DO-PFN: IN-CONTEXT LEARNING FOR CAUSAL EFFECT ESTIMATION

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

[1]ELLIS Institute Tübingen, Tübingen, Germany
[2]Max Planck Institute for Intelligent Systems, Tübingen, Germany
[3]University of Cambridge, Cambridge, United Kingdom
[4]University of Freiburg, Freiburg, Germany
[5]Prior Labs, Freiburg, Germany

June 13, 2025

ABSTRACT

Estimation of causal effects 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 often 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 and 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 and Robins, 2010; Imbens and Rubin, 2015).

[*]Equal contribution
[†]Equal supervision

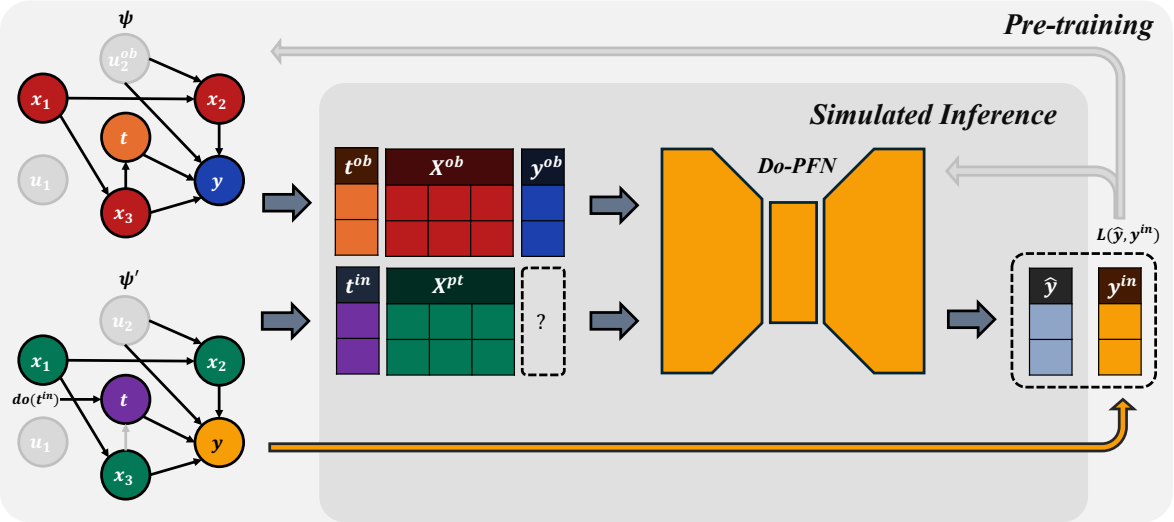

Figure 1: **Do-PFN overview**: Do-PFN performs in-context learning (ICL) for causal effect estimation, predicting conditional interventional distributions (CIDs) based on observational data alone. In pre-training, a large number of structural causal models (SCMs) is sampled. For each SCM, we sample an 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 $\mathcal{D}^{in} = \{(t_k^{in}, \mathbf{x}_k^{pt}, y_k^{in})\}_{k=1}^{M^{in}}$. To simulate inference, we input (t^{in}, x^{pt}) 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*. In real-world applications, 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.

Under the unconfoundedness assumption, a variety of estimation techniques have been developed, including causal forests (Wager and Athey, 2018), or "doubly robust" methods that combine propensity score modeling and outcome regression (Chernozhukov et al., 2018).

Many applications of causality involve tabular data, and prior-data fitted networks (PFNs) (Müller et al., 2022) have recently transformed the landscape of tabular machine learning. TabPFN (Hollmann et al., 2023, 2025), an application of PFNs to tabular classification tasks, was initially met with skepticism, arguably because of its radically different working principle compared to other state-of-the-art tabular machine learning methods. In a nutshell, TabPFN is a model that takes as input an entire dataset of labeled training data points $\{(\mathbf{x}_i, y_i)\}_{i=1}^{M}$ and unlabeled test data points $\boldsymbol{x}_{\text{new}}$ and "completes" it with an *in-context* prediction of the corresponding y_{new}, akin to a large language model answering a question by producing a likely text continuation. The approach can handle datasets of various sizes and dimensionality.

TabPFN is trained to execute this completion task across millions of synthetic datasets, for each dataset computing the likelihood of the true y_{new} under the PFN's predictive distribution and performing one step of stochastic gradient descent on this likelihood. The *context* refers to the chunk of data that is fed into the model at a time. In a language model, these are thousands of words; in the case of TabPFN it is a whole training set plus a query data point; and in our case also an added intervention. *In-context learning* refers to the ability to output a desired quantity based on what is provided in the context; the term *learning* indicates that this solves a task that would classically often require learning (e.g., estimating a regression function and using it to predict a target). For us, the desired quantity is the effect of an intervention. Training such a model then consists of *meta-learning* (across millions of meta-training datasets) the ability to provide the desired answer given a context. The term *meta* indicates that it is an outer loop around a procedure that itself is already a form of learning or estimation.

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. Due to the delicate nature of causal inference, the sensitivity of the tasks associated with it, and the scarcity of real-world causal effect estimation data with ground truth, we will do so by systematically exploring synthetic data with known ground truth and two real-world datasets with widely agreed-upon causal graphs.

Recent developments have shown that certain limitations in inferring causal structures and causal effects can be addressed by using multi-domain data in the form of mixtures of i.i.d. observational data (Guo et al., 2023, 2024). Interestingly, PFNs also leverage pre-training on a mixture of i.i.d. data to meta-learn how to solve predictive tasks at test time. We thus hypothesize that causal tasks could also be addressed through meta-learning on multi-domain data. As a first step, our goal is to extend PFNs to the problem of estimating conditional interventional distributions (CIDs).

In contrast to TabPFN, we not only simulate observational tabular data in order to predict a target feature. Rather, we additionally simulate causal interventions, teaching our model, which we call *Do-PFN*, to meta-learn how to perform causal inference.

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 interventional 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 CATE, Do-PFN (1) achieves competitive performance with baselines that have access to the true causal graph (typically not available in practice) and (2) statistically significantly outperforms standard regression models in predicting interventional outcomes as well as common methods for estimating causal effects. Our ablations show that Do-PFN works well on small datasets, is robust to varying base rates of the average treatment effect, and performs consistently on large graph structures.[3]

2 Background and related work

Structural causal models Structural causal models (SCMs; Pearl, 2009; Peters et al., 2017) represent the structure of a data-generating process. The first component of an SCM ψ is a directed acyclic graph (DAG) $\mathcal{G}_\psi$, which we assume to have K nodes, each representing a variable z_k. Furthermore, the SCM specifies the mechanisms to generate the variables from their (causal) parents 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. We use $\epsilon := (\epsilon_1, \epsilon_2, \ldots, \epsilon_K)$ to denote the vector comprising the noise terms. In our simulations these will be taken as jointly independent, but our methodolody does not require this.

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 conditional interventional distribution (**CID**; Shpitser and 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 conditional average treatment effects (**CATE**s): $\tau(x) := \mathbb{E}\left[y|do(1), \mathbf{x}\right] - \mathbb{E}\left[y|do(0), \mathbf{x}\right]$.

Estimating causal effects Various methods allow for the direct estimation of causal effects from experimental data (Shalit et al., 2017; Kennedy, 2023; Nie and 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. Back-door and front-door adjustment are popular methods to allow for estimation of the desired causal effects.

The Neyman-Rubin framework (Imbens and 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), which requires that treatment assignment is independent of potential outcomes given a set of

[3]We provide our pre-trained models, pre-training data generation code, and case study datasets at `https://github.com/jr2021/Do-PFN`.

