Undersmoothing Data Augmentation To Reduce Bias for Heterogeneous Causal Effect Estimation from Observational Data

Damian Machlanski,[1] **Spyros Samothrakis,** [2] **Paul S. Clarke** [3]
[1] School of Computer Science and Electronic Engineering
[2] Institute for Analytics and Data Science
[3] Institute for Social and Economic Research
University of Essex
Colchester, UK
d.machlanski@essex.ac.uk, ssamot@essex.ac.uk, pclarke@essex.ac.uk

Abstract

Inferring individualised treatment effects from observational data can unlock the potential for targeted interventions. It is, however, hard to infer these effects, as the data collected is often biased. One way of looking at the problem is through covariate shift — while the data (outcome) conditional distribution remains the same, the covariate (input) distribution changes between the training and test set. In an observational data setting, this problem is materialised in control and treated units coming from different distributions. A common solution is to augment learning methods through reweighing schemes (e.g. propensity scores). These are needed due to model misspecification, but might hurt performance in the individual case. In this paper, we explore a novel generative tree based approach that tackles model misspecification directly, helping downstream estimators achieve better robustness. We show empirically that the choice of model class can indeed significantly affect the final performance and that reweighing methods can struggle in individualised effect estimation. Our proposed approach is competitive with reweighing methods on average treatment effects while performing significantly better on individualised treatment effects.

Introduction

In the absence of data from randomised experiments, analysts must use observational data to make inferences about the causal effects of interventions or treatments, that is, what would happen if we intervened to change the treatment status of individual units in a population. The estimation of average causal effects — the average effect of an intervention aggregated across every unit in a population — has been studied in considerable depth. However, there is now growing interest in estimating heterogeneous intervention effects for individuals characterized by possibly large numbers of input variables or covariates. By allowing for heterogeneity across units, such systems can unlock the analysis of targeted interventions, for instance, in the form of personalised healthcare based on covariates that describe patients' symptoms and health histories.

The use of observational data creates challenges for the estimation of heterogeneous causal effects. First, the analyst must assume that treatment selection is strongly ignorable with respect to the covariates. We take this assumption to hold throughout and focus on the second problem, namely, that nonrandom treatment selection can lead to observational data in which the distributions of covariates among the treated and untreated units are very different. In practice, this can make it difficult for conventional learners to learn the true relationship between the intervention effect and covariates across the entire support of the covariates, and result in poor performance when tested on experimental data.

More generally, this issue is known as covariate shift, which in the causal inference setting means the learning target $P(y|x)$ remains unchanged, while marginal distributions of inputs $P(x)$ differ between observational and interventional distributions. Existing methods attempt to transform the observational distribution so it resembles the interventional one, mostly (but not exclusively, e.g. using domain adaptation methods) through sample reweighing that usually involves propensity scores (Chernozhukov et al. 2018; Robins, Rotnitzky, and Zhao 1994; Rosenbaum and Rubin 1983; Künzel et al. 2019; Athey, Tibshirani, and Wager 2019). However, as pointed out by Wen, Yu, and Greiner (2014), this issue can also be perceived through the lens of model misspecification (White 1981), something not taken into account by reweighing methods, leading to poor performance in estimating individual treatment effects. An interesting alternative to classic approaches is undersmoothing, where the model is allowed to fit the data very closely, which is potentially more suitable for individualised predictions. Encouraged by suggestions in Chernozhukov et al. (2016, footnote 3) and Newey, Hsieh, and Robins (1998), we explore this novel approach here.

In this paper, we hence tackle model misspecification directly through undersmoothing by augmenting existing data with fast and straightforward generative trees (Correia, Peharz, and de Campos 2020) that facilitate more robust learning of downstream estimators. More specifically, we use those trees to "descritise" the input space into subpopulations of similar units (subclassification). The distributions of those groups are then modelled separately via mixtures of Gaussians, from which we sample equally to reduce data imbalances. Ultimately, injecting new informative data points to the original data is expected to increase data complexity and bring the observational distribution closer to the inter-

ventional one, consequently leading to increased robustness of the models using the augmented data.

Data augmentation has proven effective in multiple scenarios. For instance, image transformations in computer vision (Perez and Wang 2017), or oversampling minority classes in imbalanced classification problems (Chawla et al. 2002; He et al. 2008). In our case, the method we propose could be seen as oversampling underrepresented data regions instead of just classes.

Generative models have also been investigated in causal inference literature (Athey et al. 2020; Neal, Huang, and Raghupathi 2021), though mostly for benchmarking purposes, where new synthetic data sets are created that closely resemble real data but with access to true effects. This work, on the other hand, goes beyond data modelling and focuses on targeted data augmentation instead.

In terms of this paper's contributions, we show empirically that the choice of model class can have a substantial effect on estimator's final performance, and that standard reweighing methods can struggle with individual treatment effect estimation. Given our experiments, we also provide an evidence that our proposed method increases data complexity that leads to statistically significant improvements in individual treatment effect estimation, while keeping the average effect predictions competitive. Our experimental setup incorporates a wide breadth of non-neural standard causal inference methods and data sets.

The rest of the document is structured as follows. First, we revisit fundamental concepts that should aid understanding of the technical part of the paper. Next, we formally discuss the problem of model misspecification, followed by a thorough description of our proposed method. We then present our experimental setup and obtained results. Next section provides further discussion on the results, their implications and considered limitations of the method. Final section concludes the paper.

