Multi-valued Treatment Effect Estimation for Health Technology Assessment with a Neural Network

Borja Velasco-Regúlez, Jesus Cerquides, Josep Lluis Arcos
Department of Machine Learning
Artificial Intelligence Research Institute (IIIA), CSIC
Campus UAB, Bellatera, Spain
bvelasco@iiia.csic.es

Abstract

The clinical effectiveness aspect within the Health Technology Assessment process often faces causal questions where the treatment variable can take multiple values. Nevertheless, most developments in causal inference algorithms that employ machine learning happen in binary treatment settings. In addition, there is a big gap between the algorithmic state of the art and the applied state of the art in this field. In this paper, we select a state-of-the-art, neural network-based algorithm for binary treatment effect estimation and generalize it to a multi-valued treatment setting, testing it with semi-synthetic data that could mimic an HTA process. We obtain an estimator with desirable asymptotic properties and good results in experiments. To the best of our knowledge, this work is opening ground for the benchmarking of neural network-based algorithms for multi-valued treatment effect estimation.

1 Introduction

Health Technology Assessment (HTA) is a multidisciplinary process that uses systematic and explicit methods to evaluate the properties and effects of a health technology [9]. This task is of crucial importance for the technical and financial sustainability of healthcare systems. Within the process of HTA, clinical effectiveness is defined as the comparison between several health strategies or technologies regarding an outcome of interest [5]. Usually (but not always), one of those strategies or technologies is an existing standard, and the other (or others) are new. As it can be sensed, this comparison is a causal question. In front of such type of questions, HTA agencies usually employ the classical hierarchic pyramid of evidence, which puts the evidence generated by Randomized Controlled Trials (RCT) at the cuspid. Nevertheless, those agencies are facing nowadays more and more questions in which results from RCT may not be sufficient or arrive in time [11]. This, together with the exponential growth of Real-World Data (RWD) [7], constitutes a great opportunity for causal inference methods.

In the literature, state of the art, machine learning-based algorithms for causal inference are mostly developed [14][6] and benchmarked [8][13] in binary treatment settings. The HTA process, on the contrary, happens more often than not in scenarios with multiple strategies or technologies (treatments) [2]. This is a limitation for the potential application of those algorithms for the task of HTA. In the present work, we select a state of the art, neural network-based, binary treatment effect estimation algorithm named Dragonnet[12], and we develop and test a generalized version for n-valued treatment settings. We present the theoretical and mathematical formulation, we develop a framework for experiments, and we provide results obtained in different scenarios employing semi-synthetic, HTA-related data. To the best of our knowledge, this is the first attempt to establish a benchmark for this type of algorithm in multivalued-treatment settings. The code and the data can be found in the following anonymized repository: `https://anonymous.4open.science/r/Hydranet-95F8/`

NeurIPS 2022Workshop on Causal Machine Learning for Real-World Impact (CML4Impact 2022).

2 Problem statement

Let the health technology or health strategy (from now on, for simplicity, the treatment) of interest be a discrete random variable $T \in [0..k]$ that can take $k+1$ different values. Let the outcome be a continuous random variable $Y \in \mathbb{R}$, and let the covariates (i.e. the variables affecting both the treatment and the outcome) be a random vector $X \in \mathbb{R}^j$. Thus, our set of data points is $(Y_i, T_i, X_i),\ i \in [1..N]$, generated independently and identically. This set of data points constitutes our body of observational data. We define the causal effect of the treatment t over the outcome Y as, $\mu_t = \mathbb{E}[Y|do(T=t)]$, using Pearl's *do-calculus* notation [10], which denotes intervention. It can be shown that, if our data meets certain conditions, we can estimate causal (interventional) quantities based on observational data. Those conditions are known as the identifiability conditions: positivity, consistency and "no hidden confounder" conditions. For a more detailed explanation, see [10]. Under such conditions, $\mu_t = \mathbb{E}[Y|X=x, T=t]$, which is a quantity that is inferrable from our body of observational data. Along the rest of sections 2 and 3 we will assume that the identifiability conditions are fulfilled.

We define the conditional outcome as the expectation of the outcome given the treatment and the covariates, $Q(t,x) = \mathbb{E}[Y|t,x]$. Based on Q, we can construct a simple estimator $\hat{\mu}_t$ of μ_t as $\hat{\mu}_t = \frac{1}{N}\sum_i Q(t,x_i)$. In the following, we will be interested in approximating Q. Let $\hat{Q}$ be an approximation of Q. We define $\mu_t^{\hat{Q}} = \frac{1}{N}\sum_i \hat{Q}(t,x_i)$ as the estimator of μ_t obtained replacing Q by its estimation $\hat{Q}$. Furthermore, we define the Generalized Propensity Score (GPS[1]), expressed as $\mathbf{G}(x) = [g_0(x), g_1(x), \ldots, g_k(x)] \in \mathbb{R}^{k+1}$, with $g_t(x) = P(T=t|x)$.

