

COMPREHENSIVE CAUSAL MACHINE LEARNING

Bern, March 2024



Michael Lechner & Jana Mareckova
Swiss Institute for Empirical Economic Research (SEW)
University of St. Gallen | Switzerland



Trend 1: Methods were developed to credibly estimate causal effects empirically

Trend 2: Effective prediction of newly developed Machine Learning methods

Causal Machine Learning

- ML inspired estimators applied to *well-identified* causal questions with **rich** data
- Promise: 'Better' estimation for causal effects at various aggregation level
 - Population effects, group effects, individualized effects
- Most work based on experiments & selection-on-observables (unconfoundedness)

Many fields active in methodological developments

- Universities: Statistics, Computer science, econometrics, epidemiology, marketing, ...
- Private: Tech industry & beyond

Applications in diverse empirical studies

General approaches to effect estimation

Parameter specific estimation methods

- Use 'best' method for each aggregation level (levels of granularity)
 - May work better for specific aggregation levels
 - Difficult to understand & monitor performance of different estimators
 - Many tuning parameters, ...
 - Internal consistency not guaranteed: Aggregates of heterogeneous effects may differ from average effects

Comprehensive Causal Machine Learning (CCML)

- Unified approach for estimating of all parameters of interest
- CCML estimators for different effects share important components (estimated by machine learning)

Comprehensive Causal Machine Learning | Econometrics

Double/debiased Machine Learning

- Common element: **D**oubly **R**obust **S**core

Generalized Random Forest

- Common element: Generalized Random Forest structure

Modified Causal Forest

- Common element: Causal Forest
- Aggregate effects obtained by aggregation of finer-grained effects



Double/debiased machine learning for treatment and structural parameters

VICTOR CHERNOZHUKOV[†], DENIS CHETVERIKOV[‡], MERT DEMIRER[†],
ESTHER DUFLO[†], CHRISTIAN HANSEN[§], WHITNEY NEWEY[†]
AND JAMES ROBINS^{||}

The Annals of Statistics
2019, Vol. 47, No. 2, 1148–1178
<https://doi.org/10.1214/18-AOS1709>
© Institute of Mathematical Statistics, 2019

GENERALIZED RANDOM FORESTS

BY SUSAN ATHEY*, JULIE TIBSHIRANI[†] AND STEFAN WAGER*

IZA DP No. 12040

Modified Causal Forests for Estimating Heterogeneous Causal Effects

Contribution

Compare point estimates & inference of *dml*, *grf*, *mcf* in a ***selection-on-observable (unconfoundedness)*** setting with respect to ...

- statistical guarantees (asymptotic properties)
 - Add some new statistical guarantees for the *mcf* (& *grf*)
- finite sample performance (Monte Carlo study)

Most important findings

- Efficiency – robustness trade-off for aggregate effects
 - Strength of selection into treatment is important for finite sample ranking of estimators
- Causal Forests superior (to DR-learner) for fine-grained heterogeneity

1 | Introduction

2 | Literature

3 | Notation, identification & effects at different levels of granularity

4 | Estimators under investigation

5 | Monte Carlo simulations

6 | Use effect heterogeneity to improve allocations

7 | Recommendations for empirical work

Notation | Binary treatment

D Treatment

Y^0 Potential outcome for $D=0$

Y^1 Potential outcome for $D=1$

$$Y = D Y^1 + (1-D) Y^0$$

X All confounders & heterogeneity variables

Z Specific heterogeneity variables (low dimensional; included in X)

Sample contains i.i.d. realisations of D, Y, X

Propensity score: $p(x) = P(D = 1 | X = x)$

Outcome regression: $\mu(d, x) = E(Y | D = d, X = x)$

Identification | Selection-on-observables | Unconfoundedness | CIA

Identifying assumptions

- D independent of Y^0, Y^1 given X
- Common support, no interactions between units, exogeneity of X

If assumptions are ...

- fulfilled → causal effects
- **not** fulfilled → associations adjusted for distributional differences in X

D : Treatment (0 or 1)

X : Confounder & heterogeneity variables

Y^1 : Outcome when $D = 1$

Z : Specific heterogeneity variables (low dim.)

Y^0 : Outcome when $D = 0$

Observable: D, Y, X ; $Z \subset X$

Estimands of mean causal effects at different aggregation levels | 1

Individualized (**C**onditional) **A**verage **T**reatment **E**ffect ($IATE(x)$, $CATE(x)$)

$$\begin{aligned} IATE(x) &= E(Y^1 - Y^0 \mid X = x) \\ &= E(Y \mid D = 1, X = x) - E(Y \mid D = 0, X = x) \\ &= \mu(1, x) - \mu(0, x) \end{aligned}$$

$$\mu(d, x) = E(Y \mid D = d, X = x)$$

Group **A**verage **T**reatment **E**ffect ($GATE(z)$, $CATE(z)$)

$$GATE(z) = E_{X|Z=z} IATE(X)$$

Average **T**reatment **E**ffect (ATE)