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 $\mathbf{x}^{pt}$, the value of an intervention t^{in} and an observational dataset $\mathcal{D}^{ob}$.

for $i = 1, 2, \ldots, N$ **do**
  Draw $\psi_i \sim p(\psi)$; `// Draw an SCM`
  Initialize $\mathcal{D}_i^{ob} \leftarrow \emptyset$;
  Draw $M_{ob} \sim \text{Uniform}(\{M_{min}, M_{min}+1, \ldots, M_{max}\})$; `// Number of observational data points`
  for $j = 1, \ldots, M_{ob}$ **do**
    Sample noise $\epsilon_j \sim p(\epsilon)$;
    Draw $y_j^{ob}, t_j^{ob}, \mathbf{x}_j^{ob} \sim p(y^{ob}, t^{ob}, \mathbf{x}^{ob} | \psi_i, \epsilon_j)$; `// Draw observational data`
    $\mathcal{D}_i^{ob} \leftarrow \mathcal{D}_i^{ob} \cup \left\{(y_j^{ob}, t_j^{ob}, \mathbf{x}_j^{ob})\right\}$;
  end
  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 $\mathbf{x}_k^{pt} \sim p(\mathbf{x}^{pt} | \psi_i, \epsilon_k)$; `// Pre-treatment values of covariates`
    Draw $t_k^{in} \sim p(t^{in})$; `// Draw value for intervention`
    Draw $y_k^{in} \sim p(y^{in} | do(t_k^{in}), \psi_i, \epsilon_k)$; `// Sample interventional outcomes`
    $\mathcal{D}_i^{in} \leftarrow \mathcal{D}_i^{in} \cup \left\{(y_k^{in}, t_k^{in}, \mathbf{x}_k^{pt})\right\}$;
  end
  Compute $\mathcal{L}_i(\theta) = \sum_{k=1}^{M_{in}} -\log q_\theta(y_k^{in} | do(t_k^{in}), \mathbf{x}_k^{pt}, \mathcal{D}_i^{ob})$; `// Loss computation`
  $\theta \leftarrow \theta - \alpha \nabla \mathcal{L}_i(\theta)$; `// Gradient descent`
end

observed covariates. Machine-learning based methods conceptualized in this framework include causal trees (Athey and Imbens, 2016), causal forests (Wager and Athey, 2018), as well as T-, S- and X-learners (Künzel et al., 2019).

Prior-data fitted networks 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. The model that parameterizes this mapping can be obtained by simulating a large number of samples of the form $(\mathcal{D}_i, y_i, \mathbf{x}_i)$, followed by training the model to predict y_i while conditioning on $\mathbf{x}_i$ and $\mathcal{D}_i$. Neural processes (Garnelo et al., 2018a,b; Nguyen and 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). The PFN framework has been successfully applied to diverse problems, such as time-series prediction (Dooley et al., 2023; Hoo et al., 2024), Bayesian optimization (Müller et al., 2023; Rakotoarison et al., 2024), and causal fairness (Robertson et al., 2024).

Amortized causal inference Amortized inference beyond observational distributions was first explored for causal discovery (Ke et al., 2022; Lorch et al., 2022; Dhir et al., 2025). 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. Nilforoshan et al. (2023) consider the problem of zero-shot inference for CATEs using a meta-learning approach to facilitate generalization to unseen treatments. Concurrent to our work, Bynum et al. (2025) propose to use amortized inference to learn various causal effects; however, they only 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 do 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. We furthermore assume a prior $p(t^{in})$ over possible values for the treatment variable when performing an intervention $do(t^{in})$. This prior is only required to sample values for the intervention and does not affect how we model the CID (Equation 1) provided it has sufficient support. Samples from the distribution $p(y^{in}, \mathbf{x}^{pt} | \psi, do(t^{in}))$ over outcomes and covariates given this intervention then result from forward-propagating through the intervened-upon SCM. We sample the values of the covariates $\mathbf{x}^{pt}$ prior to performing the intervention $do(t^{in})$, such that $\mathbf{x}^{pt}$ contains only pre-treatment values (also of variables that are descendants of the treatment). We do this to train our model to learn the conditional interventional distribution given only pre-treatment values which is a more realistic setup in practice. Please refer to Algorithm 1 and Appendix B for more details on the data-generating process. The assumptions above imply the following form of the CID:

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

Note that in our framework $p(\psi | \mathbf{x}^{pt}) \neq p(\psi)$ since knowing the feature vector $\mathbf{x}^{pt}$ provides information on the SCM that generated it. 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.

Approximating the conditional interventional distribution Ultimately, we are interested in obtaining a model $q_\theta(y^{in} | do(t^{in}), \mathbf{x}^{pt}, \mathcal{D}^{ob})$[4] that is as close as possible to the CID $p(y^{in} | do(t^{in}), \mathbf{x}^{pt}, \psi)$ for all relevant treatment values t, SCMs ψ, and covariate-vectors $\mathbf{x}^{pt}$, while only taking observational data $\mathcal{D}^{ob}$ into account. The core idea of PFNs is to achieve this by *prior fitting*, i.e., minimizing the negative log-likelihood $-\log q_\theta(y^{in} | do(t^{in}), \mathbf{x}^{pt}, \mathcal{D}^{ob})$ on data from the synthetic data-generating process (Müller et al., 2022) via stochastic gradient descent (lines 19 and 20 in Algorithm 1). The following proposition shows that prior-fitting according to Algorithm 1 achieves the goal of yielding an optimal approximation of the CID from observational data:

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

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

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

The proof, given in Appendix A, 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 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 $q_\theta(y^{in} | do(t^{in}), \mathbf{x}^{pt}, \mathcal{D}^{ob})$ accurately captures the uncertainty in the outcome y arising from the unidentifiability of the causal effect of $do(t^{in})$ on y^{in}. Section 4.5 discusses empirical results indicating that Do-PFN is indeed able to do so.

Architecture and training details Do-PFN is a transformer with a similar architecture as TabPFN (Hollmann et al., 2025). In order to adapt this architecture for predicting CIDs, we 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,details in Appendix B. This takes 48 hours on a single RTX 2080 GPU.

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. The **key takeaway** of our results is that Do-PFN achieves performance on par with models that take the ground truth causal graph into account, and does significantly better than baselines that (like

[4]We use the do-notation in $q_\theta(y^{in} | do(t^{in}), \mathbf{x}^{pt}, \mathcal{D}^{ob})$ to indicate that our model approximates the distribution of the outcome y^{in} given an intervention on t^{in}. This is formally *not* the result of applying the do-calculus to an observational distribution $q_\theta(y^{in} | t^{in}, \mathbf{x}^{pt}, \mathcal{D}^{ob})$.

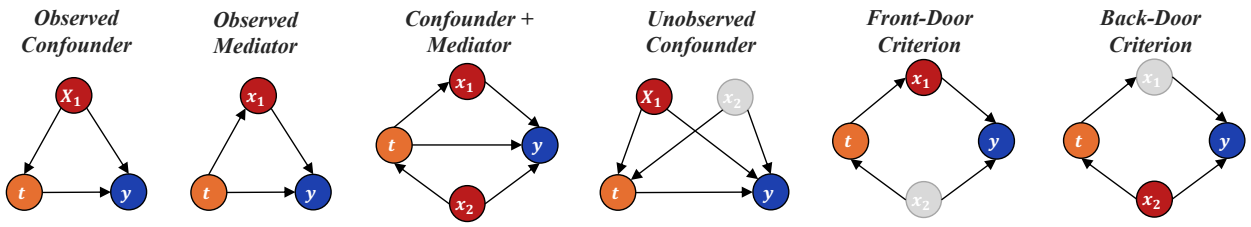

Figure 2: **Case studies**: Visualization of the graph structures of our six causal case studies, requiring Do-PFN to automatically perform adjustment based on the front-door and back-door criteria. Treatment variables t are visualized in orange, covariates $\mathbf{x}$ in red, and outcomes y in blue.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

Do-PFN) do not have access to this information, especially on the task of CATE estimation. In addition, Do-PFN accurately captures the uncertainty arising from the unidentifiability of causal effects while being slightly underconfident in identifiable cases. Furthermore, our ablations show that Do-PFN works well on small datasets, is robust to varying base rates of the average treatment effect, and performs consistently on large graph structures. Finally, we evaluate Do-PFN's performance on two real-world datasets with widely agreed-upon causal graphs, showing that its predictions mirror that of our gold-standard baseline.

4.1 Experimental setup

Case studies We introduce several causal case studies that pose unique challenges for causal effect estimation, traditionally approached via the front-door and back-door criteria (Figure 2). Moreover, the "Unobserved Confounder" case study features an unidentifiable causal effect, thereby constituting a fundamentally intractable problem for (precise) causal effect estimation. Please refer to Appendix C.1 for more details.

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. 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 16).

4.2 Predicting conditional interventional distributions (CIDs)

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 plots depicting normalized mean squared error (MSE) and 95% confidence intervals of regression baselines across our six causal case studies. For a description of normalized 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 thick 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[5] 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 offers two interesting findings.

First, the substantial difference in performance between Do-PFN and Dont-PFN provides empirical evidence that our pre-training (Algorithm 1) approximates something rather different from just a standard posterior predictive distribution of observational outcomes, which in turn allows Do-PFN to precisely estimate causal effects.