In a binary treatment setting, under the identifiability conditions, the Average Treatment Effect (ATE) is one of the most common causal quantities of interest, and it is defined as $\psi = \mu_1 - \mu_0$. Given an approximation $\hat{Q}$ of Q, we could easily estimate ψ as $\psi^{\hat{Q}} = \mu_1^{\hat{Q}} - \mu_0^{\hat{Q}}$. In a multi-valued treatment setting, a wider class of causal quantities of interest can be defined, and all the conditional outcomes must be computed in order to obtain valid estimates of those quantities[1]. In this work, we define such quantities of interest as the pair-wise average differences between the several treatments and a treatment considered the control (note that, in practice, the control treatment does not necessarily mean absence of treatment). Thus, we define a vector of ATEs $\boldsymbol{\psi} \in \mathbb{R}^k$, $\boldsymbol{\psi} = [\psi_1, \psi_2, \ldots, \psi_k]$, with $\psi_t = \mu_t - \mu_0$. We can approximate these quantities in a similar fashion as shown before, the t-th element of the vector being $\psi_t^{\hat{Q}} = \mu_t^{\hat{Q}} - \mu_0^{\hat{Q}}$. Note that if the causal quantity of interest was $\psi_{i,j} = \mu_i - \mu_j$, we could easily compute it based on the previous definition, as $\psi_{i,j} = \psi_i - \psi_j$, due to the linearity of the expectation operator.

The subject of interest in this the paper is the estimation of the vector of ATEs $\boldsymbol{\psi}$. In the next section we generalize the estimation method provided in [12], which has the objective of estimating the ATE in the binary case, to the estimation of $\boldsymbol{\psi}$ in the multivalued treatment case presented above.

3 From Dragonnet to Hydranet

Dragonnet is a high-capacity, end-to-end neural network architecture for estimating binary treatment effects[12]. We present here the variation of the architecture, mathematical formulations and proofs for adapting Dragonnet to a multivalued treatment setting. We call this adaptation Hydranet.

3.1 Architecture

The architecture of Hydranet can be seen in Figure 1. It consists of two parts: the representation part, formed by the input layer and two hidden layers, and the heads, formed by $k+2$ ends. Out of those, $k+1$ correspond to the conditional outcomes, and are formed by two more hidden layers plus the output layer. The remaining head corresponds to the GPS, $\mathbf{G}(x) = [g_0(x), g_1(x), \ldots, g_k(x)] \in \mathbb{R}^{k+1}$, with $g_t(x) = P(T=t|x)$, consisting on just the output layer. Recall that we approximate the t-th element of the vector of ATEs as $\psi_t^{\hat{Q}} = \frac{1}{N}\sum_i \hat{Q}(t,x_i) - \hat{Q}(0,x_i)$.

The baseline objective function has the shape

$$\hat{R}(\theta) = \frac{1}{N}\sum_i [(Q^{nn}(t_i,x_i;\theta) - y_i)^2 + \alpha\text{CrossEntropy}(g_t^{nn}(x_i;\theta), t_i)] \tag{1}$$

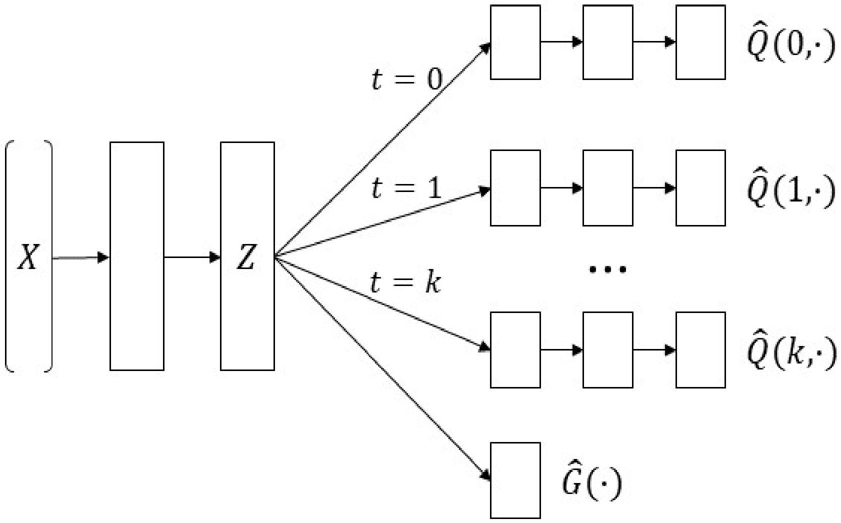


Figure 1: Hydranet architecture

and the model parameters are

$$\hat{\theta} = \arg\min_{\theta}[\hat{R}(\theta)] \tag{2}$$

3.2 Targeted regularization

Now, following the reasoning in [12], we present targeted regularization. Targeted regularization is a modification of the objective function that introduces an extra parameter, epsilon. In our setting, $\boldsymbol{\epsilon}$ is a vector in $\mathbb{R}^k$, $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, ...\epsilon_k)$, and the new objective function is

$$\bar{F}(\theta, \boldsymbol{\epsilon}) = \hat{R}(\theta) + \beta\frac{1}{N}\sum_i \gamma_i(y_i, t_i, x_i; \theta, \boldsymbol{\epsilon}), \text{ where} \tag{3}$$

$$\gamma_i(y_i, t_i, x_i; \theta, \boldsymbol{\epsilon}) = (y_i - \bar{Q}_i(\theta, \boldsymbol{\epsilon}))^2, \text{ and} \tag{4}$$