Preliminaries

This section gives a brief overview of the essential background deemed relevant to this work. For a more extensive review, we refer the reader to classic positions on causal analysis (Pearl 2009; Peters, Janzing, and Schölkopf 2017), and recent surveys on causal inference (Guo et al. 2020; Yao et al. 2020).

Given two random variables T and Y, investigating effects of interventions can be described as measuring how the outcome Y differs across different inputs T. Real world systems usually contain other background covariates, denoted as X, which have to be accounted for in the analysis as well. To formally approach the task, we take Rubin's Potential Outcomes (Rubin 1974) perspective, which is particularly convenient in outcome estimation without knowing the full causal graph.

To properly describe the problem of estimating causal effects, we start with a potential outcome defined as $\mathcal{Y}_t^{(i)}$, which is the observed outcome when individual i receives treatment t. Given this, the Individual Treatment Effect (ITE) can be written as:

$$ITE_i = \mathcal{Y}_1^{(i)} - \mathcal{Y}_0^{(i)} \tag{1}$$

Thus, to compute such a value for an individual i, we need access to both outcomes $\mathcal{Y}_1^{(i)}$ and $\mathcal{Y}_0^{(i)}$. However, in practice, only one of those outcomes is observed, called a *factual*, leaving the other outcome unobservable, referred to as *counterfactual*. The fact that we only observe factuals but need also counterfactuals to properly compute causal effects is known as the fundamental problem of causal inference.

Other commonly sought after effect values are Average Treatment Effect (ATE) and Conditional Average Treatment Effect (CATE).

$$ATE = \mathbb{E}\left[\mathcal{Y}_1 - \mathcal{Y}_0\right] \tag{2}$$

$$CATE = \mathbb{E}\left[\mathcal{Y}_1 | X = x\right] - \mathbb{E}\left[\mathcal{Y}_0 | X = x\right] \tag{3}$$

Where $\mathbb{E}[.]$ denotes mathematical expectation. In ATE, we are essentially interested in the effect value across the entire population. This approach, however, is not always meaningful due to the heterogeneity of the effect between subpopulations. In such cases, CATE seems to be more informative as it allows the effect to be conditioned on the subpopulation of interest. Also, note that ITE is a special case of CATE, where each individual i in ITE corresponds to conditional groups x in CATE. In other words, a unique individual is the smallest possible group the effect is conditioned on.

Despite the fact that the aforementioned treatment effects usually cannot be calculated directly, successful methods have been developed so far that attempt to approximate those quantities. Perhaps the simplest and most naive approach is regression adjustment, where a regressor, or multiple ones per each treatment value, is used to estimate potential outcomes. More advanced methods often incorporate propensity scores, where the estimator takes into account the probability of treatment assignment per each individual. For instance, Inverse Propensity Weighting (Rosenbaum and Rubin 1983) adjusts sample importances, further extended to more efficient and stable Doubly Robust method (Robins, Rotnitzky, and Zhao 1994; Foster and Syrgkanis 2020). Double Machine Learning (Chernozhukov et al. 2018), on the other hand, improves existing statistical estimators using base learners. Furthermore, recent surge in machine learning also delivered powerful procedures, often pushing state-of-the-art results (Johansson, Shalit, and Sontag 2016; Shalit, Johansson, and Sontag 2017; Yao et al. 2018; Louizos et al. 2017). In the realm of ensembles, there is Causal Forest (Athey, Tibshirani, and Wager 2019) that specifically targets CATE estimation. Another interesting perspective on the problem is given through meta-learners (Künzel et al. 2019; Nie and Wager 2020), where out of the box estimators are used in various combinations and strategies to collectively approximate causal effects.

These are the most common methods that employ the usual assumptions, that is, *SUTVA* and *strong ignorability*, though there are many procedures that attempt to relax some of the assumptions as well. Here, we limit our discussion to this standard set of assumptions as it is relevant to this work.

For a broader overview of available causal inference methods, as well as formal definitions of the assumptions, consult recent reviews on the topic (Guo et al. 2020; Yao et al. 2020).

Model Misspecification

The choice of model class occurs at some point in any learning task. Such a decision is made based on available data, usually the training part of it, while the environment of the actual application can be different, a scenario often mimicked via a separate test set. The occurring discrepancies between those two data sets are known as covariate shift problem. Within causal inference, this manifests as differences between observational and interventional distributions, ultimately making effect estimation extremely difficult. More formally, given input covariates x, treatment t, and outcome y, the conditional distribution $P(y|x,t)$ remains unchanged across the entire data set, whereas marginal distributions $P(x,t)$ differ between observational and interventional data. This is where model misspecification occurs as the model class is selected based on available observations only, which does not generalises well to later predicted interventions.

Let us consider a simple example as presented in Figure 1. It consists of a single input feature x, output variable y (both continuous), and binary treatment t. For convenience, let us denote this data set as $\mathcal{D}$. Note the effect is clearly heterogeneous as it differs in $\mathcal{D}(x < 0.5)$ and $\mathcal{D}(x > 0.5)$. Furthermore, the two data regions closer to the top of the figure, that is, $\mathcal{D}(x < 0.5, t = 1)$ and $\mathcal{D}(x > 0.5, t = 0)$, are in minority with respect to the rest of the data. By many learners these scarce data points will likely be treated as outliers, resulting in lower variance than needed to provide accurate estimates. Thus, naively fitting the data will lead to biased estimates, an example of which is depicted on the figure as *Biased T* and *Biased C*. However, what we aim for is an unbiased estimator that captures the data closely while still generalising well, a scenario showcased by *Unbiased T* and *Unbiased C* on the figure.