Second, Do-PFN effectively handles the fact that samples in the observational set $\mathcal{D}^{ob}$ and $\mathcal{D}^{in}$ come from different data-generating processes, namely the original SCM and the intervened-upon SCM. This causes a mismatch in the joint distributions $p(t^{ob}, y^{ob})$ and $p(t^{in}, y^{in})$, precisely when there is a directed edge between treatments t and covariates $\mathbf{x}$.

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

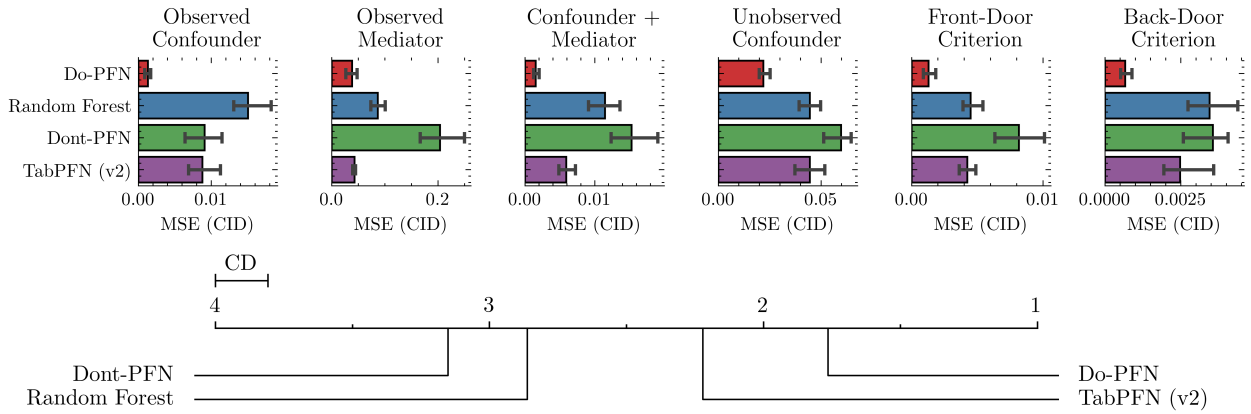

Figure 3: **Predicting conditional interventional distributions**: Bar-plots with 95% confidence intervals and critical difference (CD) diagrams depicting distributions of normalized mean squared error (MSE) of Do-PFN and our regression baselines in conditional interventional distribution (CID) prediction. Across our six causal case studies, Do-PFN achieves statistically significant improvements of MSE over regression baselines, showing that our pre-training objective is effective for predicting CIDs.

When the treatment and covariates are sampled independently, this mismatch disappears. This is empirically validated by the similar performance of Do-PFN and tabular regression models on the "Common Effect" case study (Appendix Figure 16). Furthermore, when the effect of the treatment increases, we observe that TabPFN (v2) deteriorates in performance across all case studies (Figure 7 left).

Interestingly, Do-PFN also outperforms the other methods in the unidentifiable "Unobserved Confounder" case study. While it is provably impossible to infer the precise causal effect in this case, data-generating processes can still leave causal traces that allow for narrowing down the set of possible causal effects Peters et al. (2017). We believe that Do-PFN utilizes this subtle type of causal information to output at least a plausible set of solutions—in contrast to traditional methods for which this case would be considered totally unsolvable Pearl (2009). We also observe the weakest performance of Do-PFN on the "Observed Mediator" case study (Figure 3, Appendix Figure 12), but later highlight that this results from overprediction in the CID setting that cancels out in CATE estimation.

Implicit graph identification When comparing Do-PFN to our "gold standard" baselines (Appendix Figure 12), we observe that Do-PFN performs competitively with DoWhy (Int.) and DoWhy (Cntf.), baselines which fit additive noise models (ANMs) and invertible SCMs respectively given both the observational data and the ground truth graph. DoWhy then uses the constructed SCM to predict interventional and counterfactual outcomes. We note that DoWhy baselines apply pre-trained TabPFN (v2) classification and regression models to represent endogenous structural equations, equipping the resulting causal model with significant representational capacity.

In our CD analysis (Appendix Figure 12), we observe that Do-PFN performs better on average than DoWhy (Int.), and closer to DoWhy (Cntf.) than any other baseline, despite not having access to the true graph structure. Further, when comparing different variants of Do-PFN in Appendix Figure 15, we observe that even Do-PFN-Short, our shortest pre-trained model, already performs similarly with Do-PFN-Graph, a version of Do-PFN which is pre-trained for each case study on datasets drawn exclusively to the corresponding graph structure.

4.3 Estimating conditional average treatment effects (CATEs)

We now evaluate Do-PFN's ability in CATE estimation, by calculating

$$\hat{\tau}(\mathbf{x}^{pt}) = \mathbb{E}_{y^{in} \sim q_\theta(y^{in} | do(1), \mathbf{x}^{pt}, \mathcal{D}^{ob})}[y^{in}] - \mathbb{E}_{y^{in} \sim q_\theta(y^{in} | do(0), \mathbf{x}^{pt}, \mathcal{D}^{ob})}[y^{in}]. \tag{4}$$

Comparison to causal machine learning baselines In estimating CATE values, we again observe that our largest model applied for CATE estimation, Do-PFN-CATE, statistically significantly outperforms state-of-the-art meta-learner (Künzel et al., 2019) and double machine learning (DML) approaches (Wager and Athey, 2018; Chernozhukov et al., 2018), (Figure 4). Note that the strong performance (second to Do-PFN-CATE) of TabPFN (v2) used as an S-Learner is in line with recent findings by Zhang et al. (2025).

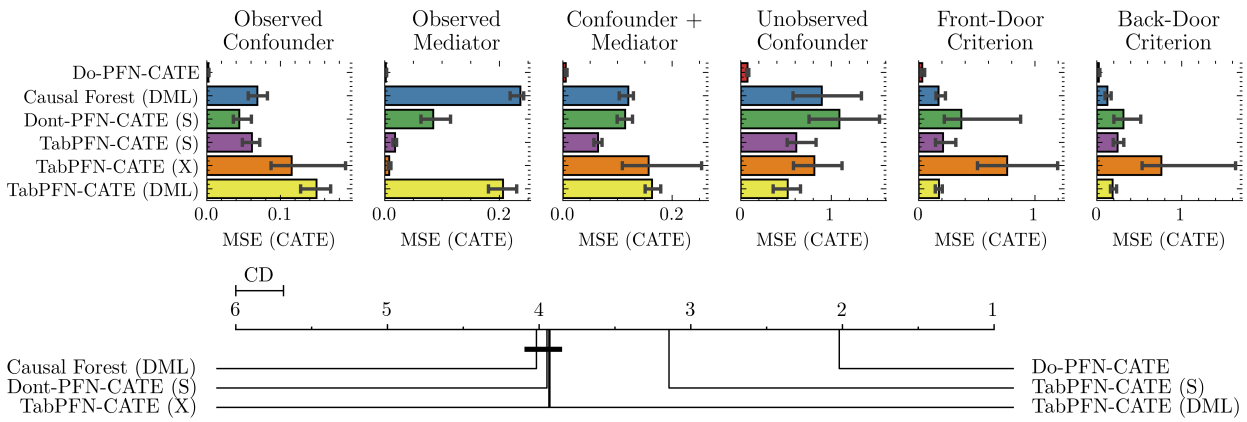

Figure 4: **Estimating conditional average treatment effects**: Bar-plots with 95% and critical difference (CD) diagrams depicting distributions of normalized mean squared error (MSE) of Do-PFN-CATE and our causal baselines in conditional average treatment effect (CATE) estimation. Across our six causal case studies, Do-PFN-CATE significantly outperforms common baselines for CATE estimation.

Comparison to gold-standard baselines We also observe that in the CATE estimation setting, Do-PFN-CATE performs in the same equivalence class as DoWhy-CATE (Cntf.) (Appendix Figure 13), outperforming the gold-standard baseline when the common "a priori" assumption of sufficiency is violated (Figure 6). To investigate this increased performance in CATE estimation, we highlight in Appendix D.4 how the relatively high bias and low variance in Do-PFN's predictions can lead to improved performance in CATE estimation compared to predicting CIDs.

4.4 Hybrid synthetic-real-world data

To assess whether Do-PFN's strong 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 (Appendix Figure 8). Those causal graphs allow us to simulate gold-standard outcomes using the DoWhy library (Sharma and 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.

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"). 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). (Figure 5 left). For CATE estimation, Do-PFN-CATE has a lower median MSE value than other CATE baselines, which have a relatively large variance in terms of their performance. (Figure 5 center right). Appendix Figure 9 visualizes the predictions of Do-PFN and our baselines vs. the gold-standard interventional outcomes for CID prediction and CATE estimation, showing that Do-PFN's predictions better align with the gold-standard targets.

