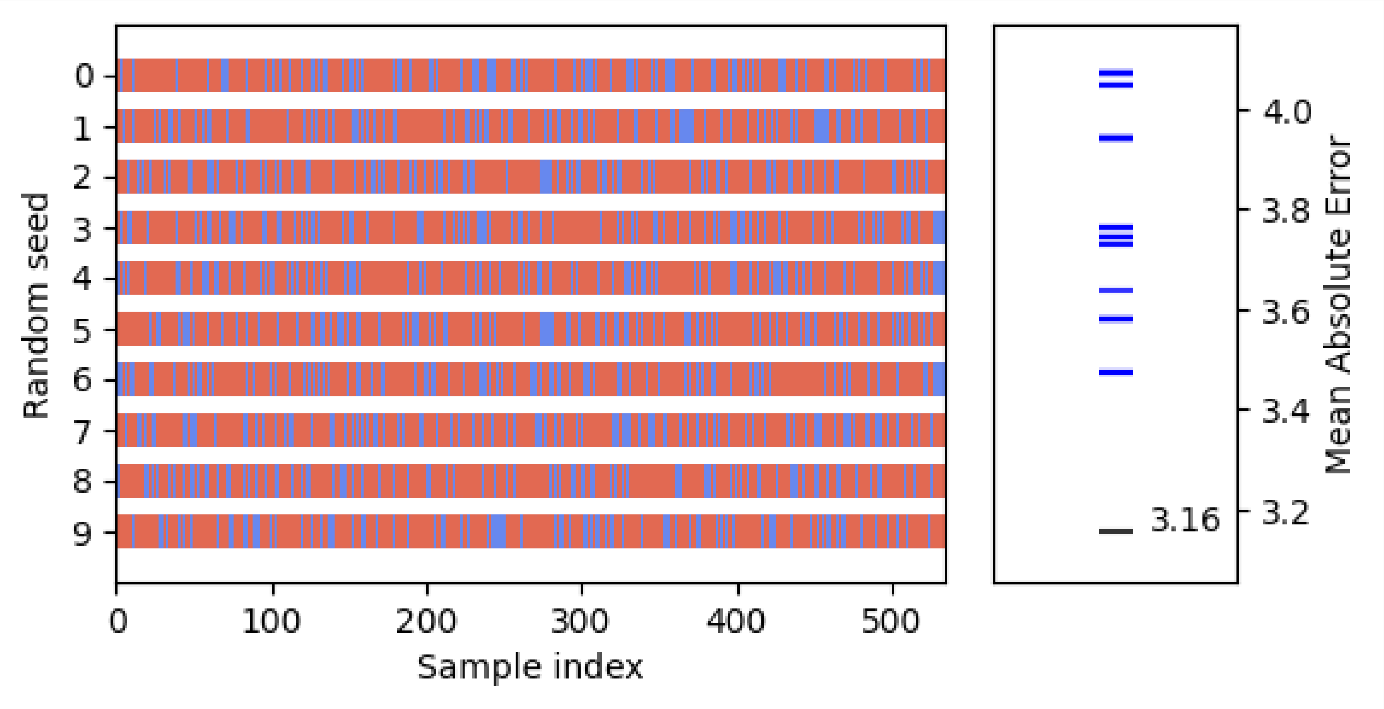
# Repeated train/test splits

## And another split...



# Repeated train/test splits

## Splitting ten times



🎉 **Distribution of MAE:** $3.71 \pm 0.26$

# Repeated train/test splits = Cross-validation

- In sklearn, it can be instantiated with `cross_validate`.

Python

```python
from sklearn.model_selection import cross_validate
from sklearn.model_selection import ShuffleSplit

cv = ShuffleSplit(n_splits=40, test_size=0.3, random_state=0)
cv_results = cross_validate(
    regressor, data, target, cv=cv, scoring="neg_mean_absolute_error"
)
```

# Repeated train/test splits = Cross-validation

- In sklearn, it can be instantiated with `cross_validate`.
- It is a more robust way to evaluate the model’s performance.
- We get a more robust estimate by taking the mean over the repetitions.
- We get a better idea of the variability of the model’s performance: similar to bootstrapping (but different).