For ITE estimation, fitting the data closely is especially important. Although in case of average effect estimation the difference between biased and unbiased estimators can be negligible, the individualised case usually exacerbates the issue. For instance, in the presented example, the difference in ATE error is 0.44, but it grows to 0.77 in ITE error.

In this work, instead of altering the sample importance, as many existing methods do, we aim to augment provided data in a way that underrepresented data regions are no longer dominated by the rest of the samples, leading to estimators no longer treating those data points as outliers and fitting them more closely, ultimately resulting in less biased solutions and more accurate ITE estimates. The following section describes our proposed method in detail.

Debiasing Generative Trees

As described in the previous section, model misspecification can be caused by underrepresented or missing data regions. Reweighing partially addresses this problem, but struggles with ITE estimation, not to mention propensity score approximators are subject to misspecification too. To avoid

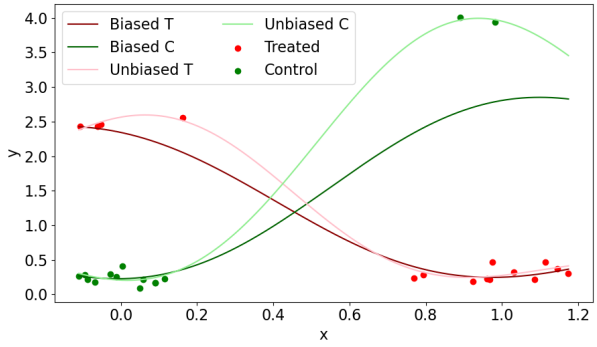

Figure 1: An example highlighting model misspecification issue. T and C denote Treated and Control respectively. The difference in ITE error is almost twice as in ATE.

these pitfalls, we tackle the misspecification through undersmoothness by augmenting the original data with new data points that carry useful information and help achieve the final estimators better ITE predictions. As the injected samples are expected to be informative to the learners, the overall data complexity increases as a consequence. Moreover, because this is a data augmentation procedure, it is estimator agnostic, that is, it can be used by any existing estimation methods. It is also worth pointing out that simply modelling and oversampling the entire joint distribution would not work as the learnt joint would include any existing data imbalances. In other words, underrepresented data regions would remain in minority, not addressing the problem at hand.

This observation led us to a conclusion there is a need to identify smaller data regions, or clusters, and model their distributions in separation instead, giving us control over which areas to sample from and with what ratios. To achieve this, we incorporate recently proposed Generative Trees (Correia, Peharz, and de Campos 2020), which retain all the benefits of standard decision trees, such as simplicity, speed and transparency. They can also be easily extended to ensembles of trees, often improving the performance significantly. In practice, a standard decision tree regressor is used to learn the data. Once the tree is constructed, the samples can be assigned to tree leaves according to the learnt decision paths, forming distinct subpopulations that we are after. The distributions of these clusters are then separately modelled through Gaussian Mixture Models (GMMs). Similarly to decision trees, we again prioritise simplicity and ease of use here, which is certainly the case with GMMs. The next step is to sample equally from modelled distributions, that is, to draw the same amount of new samples per each GMM. In this way, we reduce data imbalances. A merge of new and original data is then provided to a downstream estimator, resulting in a less biased final estimator. Through experimentation, we find that splitting the original data at the beginning of the process into treated and control units and learning two separate trees for each group helps achieve better overall effect. A step-by-step description of the proposed procedure is

Algorithm 1: Debiasing Generative Trees

Input: X - data set, E - estimator
Parameter: N - number of generated samples
Output: E_D - debiased estimator

Let $X_G = \varnothing$.
Split X into treated and control units (X_T and X_C).
Train a Decision Tree regressor on X_T.
Map X_T to tree leaves. Obtain subpopulations S.
Let $N_G = N/(2 \times len(S))$.
for S_i in S **do**
    Model S_i with Gaussian Mixture Models. Obtain G_i.
    Draw N_G samples from G_i. Store them in X_G.
end for
Repeat steps 3-9 for X_C.
Merge X and X_G into a single data set X_M.
Train estimator E on X_M. Get debiased estimator E_D.
return debiased estimator E_D

presented in Algorithm 1.

As ensembles of trees almost always improve over simple ones, we incorporate Extremely Randomised Trees for an additional performance gain. The procedure remains the same on a high level, differing only in randomly selecting inner trees at the time of sampling. Overall, we call this approach Debiasing Generative Trees (DeGeTs) as a general framework, with DeGe Decision Trees (DeGeDTs) and DeGe Forests (DeGeFs) for realisations with Decision Trees and Extremely Randomised Trees respectively.

There are a few important parameters to take care of when using the method. Firstly, depth of trees controls the granularity of identified subpopulations. Smaller clusters may translate to less accurate modelled distributions, whereas too shallow trees will bring the modelling closer to the entire joint that may result in not solving the problem of interest at all. The other tunable knob is the amount of new data samples to generate, where more data usually equates to a stronger effect, but also higher noise levels, which must be controlled to avoid destroying meaningful information in the original data. Finally, the number of components in GMMs is worth considering, where more complex distributions may require higher numbers of components.

All of the parameters can be found through cross-validation by using a downstream estimator's performance as a feedback signal as to which parameters work the best, which can also be tailored to a specific estimator of choice. The number of GMM components can be alternatively optimised through Bayesian Information Criterion (BIC) score. In order to make this method as general and easy to use as possible, we instead provide a set of reasonable defaults that we find work well across different data sets and settings. Default parameters: $max\_depth = \lceil \log_2 N_f \rceil - 1$, where N_f denotes the number of input features, $n\_samples = 0.5 \times size(training\_data)$, $n\_components \in [1, 5]$ — pick the one with the lowest BIC score.