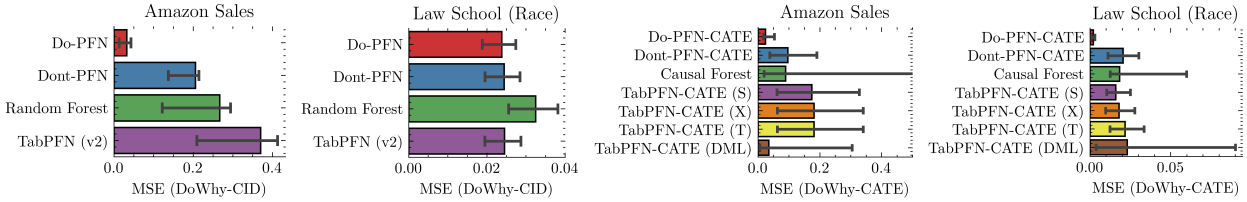

Figure 5: **Hybrid synthetic-real-world data**: Bar-plots with 95% confidence intervals depicting distributions of normalized mean squared error (MSE) of Do-PFN compared to causal and regression baselines in interventional outcome prediction (left) and conditional average treatment effect (CATE) estimation (right). Do-PFN's strong performance in synthetic settings extends to hybrid synthetic-real-world scenarios, especially in CATE estimation.

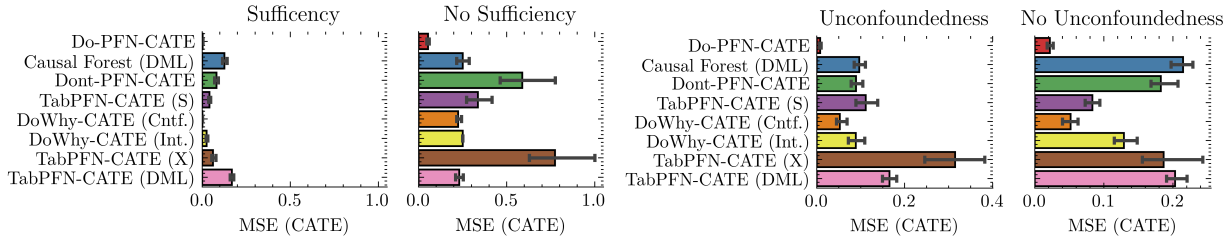

Figure 6: **Robustness to causal assumptions**: Bar plots and 95% confidence intervals depicting the median MSE in CATE estimation when sufficiency (no unobserved variables) and unconfoundedness (the treatment assignment is independent of the potential outcomes given the covariates) are fulfilled or violated. We find that while the performance of all methods degrades when sufficiency and unconfoundedness are violated, Do-PFN maintains the strongest performance in both settings.

Law School Admissions The Law School Admissions dataset (Figure 8) 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 while we do not address the topic of algorithmic bias, we simulate the effect on first-year-average ("FYA") from performing interventions on "Race"[6], a common evaluation strategy in causal fairness (Robertson et al., 2024; Kusner et al., 2017). In terms of CID prediction (Figure 5 center left) and CATE estimation (Figure 5 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 common CATE estimation baselines.

4.5 Ablation studies

Finally, we perform several ablation studies to evaluate Do-PFN's behavior across datasets of different sizes, as well as by varying several aspects of the data-generating SCMs themselves.

Robustness to causal assumptions When comparing the performance of Do-PFN against the set of baselines in CATE estimation when the causal assumptions of sufficiency (no unobserved variables) and unconfoundedness (the treatment is independent of the potential outcomes given the covariates) are fulfilled and violated, we find that all methods, including Do-PFN, benefit from the satisfaction of sufficiency and unconfoundedness (Figure 6). While those assumptions make the problems substantially easier from a causal perspective, Do-PFN maintains its superior performance for the cases where sufficiency or unconfoundedness are violated.

Dataset size and complexity First, we observe in Figure 7 (center left) 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 (Figure 7 right). We further analyze Do-PFN's performance across graph complexities in Appendix D.1. Furthermore, we find that Do-PFNs can effectively use additional data points to alleviate increasing levels of noise (Appendix D.2).

Treatment effect We also show in Figure 7 (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 does not maintain the inductive bias that the treatment *must* influence the outcome and is robust to different magnitudes of the true ATE. This 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 (Appendix D.3), where we find that Do-PFN is slightly under-confident for theoretically identifiable case studies. In the "Unobserved Confounder" case study, the model's high uncertainty is reflected by a relatively large entropy in its output distribution (Appendix 17). However, our PICP results show that the model's uncertainty for this case study is correctly calibrated. Overall, our results indicate that Do-PFN is able to correctly capture the variability in causal effects, even when the variability arises due to unidentifiability.

[6]We note that Race in the lawschool dataset is typically treated as a binary variable. We very much disagree with this formulation, and acknowledge that the term "ethnicity" better describes this complex social construct.

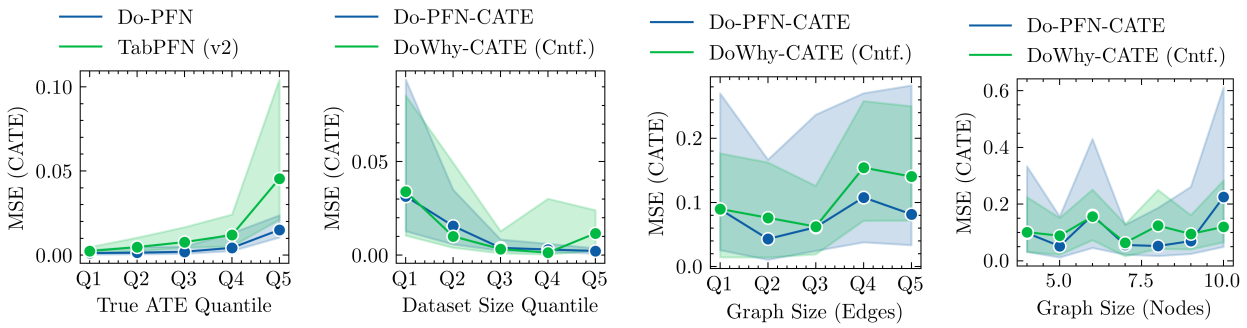

Figure 7: **Ablation studies**: Do-PFN is relatively insensitive to changing base rates of ATEs (left), improves with increased number of observational samples (center-left), and remains consistent on large graph structures (right).

5 Discussion

We introduced Do-PFN, a pre-trained transformer leveraging ICL to meta-learn to predict interventional outcomes from observational data. Our empirical results on carefully controlled synthetic and hybrid synthetic-real-world 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 provided the true underlying causal graph. Furthermore, Do-PFN performs strongly on small datasets, shows invariance to graph complexity and base treatment effects, and correctly accounts for uncertainty arising from unidentifiability. Nevertheless, we need to discuss a range of limitations and future work.

Real-world benchmarking First, Do-PFN's generalization capability critically depends on its synthetic prior over SCMs $p(\psi)$, adequately capturing real-world causal complexity. As our current validation is mainly based on synthetic data, Do-PFN's robustness to prior-reality mismatches and its performance on diverse real-world datasets require further systematic exploration, alongside principled methods for prior design and validation, as well as comprehensive benchmarks for causal effect estimation. What makes us optimistic are our initial hybrid synthetic-real-world results and empirical findings from the community providing strong evidence that synthetic prior fitting can lead to strong real-world performance (Hollmann et al., 2025; Hoo et al., 2024; Reuter et al., 2025).

Identifiability theory and statistical guarantees While our experiments suggest Do-PFN can reflect uncertainty from its training prior (including from SCMs with non-identifiable effects), a full theoretical characterization of how this learned predictive uncertainty aligns with formal causal identifiability bounds remains a nontrivial area for future investigation. Do-PFN's amortized inference approach offers efficiency in the face of small data, but has different theoretical underpinnings regarding statistical guarantees compared to traditional, non-amortized estimators under known causal structures Nagler (2023). However, the statistical theory for such amortized models is still developing.

Trust and interpretability Our theoretical results in Proposition 1 combined with Do-PFN's strong empirical performance on diverse causal setting, imply that Do-PFN is a highly effective approach for performing causal inference. It is therefore possible that Do-PFN follows the traditional approach to causal reasoning by, first, discovering the causal structure of a problem which allows, in a second step, to carry out the appropriate adjustment. Future work in mechanistic interpretability of Do-PFN would help verify its inner workings as well as provide an improved sense of transparency and trust.

