Do-PFN: In-Context Learning for Causal Effect Estimation

Jake Robertson [*1,2] **Arik Reuter** [*3] **Siyuan Guo** [3,4] **Noah Hollmann** [5] **Frank Hutter** [2,1,5] **Bernhard Schölkopf** [3,1]

Abstract

Causal effect estimation is critical to a range of scientific disciplines. Existing methods for this task either require interventional data, knowledge about the ground-truth causal graph, or rely on assumptions such as unconfoundedness, restricting their applicability in real-world settings. In the domain of tabular machine learning, Prior-data fitted networks (PFNs) have achieved state-of-the-art predictive performance, having been pre-trained on synthetic data to solve tabular prediction problems via in-context learning. To assess whether this can be transferred to the harder problem of causal effect estimation, we pre-train PFNs on synthetic data drawn from a wide variety of causal structures, including interventions, to predict interventional outcomes given observational data. Through extensive experiments on synthetic case studies, we show that our approach allows for the accurate estimation of causal effects without knowledge of the underlying causal graph. We also perform ablation studies that elucidate Do-PFN's scalability and robustness across datasets with a variety of causal characteristics.

1. Introduction

The estimation of causal effects is fundamental to scientific disciplines such as medicine, economics, and the social sciences (Pearl, 2009; Varian, 2016; Imbens, 2024; Wu et al., 2024). Questions such as "Does a new drug reduce the risk of cancer?" and "What is the impact of minimum wage on employment?" can only be answered by taking the causal nature of the problem into account.

The widely accepted gold standard for assessing causal effects are randomized controlled trials (RCTs). While RCTs allow for the direct estimation of causal effects, they can sometimes be unethical or expensive, and, in many cases, simply impossible. In contrast to experimental data from RCTs, *observational* data is more accessible, collected without interfering in the independent and identically distributed (i.i.d) data-generating process. Estimating causal effects from observational data alone can be challenging or even impossible without strict assumptions (Spirtes et al., 1993).

Various methods have been proposed to address the problem of causal effect estimation, typically relying on the assumption of unconfoundedness (Rosenbaum & Rubin, 1983). This assumption states that, conditional on a set of observed covariates, treatment assignment is independent of the potential outcomes. While this condition enables identification of causal effects from observational data, it can be difficult to verify or justify in practice, as it requires that relevant confounders are observed and properly accounted for (Hernán & Robins, 2010; Imbens & Rubin, 2015).

Many applications of causality involve tabular data. Prior-data fitted networks (PFNs; Müller et al., 2022) have recently transformed the landscape of tabular machine learning. In spite of being pre-trained only on synthetic data, TabPFN has produced impressive results on real-world machine learning benchmarks (McElfresh et al., 2023; Xu et al., 2025; Hollmann et al., 2025). Given these remarkable findings, it is timely to assess whether a similar meta-learning approach could help us tackle harder problems that are causal rather than merely predictive. As a first step, our goal is to extend PFNs to the problem of estimating conditional interventional distributions (**CIDs**).

Our contributions

1. We propose Do-PFN, a pre-trained foundation model that can predict interventional outcomes from observational data, and prove that it provides an optimal approximation of the conditional intervention distribution (CID) with respect to the chosen prior over data-generating models.
2. We evaluate the performance of Do-PFN on six case studies across more than 1,000 synthetic datasets. For both predicting CID and conditional average treatment effects (**CATEs**), Do-PFN (1) achieves competitive performance with baselines that have access to the

[*]Equal contribution [1]ELLIS Institute Tübingen, Tübingen, Germany [2]University of Freiburg, Freiburg, Germany [3]Max Planck Institute for Intelligent Systems, Tübingen, Germany [4]University of Cambridge, Cambridge, United Kingdom [5]Prior Labs, Freiburg, Germany. Correspondence to: Jake Robertson <robertsj@cs.uni-freiburg.de>, Arik Reuter <arik.reuter@tuebingen.mpg.de>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

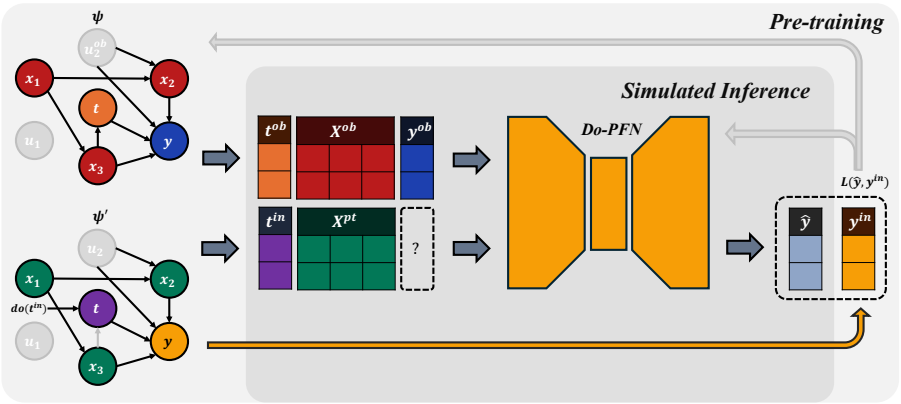

Figure 1. **Do-PFN overview**: Do-PFN performs in-context learning (ICL) for causal effect estimation, predicting conditional interventional distributions (CIDs) based on observational data alone. In pre-training, a large number of structural causal models (SCMs) is sampled. For each SCM, we sample an entire dataset of M^{ob} *observational* data points $\mathcal{D}^{ob} = \{(t_j^{ob}, \mathbf{x}_j^{ob}, y_j^{ob})\}_{j=1}^{M^{ob}}$. We also sample M^{in} *interventional* data points $\mathcal{D}^{in} = \{(t_k^{in}, \mathbf{x}_k^{in}, y_k^{in})\}_{k=1}^{M^{in}}$. To simulate inference, we input (t^{in}, x^{in}) along with the entire observational dataset $\mathcal{D}_{ob}$, which can have various sizes and dimensionalities. Subsequently, the transformer makes predictions $\hat{y}$, and we calculate the pre-training loss $L(\hat{y}, y^{in})$ between the predictions $\hat{y}$ and the ground truth interventional outcomes y^{in}. Pre-training repeats this procedure across millions of sampled SCMs to *meta-learn* how to perform causal inference *in context*. On new, unseen datasets, Do-PFN leverages the many simulated interventions it has seen during pre-training to predict CIDs, relying only on observational data and requiring no information about the causal graph or model.

true causal graph (typically not available in practice) and (2) statistically significantly outperforms all other baselines.

2. Background and related work

Structural causal models Structural causal models (SCMs; Pearl, 2009; Peters et al., 2017) specify the mechanisms to generate the variables from their (causal) parents in a a directed acyclic graph (DAG) $\mathcal{G}_\psi$ via structural equations $z_k = f_k(z_{\text{PA}(k)}, \epsilon_k)$, where f_k is a function, $z_{\text{PA}(k)}$ denotes the parents of variable k in $\mathcal{G}$ and ϵ_k is a random noise variable.

Interventions and causal effects In the context of SCMs, performing an intervention $do(t)$ for a variable $T \in \{z_1, z_2, \ldots, z_K\}$ that is part of the SCM ψ corresponds to removing all incoming edges into the node representing t and fixing the value of the variable T to the value t. We assume the "treatment" T to be binary such that $t \in \{0, 1\}$. The causal effect of this intervention on an outcome y is captured by $p(y|do(t), \psi)$.

A central object of interest for this paper is the CID (Shpitser & Pearl, 2006) that additionally conditions on a vector $\mathbf{x}$ comprising several variables in the SCM,

$$p(y|do(t), \mathbf{x}). \tag{1}$$

A CID answers a question like *"What is the distribution of outcomes given that (i) a patient has features* $\mathbf{x}$ *and (ii) an intervention* $do(t)$ *is performed?"* CIDs enable the estimation of CATEs: $\tau(x) := \mathbb{E}[y|do(1), \mathbf{x}] - \mathbb{E}[y|do(0), \mathbf{x}]$.

Estimating causal effects Various methods allow for the direct estimation of causal effects from experimental data (Shalit et al., 2017; Kennedy, 2023; Nie & Wager, 2021). However, RCT data is often difficult to access. It might be easier, or even the only option, to access an *observational* dataset $\mathcal{D}_{ob} = \{(y_j^{ob}, t_j^{ob}, x_j^{ob})\}_{j=1}^{M_{ob}}$ of passively collected samples $(y_j^{ob}, t_j^{ob}, x_j^{ob}) \sim p(y, t, \mathbf{x})$.

When approaching causal effect estimation from the framework of *do-calculus* (Pearl, 2009), practitioners first need to construct an SCM ψ that they believe (or have inferred) to represent the ground-truth data-generating process. The rules of do-calculus subsequently allow to determine whether and how the desired causal effect can be estimated from the data. The Neyman-Rubin framework (Imbens & Rubin, 2015) defines causal effects as contrasts between potential outcomes $y_1 \sim p(y|do(1))$ and $y_0 \sim p(y|do(0))$, and relies on a set of key assumptions, critically ignorability (or unconfoundedness). Machine-learning based methods that are conceptualized in this framework include causal trees (Athey & Imbens, 2016), causal forests (Wager & Athey, 2018), as well as T-, S- and X-learners (Künzel et al., 2019).

PFNs and amortized Bayesian inference In our context, we define amortized (Bayesian) inference as learning the mapping $\mathcal{D} \mapsto p(y|\mathbf{x}, \mathcal{D})$ from a dataset to a posterior; that is, the amortization occurs at the dataset level. Neural pro-

cesses (Garnelo et al., 2018a;b; Nguyen & Grover, 2022) and various techniques from the field of simulation-based inference (Wildberger et al., 2023; Gloeckler et al., 2024; Vasist et al., 2023) perform amortized inference in the aforementioned manner. Recently, PFNs have been proposed as an amortized inference framework, emphasizing the role of large-scale pre-training and realistic simulators of synthetic data, referred to as the *prior* (Müller et al., 2022).

