
CausalProfiler: Generating Synthetic Benchmarks for Rigorous and Transparent Evaluation of Causal ML

Anonymous Author(s)

Affiliation

Address

email

Abstract

1 Causal machine learning (Causal ML) aims to answer “what if” questions using
2 machine learning algorithms, making it a promising tool for high-stakes decision-
3 making. Yet, empirical evaluation practices in Causal ML remain limited. Existing
4 benchmarks often rely on a handful of hand-crafted or semi-synthetic datasets,
5 leading to brittle, non-generalizable conclusions. To bridge this gap, we introduce
6 CausalProfiler, a synthetic benchmark generator for Causal ML methods. Based on
7 a set of explicit design choices about the class of causal models, queries, and data
8 considered, the CausalProfiler randomly samples sets of data, assumptions, and
9 ground truths constituting the synthetic causal benchmarks. In this way, Causal ML
10 methods can be rigorously and transparently evaluated under a variety of conditions.
11 This work offers the first random generator of synthetic causal benchmarks with
12 coverage guarantees and transparent assumptions operating on the three levels of
13 causal reasoning—observation, intervention, and counterfactual. We demonstrate
14 its utility by evaluating several state-of-the-art methods under diverse conditions
15 and assumptions, both in and out of the identification regime, illustrating the types
16 of analyses and insights the CausalProfiler enables.

17 1 Introduction

18 Causal machine learning (Causal ML) seeks to estimate the effects of interventions and counterfactuals
19 using machine learning techniques [28], enabling principled decision making in medicine, policy,
20 and other high-stakes domains. Despite its theoretical maturity and growing relevance, Causal ML
21 remains underutilized. A key barrier to adoption lies in the current empirical evaluation landscape,
22 which is unable to support meaningful and generalizable evidence of method performance [40].

23 Recent critiques of evaluation practices in both predictive Machine Learning (ML) [21, 13, 30] and
24 causal inference [10, 14, 6] have highlighted systemic shortcomings. Lessons from predictive ML
25 show that narrow, static benchmarks can give a false sense of reliability [16, 21], underscoring the
26 need for structured diversity: systematic variation of tasks under explicit, controllable assumptions.
27 In the case of causal ML, evaluation is fundamentally more challenging due to the unobservability of
28 counterfactual outcomes [23]. Hence, researchers can only rely on scarce real-world data sources.
29 Typically, randomized controlled trials, considered as the gold standard, are expensive, ethically
30 constrained, and often encompass a low amount of data [18, 44]. As a result, existing benchmarks
31 often rely on a small number of semi-synthetic datasets (e.g., IHDP [22], Twins [32]) or model-driven
32 synthetic datasets generated from fitted causal mechanisms [33, 34, 3, 12]. These datasets typically
33 encode specific assumptions—such as structural constraints, identifiability conditions, or narrow
34 function classes—which are rarely made explicit and are difficult to generalize beyond the original
35 study context. Moreover, handcrafted synthetic datasets, where researchers explicitly define causal
36 models and choose evaluation queries, are frequently designed with specific hypotheses or methods

in mind [17], risking bias through overfitting or implicit alignment. It has been argued [40] that the problem is not synthetic evaluation itself but the lack of rigor in its design and interpretation.

In this work, we take a concrete step toward addressing these concerns by introducing a synthetic benchmark generator, called CausalProfiler, that enables empirical evaluations grounded in transparently defined synthetic causal datasets. Central to our approach is the notion of a *Space of Interest* (*SoI*) (Definition 5.1), defining the domain from which causal datasets are sampled. Given a specified *SoI*, our benchmark generator samples SCMs, data, and queries, and estimates the ground truth value of the queries to enable the evaluation of Causal ML methods.

This approach departs from existing benchmarks in several key ways. Rather than evaluating methods on a fixed or narrow set of datasets, our framework enables controlled, repeatable, and diverse sampling over structured families of tasks. It replaces opaque or implicit design choices with fully transparent specifications of model structure, noise, and query types. Crucially, it shifts the focus of empirical evaluation from performance on individual datasets to trends and patterns across a well-characterized *SoI*, reframing the evaluation question from “what dataset to use” to specifying a *SoI* that defines the scope of evaluation. This enables researchers to evaluate not only performance but also under which conditions—on graph density, or causal mechanisms complexity for instance—a method succeeds or fails and helps practitioners identify methods that remain reliable when causal assumptions are likely violated. By aggregating results over many generated datasets, the evaluation yields more robust and reliable performance estimates, helping to uncover failure modes, generalization limits, and assumption sensitivities that remain hidden in conventional evaluations.

Contributions. This paper offers the first open-source implementation of such an evaluation framework¹, and illustrates its utility by comparing state-of-the-art causal ML methods across multiple synthetic causal datasets. Our contributions are twofold. First, we present a benchmark generator (Section 5) that enables principled sampling of synthetic causal datasets over user-defined *Sols*, with built-in coverage guarantees that promote transparency and reproducibility. Secondly, we demonstrate through experiments (Section 6) how evaluation across different *Sols* yields richer and more robust insights than single-dataset evaluations.

2 Related Work

Evaluating causal ML methods. Causal ML currently lacks a rigorous, systematic paradigm for empirical evaluation. Indeed, the community has largely turned to synthetic and semi-synthetic benchmarks. Semi-synthetic datasets, such as IHDP [22] and Twins [32], combine real covariates with simulated outcomes under assumed structural models. Fully synthetic datasets, in contrast, are generated entirely from researcher-defined SCMs, allowing for greater control and access to ground truth. Yet both synthetic and semi-synthetic approaches suffer from critical limitations.

First, synthetic evaluations often lack realism, relying on overly simplistic mechanisms such as additive noise or linear functions, and frequently omitting robustness analyses [17, 10, 39, 40]. These experiments rarely reflect the complexity of real-world causal processes and are insufficient to test the limits of modern causal inference methods.

Secondly, synthetic and semi-synthetic datasets are shaped by researcher-defined design decisions, including the structure of the causal graph, the form of the outcome function, and the noise distribution. These decisions, often made implicitly, can unintentionally introduce hidden biases that favor certain methods [9, 8, 14]. Such assumptions are rarely documented or systematically varied, hindering reproducibility and fair method comparison [39, 40].

Additionally, these benchmarks are typically small in scale and narrow in scope, often covering only a limited range of causal settings. As a result, empirical evaluations raise concerns about overfitting and generalization [17, 6]. For instance, it has been shown that even small changes to the data-generating process can lead to dramatic shifts in performance rankings [9]. Moreover, methods are often evaluated only under the very conditions that guarantee their identifiability, offering little insight into robustness under assumption violations, as is common in real-world settings [40, 38, 26]. In short, without broader and more transparent evaluation across diverse causal settings, the field risks drawing conclusions that do not generalize. Addressing this gap requires moving beyond fixed benchmarks toward frameworks that support transparent, structured, and diverse experimentation across well-defined spaces of causal assumptions.

¹The code is provided in the supplementary material and will be publicly available after the review process.

Recent benchmarking efforts. Recent works have sought to address some of these gaps introducing tools to generate synthetic SCMs for causal discover [29, 19, 41] or support query estimation from hand-specified models [42, 45, 1]. However, none of these frameworks support all components required for robust evaluation of causal machine learning methods. First, causal discovery benchmarks [29, 19, 41] do not compute ground truth for intervention or counterfactual queries. Further, query estimation frameworks [42, 45, 1] often require manual specification of the SCM and do not support random sampling, diversity control, or analysis of the distribution of tasks. Even in cases where SCMs are sampled [41, 48], key properties (e.g., positivity) are neither reported nor constrained. In addition, the absence of randomness in the graph structures limits generalization. Our approach complements and extends these efforts by integrating SCM sampling, query ground truth computation, and coverage guarantees into a unified framework. To the best of our knowledge, this is the first benchmark generator that enables systematic exploration of how Causal ML methods behave across spaces of SCMs and queries defined by user-specified constraints.

3 Background & Notation

We use capital letters for random variables (e.g., X), lowercase for realizations (e.g., x), and boldface for vectors (e.g., $\mathbf{x}$). For a more complete background, please refer to Appendix A and Pearl [35].

Causal Hierarchy. The Pearl Causal Hierarchy (PCH) [36] classifies causal questions into three levels: $\mathcal{L}_1$ (associational), $\mathcal{L}_2$ (interventional), and $\mathcal{L}_3$ (counterfactual). While associative questions rely only on observed data, interventional and counterfactual questions require assumptions about the data-generating process. Importantly, lower layers are insufficient to answer higher-layer questions in almost all causal models [4].

Structural Causal Models. A Structural Causal Model (SCM) [35] is a tuple $\mathcal{M} := \{\mathbf{V}, \mathbf{U}, \mathcal{F}, P(\mathbf{U})\}$, where $\mathbf{V}$ are endogenous variables, $\mathbf{U}$ are exogenous variables, $\mathcal{F}$ is a set of structural equations $V_i = f_i(\mathbf{PA}(V_i), \mathbf{U}_{V_i})$, and $P(\mathbf{U})$ defines a distribution over the exogenous variables. SCMs induce a distribution $P_{\mathcal{M}}(\mathbf{V})$ over the endogenous variables. In this work, we additionally consider two types of endogenous variables: the observed variables, denoted $\mathbf{V}_O$, and the unobserved variables, denoted $\mathbf{V}_H$ with $\mathbf{V} = \mathbf{V}_O \cup \mathbf{V}_H$ and $\mathbf{V}_O \cap \mathbf{V}_H = \emptyset$.

Causal Graphs. We represent causal relationships using the *causal graph* of a Structural Causal Model (SCM). This is a directed acyclic mixed graph over the endogenous variables. Directed edges $X \rightarrow Y$ encode causal dependencies via structural equations, while bidirected edges $X \leftrightarrow Y$ indicate latent confounding due to shared exogenous causes.

Interventions. An *intervention* replaces one or more structural equations to model external manipulations. A common example is a *hard intervention*, written $\mathbf{do}(T = t)$, which sets a variable to a fixed value, disconnecting it from its natural causes. This defines a new SCM and alters the induced distribution.

Counterfactuals. *Counterfactual queries* reason about what would have happened under a different intervention, given an observed outcome called a factual realization. They are evaluated by conditioning on observed variables (abduction), modifying the SCM (action), and predicting outcomes under the new distribution (prediction)—a process known as the *three-step procedure* [35].

Causal Queries. A *causal query* refers to a probabilistic statement about the effect of hypothetical manipulations of the data-generating process. This includes *intervention queries*, such as Average Treatment Effect (ATE), and *counterfactual queries*, such as Counterfactual Total Effect (Ctf-TE).

Identifiability. A query is *identifiable* if its value can be uniquely determined from data, given a set of assumptions (e.g., a causal sufficiency) [35]. In other words, identifiability determines whether causal queries can be empirically estimated, and under what assumptions.

4 Problem Formulation

We consider the problem of causal inference, where the goal is to answer interventional and counterfactual queries using data drawn from an unknown SCM. Let $\mathcal{M}^* = (\mathbf{V}, \mathbf{U}, \mathcal{F}, P(\mathbf{U}))$ denote the unknown ground truth SCM, a causal query Q (e.g., Average Treatment Effect (ATE)) is defined over $\mathcal{M}^*$ and has a ground truth value $Q^* = Q(\mathcal{M}^*)$. As $\mathcal{M}^*$ is unknown, causal estimators rely

on causal assumptions $\mathbf{H}$ (e.g., causal sufficiency) and available data D drawn from $\mathcal{M}^*$ to produce an estimate $\hat{Q}$ of the target quantity Q^* . We introduce Definition 4.1 to formalize the elements of a causal dataset.

Definition 4.1 (Causal Dataset). A **causal dataset** is a tuple $\mathcal{D} = \{Q, Q^*, D, \mathcal{G}^*, \mathbf{H}^*\}$ constructed from a known SCM $\mathcal{M}^* = (\mathbf{V}, \mathbf{U}, \mathcal{F}, P(\mathbf{U}))$ where:

- Q is a causal query defined over $\mathbf{V}$;
- $Q^* = Q(\mathcal{M}^*)$ is the exact value of the query Q ;
- $D = \{D_k \sim P_{\mathcal{M}^*}(\mathbf{V} \mid \text{do}(\mathbf{V}_k) = \mathbf{v}_k)\}_{k=1}^I$ is a collection of samples under I interventional (or observational) settings;
- $\mathcal{G}^*$ is the causal graph associated with $\mathcal{M}^*$;
- $\mathbf{H}^*$ is the set of assumptions satisfied by $\mathcal{M}^*$.

Given a causal dataset $\mathcal{D} = (\{Q, Q^*, D, \mathcal{G}^*, \mathbf{H}^*\})$, one can compute the estimation error $E(\hat{Q}, Q^*)$ using a chosen error metric E (e.g., squared error). As a result, one can evaluate causal ML methods in the identification-consistent setting—where the considered causal graph and assumptions match the ground truth ones, i.e., $\mathcal{G}^*$ and $\mathbf{H}^*$ —but also test robustness by introducing assumption violations.

5 Sampling Causal Datasets with the CausalProfiler

To generate causal datasets, CausalProfiler relies on a parametric specification of the sampling domain, called the *Space of Interest*. Given an *SoI*, it samples an SCM (Section 5.2) and generates a corresponding causal dataset (Section 5.3). Appendices B, C, D, and E contain pseudocode for the sampling algorithms, and Appendix H presents a visual overview of the sampling strategy.

5.1 Defining a Space of Interest

The central abstraction of our framework is the *Space of Interest* (Definition 5.1), which provides a standardized way to specify synthetic causal datasets (Definition 4.1).

Definition 5.1 (Space of Interest). A **Space of Interest** (*SoI*) is a tuple $\mathcal{S} = \{\mathbb{M}, \mathbb{Q}, \mathbb{D}\}$, where $\mathbb{M}$ is a class of SCMs, $\mathbb{Q}$ a class of causal queries, and $\mathbb{D}$ a class of data.

Table 3 in Appendix B lists all configurable *SoI* parameters.²

5.2 Sampling Structural Causal Models

CausalProfiler samples SCMs from a user-defined *SoI* in two steps: (i) sampling a causal graph, and (ii) sampling the corresponding mechanisms.

Causal Graphs. CausalProfiler first samples a Directed Acyclic Graph over a set of endogenous variables, which defines the causal structure of the SCM. Second, if specified in the *SoI*, CausalProfiler samples a subset of endogenous variables, $\mathbf{V}_H$, to be treated as unobserved and excluded from the observed dataset. To expose only the visible causal structure to the user, we apply Verma’s latent projection algorithm [47] to the full causal graph, which produces an Acyclic Directed Mixed Graph.

Mechanisms. Given the causal graph, CausalProfiler assigns a mechanism to each endogenous variable given its parents and an exogenous noise whose distribution is set by the *SoI*. We support two types of mechanisms. First, **discrete mechanisms**, also called Regional Discrete mechanisms (see Appendix D.1 for a formal definition), are defined tabularly by associating each element of a partition of the exogenous noise with distinct parents-to-child mappings. This allows for controllable stochasticity and complexity, supporting highly non-linear and non-invertible behavior. Second, **continuous mechanisms** are defined using parametric function families—such as neural networks or linear functions—with randomly initialized parameters (e.g., He initialization [20]).

²While the current implementation of CausalProfiler supports only $\mathcal{L}_1$ training data and ATE, CATE, and CTF-TE queries, the *SoI* abstraction can, in principle, be defined over any class of queries, datasets, and SCMs.

5.3 Sampling Causal Datasets

Data D . Given an SCM $\mathcal{M}^*$ sampled from the *SoI*, we generate an observational dataset D by sampling i.i.d. data points from the entailed distribution of $\mathcal{M}^*$ over observed variables. This involves forward-sampling from the structural equations in topological order, using the noise distributions specified for each variable and marginalizing out any latent variables.

Query Q . We first sample endogenous observable variables to play the role of treatment, outcome, covariates, and factuials, depending on the class of queries of the *SoI*. To ensure that queries are well-defined and empirically grounded, we draw realizations from a large, separately sampled observational dataset, rather than from the theoretical variable domains. This avoids defining queries on realizations that may be unrepresentative or impossible under the SCM. While the currently implemented queries only involve interventions and counterfactuals, CausalProfiler also supports benchmarking causal discovery methods as the ground-truth causal graph $\mathcal{G}^*$ is directly provided in the causal dataset.

Query ground truth Q^* . Each query is estimated by drawing samples from the (manipulated) ground truth SCM: interventional queries via do-operations (action and prediction), and counterfactual queries via the three-step procedure [35]. Queries that are duplicates or yield NaN estimates are rejected and resampled to ensure valid and computable values.

Ground truth causal graph $\mathcal{G}^*$. As presented in Section 5.2, $\mathcal{G}^*$ is built as the latent projection of the ground truth SCM’s causal graph over the observed variables.

Ground truth Causal Assumptions $\mathbf{H}^*$. To characterize the properties of the ground-truth SCM from the user’s perspective, we provide an analysis module that computes summary metrics related to common causal assumptions (e.g., measuring linearity via Pearson correlation). A full list of available metrics is provided in Appendix F.

Coverage guarantee. Theorem 5.1 (proof in Appendix I) shows that, with sufficiently expressive discrete mechanisms, CausalProfiler’s sampling strategy can theoretically generate any causal dataset within a given *SoI*, guaranteeing $\mathcal{L}_3$ -expressivity. In addition, Appendix G provides an analysis exploring the empirical distribution of the sampled datasets.

Theorem 5.1 (Coverage). For a Space of Interest $\mathcal{S} = \{\mathbb{M}, \mathbb{Q}, \mathbb{D}\}$, whose class of Structural Causal Models is a class of Regional Discrete SCMs¹ with the maximum number of noise regions, denoted $\mathbb{M}_{\text{RD-SCM}, r=R_{\text{max}}}$, any causal dataset $\mathcal{D} = \{Q, Q^*, D, \mathcal{G}^*, \mathbf{H}^*\}$ has a strictly positive probability to be generated.

$$\forall \mathcal{S} = \{\mathbb{M}, \mathbb{Q}, \mathbb{D}\} \text{ s.t. } \mathbb{M} \subseteq \mathbb{M}_{\text{RD-SCM}, r=R_{\text{max}}}, P(\mathcal{D}|\mathcal{S}) > 0$$

¹Formal definition can be found in Appendix D.1.

Benchmark Design. Taken together, these design choices reflect four key properties that are considered essential for rigorous synthetic evaluation in causal ML [40]: **transparency**, by making all assumptions explicit via the parametrization of the *SoI*, which serves as a declarative specification of the evaluation domain; **repeatability**, through randomized but seed-controlled sampling procedures, ensuring that SCMs and queries can be exactly reproduced across runs; **bias awareness**, supported by the coverage guarantee and the empirical distribution analysis module and **control over experiments**, by exposing a wide range of configurable parameters in the *SoI* that allow users to tailor the causal dataset generation to their assumptions and research goals.

