An Empirical Investigation of Inference Techniques with Statistical Learning Models

Candidate number: 3190F

Abstract

Recently, econometricians started leveraging the strengths of machine learning to answer causal questions. We investigate the finite sample performance of causal machine learning estimators for heterogeneous treatment effects, by carrying out an empirical Monte Carlo study with 9 different data-generating processes. In our simulations, 4 estimators consistently outperform the rest (these are the R-learner, Causal Forest, ps-BART and Bayesian Causal Forest). These estimators are based on flexibly modelling the response surface via tree-based algorithms. Using these estimators, we estimate the individual-level treatment effects of unemployment training programs and uncover effect heterogeneity along demographic and socioeconomic characteristics.

Word count: 7500 words

1 Introduction

Quantifying the treatment effects of an intervention is of great interest in applied economics, medicine, political science, sociology, education research and marketing, among other fields. For instance, evaluating the average treatment effect (ATE) of unemployment training programs can inform which programs should continue to receive funding. Additionally, the causal parameter can vary with individual characteristics. Investigating treatment heterogeneity is insightful in and of itself, but is also of high practical relevance: the policymaker should allocate individuals to the treatments that are most effective, conditional on the participant's characteristics.

The main estimation challenge is that we have no access to the ground truth causal parameter: for each data point, we are only observing the outcome under the received treatment, and have no access to the counterfactual. In observational data with selection into treatment, some vector of covariates x may influence both the outcome and the treatment assignment probability, creating the "confoundedness problem". Identification may be possible under the assumption of unconfoundedness *conditional on observed covariates*.

The proliferation of large datasets with many variables offers new opportunities to control for confounding, but also creates new challenges in terms of how to do so. In the past, researchers have relied on ad hoc decisions in terms of variable selection and assuming that the relationship between the covariates and the outcome/assignment could be captured via a linear functional form. In contrast, machine learning (ML)[1] approaches can automate many of these choices, algorithmically selecting the relevant variables and functional forms. This can reduce the bias stemming from inadequate control for covariates.

In a rapidly emerging literature, several researchers have proposed methods that

1. A more apt name would probably be "statistical learning", but we follow the convention.

utilise the strengths of ML to answer causal inference questions on individual-specific treatment effects. However, practitioners still lack adequate guidance as to which methods are expected to perform well on a given dataset. Our paper addresses this gap in the literature.

1.1 Contributions

Our primary contribution is carrying out an empirical Monte Carlo study (EMCS), where we evaluate several methods in a realistic data setting. The EMCS methodology allows for varying parameters of the data-generating process (DGP), therefore providing insights as to which scenarios confuse the estimators. We find that a group of 4 methods consistently provides the best performance across our specifications; these estimators have in common that they are specifically designed for heterogenous treatment effect estimation and can flexibly model the response surface (by building an ensemble of regression trees).

The secondary contribution is an empirical application, where we use the best-performing methods to estimate ATE and conditional average treatment effects (CATE) on a standard dataset of the Swiss Active Labour Market Policy. Our methods are more robust, and allow for more granular estimation, than any previous research using this dataset (that we are aware of). Our findings are in line with the existing literature.

1.2 Reproducibility

In order to facilitate reproducibility, we created a GitHub user *anondissertation* and uploaded all code used to generate the results into its depository. We refer to this depository as the Online Appendix. Due to space limitations, some results have been relegated there.

1.3 Structure

Section 2 reviews the existing literature and the place of this research within it. Section 3 explains the methodology, including the estimands of interest, identification assumptions, the estimators studied, the concept of an EMCS and the evaluation criteria. Section 4 introduces the dataset with which we perform the EMCS and the data generating processes (DGPs) used to simulate treatment effects. Section 5 presents and discusses the EMCS results. In Section 6 we use the best-performing methods to estimate treatment effects in the original data. Section 7 concludes.

2 Literature review

Economists and others are interested in causal effects of policies and interventions. The literature on average treatment effect estimation, and in particular for labour market applications, goes back to at least Ashenfelter (1978). This literature is very mature by now. An excellent overview is provided by Imbens and Wooldridge (2009). In the following, we focus on CATE estimation.

In most applications, it is interesting to look beyond the average effect and understand how causal effects vary with observable characteristics. For example, medical treatments should be tailored to individuals based on heterogeneity of clinical characteristics and the unemployed should be assigned to training programs that yield the highest expected benefit, given their circumstances. In recent years, researchers have proposed many estimators that target heterogenous treatment effects. We categorise these methods into three groups: generic, specific and Bayesian approaches.

2.1 Generic approaches

Generic approaches (also called meta-algorithms) split the causal estimation problem into several standard prediction problems. The name of this group comes from the fact that any estimator can be used to estimate these "first-stage" nuisance parameters (usually outcome regressions and the propensity score). Popular choices for the first-stage estimator are OLS, Lasso (Tibshirani 1996), Random Forests (Breiman 2001), boosting (e.g. Freund, Schapire, et al. 1996), SVMs (Vapnik 1996) and Neural Networks (McCulloch and Pitts 1943). For an overview of these and other estimators, see the textbook Hastie et al. (2009). Knaus et al. (2021) provide a list of theoretical and and empirical papers within economics which make use of these estimators in a causal inference context.

The DR MOM, T-, S-, X-, F-, U-, and R-learning approaches fall into this category. In order to avoid repetition, we introduce these estimators and provide references in Section 3.3.

Powers et al. (2018) contains a discussion and comparison of many generic methods and their modifications.

2.2 Specific approaches

Specific methods modify a given machine learning algorithm to target CATEs. By far the most famous example in this category is the Causal Forest (CF), which is a special case of the Generalised Random Forest of Athey et al. (2019). CF is the first algorithm that uses Random Forests for provably valid statistical inference, with the properties derived by Wager and Athey (2018).

Modifications of the Causal Forest are CF with Local Centering (Athey et al. 2019), and Penalised CF (Lechner 2018).

Other work we are aware of is by Johansson et al. (2016), Shalit et al. (2017) and Schwab et al. (2018) who develop Neural Network-based CATE estimators and Zaidi and Mukherjee (2018), who use a Gaussian Mixture Model.[2]

2.3 Bayesian approaches

Chipman et al. (2010) proposed Bayesian Additive Regression Trees (BART) as a Bayesian ensemble regression method. In a sense, BART is just another supervised machine learning algorithm, with the prior parameters considered as hyperparameters, but has two key advantages. Firstly, it gives good performance with a wide range of hyperparameters. Secondly, the Bayesian framework makes it suitable as an inferential model through posterior sampling. Hill (2011) was the first to apply BART to CATE estimation. Overviews of base BART and its recent developments for causal inference are provided by O'Neill (2019) and Hill et al. (2020).

Hahn et al. (2020) propose the Bayesian Causal Forest (BCF), which improves upon base BART in two ways. Firstly, it tackles regularisation-induced confounding (Hahn et al. 2018) by incorporating an estimate of the propensity score. Secondly, they model the outcome as the sum of a prognostic impact and the treatment effect. By explicitly separating the treatment effect, its level of heterogeneity can be regularised and better estimates can be obtained when most outcome variation is due to the prognostic impact rather than treatment effect heterogeneity.

2.4 Empirical evaluation

Given the abundance of available estimators, and the lack of theoretical results, comparing and contrasting the above methods on real and synthetic data can provide valuable information to empiricists. Although some papers have taken up this topic, we believe that still there exists a gap in the literature. The following reviews

2. We do not include these estimators in our study because of software availability and computational time issues.

the existing literature and discusses our improvements.

Since 2016, the Atlantic Causal Inference Conference (ACIC) has a data analysis challenge, where researchers can submit their methods to be neutrally evaluated on (partly) synthetic datasets. The results of the 2016 edition are discussed by Dorie et al. (2019) and for 2017 by Hahn et al. (2019).[3] The main conclusion from these competitions is that methods that can flexibly model the response surface, in particular BART, have the best performance. Wendling et al. (2018) carry out a study using (observational) medical datasets with a binary outcome. They find that BART and causal boosting consistently provide low bias. A similar study, with more estimators, but for ATE, is McConnell and Lindner (2019). The main conclusion is that machine learning-based estimators outperform traditional ones. The most similar paper to ours is Knaus et al. (2021), and we use the same base dataset (see Section 4).

Compared to the papers coming out of ACIC, our study includes estimators that were non-existent in 2016 and 2017. Similarly, Wendling et al. (2018) only considered 4 estimators. Hence our study is more comprehensive on this dimension. Wendling et al. (2018) and McConnell and Lindner (2019) use data structures that are not common in economics, and the latter only consider ATE estimation, making their results not directly relevant to our topic.

While we take many ideas from Knaus et al. (2021), we identify and address some shortcomings. Firstly, they only consider generic approaches and CF. We incorporate BART and BCF, which have shown excellent performance in other simulations. Secondly, the DGP they chose for the treatment effect was peculiar (a strongly non-linear function of the propensity score). While it made the estimation task challenging, such a DGP would be hard to imagine for real data. We address this issue by 1) using an arguably more realistic benchmark DGP; and 2) varying the

3. Sadly, the results for 2018-2020 are not yet published.

parameters of the DGP in 8 different ways, to see which changes affect the estimators.

Finally, we use the best-performing methods to estimate the individualised treatment effects of Swiss unemployment training programs. To our knowledge, the only published paper attempting to do so is by Knaus (2020), who uses the DR MOM estimator. DR MOM is not among the best estimators in our simulations, and Knaus (2020, p. 24) note that the their estimated effect dispersion is implausibly high and hence restrict the set of covariates considered as effect moderators. Our estimates seem more robust, without restricting covariates, using 4 different estimators.

3 Methodology

3.1 Estimands of interest

We define the estimands of interest in the single treatment version of the potential outcomes framework with categorical treatment (Rubin 1974). We observe a dataset $\{(x_i, d_i, y_i)\}_{i \in I}$, where $I \equiv \{1, \ldots, N\}$. Here $x_i \in \mathbb{R}^d$ is the vector of covariates, the dummy variable $d_i \in \{0, 1\}$ indicates whether individual i received the treatment and $y_i \in \mathbb{R}$ is the outcome. (x_i, d_i, y_i) is a realisation of the random variable (X_i, D_i, Y_i), which we assume to be IID over i. We assume that each individual i has a potential outcome Y_i^0 if not treated ($D_i = 0$) and Y_i^1 if treated ($D_i = 1$). This means that, for each i, only one of the two potential outcomes is observable:

$$Y_i = D_i Y_i^1 + (1 - D_i) Y_i^0. \tag{1}$$

The *individual* treatment effect (ITE), $\xi_i = Y_i^1 - Y_i^0$ of D_i on Y_i is never observed.

The observation rule in (1) is sometimes referred to as the stable unit treatment value assumption (SUTVA) and rules out spillover effects between individuals.

Our focus is to estimate treatment effects conditional on observable characteristics. The most granular conditioning level is the *conditional* average treatment effect (CATE),

$$\tau(x) = \mathbb{E}(\xi_i \mid X_i = x) = \mu_1(x) - \mu_0(x), \tag{2}$$

which may depend on all available covariates. Here $\mu_d(x) = \mathbb{E}(Y_i^d \mid X_i = x)$ denotes the conditional expectation of the (unobserved) potential outcomes, given treatment $d \in \{0, 1\}$. CATEs are an approximation of ITEs, based on the the set of available covariates.

CATEs can be aggregated into *group* average treatment effects (GATE),

$$\tau(g) = \mathbb{E}(\xi_i \mid G_i = g) = \int \tau(x) f_{x_i|G_i=g}(x) dx, \tag{3}$$

where $\mathcal{G}$ is some pre-defined characteristic partitioning the sample (e.g. $\mathcal{G} = \{male, female\}$) and $G_i \in \mathcal{G}$ is the partition i belongs to. This is often useful when the estimated CATEs need to be summarised, communicated or acted upon, or the research interest is in group heterogeneities (e.g. gender differences).

The highest aggregation level is the average treatment effect (ATE), $\tau = \mathbb{E}(\xi_i)$ or the average treatment effect *for the treated* (ATET), $\mathbb{E}(\xi_i \mid D_i = 1)$.

To keep the number of results manageable, we focus on CATEs and ATEs.

3.2 Identification

In observational studies, the identification of treatment effects is complicated by non-random treatment assignment. That is, x_i contains the union of confounder and heterogeneity variables (which may be completely, partly, or non-overlapping) and the researcher has no knowledge of which variables in x_i belong to each group.

However, the following assumptions are sufficient for the identification of CATEs.

Assumption 1 (Conditional independence): $Y_i^0, Y_i^1 \perp D_i \mid X_i = x$, for all x in the support of X_i.

Assumption 2 (Common support): $0 < \mathbb{P}(D_i = 1 \mid X_i = x) < 1$, for all x in the support of X_i.