Amortized causal inference Regarding amortized causal inference, Sauter et al. (2025) consider the problem of meta-learning causal inference, proposing to learn the shift in distributions of all nodes in the SCM when performing an intervention. However, this approach fails to outperform a conditioning-based baseline even in a two-variable setting. Bynum et al. (2025) propose to use amortized inference to learn various causal effects, where they focus on low-dimensional SCMs with up to three nodes and do not target the CID, but only point estimates, thus ignoring uncertainty.

3. Methodology: causal inference with PFNs

Modeling assumptions We now formalize how to conduct causal inference with PFNs, more precisely how to estimate conditional interventional distributions (CIDs) defined as $p(y|do(t), \mathbf{x})$ from observational data $\mathcal{D}^{ob}$. A central component of our approach to causal effect estimation is to posit a prior $p(\psi)$ over SCMs. We further require that every sampled SCM $\psi \sim p(\psi)$ allows to simulate observational data from $p(y^{ob}, t^{ob}, \mathbf{x}^{ob}|\psi)$ by sampling noise $\epsilon \sim p(\epsilon)$ that is propagated through the SCMs ψ. Additionally, a prior $p(t^{in})$ over possible values for the treatment variable is required to sample values for the interventions. Samples from the distribution $p(y^{in}, \mathbf{x}^{in}|\psi, do(t^{in}))$ over outcomes and covariates given this intervention then result from forward-propagating through the intervened-upon SCM. Please refer to Algorithm 1 and Appendix B for more details on the data-generating process we use to train the PFN.

The assumptions above imply the following form of the CID:

$$p(y^{in}|do(t^{in}), \mathbf{x}^{in}) = \int p(y^{in}|do(t^{in}), \mathbf{x}^{in}, \psi)\; p(\psi|\mathbf{x}^{in}) d\psi. \quad (2)$$

Assuming a prior $p(\psi)$ over SCMs, and thus also over causal graphs $\mathcal{G}_\psi$, can be seen as an extension of the classical do-calculus approach where typically a fixed causal graph $\widetilde{\mathcal{G}_\psi}$, or even a fixed SCM $\widetilde{\psi}$, is used as the basis for further inference. Compared to the assumptions typically made in the potential outcomes framework, our method also includes scenarios without the unconfoundedness assumption.

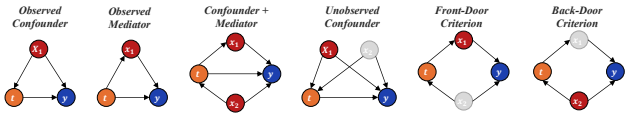

Figure 2. **Case studies**: Visualization of the graph structures of our six causal case studies, requiring Do-PFN to automatically perform adjustment based on the front-door and back-door criteria. Treatment variables t are visualized in orange, covariates $\mathbf{x}$ in red, and outcomes y in blue. Gray variables represent unobservables, not shown to any of the methods yet influencing the generated data.

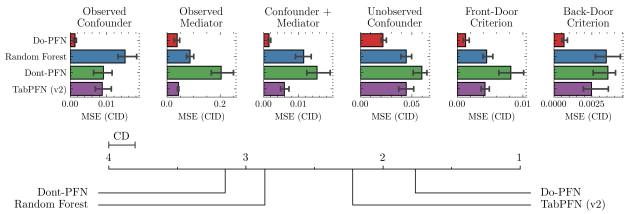

Figure 3. **CID prediction**: Bar-charts and critical difference (CD) diagrams depicting distributions of normalized mean squared error (MSE) of Do-PFN and our regression baselines in conditional interventional distribution (CID) prediction.

Approximating the CID Ultimately, we are interested in obtaining a model $q_\theta(y^{in}|do(t^{in}), \mathbf{x}^{in}, \mathcal{D}^{ob})$ The core idea of Do-PFN is to achieve this by *prior fitting*, i.e., minimizing the negative log-likelihood $-\log q_\theta(y^{in}|do(t^{in}), \mathbf{x}^{in}, \mathcal{D}^{ob})$ on data from the synthetic data-generating process (Müller et al., 2022) via stochastic gradient descent (lines 19 and 20 in Algorithm 1). Theorem A.1, detailed in Appendix A shows our prior fitting procedure (Algorithm 1) achieves the goal of yielding an optimal approximation of the CID from observational data.

Architecture and training details Do-PFN is a transformer with a similar architecture as TabPFN (Hollmann et al., 2025). In order to specialize this architecture for predicting CIDs, we simply add a special indicator to the internal representation of each input dataset to specify that the first column is the treatment and the rest are covariates. Do-PFN has 7.3 million parameters and is trained with Algorithm 1, with details in Appendix B. This takes 48 hours on a single RTX-2080.

4. Experiments

We evaluate Do-PFN's performance in CID prediction and CATE estimation against a competitive set of causal and tabular machine learning baselines (see Appendix C.3). We provide our pre-trained models, pre-training data generating code, and case study datasets at `https://github.com/jr2021/Do-PFN`.

4.1. CID prediction

First, we evaluate our longest trained model, Do-PFN, against a set of baselines for the task of predicting the CID $p(y|do(t), \mathbf{x})$. In Figure 3, we visualize bar-charts[1] of normalized mean squared error (MSE) across our six case studies. For a description of MSE, please see Appendix C.2. We also provide a critical difference (CD) diagram below, indicating average ranking across all case studies. A lower CD-value is better; and bold lines connect pairs of models whose performance does not differ by a significant amount (not applicable in Figure 3).

Effectiveness of pre-training objective In Figure 3, we first observe that Do-PFN performs statistically significantly better[2] than the following tabular regression models: Random Forest, TabPFN (v2), as well as a regression model pre-trained on our prior to predict observational outcomes (dubbed "Dont-PFN"). This result shows that our pre-training objective results in a model that precisely approximates the CID as opposed to the standard posterior predictive distribution.

Gold-standard comparison (CID) When comparing Do-PFN to our "gold standard" baselines (Appendix Figure 8), we observe that Do-PFN performs competitively with equally expressive models which (explicitly or implicitly) know the graph structure.

4.2. CATE estimation

Comparison to causal machine learning In estimating CATE values, we again observe that our largest model Do-PFN-CATE statistically significantly outperforms both the Do-PFN-CATE S-Learner (Künzel et al., 2019) and a causal forest double machine learning (DML) approach (Wager & Athey, 2018; Chernozhukov et al., 2018), even on our relatively simple case studies (Figure 4).

Gold-standard comparison (CATE) We also observe that in the CATE estimation setting, Do-PFN-CATE performs closer to the gold standard DoWhy-CATE (Cntf.) than previously in the more challenging CID prediction (Appendix Figure 9). We highlight that Do-PFN-CATE is especially competitive on the "Front-Door" and "Back-Door" case studies, where none of the models are given access to the unobserved variable; hence DoWhy loses the fundamental advantage that it had in other settings.

[1]Our bar-charts visualize median values and 95% confidence intervals

[2]Significance is assessed using a post-hoc Nemenyi test implemented in the Autorank package (Herbold, 2020).

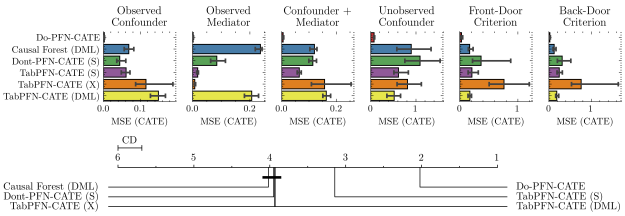

Figure 4. **CATE estimation**: Box-plots and critical difference (CD) diagrams depicting distributions of normalized mean squared error (MSE) of Do-PFN-CATE and our causal baselines in conditional average treatment effect (CATE) estimation.

4.3. Ablations

The key takeaways from our ablations are that Do-PFN performs strongly on small datasets, shows invariance to graph complexity and base treatment effects, and correctly accounts for uncertainty arising from unidentifiability. (Appendix D).

4.4. Hybrid synthetic-real-world data

We further find that Do-PFN can effectively estimate CIDs and CATEs on the Amazon product sales dataset (Blöbaum et al., 2023) and the law school admissions problem (Kusner et al., 2017), where, in the absence of ground-truth interventional outcomes we calculate MSE with respect to DoWhy (Cntf.), our "gold-standard" baseline (Appendix E).

5. Conclusion

We introduced Do-PFN, a pre-trained transformer leveraging ICL to learn to predict interventional outcomes from observational data. Our empirical results on controlled synthetic setups suggest that Do-PFN outperforms a strong set of tabular and causal machine learning baselines, while performing competitively with equally expressive models which are being given the true underlying causal graph.

Do-PFN's generalization capability critically depends on its synthetic prior, adequately capturing real-world complexity. As our current validation is mainly based on synthetic data, Do-PFN's robustness to prior-reality mismatches and its performance on real-world datasets require further systematic exploration. Furthermore, while Do-PFN can reflect uncertainty arising from unidentifiability, a complete statistical understanding remains an open research area.

In conclusion, we are optimistic about Do-PFN's prospects to become part of the standard machine learning toolkit, thus helping to give causal effect estimation the broad accessibility that its real-world relevance deserves.

Impact Statement

The goal of our work is to advance the field of causal inference, a field which we believe has broad positive applications, especially in the field of medicine, the social sciences, and the natural sciences. While we believe that the underlying goal of understanding causal mechanisms from data is exploratory by nature, we do acknowledge that causal inference could be applied by bad actors. The causal machine learning community should become increasingly aware of possible misuse and risks as their methods become more and more applicable in real-world scenarios.

References

Athey, S. and Imbens, G. Recursive partitioning for heterogeneous causal effects. *Proceedings of the National Academy of Sciences*, 113(27):7353–7360, 2016.