6 Experiments

6.1 Verification of Benchmark Correctness

To validate the soundness of our benchmark generator, we perform consistency checks based on the three levels of the Pearl Causal Hierarchy [36, 4]. Using the SCM sampler and query estimator of the CausalProfiler, we evaluate whether sampled SCMs satisfy the Markov condition, do-calculus rules, and structural counterfactual axioms [35]. We use discrete SCMs to enable exhaustive enumeration of conditioning sets for statistical tests. To ensure robustness, we iterate over a *SoI* parameter grid

spanning the number of variables, edge density, cardinalities, noise regions, and dataset sizes. For each configuration, we sample five SCMs. See Appendix J for full details and results.

L1: Markov Property Verification. We assess whether d-separations in the causal graph imply conditional independencies in the entailed observational distribution of the sampled SCMs. For each SCM, we enumerate d-separated triplets (A, B, C) and test whether $A \perp B \mid C$ holds using Pearson’s χ^2 test [37]. We filter low-sample strata (Koehler [31]) and correct for multiple tests (BH [5]). The Markov property holds in roughly 95% of tested cases, with most violations attributable to finite-sample variability (see Table 4, Appendix J).

L2: Do-Calculus Verification. We test whether the three rules of do-calculus hold empirically. For each rule, we identify variable tuples that satisfy the rule’s graphical preconditions. We then use the query estimator to generate two interventional datasets corresponding to the left- and right-hand sides of the rule. We use these datasets to compare the two distributions using Pearson’s χ^2 test, filtering low-sample strata (Koehler [31]) and correcting for multiple tests (BH [5]). Around 5.5% of tests fail, with discrepancies largely due to finite-sample noise (see Table 5, Appendix J).

L3: Structural Counterfactual Axiom Verification. We verify whether the axioms of *composition*, *effectiveness*, and *reversibility* hold exactly for sampled SCMs. Since the axioms involve deterministic functional relationships, we only count exact matches of the query estimator. All axioms hold exactly across our samples, confirming the estimator’s consistency with structural counterfactual semantics.

6.2 Setup for Experiments using the CausalProfiler

We demonstrate the utility of our benchmark framework by evaluating several recent causal inference methods across a diverse set of *SoIs*. Our goal is not to exhaustively benchmark each method but to showcase the types of structured empirical investigations our framework enables — especially those exploring robustness and violations of causal assumptions.

Evaluation Protocol. All evaluations follow the process detailed in Algorithm 1. For each *SoI*, we evaluate each method using five random seeds. For each seed, we sample 100 SCMs. For each SCM, we generate one training set and five causal queries with ground-truth values. Results are aggregated across SCMs and seeds, enabling a rigorous and reproducible assessment of performance.

Algorithm 1 Evaluation process for causal machine learning methods

```

1: Input: List of Spaces of Interest SoIs, list of seeds seeds number of examples per SCM num_examples
2: Initialize: method  $\leftarrow$  CausalMLMethod()
3: for each SoI in SoIs do
4:   for each seed in seeds do
5:     setGlobalSeed(seed)
6:     for each examples in num_examples do
7:       Generate samples, queries, and targets from the profiler
8:       Get estimates using the method on the generated samples and queries
9:       Calculate (and store) error by comparing estimates with targets
10:    end for
11:    Compute performance statistics for seed
12:  end for
13:  Compute performance statistics for SoI
14: end for
15: Output: Final summary with evaluation results

```

Hardware. All experiments were run on a single machine equipped with an Intel Core i9-14900K processor (24 cores, 32 threads) and 96GB of RAM. All CPU threads were utilized for parallel processing where applicable. Some methods (e.g., DCM) would benefit from GPU acceleration, which was not used in our evaluation.

Experiment Types. We perform two main sets of experiments: (1) ATE estimation over a set of continuous SCMs, and (2) counterfactual query estimation on discrete-variable SCMs. An additional experiment on ATE estimation under varying levels of hidden confounding is included in Appendix K.3.

Metrics and Visualization. We evaluate methods using the mean squared error between predicted and true query values. For each method and *SoI*, we report mean error, standard deviation, total runtime, and failure rate (i.e., the proportion of queries for which no valid output was returned due to numerical issues or exceptions). Tables include all numeric summaries, while box plots visualize error distributions via median, interquartile range (IQR), and whiskers extending to $1.5 \times \text{IQR}$.

Methods. We evaluate several causal inference methods: Causal Normalizing Flows (CausalNF) [27], Neural Causal Models (NCM) [11], Variational Causal Graph Autoencoder (VACA) [43], Diffusion-based Causal Models (DCM) [7], and Deconfounding Causal Normalizing Flows (DeCaFlow) [2].

Additional experiment details, results, and *SoI* configurations are provided in Appendix K.

6.3 Experiment 1: General Evaluation across Diverse SCMs

To showcase the flexibility of our benchmarking framework, we evaluate VACA, CausalNF, DCM, and NCM on a set of continuous-variable SCMs. These experiments are designed to highlight how performance can vary across diverse *SoIs*. See Table 1 for a summary of results and Figure 1 for a box plot of ATE estimation errors.

Spaces of Interest. We evaluate methods on four distinct *SoIs*: **Linear-Medium**, with linear SCMs (15-20 nodes, 1000 samples); **NN-Medium**, with neural SCMs using a 2-layer ReLU network (8 hidden units per layer, 15-20 nodes, 1000 samples); **NN-Large**, with larger neural SCMs (20-25 nodes, 1000 samples); and **NN-Large-LowData**, identical to NN-Large but with only 50 samples.

Table 1: Performance summary of CausalNF, DCM, NCM, and VACA on the general experiments.

Space	Method	Mean Error	Std Error	Max Error	Runtime (s)	Fail Rate (%)
Linear-Medium	CausalNF	0.4625	0.8985	9.6079	13790.4	0.00
Linear-Medium	DCM	0.1530	1.5289	33.9766	16541.2	0.00
Linear-Medium	NCM	0.4618	0.9001	9.6134	7384.7	0.00
Linear-Medium	VACA	0.4209	0.6195	2.3807	2734.5	53.40
NN-medium	CausalNF	0.0160	0.0107	0.1209	10732.7	0.00
NN-medium	DCM	0.0276	0.0114	0.0746	15894.4	0.00
NN-medium	NCM	0.0111	0.0121	0.1484	7322.8	0.00
NN-medium	VACA	0.0090	0.0077	0.0479	5759.6	5.00
NN-Large	CausalNF	0.0159	0.0105	0.1535	15114.8	0.00
NN-Large	DCM	0.0267	0.0100	0.0739	19166.2	0.00
NN-Large	NCM	0.0101	0.0103	0.1161	9450.6	0.00
NN-Large	VACA	0.0090	0.0094	0.0535	5690.8	11.60
NN-Large-LowData	CausalNF	0.0359	0.0146	0.1712	22138.2	0.00
NN-Large-LowData	DCM	0.0777	0.0445	0.3701	2412.1	0.00
NN-Large-LowData	NCM	0.0097	0.0107	0.1263	404.7	0.00
NN-Large-LowData	VACA	0.0103	0.0134	0.1043	5217.4	0.00

Findings (Linear-Medium vs. NN-Medium). In the Linear-Medium setting, DCM achieves the lowest average error (0.1530), indicating excellent performance. However, its error standard deviation is notably high (1.5289), driven by a few extreme outliers (max error 33.98). This implies that DCM is highly effective for most queries but may produce large errors in rare cases—potentially problematic in safety-critical applications which match this *SoI*. VACA performs competitively with lower max error and faster runtime, but suffers a high failure rate (53.4%) due to NaNs.

When moving to the NN-Medium setting, where the causal mechanisms are implemented as small neural networks, DCM’s advantage disappears. VACA emerges as the best performer, achieving both the lowest error mean (0.0090) and standard deviation (0.0077), while also reducing its failure rate to 5%. Interestingly, DCM becomes the weakest performer in this setting, highlighting that method rankings are highly sensitive to the underlying functional form of the mechanisms. This underscores the need for practitioners to evaluate methods within the *SoI* most relevant to their application.

Findings (NN-Large vs. NN-Large-LowData). In the second comparison, we increase SCM size to 20-25 nodes and investigate the effect of reducing data availability. Comparing NN-Large (1000 samples) to NN-Large-LowData (50 samples), we find that DCM is strongly affected by the data limitation: its error nearly triples (from 0.0267 to 0.0777) and its IQR expands noticeably. CausalNF also shows increased sensitivity to low-data regimes.

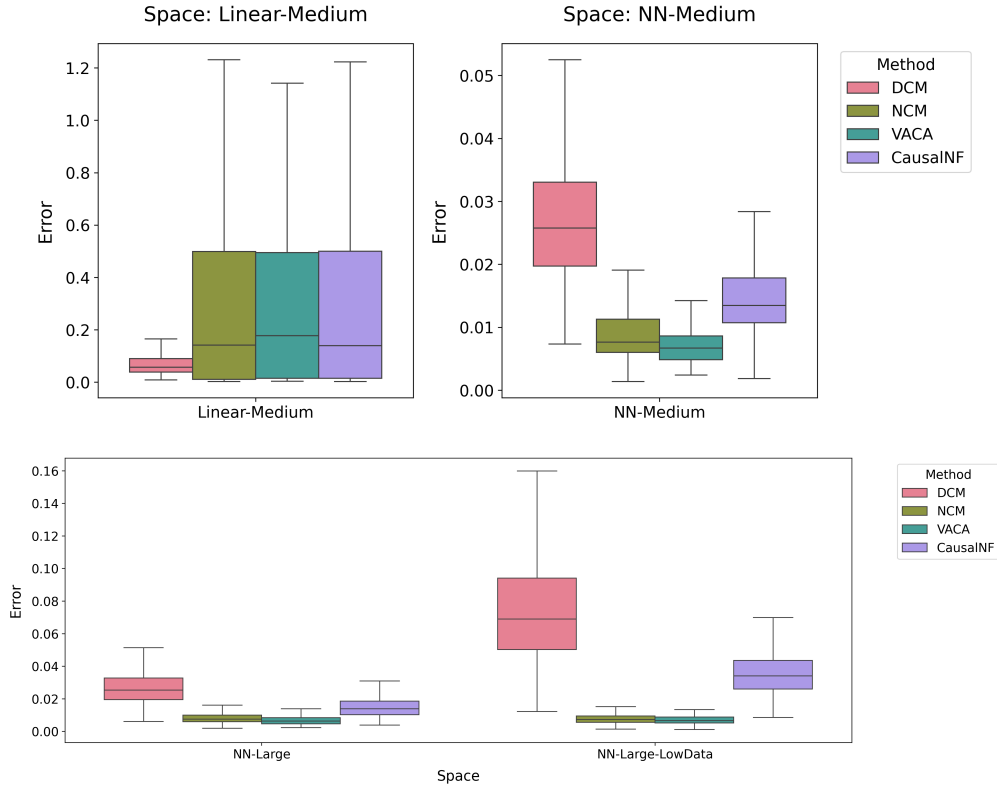


Figure 1: Box plots showing ATE estimation errors across different *SoIs*.

In contrast, both VACA and NCM maintain stable performance, with nearly unchanged mean and standard deviation values between the two *SoIs*. Notably, VACA achieves a 0% failure rate in this setting, with unexpectedly strong robustness under limited data.

Insights. While not intended as a comprehensive benchmark, these experiments illustrate the types of insights that can be drawn using our framework. For the selected Spaces of Interest, we observe that DCM tends to perform well on average but can produce large outlier errors or become less stable in low-data settings. Conversely, VACA shows promising generalization even with limited data, though it occasionally fails on certain SCMs. These findings are specific to the *SoIs* we explored, and should not be taken as general conclusions about the methods. Instead, they highlight how our framework enables structured, *SoI*-specific evaluations, helping practitioners assess which methods may be more suitable for their own modeling context.

6.4 Experiment 2: Counterfactual Estimation on Discrete SCMs

This experiment shows how our framework can evaluate counterfactual estimation methods on discrete-variable SCMs. We test CausalNF and DCM, originally designed for continuous settings, as a robustness check—motivated by prior work showing that CausalNF can sometimes effectively approximate discrete distributions [27, 11]. See Table 2 for a summary of results.

Discrete *SoIs*. We evaluate three discrete *SoIs*: **Disc-C2-Reject**, with 10-15 node graphs, binary variables, and rejection-based mechanism sampling; **Disc-C4-Unbias**, with the same graph size but 4-category variables and unbiased random mechanism sampling; and **Disc-Large-C2-Unbias**, which uses larger graphs (20-30 nodes), binary variables, and unbiased random mechanism sampling.

Findings. On Disc-C2-Reject, both CausalNF and DCM perform well and comparably, with low error means (~ 0.04) and low failure rates (8% for CausalNF, 4% for DCM). This suggests that both methods can produce reliable estimates even outside their original assumptions when the functional mechanisms are simple and binary.

Table 2: Performance summary of CausalNF and DCM on the discrete experiments.

Space	Method	Mean Error	Std Error	Max Error	Runtime	Fail Rate
Disc-C2-Reject	CausalNF	0.0415	0.1116	0.6240	212.8 s	08.08 %
Disc-C2-Reject	DCM	0.0424	0.1123	0.6240	4406.2 s	04.28 %
Disc-C4-Unbias	CausalNF	0.0431	0.1270	0.7071	190.7 s	40.68 %
Disc-C4-Unbias	DCM	0.0411	0.1199	0.7071	3839.4 s	22.60 %
Disc-Large-C2-Unbias	CausalNF	NaN	NaN	NaN	0.0 s	100.00 %
Disc-Large-C2-Unbias	DCM	0.0183	0.0814	0.5000	8192.7 s	11.32 %

However, when moving to Disc-C4-Unbias, where variables have 4 categories and mechanisms are sampled with unbiased random sampling, the failure rates increase significantly, especially for CausalNF, which fails on over 40% of SCMs (typically with NaN errors). This highlights how sensitive certain methods can be to changes in mechanism sampling or variable cardinality, even when mean errors remain similar.

To further probe robustness, we scale the graph size in Disc-Large-C2-Unbias while reverting to binary variables. CausalNF fails on all runs, returning NaNs and yielding a 100% failure rate. DCM remains functional, with an 11% failure rate, indicating greater resilience in this setting.

Insights. These results underscore the utility of our framework in systematically stress-testing methods beyond their nominal design assumptions. While CausalNF is not built for discrete data, prior examples suggested it could work in practice. Our benchmark can help clarify *when* and *how* it breaks: certain function classes and discrete configurations are more likely to cause divergence or failure. DCM appears more robust across these tests, though not immune. Importantly, this evaluation is not meant as a definitive comparison, but as a demonstration of how failure cases can be surfaced and studied in a principled way using the CausalProfiler.

7 Limitations and Future Work

Causal Datasets Distribution. While the coverage theorem guarantees that any causal dataset has a strictly positive probability of being sampled within a given *SoI* with sufficiently expressive discrete mechanisms, it does not give any information on the form of the distribution of the sampled causal datasets. In particular, certain classes of SCMs remain very unlikely to be sampled unless explicitly chosen in the *SoI* (e.g., linear SCMs). In addition, users should bear in mind that causal datasets are not uniformly generated when aggregating results, to avoid misleading interpretations. Future improvements may enable finer control over the datasets distribution and the underrepresented attributes when defining an *SoI*.

Diversify Spaces of Interest. Several directions remain open for extending the supported *SoI* by the CausalProfiler, such as support for mixed-variable SCMs, query identifiability diagnostics, sampling interventional training data, and more realistic data-generating scenarios, including selection bias, measurement noise, and partial knowledge of the causal graph.

Towards realistic causal datasets. More broadly, to increase real-world relevance, future work could enable users to define Spaces of Interest based on patterns observed in real data (e.g., a Bayesian approach), narrowing the gap between synthetic evaluation and practical deployment.

8 Conclusion

This work introduces CausalProfiler, a synthetic causal dataset generator for evaluating Causal Machine Learning methods across the three levels of the Pearl Causal Hierarchy. At its core is the notion of a Space of Interest, which replaces the ad hoc choice of a single evaluation dataset with a principled specification of the entire evaluation scope. This shift enables transparent, repeatable, and assumption-aware assessments under diverse causal conditions. We show that the performance of state-of-the-art Causal ML methods varies substantially across different *SoIs*, underscoring the importance of rigorous, distribution-level evaluation. CausalProfiler marks a first step toward more rigorous and systematic empirical practices in Causal ML—grounded not in fixed benchmarks, but in explicitly defined spaces that reflect the assumptions and structural properties relevant to each setting.

References

- [1] Oriol Abril-Pla, Virgile Andreani, Colin Carroll, Larry Dong, Christopher J Fonnesebeck, Maxim Kochurov, Ravin Kumar, Junpeng Lao, Christian C Luhmann, Osvaldo A Martin, et al. PyMC: a modern, and comprehensive probabilistic programming framework in python. *PeerJ Computer Science*, 9, 2023.
- [2] Alejandro Almodóvar, Adrián Javaloy, Juan Parras, Santiago Zazo, and Isabel Valera. DeCaFlow: A deconfounding causal generative model. *arXiv:2503.15114*, 2025.
- [3] Susan Athey, Guido W Imbens, Jonas Metzger, and Evan Munro. Using wasserstein generative adversarial networks for the design of monte carlo simulations. *Journal of Econometrics*, 240(2):105076, 2024.
- [4] Elias Bareinboim, Juan D. Correa, Duligur Ibeling, and Thomas Icard. *On Pearl’s Hierarchy and the Foundations of Causal Inference*, pages 507–556. 2022.
- [5] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society*, 57(1):289–300, 1995.
- [6] Jeroen Berrevoets, Krzysztof Kacprzyk, Zhaozhi Qian, Mihaela van der Schaar, et al. Causal deep learning: Encouraging impact on real-world problems through causality. *Foundations and Trends in Signal Processing*, 18(3):200–309, 2024.
- [7] Patrick Chao, Patrick Blöbaum, and Shiva Prasad Kasiviswanathan. Interventional and counterfactual inference with diffusion models. *arXiv:2302.00860*, 2023.
- [8] Lu Cheng, Ruocheng Guo, Raha Moraffah, Paras Sheth, K. Selçuk Candan, and Huan Liu. Evaluation methods and measures for causal learning algorithms. *IEEE Transactions on Artificial Intelligence*, 3(6):924–943, 2022.
- [9] Alicia Curth, David Svensson, Jim Weatherall, and Mihaela van der Schaar. Really doing great at estimating CATE? a critical look at ML benchmarking practices in treatment effect estimation. In *Advances in Neural Information Processing Systems*, 2021.
- [10] Alicia Curth, Richard W. Peck, Eoin McKinney, James Weatherall, and Mihaela van der Schaar. Using machine learning to individualize treatment effect estimation: Challenges and opportunities. *Clinical Pharmacology & Therapeutics*, 115(4):710–719, 2024.
- [11] Daniel de Vassimon Manela, Laura Battaglia, and Robin Evans. Marginal causal flows for validation and inference. *Advances in Neural Information Processing Systems*, 2024.
- [12] Daniel de Vassimon Manela, Laura Battaglia, and Robin J. Evans. Marginal causal flows for validation and inference. In *Advances in Neural Information Processing Systems*, 2024.
- [13] Mostafa Dehghani, Yi Tay, Alexey A Gritsenko, Zhe Zhao, Neil Houlsby, Fernando Diaz, Donald Metzler, and Oriol Vinyals. The benchmark lottery. *arXiv:2107.07002*, 2021.
- [14] Stefan Feuerriegel, Dennis Frauen, Valentyn Melnychuk, Jonas Schweisthal, Konstantin Hess, Alicia Curth, Stefan Bauer, Niki Kilbertus, Isaac S. Kohane, and Mihaela van der Schaar. Causal machine learning for predicting treatment outcomes. *Nature Medicine*, 30(4):958–968, 2024.
- [15] David Galles and Judea Pearl. An axiomatic characterization of causal counterfactuals. *Foundations of Science*, 3:151–182, 1998.
- [16] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.
- [17] Amanda Gentzel, Dan Garant, and David Jensen. The case for evaluating causal models using interventional measures and empirical data. *Advances in Neural Information Processing Systems*, 2019.
- [18] Sander Greenland and Babette Brumback. An overview of relations among causal modelling methods. *International Journal of Epidemiology*, 31(5):1030–1037, 2002.
- [19] Shantanu Gupta, Cheng Zhang, and Agrin Hilmkil. Learned causal method prediction. *arXiv:2311.03989*, 2023.
- [20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *IEEE International Conference on Computer Vision*, 2015.