Extensions to further causal tasks Many configurations and types of causal challenges were necessarily not addressed in this initial work. These include, for instance, a broader array of intervention types beyond binary interventions and counterfactuals, non-i.i.d. input data, and various observational data characteristics or modalities not explicitly part of our current experimental scope. Incorporating them into our data-generating prior can give us a completely new handle on some of these hard problems.

To conclude, the key contribution of Do-PFN is a novel methodology for causal effect estimation. We are optimistic about its prospects to become part of the standard ML toolkit, thus helping to give causal effect estimation the broad accessibility that its real-world relevance deserves.

References

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

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

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

Breiman, L. (2001). Random forests. *Machine Learning*, 45:5–32.

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

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

Dhir, A., Ashman, M., Requeima, J., and van der Wilk, M. (2025). A meta-learning approach to Bayesian causal discovery. In *The Thirteenth International Conference on Learning Representations*.

Dooley, S., Khurana, G. S., Mohapatra, C., Naidu, S. V., and White, C. (2023). Forecastpfn: Synthetically-trained zero-shot forecasting. *Advances in Neural Information Processing Systems*, 36:2403–2426.

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

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

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

Guo, S., Tóth, V., Schölkopf, B., and Huszár, F. (2023). Causal de finetti: On the identification of invariant causal structure in exchangeable data. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Guo, S., Zhang, C., Mohan, K., Huszár, F., and Schölkopf, B. (2024). Do finetti: On causal effects for exchangeable data. *Advances in Neural Information Processing Systems*, 37:127317–127345.

Herbold, S. (2020). Autorank: A python package for automated ranking of classifiers. *Journal of Open Source Software*, 5(48):2173.

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

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

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

Hoo, S. B., Müller, S., Salinas, D., and Hutter, F. (2024). 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*.

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

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

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

Ke, N. R., Chiappa, S., Wang, J., Goyal, A., Bornschein, J., Rey, M., Weber, T., Botvinic, M., Mozer, M., and Rezende, D. J. (2022). Learning to induce causal structure. *arXiv preprint arXiv:2204.04875*.

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

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

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

Kusner, M. J., Loftus, J., Russell, C., and Silva, R. (2017). Counterfactual fairness. *Conference on Neural Information Processing Systems*, 30:4069 – 4079.

Lorch, L., Sussex, S., Rothfuss, J., Krause, A., and Schölkopf, B. (2022). Amortized inference for causal structure learning. *Advances in Neural Information Processing Systems*, 35:13104–13118.

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

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

Müller, S., Feurer, M., Hollmann, N., and Hutter, F. (2023). Pfns4bo: In-context learning for Bayesian optimization. In *International Conference on Machine Learning*, pages 25444–25470. PMLR.

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

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

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

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

Nilforoshan, H., Moor, M., Roohani, Y., Chen, Y., Šurina, A., Yasunaga, M., Oblak, S., and Leskovec, J. (2023). Zero-shot causal learning. *Advances in Neural Information Processing Systems*, 36:6862–6901.

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

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

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

Rakotoarison, H., Adriaensen, S., Mallik, N., Garibov, S., Bergman, E., and Hutter, F. (2024). In-context freeze-thaw Bayesian optimization for hyperparameter optimization. In *Proceedings of the 41st International Conference on Machine Learning*, pages 41982–42008.

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

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

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

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

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

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

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

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

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

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

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

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

Waskom, M. L. (2021). seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021.

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

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

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

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

Zhang, Q., Tan, Y. S., Tian, Q., and Li, P. (2025). Tabpfn: One model to rule them all? *arXiv preprint arXiv:2505.20003*.

A Proof of Proposition 1

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

$$\mathcal{R}_\theta = \int\int\int\int -\log(q_\theta(y^{in}|do(t^{in}), \mathbf{x}^{pt}, \mathcal{D}^{ob}))p(\mathcal{D}^{ob}, t^{in}, y^{in}, \mathbf{x}^{pt})d\mathcal{D}^{ob}dt^{in}dy^{in}d\mathbf{x}^{pt} \quad (5)$$

Let's consider $p(\mathcal{D}^{ob}, t^{in}, y^{in}, \mathbf{x}^{pt})$. 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:

$$p(\mathcal{D}^{ob}, t^{in}, y^{in}, \mathbf{x}^{pt}) = \int p(\mathcal{D}^{ob}, t^{in}, y^{in}, \mathbf{x}^{pt}, \psi)d\psi = \int p(y^{in}, \mathbf{x}^{pt}|do(t^{in}), \psi)p(t^{in}|\mathcal{D}^{ob})p(\mathcal{D}^{ob}|\psi)p(\psi)d\psi \quad (6)$$

Now, we can use that

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

Further:

$$p(\mathbf{x}^{pt}|do(t^{in}), \psi)p(t^{in}|\mathcal{D}^{ob})p(\mathcal{D}^{ob}|\psi)p(\psi) = p(\mathcal{D}^{ob}, t^{in}, \mathbf{x}^{pt}, \psi). \quad (7)$$

This implies

$$p(\mathcal{D}^{ob}, t^{in}, y^{in}, \mathbf{x}^{pt}) = \int p(y^{in}|\mathbf{x}^{pt}, do(t^{in}), \psi)p(\mathcal{D}^{ob}, t^{in}, \mathbf{x}^{pt}, \psi)d\psi \quad (8)$$

Plugging this into equation 5 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 and Agakov (2004) in analogous scenarios, yields:

$$\begin{aligned}\mathcal{R}_\theta = \int\int\int\int\int & -\log(q_\theta(y^{in}|do(t^{in}), \mathbf{x}^{pt}, \mathcal{D}^{ob})) \\ & p(y^{in}|\mathbf{x}^{pt}, do(t^{in}), \psi)p(\mathcal{D}^{ob}, t^{in}, \mathbf{x}^{pt}, \psi)d\mathcal{D}^{ob}dt^{in}dy^{in}d\mathbf{x}^{pt}d\psi \\ = & \int\int\int\int \mathbb{D}_{KL}\left[p(y^{in}|\mathbf{x}^{pt}, do(t^{in}), \psi)||q_\theta(y^{in}|do(t^{in}), \mathbf{x}^{pt}, \mathcal{D}^{ob})\right] \\ & p(\mathcal{D}^{ob}, t^{in}, \mathbf{x}^{pt}, \psi)d\mathcal{D}^{ob}dt^{in}d\mathbf{x}^{pt}d\psi + C \end{aligned} \quad (9)$$

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 $p(\psi, \mathcal{D}^{ob}, t^{in}, \mathbf{x}^{pt})$.

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.

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 $\mathbf{x}_k^{pt}$ is sampled from $p(\mathbf{x}|\psi_i, \epsilon_k)$. This ensures that the vector $\mathbf{x}_k^{pt}$ 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[7].

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 $\mathbf{x}_k^{pt}$, 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 and Ba, 2014).

C Experimental Details

C.1 Details on the synthetic case studies

In this section we provide the details on all considered case studies from Section 4.1. 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.

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

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

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 normlized MSE below:

$$\text{MSE}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{n} \sum_{i=1}^{n} \left[\frac{y_i - \hat{y}_i}{\max(\mathbf{y}) - \min(\mathbf{y})} \right]^2 \tag{10}$$

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 and 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:

$$\hat{\tau} = \mathbb{E}_{y^{in} \sim q_\theta(y^{in}|do(t^{in}=1), \mathbf{x}^{pt}, \mathcal{D}^{ob})}[y^{in}] - \mathbb{E}_{y^{in} \sim q_\theta(y^{in}|do(t^{in}=0), \mathbf{x}^{pt}, \mathcal{D}^{ob})}[y^{in}] \tag{11}$$

- **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.3.3 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.