In addition, we observe the fact that DeGeTs framework goes beyond applied Generative Trees and GMMs. This is because the data splitting part can, in fact, be performed by other methods, such as clustering. Consequently, GMMs can be substituted by any other generative models.

Experiments

We follow recent literature (e.g. (Johansson, Shalit, and Sontag 2016; Shalit, Johansson, and Sontag 2017; Yao et al. 2018)) in terms of incorporated data sets and evaluation metrics. We start with defining the later as different data sets use different sets of metrics.

There are a few aspects we aim to investigate. Firstly, how the established reweighing methods perform in individual treatment effect estimation. Secondly, how the choice of model class impacts estimation accuracy (misspecification). Thirdly, how our proposed method affects the performance of the base learners, and how it compares to other methods. Finally, we also study how our method influences the number of rules in prunned decision trees as an indirect measure of data complexity.

Although we do perform hyperparameter search to some extent in order to get reasonable results, it is not our goal to achieve the best results possible, hence the parameters used here are likely not optimal and can be improved upon more extensive search. The main reason is the setups presented as part of this work are intended to be as general as possible. This is why in our analysis we specifically focus on the relative difference in performance between settings rather than comparing them to absolute state-of-the-art results.

Evaluation Metrics

The main focus of utilised metrics here is on the quantification of the errors made by provided predictions. Thus, the metrics are usually denoted as ϵ_X, which translates to the amount of error made with respect to prediction type X (lower is better). In terms of treatment outcomes, $\mathcal{Y}_t^{(i)}$ and $\hat{y}_t^{(i)}$ denote true and predicted outcomes respectively for treatment t and individual i. Thus, following the definition of ITE (Eq. (1)), the difference $\mathcal{Y}_1^{(i)} - \mathcal{Y}_0^{(i)}$ gives a true effect, whereas $\hat{y}_1^{(i)} - \hat{y}_0^{(i)}$ a predicted one. Following this, we can define Precision in Estimation of Heterogeneous Effect (PEHE), which is the root mean squared error between predicted and true effects:

$$\epsilon_{PEHE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_1^{(i)} - \hat{y}_0^{(i)} - (\mathcal{Y}_1^{(i)} - \mathcal{Y}_0^{(i)}))^2} \quad (4)$$

Following the definition of ATE (Eq. (2)), we measure the error on ATE as the absolute difference between predicted and true average effects, formally written as:

$$\epsilon_{ATE} = \left| \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_1^{(i)} - \hat{y}_0^{(i)}) - \frac{1}{n}\sum_{i=1}^{n}(\mathcal{Y}_1^{(i)} - \mathcal{Y}_0^{(i)}) \right| \quad (5)$$

Given a set of treated subjects T that are part of sample E coming from an experimental study, and a set of control group C, define the true Average Treatment effect on the Treated (ATT) as:

$$ATT = \frac{1}{|T|}\sum_{i \in T}\mathcal{Y}^{(i)} - \frac{1}{|C \cap E|}\sum_{i \in C \cap E}\mathcal{Y}^{(i)} \quad (6)$$

The error on ATT is then defined as the absolute difference between the true and predicted ATT:

$$\epsilon_{ATT} = \left| ATT - \frac{1}{|T|} \sum_{i \in T} (\hat{y}_1^{(i)} - \hat{y}_0^{(i)}) \right| \quad (7)$$

Define policy risk as:

$$\mathcal{R}_{pol} = 1 - (\mathbb{E}[\mathcal{Y}_1 | \pi(x) = 1] \mathcal{P}(\pi(x) = 1) + \mathbb{E}[\mathcal{Y}_0 | \pi(x) = 0] \mathcal{P}(\pi(x) = 0) \quad (8)$$

Where $\mathbb{E}[.]$ denotes mathematical expectation and policy π becomes $\pi(x) = 1$ if $\hat{y}_1 - \hat{y}_0 > 0$; $\pi(x) = 0$ otherwise.

Data

We incorporate a set of well-established causal inference benchmark data sets that are briefly described in the following paragraphs and summarised in Table 1.

IHDP. Introduced by Hill (2011), based on Infant Health Development Program (IHDP) clinical trial (Brooks-Gunn, Liaw, and Klebanov 1992). The experiment measured various aspects of premature infants and their mothers, and how receiving specialised childcare affected the cognitive test score of the infants later on. We use a semi-synthetic version of this data set, where the outcomes are simulated through the NPCI package[1] (setting 'A') based on real pre-treatment covariates. Moreover, the treatment groups are made imbalanced by removing a subset of the treated individuals. We report errors on estimated PEHE and ATE averaged over 1,000 realisations and split the data with 90/10 training/test ratios.

JOBS. This data set, proposed by A. Smith and E. Todd (2005), is a combination of the experiment done by LaLonde (1986) as part of the National Supported Work Program (NSWP) and observational data from the Panel Study of Income Dynamics (PSID) (Dehejia and Wahba 2002). Overall, the data captures people's basic characteristics, whether they received a job training from NSWP (treatment), and their employment status (outcome). Here, we report ϵ_{ATT} and $\mathcal{R}_{pol}$ averaged over 10 runs with 80/20 training/test ratio splits.

NEWS. Introduced by Johansson, Shalit, and Sontag (2016), which consists of news articles in the form of word counts with respect to a predefined vocabulary. The treatment is represented as the device type (mobile or desktop) used to view the article, whereas the simulated outcome is defined as the user's experience. Similarly to IHDP, we report PEHE and ATE errors for this data set, averaging over 50 realisations with 90/10 training/test ratio splits.