Assumption 3 (Exogeneity of covariates): $X_i^1 = X_i^0$. [4]

The conditional independence (also called unconfoundedness, ignorability and selection on observables) assumption states that, after conditioning on observed covariates, treatment assignment is independent of potential outcomes. Common support (also called full overlap, positivity and matching assumption) ensures the existence of a hypothetical counterfactual for each treatment assignment. Exogeneity of covariates requires that the covariates are not affected by the treatment.

Under SUTVA and Assumptions 1-3,

$$\mathbb{E}(Y_i^d \mid X_i = x, D_i = 1 - d) = \mathbb{E}(Y_i \mid X_i = x, D_i = d) \equiv \mu(d, x)$$

$$\implies \tau(x) = \mu(1, x) - \mu(0, x)$$

and therefore CATEs and the ATE are identifiable from observable data, and in particular we can estimate them via conditional expectation functions. $\mu(d, x)$ is the conditional expectation of the outcome, given the treatment $d \in \{0, 1\}$. Note that $\mu_d(x)$ and $\mu(d, x)$ are different objects, and we need the above assumptions to have $\mu(d, x) = \mu_d(x)$.

Furthermore, we denote by $m(x) = \mathbb{E}(Y_i \mid X_i = x)$ the conditional expectation of the outcome marginalised over treatment and by $p(x) = \mathbb{P}(D_i = 1 \mid X_i = x)$ the

4. The potential covariates X_i^d are defined equivalently to potential outcomes.

conditional treatment probability, called the propensity score.

3.3 Estimators

This subsection presents the estimators used to obtain ATE and CATE estimates. The word limit prevents in-depth explanations, therefore we only provide a quick overview and references for further reading. The implementation details (hyperparameters and software) are discussed on Appendix B. In the algorithms presented below, the notation $M(y \sim x)$ stands for estimating $x_i \mapsto \mathbb{E}(Y_i \mid X_i = x_i)$ with a regression method (supervised ML algorithm) M.

3.3.1 T-learner

As we saw in Section 3.2, under our assumptions, CATE is identified as the difference of two conditional mean functions, which suggests the following method:

Algorithm 1: T-learner

1 $\hat{\mu}_0 = M_0(y^0 \sim x^0)$
2 $\hat{\mu}_1 = M_1(y^1 \sim x^1)$
3 $\hat{\tau}_T(x) = \hat{\mu}_1(x) - \hat{\mu}_0(x)$

where $y^0 \sim x^0$ is notation for regressing y^0 on x in the non-treated sample, and similarly $y^1 \sim x^1$ uses only the treated observations. The algorithms M_0 and M_1 may be different.

The T-learner is also called Conditional Mean Regression. A similar approach is the S-learner, which we do not study[5] because 1) it usually does not give good performance 2) it requires the estimation of a separate nuisance function $\mathbb{E}(Y_i \mid X_i = x_i, D_i = d_i)$, hence extra computational cost.

5. Apart from BART, which is essentially implemented as an S-learner.

Algorithm 2: S-learner

1 $\hat{\mu} = M(y \sim (d, x))$
2 $\hat{\tau}_S(x) = \hat{\mu}(1, x) - \hat{\mu}(0, x)$

To motivate the use of the following estimators, we highlight the weaknesses of the T-learner approach. Minimising the mean squared error (MSE) in two separate prediction problems in not tailored to estimate CATEs. Firstly, the regression methods M_0 and M_1 are regularised towards 0 in order to avoid over-fitting. However, this regularisation might move the estimate $\hat{\mu}_1(x) - \hat{\mu}_0(x)$ *away* from 0, even if the true CATE is 0. Secondly, if the predictions at a given x are biased in the same direction, it is less harmful than biases in opposing directions. Lechner (2018) contains a discussion of this in the context of Causal Forests.

3.3.2 X-learner

The intuition behind the X-learner of Künzel et al. (2019) is to use information from the control group to derive better estimators for the treatment group and vice versa. The main idea is to compute imputed treatment effects based on estimated response functions (steps 3-4 below). It has shown good performance in simulations, especially when the number of treated observations was small.

Algorithm 3: X-learner

1 $\hat{\mu}_0 = M_0(y^0 \sim x^0)$
2 $\hat{\mu}_1 = M_1(y^1 \sim x^1)$
3 $\hat{\xi}^1 = y^1 - \hat{\mu}_0(x^1)$
4 $\hat{\xi}^0 = \hat{\mu}_1(x^0) - y^0$
5 $\hat{\tau}_1 = M_2(\hat{\xi}^1 \sim x^1)$
6 $\hat{\tau}_0 = M_3(\hat{\xi}^0 \sim x^0)$
7 $\hat{\tau}_X(x) = g(x)\hat{\tau}_0(x) + (1 - g(x))\hat{\tau}_1(x)$

In Algorithm 3, $g(x)$ is a weighting function that aims to minimise the variance of $\hat{\tau}(x)$. Sometimes it is possible to estimate $\text{Cov}(\hat{\tau}_0(x), \hat{\tau}_1(x))$, but usually $g(x) = \hat{p}(x)$, an estimate of the propensity score, is chosen.

3.3.3 Modified Outcome Methods

Abadie (2005) introduced the idea that more efficient estimators may be obtained by modifying the outcome. The F-learner, U-learner and DR MOM are in the category of Modified Outcome Methods (MOM).

The F-learner is based on inverse probability weighting (hence also called the IPW estimator), which was studied by Hirano et al. (2003) for ATE estimation.

Algorithm 4: F-learner

1 $\hat{p} = M_0(d \sim x)$
2 $y_i^F = y_i \frac{d_i - \hat{p}(x_i)}{\hat{p}(x_i)(1-\hat{p}(x_i))}$
3 $\hat{\tau}_F = M_1(y^F \sim x)$

where $M_0(d \sim x)$ denotes using a classifier algorithm M_0 to estimate propensity scores. The intuition behind IPW is to inflate the weight of "under-represented" observations, as they contain more counterfactual information.

Algorithm 5: U-learner

1 $\hat{m} = M_0(y \sim x)$
2 $\hat{p} = M_1(d \sim x)$
3 $y_i^U = \frac{y_i - \hat{m}(x_i)}{d_i - \hat{p}(x_i)}$
4 $\hat{\tau}_U = M_2(y^U \sim x)$

The DR MOM estimator is based on the Doubly Robust (DR) estimator of Robins and Rotnitzky (1995). The asymptotic properties of this estimator for ATE are well understood (Belloni et al. 2017; Chernozhukov et al. 2018) but we are not aware of any theoretical results for CATE.

The MOM estimators come from the observation that

$$\tau(x) = \mathbb{E}(Y_i^F \mid X_i = x) = \mathbb{E}(Y_i^U \mid X_i = x) = \mathbb{E}(Y_i^{DR} \mid X_i = x).$$

Algorithm 6: DR MOM

1 $\hat{\mu}_0 = M_0(y^0 \sim x^0)$
2 $\hat{\mu}_1 = M_1(y^1 \sim x^1)$
3 $\hat{p} = M_2(d \sim x)$
4 $y_i^{DR} = \hat{\mu}_1(x_i) - \hat{\mu}_0(x_i) + \frac{d_i(y_i - \hat{\mu}_1(x_i))}{\hat{p}(x_i)} - \frac{(1-d_i)(y_i - \hat{\mu}_0(x_i))}{1-\hat{p}(x_i)}$
5 $\hat{\tau}_{DR} = M_3(y^{DR} \sim x)$

3.3.4 DML

The ATE estimator derived from the DR MOM scores is called the Double/Debiased Machine Learning (DML) estimator, following Chernozhukov et al. (2018). In general form, a low-dimensional causal parameter θ is estimated as the solution to

$$\frac{1}{N}\sum_{i=1}^{N}\psi(w_i;\theta,\hat{\eta}) = 0, \tag{4}$$

where $w_i = (x_i, d_i, y_i)$ is the data, $\hat{\eta}$ is the estimated nuisance functions and ψ is a Neyman-orthogonal score function. In our case, the Interactive Regression Model (IRM) applies[6] so the estimand is the ATE, the nuisance functions are $\hat{\eta}(d,x) = (\hat{\mu}(d,x), \hat{p}(x))$ and the score function is

$$\psi(w;\theta,\hat{\eta}) = \hat{\mu}(1,x) - \hat{\mu}(0,x) + \frac{d(y-\hat{\mu}(1,x))}{\hat{p}(x)} - \frac{(1-d)(y-\hat{\mu}(0,x))}{1-\hat{p}(x)} - \theta.$$

Crucially, the nuisance parameters must be estimated with cross-fitting (Section 3.4.4). Chernozhukov et al. (2018) prove that, under fairly general conditions, the DML estimates are root-N consistent, approximately unbiased and asymptotically normally distributed.

6. The IRM model is

$$\begin{aligned} Y &= \mu(D,X) + U, & \mathbb{E}(U \mid X, D) &= 0 \\ D &= p(X) + V, & \mathbb{E}(V \mid X) &= 0. \end{aligned}$$

3.3.5 R-learner

The R-learner is based on the observation that

$$Y_i - m(X_i) = (D_i - p(X_i))\tau(X_i) + U_i(D_i) \tag{5}$$

where $U_i(d) = Y_i^d - (\mu_0(X_i) + d\tau(X_i))$.

$\mathbb{E}(U_i(D_i) \mid X_i, D_i) = 0$, hence Equation 5 can be equivalently expressed as

$$\tau(.) = \underset{\tau}{\operatorname{argmin}} \left\{ \mathbb{E}\left[(Y_i - m(X_i) - (D_i - p(X_i))\tau(X_i))^2 \right] \right\} \tag{6}$$

which suggests estimating $\tau(.)$ as

$$\hat{\tau}(.) = \underset{\tau}{\operatorname{argmin}} \left\{ \frac{1}{N} \sum_{i=1}^{N} \left[y_i - \hat{m}(x_i) - (d_i - \hat{p}(x_i))\tau(x_i) \right]^2 + \Lambda_N(\tau(.)) \right\} \tag{7}$$

where $\Lambda(\tau(.))$ is a reguliser term on the complexity of the $\tau(.)$ function.

Our implementation replaces $\tau(.)$ with a LASSO working model, which yields the following estimation procedure:

Algorithm 7: R-learner

1 $\hat{m} = M_0(y \sim x)$
2 $\hat{p} = M_1(d \sim x)$
3 $\hat{\beta}_R = \operatorname{argmin}_\beta \left\{ \frac{1}{N} \sum_{i=1}^{N} \left[Y_i - \hat{m}(x_i) - (d_i - \hat{p}(x_i)) x_i \beta \right]^2 + \lambda ||\beta||_1 \right\}$
4 $\hat{\tau}_R(x) = x\hat{\beta}_R$

The R-learner was studied by Nie and Wager (2017), who establish error bounds and show that (under some assumptions) the R-learner can perform as good as an oracle estimator that knows the real nuisance parameters.

3.3.6 Causal Forest

The Causal Forest (CF), which is a special case of the Generalised Random Forest of Athey et al. (2019), minimises the R-loss in Equation 7 by growing a (modified) Random Forest (RF).

Alternatively, for binary treatments, we can write the CF point estimates as the difference of two weighted means

$$\hat{\tau}_{CF}(x) = \sum_{i=1}^{N} D_i w_i^1(x) Y_i - \sum_{i=1}^{N} (1 - D_i) w_i^0(x) Y_i,$$

where the weights $w_i^d(x)$ define an adaptive local neighbourhood around the covariate x. While standard RFs split the sample according to observed outcomes, CF splits the sample along the gradient of the mean difference with the pseudo outcomes

$$\rho_i = (D_i - \bar{D}_P)(Y_i - \bar{Y}_P - (D_i - \bar{D}_P)\hat{\beta}_P)/\text{Var}_P(D_i),$$

where $\bar{D}_P$ and $\bar{Y}_P$ are averages of the treatment indicator and outcomes, $\hat{\beta}_P$ is the mean difference and $\text{Var}_P(D_i)$ is the variance of the treatment in the parent node P. Furthermore, *honest splitting* is applied, i.e. for each i, the algorithm only uses y_i to estimate the within-leaf treatment effect or to decide where to place the splits, but not both.

These modifications allow the derivation of theoretical results for CATE estimation, namely consistency and asymptotic normality for a fixed covariate space (Athey et al. 2019). To our knowledge, CF is the only CATE estimator with provably valid confidence intervals, to date.

3.3.7 BART

BART is also a sum-of-trees predictive model. However, unlike a Random Forest and similar to Boosting, each tree estimates (the remaining part of) $\mathbb{E}(Y|X, D)$ and subsequent trees are fit on the residuals. Each tree is constrained by a regularisation prior to be a weak-learner. BART has shown good predictive performance on many tasks with a wide range of hyperparameters.

The BART estimates for $\tau(x)$ are $\hat{\tau}_{BART}(x) = \hat{\mu}(1, x) - \hat{\mu}(0, x)$, so our implementation is basically an S-learner. The advantage over other ML methods is that BART generates a posterior distribution, so we can calculate confidence intervals by sampling from it. In causal settings, including an estimate of the propensity score as one of the covariates improved accuracy in simulations and allows the derivation of consistency results in linear and partially linear semiparametric settings, since it prevents regularisation-induced confounding (Hahn et al. 2018). Hence we follow this practice.[7] This estimator is called ps-BART in the literature.