D Supplementary Results

D.1 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 7 (right). 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

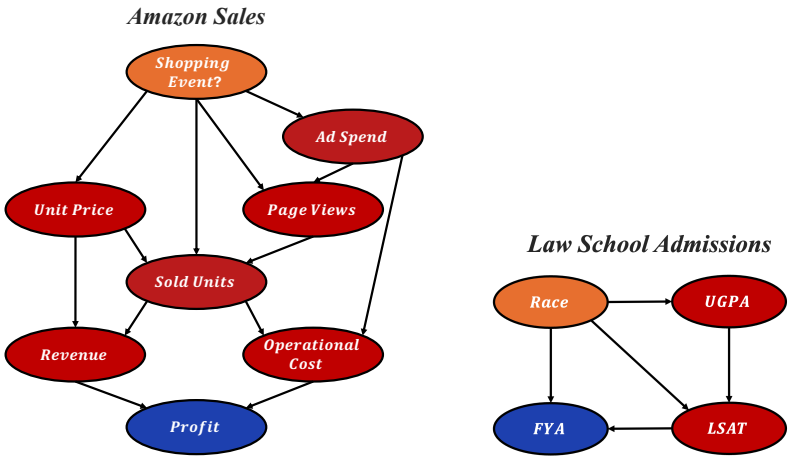

Figure 8: **Real-world case studies**: Widely agreed-upon causal graphs for our two real-world case-studies: Amazon Sales and Law School Admissions.

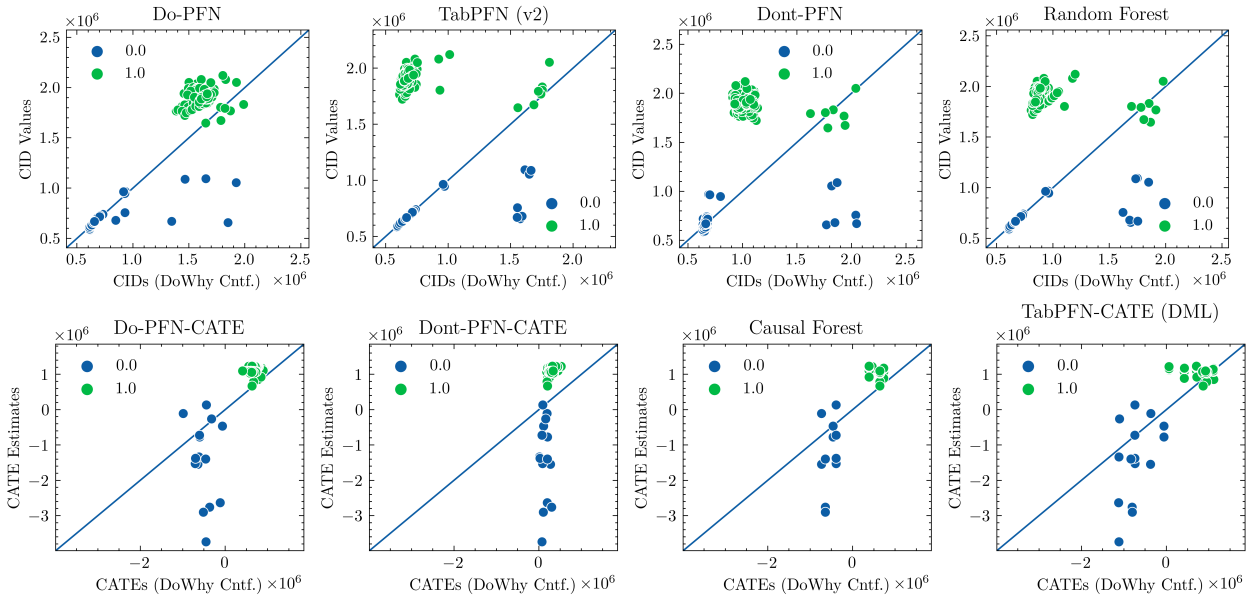

Figure 9: **Amazon Sales**: Scatter plots depicting the match between baseline predictions with gold standard outcomes produced by Do-Why-(CATE) (Int./Cntf.). Green scatter points represent individuals for which the intervention $do(ShoppingEvent = 1)$ is applied, while blue points represent $do(ShoppingEvent = 0)$.

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.

D.2 Robustness to additive noise

We also highlight in Figure 11 (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 11 (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 extent with more data.

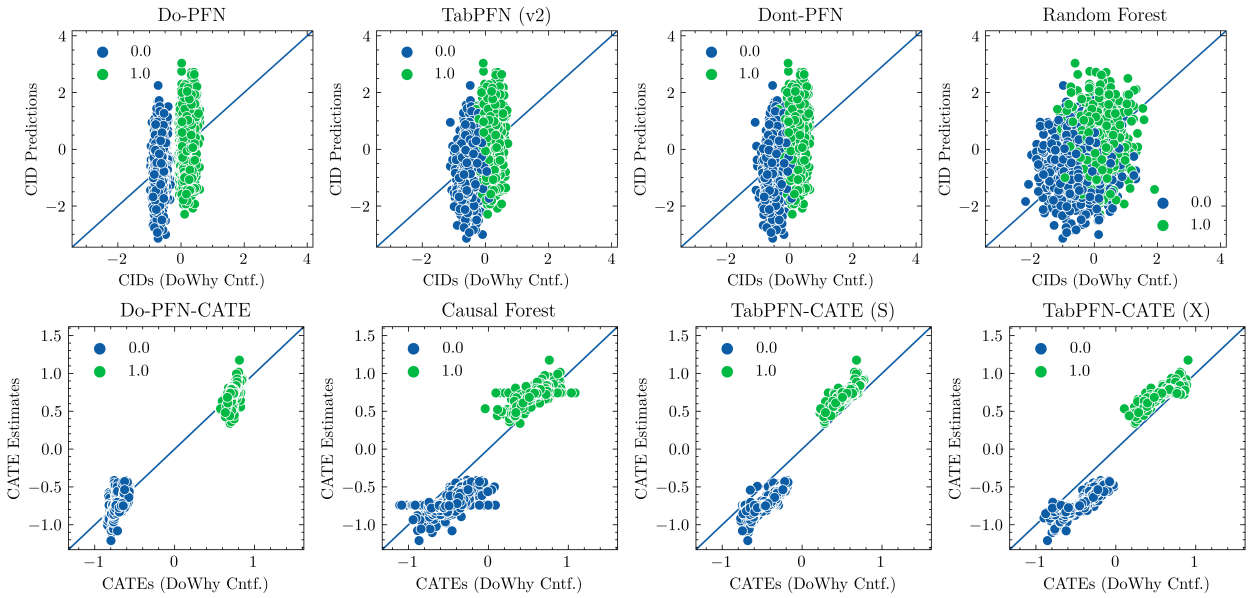

Figure 10: **Law School Admissions**: Scatter plots depicting the match between baseline predictions with gold standard outcomes produced by Do-Why-(CATE) (Int./Cntf.). Green scatter points represent individuals for which the intervention $do(Race = 1)$ is applied, while blue points represent the intervention $do(Race = 0)$.

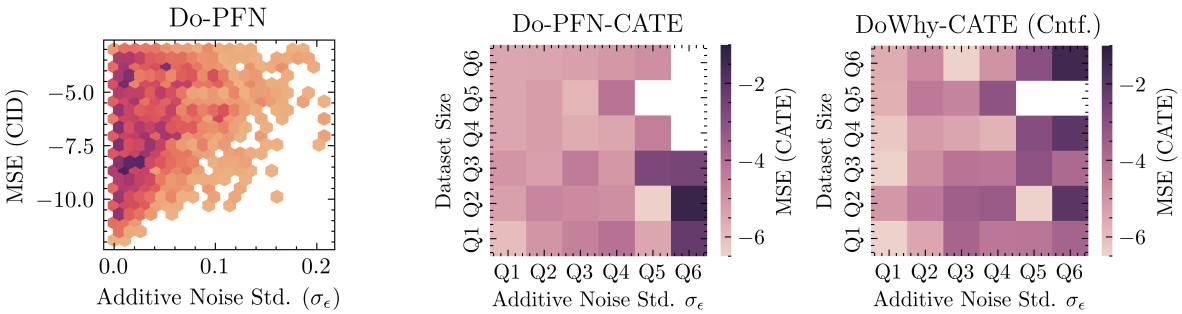

Figure 11: **Robustness to additive noise**: Evaluation of Do-PFN's performance in CID prediction and CATE estimation across different quantiles (Q1-Q6) of additive noise standard deviation. The density plot (left) shows that Do-PFN's performance decreases with (irreducible) additive noise. However, 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.).