TWINS. The data set comes from official records of twin births in the US in years 1989-1991 (Almond, Chay, and Lee 2005). The data are preprocessed to include only individuals of the same sex and where each of them weight less than 2,000 grams. The treatment is represented as whether the individual is the heavier one of the twins, whereas the outcome is the mortality within the first year of life. As both factual and counterfactual outcomes are known from the official records, that is, mortality of both twins, one of the twins is intentionally hidden to simulate an observational setting. Here, we incorporate the approach taken by Louizos et al. (2017), where new binary features are created and flipped at random (0.33 probability) in order to hide confounding information. We report ϵ_{ATE} and ϵ_{PEHE} for this data set, averaged over 10 iterations with 80/20 training/test ratio splits.

data set	# samples (t/c)	# features	outcome
IHDP	747 (139/608)	25	cont.
JOBS	3,212 (297/2,915)	17	binary
NEWS	5,000 (2,289/2,711)	3,477	cont.
TWINS	11,984 (5,992/5,992)	194	binary

Table 1: A summary of incorporated data sets. t/c denote the amount of treated and control samples respectively.

Setup

We incorporate the following estimators.

Base Learners. Linear methods: Lasso (l1) and Ridge (l2). Simple Trees: prunned Decision Trees, Extremely Randomised Trees (ET) (Geurts, Ernst, and Wehenkel 2006). Gradient Boosted Trees: CatBoost[2], LightGBM (Ke et al. 2017). Kernel Ridge regression with nonlinearities. Dummy regressor returning the mean as a reference only.

Reweighing Methods. Causal Forest (Athey, Tibshirani, and Wager 2019), Double Machine Learning (DML) (Chernozhukov et al. 2018), and Meta-Learners (Künzel et al. 2019) in the form of T and X variations.

Debiasing Generative Trees. Our proposed method. We include the stronger performing DeGeF variation.

A general approach throughout all conducted experiments was to train a method on the training set and evaluate it against appropriate metrics on the test set. 5 base learners were trained and evaluated in that way: l1, l2, Simple Trees, Boosted Trees and Kernel Ridge. DML and Meta-Learners were combined with different base learners as they need them to solve intermediate regression and classification tasks internally. This resulted in $3 \times 5 = 15$ combinations of distinct estimators. Similarly, DeGeF was combined with the same 5 base learners to investigate how they react to our data augmentation method. Causal Forest and dummy regressor were treated as standalone methods. Overall, we obtained 27 distinct estimators per each data set. In terms of Simple and Boosted Trees, we defaulted to ETs and CatBoost respectively. For NEWS, due to its high-dimensionality, we switched to computationally less expensive Decision Trees and LightGBM instead.

As our DeGeF method is a data augmentation approach, it affects only the training set that is later used by base learners. It does not change the test set in any way as the test portion is used specifically for evaluation purposes to test how methods generalise to unseen data examples. More specifically, DeGeF injects new data samples to the existing training set, and that augmented training set is then provided to base learners.

[1] https://github.com/vdorie/npci

[2] https://github.com/catboost/catboost

Hyperparameter search was also performed wherever applicable, though not too extensive to keep our study as general and accessible as possible. The following is a list of base learners and their hyperparameters we explored. ETs: *max_leaf_nodes* $\in \{10, 20, 30, None\}$, *max_depth* $\in \{5, 10, 20\}$. Kernel Ridge: *alpha* $\in \{0, 1e-1, 1e-2, 1e-3\}$, *gamma* $\in \{1e-2, 1e-1, 0, 1e+1, 1e+2\}$, *kernel* $\in \{rbf, poly\}$, *degree* $\in \{2, 3, 4\}$. CatBoost: *depth* $\in \{6, 8, 10\}$, *l2_leaf_reg* $\in \{1, 3, 10, 100\}$. LightGBM: *max_depth* $\in \{5, 7, 10\}$, *reg_lambda* $\in \{0, 0.1, 1, 5, 10\}$. Causal Forest: *max_depth* $\in \{5, 10, 20\}$. For ETs, CatBoost, LightGBM and Causal Forest we set the number of inner estimators to 1000. To find the best set of hyperparameters, we performed 5-fold cross-validation. When it comes to DeGeF, we set the number of estimators to 10. The other parameters, like number of new samples, tree depth and GMM components, were set to defaults as recommended in the description of the framework. All randomisation seeds were set to a fixed number (1) throughout all experiments.

Most of our experimental runs were performed on a Linux based machine with 12 CPUs and 60 GBs of RAM. More demanding settings, such as NEWS combined with tree-based methods, were delegated to one with 96 CPUs and 500 GBs of RAM.

Results

We incorporate the following estimator names throughout the presented tables: **l1** - Lasso, **l2** - Ridge, **kr** - Kernel Ridge, **dt** - Decision Tree, **et** - Extremely Randomised Trees, **cb** - CatBoost, **lgbm** - LightGBM, **cf** - Causal Forest, **dml** - Double Machine Learning, **xl** - X-Learner, **degef** - our DeGeF method. Combinations of the methods are denoted with a hyphen, for instance, 'dml-l1'.

Due to space limitations, we show only the most important results and exclude very similar ones. More specifically, we find that *l1* and *l2* perform similarly and hence include only the one with better performance per given setting. *T-Learner* achieved performance close to the *X-Learner*, leaving only the latter in the final results. Wherever possible, we default to stronger learners, such as CatBoost and ETs, but we fall back to Decision Trees and LightGBM if necessary (NEWS). We further limit the more advanced methods, namely *dml*, *xl* and *degef*, to only those results that achieved the best improvements and the worst decreases in performance when compared to the base learners.