$$\bar{Q}_i(\theta, \boldsymbol{\epsilon}) = Q^{nn}(t_i, x_i) + \epsilon_1\left(\frac{\mathbf{I}(T=1)}{g_1^{nn}(x_i)} - \frac{\mathbf{I}(T=0)}{g_0^{nn}(x_i)}\right) + \ldots + \epsilon_k\left(\frac{\mathbf{I}(T=k)}{g_k^{nn}(x_i)} - \frac{\mathbf{I}(T=0)}{g_0^{nn}(x_i)}\right) \tag{5}$$

with $\mathbf{I}(T = t)$ the indicator function, and thus the sought model parameters are defined by

$$\hat{\theta}, \hat{\epsilon} = \arg\min_{\theta, \boldsymbol{\epsilon}}[\hat{R}(\theta) + \beta\frac{1}{N}\sum_i \gamma_i(y_i, t_i, x_i; \theta, \boldsymbol{\epsilon})] \tag{6}$$

But why this modification in the first place? The answer lies in semiparametric estimation theory (SET) and targeted maximum likelihood estimation (TMLE). Very generally, SET provides us with conditions that ensure desirable properties of our estimator psi when they are fulfilled, and TMLE is an efficient method to impose the fulfillment of those conditions. The conditions are the set of non-parametric estimating equations, defined as

$$\mathbf{0} = \left[\frac{1}{N}\sum_i \varphi_{i,1}, \frac{1}{N}\sum_i \varphi_{i,2}, ...\frac{1}{N}\sum_i \varphi_{i,k}\right] \tag{7}$$

and they employ the elements of the vector of efficient influence curves, defined as $\boldsymbol{\varphi} \in \mathbb{R}^k, \boldsymbol{\varphi} = [\varphi_1, \varphi_2, ...\varphi_k]$, with

$$\varphi_{i,t} = Q^{nn}(t, x_i) - Q^{nn}(0, x_i) + \left(\frac{\mathbf{I}(T=t)}{g_t^{nn}(x_i)} - \frac{\mathbf{I}(T=0)}{g_0^{nn}(x_i)} \right) (y_i - Q^{nn}(t, x_i)) - \psi_t \tag{8}$$

Finally, recall that what we want is that the minimization of the modified objective function ensures the fulfillment of the non-parametric estimation equations. This can be expressed mathematically as

$$\mathbf{0} = \nabla \bar{F}\big|_{\hat{\boldsymbol{\epsilon}}} = \left[\frac{\partial \bar{F}}{\partial \epsilon_1}, \frac{\partial \bar{F}}{\partial \epsilon_2}, ... \frac{\partial \bar{F}}{\partial \epsilon_k} \right] \bigg|_{\hat{\boldsymbol{\epsilon}}} = \left[\frac{\beta}{N} \sum_i \varphi_{i,1}, \frac{\beta}{N} \sum_i \varphi_{i,2}, ... \frac{\beta}{N} \sum_i \varphi_{i,k} \right] \tag{9}$$

and the proof can be found in the supplementary material. This warrants the aforementioned desirable properties of the estimator psi, i.e. double robustness, fast convergence, and efficiency.

4 The data and the metrics

We have designed and implemented algorithms mimicking different Data Generating Processes (DGP) to produce semi-synthetic datasets to test Hydranet. In this context, semi-synthetic means that the covariates are constituted by variables from a study with real participants, while the treatments and the outcomes have been artificially generated. The covariates are a subset of the variables of the IHDP dataset [4] [3]. This data was collected for a real RCT carried out in 1985, and it is routinely used for benchmarking causal inference algorithms. Despite it may contain some variables that could be considered sensitive, we have added noise and discarded some of them, so we consider that potential privacy-related issues have a low risk. Based on those covariates, we have generated the multi-valued treatments and the continuous outcomes, in a similar fashion to [3], but adapting the DGP to our needs. Below we explain the basics of these DGP, and a more detailed explanation can be found in the supplementary material.

For computing the treatment, we have created a function $m : \mathcal{X} \to [0..k]$ that maps a given subset of the covariates to a specific treatment in a deterministic way. Then, in order to fulfill the positivity condition, we have drawn the treatment from a categorical distribution such that

$$p(t|X) = \begin{cases} 0.8, & \text{if } t = m(X) \\ \frac{0.2}{k-1}, & \text{otherwise} \end{cases}$$

We have implemented and tested different shapes of m functions. For computing the outcome, we have defined as many outcome functions as treatment values. These functions map some combination of the covariates and the treatment to the output space, $Y = L(T, X) = [l_0(T, X), l_1(T, X), \ldots, l_k(T, X)]$. We have added random noise to each outcome. Thus, we can compute all the potential outcomes $Y^0, Y^1, \ldots, Y^k$ for each data point, i.e., we have the observable and the unobservable (counterfactual) data. This way, we can compute the ground truth, true causal effects, for testing the performance of the algorithms. Of course, the algorithms have been trained only with the observable data, as it would be the case in an observational study. In addition, we have implemented and tested different shapes of the l_t functions: shapes that are partially linear (linear in the treatment and nonlienar in the covariates), and shapes that are fully nonlinear. The advantage of the partially linear ones is that it allows us to easily control the difference between the biased, observable distribution, and the unbiased, unobservable distribution. We have called this difference the bias. We provide more details of this in the supplementary material.