$$ATE = E_X IATE(X)$$

GATEs & ATEs on the treated / non-treated



1 | Introduction

2 | Literature

3 | Notation, identification & effects at different levels of granularity

4 | Estimators under investigation

5 | Monte Carlo simulations

6 | Software

7 | Recommendations for empirical work

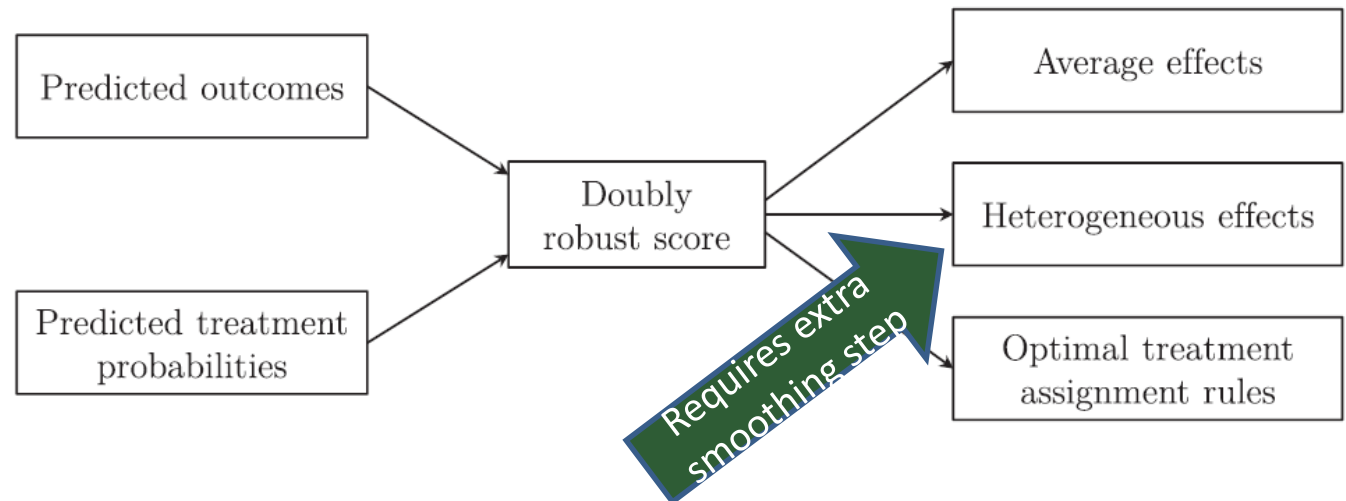
Double machine learning-based programme evaluation under unconfoundedness

MICHAEL C. KNAUS

DML for binary treatment

The central role of the
(cross-fitted)
Doubly Robust Score

$$DRS(y_i, d_i, x_i, \hat{\mu}_{-i}(\cdot), \hat{p}_{-i}(\cdot)) =$$
$$\hat{\mu}_{-i}(1, x_i) - \hat{\mu}_{-i}(0, x_i) + \frac{[y_i - \hat{\mu}_{-i}(1, x_i)]d_i}{\hat{p}_{-i}(x_i)} - \frac{[y_i - \hat{\mu}_{-i}(0, x_i)](1 - d_i)}{1 - \hat{p}_{-i}(x_i)}$$



DML for binary treatment | Averages of DR score

ATE
$$\widehat{ATE} = \frac{1}{N} \sum_{i=1}^N DRS(y_i, d_i, x_i, \hat{\mu}_{-i}(\cdot), \hat{p}_{-i}(\cdot))$$

GATE(z) (z-discrete)
$$\widehat{GATE}(z) = \frac{1}{N_z} \sum_{i=1}^N 1(z_i = z) DRS(y_i, d_i, x_i, \hat{\mu}_{-i}(\cdot), \hat{p}_{-i}(\cdot)) \quad N_z = \sum_{i=1}^N 1(z_i = z)$$

IATE(x) (DR-Learner)

- If $f(x)$ parametrically specified & estimation by OLS

$$\widehat{CATE}(x) = \arg \min_{f(x)} \frac{1}{N} \sum_{i=1}^N [DRS(y_i, d_i, x_i, \hat{\mu}_{-i}(\cdot), \hat{p}_{-i}(\cdot)) - f(x)]^2$$

- Drawback: Not robust to misspecification of $f(x)$
- If $f(x)$ is nonparametrically estimated, or by ML
 - Estimator is conjectured to be consistent

Basic principles of *Causal Forests* | 1

Homogeneous-in- X strata $\rightarrow$ no-selection-bias $\rightarrow$ difference of Y between treated & controls is 'good' estimator

Use Random-Forest-like splitting algorithm with adapted splitting rule

- Different Causal Forests use different splitting rules that try to approximate the MSE of the IATE
 - Maximise treatment effect heterogeneity: CF (Wager, Athey, 2018), **grf** (Athey, Tibshirani, Wager, 2019)
 - Approximate estimation errors directly: *mcf* (Lechner, 2018)

Honest estimation to avoid bias & getting inference

- Different data to build forest & to populate the final leaves with y_i

Generalized Random Forests | 1 | Estimation of IATE

Extension of Random Forest to nonparametric local GMM

Forest is used for obtaining weights for the local weighting $\rightarrow IATEs(x)$

- Weights obtained from gradient-based splitting rule $\rightarrow$ maximise treatment effect heterogeneity
- Honesty (instead of cross-fitting)

ATE & GATEs are obtained similarly to *dml*

- Main difference to *dml*: Doubly robust score directly obtained from estimated forest

Modified Causal Forest | Main ideas

The *mcf* is a modification of the Causal Forest by Wager & Athey (2018)

Important modification

- Splitting rule based alternative proxy for MSE & penalty function
- Representation of causal effect as weighted mean of y_i 's is used for inference
 - Requires different implementation of sample splitting
- *ATE* & *GATE*(z) are obtained by aggregating *IATE*(x)'s

Summary of theoretical properties

ATE

- All estimators are consistent, asymptotically normal
- *dml* & *grf* are efficient
- *mcf*: ATE is sum of IATEs

GATE

- All estimators are consistent, asymptotically normal (Z discrete)
- *mcf*: GATE is sum of IATEs in subsamples defined by values of Z