Barber, D. and Agakov, F. The IM algorithm: a variational approach to information maximization. *Advances in neural information processing systems*, 16(320):201, 2004.

Blöbaum, P., Budhathoki, K., and Götz, P. Root cause analysis with dowhy, an open source python library for causal machine learning, jan 2023. Categories: Amazon Machine Learning, Open Source, Technical How-to.

Blöbaum, P., Götz, P., Budhathoki, K., Mastakouri, A. A., and Janzing, D. Dowhy-gcm: An extension of dowhy for causal inference in graphical causal models. *Journal of Machine Learning Research*, 25(147):1–7, 2024.

Breiman, L. Random forests. *Machine Learning*, 45:5–32, 10 2001. doi: 10.1023/A:1010950718922.

Bynum, L. E., Puli, A. M., Herrero-Quevedo, D., Nguyen, N., Fernandez-Granda, C., Cho, K., and Ranganath, R. Black box causal inference: Effect estimation via meta prediction. *arXiv preprint arXiv:2503.05985*, 2025.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., and Robins, J. Double/debiased machine learning for treatment and structural parameters, 2018.

Garnelo, M., Rosenbaum, D., Maddison, C., Ramalho, T., Saxton, D., Shanahan, M., Teh, Y. W., Rezende, D., and Eslami, S. A. Conditional neural processes. In *International conference on machine learning*, pp. 1704–1713. PMLR, 2018a.

Garnelo, M., Schwarz, J., Rosenbaum, D., Viola, F., Rezende, D. J., Eslami, S., and Teh, Y. W. Neural processes. *arXiv preprint arXiv:1807.01622*, 2018b.

Gloeckler, M., Deistler, M., Weilbach, C., Wood, F., and Macke, J. H. All-in-one simulation-based inference. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 15735–15766, 2024.

Herbold, S. Autorank: A python package for automated ranking of classifiers. *Journal of Open Source Software*, 5(48):2173, 2020. doi: 10.21105/joss.02173. URL https://doi.org/10.21105/joss.02173.

Hernán, M. A. and Robins, J. M. Causal inference, 2010.

Hollmann, N., Müller, S., Eggensperger, K., and Hutter, F. TabPFN: A transformer that solves small tabular classification problems in a second. In *The Eleventh International Conference on Learning Representations*, 2023.

Hollmann, N., Müller, S., Purucker, L., Krishnakumar, A., Körfer, M., Hoo, S. B., Schirrmeister, R. T., and Hutter, F. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025.

Hoo, S. B., Müller, S., Salinas, D., and Hutter, F. The tabular foundation model TabPFN outperforms specialized time series forecasting models based on simple features. In *NeurIPS Workshop on Time Series in the Age of Large Models*, 2024.

Hunter, J. D. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9(3):90–95, 2007.

Imbens, G. W. Causal inference in the social sciences. *Annual Review of Statistics and Its Application*, 11, 2024.

Imbens, G. W. and Rubin, D. B. *Causal inference in statistics, social, and biomedical sciences*. Cambridge university press, 2015.

Kennedy, E. H. Towards optimal doubly robust estimation of heterogeneous causal effects. *Electronic Journal of Statistics*, 17(2):3008–3049, 2023.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

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

Kusner, M. J., Loftus, J., Russell, C., and Silva, R. Counterfactual fairness. volume 30, pp. 4069 – 4079, 2017.

Manber, U. *Introduction to algorithms: a creative approach*, volume 142. Addison-Wesley Reading, MA, 1989.

McElfresh, D., Khandagale, S., Valverde, J., Prasad C, V., Ramakrishnan, G., Goldblum, M., and White, C. When do neural nets outperform boosted trees on tabular data? *Advances in Neural Information Processing Systems*, 36: 76336–76369, 2023.

Müller, S., Hollmann, N., Arango, S. P., Grabocka, J., and Hutter, F. Transformers can do bayesian inference. In *International Conference on Learning Representations*, 2022.

Nagler, T. Statistical foundations of prior-data fitted networks. In *International Conference on Machine Learning*, pp. 25660–25676. PMLR, 2023.

Nguyen, T. and Grover, A. Transformer neural processes: Uncertainty-aware meta learning via sequence modeling. In *ICML*, 2022.

Nie, X. and Wager, S. Quasi-oracle estimation of heterogeneous treatment effects. *Biometrika*, 108(2):299–319, 2021.

Paszke, A. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.

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

Peters, J., Janzing, D., and Schölkopf, B. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

Reuter, A., Rudner, T. G., Fortuin, V., and Rügamer, D. Can transformers learn full Bayesian inference in context? *arXiv preprint arXiv:2501.16825*, 2025.

Robertson, J., Hollmann, N., Awad, N., and Hutter, F. FairPFN: Transformers can do counterfactual fairness. In *ICML 2024 Next Generation of AI Safety Workshop*, 2024.

Rosenbaum, P. R. and Rubin, D. B. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

Sauter, A., Salehkaleybar, S., Plaat, A., and Acar, E. Activa: Amortized causal effect estimation without graphs via transformer-based variational autoencoder. *arXiv preprint arXiv:2503.01290*, 2025.

Sauter, A. W. M., Acar, E., and Plaat, A. Causalplayground: Addressing data-generation requirements in cutting-edge causality research, 2024.

Shalit, U., Johansson, F. D., and Sontag, D. Estimating individual treatment effect: generalization bounds and algorithms. In *International conference on machine learning*, pp. 3076–3085. PMLR, 2017.

Sharma, A. and Kiciman, E. Dowhy: An end-to-end library for causal inference. *arXiv preprint arXiv:2011.04216*, 2020.

Shpitser, I. and Pearl, J. Identification of conditional interventional distributions. In *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, pp. 437–444, 2006.

Spirtes, P., Glymour, C., and Scheines, R. *Causation, prediction, and search*. Springer-Verlag. (2nd edition MIT Press 2000), 1993.

Varian, H. R. Causal inference in economics and marketing. *Proceedings of the National Academy of Sciences*, 113 (27):7310–7315, 2016.

Vasist, M., Rozet, F., Absil, O., Mollière, P., Nasedkin, E., and Louppe, G. Neural posterior estimation for exoplanetary atmospheric retrieval. *Astronomy & Astrophysics*, 672:A147, 2023.

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

Waskom, M. L. seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021. doi: 10.21105/joss.03021. URL https://doi.org/10.21105/joss.03021.

Wightman, L. F. LSAC national longitudinal bar passage study. *LSAC Research Report Series*, 1998.

Wildberger, J., Dax, M., Buchholz, S., Green, S., Macke, J. H., and Schölkopf, B. Flow matching for scalable simulation-based inference. *Advances in Neural Information Processing Systems*, 36:16837–16864, 2023.

Wu, X., Peng, S., Li, J., Zhang, J., Sun, Q., Li, W., Qian, Q., Liu, Y., and Guo, Y. Causal inference in the medical domain: A survey. *Applied Intelligence*, 54(6):4911–4934, 2024.

Xu, D. Q., Cirit, F. O., Asadi, R., Sun, Y., and Wang, W. Mixture of in-context prompters for tabular PFNs. In *The Thirteenth International Conference on Learning Representations*, 2025.

A. Proof of Proposition A.1

Theorem A.1. *Performing stochastic gradient descent according to Algorithm 1 corresponds to minimizing the expected forward Kullback-Leibler divergence between the conditional interventional distribution $p(y^{in}|\mathbf{x}^{in}, do(t^{in}), \psi)$ and the distribution $q_\theta(y^{in}|do(t^{in}), \mathbf{x}^{in}, \mathcal{D}^{ob})$ parameterized by the model,*

$$\mathbb{E}_{x^{in},t^{in},\mathcal{D}^{ob},\psi}\left[\mathbb{D}_{KL}\left[p(y^{in}|\mathbf{x}^{in}, do(t^{in}), \psi)||q_\theta(y^{in}|do(t^{in}), \mathbf{x}^{in}, \mathcal{D}^{ob})\right]\right]. \tag{3}$$

Here, the expectation is taken with respect to the data-generating distribution defined in Algorithm 1.

The proof follows from applying the conditional independences between variables implied by the data-generating process in Algorithm 1.

Let us try to provide some insight about Proposition A.1: (i) It does *not* state that we can estimate all causal effects in the traditional sense. To see this, note that the expectation is taken with respect to the synthetic data-generating process. We could even drop the assumption of independent noise terms in our SCMs, to train a model that covers the non-Markovian case, and the proposition would still hold. (ii) Moreover, since our prior over SCMs does *not* necessarily imply identifiability of causal effects, an ideal property of our model would be that $q_\theta(y^{in}|do(t^{in}), \mathbf{x}^{in}, \mathcal{D}^{ob})$ accurately captures the uncertainty in the outcome y arising from the unidentifiability of the causal effect of $do(t^{in})$ on y^{in}. Section D.1 discusses empirical results indicating that Do-PFN is indeed able to do so.