Tables 2 - 5 present the main results, where we specifically focus on: a) relevant to a given data set metrics, and b) changes in performance relative to a particular base learner. The latter is calculated as $((r_a - r_b)/r_b) \times 100\%$, where r_a and r_b denote results of advanced methods and base learners respectively. The reason for analysing these relative changes rather than absolute values is because in this study we are specifically interested in how more complex approaches (including ours) affect the performance of the base learners, even if not reaching state-of-the-art results. Furthermore, Table 6 shows the number of rules obtained from a prunned Decision Tree while trained on original data and augmented by *degef*. All presented numbers (excluding relative percentages) denote means and 95% confidence intervals.

name	ϵ_{ATE}	$\Delta\%$	ϵ_{PEHE}	$\Delta\%$
dummy	4.408 ± 0.103	-	7.898 ± 0.473	-
l2	0.974 ± 0.104	-	5.786 ± 0.514	-
kr	0.356 ± 0.031	-	2.276 ± 0.170	-
et	0.519 ± 0.074	-	3.093 ± 0.322	-
cb	0.404 ± 0.038	-	2.179 ± 0.210	-
cf	0.397 ± 0.045	-	3.387 ± 0.318	-
dml-l2	0.381 ± 0.040	-61	7.859 ± 0.691	36
dml-cb	1.123 ± 0.052	178	6.976 ± 0.580	220
xl-l1	0.282 ± 0.034	-71	7.660 ± 0.678	32
xl-cb	0.388 ± 0.044	-4	6.894 ± 0.604	216
degef-l2	1.093 ± 0.107	12	5.820 ± 0.514	1
degef-et	0.394 ± 0.052	-24	2.818 ± 0.273	-9
degef-cb	0.328 ± 0.032	-19	2.013 ± 0.190	-8

Table 2: IHDP results.

name	ϵ_{ATT}	$\Delta\%$	$\mathcal{R}_{pol}$	$\Delta\%$
dummy	0.029 ± 0.000	-	0.326 ± 0.000	-
l1	0.005 ± 0.000	-	0.296 ± 0.000	-
kr	0.017 ± 0.000	-	0.400 ± 0.000	-
et	0.006 ± 0.000	-	0.276 ± 0.000	-
cb	0.026 ± 0.000	-	0.308 ± 0.000	-
cf	0.025 ± 0.000	-	0.294 ± 0.000	-
dml-kr	0.007 ± 0.000	-61	0.374 ± 0.000	-7
dml-et	0.099 ± 0.000	1686	0.353 ± 0.000	28
xl-l1	0.022 ± 0.000	361	0.356 ± 0.000	20
xl-kr	0.003 ± 0.000	-81	0.279 ± 0.000	-30
degef-l1	0.054 ± 0.012	1010	0.296 ± 0.000	0
degef-kr	0.019 ± 0.012	12	0.299 ± 0.013	-25
degef-cb	0.019 ± 0.007	-27	0.257 ± 0.030	-17

Table 3: JOBS results.

Discussion

In terms of IHDP data set (Table 2), the reweighing methods strongly improve in ATE in their best cases, but can also be unstable as it is the case with *dml-cb*. Against PEHE, the situation is much worse as those methods significantly decrease in performance when compared to the base learners, even in their best cases, not to mention catastrophic setbacks in the worst cases. Our *degef*, on the other hand, improves in both ATE and PEHE in best cases. Even the worst example (*degef-l2*), is still very stable and does not destroy the predictions as it happened with the reweighing approaches.

In the JOBS data set (Table 3), classic methods again achieve strong improvements in average effect estimation (ATT) in best cases, though they can be substantially worse as well (e.g. *dml-et*). In policy predictions, an equivalent of ITE, *xl-kr* manages to obtain a good improvement. With respect to *degef*, it can also worsen the quality of predictions in ATT, as shown with *degef-l1*, though it does not get as bad as with *dml-et*. However, even in that worst example, policy predictions are not destroyed. The best cases, on the other hand, achieve strong improvements in policy. Similarly to IHDP, here *degef* provided solid improvements in ITE predictions (policy), while staying on par with traditional methods in ATT, obtaining reasonable improvements and keeping

name	ϵ_{ATE}	$\Delta\%$	ϵ_{PEHE}	$\Delta\%$
dummy	0.033 ± 0.002	-	0.318 ± 0.004	-
l1	0.042 ± 0.000	-	0.319 ± 0.004	-
kr	0.045 ± 0.001	-	0.320 ± 0.004	-
et	0.027 ± 0.006	-	0.322 ± 0.003	-
cb	0.039 ± 0.000	-	0.319 ± 0.004	-
cf	0.064 ± 0.001	-	0.323 ± 0.005	-
dml-l1	0.028 ± 0.003	-34	0.318 ± 0.004	0
dml-cb	0.078 ± 0.011	100	0.328 ± 0.002	3
xl-l2	0.042 ± 0.001	-11	0.335 ± 0.010	5
xl-et	0.050 ± 0.001	85	0.323 ± 0.006	1
degef-kr	0.033 ± 0.004	-27	0.320 ± 0.004	0
degef-et	0.054 ± 0.007	97	0.335 ± 0.002	4

Table 4: TWINS results.

