Universal Causal Evaluation Engine: An API for empirically evaluating causal inference models

Alexander Lin* alexanderlin01@g.harvard.edu
Harvard University

Amil Merchant* amilmerchant@g.harvard.edu
Harvard University

Suproteem K. Sarkar* suproteemsarkar@g.harvard.edu
Harvard University

Alexander D'Amour alexdamour@google.com
Google Research

Editor: Thuc Duy Le, Jiuyong Li, Kun Zhang, Emre Kıcıman, Peng Cui, and Aapo Hyvärinen

Abstract

A major driver in the success of predictive machine learning has been the "common task framework," where community-wide benchmarks are shared for evaluating new algorithms. This pattern, however, is difficult to implement for causal learning tasks because the ground truth in these tasks is in general unobservable. Instead, causal inference methods are often evaluated on synthetic or semi-synthetic datasets that incorporate idiosyncratic assumptions about the underlying data-generating process. These evaluations are often proposed in conjunction with new causal inference methods—as a result, many methods are evaluated on incomparable benchmarks. To address this issue, we establish an API for generalized causal inference model assessment, with the goal of developing a platform that lets researchers deploy and evaluate new model classes in instances where treatments are explicitly known. The API uses a common interface for each of its components, and it allows for new methods and datasets to be evaluated and saved for future benchmarking.

Keywords: Causal Inference, Software Engineering, Machine Learning

1. Introduction

In this paper, we present an API and preliminary implementation for comparatively evaluating causal inference models. Causal inference is used to help identify the effects of particular treatments on populations, and the application of machine learning methods to causal inference is a growing body of research (Hill, 2011; Hahn et al., 2017; Louizos et al., 2017; Nie and Wager, 2017; Shalit et al., 2017; Oprescu et al., 2018; Wager and Athey, 2018; and numerous others). Centralized methods for evaluating these models across dataset types are currently limited to pipelines not explicitly designed for extensibility, with the most common example being causal inference competitions (e.g. Dorie et al., 2019). Therefore, many papers focus on models which improve individual metrics on select datasets, instead of focusing on model generalizability, or comparing performance across a range of tasks.

∗. Equal Contribution

The predictive machine learning community has established a common task framework for model evaluation in many problem domains (Donoho, 2017), which has allowed researchers to develop models across a common set of benchmarks. Our work seeks to build a centralized resource for evaluating and contributing causal inference models, datasets, and evaluation metrics. We design an API that allows new contributions to be introduced and evaluated, allowing researchers to spend less time developing infrastructure as they introduce new models. In contrast to existing unified evaluation frameworks like Kaggle competitions, which often evaluate submitted models across a single task, our API allows researchers to run common evaluation metrics across numerous models, datasets, and metrics. Furthermore, since the "ground truth" in causal inference datasets is often parameterized, our model also allows for evaluations that sweep across parameters rather than being evaluated on one fixed dataset instantiation. We add extensibility by allowing researchers to submit new models, datasets, metrics, and parameterizations. Our goal is to scale this platform to a persistent service and encourage the causal inference community to use a centralized resource for evaluating and sharing information about its research.[1]

1.1 Preliminaries

We base our initial system on the potential outcomes framework (Rubin, 2005) for determining the effects of treatments of individuals. Suppose there are n examples (X_i, Y_i, T_i), for $i = 1, \ldots, n$, where $X_i \in \mathcal{X}$ denotes the covariate values for individual i, $Y_i \in \mathbb{R}$ is the observed outcome, and $T_i \in \{0, 1\}$ is the assigned treatment to individual i. Let $Y_i(0)$ and $Y_i(1)$ correspond to the outcome that would have been observed if $T_i = 0$ or $T_i = 1$, respectively. Let $\mu_{(0)}(x) = \mathbb{E}[Y(0)|X = x]$ and $\mu_{(1)}(x) = \mathbb{E}[Y(1)|X = x]$. The Conditional Average Treatment Effect (CATE) $\tau(x)$ is defined as

$$\tau(x) = \mathbb{E}[Y(1) - Y(0)|X = x] = \mu_{(1)}(x) - \mu_{(0)}(x). \tag{1}$$

Although the CATE is often the target of estimation in causal inference problems, as it approximates subgroup-specific effects, it is generally unobservable because only $Y_i(0)$ or $Y_i(1)$ is present for each i in real-world datasets. This "fundamental problem of causal inference" (Holland, 1986) makes model evaluation especially difficult, which is why researchers often develop their own synthetic or semi-synthetic datasets.