- [21] Moritz Herrmann, F. Julian D. Lange, Katharina Eggersperger, Giuseppe Casalicchio, Marcel Wever, Matthias Feurer, David Rügamer, Eyke Hüllermeier, Anne-Laure Boulesteix, and Bernd Bischl. Position: Why we must rethink empirical research in machine learning. In *International Conference on Machine Learning*, 2024.
- [22] Jennifer L Hill. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240, 2011.
- [23] Paul W. Holland. Statistics and causal inference. *Journal of the American Statistical Association*, 81(396):945–960, 1986.
- [24] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257, 1991.
- [25] Yimin Huang and Marco Valtorta. Pearl’s calculus of intervention is complete. In *Conference on Uncertainty in Artificial Intelligence*, 2006.
- [26] Ben Hutchinson, Negar Rostamzadeh, Christina Greer, Katherine Heller, and Vinodkumar Prabhakaran. Evaluation gaps in machine learning practice. In *ACM Conference on Fairness, Accountability, and Transparency*, 2022.
- [27] Adrián Javaloy, Pablo Sanchez-Martin, and Isabel Valera. Causal normalizing flows: from theory to practice. In *Advances in Neural Information Processing Systems*, 2023.
- [28] Jean Kaddour, Aengus Lynch, Qi Liu, Matt J Kusner, and Ricardo Silva. Causal machine learning: A survey and open problems. *arXiv:2206.15475*, 2022.
- [29] Diviyan Kalainathan, Olivier Goudet, and Ritik Dutta. Causal discovery toolbox: Uncovering causal relationships in python. *Journal of Machine Learning Research*, 21(37):1–5, 2020.
- [30] Florian Karl, Lukas Malte Kemeter, Gabriel Dax, and Paulina Sierak. Position: Embracing negative results in machine learning. In *International Conference on Machine Learning*, 2024.
- [31] Kenneth J. Koehler and Kinley Larntz. An empirical investigation of goodness-of-fit statistics for sparse multinomials. *Journal of the American Statistical Association*, 75(370):336–344, 1980.
- [32] Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. Causal effect inference with deep latent-variable models. *Advances in Neural Information Processing Systems*, 2017.
- [33] Brady Neal, Chin-Wei Huang, and Sunand Raghupathi. Realcause: Realistic causal inference benchmarking. *arXiv:2011.15007*, 2020.
- [34] Harsh Parikh, Carlos Varjao, Louise Xu, and Eric Tchetgen Tchetgen. Validating causal inference methods. In *International Conference on Machine Learning*, 2022.
- [35] Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2nd edition, 2009.
- [36] Judea Pearl and Dana Mackenzie. *The book of why: the new science of cause and effect*. Basic books, 2018.
- [37] Karl Pearson. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 50(302):157–175, 1900.
- [38] Anne Helby Petersen. Are you doing better than random guessing? a call for using negative controls when evaluating causal discovery algorithms. *arXiv:2412.10039*, 2024.
- [39] Audrey Poinot, Alessandro Leite, Nicolas Chesneau, Michèle Sébag, and Marc Schoenauer. Learning structural causal models through deep generative models: Methods, guarantees, and challenges. In *International Joint Conference on Artificial Intelligence*, 2024.
- [40] Audrey Poinot, Panayiotis Panayiotou, Alessandro Leite, Nicolas Chesneau, and Marc Schoenauer. Position: Causal machine learning requires rigorous synthetic experiments for broader adoption. Preprint to appear in the International Conference on Machine Learning, 2025. URL <https://hal.science/hal-05066031>.
- [41] Kara E Rudolph, Nicholas T Williams, Caleb H Miles, Joseph Antonelli, and Ivan Diaz. All models are wrong, but which are useful? comparing parametric and nonparametric estimation of causal effects in finite samples. *Journal of Causal Inference*, 11(1), 2023.

- 457 [42] Amit Sharma and Emre Kiciman. DoWhy: An end-to-end library for causal inference.
458 *arXiv:2011.04216*, 2020.
- 459 [43] Pablo Sánchez-Martin, Miriam Rateike, and Isabel Valera. Vaca: Designing variational graph
460 autoencoders for causal queries. *AAAI Conference on Artificial Intelligence*, 2022.
- 461 [44] Peter WG Tennant, Eleanor J Murray, Kellyn F Arnold, Laurie Berrie, Matthew P Fox, Sarah C
462 Gadd, Wendy J Harrison, Claire Keeble, Lynsie R Ranker, Johannes Textor, et al. Use of
463 directed acyclic graphs (DAGs) to identify confounders in applied health research: review and
464 recommendations. *International Journal of Epidemiology*, 50(2):620–632, 2021.
- 465 [45] Johannes Textor, Benito van der Zander, Mark S Gilthorpe, Maciej Liśkiewicz, and George TH
466 Ellison. Robust causal inference using directed acyclic graphs: the r package ‘dagitty’. *International
467 Journal of Epidemiology*, 45(6):1887–1894, 2017.
- 468 [46] Jin Tian and Judea Pearl. A general identification condition for causal effects. In *AAAI
469 Conference on Artificial Intelligence*, 2002.
- 470 [47] Thomas Sadanand Verma. Graphical aspects of causal models. In *UCLA Cognitive Systems
471 Laboratory, Technical Report (R-191)*, 1993.
- 472 [48] Kevin Xia, Yushu Pan, and Elias Bareinboim. Neural causal models for counterfactual identifi-
473 cation and estimation. In *International Conference on Learning Representations*, 2023.
- 474 [49] Junzhe Zhang, Jin Tian, and Elias Bareinboim. Partial counterfactual identification from
475 observational and experimental data. In *International Conference on Machine Learning*, 2022.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: The claims in the abstract and introduction clearly state the paper's contributions and scope.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: We discuss the limitations of our method throughout the paper, but also explicitly in Section 7.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[Yes\]](#)

Justification: All theoretical claims are supported by clearly stated assumptions and formal proofs. The full proof of the coverage guarantee can be found in Appendix I.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: Yes. The paper fully specifies the steps needed to reproduce our main experimental results. We include pseudocode for all the algorithms in the Appendices.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: All code necessary to reproduce the experimental results is included in the supplementary material in a single zip file (allowing for double-blind reviewing), along with clear instructions. If the paper gets accepted, we will publicly release the full codebase on GitHub, including the URL in the camera-ready version of the paper. No data is required to reproduce the results so we don't provide any.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: All experimental configurations, including parameter settings and hyperparameter choices, are detailed in the appendix. Our codebase also reflects the exact setup used in all reported experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We visualize results using standard box plots with whiskers, and report means, standard deviations, failure rates, and runtimes in summary tables. All details about the verification statistical tests based on Pearson's χ^2 test are detailed in the appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.

- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We specify the hardware used for all experiments and report runtime metrics in detail, making the computational requirements clear.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines?>

Answer: [Yes]

Justification: We have reviewed the NeurIPS Code of Ethics and ensured our work complies with its principles. Our research involves no human subjects, private data, or deployment-related risks.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: Our work focuses on a synthetic benchmark generator for evaluating causal inference methods. As a methodological tool rather than an application-facing system, it is unlikely to have direct societal impact or pose misuse risks.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: We do not release pretrained models or real-world datasets. Instead, we provide code that generates fully synthetic data, which poses no known misuse risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All the code for the benchmark generator was developed by the paper authors and relies only on standard open-source libraries (NumPy and PyTorch). We will release the code under the MIT license and make it installable via pip. We will release it under the MIT license and make it available as a pip-installable package. For external causal inference methods used in our experiments (DeCaFlow, DCM, VACA, CausalNF, NCM), we cite

the original papers and respect their licensing terms. Due to license incompatibilities, the experimental code will be released separately under the GPL-3.0 license.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[Yes\]](#)

Justification: Yes, we provide extensive documentation and usage examples.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [\[NA\]](#)

Justification: The paper doesn't involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

793 Answer: [NA]
 794 Justification: The paper doesn't involve study participants.
 795 Guidelines:
 796 • The answer NA means that the paper does not involve crowdsourcing nor research with
 797 human subjects.
 798 • Depending on the country in which research is conducted, IRB approval (or equivalent)
 799 may be required for any human subjects research. If you obtained IRB approval, you
 800 should clearly state this in the paper.
 801 • We recognize that the procedures for this may vary significantly between institutions
 802 and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the
 803 guidelines for their institution.
 804 • For initial submissions, do not include any information that would break anonymity (if
 805 applicable), such as the institution conducting the review.

806 **16. Declaration of LLM usage**

807 Question: Does the paper describe the usage of LLMs if it is an important, original, or
 808 non-standard component of the core methods in this research? Note that if the LLM is used
 809 only for writing, editing, or formatting purposes and does not impact the core methodology,
 810 scientific rigor, or originality of the research, declaration is not required.

811 Answer: [NA]

812 Justification: The core method development in this research does not involve LLMs as any
 813 important, original, or non-standard component.

814 Guidelines:
 815 • The answer NA means that the core method development in this research does not
 816 involve LLMs as any important, original, or non-standard components.
 817 • Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>)
 818 for what should or should not be described.

819 A Additional definitions & Notations

Definition A.1 (Semi-Markovian and Markovian SCMs). An SCM is said to be **semi-Markovian** [35] if its set of structural equations is acyclic, meaning there exists an ordering of the equations such that for any two functions $f_i, f_j \in \mathcal{F}$, if $f_i < f_j$, then $V_j \notin \mathbf{PA}(V_i)$. This condition ensures that the causal dependencies among endogenous variables form a Directed Acyclic Graph (DAG).

An SCM is **Markovian** [35] if the exogenous variables influencing different endogenous variables are mutually independent. Formally, for all distinct $V_i, V_j \in \mathbf{V}$, we have $\mathbf{U}_{V_i} \perp\!\!\!\perp \mathbf{U}_{V_j}$. This implies the absence of latent confounding, allowing the model to be fully described by a DAG with independent noise terms.

Definition A.2 (Causal Graph of a Semi-Markovian SCM). The causal graph of a Semi-Markovian [4] SCM is an acyclic directed mixed graph (ADMG) with:

- Directed edge $V_i \rightarrow V_j$ if $V_i \in \mathbf{PA}(V_j)$
- Bi-directed edge $V_i \leftrightarrow V_j$ if $\mathbf{U}_{V_i} \not\perp\!\!\!\perp \mathbf{U}_{V_j}$

822 A.1 Interventional Quantities ($\mathcal{L}_2$)

Average Treatment Effect (ATE):

$$\text{ATE}_{T \rightarrow Y} = \mathbb{E}[Y | \text{do}(T = 1)] - \mathbb{E}[Y | \text{do}(T = 0)]$$

Conditional Average Treatment Effect (CATE):

$$\text{CATE}_{T \rightarrow Y}(\mathbf{x}) = \mathbb{E}[Y | \text{do}(T = 1), \mathbf{X} = \mathbf{x}] - \mathbb{E}[Y | \text{do}(T = 0), \mathbf{X} = \mathbf{x}]$$

Controlled Direct Effect (CDE):

$$\text{CDE}_{T \rightarrow Y}(t, c, \mathbf{m}) = \mathbb{E}[Y | \text{do}(T = t, \mathbf{M} = \mathbf{m})] - \mathbb{E}[Y | \text{do}(T = c, \mathbf{M} = \mathbf{m})]$$

Natural Direct Effect (NDE):

$$\text{NDE}_{T \rightarrow Y}(t, c) = \mathbb{E}[Y | \text{do}(T = t), \text{do}(\mathbf{M} = \mathbf{M}_c)] - \mathbb{E}[Y | \text{do}(T = c), \text{do}(\mathbf{M} = \mathbf{M}_c)]$$

823 A.2 Counterfactual Quantities ($\mathcal{L}_3$)

824 A counterfactual query such as $P(Y_{\text{do}(T=t)} | \mathbf{V}_F = \mathbf{v}_F)$ is computed by abduction (conditioning on
825 factual data), action (intervening), and prediction (computing the outcome) [35].

Ctf-TE / Ctf-DE / Ctf-IE:

$$\text{Ctf-TE}_{T \rightarrow Y}(y, t, c, \mathbf{v}_F) = P(y_{\text{do}(T=t)} | \mathbf{V}_F = \mathbf{v}_F) - P(y_{\text{do}(T=c)} | \mathbf{V}_F = \mathbf{v}_F)$$

$$\text{Ctf-DE}_{T \rightarrow Y}(y, t, c, \mathbf{v}_F) = P(y_{\text{do}(T=t), \text{do}(\mathbf{M}=\mathbf{M}_c)} | \mathbf{V}_F = \mathbf{v}_F) - P(y_{\text{do}(T=c)} | \mathbf{V}_F = \mathbf{v}_F)$$

$$\text{Ctf-IE}_{T \rightarrow Y}(y, t, c, \mathbf{v}_F) = P(y_{\text{do}(T=c), \text{do}(\mathbf{M}=\mathbf{M}_t)} | \mathbf{V}_F = \mathbf{v}_F) - P(y_{\text{do}(T=c)} | \mathbf{V}_F = \mathbf{v}_F)$$

826 B Space of Interest

827 Each Space of Interest is defined by a set of parameters that control the *SCM space*, the causal
828 queries of interest (*Query space*), and the dataset used for estimation (*Data space*). Table 3 provides
829 an overview of all configurable parameters in a Space of Interest instance, along with their default
830 values. Some parameters are only relevant under specific conditions—for instance, kernel parameters
831 are used only with continuous variables (e.g., when evaluating conditional expectations), function

sampling strategies apply exclusively to discrete mechanisms, noise regions apply only for discrete SCMs, and noise mode is ignored for tabular mechanisms (noise is already embedded in the table). Note that one can use symbolic expressions involving N (the number of nodes) and V (the cardinality of a variable) to define parameters that depend on sampled values. For example, the expected number of edges can be set as $0.5 * N$, or the number of noise regions in a discrete SCM can be set to V .

Category	Parameter	Default Value
SCM structure	Number of endogenous variables	[5, 15]
	Variable dimensionality	[1, 1]
	Expected number of edges (required)	—
	Proportion of hidden variables	0.0
	Markovian boolean flag	True
	Semi-Markovian boolean flag	False
	Predefined causal graph	—
Mechanisms	Mechanism family (e.g., Linear, NN, Tabular)	Linear
	Mechanism arguments (used to define custom NN/tabular mechanisms)	—
	Endogenous variable cardinality (for discrete variables only)	2
	Variable type	Continuous
	Discrete function sampling	Sample Rejection
	Noise mode	Additive
Noise	Noise distribution	Uniform
	Noise distribution arguments	[-1, 1]
	Number of noise regions (controls stochasticity)	N
Query	Number of queries per sample	1
	Query type	ATE
	Specific query (overrides random query sampling)	—
Kernel	Kernel type	Gaussian
	Kernel bandwidth	0.1
	Custom kernel function	—
Data	Number of samples in the set of observed data	1000

Table 3: Parameters defining a Space of Interest instance and their default values. The double lines in the table conceptually separate the SCM space, Query space, and Data space.

C Causal Graph Sampling

We first generate a random Directed Acyclic Graph (DAG) that specifies causal relations between variables. This structure is then extended by designating a subset of variables as hidden/unobserved, enabling the creation of both Markovian and semi-Markovian SCMs depending on the *SoI* spec. We separate these two steps in separate algorithms for clarity (algorithm 3 uses algorithm 2).

Algorithm 2 Generate a Random DAG with Expected Degree

Inputs: number of nodes N , expected degree d

```

1:  $V \leftarrow \{1, \dots, N\}$ 
2:  $E \leftarrow \{\}$ 
3:  $p_{edge} \leftarrow \frac{2d}{N-1}$ 
4: for  $i \in [1, N]$  do
5:    $N_{PA(i)} \sim B(i-1, p_{edge})$ 
6:    $PA(i) \leftarrow N_{PA(i)}$  nodes sampled without replacement from  $V$ 
7:    $E \leftarrow E \cup \{j \rightarrow i \mid j \in PA(i)\}$ 
8: end for

```

Output: $\mathcal{G} = \{V, E\}$

Algorithm 3 Generate a DAG with Observed and Hidden Variables

Inputs: number of nodes N , expected degree d , proportion of hidden variables p_h

- 1: $\mathcal{G} = (V, E) \leftarrow \text{DAG_sampling}(N, d)$ (see algorithm 2)
- 2: $N_h \sim B(N, p_h)$
- 3: $V_h \leftarrow N_h$ nodes sampled without replacement from V
- 4: $V_o \leftarrow V \setminus V_h$

Output: $\mathcal{G} = \{V = V_o V_h, E\}$

842 Because some variables in the DAG are unobserved, we expose only the observed structure to the user
843 in the form of an Acyclic Directed Mixed Graph. To obtain this, we apply Verma’s latent projection
844 algorithm to the causal graph of each sampled regional discrete SCM (see Algorithm 4). If a method
845 requires the true SCM, including the hidden confounders, that can be accessed as well.