We have generated datasets for the particular cases of 3, 5 and 8-valued treatments. In addition, we have tested the influence over the performance of other parameters such as the dataset size, the bias size and the number of covariates. The details of the DGP can be found in the supplementary material. We have also generated datasets that quasiviolate or fully violate the positivity condition. This is explained in a later subsection.

4.1 Metrics

For performance benchmarking purposes, we have employed the sum of errors of the vector of ATEs. This is computed as the sum of the absolute values of the differences of all the estimated ATEs with respect to their true values, $E = \sum_{t=1}^{k} |\psi_t - \hat{\psi}_t|$. This choice allows to have a single real number as a final result, making comparisons simpler. We have also computed the error relative to the total effect size or total ATE, $tATE = \sum_{t=1}^{k} |\psi_t|$, as a percentage: $EP = \frac{E}{tATE}$, as it gives a better intuition of the goodness of the ATE estimation. All values have been computed as averages across several dataset realizations (50 or 100) to increase the robustness of the results, with 95% confidence intervals computed with Bootstrapping.

5 Experiments and results

In the case of binary treatment settings, there are *de facto* benchmarking datasets and metrics, i.e., datasets and metrics that are widely used in the literature and thus serve for algorithmic comparison purposes. The IHDP dataset and the metrics presented in[3] are an example of this. This is not the case in multi-valued treatment settings; we did not find any suitable comparator, and thus we have not been able to directly compare the performance of Hydranet with the performance of other algorithms. Despite this, we affirm that Hydranet performs well in all the tested scenarios, reaching low or very low error values in all the 3, 5 and 8-valued treatment settings, for different dataset sizes, bias sizes, and number of covariates respectively. These results can be found in the supplementary material.

In each setting we have computed the error and the relative error in the train and test sets, for two algorithms: baseline and targeted regularization (or t-reg). For comparison purposes, we have also computed the results of a naive estimator, i.e., the one obtained with the biased, observable outcomes, not controlled for covariates. This value has been calculated only for the purpose of showing the bias of the observational data without any controlling, and should not be taken as an alternative estimator. We have included the results both in graphics and tables, avoiding the confidence intervals in the tables for simplicity. Note that we have not tested all the possible combinations of treatment values (3, 5, 8) and variables of analysis (dataset size, bias size, number of covariates), but just one combination. Nevertheless, based on the results, we expect the behavior to be similar across scenarios.

The optimizer selection, optimization strategy and other hyperparameters have suffered minor modifications with respect to those in [12]

5.1 Quasiviolation and violation of the positivity condition

Identifiability conditions are one of the main limitations in causal inference with observational data. For this reason, there is a big interest in the causal inference community to build algorithms that are as robust as possible in the presence of quasiviolations of these conditions. We have checked the performance of Hydranet in one of these situations (quasiviolation of the positivity condition), obtaining good results. For this purpose, we have adjusted the probability of getting a particular treatment given the covariates, as

$$p(t|X) = \begin{cases} P, & \text{if } t = m(X) \\ \frac{1-P}{k}, & \text{otherwise} \end{cases}$$

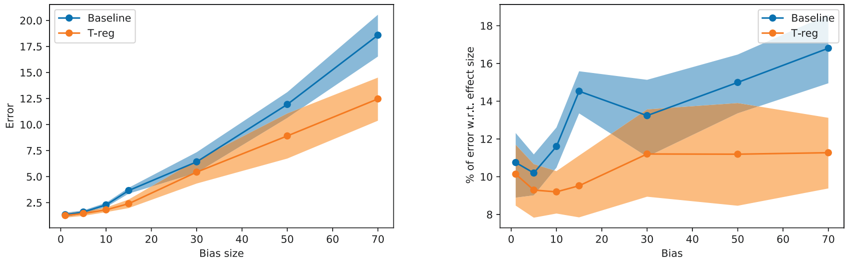
setting P to 0.95 and 1 and thus getting scenarios of quasiviolation and total violation respectively. Table 1 shows the results of the algorithms, where it can be observed that in the total violation scenario all algorithms perform equally bad, while in the quasiviolation scenario Hydranet (both Baseline and T-reg) is able to correctly learn the conditional outcomes and propensity scores, despite having very few examples of some of the treatment-covariate combinations.

Table 1: True effect, absolute error and relative error of the algorithms in scenarios of total violation and quasiviolation of the positivity condition. Both the baseline and the t-reg algorithms perform well.

Setting	True effect	Naive err.	% naive err.	Baseline err.	% baseline err.	T-reg err.	% t-reg err.
Total viol.	37.77	30.26	80.14	25.68	68.01	26.00	68.85
Quasiv.	37.77	12.41	32.87	2.50	6.62	3.47	9.20

Table 2: True effect, absolute error and relative error of the algorithms in realizations where the baseline algorithm has a poor performance on ATE estimation. T-reg. improves the estimation in this scenario.

Bias s.	True effect	Naive err.	% naive err.	Baseline err.	% baseline err.	T-reg err.	% t-reg err.
1	12.49	1.00	7.99	1.34	10.75	1.27	10.14
5	15.70	3.72	23.69	1.60	10.20	1.46	9.29
10	19.70	7.34	37.25	2.29	11.60	1.81	9.20
15	25.22	10.97	43.50	3.66	14.53	2.40	9.52
30	48.51	21.83	45.00	6.42	13.24	5.44	11.20
50	79.60	36.61	45.99	11.94	15.00	8.91	11.19
70	110.58	50.82	45.96	18.59	16.81	12.46	11.27



(a) Absolute error

(b) Error relative to effect size

Figure 2: Errors w.r.t. bias size in realizations where the baseline algorithm has a poor performance on ATE estimation. T-reg improves the estimation in this scenario.