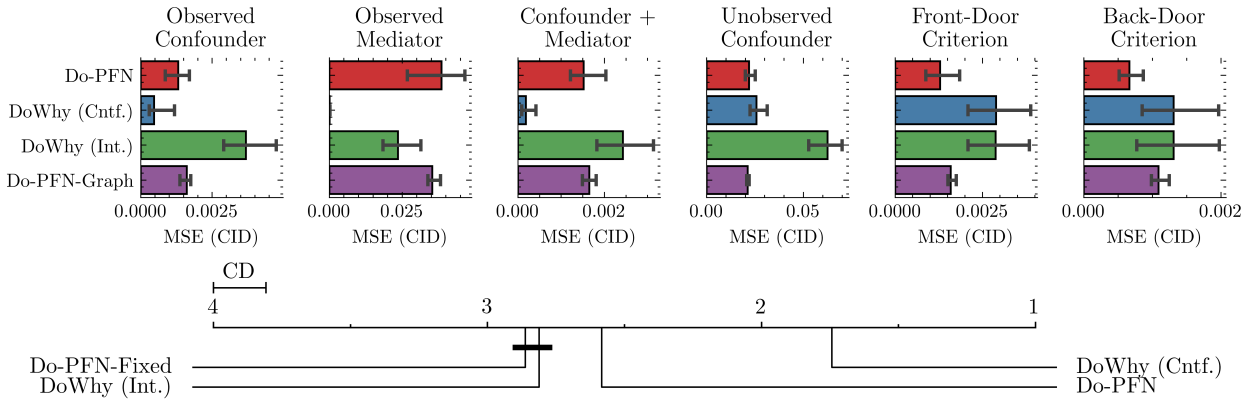

Figure 12: **Gold-standard comparison (CID)**: Bar-plots with 95% confidence intervals and critical difference (CD) diagrams depicting distributions of normalized mean squared error (MSE) of Do-PFN and our "gold-standard" baselines in conditional interventional distribution (CID) estimation on our six synthetic case studies. Do-PFN significantly outperforms Do-PFN-Graph and DoWhy (Int.), while performing closer to DoWhy (Cntf.) than other baselines.

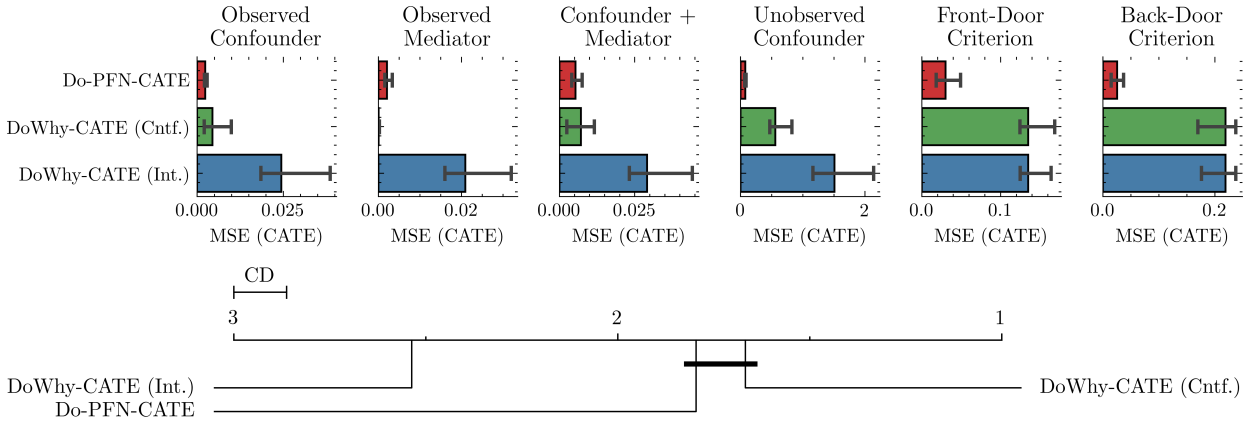

Figure 13: **Gold-standard comparison (CATE)**: Bar-plots with 95% confidence intervals and critical difference (CD) diagrams depicting distributions of normalized mean squared error (MSE) of variants of Do-PFN and our baselines in conditional average treatment effect (CATE) estimation on our six synthetic case studies. Do-PFN-CATE outperforms DoWhy-CATE (Int.) and performs competitively with DoWhy-CATE (Cntf.).

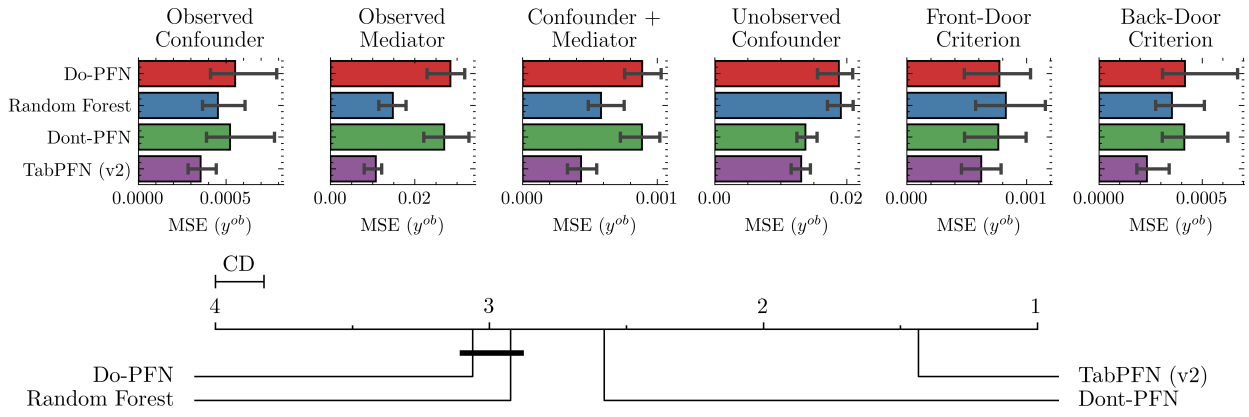

Figure 14: **Comparison on regression problems**: Bar-plots with 95% confidence intervals and critical difference (CD) diagrams depicting distributions of normalized mean squared error (MSE) of variants of Do-PFN and our baselines when predicting observational outcomes (no interventions) in our six causal case studies.

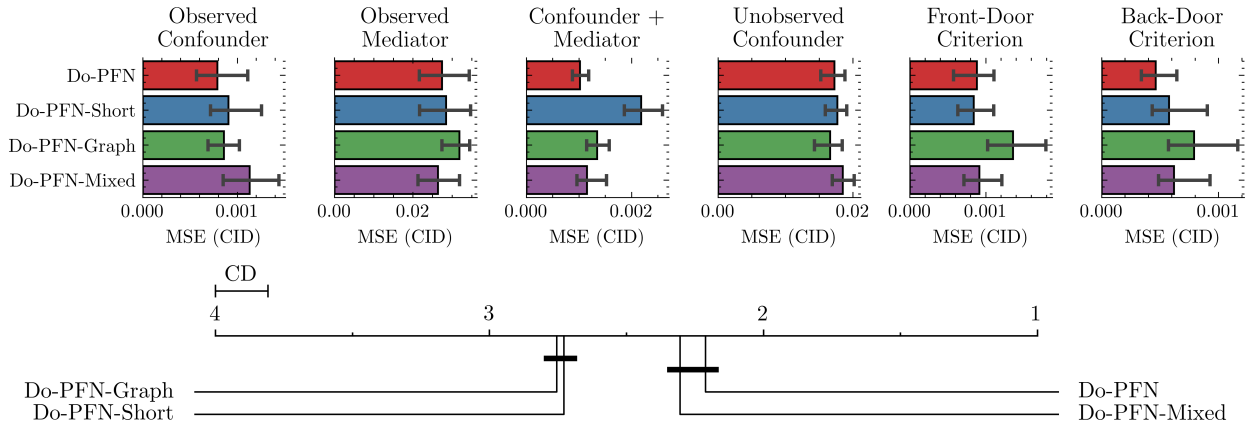

Figure 15: **Comparison of Do-PFN variants (CID)**: Bar-plots with 95% confidence intervals and critical difference (CD) diagrams depicting distributions of normalized mean squared error (MSE) of Do-PFN variants in conditional interventional distribution (CID) estimation on our six synthetic case studies. Do-PFN significantly outperforms other variants except Do-PFN-Mixed, which achieves statistically similar performance in half the pre-training time.