IATEs (finite dimensional X)

- *mcf* & *grf* are consistent, asymptotically normal
- *dml* (*DR-learner*) is consistent, asymptotically normal for parametric approximations (e.g., best linear predictors)

1 | Introduction

2 | Literature

3 | Notation, identification & effects at different levels of granularity

4 | Estimators under investigation

5 | Monte Carlo simulations

6 | Software

7 | Recommendations for empirical work

Goal | 1

Compare finite sample behaviour of *dml*, *grf*, *mcf*

- Quality of point estimates & inference procedures (out-of-training sample)

Synthetic Monte Carlo varying study with different data generating processes

- **Sample size:** $N=2'500$ & $10'000$
- **Covariates**
 - # of covariates in data: $p = 10, 20, 50$
 - # of covariates in DGP (sparsity): $k = 0.2 p, 0.5 p, 0.8 p$
 - Type of covariates: Normal, uniform, dummy (pure & mixtures of different types)
 - Z : Continuous covariate split into 5, 10, 20, 40 groups
- **Treatment**
 - # of treatments: 2, 4
 - Strength of selection ($R^2 = 0, 10\%, 42\%$)
 - Treatment shares: 25%, 50%
- **Potential outcomes**
 - Y^0 : Sine function of linear index, $R^2 = 0, 10\%, 45\%$
 - Y^1 : $Y^0 + \text{IATE} + \text{noise}$
 - IATE: 0, Linear, quadratic, logistic, step functions

Summary | 1

dml

- Dominates when # of groups is small & selectivity is not too large
- Performance may drastically deteriorate for a larger # of groups or stronger selectivity
- Not useful for IATEs

grf (pre-centred)

- Similar good than *dml* when # of groups is small & selectivity is not too large
- Performance drastically deteriorate for a larger # of groups (& somewhat for strong selectivity)
- Good performance for IATE

Summary | 2

mcf (pre-centred)

- Less efficient when # of groups is small & selectivity is not too large
- Much less affected by larger # of groups & stronger selectivity
- Overall robust (& competitive) behaviour

1 | Introduction

2 | Literature

3 | Notation, identification & effects at different levels of granularity

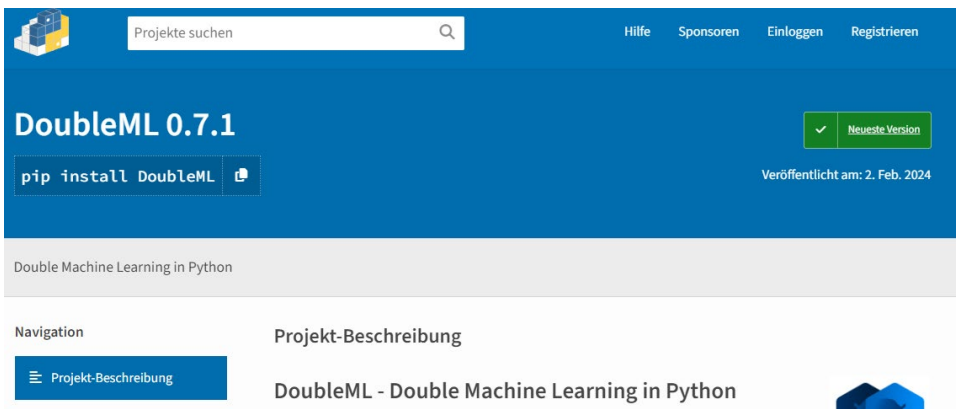
4 | Estimators under investigation

5 | Monte Carlo simulations

6 | Software

7 | Recommendations for empirical work

Some packages



DoubleML 0.7.1

pip install DoubleML

Neueste Version

Veröffentlicht am: 2. Feb. 2024

Double Machine Learning in Python

Navigation

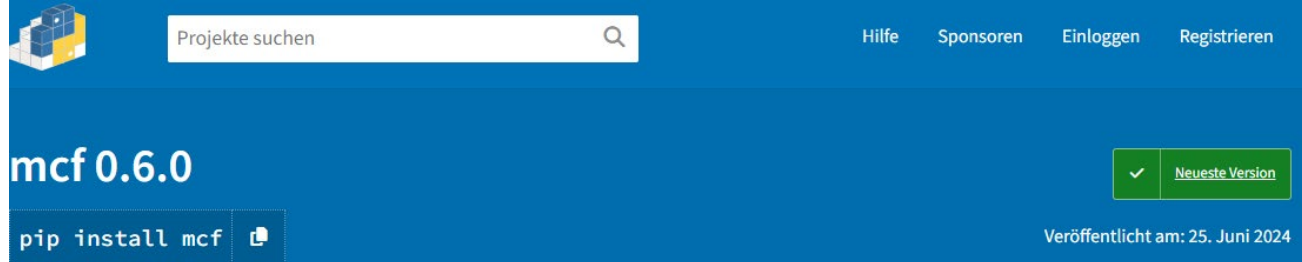
Projekt-Beschreibung

DoubleML - Double Machine Learning in Python

DoubleML: Double Machine Learning in R

Implementation of the double/debiased machine learning framework of Chernozhukov et al. 'DoubleML' allows estimation of the nuisance parts in these models by based on the 'R6' package is very flexible. More information available in the publication in

Version: 1.0.0
 Depends: R ($\geq 3.5.0$)
 Imports: R6 ($\geq 2.4.1$), data.table ($\geq 1.12.8$), stats, checkmate, mlr3 ($\geq 0.5.0$), m
 Suggests: knitr, rmarkdown, testthat, covr, patrick ($\geq 0.1.0$), paradox ($\geq 0.4.0$), d
 Published: 2024-02-15
 Author: Philipp Bach [aut, cre], Victor Chernozhukov [aut], Malte S. Kurz [au



econml 0.15.0

pip install econml

Neueste Version

Veröffentlicht am: 14. Feb. 2024

This package contains several methods for calculating Conditional Average Treatment Effects

grf: Generalized Random Forests

Forest-based statistical estimation and inference. GRF provides non-parametric methods for heterogeneous tree regression, and survival regression, all with support for missing covariates.