5.2 Why targeted regularization?

In general, the results of Baseline and T-reg are similar. This changes in one particular scenario: when the baseline's estimate of the ATE is poor. In that case, targeted regularization improves the estimation. Figure 2 and table 2 show the absolute and relative errors of baseline and targeted regularization algorithms for varying bias size, for realizations of the dataset where the baseline's estimation of the ATE was poor. We have defined *poor estimates* as those with an error above the mean error computed across all realizations. We have set both baseline and t-reg algorithms to the same initialization, to ensure that the observed improvement is due to targeted regularization. We do not have a theoretical explanation for this phenomenon, but the targeted regularization algorithm yields an estimator that is doubly robust and asymptotically efficient, and this could point towards such explanation.

6 Discussion

In this work, we have generalized a state of the art, neural network-based algorithm for ATE estimation from a binary treatment setting to a n-valued treatment setting. We have derived a doubly-robust and data-efficient estimator, i.e., Hydranet with targeted regularization, obtaining very good results with semi-synthetic datasets generated *ad-hoc*, even in a scenario of quasi-violation of the positivity condition. We show that, despite the generalizability to n-valued treatment settings being possible, it has its own challenges, and the behavior of the algorithms in each particular case requires its own interpretation. As far as we know this paper is opening ground on the proposal and evaluation of methods for benchmarking and estimation of ATEs in multivalued treatment scenarios.

The main limitations of this work are twofold: first, we have not been able to directly compare the performance of our algorithm to others, due to the lack of benchmarking data in the literature. Also, we have not compared our algorithm against k binary-treatment algorithms. Despite we expect our algorithm to perform better based on theoretical results, it would have been interesting to compare the experimental results directly, and thus this is both a limitation and a line of future work. Second, we have only employed one dataset, the IHDP dataset. It is also a line of future work to test the performance over other available adapted semi-synthetic datasets.

With real-world impact in mind, we set another line of future work not in the direction of further algorithmic development but in the direction of usage of the current algorithm for real problems. There exists a big gap between the algorithmic state of the art and the applied state of the art in causal inference: some of the best pieces of applied work (causal inference studies for HTA and/or healthcare) make use of very simple methods such as regressions for modelling the data, and implicitly or explicitly claim that more advanced methods such as neural networks still have to prove their benefits. Despite we believe that this gap is, to some extent, natural, we also believe that the work of bringing the algorithmic state of the art to application is crucial for the potentially positive real-world impact of these methods.

We believe that the main potential negative societal aspect of our work could be the generation of low-quality or wrong evidence in the clinical field, due to inherent limitations of inferring causality with observational data. This is an ongoing debate in the scientific community. Nevertheless, we believe that good evidence can be obtained with causal inference methods by ensuring the highest possible methodological quality of these studies. We will remain attentive to the accumulating evidence about this matter.

Acknowledgments

This work was supported by Doctorat Industrial funded by Generalitat de Catalunya [DI-2020-18] and by project CI-SUSTAIN funded by the Spanish Ministry of Science and Innovation [PID2019-104156GB-I00]. Borja Velasco-Regúlez is a PhD Student of the doctoral program in Computer Science at the Universitat Autonoma de Barcelona.

References

[1] Matias D. Cattaneo. Efficient semiparametric estimation of multi-valued treatment effects under ignorability. *Journal of Econometrics*, 155(2):138–154, April 2010.

[2] Matias D. Cattaneo, David M. Drukker, and Ashley D. Holland. Estimation of Multivalued Treatment Effects under Conditional Independence. *The Stata Journal: Promoting communications on statistics and Stata*, 13(3):407–450, September 2013.

[3] Vincent Dorie, Jennifer Hill, Uri Shalit, Marc Scott, and Dan Cervone. Automated versus do-it-yourself methods for causal inference: Lessons learned from a data analysis competition. *arXiv:1707.02641 [stat]*, July 2018. arXiv: 1707.02641.

[4] Ruth T. Gross. Infant health and development program (ihdp): Enhancing the outcomes of low birth weight, premature infants in the united states, 1985-1988. 1993.

[5] Kristian Lampe, Marjukka Mäkelä, Marcial Velasco Garrido, Heidi Anttila, Ilona Autti-Rämö, Nicholas J. Hicks, Björn Hofmann, Juha Koivisto, Regina Kunz, Pia Kärki, and et al. The hta core model: A novel method for producing and reporting health technology assessments. *International Journal of Technology Assessment in Health Care*, 25(S2):9–20, 2009.

[6] Samuel David Lendle. *Targeted Minimum Loss Based Estimation: Applications and Extensions in Causal Inference and Big Data*. PhD thesis, UC Berkeley, 2015.

[7] Meng Li, Shengqi Chen, Yunfeng Lai, Zuanji Liang, Jiaqi Wang, Junnan Shi, Haojie Lin, Dongning Yao, Hao Hu, and Carolina Oi Lam Ung. Integrating real-world evidence in the regulatory decision-making process: A systematic analysis of experiences in the us, eu, and china using a logic model. *Frontiers in Medicine*, 8, 2021.