3.3.8 Bayesian Causal Forest

BCF is a re-parameterisation of ps-BART to target causal effects. The conditional expectation function $\mu(d, x)$ is written as $\mu(d, x) = f(\hat{p}(x), x) + \tau(x)d$ where f and τ have independent BART priors. By separating the prognostic function f from the estimand of interest, we can regularise τ toward homogeneity, with a view to reduce the variance of our estimator.

3.4 First-stage estimators

All of the above methods (except CF) require the estimation of "first-stage" nuisance parameters (propensity score and/or outcome regressions). To gain insight into how the choice of first-stage estimator affects CATE prediction accuracy, we combine

7. It marginally improved performance in our case.

the methods with 3 different first-stage estimators: Ordinary Least Squares (OLS), Random Forest (RF) and Automated Machine Learning (AML).

3.4.1 OLS

OLS finds the hyperplane that minimises MSE:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}(y - x\beta)^T(y - x\beta) = (x^T x)^{-1}x^T y$$

where x is the design matrix and y is the outcome vector. The OLS predictions are then $\hat{y}(x) = x\hat{\beta}$. In high-dimensional cases, the inverse of $x^T x$ may be computationally expensive to calculate; in that case gradient descent methods can be used to find $\hat{\beta}$.

For propensity score prediction, we use a Logistic Regression, which can be thought of as the classifier counterpart of OLS. The functional form assumption is

$$\mathbb{P}(D = 1 \mid X = x; \theta) = \frac{1}{1 + e^{-x\theta}}$$

with independent observations and θ is estimated by using gradient descent to maximise the (log) likelihood:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}}\, N^{-1} \sum_{i=1}^{N} \ln \mathbb{P}(d_i \mid x_i; \theta).$$

The predicted propensity score is $\hat{p}(x) = \frac{1}{1+e^{-x\hat{\theta}}}$.

While OLS and Logistic Regression are unlikely to overfit the training data, they can give biased results if nonlinearities or interactions are a important in the data-generating process, because of the imposed functional form.

3.4.2 RF

A Random Forest, introduced by Breiman (2001), is an ensemble of many decorrelated regression trees. Regression trees (Breiman et al. 1984) recursively partition the sample along covariates to minimise the MSE of the outcome. However, regression trees are unstable and exhibit high variance. The RF combats this issue by using random subsamples to grow many trees, and by choosing the covariates for each partition randomly. A popular textbook treatment is Hastie et al. (2009), Chapters 9 and 15. RF can be used for both regression and classification (propensity score prediction).

We chose RF because 1) it usually performs very well in tabular prediction tasks with a wide range of hyperparameters; and 2) fitting times are usually less than for competing methods (e.g. Neural Networks or SVMs).

3.4.3 AML

There is growing practitioner and academic interest in Automated Machine Learning (AML), which aims to automate the traditionally tedious and error-prone task of hyperparameter-tuning. A popular AML package for Python is *auto-sklearn* (Feurer et al., n.d.). It is based upon the *scikit-learn* package, which is a collection of many regression and classification algorithms. *auto-sklearn* uses Bayesian optimisation to find good algorithms and hyperparameters, and then build an ensemble of the best-performing methods. It incorporates prior information on past performance on similar datasets. *auto-sklearn* has won multiple editions of the ChaLearn AutoML Challenge (Guyon et al. 2019).

AML is our attempt at a "carefully trained" nuisance parameter estimator. Using AML also eliminates any subjectivity in choosing between estimators (or choosing the set of estimators to be considered for cross-validation) and is likely to have the best performance of any off-the-shelf method.

3.4.4 Cross-fitting

We apply 5-fold cross-fitting with all the approaches that require the estimation of nuisance parameters in a first step. This means that we split the sample into 5 parts ("folds") of equal size, $I_1, \ldots, I_5$. First we train the ML models on the combined sample $I_2 \cap I_3 \cap I_4 \cap I_5$ and then produce predictions in sample I_1. We repeat this 4 more times, leaving out $I_2, \ldots, I_5$ in turn, to have an out-of-sample estimate for each individual. This way, the nuisance parameters and the CATE are estimated on different samples, in order to remove overfitting bias (Chernozhukov et al. 2018).

3.5 EMCS

The idea of an Empirical Monte Carlo Study was introduced by Huber et al. (2013) and Lechner and Wunsch (2013). It aims to take as many components of the DGP as possible from real data. The steps of our study are outlined below.

1. Take the full sample I_{full} that we get after the pre-processing steps in Section 4.1 and estimate the propensity score $p_{full}(x)$, using a method of choice (see Appendix B).

2. Remove all treated individuals from the sample, leaving $I = \{i \in I_{full} : D_i = 0\}$. So y_i^0 is observed for all members of I.

3. Calculate a simulated propensity score $\tilde{p}(x)$ for observations in I, which may be a modified version of $p_{full}(x)$ to control for the ratio of treated or other features of the selection process (see Section 4.2).

4. Repeat for $r = 1, \ldots R$:

   4.1. Simulate the true ITEs, $\xi_i^{(r)}$ as per Section 4.2.

   4.2. Calculate the potential outcomes under treatment as $y_i^{1(r)} = y_i^0 + \xi_i^{(r)}$ for all $i \in I$.

4.3. Simulate the treatment indicators $D_i^{(r)} \sim Bernoulli(\tilde{p}(x_i))$, for all $i \in I$.

4.4. Create the observable outcomes $y_i^{(r)} = D_i^{(r)} y_i^{1(r)} + \left(1 - D_i^{(r)}\right) y_i^0$, for all $i \in I$.

4.5. Estimate τ, the confidence interval and $\tau(x)$, from $\{(x_i, D_i^{(r)}, y_i^{(r)})\}_{i \in I}$ with all estimators of Section 3.3, yielding $\hat{\tau}_r$, $(\hat{l}_r, \hat{u}_r)$ and $\hat{\tau}_r(x_i)$.

4.6. Calculate the sample ATE $\tau_r = \frac{1}{|I|} \sum_{i \in I} \xi_i^{(r)}$

5. Calculate the performance measures of Section 3.6 for each estimator.

Steps 1 and 2 are only done once, while 3-5 are repeated for each DGP.

3.6 Performance measures

Firstly, we are interested in the ability of the various methods to predict causal effects accurately. Therefore we calculate the mean squred error (MSE) of the estimates across replications (for a given DGP) and, in the case of CATE, across individuals:

$$\widehat{MSE}_{ATE} = \frac{1}{R} \sum_{r=1}^{R} (\tau_r - \hat{\tau}_r)^2$$

$$\widehat{MSE}_{CATE} = \frac{1}{R|I|} \sum_{r=1}^{R} \sum_{i \in I} \left(\xi_i^{(r)} - \hat{\tau}_r(x_i)\right)^2$$

For CATE, sometimes we normalise the MSE for easier comparison:

$$\bar{R}^2 = 1 - \frac{1}{R} \sum_{r=1}^{R} \frac{\sum_{i \in I} \left(\xi_i^{(r)} - \hat{\tau}_r(x_i)\right)^2}{\sum_{i \in I} \left(\xi_i^{(r)} - \bar{\xi}^{(r)}\right)^2},$$

where $\bar{\xi}^{(r)} = \frac{1}{|I|} \sum_{i \in I} \xi_i^{(r)}$.

Secondly, we should prefer unbiased estimators. Even though higher (absolute) bias usually results in higher variance (since $MSE = variance + bias^2$), the MSE-

minimising estimator could be biased (if the unbiased estimators have higher variance). In a causal inference or policymaking context, we might prefer an unbiased estimator even if it has higher variance. So we calculate the mean absolute error (MAE) across replications:

$$\widehat{MAE}_{ATE} = \frac{1}{R}\left|\sum_{r=1}^{R} \tau_r - \hat{\tau}_r\right|$$

$$\widehat{MAE}_{CATE} = \frac{1}{R|I|}\sum_{i\in I}\left|\sum_{r=1}^{R}(\xi_i^{(r)} - \hat{\tau}_r(x_i))\right|$$

which is an approximation of (finite-sample) bias.

Thirdly, causal methods must produce estimates of the standard errors associated with the point estimates. To compare the "confidence"[8] of each method, we calculate the average length of the 95% confidence intervals across replications:

$$\widehat{length}_{ATE} = \frac{1}{R}\sum_{r=1}^{R}(\hat{u}_r - \hat{l}_r)$$

Even if there are theoretical guarantees about the validity of the standard error estimates, these only hold asymptotically. It is therefore useful to investigate whether the nominal 95% coverage is actually achieved in our finite samples. Highly confident estimators are not of much use if the standard errors are very different in finite samples. So we calculate coverage as

$$\widehat{coverage}_{ATE} = \frac{1}{R}\sum_{r=1}^{R}\mathbb{1}[\hat{l}_r < \tau_r < \hat{u}_r].$$

8. This is a proxy for the power against alternative hypotheses on the causal parameter.

3.6.1 A note on confidence intervals

Only 3 of our estimators have confidence intervals associated with CATE: CF, BART and BCF. Furthermore, with the DGPs we use (Section 4.2), we know the ground truth realisations of ITE, but not the CATE. Hence we do not report length and coverage for CATE confidence intervals.

Since our main focus is CATE estimation, we only consider ATE estimators that we can obtain as by-products from CATE estimators. However, we require that these methods have associated confidence intervals; this restricts us to report ATE results only for DML, CF, BART and BCF.

4 Simulation Study

We design our simulations around a particular dataset, which we also use in the empirical application (Section 6).

4.1 Dataset

We use a standard dataset of Swiss Active Labor Market Policy (ALMP), which contains rich information on all individuals who registered as unemployed at Swiss regional employment agencies in 2003. The data is not in the public domain but researchers can request it via FORSbase. Previous studies using this dataset are, among others, Lechner (2018), Knaus et al. (2020), and Knaus (2020). Lalive et al. (2002) provide a detailed description of the surrounding institutional setting. The starting number of observations is 100,120. We consider non-participants and participants in four different program types: job search, vocational training, computer programs and language courses. Since the assignment policies differ across the three language regions, we remove individuals living in the French- and Italian-speaking parts.

We focus on the first program participation within the first six months after the beginning of the unemployment spell. This definition raises the issue that the non-participant group includes individuals who quickly came back into employment, before they would be assigned to a training program. This could result in an overly optimistic evaluation of non-participation. Thus we follow Michael Lechner (1999) and Knaus (2020) and assign pseudo program starting points to the non-participants and drop those from the sample who are employed at this point. This is done using a Random Forest specification (see Section 5 of Knaus (2020) and the Online Appendix). After these cleaning steps, we have a dataset of 62,561 datapoints.

To perform the EMCS, we drop the treated individuals, resulting in a sample size of 47,684. The outcome of interest is the number of months in employment during the 31 months (which is the maximum available time span) after the program start.

Regarding the identification assumptions, SUTVA is very reasonable since the size of the workforce is sufficiently large (compared to the number of programme participants) that general equilibrium or network effects should not be significant. Exogeneity of covariates is satisfied since all the covariates were measured *before* enrollment in the programme. Conditional independence and common support are satisfied in the artificial datasets, by construction. However, they may not be satisfied in the real dataset and cannot be tested (see further discussion in Section 6).

4.2 DGPs

Since we are not able to observe ITEs or their aggregates in a real world dataset, we need to simulate them in order to assess model performance. This subsection describes our specifications.

It is henceforth assumed that all covariates have been normalised to have mean zero

and variance one. In our benchmark specification

$$\tilde{\xi}(x) = \alpha(x_0 + x_1 + x_2 x_3 + \mathbb{1}[x_4 > 0.5] + 1) + \varepsilon, \tag{8}$$

where $\varepsilon \sim N(0, \sigma^2)$ is IID across observations and the covariates x_j are *female, past income, age, city big* and *employability* respectively for $j \in \{0, \ldots, 4\}$. The other 40 covariates do not enter $\tilde{\xi}(x)$. All of these covariates are predictors of the propensity score (see Table 5). This is desirable, since in observational studies, the estimators must be able to disentangle selection bias and effect heterogeneity. Note that α is not necessarily the ATE due to correlating covariates.

The outcome variable is the number of months employed in the 31 months after the start of the unemployment spell, so $y \in \{0, 1, \ldots 31\}$. To mimic this property, we calculate the ITE as

$$\xi(x, y^0) = \begin{cases} \lfloor \tilde{\xi}(x) \rceil & \text{if } 0 \leq y^0 + \lfloor \tilde{\xi}(x) \rceil \leq 31 \\ -y^0 & \text{if } y^0 + \lfloor \tilde{\xi}(x) \rceil < 0 \\ 31 - y^0 & \text{if } y^0 + \lfloor \tilde{\xi}(x) \rceil > 31, \end{cases} \tag{9}$$

where $\lfloor . \rceil$ is rounding to the nearest integer. Hence $\xi_i = \xi(x_i, y_i^0)$ is a random variable through ε in 8 for a given individual i, and its realisation in replication r is $\xi_i^{(r)}$.