The risk for a single interventional data point when using the NLL loss, as in Algorithm 1 takes the following form:

$$\mathcal{R}_\theta = \int\int\int\int -\log(q_\theta(y^{in}|do(t^{in}), \mathbf{x}^{in}, \mathcal{D}^{ob}))p(\mathcal{D}^{ob}, t^{in}, y^{in}, \mathbf{x}^{in})d\mathcal{D}^{ob}dt^{in}dy^{in}d\mathbf{x}^{in} \tag{4}$$

Let's consider $p(\mathcal{D}^{ob}, t^{in}, y^{in}, \mathbf{x}^{in})$. Then we can obtain by first marginalizing out the distribution $p(\psi)$ of Structural Causal Models (SCMs) and, second, utilizing the factorization of the joint distribution implied by the data generating process in Algorithm 1:

$$p(\mathcal{D}^{ob}, t^{in}, y^{in}, \mathbf{x}^{in}) = \int p(\mathcal{D}^{ob}, t^{in}, y^{in}, \mathbf{x}^{in}, \psi)d\psi = \int p(y^{in}, \mathbf{x}^{in}|do(t^{in}), \psi)p(t^{in}|\mathcal{D}^{ob})p(\mathcal{D}^{ob}|\psi)p(\psi)d\psi \tag{5}$$

Now, we can use that

$$p(y^{in}, \mathbf{x}^{in}|do(t^{in}), \psi) = p(y^{in}|\mathbf{x}^{in}, do(t^{in}), \psi)p(\mathbf{x}^{in}|do(t^{in}), \psi).$$

Further:

$$p(\mathbf{x}^{in}|do(t^{in}), \psi)p(t^{in}|\mathcal{D}^{ob})p(\mathcal{D}^{ob}|\psi)p(\psi) = p(\mathcal{D}^{ob}, t^{in}, \mathbf{x}^{in}, \psi). \tag{6}$$

This implies

$$p(\mathcal{D}^{ob}, t^{in}, y^{in}, \mathbf{x}^{in}) = \int p(y^{in}|\mathbf{x}^{in}, do(t^{in}), \psi)p(\mathcal{D}^{ob}, t^{in}, \mathbf{x}^{in}, \psi)d\psi \tag{7}$$

Plugging this into equation 4 followed by using that the cross entropy between two distributions p and q is equal to the Kullback-Leibler divergence between p and q plus the entropy of p, formally $H(p, q) = H(p) + \mathbb{D}_{KL}(p||q)$, a fact used by Müller et al. (2022) and Barber & Agakov (2004) in analogous scenarios, yields:

$$\begin{aligned}\mathcal{R}_\theta = \int\int\int\int\int & -\log(q_\theta(y^{in}|do(t^{in}), \mathbf{x}^{in}, \mathcal{D}^{ob})) \\ & p(y^{in}|\mathbf{x}^{in}, do(t^{in}), \psi)p(\mathcal{D}^{ob}, t^{in}, \mathbf{x}^{in}, \psi)d\mathcal{D}^{ob}dt^{in}dy^{in}d\mathbf{x}^{in}d\psi \\ = & \int\int\int\int \mathbb{D}_{KL}\left[p(y^{in}|\mathbf{x}^{in}, do(t^{in}), \psi)||q_\theta(y^{in}|do(t^{in}), \mathbf{x}^{in}, \mathcal{D}^{ob})\right] \\ & p(\mathcal{D}^{ob}, t^{in}, \mathbf{x}^{in}, \psi)d\mathcal{D}^{ob}dt^{in}d\mathbf{x}^{in}d\psi + C \quad (8)\end{aligned}$$

This implies that minimizing $\mathcal{R}_\theta$ results in a (forward) Kullback-Leibler optimal approximation of $p(y^{in}|do(t^{in}), \psi, \mathbf{x}^s)$ with the model $q_\theta(y^{in}|do(t^{in}), \mathbf{x}^s, \mathcal{D}^{ob})$ **in expectation** over the data simulated from $p(\psi, \mathcal{D}^{ob}, t^{in}, \mathbf{x}^{in})$.

Please note that analogous to PFNs, the optimality only holds when the expectation is taken with respect to the synthetic data-generating process. However, theoretical results by Nagler (2023) and a plethora of empirical findings regarding the transferability of PFNs to real-world scenarios, as well as related approaches (Hollmann et al., 2025; Hoo et al., 2024; Reuter et al., 2025), provide evidence that synthetic prior fitting can lead to strong real-world performance.

Algorithm 1 Prior-fitting with SGD. Do-PFN is pre-trained on pairs of synthetic observational and interventional datasets; the model is trained to predict interventional outcomes y^{in} given a covariate-vector $\mathbf{x}^{in}$, the value of an intervention t^{in} and an observational dataset $\mathcal{D}^{ob}$.

Input: Number of datasets N, minimum and maximum observational samples M_{min}, M_{max}, learning rate α
for $i = 1, 2, \ldots, N$ **do**
  Draw $\psi_i \sim p(\psi)$ {Draw an SCM}
  Initialize $\mathcal{D}_i^{ob} \leftarrow \emptyset$
  Draw $M_{ob} \sim$ Uniform($\{M_{min}, M_{min}+1, \ldots, M_{max}\}$) {Number of observational data points}
  for $j = 1, \ldots, M_{ob}$ **do**
    Sample noise $\epsilon_j \sim p(\epsilon)$
    Draw $y_j^{ob}, t_j^{ob}, \mathbf{x}_j^{ob} \sim p(y^{ob}, t^{ob}, \mathbf{x}^{ob} | \psi_i, \epsilon_j)$
    $\mathcal{D}_i^{ob} \leftarrow \mathcal{D}_i^{ob} \cup \{(y_j^{ob}, t_j^{ob}, \mathbf{x}_j^{ob})\}$
  end for
  Initialize $\mathcal{D}_i^{in} \leftarrow \emptyset$
  Set $M_{in} = M_{max} - M_{ob}$
  for $k = 1, 2, \ldots, M_{in}$ **do**
    Sample noise $\epsilon_k \sim p(\epsilon)$
    Draw $\mathbf{x}_k^{in} \sim p(\mathbf{x}^{in} | \psi_i, \epsilon_k)$ {Pre-treatment values of covariates}
    Draw $t_k^{in} \sim p(t^{in})$ {Draw value for intervention}
    Draw $y_k^{in} \sim p(y^{in} | do(t_k^{in}), \psi_i, \epsilon_k)$
    $\mathcal{D}_i^{in} \leftarrow \mathcal{D}_i^{in} \cup \{(y_k^{in}, t_k^{in}, \mathbf{x}_k^{in})\}$
  end for
  Compute $\mathcal{L}_i(\theta) = \sum_{k=1}^{M_{in}} - \log q_\theta(y_k^{in} | do(t_k^{in}), \mathbf{x}_k^{in}, \mathcal{D}_i^{ob})$
  $\theta \leftarrow \theta - \alpha \nabla \mathcal{L}_i(\theta)$ {Gradient descent}
end for

B. Details on the prior-fitting procedure

In this section, we provide the details of the data-generating process in Algorithm 1 that represents our modeling assumptions. From the perspective of PFNs, this data-generating process represents Do-PFN's "prior". Concretely, our prior-fitting procedure involves the following key steps:

Sampling the SCM: First, for every iteration $i = 1, 2, \ldots, N$, an SCM ψ_i is sampled. This is achieved by first sampling a DAG via topological sorting of vertices (Manber, 1989). For each node k in the graph, we uniformly at random sample the nonlinearity γ to be one of the following functions: the quadratic function $x \mapsto x^2$, $x \mapsto \text{ReLU}(x)$, and $x \mapsto \tanh(x)$. We define the mechanisms in the SCM to take the form of an additive noise model (ANM) $f_k(z_{\text{PA}(k)}, \epsilon_k) = \gamma(\sum_{l \in \text{PA}(k)} w_l z_l) + \epsilon_k$. The weights of the SCM are sampled using a Kaiming initialization $w_l \sim \text{Uniform}(-\frac{1}{\sqrt{|\text{PA}(k)|}}, \frac{1}{\sqrt{|\text{PA}(k)|}})$ for $l = 1, 2, \ldots, |\text{PA}(k)|$, where $|\text{PA}(k)|$ denotes the number of parents of node k.

Sampling observational data: Next, observational data is sampled according to the SCM ψ_i. More specifically, a dataset $\mathcal{D}_i^{ob}$ is filled with M_{ob} data points, where the number of data points is drawn uniformly between $M_{min} = 10$ and $M_{max} = 2,200$. Each element in $\mathcal{D}_i^{ob}$ is generated by first sampling a noise vector $\epsilon_j \sim p(\epsilon)$ which is passed through the SCM to generate each element $y_j^{ob}, t_j^{ob}, \mathbf{x}_j^{ob}$.

Sampling interventional data: To sample an element in the interventional dataset $\mathcal{D}_i^{in}$, with $M^{in} = M_{max} - M^{ob}$ data points, first, a noise vector $\epsilon_k \sim p(\epsilon)$ is sampled again. Subsequently a covariate-vector $\mathbf{x}_k^{in}$ is sampled from $p(\mathbf{x} | \psi_i, \epsilon_k)$. This ensures that the vector $\mathbf{x}_k^{in}$ characterizes the subject k prior to the intervention. After sampling the value for the treatment t_k^{in}, we perform the intervention $do(t_k^{in})$ and sample y_k^{in} from the intervened-upon SCM using the same noise ϵ_k as before[3].

[3]Because the noise is held constant to produce the pre-interventional covariate-vector, $\mathbf{x}_k^{in}$, and interventional outcomes, y_k^{in}, this process can also be seen as simulating counterfactuals or single potential outcomes.

Gradient descent For each iteration $i = 1, 2, \ldots, N$, an observational dataset $\mathcal{D}_i^{ob}$ and an interventional dataset $\mathcal{D}_i^{in}$ are generated. These datasets are utilized to compute the negative log-likelihood under our model q_θ. This loss is calculated with respect to predicting the interventional outcome y_k^{in} based on the value of the intervention t_k^{in}, the covariates $\mathbf{x}_k^{in}$, and the observational dataset $\mathcal{D}_i^{ob}$. Subsequently, a gradient step is taken on the negative log-likelihood. In practice, we perform mini-batch stochastic gradient descent using the Adam optimizer (Kingma & Ba, 2014).

C. Experimental details

C.1. Synthetic case studies

Case studies We introduce several causal case studies that pose unique challenges for causal effect estimation, requiring adjustment based on the satisfaction of front-door and back-door criteria (Figure 2). We additionally generate three case studies not visualized in Figure 2, which ablate over smaller dataset sizes $M_{max} \sim \text{Uniform}([5, 100])$, complex graph structures with number of nodes $K \sim \text{Uniform}([4, 10])$, and finally a "Common Effect" case study which we show to be easily solved even by standard regression models (Appendix Figure 11).