[8] Alexander Lin, Amil Merchant, Suproteem K Sarkar, and Alexander D'Amour. Universal Causal Evaluation Engine: An API for empirically evaluating causal inference models. page 9.

[9] Brian O'Rourke, Wija Oortwijn, and Tara Schuller. The new definition of health technology assessment: A milestone in international collaboration. *International Journal of Technology Assessment in Health Care*, 36(3):187–190, 2020.

[10] Judea Pearl, Madelyn Glymour,, and Nicholas P. Jewell. *Causal Inference in Statistics. A Primer*. John Wiley and Sons Ltd, United States, 2016.

[11] Pedro Serrano-Aguilar, Iñaki Gutierrez-Ibarluzea, Pilar Díaz, Iñaki Imaz-Iglesia, Jesús González-Enríquez, José Luis Castro, Mireia Espallargues, Sandra García-Armesto, Paloma Arriola-Bolado, Amado Rivero-Santana, and et al. Postlaunch evidence-generation studies for medical devices in spain: the redets approach to integrate real-world evidence into decision making. *International Journal of Technology Assessment in Health Care*, 37(1):e63, 2021.

[12] Claudia Shi, David M. Blei, and Victor Veitch. Adapting Neural Networks for the Estimation of Treatment Effects. *arXiv:1906.02120 [cs, stat]*, October 2019. arXiv: 1906.02120.

[13] Yishai Shimoni, Chen Yanover, Ehud Karavani, and Yaara Goldschmnidt. Benchmarking Framework for Performance-Evaluation of Causal Inference Analysis. *arXiv:1802.05046 [cs, stat]*, March 2018. arXiv: 1802.05046.

[14] Jinsung Yoon, James Jordon, and Mihaela Van Der Schaar. Ganite: Estimation of individualized treatment effects using generative adversarial nets. In *International Conference on Learning Representations*, 2018.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Discussion Section, paragraph 2.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Discusion Section, paragraph 3.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See the Problem Statement and From Dragonnet to Hydranet Sections.
   (b) Did you include complete proofs of all theoretical results? [Yes] See the Supplementary Material, Section A.1, for the main proof of the paper.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See the Introduction Section. The code and the data can be found in the following anonymized repository: `https://anonymous.4open.science/r/Hydranet-95F8/`. Nevertheless, there are no explicit instructions and thus reproducing the results might be difficult.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Yes, by referencing the work that serves as a base for our paper [12], where the whole setting is defined. See Experiments and Results Section, last paragraph.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] Yes, although only in graphs, not in tables. See Experiments and Results Section, as well as Supplementary Material Section A.3.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] Across the text both the work that serves as a base for our paper [12] as well as the source of the employed data [4] are cited.
   (b) Did you mention the license of the assets? [No] We did not find the license of the assets.
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] These aspects should be found in the original data repository [4].
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] See The data and the metrics Section.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Supplementary material

A.1 Hydranet

We want to prove that

$$\left.\frac{\partial \bar{F}}{\partial \epsilon_t}\right|_{\hat{\epsilon}_t} = \frac{1}{N}\sum_i \varphi_{i,t}, \qquad \forall\, t \text{ in}[0,k] \tag{10}$$

Proof. On one hand, using equations (3), (4) and (5) we get

$$\begin{aligned}
\left.\frac{\partial \bar{F}}{\partial \epsilon_t}\right|_{\hat{\theta},\hat{\epsilon}_t} &= \left.\frac{\partial}{\partial \epsilon_t}\left(\hat{R}(\theta) + \beta\frac{1}{N}\sum_i \gamma_i(y_i, t_i, x_i; \theta, \boldsymbol{\epsilon})\right)\right|_{\hat{\theta},\hat{\epsilon}_t} \\
&= \left.\frac{\beta}{N}\sum_i \frac{\partial}{\partial \epsilon_t}\gamma_i(\theta, \boldsymbol{\epsilon})\right|_{\hat{\theta},\hat{\epsilon}_t} \\
&= \left.\frac{2\beta}{N}\sum_i (y_i - \bar{Q}_i(\theta, \boldsymbol{\epsilon}))\frac{\partial \bar{Q}_i(\theta, \boldsymbol{\epsilon})}{\partial \epsilon_t}\right|_{\hat{\theta},\hat{\epsilon}_t} \\
&= \left.\frac{2\beta}{N}\sum_i \left[(y_i - \bar{Q}_i(\theta, \boldsymbol{\epsilon}))\left(\frac{\mathbf{I}(T=t)}{g_t^{nn}(\theta)} - \frac{\mathbf{I}(T=0)}{g_0^{nn}(\theta)}\right)\right]\right|_{\hat{\theta},\hat{\epsilon}_t} \\
&= \frac{2\beta}{N}\sum_i \left[(y_i - \hat{Q}(t, x_i)\left(\frac{\mathbf{I}(T=t)}{\hat{g}_t} - \frac{\mathbf{I}(T=0)}{\hat{g}_0}\right)\right] \text{(evaluate at } \hat{\theta}, \hat{\epsilon}) \\
&= \frac{2\beta}{N}\sum_i \left(\hat{Q}(t, x_i) - \hat{Q}(0, x_i)\right) - \frac{\beta}{N}\sum_i \left(\hat{Q}(t, x_i) - \hat{Q}(0, x_i)\right) + \\
&\frac{\beta}{N}\sum_i \left[(y_i - \hat{Q}(t, x_i))\left(\frac{\mathbf{I}(T=t)}{\hat{g}_t} - \frac{\mathbf{I}(T=0)}{\hat{g}_0}\right)\right] \text{(add and subtract term)} \\
&= \frac{2\beta}{N}\sum_i \left[\hat{Q}(t, x_i) - \hat{Q}(0, x_i) + (y_i - \hat{Q}(t, x_i))\left(\frac{\mathbf{I}(T=t)}{\hat{g}_t} - \frac{\mathbf{I}(T=0)}{\hat{g}_0}\right) - \hat{\psi}_t\right] \text{(group sums)}
\end{aligned}$$