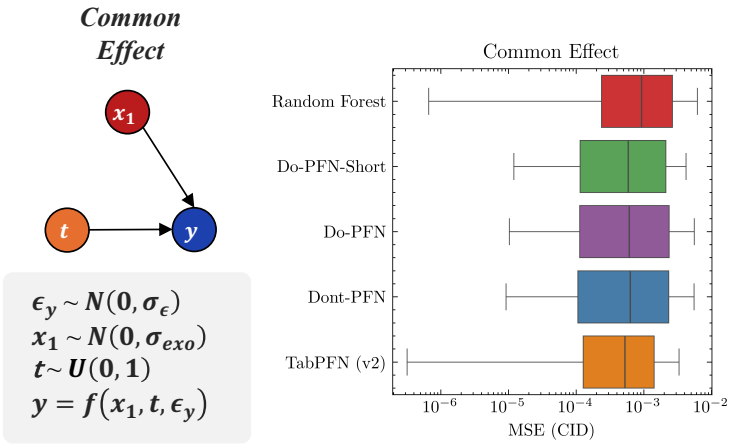

Figure 16: **Common effect case study**: Visualization of graph structure and structural equations (left) for our "common effect" case study, as well as box plots depicting distributions of normalized mean squared error (MSE) of Do-PFN variants compared to regression baselines in conditional interventional distribution prediction. Regression baselines perform similarly to Do-PFN variants, as the intervention does not cause a distribution shift between $\mathcal{D}^{ob}$ and $\mathcal{D}^{in}$.

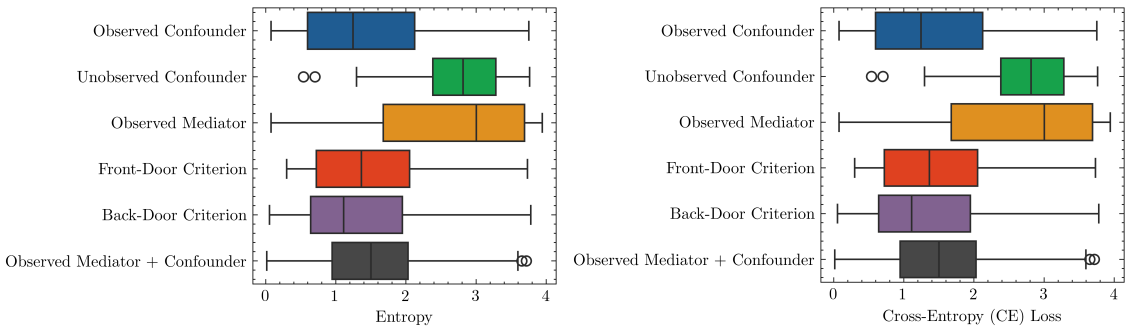

Figure 17: **Uncertainty quantification**: Cross-entropy (CE) loss (right) and entropy (left) of Do-PFN's bar distribution output. Do-PFN is highly uncertain on the "Unobserved Confounder" case study due to unidentifiability. Do-PFN also shows high uncertainty on the "Observed Mediator" case study, which 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.3 Uncertainty calibration

We investigate the calibration of Do-PFN by visualizing the prediction interval coverage probability (PICP) in Figure 7. 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.

In summary, we find that Do-PFN is slightly underconfident for identifiable case-studies, but correctly specifies the uncertainty arising from unidentifiable causal effects.

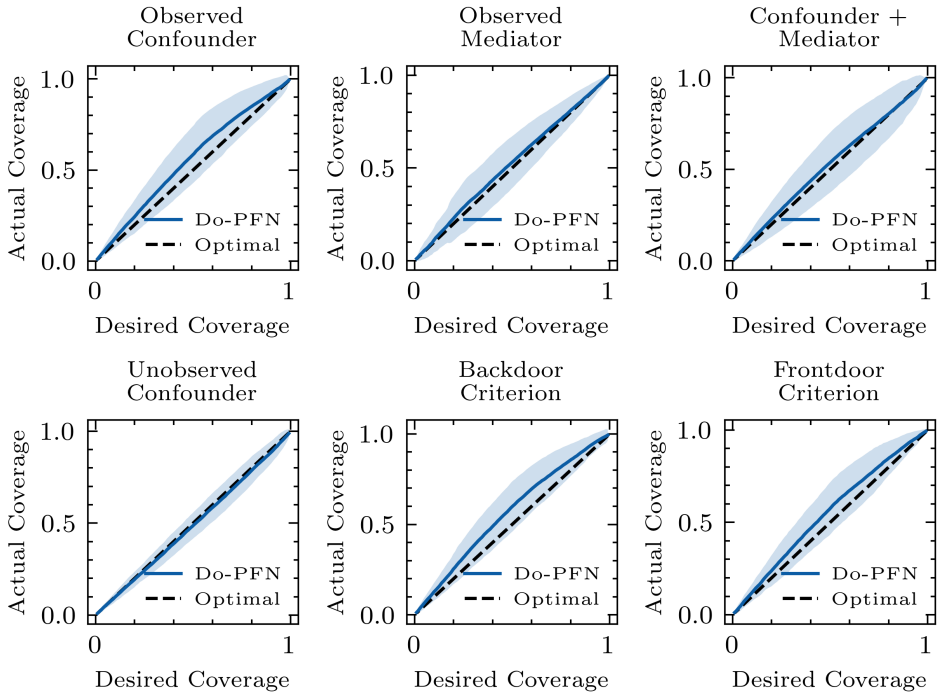

Figure 18: **Uncertainty calibration**: Prediction interval coverage probability (PICP) plots for the "Observed Mediator", "Confounder + Mediator", "Backdoor Criterion" and "Frontdoor Criterion" cases. The solid blue line shows the coverage and standard deviation achieved by Do-PFN, spanning desired 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.

D.4 Bias decomposition

In order to investigate Do-PFN's comparatively strong performance in CATE estimation as opposed to CID prediction where it performed better than DoWhy (Int.) but worse than DoWhy (Cntf.), we decompose the bias of Do-PFN and DoWhy (Cntf.) under the two interventions $do(0)$ and $do(1)$, calculating the median of the average residual errors across all datasets. We observe in Figure 19 that while DoWhy (Cntf.) has low bias for all case studies except "Unobserved Confounder", Do-PFN's bias is larger, however approximately equal between interventions $do(0)$ and $do(1)$. This hurts Do-PFN's performance in CID prediction as it systematically slightly over-predicts. However, in CATE estimation, the bias terms cancel out, resulting in a better CATE estimate relative to predicting interventional outcomes.

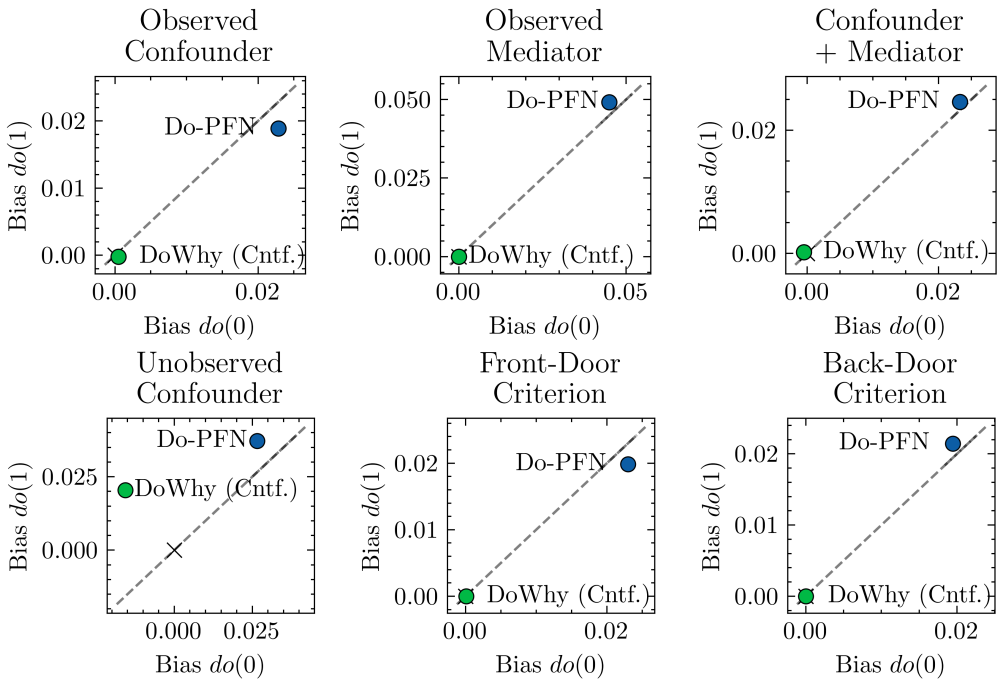

Figure 19: **Bias Decomposition**: The median of the bias of DoWhy (Cntf.) and Do-PFN across 100 synthetic datasets for the interventions $do(0)$ and $do(1)$. The gold-standard DoWhy (Cntf.) maintains a bias very close to zero for all case-studies while Do-PFN has a small positive bias that takes almost the same value for $do(0)$ and $do(1)$.