1.2 Paper Organization

In Section 2, we outline the structure of our API, and describe how new models and datasets can be added and evaluated. Section 3 presents sample evaluations from our API and evaluates models across datasets. Finally, Section 4 concludes the paper with our plans to scale the API.

2. API

The major contribution of this work is to create a centralized API that is able to exhaustively test new causal models, datasets, and metrics. In this section, we explain the different

1. A (pre-alpha) implementation of the API can be found at `https://github.com/amerch/causal_inference_evaluation`.

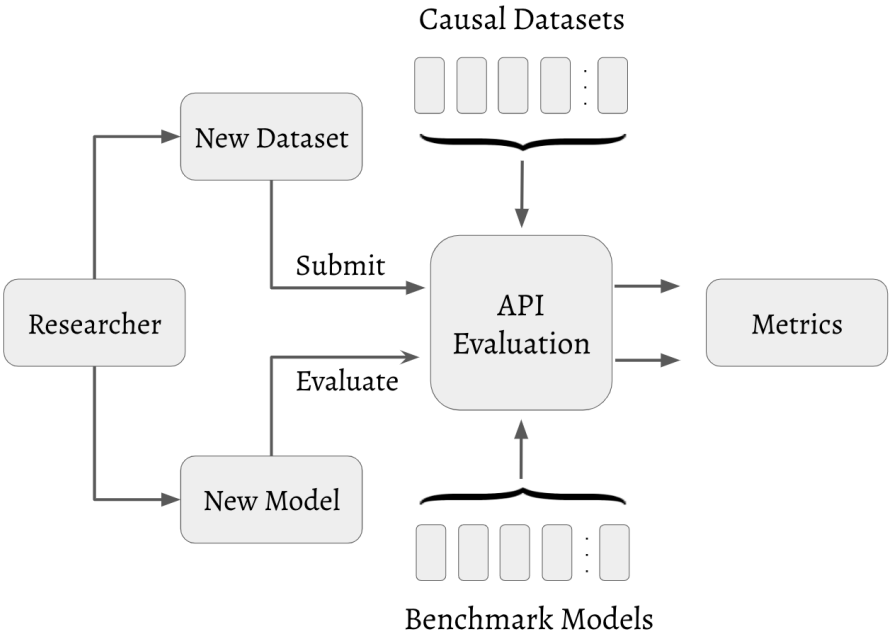

Figure 1: Schematic Diagram of the API

components of the API and how the system is built to flexibly scale to new experiments. Figure 1 schematically diagrams the API—a researcher using this software can easily submit a new dataset and/or a new model to the program. The API will then evaluate and produce a summary of relevant metrics. Our API standardizes data and evaluation metrics, leading to easily reproducible results.

2.1 Models

Our framework adopts an scikit-learn (Pedregosa et al., 2011) style approach to developing new models. We define an abstract superclass so that any newly proposed model can be defined by implementing the `.fit(X, Y, T)` and `.predict(X, T)` functions. This simple structure allows for rapid model prototyping and testing on a range of the benchmark datasets. The provided framework also allows the model to be tuned for the specific prediction tasks. For example, a binary tag is provided to allow for models to switch between classification and regression. Also, k-fold and leave-one-out cross validation are both implemented to allow for model hyperparameters to be tuned on any given problem.

A number of these models are already included in the software implementation of this API. These benchmarks include fairly simple implementations including k-nearest neighbors, a naive average of treatment and control groups, Ordinary Least Squares with treatment as a feature, Ordinary Least Squares separated by treatment, and random forests. More sophisticated methods include the S-learner, R-learner, and T-learner, all discussed in Nie and Wager (2017). The diversity of these methods has allowed for initial evaluation and analysis on some standard causal inference tasks, as discussed in Section 3.1. As new

models are completed and published, they can be added to the set of benchmark models for future API evaluation and comparisons.