Synthetic data generation For each case study visualized in Figure 2, we independently sample 100 datasets with the corresponding graph structure, varying the SCM parameters as described in Appendix B. We also vary the number of samples, standard deviation of noise terms, as well as edge weights and non-linearities. The structural equations for our case studies, as well as details regarding how SCM parameters are sampled, are provided in Appendix C.1 and Appendix Table 1.

The standard deviation σ_{exo} of the exogenous noise is sampled from $\sigma_{exo} \sim \text{Uniform}([1, 3])$. For the standard deviation of the additive noise terms, we sample $\beta \sim \text{Beta}(1, 5)$, and then set $\sigma_\epsilon = 0.3 \cdot \beta$.

The functions f_{z_k} take the form $f_a(z_k, \epsilon) = \gamma(\sum_{l \in \text{PA}(k)} w_l z_l) + \epsilon$. The weights of the SCM are sampled using a Kaiming initialization $w_l \sim \text{Uniform}(-\frac{1}{\sqrt{|\text{PA}(k)|}}, \frac{1}{\sqrt{|\text{PA}(k)|}})$ for $l = 1, 2, \ldots, |\text{PA}(k)|$, where $|\text{PA}(k)|$ denotes the number of parents of node k. The nonlinearities f_a are sampled uniformly at random from the set $\{f_1, f_2, f_3\}$ where $f_1(x) = x^2$, $f_2(x) = \tanh(x)$ and $f_3 = ReLU(x) = \max(0, x)$. Details on the case studies can be found in Table 1.

In this section we provide the details on all considered case studies from Section C.1.

Observed Confounder	**Observed Mediator**	**Confounder + Mediator**
$\epsilon_t, \epsilon_y \sim \mathcal{N}(0, \sigma_\epsilon)$ $x_1 \sim \mathcal{N}(0, \sigma_{exo})$ $t = f_t(x_1, \epsilon_t)$ $y = f_y(x_1, t, \epsilon_y)$	$\epsilon_{x_1}, \epsilon_y \sim \mathcal{N}(0, \sigma_\epsilon)$ $t \sim \text{Uniform}(\{0, 1\})$ $x_1 = f_{x_1}(t, \epsilon_{x_1})$ $y = f_y(x_1, t, \epsilon_y)$	$\epsilon_{x_1}, \epsilon_t, \epsilon_y \sim \mathcal{N}(0, \sigma_\epsilon)$ $x_2 \sim \mathcal{N}(0, \sigma_{exo})$ $t = f_t(x_1, \epsilon_t)$ $x_1 = f_{x_1}(t, \epsilon_{x_1})$ $y = f_y(x_1, x_2, t, \epsilon_y)$
Unobserved Confounder	**Back-Door Criterion**	**Front-Door Criterion**
$\epsilon_t, \epsilon_y \sim \mathcal{N}(0, \sigma_\epsilon)$ $x_1, x_2 \sim \mathcal{N}(0, \sigma_{exo})$ $t = f_t(x_1, x_2, \epsilon_t)$ $x_1 = f_{x_1}(t, \epsilon_{x_1})$ $y = f_y(x_1, x_2, t, \epsilon_y)$	$\epsilon_{x_1}, \epsilon_t, \epsilon_y \sim \mathcal{N}(0, \sigma_\epsilon)$ $x_2 \sim \mathcal{N}(0, \sigma_{exo})$ $t = f_t(x_1, \epsilon_t)$ $x_1 = f_{x_1}(x_2, \epsilon_{x_1})$ $y = f_y(x_1, x_2, \epsilon_y)$	$\epsilon_{x_1}, \epsilon_t, \epsilon_y \sim \mathcal{N}(0, \sigma_\epsilon)$ $x_2 \sim \mathcal{N}(0, \sigma_{exo})$ $t = f_t(x_1, \epsilon_t)$ $x_1 = f_{x_1}(x_2, \epsilon_{x_1})$ $y = f_y(x_1, x_2, \epsilon_y)$

Table 1. Structural equations for all causal case studies.

C.2. Evaluation metric

We evaluate our results in terms of normalized mean squared error (MSE), as it allows results to be compared across datasets. We define MSE below:

$$\text{MSE}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{n} \sum_{i=1}^{n} \left[\frac{y_i - \hat{y}_i}{\max(\mathbf{y}) - \min(\mathbf{y})} \right]^2 \tag{9}$$

C.3. Description of baselines

C.3.1. Conditional interventional distribution prediction

- **Dont-PFN**: a TabPFN regression model (Hollmann et al., 2025) pre-trained on our prior to approximate the posterior predictive distribution (PPD) $p(y^{ob}|\mathbf{x}^{ob}, \mathcal{D}^{ob})$.
- **DoWhy (Int./Cntf.)**: a structural causal model ψ fit to observational samples $\mathcal{D}^{ob}$ and the graph structure $\mathcal{G}_\psi$. The constructed SCM is used to predict interventional (Int.) and counterfactual (Cntf.) outcomes. Crucially, TabPFNClassifier and TabPFNRegressor models (Hollmann et al., 2025) are used approximate binary and continuous structural equations.
- **Random Forest**: an ensemble of decision trees (Breiman, 2001) trained on $\mathcal{D}^{ob}$
- **Do-PFN-Graph**: a TabPFN regression model pre-trained for 5 hours to approximate the CID $p(y^{do}|\mathbf{x}^{ob}, \mathcal{D}^{ob})$ on *fixed* graph structures from our case studies.
- **Do-PFN-Short**: a TabPFN regression model pre-trained for 20 hours on varying graph structures of up to 5 nodes to approximate the CID $p(y^{do}|\mathbf{x}^{ob}, \mathcal{D}^{ob})$
- **Do-PFN**: a TabPFN regression model pre-trained for 40 hours on varying graph structures of up to 10 nodes to approximate the CID $p(y^{do}|\mathbf{x}^{ob}, \mathcal{D}^{ob})$
- **Do-PFN-Mixed**: Do-PFN-Short pre-trained varying whether additive noise terms are sampled from zero-mean Gaussian, Laplacian, Students-T, and Gumbel distributions.

C.3.2. Conditional average treatment effect estimation

- **Causal Forest (DML)**: a double machine learning (DML) approach based on (Wager & Athey, 2018) that combines multiple causal trees to estimate conditional average treatment effects (CATEs). Hyperparameters are tuned using exhaustive search.
- **Do-PFN-CATE**: Do-PFN applied to predict the specific quantity:

$$\hat{\tau} = \mathbb{E}_{y^{in} \sim q_\theta(y^{in}|do(t^{in}=1), \mathbf{x}^{in}, \mathcal{D}^{ob})}[y^{in}] - \mathbb{E}_{y^{in} \sim q_\theta(y^{in}|do(t^{in}=0), \mathbf{x}^{in}, \mathcal{D}^{ob})}[y^{in}] \tag{10}$$

- **DoWhy-CATE (Int./Cntf.)**: DoWhy (Int./Cntf.) used as an S-Learner (Künzel et al., 2019) to estimate conditional average treatment effects (CATEs). When DoWhy (Cntf.) is used, noise terms are inferred and held constant across forward passes.
- **Dont-PFN-CATE**: Dont-PFN used as an S-Learner (Künzel et al., 2019) to estimate conditional average treatment effects (CATEs)

C.4. Software

We use Pytorch (Paszke, 2019) to implement all our experiments. Our implementation of the causal prior is based on the Causal Playground library (Sauter et al., 2024) and the codebase used for TabPFN (Hollmann et al., 2023; 2025). We use Matplotlib (Hunter, 2007), Autorank (Herbold, 2020) and Seaborn (Waskom, 2021) for our plots.

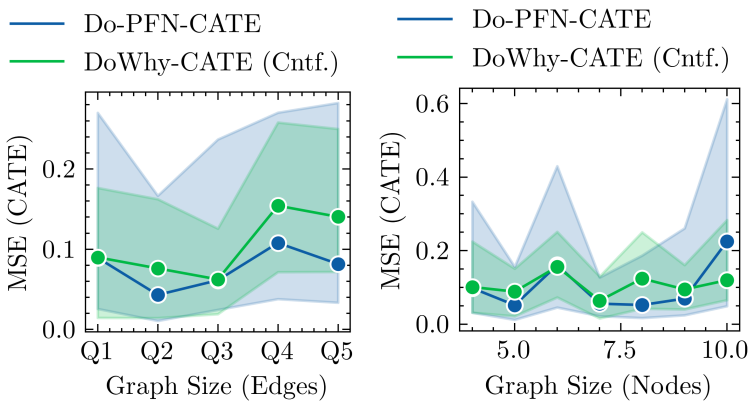

Figure 5. **Graph size and complexity**: Ablating the performance of Do-PFN and DoWhy across different graph complexities. We find that Do-PFN maintains its competitive performance for increasingly complex graphs.

D. Supplementary Results

D.1. Ablation studies

Dataset size and complexity First, we observe in Figure 6 (right) that Do-PFN exhibits strong performance on small datasets. In an evaluation of MSE in CATE estimation across datasets with a varying number of samples drawn such that $M_{max} \sim \text{Uniform}([5, 2000])$, we observe that Do-PFN-CATE performs competitively with DoWhy-CATE (Cntf.) and its performance continues to improve and becomes more consistent as dataset size grows. We also find that Do-PFN performs competitively to DoWhy-CATE (Cntf.) across graph complexities, with slightly larger improvements for more complex graphs (Appendix D.1). Furthermore, we find that Do-PFNs can effectively use additional data points to alleviate increasing levels of noise (Appendix D.1).