Algorithm 4 Projection Algorithm [47]

Input: an Acyclic-Directed Mixed Graph (ADMG) $\mathcal{G} = \{\mathbf{V}_O, \mathbf{V}_H, \mathbf{E}\}$, with $\mathbf{V}_O$ the set of observed variables, $\mathbf{V}_H$ the set of hidden variables and $\mathbf{E}$ the mixed edges

- 1: $\mathbf{E}' \leftarrow \{\}$
- 2: **for** $A, B \in \mathbf{V}_O$ **do**
- 3: **if** there is a directed path $A \rightarrow \dots \rightarrow B$ in $\mathcal{G}$ with all intermediate nodes belonging to $\mathbf{V}_H$ **then**
- 4: $\mathbf{E}' \leftarrow \mathbf{E}' \cup \{A \rightarrow B\}$
- 5: **end if**
- 6: **if** there is a collider-free path $A \leftarrow \dots \rightarrow B$ in $\mathcal{G}$ with all intermediate nodes belonging to $\mathbf{V}_H$ **then**
- 7: $\mathbf{E}' \leftarrow \mathbf{E}' \cup \{A \leftrightarrow B\}$
- 8: **end if**
- 9: **end for**
- 10: $\mathbf{G}' \leftarrow \{\mathbf{V}_O, \mathbf{E}'\}$

Output: $\mathbf{G}'$ the latent projection of $\mathbf{G}$ over $\mathbf{V}_O$

846 D Sampling Discrete SCMs

847 D.1 Regional Discrete SCMs

848 In this work, we sample discrete Markovian SCMs inspired by [49] and [48] which we refer to as
849 **Regional discrete SCMs** as presented in definition D.1. For a description of how we generate the
850 causal graph, check Appendix C.

851

Definition D.1. Regional discrete SCM

A **regional discrete SCM** is a markovian SCM $\mathcal{M} := \{\mathbf{V}, \mathbf{U}, \mathcal{F}, P(\mathbf{U})\}$ where:

- $\mathbf{V} = \{V_1, \dots, V_d\}$ the set of finite discrete endogenous variables is divided into two sets $\mathbf{V}_o$ and $\mathbf{V}_h$ respectively representing the set of observed and hidden variables such that $\mathbf{V} = \mathbf{V}_o \cup \mathbf{V}_h$ and $\mathbf{V}_o \cap \mathbf{V}_h = \emptyset$
- $\mathbf{U} = \{U_1, \dots, U_d\}$ the set of mutually independent continuous exogenous variables is such that $\forall i \in [1, d], U_{V_i} = U_i$
- $\mathcal{F}$ the structural equations are regional discrete mechanisms as defined in definition D.2

The class of regional discrete SCMs is denoted $\mathbb{M}_{\text{RD-SCM}}$.

852

Definition D.2. Regional discrete mechanism

Given $\mathbf{I}_V = \{I_V^r\}_{r \in [1, R]}$ a partition of R parts of Ω_{U_V} and $m_V = \{m_V^r : \Omega_{\mathbf{PA}(V)} \mapsto \Omega_V\}_{r \in [1, R]}$ a set of R distinct mappings from $\Omega_{\mathbf{PA}(V)}$ to Ω_V , the **regional discrete mechanism** of an endogenous variables V is a function $f_V : \Omega_{\mathbf{PA}(V)}, \Omega_{U_V} \mapsto \Omega_V$ such that:

$$f_V(\mathbf{pa}(V), u_V) = m_r(\mathbf{PA}(V) \mapsto V) \text{ when } u_V \in I_V^r$$

I_V^r and m_r are called the r^{th} noise region and mapping of the regional discrete mechanism f_V .

853

Remark on Ω_{U_V} and R : In the definition of a regional discrete mechanism (definition D.2), no constraints are imposed on Ω_{U_V} . However, if Ω_{U_V} is discrete, then $|\Omega_{U_V}| \geq R$ is required to form a partition of R elements of Ω_{U_V} . Consequently, in order to be able to constitute such a partition for any finite R , we decided to consider continuous exogenous variables in the definition of a regional discrete SCM (definition D.1). In addition, since the m_V^r mappings are considered distinct and there are exactly $|\Omega_V|^{|\Omega_{\mathbf{PA}(V)}|}$ different mappings from V to $\mathbf{PA}(V)$, $R \leq |\Omega_V|^{|\Omega_{\mathbf{PA}(V)}|}$ is required.

860

The fact that regional discrete SCMs contains two types of endogenous variables (i.e., observed and unobserved by the user) enables the representation of complex situations where not all variables are observable. This induces the presence of potential hidden confounders from the user's perspective. As a result, the causal sufficiency assumption is no longer always respected. In our parametric definition of a Space of Interest (SoI), this phenomenon is controlled by the parameter specifying the proportion of unobserved variables among the endogenous variables. Thus, if this parameter is set to 0, the SoI's class of SCMs is included in the class of causally sufficient discrete SCMs.

868

The complexity of discrete mechanisms can be controlled by the number of noise regions. Indeed, as the number of noise regions increases, so does the complexity of the causal mechanism, in the sense that it becomes a mixture of a larger number of mappings. The distribution of a variable given its parents is, hence, more stochastic. As a result, the user-defined class of regional discrete SCMs can be very broad. and therefore more oversimplified. This provides an additional degree of complexity to make our synthetic causal datasets less trivial.

875

D.2 Discrete Mechanism Sampling strategies

We use *regional discrete mechanisms* (definition D.2), which define tabular mappings from parent variables to a target variable, conditioned on regions of the exogenous noise space. Each region induces a distinct mapping, enabling both stochasticity and high functional expressivity.

To generate these mechanisms, we support three sampling strategies described below. All methods define a partition of the exogenous noise domain Ω_U into R regions, and assign a parent-to-child mapping to each region. Let C be the cardinality of the variables, and $\Omega_{\mathbf{Pa}(V)}$ the space of parent configurations for variable V .

Controlling complexity. The number of possible mappings from parent configurations to output values grows as $|\Omega_V|^{|\Omega_{\mathbf{Pa}(V)}|}$. To keep simulations tractable, users can control the number of noise regions R . When R is small, sampling provides diverse but lightweight mechanisms. When R approaches the total number of mappings, full enumeration becomes feasible but computationally expensive.

We now describe the three supported sampling strategies.

890 Exhaustive partition

891 This strategy enumerates all possible mappings from parent configurations to output values and
 892 assigns each one to a distinct noise region ($R = |\Omega_V|^{|\Omega_{Pa(V)}|}$), ensuring complete coverage of the
 893 function space. This method guarantees maximal functional diversity across regions and can serve as
 894 a stress test for generalization under highly non-linear mechanisms. This is the only strategy where
 895 the number of noise regions is not decided by the user but rather set to the maximum.

896 Sample rejection

897 This strategy samples parent-to-output mappings uniformly at random, rejecting duplicates to ensure
 898 that each region corresponds to a distinct function. As mappings are sampled with replacement,
 899 rejection may require several attempts when R approaches the number of possible mappings.

900 We provide below a pseudocode version of this strategy. Note that lines 10–12 correspond to the
 901 rejection logic.

Algorithm 5 Generating regional discrete mechanisms with sample rejection

Inputs: set of endogenous variables $\mathbf{V}$ of cardinality C , causal graph $\mathcal{G}$, Ω_U domain of exogenous variables,
 number of noise regions R

```

1:  $\mathcal{F} \leftarrow \{\}$ 
2: for  $V \in \mathbf{V}$  do
3:    $\Omega_V \leftarrow \{1, \dots, C\}$ 
4:    $\Omega_{Pa_{\mathcal{G}}(V)} \leftarrow \{1, \dots, C\}^{|\Omega_{Pa(V)}|}$ 
5:    $R \leftarrow \min(R, |\Omega_V|^{|\Omega_{Pa(V)}|})$ 
6:    $l_{\min} \leftarrow \inf(\Omega_U)$ 
7:    $l_{\max} \leftarrow \sup(\Omega_U)$ 
8:    $\mathbf{L} = \{l_i \sim \mathcal{U}[l_{\min}, l_{\max}] \mid i \in [1, R-1]\} \cup \{l_{\min}, l_{\max}\}$ 
9:   Sort  $\mathbf{L}$  in ascending order
10:   $f_V \leftarrow \{\}$ 
11:   $m_V \leftarrow \{\}$ 
12:  for  $r \in [1, R]$  do
13:     $I_V^r \leftarrow [\mathbf{L}_r, \mathbf{L}_{r+1}[$  with  $\mathbf{L}_r$  the  $r^{th}$  element of  $\mathbf{L}$ 
14:     $m_V^r \leftarrow \{\}$ 
15:    while  $m_V^r = \{\}$  or  $m_V^r \in m_V$  do
16:       $m_V^r \leftarrow |\Omega_{Pa(V)}|$  elements sampled with replacement from  $\Omega_V$ 
17:    end while
18:     $m_V \leftarrow m_V \cup m_V^r$ 
19:     $f_V \leftarrow f_V \cup \{m_V^r; I_V^r\}$ 
20:  end for
21:   $\mathcal{F} \leftarrow \mathcal{F} \cup f_V$ 
22: end for
Output:  $\mathcal{F}$ 

```

902 Unbiased random assignment

903 In this strategy, each noise region is assigned a mapping sampled independently and without en-
 904 forcing uniqueness. As a result, multiple regions may correspond to the same function from parent
 905 configurations to outputs.

906 For example, suppose a variable has one binary parent taking values in $\{0, 1\}$, and the output variable
 907 takes values in $\{0, 1, 2\}$. One randomly sampled mapping might assign output 0 to parent value 0,
 908 and output 2 to parent value 1. Since mappings are sampled independently for each region, this same
 909 function ($0 \rightarrow 0, 1 \rightarrow 2$) may appear in multiple regions by chance.

910 This approach reflects scenarios where mechanisms are drawn independently from a distribution over
 911 functions, without enforcing any requirements on uniqueness or coverage. As a result, the effective
 912 variability in the entire system may be lower compared to other strategies, but the sampling is a lot
 913 more computationally efficient.

914 E Query Sampling and Estimation

In this work, we consider the following types of queries: ATE, Conditional Average Treatment Effect (CATE) and Counterfactual Total Effect (Ctf-TE). Their definitions can be found in Appendix A. All the queries can be defined for sets of covariates and factials belonging to the set of endogenous variables. In other words, we do not implement multi-interventions, but we consider conditioning and observing factials on several variables. Finally, the values taken by these variables (e.g., treatment and control values for ATE) must belong to their definition domain. The only parameter that controls the queries class is the type of queries chosen by the user (i.e., ATE, CATE and Ctf-TE). Thus, the class of considered queries can be defined as follows:

$$\mathcal{Q}_{\text{ATE}} = \{\text{ATE}_{T \rightarrow Y}(t, c) \mid T, Y \subseteq \mathbf{V} \text{ and } t, c \in \Omega_T\}$$

$$\mathcal{Q}_{\text{CATE}} = \{\text{CATE}_{T \rightarrow Y|\mathbf{X}}(t, c, \mathbf{x}) \mid T, Y \subseteq \mathbf{V}, \mathbf{X} \subseteq \mathbf{V} \setminus \{T, Y\} \text{ and } t, c \in \Omega_T, \mathbf{x} \in \Omega_{\mathbf{X}}\}$$

$$\mathcal{Q}_{\text{Ctf-TE}} = \{\text{Ctf-TE}_{T \rightarrow Y}(y, t, c, \mathbf{v}_F) \mid T, Y, \mathbf{V}_F \subseteq \mathbf{V} \text{ and } t, c \in \Omega_T, y \in \Omega_Y, \mathbf{v}_F \in \Omega_{\mathbf{V}_F}\}$$

915

916

917 Formally speaking, we have not integrated the causal graph as a causal query but rather as a hypothesis
 918 or prior knowledge. Indeed, except for causal discovery tasks, the causal graph is most often assumed
 919 to be known (or at least some information derived from the graph, such as the constitution of a valid
 920 adjustment set, or a valid causal ordering). Nevertheless, one can use our random causal dataset
 921 generator to evaluate causal discovery or causal representation learning methods. To do so, one just
 922 needs to retrieve the causal graph from the causal dataset directly instead of using a query.

923 Finally, a user can also implement a specific query and use it to generate synthetic causal datasets. To
 924 do this, the user has to use the Query class in our code base.

925

926 In the following algorithms, given a dataset D , a variable X and a realization x of X , we use the
 927 notation $D|_X$ (resp. $D|_{X=x}$) to represent the dataset D restricted to the variable X (resp. restricted
 928 to the samples whose X realization equals x). In addition, $B(n, p)$ denotes the Binomial law of
 929 parameters n and p .

930

931 E.1 Query Sampling

932 The following algorithms detail the procedures for sampling ATE, CATE, and CTF-TE queries.

Algorithm 6 Generating sets of observed data

Inputs: causal graph $\mathcal{G}$, causal mechanisms $\mathcal{F}$, distribution of the exogenous variables $P(\mathbf{U})$, dataset size N

```

1:  $D \leftarrow \{\}$ 
2:  $D_o \leftarrow \{\}$ 
3:  $\{\mathbf{u}_1, \dots, \mathbf{u}_N\} \sim P(\mathbf{U})$ 
4: for  $V \in \mathbf{V}$  following a causal order given by  $\mathcal{G}$  do
5:    $\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\} \leftarrow D|_{\mathbf{pa}(V)}$ 
6:    $\{u_{V_1}, \dots, u_{V_N}\} \leftarrow D|_{\mathbf{U}_V}$ 
7:    $\{v_1, \dots, v_N\} \leftarrow f_V(\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\}, \{u_{V_1}, \dots, u_{V_N}\})$ 
8:    $D \leftarrow D \cup \{v_1, \dots, v_N\}$ 
9:   if  $V \in \mathbf{V}_o$  then
10:     $D_o \leftarrow D_o \cup \{v_1, \dots, v_N\}$ 
11:   end if
12: end for
```

Output: D_o

Algorithm 7 Generating ATE queries

Inputs: set of observable endogenous variables $\mathbf{V}_o$, training set D

- 1: $T \leftarrow$ one variable randomly sampled from $\mathbf{V}_o$
- 2: $Y \leftarrow$ one variable randomly sampled from $\mathbf{V}_o$
- 3: $t \leftarrow$ one realization of T randomly sampled from $D|_T$
- 4: $c \leftarrow$ one realization of T randomly sampled from $D|_T$

Output: $Q_{ATE} = \{T, Y, t, c\}$

Algorithm 8 Generating CATE queries

Inputs: set of observable endogenous variables $\mathbf{V}_o$, training set D

- 1: $T \leftarrow$ one variable randomly sampled from $\mathbf{V}_o$
- 2: $Y \leftarrow$ one variable randomly sampled from $\mathbf{V}_o$
- 3: $d_{\mathbf{X}} \leftarrow$ an integer randomly sampled from $[1, \dots, |\mathbf{V}_o| - 2]$
- 4: $\mathbf{X} \leftarrow d_{\mathbf{X}}$ variables randomly sampled from $\mathbf{V}_o \setminus \{T, Y\}$
- 5: $t \leftarrow$ one realization of T randomly sampled from $D|_T$
- 6: $c \leftarrow$ one realization of T randomly sampled from $D|_T$
- 7: $\mathbf{x} \leftarrow$ one realization of $\mathbf{X}$ randomly sampled from $D|_{\mathbf{X}}$

Output: $Q_{CATE} = \{T, Y, \mathbf{X}, t, c, \mathbf{x}\}$

Algorithm 9 Generating Ctf-TE queries

Inputs: set of observable endogenous variables $\mathbf{V}_o$, training set D

- 1: $T \leftarrow$ one variable randomly sampled from $\mathbf{V}_o$
- 2: $Y \leftarrow$ one variable randomly sampled from $\mathbf{V}_o$
- 3: $d_{\mathbf{V}_F} \leftarrow$ an integer randomly samples from $[1, \dots, |\mathbf{V}_o|]$
- 4: $\mathbf{V}_F \leftarrow d_{\mathbf{V}_F}$ variables randomly sampled from $\mathbf{V}_o$
- 5: $t \leftarrow$ one realization of T randomly sampled from $D|_T$
- 6: $c \leftarrow$ one realization of T randomly sampled from $D|_T$
- 7: $\mathbf{v}_F \leftarrow$ one realization of $\mathbf{V}_F$ randomly sampled from $D|_{\mathbf{V}_F}$

Output: $Q_{CTF-TE} = \{T, Y, \mathbf{V}_F, t, c, \mathbf{v}_F\}$

E.2 SCM-Based Query Estimation

Each query is evaluated by modifying the SCM, sampling the exogenous variables, and computing expectations over the outcomes. In practice, we simulate interventions and counterfactuals by directly manipulating structural equations and conditioning on sampled variables. Our implementation supports efficient batch estimation using the same random seeds for reproducibility.

Counterfactual queries are estimated using the standard three-step procedure [35]:

1. **Abduction:** Condition on the factual realization to compute $P(\mathbf{U}|\mathbf{V}_F = \mathbf{v}_F)$
2. **Action:** Modify the SCM with the desired intervention
3. **Prediction:** Compute the outcome using the intervened model and posterior samples

The following algorithms detail the procedures for estimating ATE, CATE, and CTF-TE queries.