2.2 Data

All datasets within the API reflect the standard form of causal inference problems. Each task is defined by at least a X, T, Y, which are required in training and to create predictions. Additional variables such as the true values of $\mu_{(1)}(X)$ and $\mu_{(0)}(X)$ can be included if known in order to properly evaluate the appropriate predictions. These variables are stored in zip archive files available in the Data folder of the software. Specifically, we use the `.npz` file format available through the numpy library to create a common method for storing and loading datasets. Additionally, it is worth mentioning that each tensor for $X, T, Y, \ldots$ contains a final axis that corresponds to a new realization of the complete dataset. This will allow us to produce confidence intervals on the calculated metrics, further described in Section 2.3.

The API is initialized with a number of benchmark datasets that can be used for evaluation. These represent a sample of the common datasets used in causal inference papers, including synthetic and semi-synthetic data, and reflect varying levels of confounding. As this API is further developed, new datasets can be added to this benchmarking procedure for a more comprehensive selection of causal problems. In the mean time, the currently available datasets are detailed below:

1. **IHDP**
   The IHDP dataset contains records from the Infant Health and Development Program. From 1985, the randomized experiment evaluated the effect of high-quality child care on the cognitive test scores at the end of the intervention. Hill (2011) first applied this data to causal inference by systematically removing children with non-white mothers from the treated set. The data was obtained from Shalit et al. (2017) and contains 100 replications.

2. **Jobs**
   The Jobs dataset originally comes from LaLonde (1986) and is a commonly used benchmark within the causal inference community. The features include 8 covariates, such as race, age, education, and previous earnings. The treatment involved participation in a job training program, and the predictor is a binary outcome of unemployment. We follow Dehejia and Wahba (2002) in the creation of this dataset and specific samples used.

3. **Voting**
   The Voting dataset was first proposed by Arceneaux et al. (2006) to study the effect of get-out-the-vote calls on turnout in elections. We follow the use of the dataset by Nie and Wager (2017). These authors injected a synthetic treatment effect into the data—allowing for measurements of how effective the proposed causal inference methods were. The features include state, county, age, gender. The treatment is the binary get-out-the-vote call, and the outcome is participation in the election.

4. **Twins**
   The Twins dataset comes from a study by Almond et al. (2005) on the effects of low birth weight on twins between 1989 and 1991. Louizos et al. (2017) created a semi-synthetic dataset based on this twins data, using gestational time as a confounder for the mortality outcome. Features included maternal risk factors, race, and quality of care.

Of the included datasets, Jobs, Voting, and Twins are all thought to have confounding variables that are not directly observable by the models. This is important for model evaluation, as many techniques for causal inference make unconfoundedness assumptions that may not always hold.

2.3 Evaluation

As noted in Section 2.2, the datasets contain multiple realizations of each causal inference problem. In the outer loop of our evaluation metric, for each realization, we initialize a new version of the model. This model is trained with hyper-parameter optimization using cross validation. The model is then evaluated on the specific test set realization, and the results are stored in internal databases. By using this standard procedure for all models, the resulting metrics are comparable and can be used to analyze the efficacy of the various methods. As an example, in the next section, we present metrics using the benchmark models and datasets described above.

3. Model Comparison Examples

A key contribution of the API is the provision of a common process for evaluating models. In this section, we give examples of sample comparative results that can be gleaned from our interface. The use of a single module for standardizing traditionally variable factors—such as training set size, cross-validation splits, and performance metrics—allows for fairer comparisons than the common approach of comparing numbers found across papers.

3.1 Comparative Evaluation

First, we explore the use of common metrics for evaluating the performance of several models on a single dataset. In this case, the dataset is IHDP, which is described in the previous section. The evaluation process is conducted using the method outlined in Section 2.3. Our metrics of choice are the expected precision in estimation of heterogeneous effect (PEHE) and the root mean square error of the individual treatment effect (RMSE ITE). Figure 2 displays a sampling of the results across several models. From this graph, we can glean conclusions about how the different models compare in terms of accuracy and precision.