name	ϵ_{ATE}	$\Delta\%$	ϵ_{PEHE}	$\Delta\%$
dummy	2.714 ± 0.212	-	4.381 ± 0.361	-
l1	0.244 ± 0.068	-	3.370 ± 0.365	-
dt	0.344 ± 0.076	-	2.717 ± 0.277	-
kr	0.715 ± 0.133	-	3.316 ± 0.367	-
lgbm	0.162 ± 0.045	-	2.074 ± 0.241	-
cf	0.544 ± 0.089	-	3.907 ± 0.481	-
dml-l1	0.233 ± 0.062	-4	2.469 ± 0.269	-27
dml-dt	4.523 ± 0.783	1216	5.875 ± 0.676	116
xl-l2	0.174 ± 0.036	-33	4.162 ± 0.345	23
xl-kr	0.229 ± 0.112	-68	2.695 ± 0.297	-19
degef-l2	0.178 ± 0.041	-32	3.366 ± 0.362	0
degef-dt	0.355 ± 0.080	3	2.727 ± 0.266	0
degef-kr	0.582 ± 0.102	-19	3.256 ± 0.349	2

Table 5: NEWS results.

the worst cases still better than the worst *dml*.

TWINS data set (Table 4), proved to be very difficult for all considered methods when it comes to PEHE, though they did not worsen the predictions as well. Some good improvements in ATE can be observed, but also noticeable decreases in performance in the worst cases. Our method behaves similarly to the classic ones, offering stable gains and keeping the decreases in reasonable bounds.

The last data set, NEWS (Table 5), showed the reweighing approaches can provide some improvements in PEHE as well, at least in their best efforts, though performance decreases are also noticeable in the worst ones. They also offer quite stable improvements in ATE, except extremely poor *dml-dt*. Our proposed method offers reasonable gains in ATE as well, while keeping performance decreases at bay even in the worst efforts. Even though *degef* does not improve in PEHE, it does not destroy individualised predictions either, making it more stable than reweighing methods.

In general terms, the results show that performance can vary substantially depending on the model class, even within the same advanced method (*dml, xl, degef*). Our proposed technique usually offers significant improvements in ITE predictions in best cases, often better than reweighing methods, while keeping the predictions stable even in the worst examples. Classic methods are clearly strong in ATE estimates, but can also struggle in individualised predictions. Overall, these methods (*dml, xl*) proved to be less stable than ours, where the worst cases can perform quite poorly, especially *dml*.

data set	dt	degef-dt
IHDP	33.6 ± 2.0	53.3 ± 2.6
JOBS	6.0 ± 0.0	11.3 ± 5.3
TWINS	9.6 ± 0.9	59.1 ± 11.9
NEWS	19.4 ± 2.5	32.0 ± 4.7

Table 6: Number of rules in a prunned Decision Tree with and without *degef* augmentation.

We also investigate the number of rules in prunned Decision Trees as a proxy for data complexity. As presented in Table 6, *degef* significantly increases the amount of rules across all data sets, translating to an increase in data complexity. This proves the undersmoothing effect we aim for has been achieved.

In terms of possible limitations of our method, we assume the data sets we work with have relatively low noise levels. This is because in noisy environments, the inner GMMs would likely pick up a lot of noise and thus sampling from them would result in even more noisy data samples. The result would be the opposite of what we aim for, that is, to increase data complexity and bring new informative samples, not to introduce bias in the form of noise. Thus, our method would likely worsen base learners performance in such environments. Furthermore, we expect extremely high-dimensional data sets may cause computational issues due to the increasing depth of the inner trees. This is partly why setting a reasonable depth limit is important.

Conclusions

Treatment effect estimation tasks are often subject to the covariate shift problem that is exhibited by discrepancies between observational and interventional distributions. This leads to model misspecification, which we tackle directly in this work by introducing a novel data augmentation method based on generative trees that provides an undersmoothing effect and helps downstream estimators achieve better robustness, ultimately leading to less biased estimators. Through our experiments, we show that the choice of model class matters, and that reweighing methods can struggle in individualised effect estimation. Our proposed approach presented competitive results with existing reweighing procedures on average effect tasks while offering significantly better performance improvements on individual effect problems.

In terms of possible future directions, it might be interesting to investigate the feasibility of replacing generative trees with neural networks to handle extremely high-dimensional problems. Another direction would be to instantiate *DeGeTs* framework with alternative methods, such as standard clustering and generative neural networks. Lastly, extending our approach to noisy data sets would likely increase its potential applicability to real world problems.

References

A. Smith, J.; and E. Todd, P. 2005. Does Matching Overcome LaLonde's Critique of Nonexperimental Estimators? *Journal of Econometrics*, 125(1-2): 305–353.

Almond, D.; Chay, K. Y.; and Lee, D. S. 2005. The Costs of Low Birth Weight. *The Quarterly Journal of Economics*, 120(3): 1031–1083.

Athey, S.; Imbens, G.; Metzger, J.; and Munro, E. 2020. Using Wasserstein Generative Adversarial Networks for the Design of Monte Carlo Simulations. *arXiv:1909.02210 [econ, stat]*.

Athey, S.; Tibshirani, J.; and Wager, S. 2019. Generalized Random Forests. *Annals of Statistics*, 47(2): 1148–1178.

Brooks-Gunn, J.; Liaw, F. R.; and Klebanov, P. K. 1992. Effects of Early Intervention on Cognitive Function of Low Birth Weight Preterm Infants. *The Journal of Pediatrics*, 120(3): 350–359.