Algorithm 10 Estimating ATE queries

Inputs: ATE query to estimate $Q = \{T, Y, t, c\}$, causal graph $\mathcal{G}$, causal mechanisms $\mathcal{F}$, distribution of the exogenous variables $P(\mathbf{U})$, number of samples to draw for estimation N

```
1:  $\{\mathbf{u}_1, \dots, \mathbf{u}_N\} \sim P(\mathbf{U})$ 
2:  $D_t \leftarrow \{\mathbf{u}_1, \dots, \mathbf{u}_N\}$ 
3: for  $V \in \mathbf{V}$  following a causal order given by  $\mathcal{G}$  do
4:   if  $V = T$  then
5:      $\{v_1, \dots, v_N\} \leftarrow \{t, \dots, t\}$ 
6:   else
7:      $\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\} \leftarrow D_{t|\mathbf{PA}(V)}$ 
8:      $\{u_{V_1}, \dots, u_{V_N}\} \leftarrow D_{t|\mathbf{U}_V}$ 
9:      $\{v_1, \dots, v_N\} \leftarrow f_V(\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\}, \{u_{V_1}, \dots, u_{V_N}\})$ 
10:   end if
11:    $D_t \leftarrow D_t \cup \{v_1, \dots, v_N\}$ 
12: end for
13:  $D_c \leftarrow \{\mathbf{u}_1, \dots, \mathbf{u}_N\}$ 
14: for  $V \in \mathbf{V}$  following a causal order given by  $\mathcal{G}$  do
15:   if  $V = T$  then
16:      $\{v_1, \dots, v_N\} \leftarrow \{c, \dots, c\}$ 
17:   else
18:      $\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\} \leftarrow D_{c|\mathbf{PA}(V)}$ 
19:      $\{u_{V_1}, \dots, u_{V_N}\} \leftarrow D_{c|\mathbf{U}_V}$ 
20:      $\{v_1, \dots, v_N\} \leftarrow f_V(\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\}, \{u_{V_1}, \dots, u_{V_N}\})$ 
21:   end if
22:    $D_c \leftarrow D_c \cup \{v_1, \dots, v_N\}$ 
23: end for
24:  $Q^* \leftarrow \text{avg}(D_{t|Y}) - \text{avg}(D_{c|Y})$ 
```

Output: Q^*

Algorithm 11 Estimating CATE queries

Inputs: CATE query to estimate $Q = \{T, Y, \mathbf{X}, t, c, \mathbf{x}\}$, causal graph $\mathcal{G}$, causal mechanisms $\mathcal{F}$, distribution of the exogenous variables $P(\mathbf{U})$, number of samples to draw for estimation N

```
1:  $\{\mathbf{u}_1, \dots, \mathbf{u}_N\} \sim P(\mathbf{U})$ 
2:  $D_t \leftarrow \{\mathbf{u}_1, \dots, \mathbf{u}_N\}$ 
3: for  $V \in \mathbf{V}$  following a causal order given by  $\mathcal{G}$  do
4:   if  $V = T$  then
5:      $\{v_1, \dots, v_N\} \leftarrow \{t, \dots, t\}$ 
6:   else
7:      $\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\} \leftarrow D_{t|\mathbf{pa}(V)}$ 
8:      $\{u_{V_1}, \dots, u_{V_N}\} \leftarrow D_{t|\mathbf{U}_V}$ 
9:      $\{v_1, \dots, v_N\} \leftarrow f_V(\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\}, \{u_{V_1}, \dots, u_{V_N}\})$ 
10:   end if
11:    $D_t \leftarrow D_t \cup \{v_1, \dots, v_N\}$ 
12: end for
13:  $D_c \leftarrow \{\mathbf{u}_1, \dots, \mathbf{u}_N\}$ 
14: for  $V \in \mathbf{V}$  following a causal order given by  $\mathcal{G}$  do
15:   if  $V = T$  then
16:      $\{v_1, \dots, v_N\} \leftarrow \{c, \dots, c\}$ 
17:   else
18:      $\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\} \leftarrow D_{c|\mathbf{pa}(V)}$ 
19:      $\{u_{V_1}, \dots, u_{V_N}\} \leftarrow D_{c|\mathbf{U}_V}$ 
20:      $\{v_1, \dots, v_N\} \leftarrow f_V(\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\}, \{u_{V_1}, \dots, u_{V_N}\})$ 
21:   end if
22:    $D_c \leftarrow D_c \cup \{v_1, \dots, v_N\}$ 
23: end for
24:  $D_t \leftarrow D_{t|\mathbf{X}=\mathbf{x}}$ 
25:  $D_c \leftarrow D_{c|\mathbf{X}=\mathbf{x}}$ 
26:  $Q^* \leftarrow \text{avg}(D_{t|Y}) - \text{avg}(D_{c|Y})$ 
```

Output: Q^*

Algorithm 12 Estimating Ctf-TE queries

Inputs: Ctf-TE query to estimate $Q = \{T, Y, \mathbf{V}_F, t, c, \mathbf{v}_F\}$, causal graph $\mathcal{G}$, causal mechanisms $\mathcal{F}$, distribution of the exogenous variables $P(\mathbf{U})$, number of samples to draw for estimation N

```
1:  $\{\mathbf{u}_1, \dots, \mathbf{u}_N\} \sim P(\mathbf{U})$ 
2:  $D_{\mathbf{U}_{\mathbf{v}_F}} \leftarrow \{\mathbf{u}_1, \dots, \mathbf{u}_N\}$ 
3: for  $V \in \mathbf{V}$  following a causal order given by  $\mathcal{G}$  do
4:    $\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\} \leftarrow D_{\mathbf{U}_{\mathbf{v}_F} | \mathbf{pa}(V)}$ 
5:    $\{u_{V_1}, \dots, u_{V_N}\} \leftarrow D_{\mathbf{U}_{\mathbf{v}_F} | \mathbf{U}_V}$ 
6:    $\{v_1, \dots, v_N\} \leftarrow f_V(\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\}, \{u_{V_1}, \dots, u_{V_N}\})$ 
7:    $D_{\mathbf{U}_{\mathbf{v}_F}} \leftarrow D_{\mathbf{U}_{\mathbf{v}_F}} \cup \{v_1, \dots, v_N\}$ 
8: end for
9:  $D_{\mathbf{U}_{\mathbf{v}_F}} \leftarrow D_{\mathbf{U}_{\mathbf{v}_F} | \mathbf{V}_F = \mathbf{v}_F}$ 
10:  $M \leftarrow |D_{\mathbf{U}_{\mathbf{v}_F}}|$ 
11:  $\{\mathbf{u}_1, \dots, \mathbf{u}_M\} \leftarrow D_{\mathbf{U}_{\mathbf{v}_F} | \mathbf{U}}$ 
12:  $D_t \leftarrow \{\mathbf{u}_1, \dots, \mathbf{u}_M\}$ 
13: for  $V \in \mathbf{V}$  following a causal order given by  $\mathcal{G}$  do
14:   if  $V = T$  then
15:      $\{v_1, \dots, v_N\} \leftarrow \{t, \dots, t\}$ 
16:   else
17:      $\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\} \leftarrow D_{t | \mathbf{pa}(V)}$ 
18:      $\{u_{V_1}, \dots, u_{V_N}\} \leftarrow D_{t | \mathbf{U}_V}$ 
19:      $\{v_1, \dots, v_N\} \leftarrow f_V(\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\}, \{u_{V_1}, \dots, u_{V_N}\})$ 
20:   end if
21:    $D_t \leftarrow D_t \cup \{v_1, \dots, v_N\}$ 
22: end for
23:  $D_c \leftarrow \{\mathbf{u}_1, \dots, \mathbf{u}_M\}$ 
24: for  $V \in \mathbf{V}$  following a causal order given by  $\mathcal{G}$  do
25:   if  $V = T$  then
26:      $\{v_1, \dots, v_N\} \leftarrow \{c, \dots, c\}$ 
27:   else
28:      $\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\} \leftarrow D_{c | \mathbf{pa}(V)}$ 
29:      $\{u_{V_1}, \dots, u_{V_N}\} \leftarrow D_{c | \mathbf{U}_V}$ 
30:      $\{v_1, \dots, v_N\} \leftarrow f_V(\{\mathbf{pa}(V)_1, \dots, \mathbf{pa}(V)_N\}, \{u_{V_1}, \dots, u_{V_N}\})$ 
31:   end if
32:    $D_c \leftarrow D_c \cup \{v_1, \dots, v_N\}$ 
33: end for
34:  $Q^* \leftarrow \text{avg}(D_{t|Y}) - \text{avg}(D_{c|Y})$ 
```

Output: Q^*

943 F Assumptions analysis module's metrics

944 In order to analyze the characteristics of the sampled SCMs we implemented the following metrics.
 945 Let us imagine we sampled a regional discrete SCM $\mathcal{M} := \{\mathbf{V}, \mathbf{U}, \mathcal{F}, P(\mathbf{U})\}$ with $\mathbf{V} = (\mathbf{V}_o, \mathbf{V}_h)$
 946 and whose causal graph is denoted $\mathcal{G}$. The projection of $\mathcal{G}$ over the observable variables $\mathbf{V}_o$ is
 947 denoted $\mathcal{G}_{\mathbf{V}_o}$.

949 Analysis of the causal graph $\mathcal{G}$:

- 950 • Average in-degree: $\bar{d}_{in} = \frac{1}{|\mathbf{V}|} \sum_{V \in \mathbf{V}} |\mathbf{PA}(V)|$
- 951 • Variance of in-degree: $\text{var}(d_{in}) = \frac{1}{|\mathbf{V}|} \sum_{V \in \mathbf{V}} (|\mathbf{PA}(V)| - \bar{d}_{in})^2$
- 952 • Average number of ancestors: $|\overline{An(V)}| = \frac{1}{|\mathbf{V}|} \sum_{V \in \mathbf{V}} |An(V)|$ where $An(V)$ denotes the
 953 set of ancestors of V
- 954 • Variance of number of ancestors: $\text{var}(|An(V)|) = \frac{1}{|\mathbf{V}|} \sum_{V \in \mathbf{V}} (|An(V)| - |\overline{An(V)}|)^2$
- 955 • Average number of descendants: $|\overline{De(V)}| = \frac{1}{|\mathbf{V}|} \sum_{V \in \mathbf{V}} |De(V)|$ where $De(V)$ denotes
 956 the set of descendants of V
- 957 • Variance of number of descendants: $\text{var}(|De(V)|) = \frac{1}{|\mathbf{V}|} \sum_{V \in \mathbf{V}} (|De(V)| - |\overline{De(V)}|)^2$
- 958 • Average length of causal paths: $\bar{L} = \frac{1}{|\mathbf{p}_{\mathcal{G}}|} \sum_{p \in \mathbf{p}_{\mathcal{G}}} |p|$ where $\mathbf{p}_{\mathcal{G}}$ denotes the set of directed
 959 paths in $\mathcal{G}$
- 960 • Variance length of causal paths: $\text{var}(L) = \frac{1}{|\mathbf{p}_{\mathcal{G}}|} \sum_{p \in \mathbf{p}_{\mathcal{G}}} (|p| - \bar{L})^2$
- 961 • Maximum length of causal paths: $L_{\max} = \max_{p \in \mathbf{p}_{\mathcal{G}}} |p|$

962 Analysis of the projected causal graph $\mathcal{G}_{\mathbf{V}_o}$:

- 963 • Average number of siblings³: $|\overline{Si(V)}| = \frac{1}{|\mathbf{V}_o|} \sum_{V \in \mathbf{V}_o} |Si(V)|$ where $Si(V)$ denotes the
 964 set of siblings of V
- 965 • Variance of number of siblings: $\text{var}(|Si(V)|) = \frac{1}{|\mathbf{V}_o|} \sum_{V \in \mathbf{V}_o} (|Si(V)| - |\overline{Si(V)}|)^2$
- 966 • Number of maximal confounded components (c-comps)⁴: $|\mathbf{C}|$ where $\mathbf{C}$ denotes the set of
 967 maximal c-comps in $\mathcal{G}_{\mathbf{V}_o}$
- 968 • Average size of maximal c-comps: $|\overline{\mathbf{C}}| = \frac{1}{|\mathbf{C}|} \sum_{C \in \mathbf{C}} |C|$
- 969 • Variance of the size of maximal c-comps: $\text{var}(|\mathbf{C}|) = \frac{1}{|\mathbf{C}|} \sum_{C \in \mathbf{C}} (|C| - |\overline{\mathbf{C}}|)^2$

970 Analysis of the observational distribution $P_{\mathcal{M}}(\mathbf{V}_o)$:

- 971 • Minimum probability of the joint distribution: $p_{\mathbf{V}_o, \min} = \min_{\mathbf{v}_o \in \Omega_{\mathbf{V}_o}} P_{\mathcal{M}}(\mathbf{V}_o = \mathbf{v}_o)$
- 972 • Proportion of events with a null probability: $p_0 = \frac{1}{|\Omega_{\mathbf{V}_o}|} \sum_{\mathbf{v}_o \in \Omega_{\mathbf{V}_o}} \mathbf{1}_{P_{\mathcal{M}}(\mathbf{V}_o = \mathbf{v}_o) = 0}$ where
 973 $\mathbf{1}_\cdot$ denotes the indicator function
- Minimum probability of the marginal distributions:

$$p_{\min} = \min_{V \in \mathbf{V}_o} \min_{v \in \Omega_V} P_{\mathcal{M}}(V = v)$$

- Average minimum probability of the marginal distributions:

$$\bar{p}_{\min} = \frac{1}{|\mathbf{V}_o|} \sum_{V \in \mathbf{V}_o} \frac{1}{|\Omega_V|} \min_{v \in \Omega_V} P_{\mathcal{M}}(V = v)$$

³Two variables are considered siblings if they are linked by a bi-directed edge.

⁴We use [46] definition of (maximal) confounded components.

- Variance of the minimum probability of the marginal distributions:

$$\text{var}(p_{\min}) = \frac{1}{|\mathbf{V}_o|} \sum_{V \in \mathbf{V}_o} \left(\min_{v \in \Omega_V} P_{\mathcal{M}}(V = v) - \bar{p}_{\min} \right)^2$$

- Distance (L_1) of the joint distributions to the uniform one:

$$d(P_{\mathcal{M}}; \mathcal{U}) = \sum_{\mathbf{v}_o \in \Omega_{\mathbf{V}_o}} \left| P_{\mathcal{M}}(\mathbf{V}_o = \mathbf{v}_o) - \frac{1}{|\Omega_{\mathbf{V}_o}|} \right|$$

- Average distance (L_1) of the marginal distributions to the uniform one:

$$\overline{d(P_{\mathcal{M}}; \mathcal{U})} = \frac{1}{|\mathbf{V}_o|} \sum_{V \in \mathbf{V}_o} \sum_{v \in \Omega_V} \left| P_{\mathcal{M}}(V = v) - \frac{1}{|\Omega_V|} \right|$$

- Variance of the distance (L_1) of the marginal distributions to the uniform one:

$$\text{var}(d(P_{\mathcal{M}}; \mathcal{U})) = \frac{1}{|\mathbf{V}_o|} \sum_{V \in \mathbf{V}_o} \left(\sum_{v \in \Omega_V} \left| P_{\mathcal{M}}(V = v) - \frac{1}{|\Omega_V|} \right| - \overline{d(P_{\mathcal{M}}; \mathcal{U})} \right)^2$$

- Entropy of the joint distribution: $H(P_{\mathcal{M}}(\mathbf{V}))$

All the above-mentioned probabilities are computed from a set of 1M samples drawn from the SCM $\mathcal{M}$.

Let us note that $p_{\min}$ enables the user to check if the strong positivity assumption holds. If $p_{\mathbf{V}_o, \min} > 0$, then strong positivity is respected. In addition, if strong positivity does not hold, $p_{\mathbf{V}_o, \min}$ and p_0 indicate the extent to which the assumption is not met – the higher the metrics, the less the hypothesis is respected. On the other hand, $p_{\min}$ indicates whether the weak positivity assumption holds. If $p_{\min} > 0$, then weak positivity is respected. Finally, $d(P_{\mathcal{M}}; \mathcal{U})$, $\overline{d(P_{\mathcal{M}}; \mathcal{U})}$ and $\text{var}(d(P_{\mathcal{M}}; \mathcal{U}))$ enables the user to assess to which extent the observational distribution is imbalanced.

Analysis of the causal mechanisms $\mathcal{F}$:

- Average Pearson's correlation between the parent-child pairs⁵:

$$\bar{\rho}_P = \frac{1}{|\mathbf{V}|} \sum_{V \in \mathbf{V}} \frac{1}{|\mathbf{PA}(V) \cup U_V|} \sum_{V_j \in \mathbf{PA}(V) \cup U_V} \rho_P(V, V_j)$$

- Variance of Pearson's correlation between the parent-child pairs:

$$\text{var}(\rho_P) = \frac{1}{|\mathbf{V}|} \sum_{V \in \mathbf{V}} \frac{1}{|\mathbf{PA}(V) \cup U_V|} \sum_{V_j \in \mathbf{PA}(V) \cup U_V} (\rho_P(V, V_j) - \bar{\rho}_P)^2$$

- Average Spearman's correlation between the parent-child pairs³

$$\bar{\rho}_S = \frac{1}{|\mathbf{V}|} \sum_{V \in \mathbf{V}} \frac{1}{|\mathbf{PA}(V) \cup U_V|} \sum_{V_j \in \mathbf{PA}(V) \cup U_V} \rho_S(V, V_j)$$

- Variance of Spearman's correlation between the parent-child pairs:

$$\text{var}(\rho_S) = \frac{1}{|\mathbf{V}|} \sum_{V \in \mathbf{V}} \frac{1}{|\mathbf{PA}(V) \cup U_V|} \sum_{V_j \in \mathbf{PA}(V) \cup U_V} (\rho_S(V, V_j) - \bar{\rho}_S)^2$$

- Average conditional entropy of a variable given its parents:

$$\bar{H} = \frac{1}{|\mathbf{V}|} \sum_{V \in \mathbf{V}} H(V | \mathbf{PA}(V))$$

⁵ ρ_P and ρ_S respectively denote the Pearson's and Spearman's correlation

- Variance of conditional entropy of a variable given its parents:

$$\text{var}(\mathbf{H}) = \frac{1}{|\mathbf{V}|} \sum_{V \in \mathbf{V}} (\mathbf{H}(V|\mathbf{PA}(V)) - \bar{\mathbf{H}})^2$$

In order to be able to use person correlations, spearman correlations, and conditional entropy as indicators of degrees of linearity, monotonicity, and stochasticity of causal mechanisms, we do not derive these quantities from samples drawn from the entailed distribution. Instead, for each variable, we create a dataset resulting from the application of its causal mechanism to the cartesian product of the values taken by its endogenous and exogenous parents. In other words, we analyze the mechanisms' images of their input space. This allows us to analyze each mechanism independently of the others.

Thus, $\bar{\rho}_P$ and $\text{var}(\rho_P)$ can be interpreted as the average degree of linearity of causal mechanisms and their variance. Furthermore, $\bar{\rho}_S$ and $\text{var}(\rho_S)$ can be interpreted as the average degree of monotonicity of causal mechanisms and their variance. Finally, $\bar{\mathbf{H}}$ and $\text{var}(\mathbf{H})$ can be interpreted as the average level of stochasticity of causal mechanisms and its variance.

G Analysis of the empirical distribution of the generated SCMs

As we do not provide the user with an expression of the distribution of the sampled regional discrete SCMs, we need to investigate if some SCMs classes are over/underrepresented. This analysis is important to identify the potential biases our random causal dataset generator might create to take them into account when using it to evaluate any Causal machine learning (Causal ML) method. Indeed, as our goal is to provide a tool for rigorous empirical evaluation of causal methods, we need to be transparent on the limitations of our generator such that researchers and practitioners can interpret the results of their methods with full knowledge of the potential biases coming from the generator.

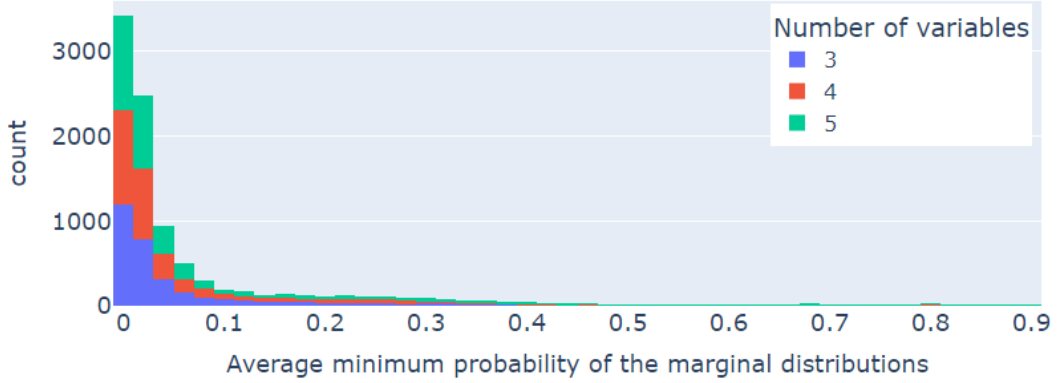
G.1 Experiment

To visualize the distribution of the SCMs generated, we analyze the distribution of the metrics of the assumption analysis module characterizing the SCMs. For each SCM sampled, all the implemented metrics (see Appendix F) are computed.