3.2 Cross-Model, Cross-Dataset Evaluation

Table 1 presents sample results from a cross-model, cross-dataset evaluation of the out-of-sample average treatment effect (ATE) error. This evaluation is across eight different models and four different datasets. We envision the API being used to generate types of results that are similar to this one.

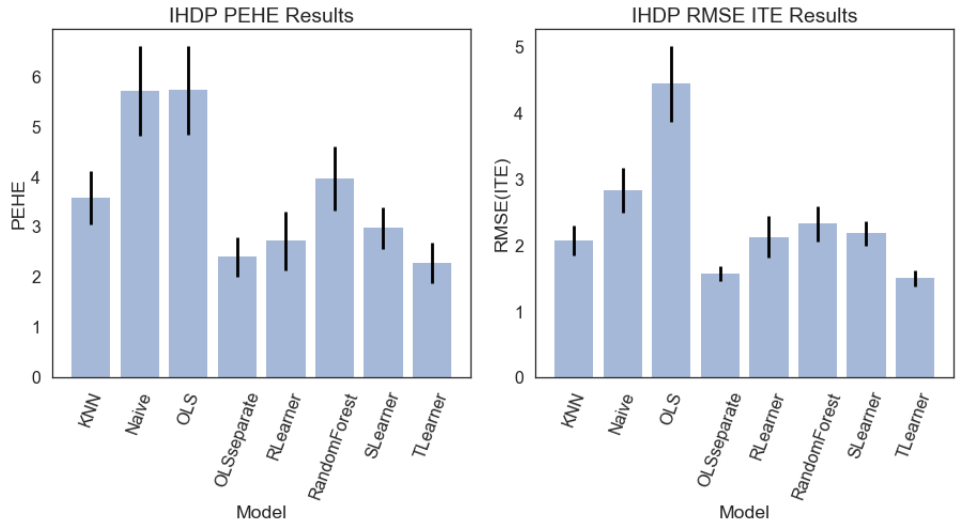

Figure 2: Examples of comparative evaluations between models for the metrics PEHE (left) and RMSE ITE (right) on the IHDP dataset. For these metrics, lower values correspond to higher accuracy. Our results correspond to the available metrics that appeared in previous papers using these models.

	IHDP	Twins	Jobs	Voting
Naive	1.09 ± 0.20	0.0538 ± 0.0018	0.1534	0.0314 ± 0.0011
OLS	0.78 ± 0.11	0.0385 ± 0.0014	0.0414	0.0045 ± 0.0012
OLS-Separate	0.27 ± 0.04	0.0246 ± 0.0015	0.0819	0.0020 ± 0.0009
S-Learner	0.48 ± 0.06	0.0400 ± 0.0021	0.0427	0.0065 ± 0.0010
T-Learner	0.26 ± 0.03	0.0312 ± 0.0015	0.0706	0.0020 ± 0.0009
KNN	0.66 ± 0.12	0.0440 ± 0.0015	0.1128	0.0038 ± 0.0011
Random Forest	0.80 ± 0.12	0.0317 ± 0.0025	0.0882	0.0131 ± 0.0011

Table 1: Example comparison of out-of-sample average treatment effect absolute error for various models across various datasets.

3.3 Extensibility of Evaluations

The API also allows for the standardized incorporation of novel evaluations. If a particular paper introduces a new, creative method of benchmarking models on datasets, this method can be easily incorporated into the API, which allows it to be quickly accessed by all models and all (relevant) datasets. For example, Louizos et al. (2017) introduce a way to evaluate how the absolute ATE error changes as a function of proxy noise level for the TWINS dataset. They evaluate their particular models in this context. Figure 3 shows how the API can be utilized to provide similar results for a variety of models in the literature, beyond the ones initially tested by Louizos et al. (2017).