Version: 2.3.2
 Depends: R ($\geq 3.5.0$)
 Imports: DiceKriging, lmtest, Matrix, methods, Rcpp ($\geq 0.12.15$), sandwich ($\geq 2.4-0$)
 LinkingTo: Rcpp, RcppEigen
 Suggests: DiagrammeR, MASS, rdd, survival ($\geq 3.2-8$), testthat ($\geq 3.0.4$)
 Published: 2024-02-25
 Author: Julie Tibshirani [aut], Susan Athey [aut], Rina Friedberg [ctb], Vitor Hadad [ctb], David
 Maintainer: Erik Sverdrup <erik.sverdrup@monash.edu>

1 | Introduction

2 | Literature

3 | Effects at different levels of granularity

4 | Comprehensive CML estimators

5 | Monte Carlo simulations

6 | Software

7 | Recommendations for empirical work

Recommendations for empirical work

Large sample available (standard error reduction not a major concern)

- *mcf*

Smaller sample (standard error reduction is important)

- ATE, GATE (few groups): *dml* or *grf* (robustness check with more robust *mcf*)
- GATE (many groups): *mcf*
- IATEs: *grf* or *mcf*

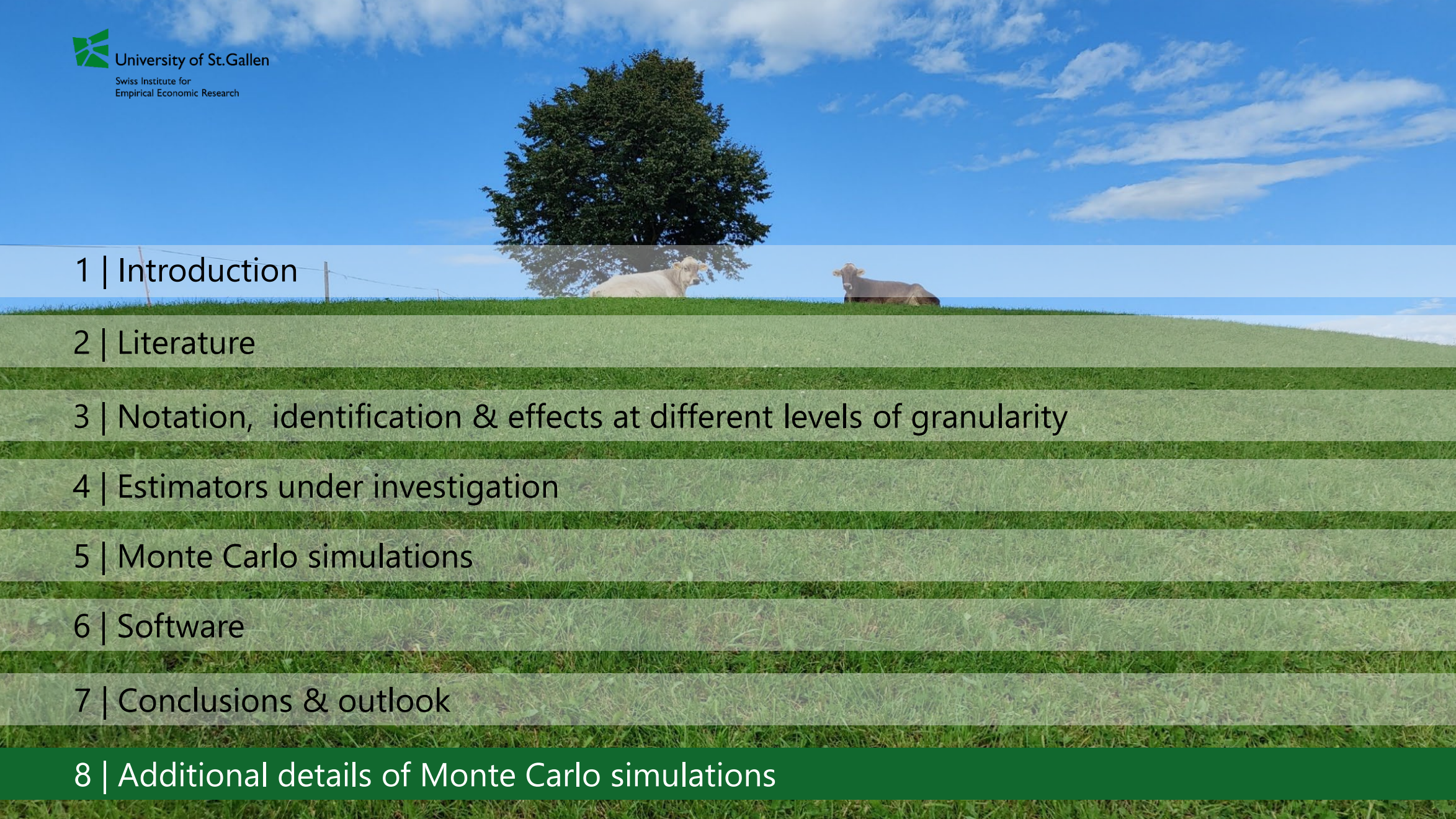
A note on computation

- User friendly packages available in *Python* & *R* (& Stata for *dml*)

The end. Thanks.