Note that through 9, ξ depends on y^0 which is taken directly from real data (it is observed for all $i \in I$). Therefore, we know the true ITEs but we do not know the functional form of the true CATE, $\tau(x) = \mathbb{E}(\xi_i \mid x_i = x)$, because ξ_i depends on y_i^0, which in turn depends on x_i in some unknown way.

On top of the benchmark DGP, we consider 8 other specifications, by varying the *effect size* α, *noise level* σ, amount of *confounding*, complexity of *heterogeneity* and the *number of treated* individuals, as well as include a DGP with *linear* conditional

treatment effects. The functional forms are in Appendix A. The purpose of this is to create more challenging setups, in order to test if estimation performance decreases. In theory, we could try each possible combination of parameters (e.g. high confounding combined with high noise level and few treated individuals), but we had to compromise to keep computational time and the number of results manageable. By altering each parameter one at a time, as shown in Table 1, we also hope to disentangle the effect of each.

DGP	α	σ	Extra confounding	Heterogeneity	Linear	Few treated
0 (b.m.)	1	1	no	medium	no	no
1	**0**	1	no	medium	no	no
2	**5**	1	no	medium	no	no
3	1	**25**	no	medium	no	no
4	1	1	**yes**	medium	no	no
5	1	1	no	**high**	no	no
6	1	1	no	**none**	no	no
7	1	1	no	medium	**yes**	no
8	1	1	no	medium	no	**yes**

Table 1: DGPs (altering parameters one at a time)

5 EMCS results

Results are displayed in Table 2 for ATE and Table 4 for CATE. The rest of this section interprets the results and highlights interesting features.

5.1 ATE

All of the methods succeed in accurately estimating the ATE across our DGP specifications. The overall best-performing method is DML AML. The two DGPs that result in somewhat less accurate estimation are DGPs 4 (extra confounding) and 8 (few treated observations).

DGP	average				0			
metric	MSE	MAE	length	cov.	MSE	MAE	length	cov.
DML OLS	0.031	0.115	0.666	1.000	0.022	0.133	0.555	1.0
DML RF	0.055	0.178	0.894	0.978	0.039	0.190	0.749	1.0
DML AML	**0.024**	0.052	0.668	1.000	**0.004**	0.043	0.562	1.0
CF	0.039	0.132	0.668	1.000	0.020	0.133	0.564	1.0
BART	0.028	0.073	0.713	0.900	0.011	0.082	0.621	1.0
BCF	0.032	0.100	0.634	0.889	0.012	0.089	0.510	0.9
DGP	4				8			
metric	MSE	MAE	length	cov.	MSE	MAE	length	cov.
DML OLS	**0.012**	0.090	0.545	1.0	0.120	0.090	1.635	1.0
DML RF	0.077	0.259	0.721	0.9	0.153	0.174	2.010	1.0
DML AML	0.039	0.191	0.548	1.0	0.138	0.049	1.506	1.0
CF	0.122	0.345	0.552	1.0	0.116	0.132	1.498	1.0
BART	0.106	0.316	0.691	0.4	0.092	0.043	1.496	0.7
BCF	0.131	0.356	0.528	0.1	**0.091**	0.174	1.422	1.0

Table 2: ATE results: average across DGPs and selected DGPs. DML AML is the best performer. The rest of the results have been relegated to the Online Appendix.

Regarding the choice of first-stage estimator for DML, Table 3 presents the MSEs of (out-of-sample) predictions from the 4 nuisance functions $\hat{y} = \hat{m}(x)$, $\hat{y}^0 = \hat{\mu}_0(x)$, $\hat{y}^1 = \hat{\mu}_1(x)$ and $\hat{p} = \hat{p}(x)$ for the 3 applied methods, for repetition 1 of DGP 0. OLS and RF perform very similarly (except for $\hat{p}$), suggesting that interactions and nonlinearities are not very important for outcome prediction in this dataset. As expected, AML outperforms the other methods on all 4 prediction tasks. Turning to ATE estimation, we see in Table 2 that AML-based DML consistently outperforms it OLS- or RF-based counterparts. This suggests that better first-stage predictions translate to better ATE estimation. Interestingly, the while OLS and RF give similar predictive performance, the RF-based DML does worse. This may be due to the higher variance of RF compared to OLS.

DML has the highest MSE (although small bias) on DGP 8 (few treated observations), possibly because the low propensity scores in the denominator of the DR MOM scores lead to large variance. Such behaviour has been documented by Kang, Schafer, et al. (2007). Unfortunately, imbalanced classes is a common phenomenon in empirical econometrics.

Regarding confidence intervals, all the methods produce similar lengths. DML and CF achieve 100% coverage, suggesting that the confidence intervals are on the conservative side.[9] BART and BCF do not cover appropriately on DGP 4 (extra confounding), achieving only 40% and 10% coverage, respectively.

	$\hat{y}$	$\hat{y}^0$	$\hat{y}^1$	$\hat{p}$
OLS	130.90	130.16	123.42	0.00672
RF	130.31	128.34	123.64	0.00443
AML	125.80	125.66	121.05	0.00142

Table 3: MSEs of nuisance parameter estimates

5.2 CATE

No estimator performs uniformly best. Overall, the R-learner (with AML) and BCF are the best-performing estimators, closely followed by CF and BART. These four estimators consistently provide the lowest MSE and bias. The X-learner and DR MOM also provide performance gains over the simple T-learner. The "traditional" estimators (T OLS and F (IPW) OLS) are clearly outperformed by the ML-based estimators.

The DML estimator based on the DR MOM scores performed best for ATE estimation, but DR MOM is not among the top 4 methods for CATE. Its relative performance is worst for DGP 8 (few treated observations), which we also saw for ATE.

Regarding the choice of first-stage estimators, the same observation applies as for ATE: estimators based on the highly tuned AML outperform the OLS- and RF-based counterparts (at least for those methods that perform well in general: DR MOM, X, R). Interestingly, the T-learner (and the badly performing U- and F-learners) with

9. "Statistics is the only profession which demands [...] to make mistakes 5 per cent of the time" - Thomas Huxley

DGP	0	1	2	3	4	5	6	7	8
T OLS	3.09	2.05	25.20	182.66	8.16	4.68	2.50	2.88	10.32
T RF	9.42	8.62	24.76	186.46	12.57	12.09	8.94	9.53	13.67
T AML	3.80	3.07	20.78	181.02	7.35	5.71	3.52	4.01	8.19
X OLS	3.01	1.98	25.09	182.27	8.10	4.61	2.44	2.83	8.58
X RF	2.80	1.77	24.98	182.43	7.79	4.46	2.22	2.64	8.33
X AML	2.64	1.59	24.79	182.15	7.88	4.25	2.04	2.45	8.14
F OLS	3.94	2.81	26.37	183.49	8.86	5.71	3.42	3.77	20.49
F RF	7.08	5.67	30.52	187.78	11.33	9.51	6.90	6.85	45.36
F AML	5.90	4.71	28.84	186.03	10.24	7.93	5.52	5.68	31.96
U OLS	10.10	9.16	31.99	189.70	15.50	11.83	9.56	9.97	567.87
U RF	51.96	51.04	73.28	226.59	58.54	52.34	50.39	51.15	2783.78
U AML	12.80	15.83	34.64	198.16	21.36	16.40	14.86	14.58	789.77
DR MOM OLS	3.16	2.13	25.29	182.97	8.01	4.74	2.60	2.96	23.44
DR MOM RF	3.19	2.16	25.22	183.22	8.03	4.75	2.61	2.98	10.61
DR MOM AML	3.10	2.10	25.16	183.06	7.91	4.68	2.53	2.90	10.22
R OLS	**2.12**	0.89	24.62	182.06	7.46	4.26	1.43	1.98	2.69
R RF	2.26	0.90	24.78	182.07	7.82	4.24	1.42	2.09	2.86
R AML	2.14	0.87	24.56	182.10	7.54	**4.11**	**1.38**	**1.96**	2.71
CF	2.18	1.07	18.91	**179.73**	6.87	4.95	1.53	2.33	2.69
BCF	2.34	**0.83**	**16.46**	181.24	7.45	4.58	1.43	2.30	**2.66**
BART	2.19	0.86	20.76	182.23	**6.49**	4.82	1.42	2.32	2.78

Table 4: MSE results. Overall, the best performers are R-learning, CF, BCF and BART, followed by the X-learner and DR MOM. More granular results and MAE have been relegated to the Online Appendix. Estimators with lower MSE almost always have lower MAE.

RF perform much worse than the T-learner with OLS, but the difference disappears with DR MOM, the X-learner, and the R-learner. Hence, more accurate first-stage predictions seem to translate into better CATE predictions.

5.2.1 Benchmark DGP

First, we consider our models' ability to learn a good approximation of the CATE function of DGP 0. As Figure 1 shows, some, but not all, methods achieve a positive R^2.[10] The best-performers are the R-learner, BART and CF, substantially outperforming the T-learner.

10. Note that a method would achieve an R^2 of 0 if it *perfectly* estimated ATE but did not detect any heterogeneity. On DGPs 1 and 6, we should expect that all methods achieve a *negative* R^2, since most of the variation in ITE comes from random error, not the covariates.

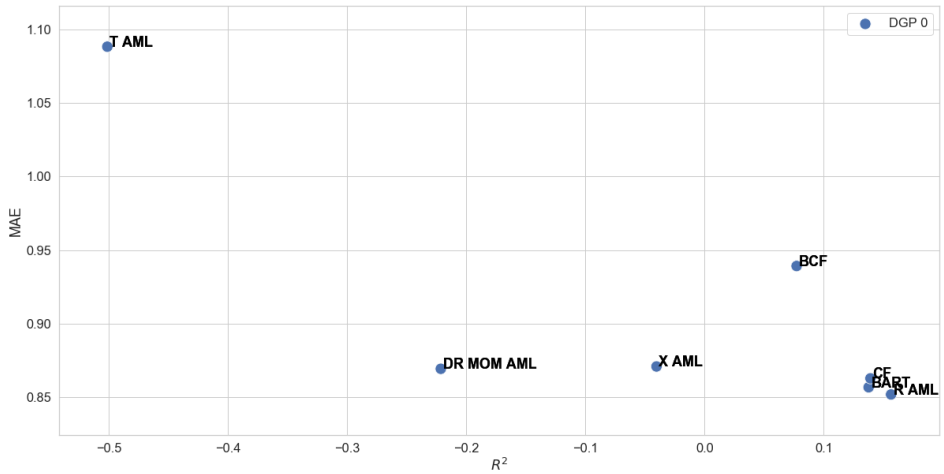


Figure 1: R^2 and bias of ITE prediction on DGP 0.

To see how similar the estimates from different models are, Figure 2 shows the correlation matrix of the true ITE and the predictions in replication 1. The correlations appear modest, even among some of the best-performers, suggesting that combining estimates from different methods could further improve performance.[11]

To investigate which variables were picked up as effect moderators, we calculated the importance metrics introduced in Section 6.3.2 for R AML, CF, BCF and BART. The estimators found, in this order, *female*, *past income*, *age* and *city big* as the most important effect moderators; these are 4 of the 5 variables entering the CATE specification. *Employability* was not identified (except for BCF), possibly because its contribution to the CATE is the same for 91% of the sample. The results are available in the Online Appendix.

Overall, the results show that the studied estimators successfully found the important variables and estimate the CATE function well.

11. Section 4.2 of Nie and Wager (2017) proposes a stacking method based on the R-objective. Unfortunately, we are not aware of any documented software package that supports R-stacking.

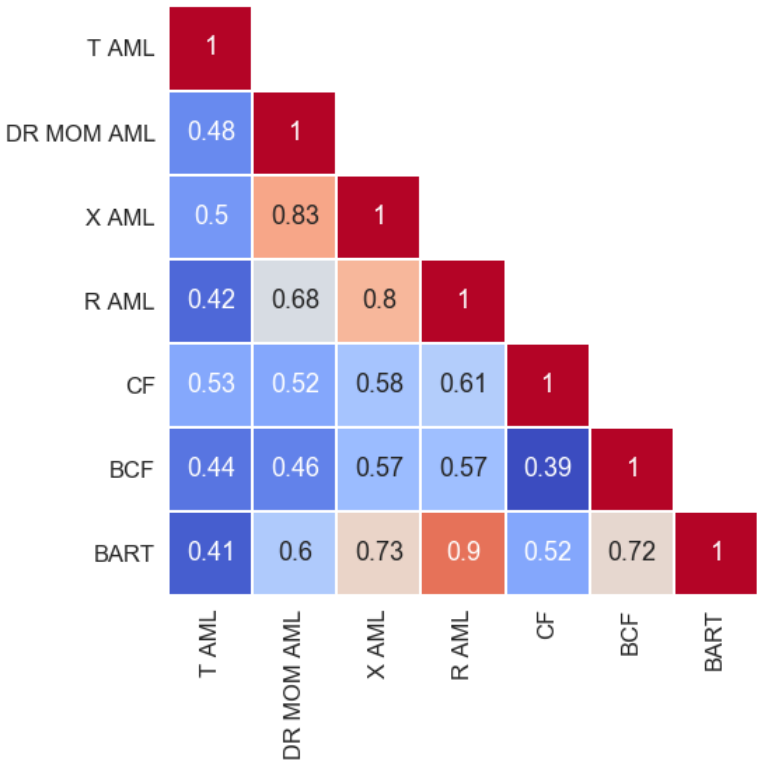


Figure 2: Correlation of CATE estimates on DGP 0, replication 1

5.2.2 DGPs 1-2 (effect size)

Figure 3 shows how the MSEs change as we increase the effect size α from 0 to 1 to 5 (DGPs 1, 0, 2, respectively). For each method, MSE is increasing in the effect size. However, R^2 increases from about -8% to 15% to 44% for the best-performing methods as we increase the effect size (i.e. the increase in $\text{Var}(\xi_i)$ dominates). Interestingly, CF and BCF start outperforming R-learning when the effect size is large. BCF is the most accurate for both $\alpha = 0$ and $\alpha = 5$.