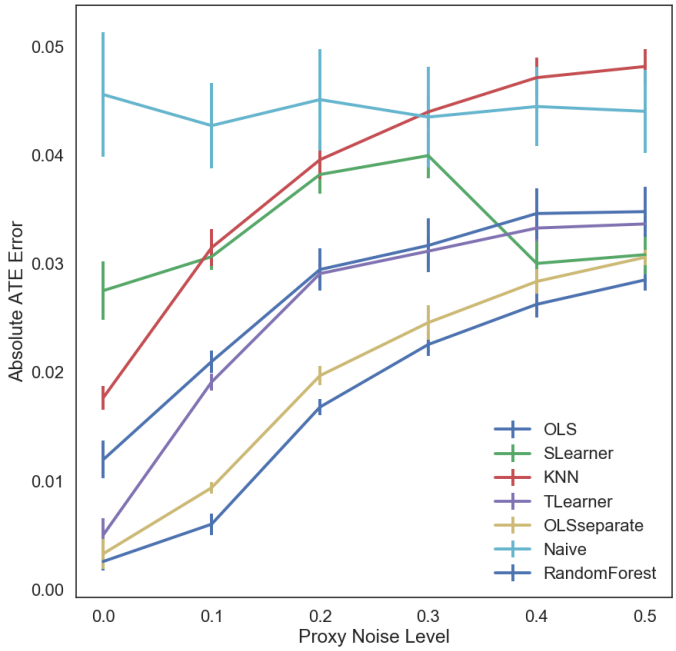

Figure 3: Example of an extension evaluation on the API, depicting how absolute ATE error changes as a function of proxy noise level for selected models fitted to the Twins dataset.

4. Discussion and Scale Up

In this paper, we introduce an API for the centralized evaluation and dissemination of causal inference models, datasets, and metrics. Currently, our API is a local service with a standard interface for adding data and models, with a model interface similar to that of scikit-learn. Researchers may submit to it via pull request. Our goal is to move the evaluation suite to a hosted platform, allowing the research community to upload new models using the API, and share results of training across different datasets. We also seek to gather feedback on our current design and evaluation pipelines, so the service can better fit the needs of the community. The goal of our service is to make comprehensive evaluation more readily available, so researchers can spend more time model-building and less time plumbing. We seek to develop a centralized platform for publicizing and viewing contributions from the community, and welcome feedback on our service as it is in development.

Acknowledgments

We thank researchers at Google Brain and participants in Harvard's topics in machine learning seminar for their support and comments.

References

Douglas Almond, Kenneth Y Chay, and David S Lee. The costs of low birth weight. *The Quarterly Journal of Economics*, 120(3):1031–1083, 2005.

Kevin Arceneaux, Alan S Gerber, and Donald P Green. Comparing experimental and matching methods using a large-scale voter mobilization experiment. *Political Analysis*, 14(1):37–62, 2006.

Rajeev H Dehejia and Sadek Wahba. Propensity score-matching methods for nonexperimental causal studies. *Review of Economics and Statistics*, 84(1):151–161, 2002.

David Donoho. 50 years of data science. *Journal of Computational and Graphical Statistics*, 26(4):745–766, 2017.

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

P Richard Hahn, Jared Murray, and Carlos M Carvalho. Bayesian regression tree models for causal inference: regularization, confounding, and heterogeneous effects. *Confounding, and Heterogeneous Effects (October 5, 2017)*, 2017.

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

Paul W Holland. Statistics and causal inference. *Journal of the American statistical Association*, 81(396):945–960, 1986.

Robert J LaLonde. Evaluating the econometric evaluations of training programs with experimental data. *The American economic review*, pages 604–620, 1986.

Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. Causal effect inference with deep latent-variable models. In *Advances in Neural Information Processing Systems*, pages 6446–6456, 2017.

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

Miruna Oprescu, Vasilis Syrgkanis, and Zhiwei Steven Wu. Orthogonal random forest for heterogeneous treatment effect estimation. *arXiv preprint arXiv:1806.03467*, 2018.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

Donald B Rubin. Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469):322–331, 2005.

Uri Shalit, Fredrik D Johansson, and David Sontag. Estimating individual treatment effect: generalization bounds and algorithms. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3076–3085. JMLR. org, 2017.

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