# Cross-validation is not model selection

## Cross-validation

🤩 **Robustly estimate one model’s generalization performance**

# Cross-validation is not model selection

## Cross-validation

🤩 **Robustly estimate one model’s generalization performance**

**But still, how to select the best model among multiple models with different hyper-parameters?s**

# Naive cross-validation to select the best model

# Nested cross-validation to select the best model

# Tree, random forests and boosting

# Random Forests for predictive inference

# Boosting

# Ensemble models

# A word on other families of models

# Why not use deep learning everywhere?

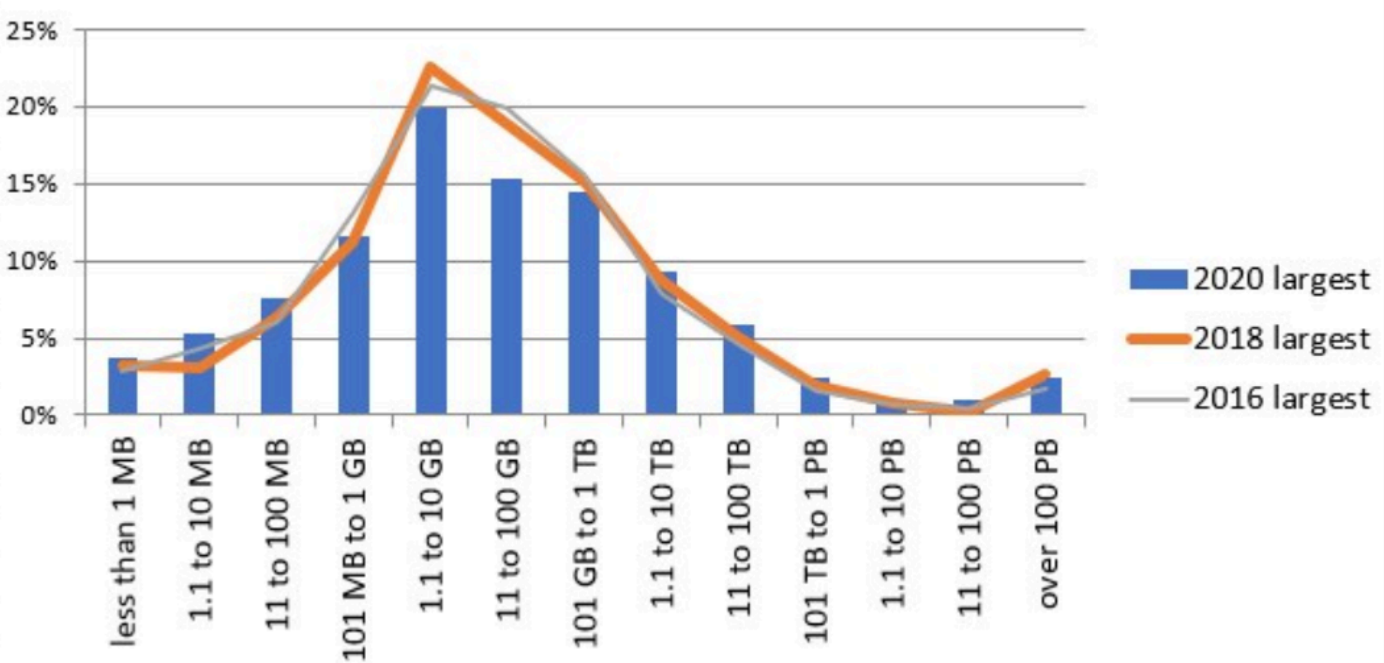
- Success of deep learning (aka deep neural networks) in image, speech recognition and text
- Why not so used in econometrics?