Michael Lechner

Swiss Institute for Empirical Economic Research (SEW)
University of St. Gallen | Switzerland



1 | Introduction

2 | Literature

3 | Notation, identification & effects at different levels of granularity

4 | Estimators under investigation

5 | Monte Carlo simulations

6 | Software

7 | Conclusions & outlook

8 | Additional details of Monte Carlo simulations

Implementation details of estimators in Monte Carlo study | *dml*

Nuisance parameter

- Regression Forests
- 5-fold cross-fitting

Normalized *dml*

- Normalise sum of (implied) weights to 1
- Truncate normalized weights if larger than 0.05 → renormalize

Implementation details of estimators in Monte Carlo study | *mcf*

Software

- Python package *mcf* (available on PyPI), most simulations based on version 0.3.3, some on 0.4.3

Tuning parameters

- Minimum leaf size of 5, 1000 trees, 50% subsamples

Other decisions

- Local centring ($E(Y|X=x)$) by Random Forest (scikit-learn) by 5-fold cross-fitting (in sample used to build Causal Forest)
- Nearest Neighbour Matching based on 1 (or 3 when 4 treatments) prognostic scores for MCE (weighting by Mahalanobis distance when 3 scores)
- Nonparametric estimations for weight-based inference use k-NN estimator ($k=2\sqrt{N}$)

Implementation details of estimators in Monte Carlo study | *grf*

Software

- R package *grf* (2.3.1)

Tuning parameters

- 2000 trees
- Minimum node size = 5
- Number of variables used for splitting drawn from $Poisson(\min(p, (p+20)^{0.5}))$
- Explicit centring with k-fold cross-fitting

Goal | 1

Comprehensive analysis of behaviour of *grf*, *dml*, *mcf* (& in different versions, OLS) in finite (but larger) samples

- Quality of point estimates & inference procedures

Computationally very demanding

- Many scenarios needed to see if some method breaks down in certain situations

General considerations | 1

Simulations (Econometrics) vs. *benchmark studies* (Machine learning)

- Difficult to use for inference (requires replications) → *simulations*

Empirical Monte Carlo Study (EMCS) vs. *synthetic design*

- EMCS somewhat more realistic for special case
- We want to vary very many characteristics → *synthetic design*
 - Type & quantity of covariates, # of treatments, effect heterogeneity, degree of selectivity, sample size, functional forms, influence of covariates on outcomes & heterogeneity, share of treated

Fixed or random features (X)

- Only fixed- X design allows to analyse distributions of $IATEs(x)$ for given x
- Random- X design better reflects distribution theory → *fixed X*

Simulation design | 1

Repeat R times

- Draw 2 data sets of size N
 - Data set 1 (to train forest): contains D, Y, X
 - Data set 2 (to predict effects): contains $IATE(x), X$
- Estimate effects & standard errors
- Save results

Compute performance measures

- 1st to 4th moments, bias, MSE, absolute error, bias of standard error, coverage probability

Data generating process (DGPs) | 1

Sample size: $N=2'500$ ($R=1'000$), $10'000$ ($R=250$)

- About constant simulation errors for ATE & GATE

Covariates

- # of covariates in data: $p = 10, 20, 50$
- # of covariates in DGP (sparsity): $k = 0.2 p, 0.5 p, 0.8 p$
- Type of covariates: Normal, uniform, dummy (pure & mixtures of different types)
- Z: Continuous covariate split into 5, 10, 20, 40 groups of equal size

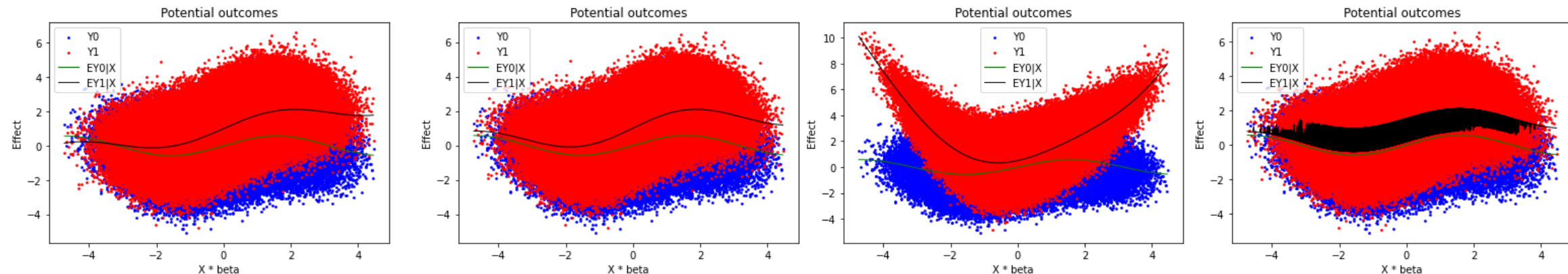
Treatment

- # of treatments: 2, 4
- Selection follows linear index model ($R^2 = 0, 10\%, 42\%$)
- Treatment shares: 25%, 50%

Data generating process (DGPs) | 2

Potential outcomes

- Y^0 : Sine function of linear index, $R^2 = 0, 10\%, 45\%$
- Y^1 : $Y^0 + \text{IATE} + \text{noise}$
- IATE: 0, linear & non-linear functions
 - Linear, quadratic, logistic function of linear index
 - Specification of Wager & Athey (2018): Complicated step-function



Estimators

dml (own Python code)

