

Double ML: Causal Inference based on ML

Part II: Double Machine Learning in Practice

uai2022, August 1, 2022, Eindhoven

Philipp Bach, Martin Spindler (UHH & Economic AI)

Collaborators: Victor Chernozhukov (MIT), Malte Kurz (TUM)

Introduction to DoubleML



- **DoubleML** provides a general implementation of the Double Machine Learning approach by Chernozhukov et al. (2018) in Python and R
- There are also other open source libraries available for causal machine learning
 - **Causa1ML** (uber, <https://github.com/uber/causalml>, Chen et al., 2020) - variety of causal ML learners, i.a. with focus on uplift modeling, CATEs and IATEs
 - **EconML** (microsoft research, <https://github.com/microsoft/EconML>, Battocchi et al., 2021) - various causal estimators based on machine learning, among others based on double machine learning approach
 - ...

Causa1ML and **EconML** have a focus on heterogeneity of treatment effects from their start on

DoubleML focuses on implementing the DML approach and its extensions (example: heterogeneity)

→ Object-orientated implementation based on orthogonal score

→ Extendibility and flexibility

Key ingredient and Implementation

- Orthogonal Score
 - Object-oriented implementation
 - Exploit common structure being centered around a (linear) score function $\psi(\cdot)$
- High-quality ML
 - State-of-the-art ML prediction and tuning methods
 - Provided by `scikit-learn` and `sckit-learn`-like learners
- Sample Splitting
 - General implementation of sample splitting