**Deep learning needs a lot of data (typically $N \approx 1$ million)**

- Do we have this much data in econometrics?

# Limited data settings

- Typically in economics (but also everywhere), we have a limited number of observations



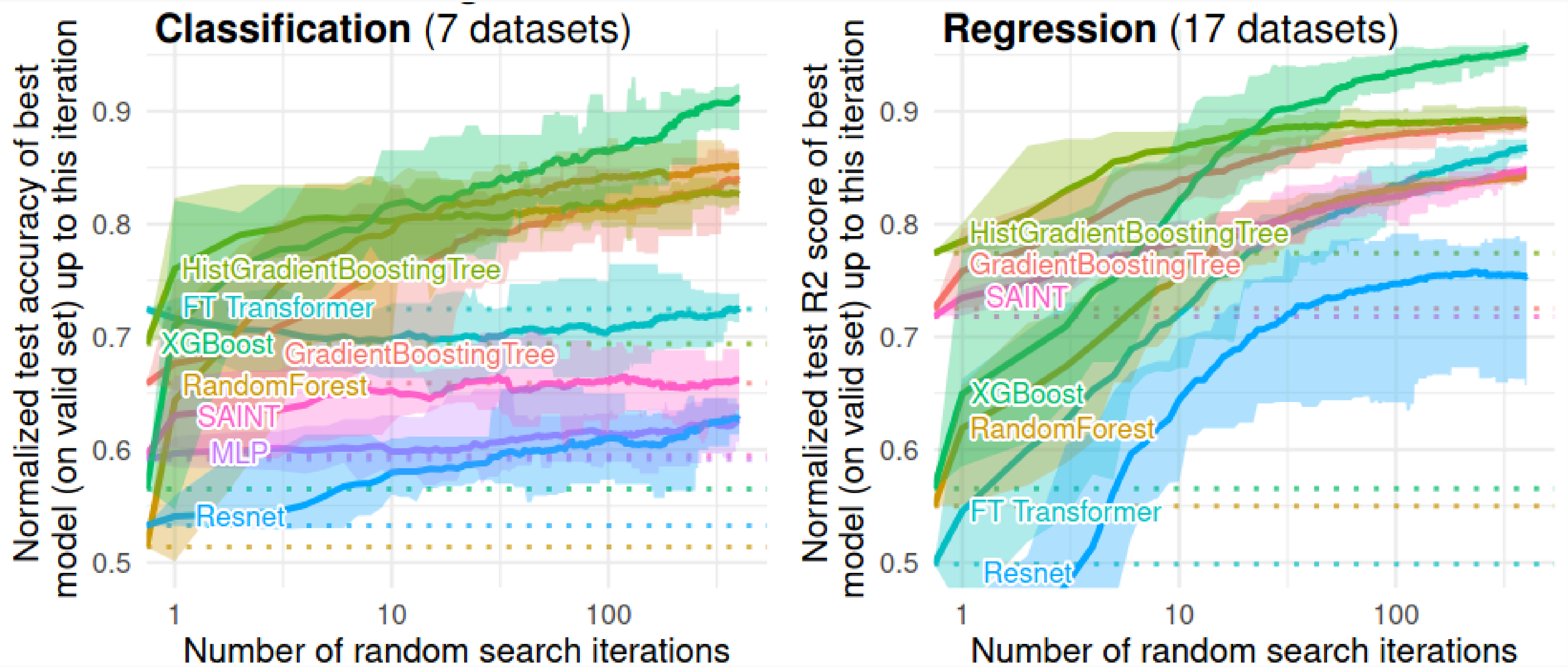
Typical dataset are mid-sized. This does not change with time.[1]

[1] 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)**

# Deep learning underperforms on data tables



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

# Other well known families of models

**Generalized linear models**

**Support vector machines**

**Gaussian processes**

# 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.*