- Nuisances estimated by Regression Random Forests (*scikit-learn*)
- With and w/o normalized weights (robustness against extreme propensity scores)

grf (R package)

- Outcomes centred (k-fold cross-fitting) before start of estimation // no external centring
- *Not competitive without explicit centring of Y*

mcf (Python package)

- Outcomes centred (k-fold cross-fitting) before start of estimation // no external centring
- More efficient version without inference (IATEs only)
- *Not competitive without explicit centring of Y, D*

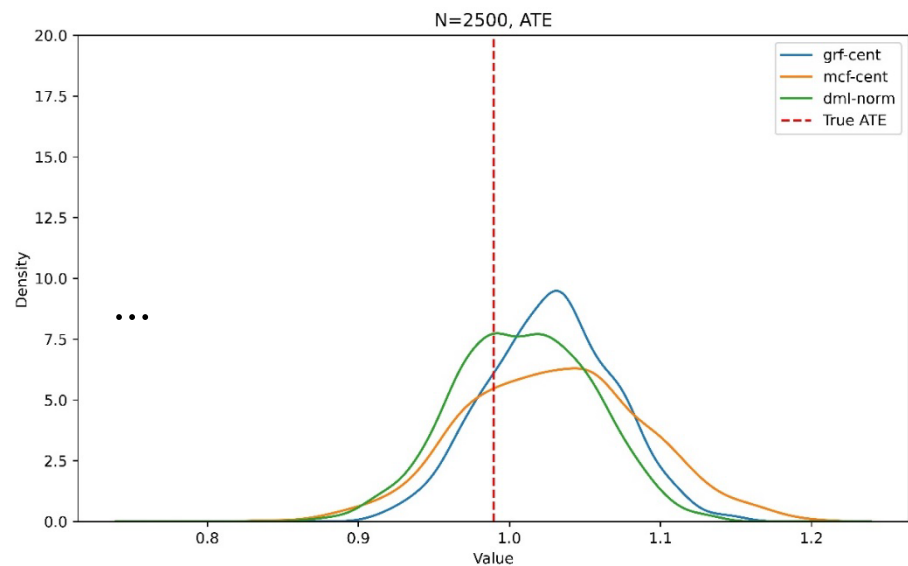
OLS

- *Not competitive*

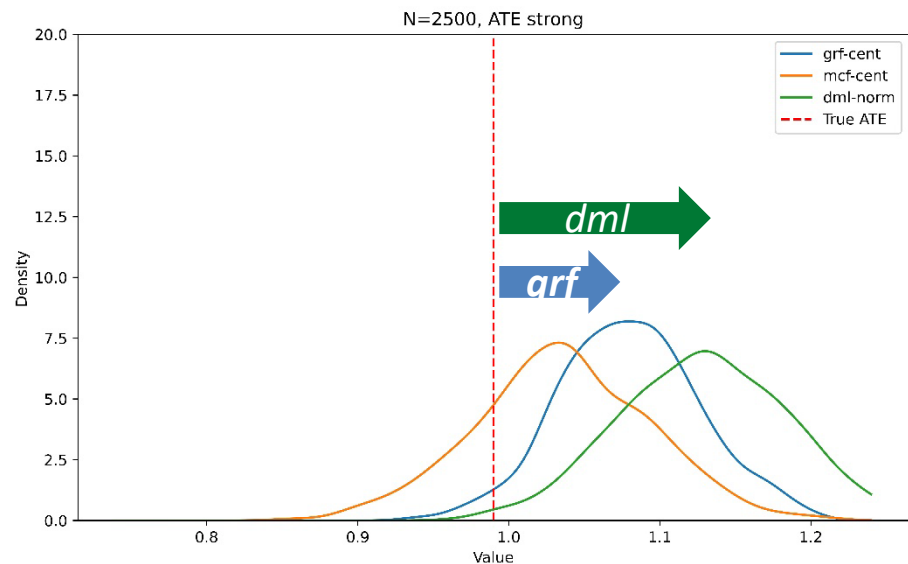
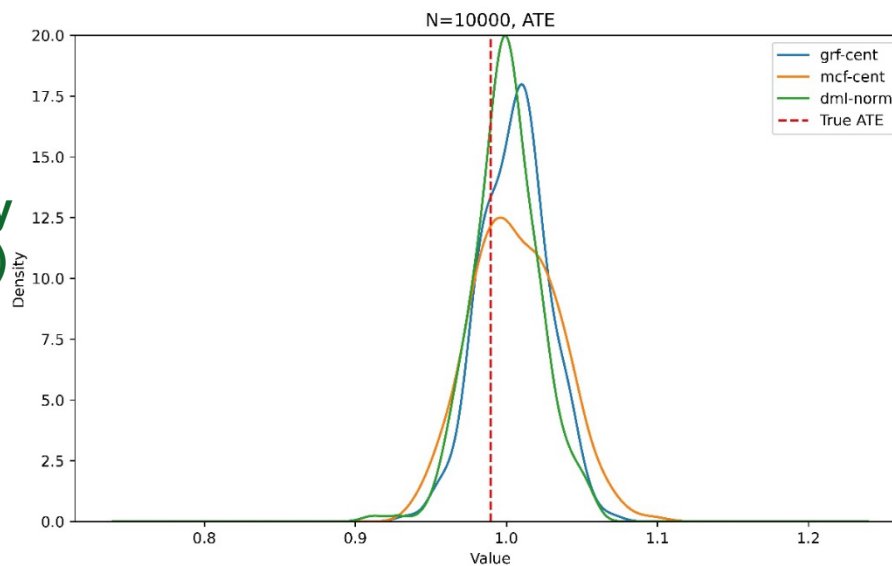
Important note