5.2.3 DGP 3 (high noise)

As we increase the ITE conditional standard deviation σ from 1 to 25, almost all the variation in ITE comes from noise, rather than the covariates. MSE increases, as expected, and R^2 decreases to 4% for CF (which is the best-performing method on this DGP). The relative gap between the best and worst methods becomes small.

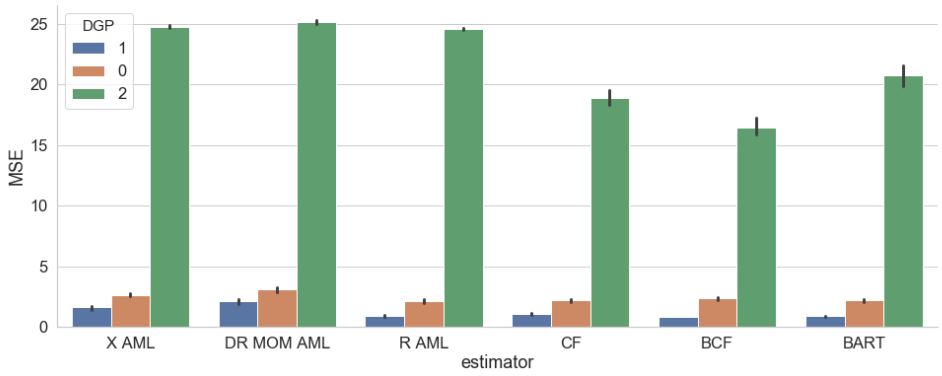


Figure 3: MSE for DGPs 0-2. The height of the bars shows the average, and the black segments show the range of results across replications.

5.2.4 DGP 4 (extra confounding)

Making the CATE a strongly nonlinear function of the propensity score increases MSE and bias substantially. This suggests that the estimators cannot perfectly control for confounding.

5.2.5 DGPs 5-6 (heterogeneity)

In DGP 5, the CATE is a function of 30 covariates, while in DGP 6 the CATE is constant (apart from the effects of truncation at 0 and 31). Figure 4 shows that higher levels of heterogeneity lead to higher MSE, although the deterioration is not as large as in the case of larger effect size.

5.2.6 DGP 7 (linear)

Specifying the CATE as a linear function does not have much of an effect on prediction accuracy. This suggests that the estimators correctly picked up the interaction and the step function part of CATE in DGP 0.

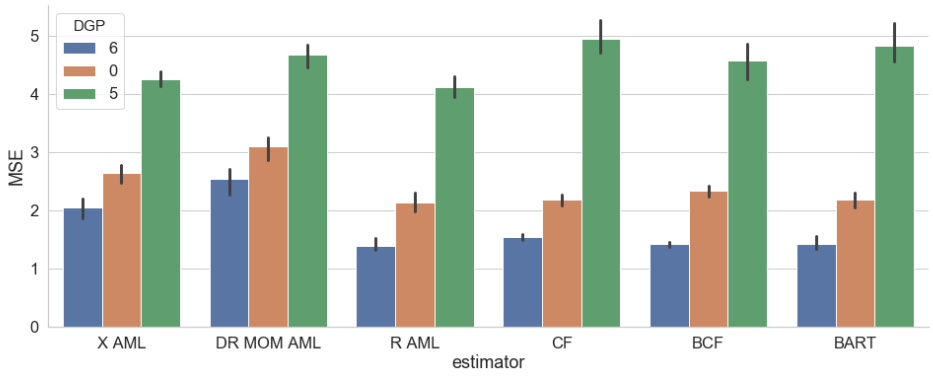


Figure 4: MSE for DGPs 0, 5 and 6. The height of the bars shows the average, and the black segments show the range of results across replications.

5.2.7 DGP 8 (few treated observations)

When we reduce the propensity scores so that only about 2% (950) of the observations get treated, the estimators using the propensity score in the denominator (F, U, DR MOM) get much worse. Interestingly, the MSE of the X-learner also trebles. On the other hand, the 4 best-performers (R AML, CF, BCF, BART) all show only modest increases in MSE.

6 Empirical application

In this section, we use the best-performing methods of the EMCS (which were the R-learner, CF, BCF and BART) to estimate heterogenous treatment effects on the original data. Schuler et al. (2017) introduce "synth-validation" for causal methods, an analogy of cross-validation for prediction tasks. It involves generating synthetic ITEs on top of the real covariates, and using the best-performing methods to estimate parameters in the real data. Hence the EMCS can be viewed as our synth-validation for the empirical application.

There are 5 treatment groups, *no program* (47684 observations), *job search* (11610 obs.), *vocational* training (858 obs.), *computer* training (905 obs.) and *language* training (1504 obs.).

6.1 Identification assumptions

We rely on the identification assumptions of Section 3.2. The SUTVA assumption, i.e. that units are not affected by the treatment assignment of other units, is a standard and very reasonable assumption in our setting. Exogeneity of covariates is also satisfied, since all covariates were measured at the point of filing as unemployed, i.e. before any treatment assignment.

The common support and conditional independence assumptions, which were satisfied *by design* in the EMCS, are more problematic in the empirical application. We try to assess the common support assumption by using a logit model to predict the probability of seeing an individual in each program, given their covariates. The minimum estimated propensity score in the *no program* sample is 2.7%, 0.1%, 0.02% and 0.04% for the programs *job search*, *vocational*, *computer* and *language*, respectively. The data are not linearly separable (otherwise logit would not converge), but the extremely low propensity scores for the 3 smaller programs are cause for concern. One remedy would be to drop low-propensity-score observations, but this introduces sample-selection issues. Since a violation of common support is not important as long as conditional mean regressions provide good counterfactuals, we leave the sample as is; a fuller analysis would incorporate sensitivity analysis regarding this decision. The maximum propensity scores are clearly bounded away from 1.

The conditional independence (unconfoundedness) assumption is not amenable to statistical tests, by its nature of concerning unobserved factors. It could be argued that characteristics like the "go for it attitude" increase both the participation probability and the employment outcome. To relax the conditional independence assumption, one could try to find an instrument for treatment assignment (Angrist

and Pischke 2008), or conduct sensitivity analysis for hidden confounding (Rosenbaum 2002). The DML framework of Chernozhukov et al. (2018) allows for IV estimation in the context of machine learning nuisance parameter estimation. However, this topic is not the focus of our study and hence we do not take that direction.

6.2 Preliminary analysis

Table 5 presents statistics for selected covariates. The *mean* columns show the sample mean of the outcome and covariates in each of the 5 treatment groups. Most of the covariates have statistically significantly different means in the treated groups compared to *no programme*. (* denotes 1% significance for the difference of means, using a standard two-sided 2-sample t-test.) This suggests that treatment assignment is not random, and causal parameter estimation should take place with estimators designed for observational, rather than experimental, data.

To further investigate treatment assignment, the columns under *APE* display the Average Partial Effects of the covariates when we fit logistic regressions.[12] Many variables, including previous labour market success, cantonal macroeconomic indicators, caseworker characteristics and gender statistically significantly predict treatment assignment. (* denotes rejecting APE=0 at 1% significance.)

The column *no prog.* under *OLS* shows the OLS coefficients from regressing the outcome on all covariates in the non-treated sample. Most coefficients, including many that predicted treatment assignment, are significant at 1%. Since the same variables influence assignment and the outcome, we have have to control for observed confounding.

12. We combine the *no program* sample with the treated groups, one at a time.

	mean					APE				OLS				
	no prog.	job sch.	vocat.	comp.	lang.	job sch.	vocat.	comp.	lang.	no prog	job sch.	vocat.	comp.	lang.
outcome	14.696	14.369*	18.414*	19.197*	13.521*	-	-	-	-	-	-	-	-	-
age	36.609	37.310*	37.449*	39.075*	35.281*	0.001*	0.000	0.001*	-0.001*	-0.211*	-0.006	0.061	0.002	-0.027
city_big	0.194	0.185	0.209	0.114*	0.227*	-0.000	0.005*	-0.010*	0.001	-0.119	-0.069	0.014	-0.985	-1.465
city_medium	0.122	0.135*	0.119	0.155*	0.147*	0.014*	0.002	0.005*	0.003	-1.033*	0.010	1.561	-0.020	0.798
cw_cooperative	0.481	0.500*	0.411*	0.418*	0.452	0.007	-0.004*	-0.003*	-0.005*	0.088	0.069	-1.079	0.663	0.328
cw_female	0.435	0.473*	0.386*	0.442	0.473*	0.025*	-0.002	-0.003	0.003	0.266	-0.141	-0.739	1.138	-1.054
cw_tenure	5.478	5.438	5.730	5.831*	5.606	0.001*	0.000	0.001*	0.000	0.053*	-0.011	-0.012	0.001	0.018
cw_voc_degree	0.255	0.273*	0.216*	0.248	0.215*	0.020*	-0.003	-0.003	-0.004	0.342*	0.179	0.593	-0.433	-0.036
emp_sh_last_2yrs	0.810	0.841*	0.832*	0.839*	0.716*	0.034*	-0.001	0.002	-0.017*	4.229*	-0.914	-0.421	-2.799	-2.010
emp_spells_5yrs	1.213	0.970*	0.925*	0.862*	0.783*	-0.011*	-0.002*	-0.002*	-0.006*	-0.398*	-0.046	0.216	-0.671	0.183
employability	1.927	1.980*	1.930	1.972*	1.848*	0.026*	-0.000	-0.000	-0.004	2.280*	-0.470	-1.770*	-1.232	-0.738
female	0.436	0.442	0.330*	0.602*	0.549*	0.010*	-0.005*	0.012*	0.010*	0.697*	0.161	-1.328	2.512**	-3.519**
foreigner_b	0.134	0.106*	0.125	0.040*	0.438*	-0.015	-0.001	-0.015*	0.029*	-1.508*	0.169	2.083	-1.072	-3.266**
foreigner_c	0.231	0.223	0.178*	0.168*	0.225	-0.001	-0.006*	-0.004	0.008*	-1.885*	-0.098	1.744	-0.936	-3.152**
gdp_pc	0.524	0.527	0.514*	0.526	0.540*	-0.294*	-0.015	0.057*	0.046*	-12.725*	1.905	-3.735	1.305	8.523
married	0.471	0.462	0.478	0.450	0.717*	-0.002	0.001	0.001	0.010*	0.288	-0.418	0.820	-0.961	-0.121
other_m_tongue	0.334	0.290*	0.308	0.177*	0.641*	-0.014*	0.002	-0.008*	0.021*	-1.452*	-0.223	-0.562	-3.207*	-1.924*
past_income	4.252	4.669*	4.865*	4.321	3.730*	0.012*	0.002*	-0.000	0.001*	0.833*	-0.206**	-0.527*	-0.457	0.044
prev_job_skilled	0.601	0.648*	0.648*	0.748*	0.426*	-0.024	-0.008	0.007	0.000	-3.333*	-0.176	-0.635	9.587**	2.497
prev_job_unskilled	0.295	0.245*	0.219*	0.149*	0.484*	-0.026	-0.011*	-0.002	0.003	-4.797*	0.375	-1.079	9.164*	1.355
qual_degree	0.580	0.621*	0.634*	0.717*	0.377*	-0.016*	0.000	0.005	-0.001	1.771*	0.423	0.739	-0.339	-1.298
qual_wo_degree	0.032	0.032	0.021	0.024	0.074*	0.004	-0.006	0.006	0.006	0.426	0.313	-2.347	0.361	-0.035
ue_spells_2yrs	0.569	0.394*	0.524	0.372*	0.432*	-0.015*	0.000	-0.001	-0.002*	-0.833*	0.128	-0.986*	0.366	0.170
unemp_rate	3.521	3.590*	3.410*	3.362*	3.626*	0.029*	-0.003	-0.007*	-0.002	0.648*	0.272	-0.265	0.189	-0.540

Table 5: Sample means, logit APE and OLS results for selected covariates. * denotes 1% significance, except in the last 4 columns, where * is 5% and ** is 1%.