DoubleML Package Dependencies

scikit-learn



pandas



numpy



SciPy



statsmodels



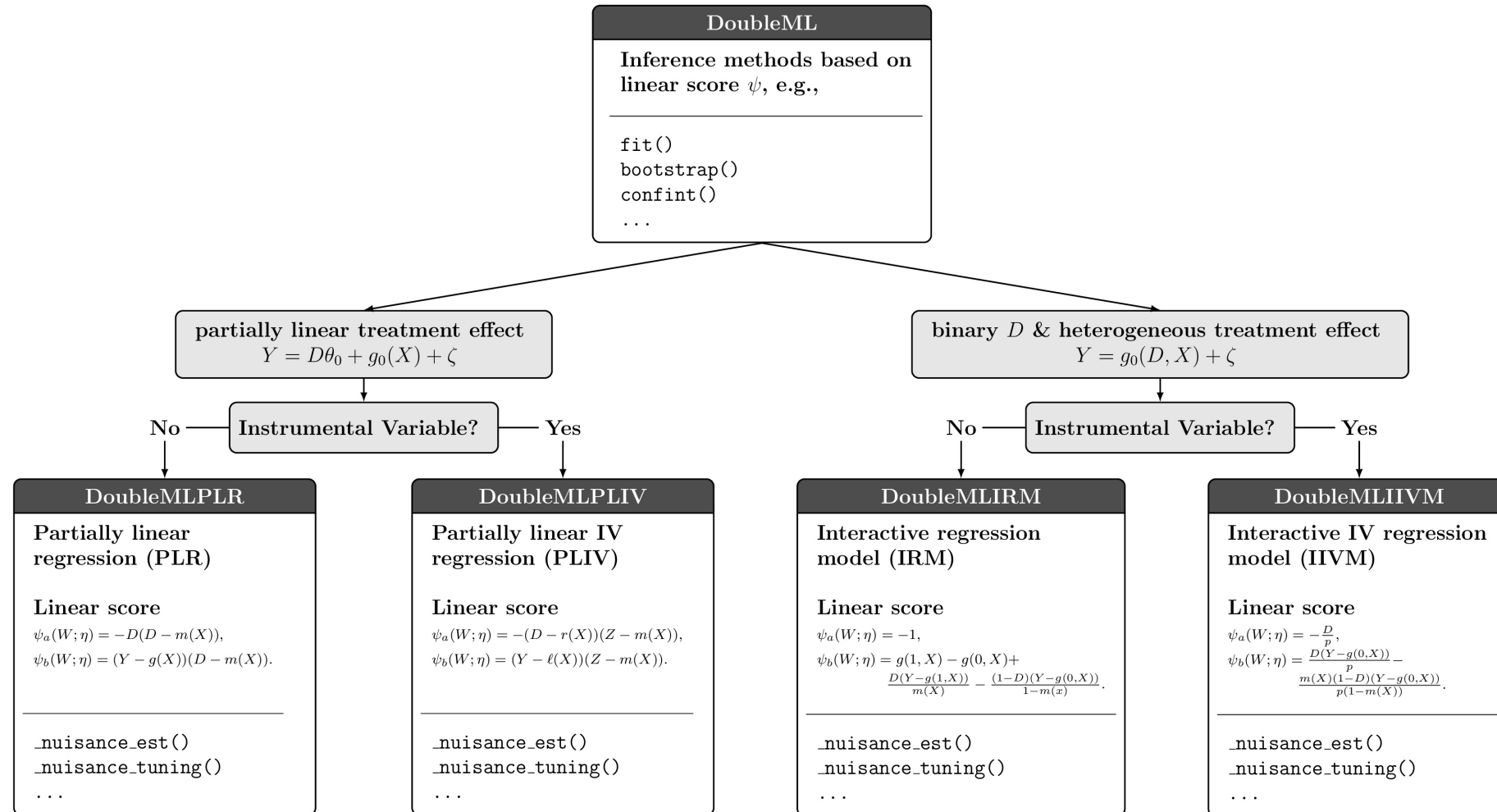
joblib



Why an Object-Orientated Implementation?

- Given the components $\psi^a(\cdot)$ & $\psi^b(\cdot)$ of a linear Neyman orthogonal score function $\psi(\cdot)$, a **general implementation** is possible for
 - The estimation of the **orthogonal parameters**
 - The computation of the **score** $\psi(W; \theta, \eta)$
 - The estimation of **standard errors**
 - The computation of **confidence intervals**
 - A **multiplier bootstrap** procedure for simultaneous inference
- The **sample splitting** can be implemented in general as well
 - Implemented in the **abstract base class DoubleML**
- The **score components** and the estimation of the **nuisance models** have to be implemented **model-specifically**
 - Implemented in **model-specific classes** inherited from **DoubleML**

Class Structure and Causal Models



Advantages of the Object-Orientation

- **DoubleML** gives the user a **high flexibility** with regard to the specification of DML models:
 - Choice of ML methods for approximating the nuisance functions
 - Different resampling schemes (repeated cross-fitting)
 - DML algorithms DML1 and DML2
 - Different Neyman orthogonal score functions
- **DoubleML** can be **easily extended**
 - New model classes with appropriate Neyman orthogonal score function can be inherited from **DoubleML**
 - The package features **callables** as score functions which makes it easy to extend existing model classes
 - The resampling schemes are customizable in a flexible way

Getting started with DoubleML!

Installation

Install the latest release via pip or conda, see [installation guide](#)

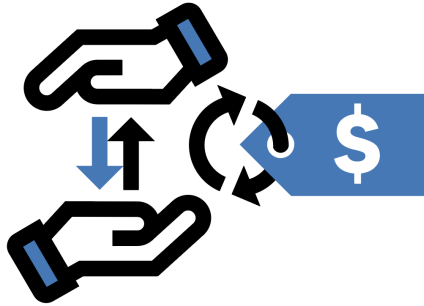
```
pip install -U DoubleML
```

```
conda install -c conda-forge doubleml
```

Install development version from GitHub <https://github.com/DoubleML/doubleml-for-py>

See the [Getting Started](#) page of the tutorial website for more information on prerequisites.

Data Example: Demand Estimation



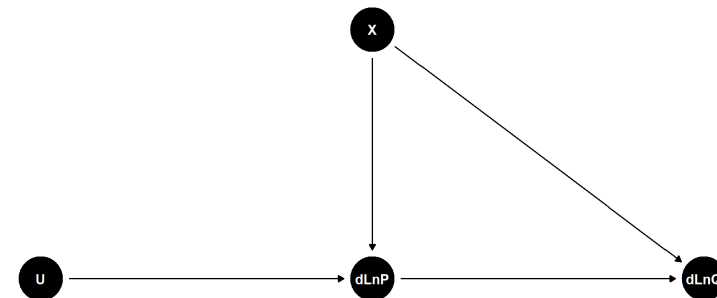
Data Source

- Data example based on a [blogpost](#) by Lars Roemheld (Roemheld, 2021)
- Original real data set publicly available via [kaggle](#), preprocessing notebook available [online](#)