On the other hand, by substituting the definition of the efficient influence curves (8) in the set of non-parametric estimation equations (9), multiplying by β and particularizing at $\hat{Q}, \hat{g}, \hat{\psi}$ (the functions modelled by the neural network at the optimal point of the parameter space), we obtain an expression equal to the one in the last line of the proof. Thus, the non-parametric estimation equations (9) are satisfied, and the proof is complete. □ □

A.2 Details about the DGP

The DGP consists mainly on two collections of functions, the function for the treatment and the functions for the conditional outcome.

The function for the treatment is

$$p(t|X) = \begin{cases} 0.8, & \text{if } t = m(X) \\ \frac{0.2}{k-1}, & \text{otherwise} \end{cases}$$

with $m : \mathcal{X} \to [0..k]$ a function mapping a given subset of the potential covariates to an integer (the treatment). We have implemented different shapes of m, from simple indicator functions based on one or two covariates, to a more complex function accepting a varying number of covariates in order to test the impact of this variation (as explained in A.2.3).

The conditional outcome functions are a collection of nonlinear functions of the shape $Y = L(T, X) = [l_0(T, X), l_1(T, X), \dots, l_k(T, X)]$ that map some of the potential covariates, the actual covariates (both included in X) and the treatment to the output space. After this mapping, we have added random noise to the output. These functions are linear and nonlinear combinations and compositions of functions such as the log function, the absolute value function, etc. We have implemented partially linear shapes such as $l_t(T, X) = f(X) + a * \mathbf{1}\{T = t\}$ and fully nonlinear shapes such as $l_t(T, X) = g(B \cdot X + cT^2)$.

We have generated datasets for the particular cases of 3, 5 and 8-valued treatments. In addition, we have tested the influence over the performance of other parameters such as the dataset size, the bias size and the number of covariates. Each dimension is explained below.

A.2.1 Dataset size

We have followed a data augmentation strategy and varied the size of the dataset, repeating each data point a given number of times and then adding noise to the whole dataset. To prevent an undesired, potentially subtle form of leakage between the train and the test sets, we have avoided shuffling the dataset until the splitting in sets was done. Leakage could appear if two augmented data points originated from the same initial data point end up in different splits of the dataset. We speak about a subtle form of leakage because the duplicated individuals are not exactly the same due to the addition of noise.

A.2.2 Bias

We have varied the bias size by means of the treatment effect size. The bias size is the magnitude of the difference between the observable, biased distribution, and the unobservable, unbiased distribution. An easy way to vary this magnitude is by implementing l_t functions that are partially linear, i.e., linear in the treatment: $l_t(T, X) = f(X) + a * \mathbf{1}\{T = t\}$, where f is some nonlinear function, a is the effect of the treatment and $\mathbf{1}\{T = t\}$ is an indicator function that takes value 1 if $T = t$ and 0 otherwise. This way, we can control the difference between the observable and the unobservable distributions of Y: the bigger a, the bigger the difference (the bias).

A.2.3 Number of covariates

We have tested the algorithms employing different numbers of covariates. Recall that a covariate is a variable that affects both the treatment and the outcome. In the IHDP dataset there are twenty seven variables that are potential covariates. In the classical binary treatment example from [3], only one of those twenty seven variables is actually a covariate. We have designed a data generating process that accepts a varying number of them, testing the cases of 4, 8, 12, 16 and 20 covariates.

A.3 Results in the test set

A.3.1 3-valued treatment setting

In the 3-valued treatment setting we have studied the performance of the algorithm under different dataset sizes. Table 3 shows the aforementioned performance metrics for each dataset size. Figure 3 shows the trends of the errors and relative errors. We can see the errors from both the Baseline and the T-reg algorithms decrease as the dataset size increases. This is to be expected: up to some limit, more data means a better training and a better fit of the neural net in the test set. There are no appreciable differences between baseline and T-reg algorithms.