The last 4 columns detect (linear) heterogeneity in treatment effects. We combine the *no program* sample with one of the treated samples and fit the regression

$$y_i = \alpha + \beta x_i + \gamma d_i + \delta d_i x_i + \varepsilon_i$$

where x_i is a column vector of covariates, d_i is the treatment indicator and β, δ are row parameter vectors. The last 4 columns are the OLS estimates $\hat{\delta}$ when each of the 4 treated groups are used (now * denotes 5% and ** denotes 1% significance). There is some evidence of treatment effect heterogeneity with respect to gender and past income, and foreigner status in the case of *language* programs. However, by imposing a linear structure, we potentially miss non-linear and interactive relationships.

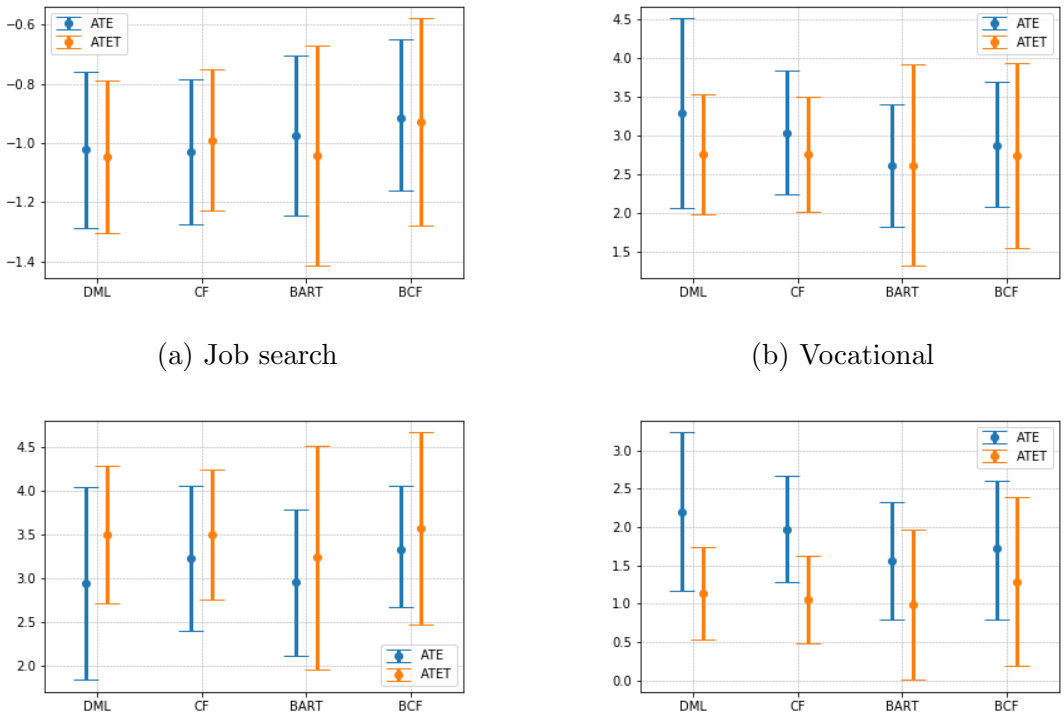
The above observations motivate the use of methods that can flexibly capture heterogeneity and learn confounding effects among a large number of covariates, without requiring the imposition of strong functional form assumptions. The estimators studied thus far aim to achieve exactly this.

6.3 Results

6.3.1 ATE and ATET

We apply DML (with AML), CF, BCF and BART to estimate the ATE of each program. Figure 5 shows the point estimates and 95% confidence intervals produced by each estimator.

The estimates and confidence intervals from the 4 methods are remarkably similar, especially for the *job search* program, where we have the most treated observations. We find substantial differences in the effectiveness of programs. The *job search* program decreases expected employment by about a month in the 31-month data period. On the other hand, the programs teaching hard skills increase expected



(a) Job search

(b) Vocational

(c) Computer

(d) Language

Figure 5: Point estimates and 95% confidence intervals for ATE by the 4 estimators

employment by about 3 months (*vocational* and *computer* training) or 1.5-2 months (*language* course). All of the results are highly statistically significant.

Since our data records employment status in each month for 31 months after the beginning of the unemployment spell, we can estimate the ATEs for any shorter timeframe. In the first two months, all 4 programs reduce expected employment. This is because participants spend their time on the program rather than working (or looking for a job). After this initial lock-in period, the participants of the hard-skill programs catch up (i.e. the program benefits outweigh the time spent in the program) but there is no such catch-up for *job search* participants. Hence, the *job search* program could be harmful; however it is also possible that *job search* participants become more picky regarding job offers (which could lead to better labour market matches). Unfortunately, our data does not allow to test this hypothesis.

Comparing ATE and ATET shows no big differences.[13] This implies that there is either no effect heterogeneity correlated with observables or that the treatment assignment does not take advantage of such heterogeneity.

6.3.2 CATE

Figure 7 presents the histograms (kernel density estimates) of estimated CATE for all 4 treatment groups and 4 estimators. All estimators suggest a similar range for the majority of treatment effects, although the CF estimates are somewhat more spread out. Many histograms are multi-modal, especially for the *computer* training program. This happens because categorical variables are picked up as effect moderators.

Similarly to Figure 2, we calculate the correlations among the estimated CATEs for each program. The correlations all fall in the range 0.5-0.7, suggesting that the 4 estimators find similar patterns. In the following, we average the 4 estimates.

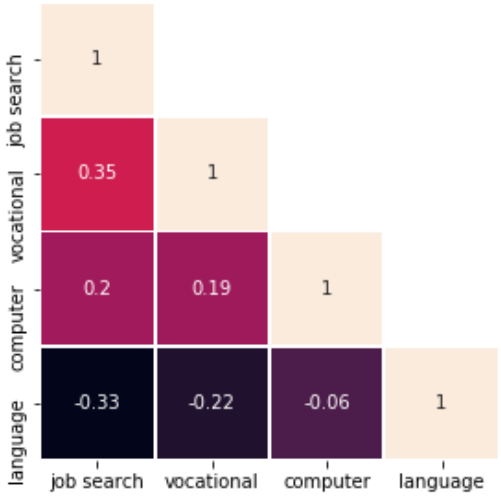
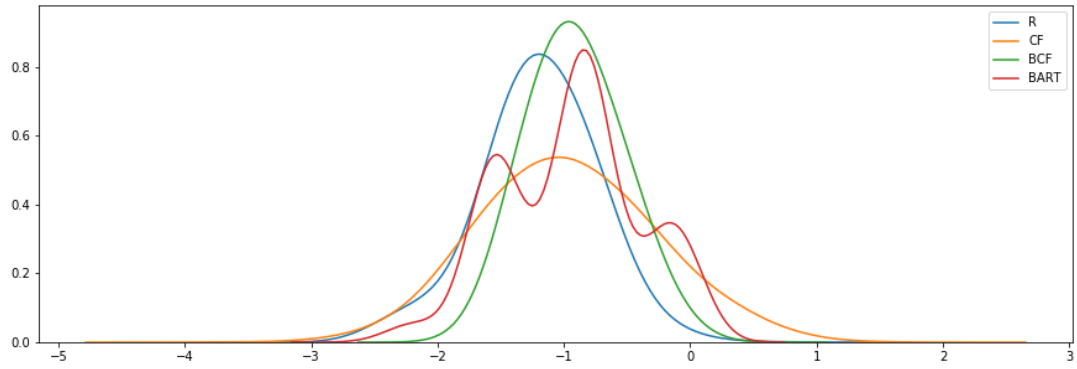


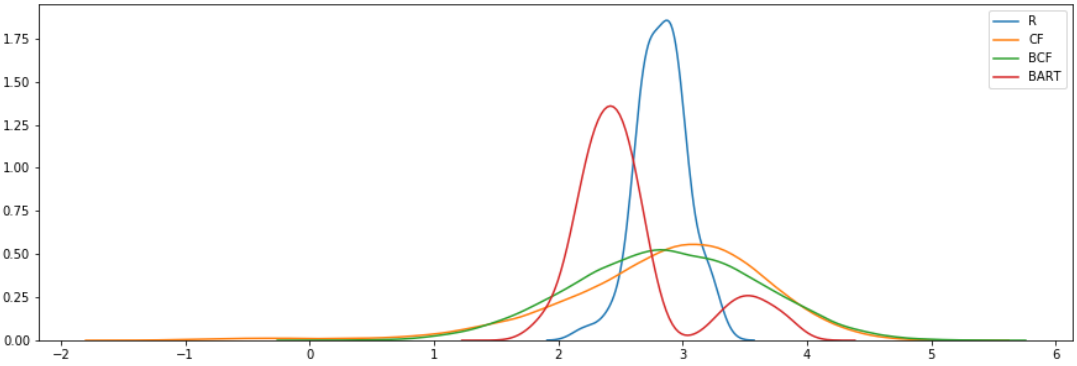
Figure 6: Correlation of estimated CATEs across programs

Another interesting question is whether the CATEs are correlated over different

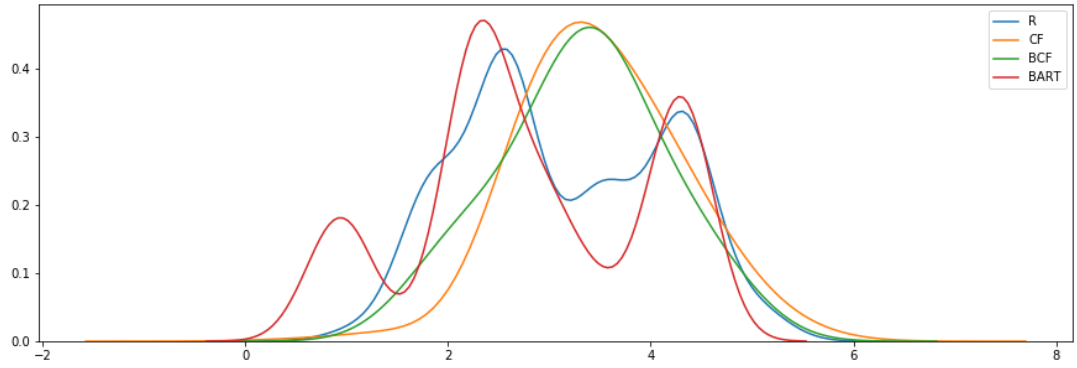
13. Maybe apart from the *language* program.



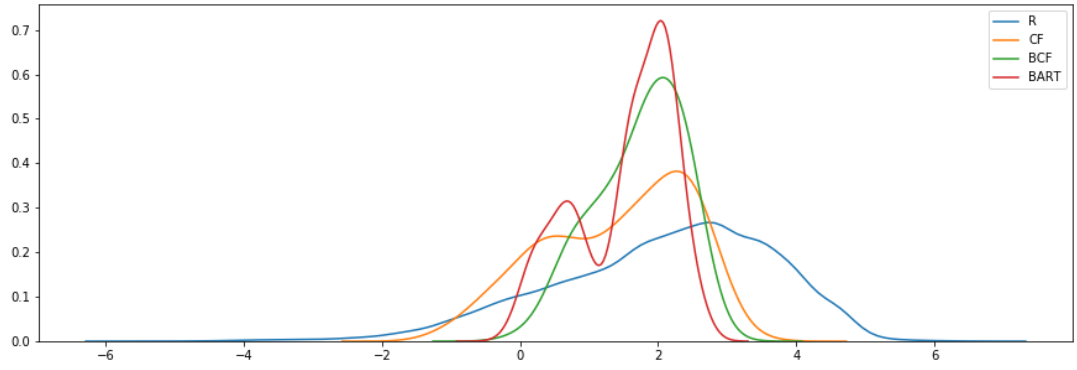
(a) Job search



(b) Vocational



(c) Computer



(d) Language

Figure 7: CATE kernel density estimates

treatments. Figure 6 suggests that the *job search*, *vocational* and *computer* programs are substitutes in the sense that they work better for the same group of individuals, while the *language* program is a complement to the others. Although the correlations do not take into account that the average effects are at different levels, it is suggested that different programs work better for different groups and this could be exploited for improved targeting.

To investigate which variables were picked up as effect moderators, we regress the CATE estimates on all covariates by OLS. A linear equation is most likely misspecified, but estimates the best linear predictor of CATEs, allows for *ceteris paribus* interpretation and makes result communication straightforward. In order to allow for non-linear and interactive relationships, we also fit Gradient Boosted Trees.[14] To gain insight into which variables are important, we calculate the Permutation Importance (Breiman 2001; Fisher et al. 2019) and Shapley Additive Explanations (SHAP) values (Shapley 1953; Štrumbelj and Kononenko 2014; Lundberg and Lee 2017) for each covariate. An excellent overview of the broader topic of "interpretable machine learning" (including Permutation Importance and SHAP) is Molnar (2020).