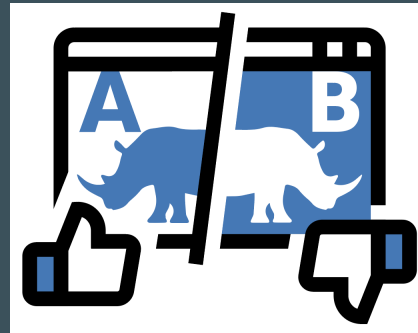
Causal Problem

- Price elasticity of demand: What is the effect of a price change, $d\ln P$, on demanded quantity, $d\ln Q$?
- **Observational study:** Flexibly adjust for confounding variables \mathbf{X} , e.g. product characteristics

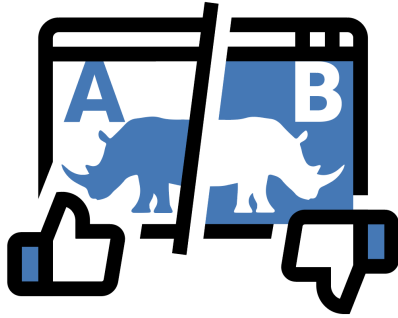
Causal Diagram (DAG)



Hands On Notebook



Data Example: A/B Testing



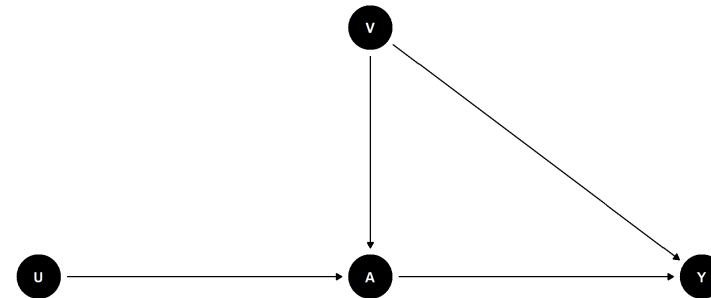
Data Source

- Data example based on a randomly chosen DGP created for the 2019 ACIC Data Challenge.

Causal Problem

- Online shop: What is the effect of a new ad design A on sales Y (in \$100)?
- Observational study: Necessary to adjust for confounding variables V

Causal Diagram (DAG)



Online Resources

- The notebook is organized according to the [DoubleML Workflow](#)
- Extensive [User Guide](#) available via docs.doubleml.org
- Documentation for the Python API available via <https://docs.doubleml.org/stable/api/api.html>
- Paper for the Python package available from [JMLR](#) or [arxiv](#)

References

Open-Source Libraries for Causal Machine Learning

- Battocchi, K, Dillon, E., Hei, M., Lewis, G., Oka, P., Oprescu, M. and Syrgkanis, V. (2021), EconML: A Python package for ML-based heterogeneous treatment effects estimation, <https://github.com/microsoft/EconML>. Version 0.11.1
- Chen, H., Harinen, T., Lee, J.-Y., Yung, M. and Zhao, Z. (2020), CausalML: Python package for causal machine learning, [arXiv:2002.11631](https://arxiv.org/abs/2002.11631) [cs.CY]

DoubleML Package for Python and R

- Bach, P., Chernozhukov, V., Kurz, M. S., and Spindler, M. (2021), DoubleML - An Object-Oriented Implementation of Double Machine Learning in R, [arXiv:2103.09603](https://arxiv.org/abs/2103.09603).
- Bach, P., Chernozhukov, V., Kurz, M. S., and Spindler, M. (2022), DoubleML - An Object-Oriented Implementation of Double Machine Learning in Python, *Journal of Machine Learning Research*, 23(53): 1-6, <https://www.jmlr.org/papers/v23/21-0862.html>.

Double Machine Learning Approach

- Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W. and Robins, J. (2018), Double/debiased machine learning for treatment and structural parameters. The Econometrics Journal, 21: C1-C68, doi:10.1111/ectj.12097.
- Chernozhukov, V., Hansen, C., Kallus, N., Spindler, M., and Syrgkanis, V. (forthcoming), Applied Causal Inference Powered by ML and AI.

Appendix

DoubleML Class Structure