Treatment effect We also show in Figure 6 (left) that Do-PFN remains relatively consistent in MSE across different base rate levels of the average treatment effect (ATE). This result shows that Do-PFN is robust to different magnitudes of the ATE, which is beneficial in cases of problem misspecification, for example when a specified treatment does not influence an outcome.

Uncertainty calibration Next, we explore Do-PFN's uncertainty calibration, by visualizing the prediction interval coverage probability (PICP) in Figure 6. A PICP curve equal to the 45-degree diagonal corresponds to a model consistently yielding prediction intervals with exactly the desired coverage. Being above the diagonal corresponds to under-confident and being below the diagonal to over-confident prediction intervals. First, we observe that Do-PFN's uncertainty is slightly under-confident for theoretically identifiable case studies (the full set of calibration plots can be found in Appendix D.2). In the "Unobserved Confounder" case study, the model's high uncertainty is reflected by a relatively large entropy in its output distribution (Appendix 12). However, our PICP results show that the model's uncertainty for this case study is correctly calibrated.

Graph size and complexity We evaluate Do-PFN's performance across data generated from graphs of increasing complexity, sampling 500 datasets generated with graph structures consisting of 4 to 10 nodes and 2 to 43 edges. The result is visualized in Figure 5. We note that while our data-generating mechanisms are relatively simple from a mathematical perspective, graph identification is a combinatorially hard problem, with the number of unique Directed Acyclic Graphs (DAGs) of 10 nodes reaching 4.17×10^{18}. Do-PFN performs competitively to DoWhy-CATE (Cntf.) across graph complexities, with slightly larger improvements for more complex graphs.

Robustness to additive noise We also highlight in Figure 7 (left) that the performance of Do-PFN decreases with an increase in the standard deviation of additive noise, which corresponds to a larger irreducible error. However, we also observe in Figure 7 (center-right) that Do-PFN's performance for different levels of additive noise seems to increase with dataset size. This means that the MSE for datasets with a certain amount of additive noise can be reduced up to a certain

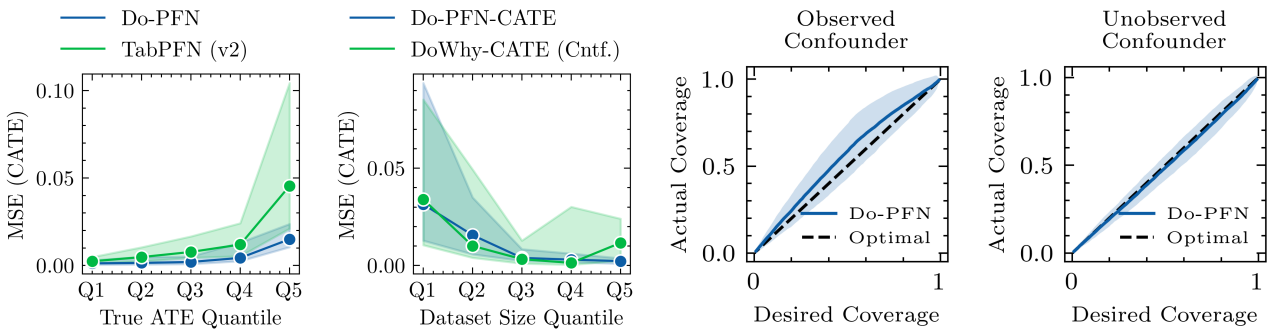

Figure 6. **Ablation studies**: Do-PFN is relatively insensitive to changing base rates of ATEs (left), and improves with increased number of observational samples (center-left). The two plots on the right visualize the Prediction interval coverage probability (PICP) of Do-PFN. The center-right plot shows that in the "observed confounder" case-study the model is slightly under-confident while, crucially, Do-PFN is correctly unconfident for the unidentifiable "Unobserved Confounder" case.

extent with more data.

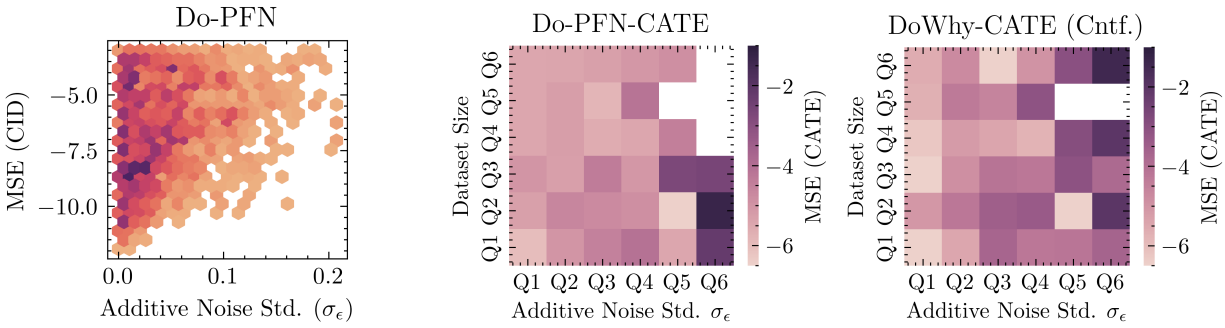

Figure 7. **Robustness to additive noise**: Evaluation of Do-PFN's performance in CID prediction and CATE estimation across different quantiles (Q1-Q5) of additive noise standard deviation. The density plot (left) shows that Do-PFN's performance decreases with (irreducible) additive noise. However, the heatmap (center) shows that for datasets with similar additive noise levels, Do-PFN's performance increases with dataset size. This effect is even stronger than for DoWhy-CATE (Cntf.).

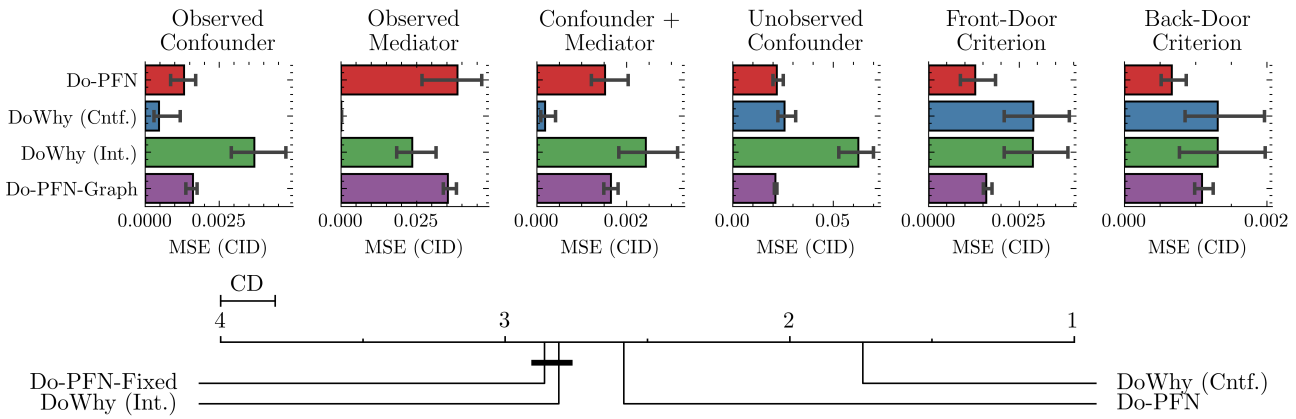

Figure 8. **Gold-standard comparison (CID)**: Bar-charts and critical difference (CD) diagrams depicting distributions of normalized mean squared error (MSE) of Do-PFN and our "gold-standard" baselines in conditional interventional distribution (CID) estimation on our six synthetic case studies. Do-PFN significantly outperforms Do-PFN-Graph and DoWhy (Int.), while performing competitively well with DoWhy (Cntf.).

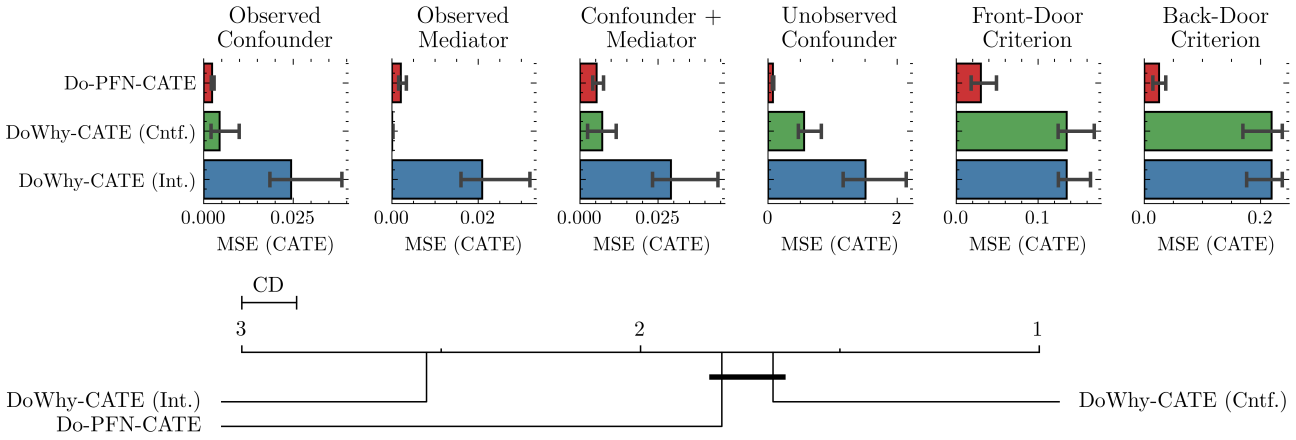

Figure 9. **Gold-standard comparison (CATE)**: Bar-charts and critical difference (CD) diagrams depicting distributions of normalized mean squared error (MSE) of variants of Do-PFN and our baselines in conditional average treatment effect (CATE) estimation on our six synthetic case studies. Do-PFN-CATE outperforms DoWhy-CATE (Int.) and performs competitively with DoWhy-CATE (Cntf.).