Permutation Importance measures the importance of a covariate by calculating the increase in the model's prediction error when the values for the covariate are randomly shuffled. A covariate is "important" if shuffling its values increases the model error, because in this case the model relied on the covariate for prediction.

The goal of SHAP is to explain the prediction of a model at given covariates x. A prediction can be explained by assuming that each covariate value of the observation is a "player" in a game where the prediction is the payout. The Shapley value – a solution concept from coalitional game theory – tells us how to fairly distribute the "payout" among the covariates. To produce a global importance metric, we average

14. Calculating SHAP values for a Random Forest is computationally infeasible, hence the decision to use Boosting instead.

program	job search			vocational			computer			language		
metric	OLS	perm	shap	OLS	perm	shap	OLS	perm	shap	OLS	perm	shap
age	0.004*	0.05	0.15	0.004*	0.05	0.22	0.011*	0.03	0.12	-0.02*	0.14	0.30
female	0.03*	0.00	0.03	-0.02*	0.01	0.09	1.33*	1.00	1.00	-0.67*	0.52	0.70
past_income	-0.08*	0.77	0.68	-0.07*	0.68	0.92	-0.08*	0.12	0.22	0.04*	0.05	0.14
employability	-0.03*	0.01	0.04	-0.41*	1.00	1.00	-0.34*	0.05	0.13	-0.00*	0.00	0.01
emp_last_2yrs	-0.74*	1.00	1.00	0.00	0.01	0.08	-0.29*	0.02	0.10	-0.48*	0.07	0.14
foreigner_c	0.01*	0.00	0.01	0.01*	0.00	0.02	-0.03*	0.00	0.01	-0.57*	0.20	0.32
other_m_tongue	0.02*	0.00	0.03	0.02*	0.01	0.09	-0.73*	0.26	0.46	-0.94*	1.00	1.00
unemp_rate	0.05*	0.07	0.18	0.02*	0.04	0.18	0.09*	0.00	0.05	-0.02*	0.00	0.04
ue_cw_alloc4	-0.04*	0.00	0.02	0.02*	0.00	0.03	-0.00	0.00	0.00	-0.00	0.00	0.00

Table 6: Variable importance in CATE prediction: OLS coefficients, Permutation Importance and SHAP values for selected variables. For easier interpretation, we normalise the Permutation Importance and SHAP values such that the most important covariate receives a score of 1. The covariate *ue_cw_alloc4* concerns the caseworker allocation procedure, which we do not expect to influence CATE. We included it to confirm that "unimportant" covariates receive low scores. * denotes 1% significance. Full results are in the Online Appendix.

the (absolute values of) Shapley values at each observation.

Table 6 presents the results for selected covariates. We see that the strongest predictor of *job search* and *vocational* training treatment effects is previous labour market success; higher past success predicts lower CATE. Being *female* is associated with higher benefits from *computer* and lower benefits from *language* training. Being a foreigner, interestingly, negatively predicts benefits from the *language* course. All of these findings are in line with the previous literature (see Knaus 2020).

Finally, we estimate the optimal tree-based treatment assignment rule, and the gains from it, using honest splitting[15] (Sverdrup et al. 2020; Athey and Wager 2021). The policy tree in Figure 8 shows that females should be allocated to the *computer* program and males to the *vocational* program,[16] unless they have high past income.

15. Half the sample is used to find splitting points, and the other half to estimate ATE at each node.

16. Such an assignment rule might be considered discriminatory. It is possible to restrict the covariates the policy tree is allowed to split on to produce socially acceptable assignment rules.

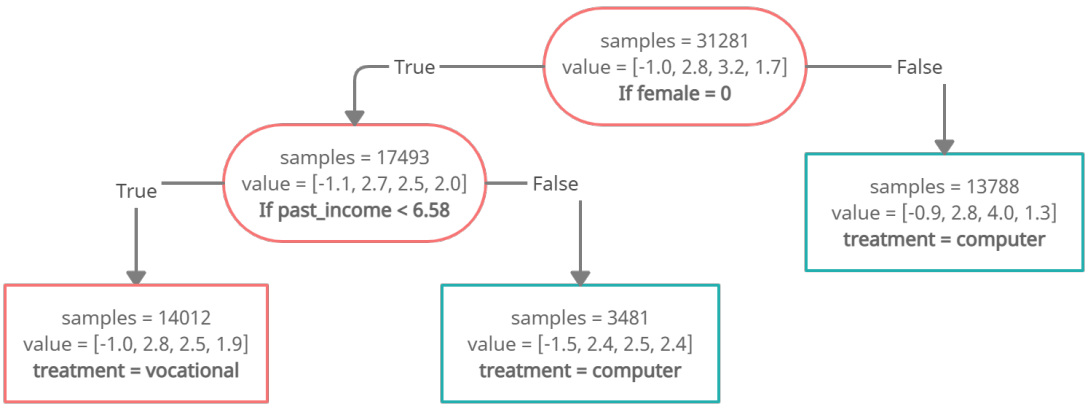


Figure 8: Policy tree of depth 2. The "value" vector shows the estimated employment gains for the *job search*, *vocational*, *computer* and *language* programs, respectively, for the sub-sample corresponding to the given node.

By using this policy tree, an average 3.29-month increase in the months of employment is achieved compared to *no program*. This is only a 0.13-month increase compared to assigning everyone to the *computer* program. Such a small performance gain can be explained by the substantial differences between the 4 program ATEs, limited heterogeneity, and the substitutability of the *vocational* and *computer* programs (which have the highest ATEs). The estimated performance gain goes up by a further 0.07 when we allow the decision tree to be of depth 5, but does not increase further with increasing depth.

7 Conclusion

7.1 Limitations and Future Research

Arguably, synthetic evaluations of causal inference methods lack realism because subject-matter knowledge is an important part of causal analysis (Hernán 2019). While we agree that using prior knowledge is often beneficial, we believe that letting the data speak, without allowing the (possibly biased) views of the researcher to influence the results, is also important. Hence studying and using off-the-shelf methods for causal inference, which do not require functional form specifications or manual variable selection, is important.

Other interesting modifications to our DGP would have been varying the error distribution, sample size, number of covariates or the strength of selection into treatment; or making the unconfoundedness or overlap assumptions fail. A DGP with known functional form for CATE would have allowed evaluating confidence intervals for CF, BART and BCF. We did not pursue these ideas due to space limitations.

The advantage of using several DGPs is that we can assess the generalisability of our results, i.e. if the same estimators perform well in different scenarios. However, other fields can have radically different data structures, making it possible that other estimators become more suitable. Therefore, future simulation studies will complement our results.

Currently, the estimators of this study are scattered across Python and R packages. A software package providing a common, intuitive API to use the various causal ML methods would facilitate the adaptation of such techniques by researchers.

The policymaker might want to assess the stability of the estimated optimal treatment assignment rules before applying them. However, statistical inference on optimal treatment assignment rules is (to our knowledge) still an open question.

7.2 Summary

This is one of the first comprehensive simulation studies in economics that investigates the finite-sample performance of a large number of causal machine learning estimators. We rely on arguably realistic DGPs by taking covariates from a real dataset. Our main goal is to estimate individual-level treatment effects (CATE), and we additionally report results for average effects (ATE).

We find that the DML method with carefully trained first-stage estimators performs

best for ATE estimation (except with imbalanced treatment classes), although all studied estimators provide low bias and (mostly) adequate confidence interval coverage.

Regarding CATE, 4 estimators (R-learner, CF, BART and BCF) perform best consistently across our DGP specifications. Methods based on the highly flexible AML estimator outperform their OLS/logit-based counterparts.

In our empirical application on Swiss unemployment training programs, we uncover treatment effect heterogeneity with respect to demographic and individual labour market variables.

Economics is the study of the allocation of scarce resources. In many situations, policymakers have to allocate scarce treatments – e.g. seats at unemployment training programs or Covid-19 vaccines – where the effectiveness of those treatments depends on the individual characteristics of the potential recipients. Data can help us discover such dependencies, and therefore to optimise the allocation process. Our study illustrates how this can be done.

References

Abadie, Alberto. 2005. "Semiparametric difference-in-differences estimators." *The Review of Economic Studies* 72 (1): 1–19.

Angrist, Joshua D, and Jörn-Steffen Pischke. 2008. *Mostly harmless econometrics: An empiricist's companion.* Princeton university press.

Ashenfelter, Orley. 1978. "Estimating the effect of training programs on earnings." *The Review of Economics and Statistics,* 47–57.

Athey, Susan, Julie Tibshirani, Stefan Wager, et al. 2019. "Generalized random forests." *Annals of Statistics* 47 (2): 1148–1178.

Athey, Susan, and Stefan Wager. 2021. "Policy learning with observational data." *Econometrica* 89 (1): 133–161.

Bach, P., V. Chernozhukov, M. S. Kurz, and M. Spindler. 2020. "DoubleML - Double Machine Learning in Python." Accessed April 16, 2021. https://github.com/DoubleML/doubleml-for-py.

Belloni, Alexandre, Victor Chernozhukov, Ivan Fernández-Val, and Christian Hansen. 2017. "Program evaluation and causal inference with high-dimensional data." *Econometrica* 85 (1): 233–298.

Breiman, Leo. 2001. "Random forests." *Machine learning* 45 (1): 5–32.

Breiman, Leo, Jerome Friedman, Charles J Stone, and Richard A Olshen. 1984. *Classification and regression trees.* CRC press.

Caron, Alberto, Gianluca Baio, and Ioanna Manolopoulou. 2021. "Sparse Bayesian Causal Forests for Heterogeneous Treatment Effects Estimation." *arXiv preprint arXiv:2102.06573.*

Chen, Tianqi, and Carlos Guestrin. 2016. "XGBoost: A Scalable Tree Boosting System." In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 785–794. KDD '16. San Francisco, California, USA: ACM. ISBN: 978-1-4503-4232-2. https://doi.org/10.1145/2939672.2939785. http://doi.acm.org/10.1145/2939672.2939785.

Chernozhukov, Victor, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. 2018. "Double/debiased machine learning for treatment and structural parameters" [in eng]. *The econometrics journal* 21 (1): C1–C68. ISSN: 1368-423X.

Chipman, Hugh A, Edward I George, Robert E McCulloch, et al. 2010. "BART: Bayesian additive regression trees." *The Annals of Applied Statistics* 4 (1): 266–298.

Dorie, Vincent, and Jennifer Hill. 2020. *bartCause: Causal Inference using Bayesian Additive Regression Trees.* R package version 1.0-4. https://CRAN.R-project.org/package=bartCause.

Dorie, Vincent, Jennifer Hill, Uri Shalit, Marc Scott, Dan Cervone, et al. 2019. "Automated versus do-it-yourself methods for causal inference: Lessons learned from a data analysis competition." *Statistical Science* 34 (1): 43–68.

Feurer, Matthias, Aaron Klein, Katharina Eggensperger, Jost Springenberg, Manuel Blum, and Frank Hutter. 2015. "Efficient and Robust Automated Machine Learning." In *Advances in Neural Information Processing Systems 28,* edited by C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, 2962–2970. Curran Associates, Inc. http://papers.nips.cc/paper/5872-efficient-and-robust-automated-machine-learning.pdf.

———. n.d. "Efficient and Robust Automated Machine Learning, 2015." *URL http://papers.nips. cc/paper/5872-efficient-and-robust-automated-machine-learning.*

Fisher, Aaron, Cynthia Rudin, and Francesca Dominici. 2019. *All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously.* arXiv: 1801.01489 [stat.ME].

Freund, Yoav, Robert E Schapire, et al. 1996. "Experiments with a new boosting algorithm." In *icml,* 96:148–156. Citeseer.

Guyon, Isabelle, Lisheng Sun-Hosoya, Marc Boullé, Hugo Jair Escalante, Sergio Escalera, Zhengying Liu, Damir Jajetic, et al. 2019. "Analysis of the AutoML Challenge Series 2015–2018." In *Automated Machine Learning: Methods, Systems, Challenges,* edited by Frank Hutter, Lars Kotthoff, and Joaquin Vanschoren, 177–219. Cham: Springer International Publishing. ISBN: 978-3-030-05318-5. https://doi.org/10.1007/978-3-030-05318-5_10. https://doi.org/10.1007/978-3-030-05318-5_10.

Hahn, P Richard, Carlos M Carvalho, David Puelz, Jingyu He, et al. 2018. "Regularization and confounding in linear regression for treatment effect estimation." *Bayesian Analysis* 13 (1): 163–182.

Hahn, P Richard, Vincent Dorie, and Jared S Murray. 2019. "Atlantic causal inference conference (acic) data analysis challenge 2017." *arXiv preprint arXiv:1905.09515*.

Hahn, P Richard, Jared S Murray, Carlos M Carvalho, et al. 2020. "Bayesian regression tree models for causal inference: Regularization, confounding, and heterogeneous effects (with discussion)." *Bayesian Analysis* 15 (3): 965–1056.

———. 2021. "bcf on GitHub." Accessed April 16, 2021. https://github.com/jaredsmurray/bcf.