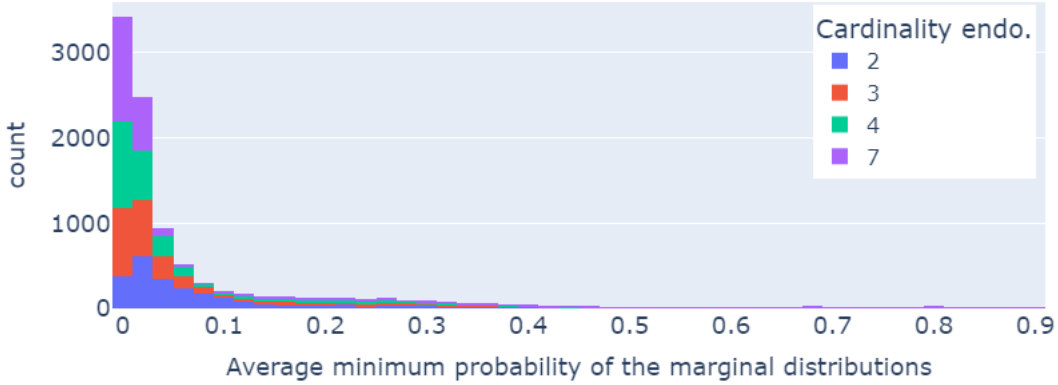
The studied SCMs are sampled from the SoIs defined by the cartesian product of the following parameters:

- **Number of endogenous variables:** [3, 4, 5]
- **Expected edge probability:** [0.2, 0.4, 0.6, 0.8]
- **Proportion of unobserved endogenous variables:** [0, 0.1, 0.2, 0.3]
- **Number of noise regions:** [2, 5, 10, 20, 50]
- **Cardinality of endogenous variables:** [2, 3, 4, 7]
- **Distribution of exogenous variables:** set to $\mathcal{U}[0, 1]$

For each SoI 10 SCMs are sampled, making a total of 9600 SCMs studied. Let us mention that we sample more SCMs than for verification (Section 6.1 for two reasons. First, it enables us to have a better approximation of the SCMs distribution. Second, the computation of all the assumptions and characteristics metrics is, in fact, less computationally expensive than computing all the independence tests.



(a) Pearson



(b) Spearman

Figure 2: Distribution of the average minimum probability of the marginal entailed distributions of the generated SCMs depending on the number of variables and their cardinality

1027 G.2 Results

1028 A number of findings about the distribution of the sampled SCMs can be drawn. For instance, the
 1029 level of stochasticity of the SCMs roughly follows a long-tailed distribution whose mean increases
 1030 with the number of variables and their cardinality. This can be seen in fig. 4.

1031 Then, the levels of linearity and monotonicity (measured using Pearson and Spearman correlations
 1032 respectively) follow roughly Gaussian distributions, see fig. 3. Distribution of mean of 0.3 and a
 1033 standard deviation of 0.1 for linearity, while for monotonicity, the standard deviation increases to 0.2.
 1034 This means that, on average, the causal mechanisms are neither linear nor monotonic.

1035 Moreover, the number and size of confounded components follow a roughly exponential distribution
 1036 (i.e., high mass close to 0, followed by exponential decay) as depicted in fig. 6. Hence, "highly
 1037 confounded" SCMs are rare.

1038 Finally, the assumption of strong positivity is rarely respected for all kind of SCMs, whereas weak
 1039 positivity is more often respected. In addition, there does not seem to be a correlation between the
 1040 size of the SCMs (i.e., number of endogenous variables and their cardinality) and the validation of
 1041 the positivity assumption. This is illustrated in figs. 2 and 5. Failure to respect these assumptions is a
 1042 direct consequence of working with finite data where infinitesimal probabilities are rounded to 0.

1043
 1044 As a result, the generated SCMs belong mainly to the non-identifiable domain of Causal ML
 1045 methods, as positivity is poorly respected. Users must, therefore, be careful in their interpretations
 1046 when evaluating methods, as identifiable SCMs are much less represented than non-identifiable
 1047 ones. We recommend starting the evaluation on small SoIs close to the identifiable domain, before

1048 progressively increasing the complexity of the causal datasets generated.
 1049

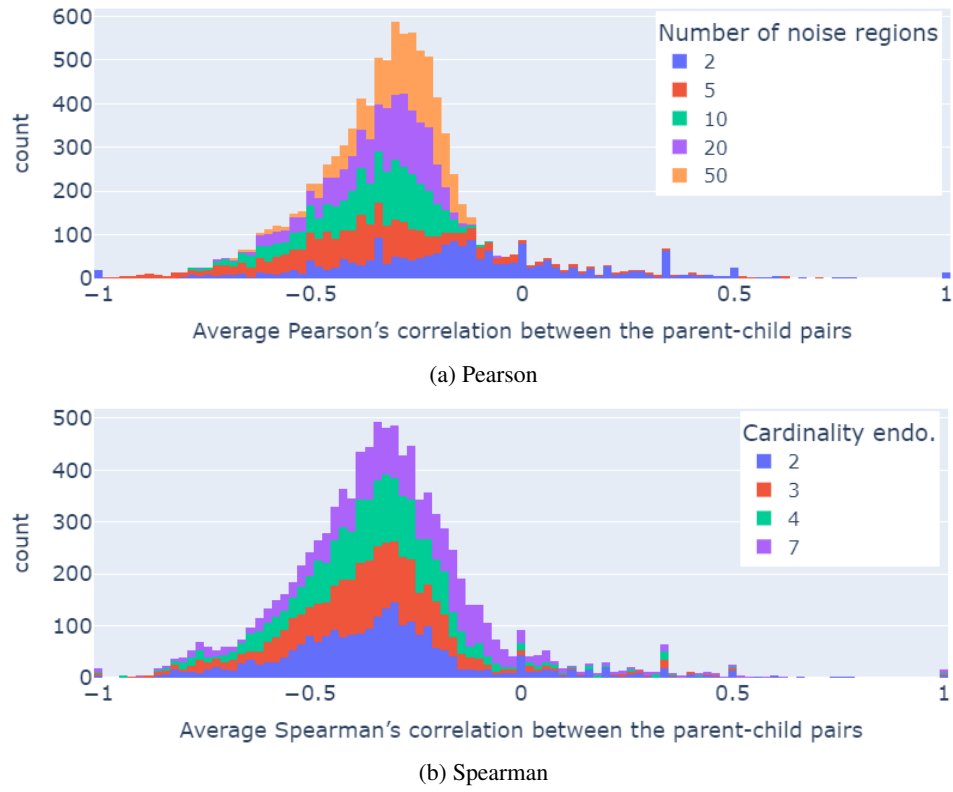
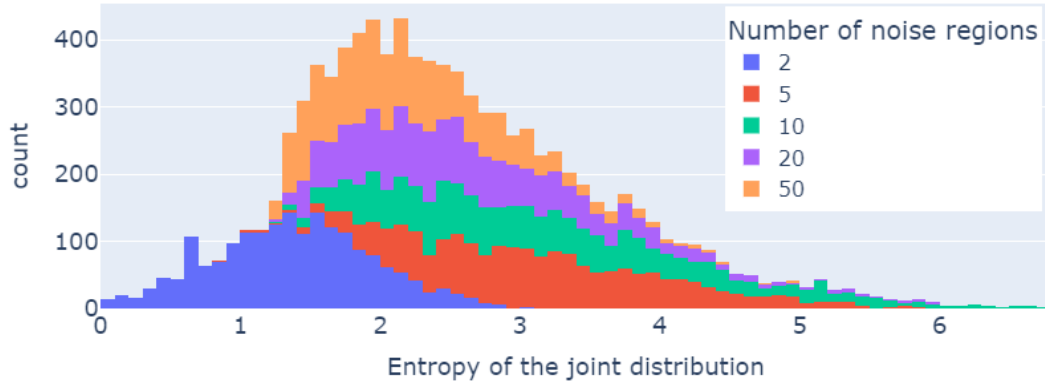
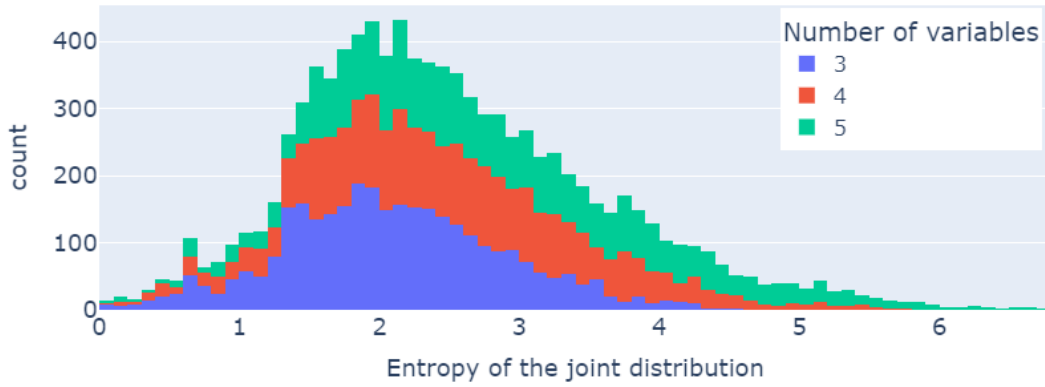


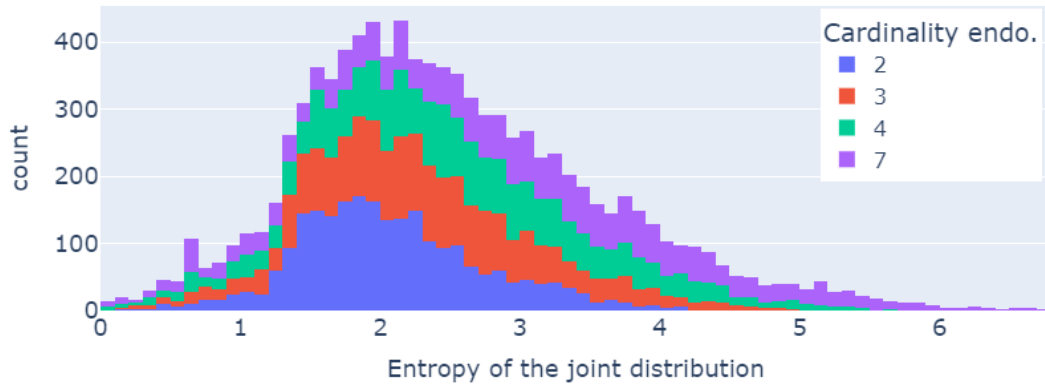
Figure 3: Distribution of the average Pearson's and Spearman's correlation between the parent-child pairs of the generated SCMs



(a) Entropy distribution depending on the number of noise regions

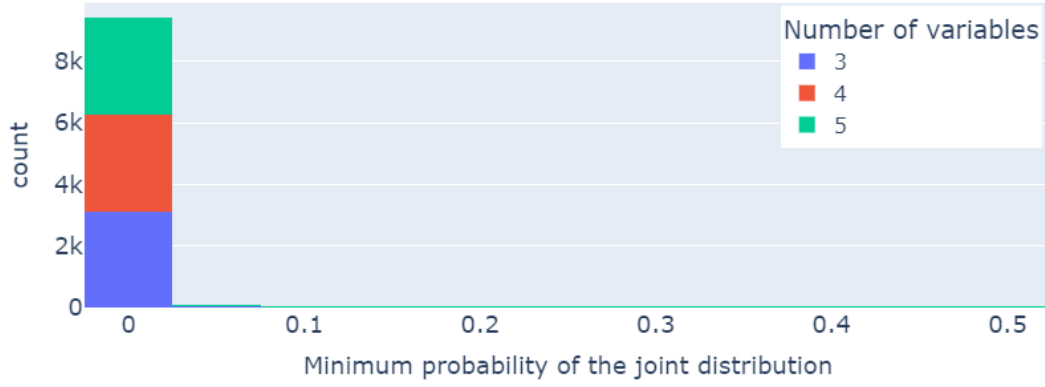


(b) Entropy distribution depending on the number of variables

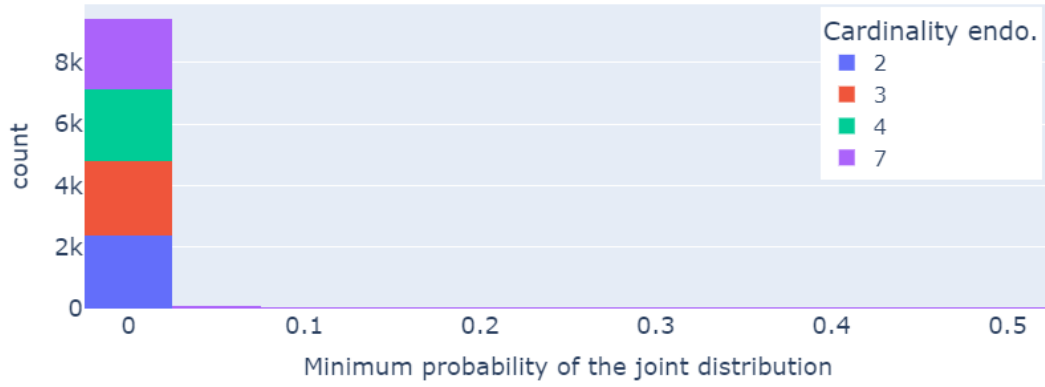


(c) Entropy distribution depending on the cardinality of variables

Figure 4: Distribution of the entropy of the entailed distribution of the generated SCMs depending on the number of noise regions, the number of variables and their cardinality



(a) Pearson



(b) Spearman

Figure 5: Distribution of the minimum probability of the joint entailed distribution of the generated SCMs depending on the number of variables and their cardinality

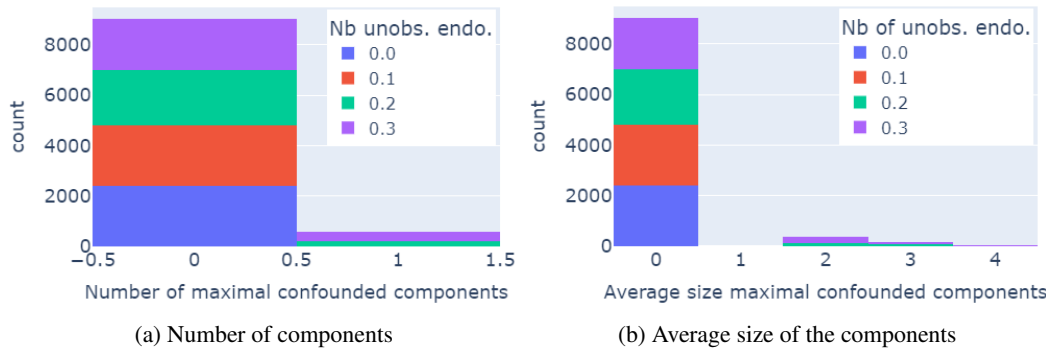


Figure 6: Distribution of the number and average size of maximal confounded components in the causal graphs of the generated SCMs depending on the number of unobserved variables

1050 H Visual overview of CausalProfiler’s sampling strategy

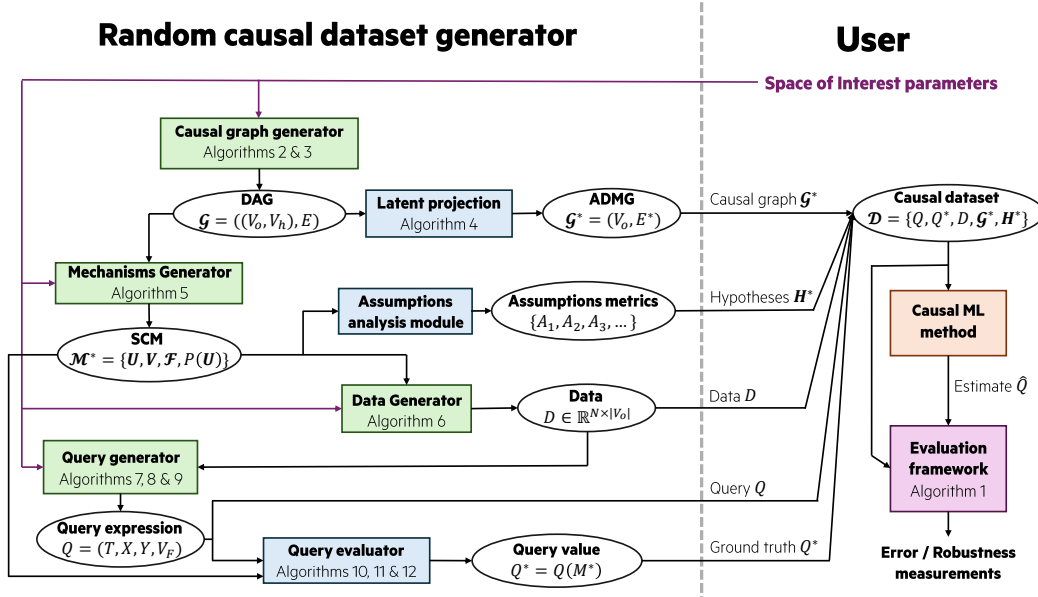


Figure 7: CausalProfiler structure. The left-hand side of the figure represents the code structure of the causal dataset generator. The right-hand side represents the user code. It illustrates how CausalProfiler can be used to evaluate a Causal ML method.

1051 I Proof of Theorem 5.1 (Coverage)

1052 This section presents the proof of Theorem 5.1 stating that: For a Space of Interest $\mathcal{S} = \{\mathbb{M}, \mathbb{Q}, \mathbb{D}\}$,
 1053 whose class of Structural Causal Models is a class of Regional Discrete SCMs with the maximum num-
 1054 ber of noise regions, any causal dataset $\mathcal{D} = \{Q, Q^*, D, \mathcal{G}^*, \mathbf{H}^*\}$ has a strictly positive probability
 1055 to be generated.