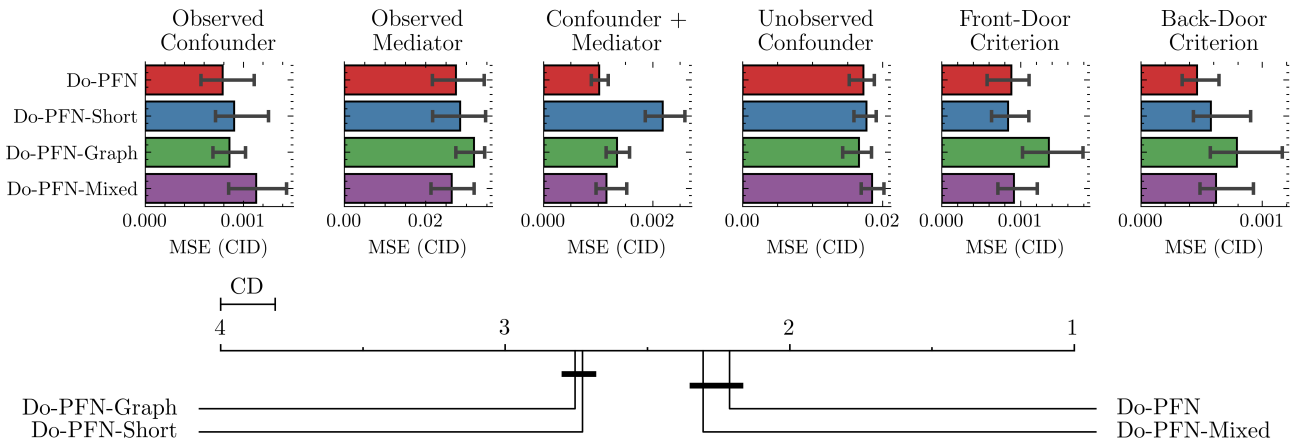

Figure 10. **Comparison of Do-PFN variants (CID)**: Bar-charts and critical difference (CD) diagrams depicting distributions of normalized mean squared error (MSE) of Do-PFN variants in conditional interventional distribution (CID) estimation on our six synthetic case studies. Do-PFN significantly outperforms other variants except Do-PFN-Mixed, which achieves statistically similar performance in half the pre-training time.

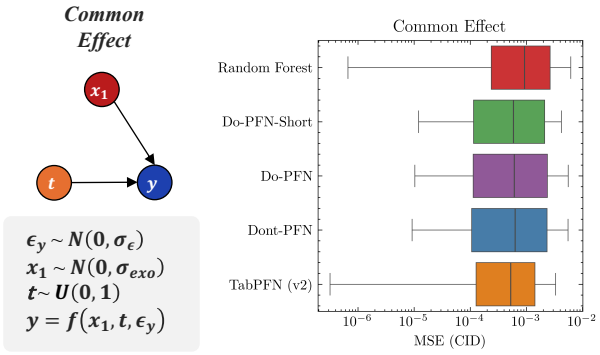

Figure 11. **Common effect case study**: Visualization of graph structure and structural equations (left) for our "common effect" case study, as well as bar plots depicting distributions of normalized mean squared error (MSE) of Do-PFN variants compared to regression baselines in conditional interventional distribution prediction. Regression baselines perform similarly to Do-PFN variants, as the intervention does not cause a distribution shift between $\mathcal{D}^{ob}$ and $\mathcal{D}^{in}$.

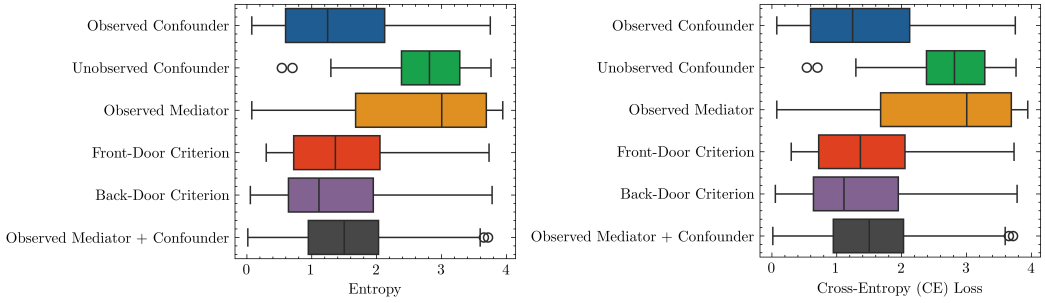

Figure 12. **Uncertainty quantification**: Cross-entropy (CE) loss (right) and entropy (left) of Do-PFN's bar distribution output. Do-PFN is highly uncertain on the "Unobserved Confounder" case study due to unidentifiability. Do-PFN also shows high uncertainty on the "Observed Mediator" case study, which we argue is due to its only exogenous term being a binary variable, causing the continuous effect in the outcome to only come from additive noise.)

D.2. Further calibration plots

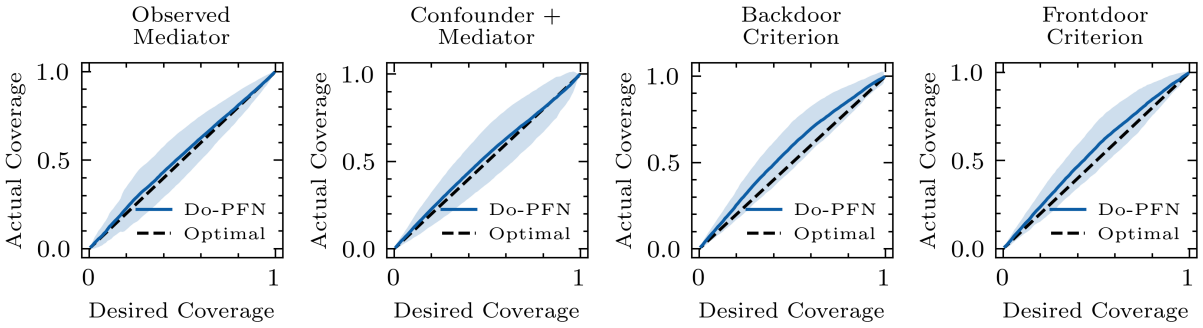

Figure 13. **Uncertainty calibration**: Prediction interval coverage probability (PICP) plots for the "Observed Mediator", "Confounder + Mediator", "Backdoor Criterion" and "Frontdoor Criterion" cases. The solid blue line shows the coverage and standard deviation achieved by Do-PFN, spanning desired probabilities from 0 to 1. The dashed line represents the ideal calibration achievable with access to the ground-truth CID. Do-PFN is slightly under-confident for identifiable case studies, and, crucially, correctly unconfident for the "Unobserved Confounder" case.

E. Hybrid synthetic-real-world data

To assess whether Do-PFN's performance on our synthetic case-studies also extends to real-world-data, we conduct experiments on two real-world datasets with agreed-upon causal graphs (Figure 14). Those causal graphs allow us to simulate gold-standard outcomes using the DoWhy library (Sharma & Kiciman, 2020), which makes the evaluation of Do-PFN and our baselines possible.

The **key takeaway** of these results is that Do-PFN's strong performance on synthetic data seems to extend well to real-world data, producing similar predictions to our gold-standard baselines which receive access to a widely accepted causal graph.

E.1. Amazon sales

The Amazon sales dataset (Blöbaum et al., 2024) contains data on the effect of special shopping events ("Shopping Event?") on the profit made from smartphone sales ("Profit"). It further provides variables with information on the spending on ad campaigns ("Ad Spend"), the price of the device ("Unit Price"), the number of phones sold ("Sold Units"), the number of page view ("Page Views"), the revenue that day ("Revenue") and the operational cost ("Operational Cost"). Those eight variables are connected via the causal graph in Figure 14.

In terms of predicting interventional outcomes, we find that DoPFN has a substantially better normalized mean-squared-error (MSE) score than Dont-PFN, Random Forest, and TabPFN (v2). (Plot on the left in Figure 15). DoWhy (Int.), which we provide with the exact underlying graph, has the best MSE value that is close to zero. For CATE predictions, Do-PFN-CATE has a noticeably lower median MSE value than Dont-PFN-CATE and Causal Forest, which both have a relatively large variance in terms of their performance. (Plot on the right in 15). Figure 16 visualizes the predictions of Do-PFN and our baselines vs. the gold-standard interventional outcomes for CID prediction and CATE estimation. Those plots detail the results above showing that Do-PFN's predictions closely align with the gold-standard targets.

E.2. Law school admissions

The law school admissions dataset (Figure 14) was drawn from the 1998 LSAC National Longitudinal Bar Passage Study (Wightman, 1998) and was made popular in the realm of counterfactual fairness due to its appearance in Kusner et al. (2017), where the variable "Race" was treated as a protected attribute. We note that we do not address the topic of algorithmic fairness, but would like to highlight that the ability of Do-PFN to predict interventional outcomes on demographic information could be a fruitful application in model-bias assessment.