A.3.2 5-valued treatment setting

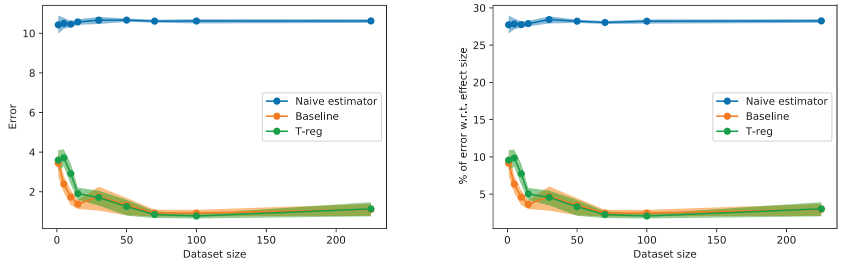
For the 5-valued treatment setting, we have studied the algorithm in the presence of different bias sizes. The results can be seen in Figure 4 and Table 4. As expected, the errors of the naive estimator increase as the bias increases. Both the Baseline and the T-reg algorithms perform well, with a maximum relative error around 10% in the case of the biggest bias. There is no significant difference between T-reg and Baseline algorithms for small bias sizes, although T-reg performs slightly better for bigger bias sizes. Note that the bias size variable is just a multiplying term in the DGP, and the number should not be taken at face value.

A.3.3 8-valued treatment setting

In the 8-valued treatment setting, we have tested the impact of different numbers of covariates. The results can be seen in Figure 5 and Table 5. The error of the naive estimator diminishes, then remains stable, then diminishes again when the number of covariates increases. We hypothesize that this is due to some sort of "randomization effect": due to the shape of the functions employed for constructing the outcomes in the DGPs, the more covariates we include, the more the observable and unobservable outcome distributions resemble to a normal distribution, and thus the smaller the bias. This would not necessarily be the case with a real-world (not synthetic) DGP. Both Baseline and T-reg algorithms perform well, with low relative errors, below 4% in most cases. The error of the T-reg algorithm is bigger than the Baseline algorithm for smaller number of confounders, but it becomes similar or smaller with increasing number of confounders.

Table 3: 3-valued treatment, test set

Data s.	True effect	Naive err.	% naive err.	Baseline err.	% baseline err.	T-reg err.	% t-reg err.
985	37.60	10.42	27.72	3.44	9.14	3.60	9.57
4925	37.71	10.50	27.83	2.38	6.32	3.71	9.85
9850	37.69	10.46	27.76	1.73	4.59	2.92	7.74
14775	37.90	10.57	27.89	1.38	3.63	1.91	5.03
29550	37.50	10.66	28.43	1.75	4.68	1.71	4.56
49250	37.80	10.66	28.21	1.33	3.52	1.26	3.32
68950	37.84	10.61	28.04	0.93	2.46	0.85	2.25
98500	37.65	10.62	28.20	0.93	2.46	0.78	2.06
221625	37.59	10.62	28.25	1.11	2.95	1.14	3.03



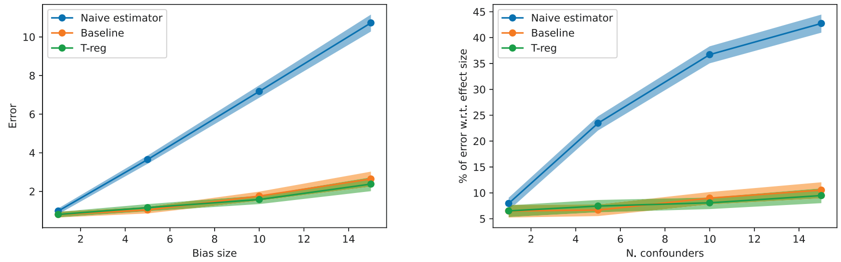
(a) Absolute error

(b) Error relative to effect size

Figure 3: 3-valued treatment, test set

Table 4: 5-valued treatment, test set

Bias s.	True effect	Naive err.	% naive err.	Baseline err.	% baseline err.	T-reg err.	% t-reg err.
1	12.35	0.98	7.96	0.80	6.50	0.81	6.52
5	15.56	3.65	23.47	1.04	6.67	1.16	7.45
10	19.56	7.18	36.71	1.76	8.99	1.58	8.08
15	25.10	10.73	42.74	2.64	10.53	2.38	9.48



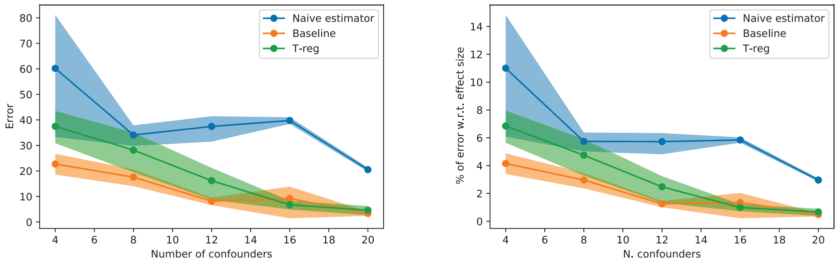
(a) Absolute error

(b) Error relative to effect size

Figure 4: 5-valued treatment, test set

Table 5: 8-valued treatment, test set

N. covars.	True effect	naive err.	% naive err.	Baseline err.	% baseline err.	T-reg err.	% t-reg err.
4	547.16	60.19	11.00	22.72	4.15	37.47	6.85
8	593.74	34.09	5.74	17.59	2.96	28.18	4.75
12	654.15	37.44	5.72	8.19	1.25	16.20	2.48
16	680.73	39.77	5.84	9.20	1.35	6.75	0.99
20	692.10	20.51	2.96	3.35	0.48	4.63	0.67



(a) Absolute error

(b) Error relative to effect size

Figure 5: 8-valued treatment case, test set