1056 Firstly, let us note that:

- 1057 • Stating that any query Q can have any ground truth value Q^* given $\mathcal{S}$ is equivalent to
 1058 saying that the class of considered SCMs, i.e., the class of Regional Discrete SCMs with the
 1059 maximum number of noise regions, is $\mathcal{L}_3$ -expressive with regards to the class of Markovian
 1060 discrete SCMs (i.e., any $\mathcal{L}_3$ -distribution of the class of Markovian discrete SCMs can be
 1061 expressed with a Regional Discrete SCM).
- 1062 • As the set of hypotheses $\mathbf{H}^*$ can contain at most $\mathcal{L}_3$ conditions, if the class of considered
 1063 SCMs is $\mathcal{L}_3$ -expressive, then any set of hypotheses $\mathbf{H}^*$ can be represented.
- 1064 • If the class of considered SCMs is $\mathcal{L}_3$ -expressive, then it is also $\mathcal{L}_1$ -expressive, hence, D
 1065 can be sampled from any distribution

1066 As a result, our proof consists of showing that $P(Q, \mathcal{G}^* | \mathcal{S}) > 0$ and that the class of Regional Discrete
 1067 SCMs with the maximum number of noise regions, denoted $\mathbb{M}_{\text{RD-SCM}, r=R_{\max}}$, is $\mathcal{L}_3$ -expressive with
 1068 regards to the class of Markovian discrete SCMs given an *SoI* $\mathcal{S}$ and a causal graph $\mathcal{G}$.

1070 Let us consider a *SoI* $\mathcal{S} = \{\mathbb{M}, \mathbb{Q}, \mathbb{D}\}$ with $\mathbb{M} \subseteq \mathbb{M}_{\text{RD-SCM}, r=R_{\max}}$.

1072 **Proving $P(\mathcal{G}^* | \mathcal{S}) > 0$:**

1073 $\mathcal{G}^*$ is built through Algorithm 3 as the latent projection of a DAG $\mathcal{G} = \{(\mathbf{V}_H, \mathbf{V}_O), E\}$ over $\mathbf{V}_O$
 1074 where $\mathcal{G}$ is sampled using Algorithm 2. As a result, following the steps of Algorithms 2 and 3:

$$\begin{aligned}
 P(\mathcal{G}^*|\mathcal{S}) &= P(\{(\mathbf{V}_H, \mathbf{V}_O), E\}|\mathcal{S}) \\
 &= P(E|\mathbf{V})P(\mathbf{V}_H, \mathbf{V}_O|\mathcal{S}) && \text{Edges are sampled independently of the} \\
 &&& \text{observability of the variables} \\
 &= P(E|\mathbf{V})P(\mathbf{V}_H, \mathbf{V}_O|\mathbf{V})P(|\mathbf{V}|) && |\mathbf{V}| \text{ and } p_h \text{ are the only parameters influ-} \\
 &&& \text{encing the observability of the variables} \\
 &= P(E|\mathbf{V})P(\mathbf{V}_H, \mathbf{V}_O|\mathbf{V})\frac{1}{N_{\max} - N_{\min}} && |\mathbf{V}| \sim \mathcal{U}[N_{\min}, N_{\max}] \\
 &= P(E|\mathbf{V})\frac{|\mathbf{V}_H|!}{|\mathbf{V}|!}\frac{1}{N_{\max} - N_{\min}} && \mathbf{V}_H \subseteq \mathbf{V} \text{ sampled without replacement} \\
 &= \frac{|\mathbf{V}_H|!}{|\mathbf{V}|!(N_{\max} - N_{\min})}P(E|\mathbf{V})
 \end{aligned}$$

1075 As $E = \{V_k \rightarrow V_i \mid V_k \in \mathbf{PA}(V_i), \forall V_i \in \mathbf{V}\}$ and the edges are sampled along the causal order $[1, N]$
 1076 with probability p_{edge} :

$$\begin{aligned}
 P(\mathcal{G}^*|\mathcal{S}) &= \frac{|\mathbf{V}_H|!}{|\mathbf{V}|!(N_{\max} - N_{\min})} \prod_{i=1}^N P(\{V_k \rightarrow V_i \mid V_k \in \mathbf{PA}(V_i)\}) \\
 &= \frac{|\mathbf{V}_H|!}{|\mathbf{V}|!(N_{\max} - N_{\min})} \prod_{i=1}^N p_{edge}^{|\mathbf{PA}(V_i)|} (1 - p_{edge})^{i-1-|\mathbf{PA}(V_i)|}
 \end{aligned}$$

1077 Let us note that $p_{edge} = 0 \implies |\mathbf{PA}(V_i)| = 0$ and $p_{edge} = 1 \implies |\mathbf{PA}(V_i)| = i - 1$. As a result,
 1078 $P(\mathcal{G}^*|\mathcal{S}) > 0$.

1080 **Proving that $\mathbb{M}_{\text{RD-SCM}, r=R_{\max}}$ is $\mathcal{L}_3$ -expressive with regards to the class of Markovian discrete**
 1081 **SCMs:** Regional discrete SCMs are, by construction, Markovian Canonical SCMs [49]. Further-
 1082 more, if the number of noise regions is chosen to be large enough (typically set to its maximum
 1083 value), any Markovian Canonical SCM can be represented using a Regional Discrete SCM⁶. Thus,
 1084 applying Zhang et al. [49] Theorem 2.4, we can assert that: for an arbitrary Markovian discrete
 1085 SCM, there exists a Regional Discrete SCM such that they both have the same causal graph and the
 1086 same $\mathcal{L}_3$ -distribution. Consequently, the class of Regional Discrete SCMs is $\mathcal{L}_3$ -expressive with
 1087 respect to the class of Markovian discrete SCMs given the causal graph $\mathcal{G}$. Moreover, $P(\mathcal{G}) > 0$ for
 1088 all $\mathcal{G}$ because $\prod_{i=1}^N p_{edge}^{|\mathbf{PA}(V_i)|} (1 - p_{edge})^{i-1-|\mathbf{PA}(V_i)|} > 0$ (cf. previous paragraph). Thus, more
 1089 generally, the class of Regional Discrete SCMs sampled by our CausalProfiler is $\mathcal{L}_3$ -expressive with
 1090 respect to the class of Markovian SCMs.

1092 **Proving $P(Q|\mathcal{G}^*, \mathcal{S}) > 0$:** Q is sampled given $\mathbb{Q}, D$ and $\mathcal{G}^*$. Even though we currently only
 1093 implement queries sampling for the classes $\mathcal{Q}_{\text{ATE}}, \mathcal{Q}_{\text{CATE}}$ and $\mathcal{Q}_{\text{Cf-TE}}$ (cf. Appendix E and Algorithms
 1094 7, 8 and 9), we can generalize our proof to any other query class (e.g., CDE, NDE). We simply
 1095 assume that these classes translate the set of constraints on the variables under consideration (e.g.,
 1096 conditioning variables have to be distinct from treatment variables or any other graphical constraints
 1097 that can be checked with $\mathcal{G}^*$) and express the probabilistic causal formula to be estimated. Once
 1098 such a query class $\mathbb{Q}$ is defined, our method randomly samples variables from $\mathbf{V}_O$ in accordance
 1099 with $\mathbb{Q}$ constraints and by sampling realizations from D . We showed in the previous paragraph that
 1100 $\mathbb{M}_{\text{RD-SCM}, r=R_{\max}}$ is $\mathcal{L}_3$ -expressive implying that it is $\mathcal{L}_1$ -expressive too. So, any realization can be
 1101 present in D . As a result, for a given query class $\mathbb{Q}$, any Q can be generated. Hence, $P(Q|\mathcal{G}^*, \mathcal{S}) > 0$.

⁶The distinction between $\mathbf{V}_O$ and $\mathbf{V}_H$ is of no importance for $\mathcal{L}_3$ -expressiveness. $\mathbf{V}_O$ and $\mathbf{V}_H$ are only used to determine what will be visible to the user as benchmark.

1103 **Proving Theorem 5.1 by combining previous results:** We proved that $\mathbb{M}_{\text{RD-SCM}, r=R_{\max}}$
 1104 is $\mathcal{L}_3$ -expressive, hence any training set D , ground truth query Q^* and set of hypotheses
 1105 $\mathbf{H}^*$ can be generated given an *SoI* $\mathcal{S}$, a causal graph $\mathcal{G}$ and a causal query Q . In addition,
 1106 $P(Q, \mathcal{G}^*|\mathcal{S}) = P(Q|\mathcal{G}^*, \mathcal{S})P(\mathcal{G}^*|\mathcal{S})$ and we also prove that $P(Q|\mathcal{G}^*, \mathcal{S}) > 0$ and $P(\mathcal{G}^*|\mathcal{S}) > 0$.
 1107 Hence, $P(Q, \mathcal{G}^*|\mathcal{S}) > 0$. As a result, any causal dataset $\mathcal{D}$ has a strictly positive probability to be
 1108 generated.

1110 **Remark on continuous SCMs.** The universal approximation theorem [24] states that NNs (with
 1111 non-polynomial activation functions) are dense in the space of continuous functions, meaning that
 1112 any continuous function can be approximated by a sequence of NNs converging to this function.
 1113 However, this does not guarantee that they strictly cover the space of continuous functions. In
 1114 particular, whenever the number of layers and neurons is finite, one can always build a continuous
 1115 function too complex to be represented with this finite number of parameters. Hence, Theorem 5.1
 1116 cannot be extended to any class of continuous SCMs. However, it could potentially be adapted not to
 1117 ask for strict coverage but rather density. We leave this question for future work.

1119 J Verification Results

1120 We design and run verification experiments targeting each level of the Pearl Causal Hierarchy (PCH).

1121 J.1 $\mathcal{L}_1$ verification

1122 Consistency with $\mathcal{L}_1$ level of the Pearl Causal Hierarchy (PCH) is tested through the verification that
 1123 the Markov property holds on randomly sampled regional discrete SCMs. Below is a description of
 1124 the experimental design choices made and the associated results.

1126 J.1.1 Experiment

1127 For a given SCM $\mathcal{M} := \{\mathbf{V}, \mathbf{U}, \mathcal{F}, P(\mathbf{U})\}$, we check that the Markov property is satisfied by
 1128 assessing whether there is a statistically significant amount of d-separations not leading to conditional
 1129 independence in the entailed distribution.
 1130 To do so, we first enumerate the list of sets of variables $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ in $\mathbf{V}$ corresponding to d-separations
 1131 in $\mathcal{M}$'s causal graph $\mathcal{G}_{\mathcal{M}}$, ie $\mathbf{A} \perp\!\!\!\perp_{\mathcal{G}_{\mathcal{M}}} \mathbf{B}|\mathbf{C}$. Second, for each d-separated set $(\mathbf{A}, \mathbf{B}, \mathbf{C})$, we test
 1132 whether $\mathbf{A} \perp\!\!\!\perp_{P_{\mathcal{M}}} \mathbf{B}|\mathbf{C}$ by sampling 50k data points from the entailed distribution $P_{\mathcal{M}}$.

1133 In practice, enumerating all the d-separations can be very costly. Moreover, as the set of variables
 1134 $\mathbf{C}$ increases, it becomes increasingly complicated to robustly test the conditional independence
 1135 $\mathbf{A} \perp\!\!\!\perp_{P_{\mathcal{M}}} \mathbf{B}|\mathbf{C}$. Indeed, as the cardinality of $\mathbf{C}$ increases, so does the number of combinations of
 1136 values for which to test independence between variables $\mathbf{A}$ and $\mathbf{B}$. Running the statistical test becomes
 1137 costly, and the data volume required for robust independence test results increases exponentially. This
 1138 is why we limit ourselves to listing the d-separated sets $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ such that $A \in \mathbf{V}$, $B \in \mathbf{V} \setminus A$, and
 1139 $C \in \mathbf{V} \cup \mathbf{V}^2 \cup \mathbf{V}^3$ by enumerating all the possible $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ tuples, and testing whether they are
 1140 d-separated in $\mathcal{G}_{\mathcal{M}}$.

1141 As the sampled SCMs are regional discrete, the conditional independence $\mathbf{A} \perp\!\!\!\perp_{P_{\mathcal{M}}} \mathbf{B}|\mathbf{C}$ can be tested
 1142 with Pearson's χ^2 independence tests [37]. More precisely, A and B are considered independent
 1143 conditionally to $\mathbf{C}$ if for all values c of $\mathbf{C}$, the H_0 hypothesis " A and B are independent" is not
 1144 rejected. Since Pearson's χ^2 test is based on the assumption that the number of samples is large, we
 1145 decide to skip tests where the Koehler criterion [31] is not met. Based on empirical analyses, this
 1146 criterion indicates whether the χ^2 test is reliable depending on the number of samples considered.
 1147 In addition, as we conduct tests for each observed value c , we need to control for the expected
 1148 proportion of false positives (represented by the Type I error of the test). To do so, we apply the
 1149 Benjamini-Hochberg correction [5].

1150 For each SoI, defined by the Cartesian product of the following parameters, we sample 5 SCMs:

- 1151 • **Number of endogenous variables:** [4, 5, 6]
- 1152 • **Expected edge probability:** [0.1, 0.4]

- **Proportion of unobserved endogenous variables:** set to 0 because the Markov property only hold for Markovian SCMs
- **Number of noise regions:** [5, 10]
- **Cardinality of endogenous variables:** [2, 3, 10]
- **Distribution of exogenous variables:** set to $\mathcal{U}[0, 1]$
- **Number of data points:** 50000

J.1.2 Results

Conditioning set size	$A \perp\!\!\!\perp_{P_{\mathcal{M}}} B C$ tests				χ^2 independence tests			
	Total	Pass	Fail	Skip	Total	Pass	Fail	Skip
$ C = 1$	100 (2 391)	91.76 (2 194)	4.94 (118)	3.3 (79)	100 (9 130)	85.4 (7 797)	1.43 (131)	13.17 (1 202)
$ C = 2$	100 (2 986)	91.16 (2 722)	5.63 (168)	3.22 (96)	100 (53 040)	45.2 (23 976)	0.33 (177)	54.46 (28 887)
$ C = 3$	100 (1 693)	91.08 (1 542)	5.67 (96)	3.25 (55)	100 (145 320)	18.49 (26 874)	0.07 (106)	81.43 (118 340)
TOTAL	100 (7 070)	91.34 (6 458)	5.40 (382)	3.25 (230)	100 (207 490)	28.26 (58 647)	0.2 (414)	71.54 (148 429)

Table 4: Conditional independence tests based on χ^2 independence tests to assess compliance of sampled SCMs with the Markov property. Results are expressed as a percentage of the total of each test type for each conditioning set size. The number of tests is also shown in brackets.

The experimental results are summarized in table 4, where it can be seen that 5.4% of the conditional independence tests failed. Despite the use of the Koehler criterion and Benjamini-Hochberg correction, some tests can still be rejected due to the random nature of finite data sampling, which can produce slight artificial correlations in the data. Moreover, on closer inspection, the majority of the failed tests (at least 350 out of 382) are unsuccessful because of a single failed χ^2 independence test. This reinforces our previous argument about the random nature of finite data sampling.

One can also notice that the number of skipped χ^2 independence tests increases with the size of the conditioning set. Such behavior is to be expected, since the number of realizations of the conditioning set increases exponentially with its cardinality, while the number of observations sampled to perform the independence tests remains constant. As a result, there are fewer and fewer observations available to perform each χ^2 test. In contrast, the number of fully skipped conditional independence tests remains constant. This means that the χ^2 skipped tests are relatively homogeneously distributed across all the conditional independence tests.

Someone might argue that the number of sampled observations should simply be automatically computed to verify the Koehler criterion. However, in general, such a calculation is complicated, if not impossible, to automate, as causal mechanisms are randomly sampled. As a result, all kinds of observational distributions can be induced with potentially very low probability realizations, for which the Koehler criterion could never be validated because the number of data to be sampled would be too large.

To conclude, these results are sufficient to conclude that the Markov property is empirically verified by the sampled SCMs.

J.2 $\mathcal{L}_2$ verification

Consistency with $\mathcal{L}_2$ level of the PCH is tested through the verification that the Do-calculus rules hold on randomly sampled regional discrete SCMs. Below is a description of the experimental design choices made (Section J.2.1) and the associated results (Section J.2.2).

1188 J.2.1 Experiment

Definition J.1. Do-Calculus rules [35]

Given an SCM $\mathcal{M} := \{\mathbf{V}, \mathbf{U}, \mathcal{F}, P(\mathbf{U})\}$ whose causal graph $\mathcal{G}$ is a DAG, and disjoint subsets $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$, and $\mathbf{W}$ of $\mathbf{V}$, the rules of the **Do-Calculus** are defined as follows:

1. **Insertion/deletion of observation:** if $\mathbf{Y}$ and $\mathbf{Z}$ are d-separated by $\mathbf{X} \cup \mathbf{W}$ in $\mathcal{G}_{\overline{\mathbf{X}}}$, then $P(\mathbf{Y}|\mathbf{do}(\mathbf{X} = \mathbf{x}), \mathbf{W}, \mathbf{Z}) = P(\mathbf{Y}|\mathbf{do}(\mathbf{X} = \mathbf{x}), \mathbf{W})$
2. **Action/observation exchange:** if $\mathbf{Y}$ and $\mathbf{Z}$ are d-separated by $\mathbf{X} \cup \mathbf{W}$ in $\mathcal{G}_{\overline{\mathbf{X}}, \underline{\mathbf{Z}}}$, then $P(\mathbf{Y}|\mathbf{do}(\mathbf{X} = \mathbf{x}), \mathbf{do}(\mathbf{Z} = \mathbf{z}), \mathbf{W}) = P(\mathbf{Y}|\mathbf{do}(\mathbf{X} = \mathbf{x}), \mathbf{Z}, \mathbf{W})$
3. **Insertion/deletion of action:** if $\mathbf{Y}$ and $\mathbf{Z}$ are d-separated by $\mathbf{X} \cup \mathbf{W}$ in $\mathcal{G}_{\overline{\mathbf{X}}, \overline{\mathbf{Z}(\mathbf{W})}}$, then $P(\mathbf{Y}|\mathbf{do}(\mathbf{X} = \mathbf{x}), \mathbf{do}(\mathbf{Z} = \mathbf{z}), \mathbf{W}) = P(\mathbf{Y}|\mathbf{do}(\mathbf{X} = \mathbf{x}), \mathbf{W})$

where $\mathcal{G}_{\overline{\mathbf{X}}}$ (resp. $\mathcal{G}_{\underline{\mathbf{X}}}$) represents the graph $\mathcal{G}$ where the incoming edges in (resp. outgoing edges from) $\mathbf{X}$ have been removed and $\mathbf{Z}(\mathbf{W})$ is the subset of nodes in $\mathbf{Z}$ that are not ancestors of any node in $\mathbf{W}$ in $\mathcal{G}_{\overline{\mathbf{X}}}$

Theorem J.1. Soundness and Completeness of the Do-Calculus rules [25]

The rules of the do-calculus are **sound** and **complete**; that is, they hold in all causal models, and all identifiable intervention distributions can be computed by an iterative application of these three rules.

1190

1191 For a given SCM, we check each rule by first enumerating the sets of d-separated variables of interest.
 1192 Second, for each d-separated set, we test whether the distributions are statistically significantly similar
 1193 by sampling 50k data points from the intervened SCMs and testing whether they are drawn from the
 1194 same distribution.