We delve into the effect of "Race"[4] on first-year-average ("FYA"), which is mediated by two variables: undergraduate

[4]We note that Race in the lawschool dataset is typically treated as a binary variable. We very much disagree with this formulation, and

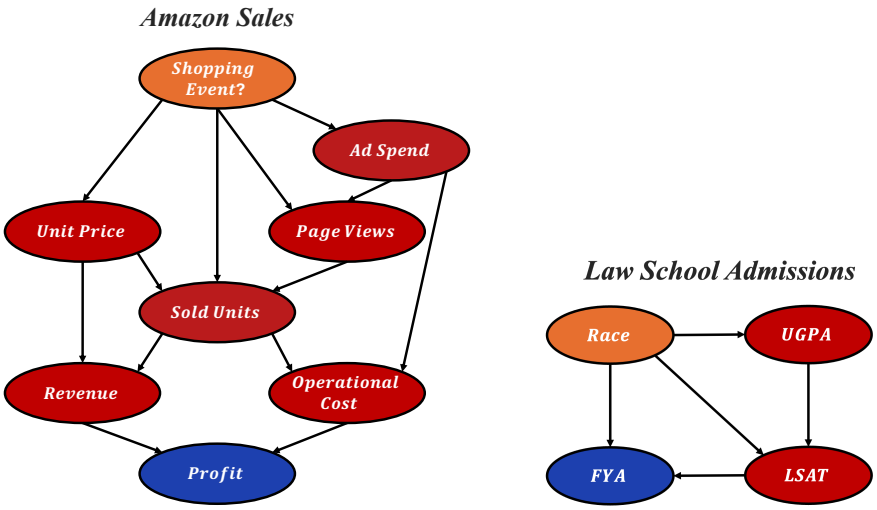

Figure 14. **Real-world case studies**: Agreed-upon causal graphs for our two real-world case-studies: Amazon Sales and Law School Admissions.

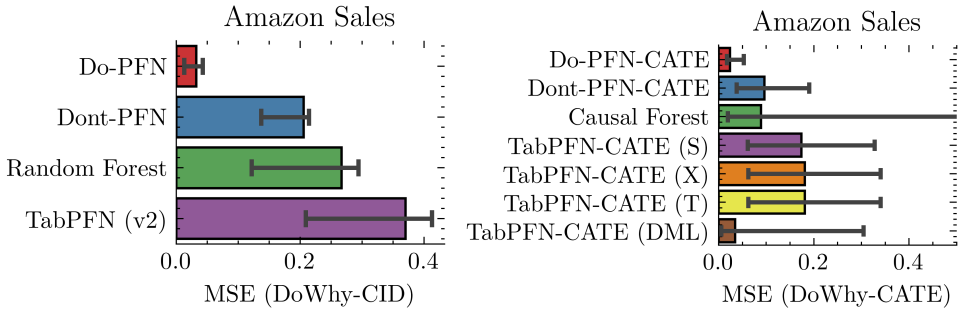

Figure 15. **Amazon sales**: Bar-charts depicting distributions of normalized mean squared error (MSE) of Do-PFN-CATE and our causal baselines in interventional outcome prediction (left) and conditional average treatment effect (CATE) estimation (right).

grade-point-average (”UGPA”) and ”LSAT”, a law school entrance exam in the United-States.

In terms of CID prediction (Figure 18 left) and CATE estimation (Figure 18 right), we find that Do-PFN outperforms all baselines in its approximation of both quantities. We do however, observe especially strong performance in CATE estimation, where Do-PFN performs significantly better than S-learner and double machine learning (DML) approaches.

This result is mirrored in Figure 17, where we visualize the match between baseline predictions and gold standard outcomes. Interestingly, we observe that while Do-PFN is yet to estimate the gold-standard CID values, it does provide a better separation in predictions between interventions $do(0)$ and $do(1)$. This outcome is consistent with (Robertson et al., 2024), which shows that a similar strategy of estimating fair outcomes by taking average predictions of only *simulating* interventions on the protected attribute and passing this data into standard classification models only removes the *direct-effect* of discrimination. We hypothesize that S-learner approaches thus only model the *direct-effect* of the intervention on the outcome, and thus fail to include the indirect effects through mediators.

acknowledge that the term ”ethnicity” better describes this complex social construct.

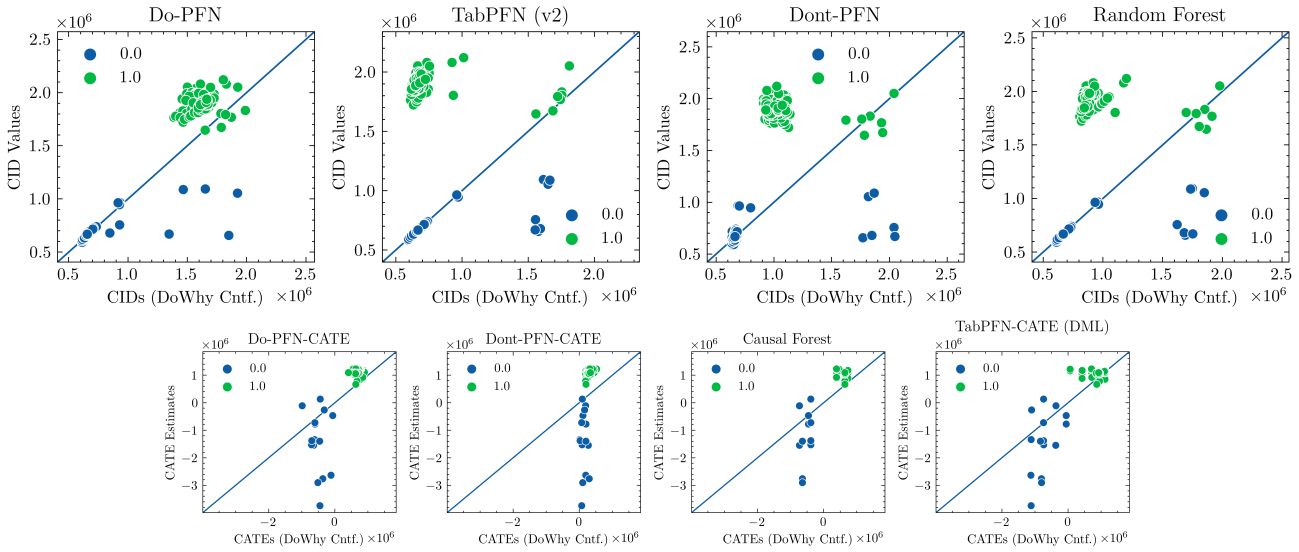

Figure 16. **Amazon Sales**: Scatter plots depicting the match between baseline predictions with gold standard outcomes produced by Do-Why-(CATE) (Int./Cntf.). Green scatter points represent individuals for which the intervention $do(ShoppingEvent = 1)$ is applied, while blue points represent $do(ShoppingEvent = 0)$.

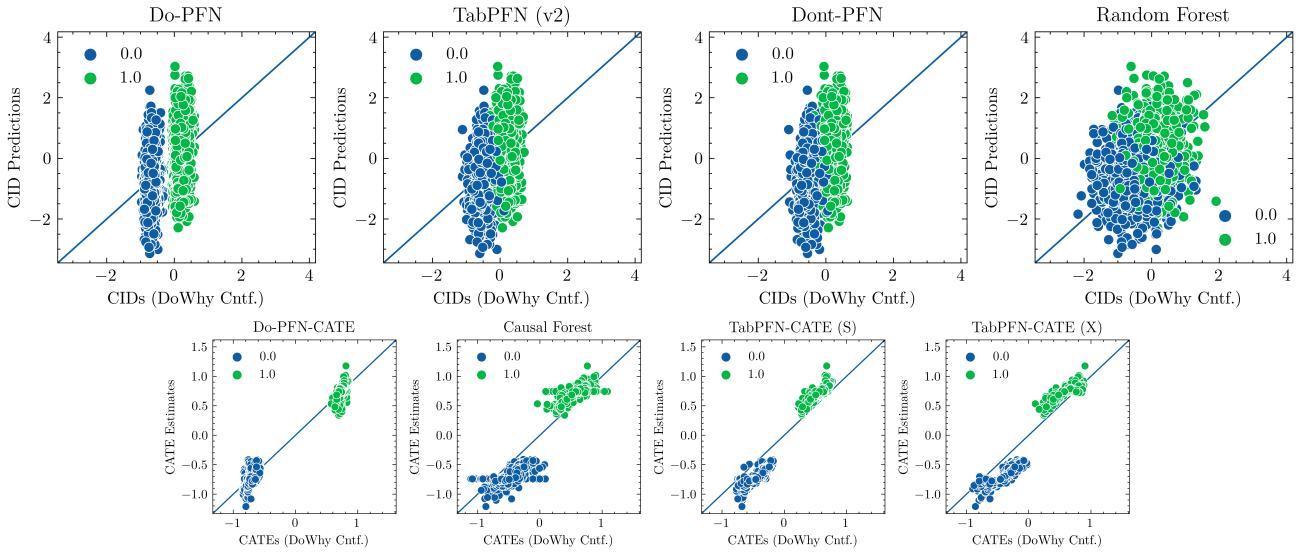

Figure 17. **Law school admissions**: Scatter plots depicting the match between baseline predictions with gold standard outcomes produced by Do-Why-(CATE) (Int./Cntf.). Green scatter points represent individuals for which the intervention $do(Race = 1)$ is applied, while blue points represent the intervention $do(Race = 0)$.

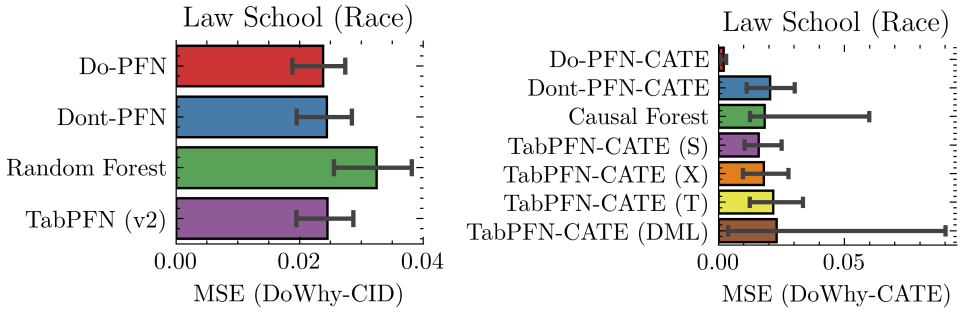

Figure 18. **Law school admissions**: Bar-charts depicting distributions of normalized mean squared error (MSE) of Do-PFN-CATE and our causal baselines in interventional outcome prediction (left) and conditional average treatment effect (CATE) estimation (right).