*All ML & CML estimators
use defaults
(no explicit tuning)*

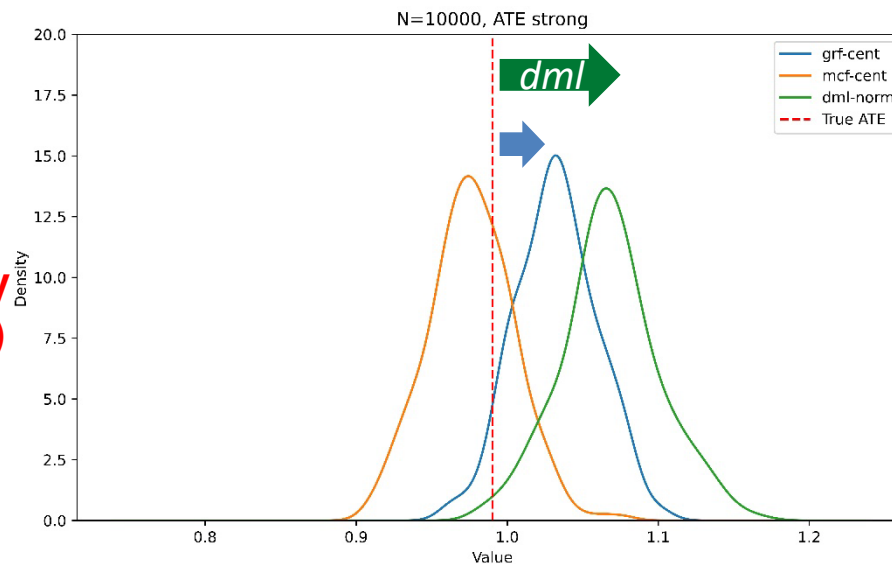
Results | ATE in main scenario (main scenario)



**Medium
selectivity
($R^2=10\%$)**



**Strong
selectivity
($R^2=42\%$)**



ATE

Selectivity

- General: Bias increases with increasing selectivity for all estimators (leading to worsening coverage)
- No & medium selectivity: *dml* & *grf* best, *mcfr* ok, coverage ok for $N=10'000$
- Strong selectivity: *dml* collapses, *grf* deteriorates, *mcfr* ok-ish
 - Conjecture: DR scores getting too extreme because of extreme propensity scores (more trimming needed?)