Chawla, N. V.; Bowyer, K. W.; Hall, L. O.; and Kegelmeyer, W. P. 2002. SMOTE: Synthetic Minority over-Sampling Technique. *Journal of Artificial Intelligence Research*, 16(1): 321–357.

Chernozhukov, V.; Chetverikov, D.; Demirer, M.; Duflo, E.; Hansen, C.; Newey, W.; and Robins, J. 2018. Double/Debiased Machine Learning for Treatment and Structural Parameters. *The Econometrics Journal*, 21(1): C1–C68.

Chernozhukov, V.; Chetverikov, D.; Demirer, M.; Duflo, E.; Hansen, C.; and Newey, W. K. 2016. Double Machine Learning for Treatment and Causal Parameters. Technical Report CWP49/16, Centre for Microdata Methods and Practice, Institute for Fiscal Studies.

Correia, A.; Peharz, R.; and de Campos, C. P. 2020. Joints in Random Forests. In *Advances in Neural Information Processing Systems*, volume 33, 11404–11415. Curran Associates, Inc.

Dehejia, R. H.; and Wahba, S. 2002. Propensity Score-Matching Methods For Nonexperimental Causal Studies. *The Review of Economics and Statistics*, 84(1): 151–161.

Foster, D. J.; and Syrgkanis, V. 2020. Orthogonal Statistical Learning. *arXiv:1901.09036 [cs, econ, math, stat]*.

Geurts, P.; Ernst, D.; and Wehenkel, L. 2006. Extremely Randomized Trees. *Machine Learning*, 63(1): 3–42.

Guo, R.; Cheng, L.; Li, J.; Hahn, P. R.; and Liu, H. 2020. A Survey of Learning Causality with Data: Problems and Methods. *ACM Computing Surveys*, 53(4): 75:1–75:37.

He, H.; Bai, Y.; Garcia, E. A.; and Li, S. 2008. ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 1322–1328.

Hill, J. L. 2011. Bayesian Nonparametric Modeling for Causal Inference. *Journal of Computational and Graphical Statistics*, 20(1): 217–240.

Johansson, F. D.; Shalit, U.; and Sontag, D. 2016. Learning Representations for Counterfactual Inference. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, 3020–3029. New York, NY, USA: JMLR.org.

Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; and Liu, T.-Y. 2017. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Künzel, S. R.; Sekhon, J. S.; Bickel, P. J.; and Yu, B. 2019. Meta-Learners for Estimating Heterogeneous Treatment Effects Using Machine Learning. *Proceedings of the National Academy of Sciences*, 116(10): 4156–4165.

LaLonde, R. J. 1986. Evaluating the Econometric Evaluations of Training Programs with Experimental Data. *The American Economic Review*, 76(4): 604–620.

Louizos, C.; Shalit, U.; Mooij, J. M.; Sontag, D.; Zemel, R.; and Welling, M. 2017. Causal Effect Inference with Deep Latent-Variable Models. *Advances in Neural Information Processing Systems*, 30.

Neal, B.; Huang, C.-W.; and Raghupathi, S. 2021. RealCause: Realistic Causal Inference Benchmarking. *arXiv:2011.15007 [cs, stat]*.

Newey, W.; Hsieh, F.; and Robins, J. 1998. Undersmoothing and Bias Corrected Functional Estimation. Working Paper 98-17, Massachusetts Institute of Technology (MIT), Department of Economics.

Nie, X.; and Wager, S. 2020. Quasi-Oracle Estimation of Heterogeneous Treatment Effects. *arXiv:1712.04912 [econ, math, stat]*.

Pearl, J. 2009. *Causality*. Cambridge university press.

Perez, L.; and Wang, J. 2017. The Effectiveness of Data Augmentation in Image Classification Using Deep Learning. *arXiv:1712.04621 [cs]*.

Peters, J.; Janzing, D.; and Schölkopf, B. 2017. *Elements of Causal Inference: Foundations and Learning Algorithms*. The MIT Press. ISBN 978-0-262-03731-0.

Robins, J. M.; Rotnitzky, A.; and Zhao, L. P. 1994. Estimation of Regression Coefficients When Some Regressors Are Not Always Observed. *Journal of the American Statistical Association*, 89(427): 846–866.

Rosenbaum, P. R.; and Rubin, D. B. 1983. The Central Role of the Propensity Score in Observational Studies for Causal Effects. *Biometrika*, 70(1): 41–55.

Rubin, D. B. 1974. Estimating Causal Effects of Treatments in Randomized and Nonrandomized Studies. *Journal of Educational Psychology*, 66(5): 688–701.

Shalit, U.; Johansson, F. D.; and Sontag, D. 2017. Estimating Individual Treatment Effect: Generalization Bounds and Algorithms. In *International Conference on Machine Learning*, 3076–3085. PMLR.

Wen, J.; Yu, C.-N.; and Greiner, R. 2014. Robust Learning under Uncertain Test Distributions: Relating Covariate Shift to Model Misspecification. In *International Conference on Machine Learning*, 631–639. PMLR.

White, H. 1981. Consequences and Detection of Misspecified Nonlinear Regression Models. *Journal of the American Statistical Association*, 76(374): 419–433.

Yao, L.; Chu, Z.; Li, S.; Li, Y.; Gao, J.; and Zhang, A. 2020. A Survey on Causal Inference. *arXiv:2002.02770 [cs, stat]*.

Yao, L.; Li, S.; Li, Y.; Huai, M.; Gao, J.; and Zhang, A. 2018. Representation Learning for Treatment Effect Estimation from Observational Data. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, 2638–2648. Montréal, Canada: Curran Associates Inc.