Harris, Charles R., K. Jarrod Millman, St'efan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, et al. 2020. "Array programming with NumPy." *Nature* 585, no. 7825 (September): 357–362. https://doi.org/10.1038/s41586-020-2649-2. https://doi.org/10.1038/s41586-020-2649-2.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. *The elements of statistical learning: data mining, inference, and prediction.* Springer Science & Business Media.

Hernán, Miguel A. 2019. "Comment: Spherical cows in a vacuum: data analysis competitions for causal inference." *Statistical Science* 34 (1): 69–71.

Hill, Jennifer, Antonio Linero, and Jared Murray. 2020. "Bayesian additive regression trees: a review and look forward." *Annual Review of Statistics and Its Application* 7:251–278.

Hill, Jennifer L. 2011. "Bayesian nonparametric modeling for causal inference." *Journal of Computational and Graphical Statistics* 20 (1): 217–240.

Hirano, Keisuke, Guido W Imbens, and Geert Ridder. 2003. "Efficient estimation of average treatment effects using the estimated propensity score." *Econometrica* 71 (4): 1161–1189.

Huber, Martin, Michael Lechner, and Conny Wunsch. 2013. "The performance of estimators based on the propensity score." *Journal of Econometrics* 175 (1): 1–21.

Imbens, Guido W, and Jeffrey M Wooldridge. 2009. "Recent developments in the econometrics of program evaluation." *Journal of economic literature* 47 (1): 5–86.

Johansson, Fredrik, Uri Shalit, and David Sontag. 2016. "Learning representations for counterfactual inference." In *International conference on machine learning,* 3020–3029. PMLR.

Kang, Joseph DY, Joseph L Schafer, et al. 2007. "Demystifying double robustness: A comparison of alternative strategies for estimating a population mean from incomplete data." *Statistical science* 22 (4): 523–539.

Knaus, Michael C. 2020. "Double machine learning based program evaluation under unconfoundedness." *arXiv preprint arXiv:2003.03191.*

Knaus, Michael C, Michael Lechner, and Anthony Strittmatter. 2020. "Heterogeneous employment effects of job search programmes: A machine learning approach." *Journal of Human Resources,* 0718–9615R1.

———. 2021. "Machine learning estimation of heterogeneous causal effects: Empirical monte carlo evidence." *The Econometrics Journal* 24 (1): 134–161.

Korobov, Mihail, and Konstantin Lopuhin. 2021. "ELI5 documentation." Accessed April 16, 2021. https://eli5.readthedocs.io/en/latest/overview.html.

Künzel, Sören R, Jasjeet S Sekhon, Peter J Bickel, and Bin Yu. 2019. "Metalearners for estimating heterogeneous treatment effects using machine learning." *Proceedings of the national academy of sciences* 116 (10): 4156–4165.

Lalive, Rafael, JC Ours, and Josef Zweimüller. 2002. "The impact of active labor market programs on the duration of unemployment." *Working paper/Institute for Empirical Research in Economics* 41.

Lechner, M. 2018. *Penalized causal forests for estimating heterogeneous causal effects.* Technical report. Working Paper.

Lechner, Michael. 1999. "Earnings and employment effects of continuous off-the-job training in East Germany after unification." *Journal of Business & Economic Statistics* 17 (1): 74–90.

Lechner, Michael, and Conny Wunsch. 2013. "Sensitivity of matching-based program evaluations to the availability of control variables." *Labour Economics* 21:111–121.

Lundberg, Scott, and Su-In Lee. 2017. "A unified approach to interpreting model predictions." *arXiv preprint arXiv:1705.07874.*

Lundberg, Scott M., Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. 2020. "From local explanations to global understanding with explainable AI for trees." *Nature Machine Intelligence* 2 (1): 2522–5839.

McConnell, K John, and Stephan Lindner. 2019. "Estimating treatment effects with machine learning." *Health services research* 54 (6): 1273–1282.

McCulloch, Warren S, and Walter Pitts. 1943. "A logical calculus of the ideas immanent in nervous activity." *The bulletin of mathematical biophysics* 5 (4): 115–133.

Molnar, Christoph. 2020. *Interpretable machine learning.* Lulu.com. https://christophm.github.io/interpretable-ml-book/.

Nie, Xinkun, Alejandro Schuler, and Stefan Wager. 2021. *rlearner: R-learner for Heterogeneous Treatment Effect Estimation.* R package version 1.1.0.

Nie, Xinkun, and Stefan Wager. 2017. "Quasi-oracle estimation of heterogeneous treatment effects." *arXiv preprint arXiv:1712.04912.*

O'Neill, Eoghan. 2019. "State-of-the-BART: Simple Bayesian Tree Algorithms for Prediction and Causal Inference."

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, et al. 2011. "Scikit-learn: Machine Learning in Python." *Journal of Machine Learning Research* 12:2825–2830.

Powers, Scott, Junyang Qian, Kenneth Jung, Alejandro Schuler, Nigam H Shah, Trevor Hastie, and Robert Tibshirani. 2018. "Some methods for heterogeneous treatment effect estimation in high dimensions." *Statistics in medicine* 37 (11): 1767–1787.

R Core Team. 2020. *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Robins, James M, and Andrea Rotnitzky. 1995. "Semiparametric efficiency in multivariate regression models with missing data." *Journal of the American Statistical Association* 90 (429): 122–129.

Rosenbaum, Paul R. 2002. "Overt bias in observational studies." In *Observational studies,* 71–104. Springer.

Rossum, Guido van, et al. 2021. "Python.org website." Accessed April 16, 2021. https://www.python.org/.

Rubin, Donald B. 1974. "Estimating causal effects of treatments in randomized and nonrandomized studies." *Journal of educational Psychology* 66 (5): 688.

Schuler, Alejandro, Ken Jung, Robert Tibshirani, Trevor Hastie, and Nigam Shah. 2017. "Synth-validation: Selecting the best causal inference method for a given dataset." *arXiv preprint arXiv:1711.00083.*

Schwab, Patrick, Lorenz Linhardt, and Walter Karlen. 2018. "Perfect match: A simple method for learning representations for counterfactual inference with neural networks." *arXiv preprint arXiv:1810.00656.*

Seabold, Skipper, and Josef Perktold. 2010. "statsmodels: Econometric and statistical modeling with python." In *9th Python in Science Conference.*

Shalit, Uri, Fredrik D Johansson, and David Sontag. 2017. "Estimating individual treatment effect: generalization bounds and algorithms." In *International Conference on Machine Learning,* 3076–3085. PMLR.

Shapley, Lloyd S. 1953. "A value for n-person games." *Contributions to the Theory of Games* 2 (28): 307–317.

Sparapani, Rodney, Charles Spanbauer, and Robert McCulloch. 2021. "Nonparametric Machine Learning and Efficient Computation with Bayesian Additive Regression Trees: The BART R Package." *Journal of Statistical Software* 97 (1): 1–66. https://doi.org/10.18637/jss.v097.i01.

Štrumbelj, Erik, and Igor Kononenko. 2014. "Explaining prediction models and individual predictions with feature contributions." *Knowledge and information systems* 41 (3): 647–665.

Sverdrup, Erik, Ayush Kanodia, Zhengyuan Zhou, Susan Athey, and Stefan Wager. 2020. "policytree: Policy learning via doubly robust empirical welfare maximization over trees." *Journal of Open Source Software* 5 (50): 2232.

Tibshirani, Julie, Susan Athey, and Stefan Wager. 2020. *grf: Generalized Random Forests.* R package version 1.2.0. https://CRAN.R-project.org/package=grf.

Tibshirani, Robert. 1996. "Regression shrinkage and selection via the lasso." *Journal of the Royal Statistical Society: Series B (Methodological)* 58 (1): 267–288.

Vapnik, Vladimir. 1996. *The nature of statistical learning theory.* Springer.

Wager, Stefan, and Susan Athey. 2018. "Estimation and inference of heterogeneous treatment effects using random forests." *Journal of the American Statistical Association* 113 (523): 1228–1242.

Wendling, T, K Jung, A Callahan, A Schuler, NH Shah, and B Gallego. 2018. "Comparing methods for estimation of heterogeneous treatment effects using observational data from health care databases." *Statistics in medicine* 37 (23): 3309–3324.

Zaidi, Abbas, and Sayan Mukherjee. 2018. "Gaussian Process Mixtures for Estimating Heterogeneous Treatment Effects." *arXiv preprint arXiv:1812.07153.*

A The DGPs

Here we describe the DGPs 1-8 of Table 1. The functional form of $\tilde{\xi}$ in Equation 8 is altered, but we keep the rounding-truncating rule in Equation 9.

For DGPs 1, 2 and 3, the functional form is the same as for DGP 0, given by Equation 8, only the effect size α and noise level σ are varied, as shown in Table 1.

For DGP 4 (extra confounding) we construct

$$w(x) = \sin\left(1.25\pi \frac{p_{full}(x)}{\max(p_{full}(x))}\right),$$

where $p_{full}(x)$ is the propensity score (from step 1 of the EMCS) and $\max(p_{full}(x))$ is the maximum propensity score in the sample. We then normalise by $W(x) = 3\frac{w(x)-\bar{w}}{sd(w(x))}$ so that W has mean zero and variance 3. Then we replace Equation 8 by

$$\tilde{\xi}(x) = W(x) + \alpha(x_0 + x_1 + x_2 x_3 + \mathbb{1}[x_4 > 0.5] + 1) + \varepsilon.$$

$W(x)$ is a (highly nonlinear) function of the propensity score, so it is expected to be hard to disentangle the effect each covariate plays in treatment allocation and the outcome (through ITE). We took the idea for $W(x)$ from Knaus et al. (2021), who used it in their main DGP.

For DGP 5 (high heterogeneity), we add linear and quadratic effects in many covariates. We uniformly randomly select 20 covariates x_j and put them into a vector x_l, and draw 20 IID $U(0,1)$ values β_j and put them into vector β_l. Similarly, we randomly select further 5 covariates (vector x_q) and draw 5 IID $U(0,0.5)$ (vector β_q). We rewrite 8 as

$$\tilde{\xi}(x) = x_l\beta_l + x_q^2\beta_q + \alpha(x_0 + x_1 + x_2x_3 + \mathbb{1}[x_4 > 0.5] + 1) + \varepsilon,$$

where squaring is element-wise.

For DGP 6 (no heterogeneity), we set a $\tilde{\xi}(x) = 2 + \varepsilon$ independent of x.

For DGP 7 (linear ITE), 8 becomes

$$\tilde{\xi}(x) = 1 + female + pastincome + 2 \cdot age + \varepsilon.$$

For DGP 8, we have the original functional form for the ITE, but we scale the propensity score such that only 2% of the sample gets treated (in expectation):

$$\tilde{p}(x) = 0.02 \cdot \frac{|I| p_{full}(x)}{\sum_{i \in I} p_{full}(x_i)}$$

in step 3 of the EMCS. (For all other DGPs, we use $\tilde{p}(x) = p_{full}(x)$.)

B Implementation details

We rely on the following software packages.

The T-, X-, U-, F-learners and DR MOM are based on own implementation. Code is provided in the Online Appendix.

Name	Version	Citation	Used for
Python	3.7.6	Rossum et al. (2021)	Packages NumPy, pandas, statsmodels, scikit-learn, xgboost, auto-sklearn, doubleml, ELI5, SHAP.
R	4.0.3	R Core Team (2020)	Packages rlearner, grf, BART, bartCause, bcf.
NumPy	1.18.1	Harris et al. (2020)	Data management, random number generation.
pandas	1.0.1	The pandas development team (2020)	Data management.
statsmodels	0.10.2	Seabold and Perktold (2010)	OLS, logit with p-values.
scikit-learn	0.22.1	Pedregosa et al. (2011)	OLS, logit, Random Forest (regressor, classifier).
xgboost	0.90	Chen and Guestrin (2016)	Gradient Boosting (regressor).
auto-sklearn	0.12.4	Feurer et al. (2015)	AML (regressor, classifier).
DoubleML	0.2.1	Bach et al. (2020)	DML.
ELI5	0.10.1	Korobov and Lopuhin (2021)	Permutation Importance.
SHAP	0.37.0	Lundberg et al. (2020)	SHAP values (TreeExplainer).
rlearner	1.1.0	Nie et al. (2021)	R-learner.
grf	1.2.0	Tibshirani et al. (2020)	Causal Forest.
BART	2.9	Sparapani et al. (2021)	BART for CATE.
bartCause	1.0-4	Dorie and Hill (2020)	BART for ATE and ATET.
bcf	2.0.0	Hahn et al. (2021)	BCF.

Table 7: List of software packages used.

Regarding hyperparameter choices for the machine learners/estimators, we usually used the default parameters (which can be found in the package documentation). Sometimes we increased the number of trees to obtain more stable estimates. The exact details can be found in our code, uploaded to the Online Appendix.

We also fit CF with Local Centering, and using the cross-validation option in the *grf* package on some datasets. These options did not seem to improve performance and hence we did not pursue them further. The same applies to the Sparse version of BCF (Caron et al. 2021).