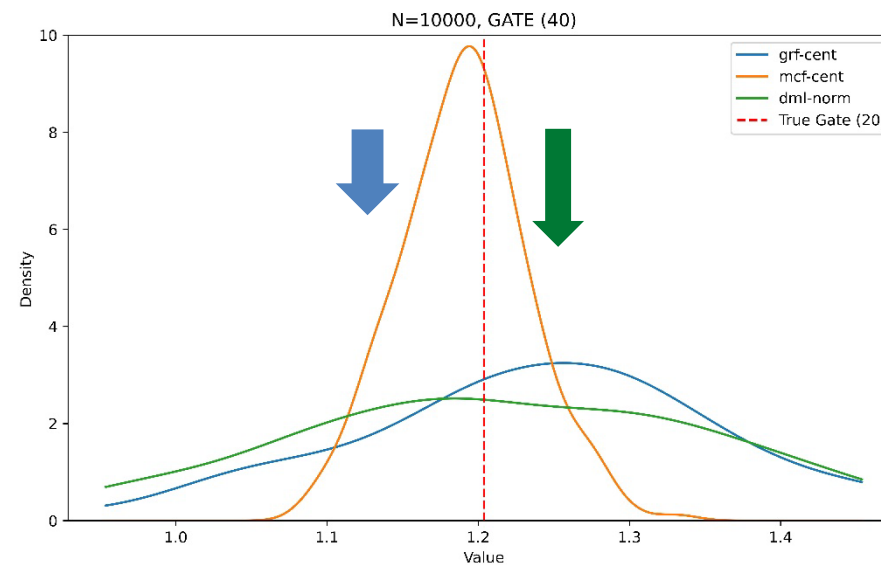
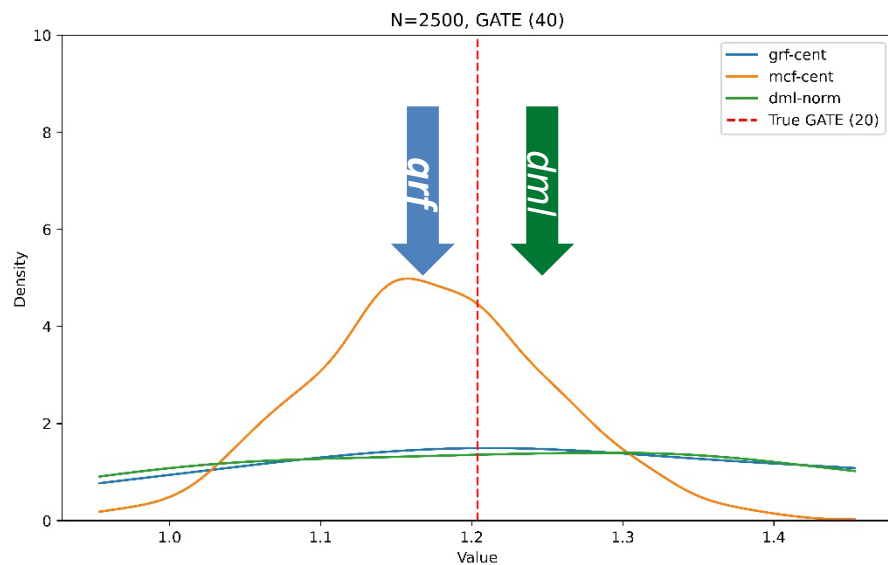
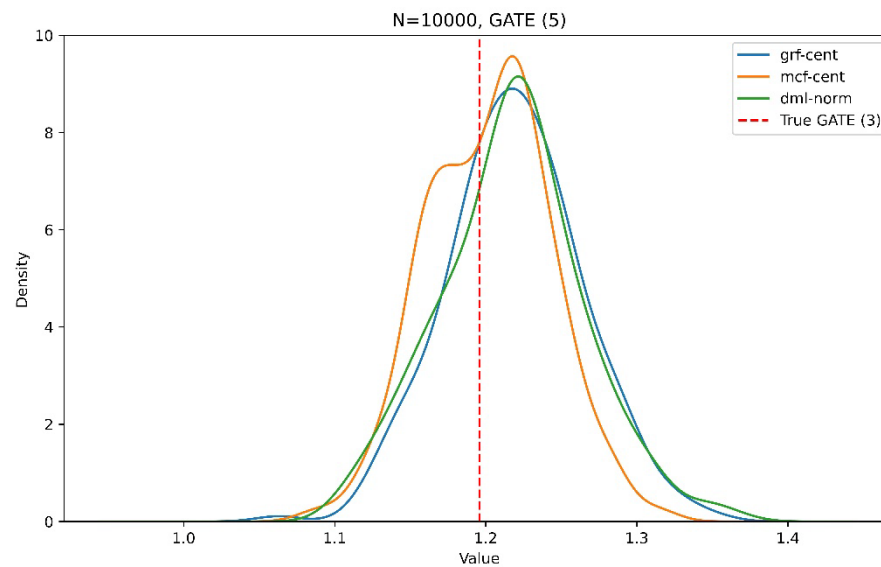
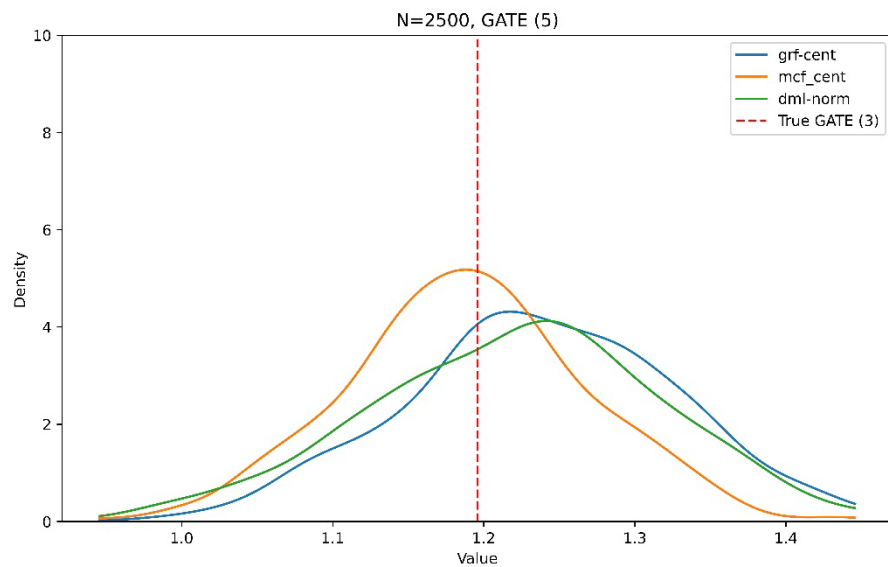
Sample size

- Speed of reduction of SE suggestive of $\sqrt{N}$ -convergence

Further observations

- All estimators deteriorate when R^2 of Y^0 is large

Results | GATE in main scenario for 5 & 40 groups



GATEs

5 groups: Generally same findings than for ATE

- *grf* almost always best estimator
- Coverage of *mcf* deteriorates (due to bias by cross-cell-of-Z smoothing)
- Strong selectivity: Relative performance loss of *dml* & *grf* less pronounced than for ATE

More groups: Performance of *dml* & *grf* deteriorates dramatically

- *dml* & *grf*: DR-scores regressed on up to 5, 10, 20, 40 dummies
 - Information from other values of *Z* only used in nuisance functions (no smoothing across *z*-values)
 - ➔ # of observation for each value of *z* crucially important
- *mcf*: Aggregates of IATE(*x*) in cells by *z*-value
 - IATEs smooths across *z*-values ➔ # of observations for each value of *z* less relevant

IATEs

grf & *mcf* **clearly** outperform *dml* (OLS & Random Forest based)

- Even true when $IATE(x)$ is 0, or a linear function of x (when there is selectivity)
- Only exception: No selectivity together with not much nonlinearity in IATE

Coverage is bad for all methods (worst for *dml*)

- If inference is not needed, *grf* & *mcf* can be improved
 - Problem seems to come from too much bias in point estimates

Summary | 1

dml

- Dominates when # of groups is small & selectivity is not too large
- Performance may drastically deteriorate for a larger # of groups or stronger selectivity
- Not useful for IATEs

grf (pre-centred)

- Similar good than *dml* when # of groups is small & selectivity is not too large
- Performance may drastically deteriorate for a larger # of groups (& to a lesser extend for strong selectivity)
- Good performance for IATE

Summary | 2

mcf (pre-centred)

- Less efficient when # of groups is small & selectivity is not too large
- Much less affected by larger # of groups & stronger selectivity
- Overall robust (& competitive) behaviour