1195 For the same computational cost reasons as for $\mathcal{L}_1$ verification, we consider only univariate
 1196 sets of variables X, Y, Z , and W . In addition, the studied SCMs are sampled from the same
 1197 *SoIs* as defined in the $\mathcal{L}_1$ -verification experiment (Section J.1.1). Finally, to assess whether
 1198 two conditional distributions are identical, we used Pearson's χ^2 goodness of fit tests [37]. As
 1199 done in Section J.1, we also use the Koehler criterion [31] and the Benjamini-Hochberg correction [5].
 1200

1201 For each SoI, defined by the Cartesian product of the following parameters, we sample 5 SCMs:

- 1202 • **Number of endogenous variables:** $[4, 5, 6]$
- 1203 • **Expected edge probability:** $[0.1, 0.4]$
- 1204 • **Proportion of unobserved endogenous variables:** set to 0 because the Markov property
 1205 only hold for Markovian SCMs
- 1206 • **Number of noise regions:** $[5, 100]$
- 1207 • **Cardinality of endogenous variables:** $[2, 5]$
- 1208 • **Distribution of exogenous variables:** set to $\mathcal{U}[0, 1]$
- 1209 • **Number of data points:** 50000

1210 J.2.2 Results

1211 The experimental results are summarized in table 5 where it can be seen that they are very similar to
 1212 the $\mathcal{L}_1$ verification ones: roughly 6% of the conditional goodness of fit tests were not validated, some
 1213 tests are rejected due to the random nature of finite data sampling but the majority them (at least 570
 1214 out of 755) are unsuccessful because of a single failed χ^2 goodness of fit test.

1215 One can also notice that the percentage of skipped χ^2 goodness of fit tests is similar for rules 1 and 3
 1216 but increases by roughly 50% for rule 2. Such behavior is to be expected as rule 2 is the only rule to
 1217 have conditioning sets of size 3 on both sides of the equality. However, the number of skipped tests
 1218 remains low, with a maximum of 16%.

Do-Calculus Rule	Cond. goodness of fit				χ^2 goodness of fit			
	Total	Pass	Fail	Skip	Total	Pass	Fail	Skip
Rule 1 Insertion/deletion of observation	100 (3 378)	96.15 (3 248)	3.85 (130)	0 (0)	100 (171 092)	88.84 (152 004)	0.1 (172)	11.06 (18 916)
Rule 2 Action/observation exchange	100 (5 065)	94.04 (4 763)	5.96 (302)	0 (0)	100 (259 509)	83.84 (217 578)	0.09 (241)	16.06 (41 690)
Rule 3 Insertion/deletion of action	100 (5 169)	93.75 (4 846)	6.25 (323)	0 (0)	100 (282 184)	89.21 (251 731)	0.06 (157)	10.74 (30 296)
TOTAL	100 (13 612)	94.45 (12 857)	5.55 (755)	0 (0)	100 (712 785)	87.17 (621 313)	0.08 (570)	12.75 (90 902)

Table 5: Conditional independence tests based on χ^2 goodness of fit tests to assess compliance of sampled SCMs with the Do-Calculus rules. Results are expressed as a percentage of the total of each test type for each conditioning set size. The number of tests is also shown in brackets.

As a result, we estimate that these results are sufficient to conclude that the Do-calculus rules are respected by the sampled SCMs.

J.3 $\mathcal{L}_3$ verification

Consistency with $\mathcal{L}_3$ level of the PCH is tested through the verification that the axiomatic characterization of structural counterfactuals holds on randomly sampled regional discrete SCMs. Below is a description of the experimental design choices made (Section J.3.1) and the associated results (Section J.3.2).

Definition J.2. Axiomatic characterization of structural counterfactuals [35]

Given an SCM $\mathcal{M} := \{\mathbf{V}, \mathbf{U}, \mathcal{F}, P(\mathbf{U})\}$ whose causal graph $\mathcal{G}$ is a DAG, the **axioms of structural counterfactuals** are defined as follows:

1. **Composition:** For any sets of endogenous variables $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{W}$ in $\mathbf{V}$ and any realization $\mathbf{u}$ of $\mathbf{U}$, if $\mathbf{W}_{do(\mathbf{X}=\mathbf{x})}(\mathbf{u}) = \mathbf{w}$ then $\mathbf{Y}_{do(\mathbf{X}=\mathbf{x}), do(\mathbf{W}=\mathbf{w})}(\mathbf{u}) = \mathbf{Y}_{do(\mathbf{X}=\mathbf{x})}(\mathbf{u})$
2. **Effectiveness:** For any disjoint sets of endogenous variables $\mathbf{X}$, and $\mathbf{W}$ in $\mathbf{V}$ and any realization $\mathbf{u}$ of $\mathbf{U}$, $\mathbf{X}_{do(\mathbf{X}=\mathbf{x}), do(\mathbf{W}=\mathbf{w})}(\mathbf{u}) = \mathbf{x}$
3. **Reversibility:** For any two distinct variables X and W and any sets of other variables $\mathbf{X}$ in $\mathbf{V}$ and any realization $\mathbf{u}$ of $\mathbf{U}$, if $Y_{do(\mathbf{X}=\mathbf{x}), do(W=w)}(\mathbf{u}) = y$ and $W_{do(\mathbf{X}=\mathbf{x}), do(Y=y)}(\mathbf{u}) = w$ then $Y_{do(\mathbf{X}=\mathbf{x})}(\mathbf{u}) = y$

Note that we do not write $P(\mathbf{W}_{do(\mathbf{X}=\mathbf{x})}|\mathbf{U})$ but rather $\mathbf{W}_{do(\mathbf{X}=\mathbf{x})}(\mathbf{u})$ as it is a deterministic expression. Indeed, if $\mathbf{U}$ is fixed, there is no stochastically anymore, so we no longer need to reason in distributions but rather in functional forms.

Theorem J.2. Soundness and Completeness of structural counterfactual axioms [15]

Completeness, effectiveness, and reversibility are **sound** and **complete** in structural causal model semantics; that is they hold in all causal models and all identifiable counterfactual distributions can be computed by an iterative application of these three axioms.

J.3.1 Experiment

For a given SCM, using definition J.1 notations, we check that:

1. The **Composition** axiom is satisfied by assessing whether $\mathbf{W}_{do(\mathbf{X}=\mathbf{x})}(\mathbf{u}) = \mathbf{w}$ implies $\mathbf{Y}_{do(\mathbf{X}=\mathbf{x}), do(\mathbf{W}=\mathbf{w})}(\mathbf{u}) = \mathbf{Y}_{do(\mathbf{X}=\mathbf{x})}(\mathbf{u})$ for any sets of endogenous variables $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{W}$ in $\mathbf{V}$ and any realization $\mathbf{u}$ of $\mathbf{U}$

1236 2. The **Effectiveness** axiom is satisfied by assessing whether $\mathbf{X}_{do(\mathbf{X}=\mathbf{x}),do(\mathbf{W}=\mathbf{w})}(\mathbf{u}) = \mathbf{x}$ for
 1237 any sets of endogenous variables $\mathbf{X}$, and $\mathbf{W}$ in $\mathbf{V}$ and any realization $\mathbf{u}$ of $\mathbf{U}$

1238 3. The **Reversibility** axiom is satisfied by assessing whether $Y_{do(\mathbf{X}=\mathbf{x}),do(W=w)}(\mathbf{u}) = y$ and
 1239 $W_{do(\mathbf{X}=\mathbf{x}),do(Y=y)}(\mathbf{u}) = w$ implies $Y_{do(\mathbf{X}=\mathbf{x})}(\mathbf{u}) = y$ for any two (distinct) variables Y and
 1240 W and any sets of variables $\mathbf{X}$ in $\mathbf{V}$ and any realization $\mathbf{u}$ of $\mathbf{U}$

1241 For each SoI, defined by the Cartesian product of the following parameters, we sample 5 SCMs:

- 1242 • **Number of endogenous variables:** $[3, 5, 10]$
- 1243 • **Expected edge probability:** $[0.1, 0.5, 0.7]$
- 1244 • **Proportion of unobserved endogenous variables:** set to 0 because the Markov property
 1245 only hold for Markovian SCMs
- 1246 • **Number of noise regions:** $[3, 5, 10]$
- 1247 • **Cardinality of endogenous variables:** $[2, 5, 7]$
- 1248 • **Distribution of exogenous variables:** set to $\mathcal{U}[0, 1]$
- 1249 • **Number of data points:** 50000

1250 For each SCM, instead of enumerating all the possible four sets of variables $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{W}$, we sample
 1251 a partition of three elements of a randomly sampled subset of $\mathbf{V}$ of a size randomly picked in $[3, |\mathbf{V}|]$.
 1252 This sampling strategy enables us to make sure the three sets are disjoint and of randomly varying
 1253 size. In addition, for each four sets, we sample 50k realizations of $\mathbf{U}$.

1254 Let us note that the axioms now correspond to exact realizations and not equal probabilities. As a
 1255 result, we expect no failure as no approximation is made in this experiment.

1256 J.3.2 Results

1257 As expected, all the tested equalities are verified in our experiments. We can, therefore, consider that
 1258 the SCMs created by our generator allows the estimation of any structural counterfactual queries.

1259 K Extended Experimental Results

1260 K.1 Experiment 1: Additional Information

1261 We provide more details about the *SoI* used in our experiments in Table 6 and present extended
 1262 performance metrics in Table 7, complementing those already shown in Table 1. Parameters not
 1263 explicitly listed for a given *SoI* are set to their default values as per the benchmark configuration.
 1264 Neural Networks for our experiments have two 8-neuron layers and use ReLU activation.

Table 6: Specification of each *SoI* used in the general experiments. N denotes the sampled number of nodes.

Name	Linear-Medium	Name	NN-Medium
# Nodes	15-20	# Nodes	15-20
Mechanism	Linear	Mechanism	NN
Expected Edges	$2 \times N$	Expected Edges	$2 \times N$
Variable Type	Continuous	Variable Type	Continuous
Samples	1000	Samples	1000
Query Type	ATE	Query Type	ATE
Seeds	[10, 11, 12, 13, 14]	Seeds	[10, 11, 12, 13, 14]

Name	NN-Large	Name	NN-Large-LowData
# Nodes	20-25	# Nodes	20-25
Mechanism	NN	Mechanism	NN
Expected Edges	$2 \times N$	Expected Edges	$2 \times N$
Variable Type	Continuous	Variable Type	Continuous
Samples	1000	Samples	50
Query Type	ATE	Query Type	ATE
Seeds	[10, 11, 12, 13, 14]	Seeds	[10, 11, 12, 13, 14]

Table 7: Additional performance metrics of CausalNF, DCM, NCM, and VACA on the general experiments.

Space	Method	Min Error	Total Fail	Runtime Mean	Runtime Std
Linear-Medium	CausalNF	0.0024	0	27.58 s	18.33 s
Linear-Medium	DCM	0.0086	0	33.08 s	9.71 s
Linear-Medium	NCM	0.0024	0	14.77 s	1.42 s
Linear-Medium	VACA	0.0038	1335	11.69 s	4.54 s
NN-Medium	CausalNF	0.0019	0	21.47 s	19.52 s
NN-Medium	DCM	0.0073	0	31.79 s	10.62 s
NN-Medium	NCM	0.0014	0	14.65 s	1.43 s
NN-Medium	VACA	0.0024	125	12.13 s	4.41 s
NN-Large	CausalNF	0.0038	0	30.23 s	25.33 s
NN-Large	DCM	0.0060	0	38.33 s	14.02 s
NN-Large	NCM	0.0018	0	18.90 s	1.38 s
NN-Large	VACA	0.0023	290	12.88 s	4.31 s
NN-Large-LowData	CausalNF	0.0086	0	44.28 s	17.10 s
NN-Large-LowData	DCM	0.0121	0	4.82 s	1.34 s
NN-Large-LowData	NCM	0.0013	0	0.81 s	0.11 s
NN-Large-LowData	VACA	0.0010	0	10.43 s	4.59 s

1265 K.2 Experiment 2: Additional Information

1266 We provide more details about the *SoI* used in our experiments in Table 8 and present extended
1267 performance metrics in Table 9, complementing those already shown in Table 2. Parameters not
1268 explicitly listed for a given *SoI* are set to their default values as per the benchmark configuration.

Table 8: Specification of the Spaces of Interest used for evaluating discrete SCMs with CTF-TE queries. N denotes the sampled number of nodes.

Name	Disc-C2-Reject	Name	Disc-C4-Unbias
# Nodes	10–15	# Nodes	10–15
# Categories	2	# Categories	4
Mechanism	Tabular	Mechanism	Tabular
Sampling Strategy	Rejection	Sampling Strategy	Random
Edges	N	Edges	N
Samples	500	Samples	500
Query Type	Ctf-TE	Query Type	Ctf-TE
Seeds	[1, 2, 3, 4, 5]	Seeds	[1, 2, 3, 4, 5]

Name	Disc-Large-C2-Unbias
# Nodes	20–30
# Categories	2
Mechanism	Tabular
Sampling Strategy	Random
Edges	N
Samples	500
Query Type	Ctf-TE
Seeds	[1, 2, 3, 4, 5]

Table 9: Additional performance metrics of CausalNF and DCM on the discrete experiments.

Space	Method	Min Error	Total Fail	Runtime Mean	Runtime Std
Disc-C2-Reject	CausalNF	0.0000	202	0.46 s	0.04 s
Disc-C2-Reject	DCM	0.0000	107	8.81 s	3.55 s
Disc-C4-Unbias	CausalNF	0.0000	1017	0.42 s	0.03 s
Disc-C4-Unbias	DCM	0.0000	565	7.68 s	3.43 s
Disc-Large-C2-Unbias	CausalNF	NaN	2500	0 s	0 s
Disc-Large-C2-Unbias	DCM	0.0000	283	16.39 s	6.42 s

K.3 Experiment 3: ATE Estimation under Hidden Confounding

In this experiment, we demonstrate how our framework can be used to evaluate methods in the presence of latent confounders — a common challenge in real-world causal inference. A key goal here is not only to confirm theoretical limitations but to investigate how quickly and severely performance degrades when assumptions are violated. While theory can tell us whether identification holds, it is often agnostic to the *degree* of failure. See Table 11 for a summary of results, Table 12 for a few additional performance metrics, and Figure 8 for a boxplot of ATE estimation errors over the different *SoI*.

We focus on two linear SCM settings:

- **Linear-No-Hidden:** Linear SCMs with 10-15 nodes and full observability (no hidden confounders), using 1000 data points per SCM.
- **Linear-60-Hidden:** Same setup as above, but with 60% of the variables unobserved (hidden).

We provide more details about the *SoI* used in our experiments in Table 10. Parameters not explicitly listed for a given *SoI* are set to their default values as per the benchmark configuration.

Table 10: Specification of the *SoIs* used to evaluate performance under hidden confounding. N denotes the sampled number of nodes.

Name	Linear-No-Hidden	Name	Linear-60-Hidden
# Nodes	10-15	# Nodes	10-15
Mechanism	Linear	Mechanism	Linear
Expected Edges	$2 \times N$	Expected Edges	$2 \times N$
Variable Type	Continuous	Variable Type	Continuous
Prop. Hidden Nodes	0%	Prop. Hidden Nodes	60%
Samples	1000	Samples	1000
Query Type	ATE	Query Type	ATE
Seeds	[42, 43, 44, 45, 46]	Seeds	[42, 43, 44, 45, 46]

Setup. We evaluate three methods: CausalNF, DCM, and DeCaFlow. The first two methods assume causal sufficiency, and therefore cannot, in theory, handle hidden confounding. DeCaFlow, in contrast, is explicitly designed for this setting but requires access to the full causal graph (including hidden variables) and does not run when all variables are observed. Thus, we include it only in the hidden confounding *SoI*.

Results (Linear-No-Hidden). As expected, both CausalNF and DCM perform well when all variables are observed. DCM achieves lower mean error (0.0845) and standard deviation (0.1515), with a maximum error of 2.89. The upper whisker of DCM’s box plot lies below the median of CausalNF, indicating consistent superior performance. These results serve as a reference point for comparison when introducing hidden variables.

Results (Linear-60-Hidden). With 60% of variables hidden, method performance degrades significantly. DeCaFlow performs reliably, with an error mean of 0.3405 and low variance. In contrast, CausalNF—despite a box plot that visually appears well-behaved—has a massive error mean of 2.67×10^{12} and a maximum error exceeding 10^{15} . This is due to a small subset of SCMs producing extremely large errors (14 with error > 1000), illustrating that, when assumptions are violated, error can become arbitrarily large. While DCM does not show such instability on this particular sample, its theoretical limitations under hidden confounding still hold — the expectation is that if we evaluate over enough SCMs we will eventually also get arbitrarily large errors due to the violation of the causal sufficiency assumption.

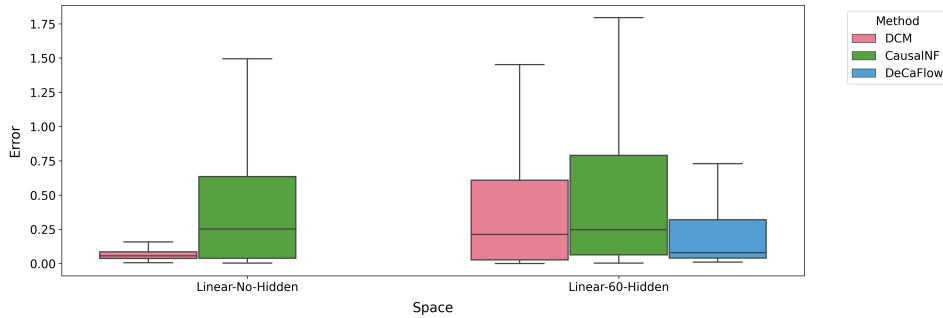


Figure 8: Box plots of ATE estimation errors in the presence and absence of hidden confounding. Each box shows the interquartile range and median, with whiskers extending to $1.5 \times \text{IQR}$. CausalNF and DCM are shown for both *SoIs*; DeCaFlow is shown only for the hidden setting.

Table 11: Performance summary of CausalNF, DCM, and DeCaFlow on the hidden confounder experiments.

Space	Method	Mean Error	Std Error	Max Error	Runtime (s)
Linear-No-Hidden	CausalNF	0.5538	0.9866	14.2495	8570.0
Linear-No-Hidden	DCM	0.0845	0.1515	2.8954	12144.6
Linear-60-Hidden	CausalNF	2.667e+12	5.497e+13	1.225e+15	293.2
Linear-60-Hidden	DCM	0.5584	1.2122	17.2049	4187.6
Linear-60-Hidden	DeCaFlow	0.3405	0.6799	5.9435	2264.0

Table 12: Additional performance metrics of CausalNF, DCM, and DeCaFlow on the hidden confounder experiments.

Space	Method	Min Error	Total Fail	Runtime Mean	Runtime Std
Linear-No-Hidden	CausalNF	0.0036	0	17.14 s	10.61 s
Linear-No-Hidden	DCM	0.0068	0	24.29 s	7.64 s
Linear-60-Hidden	CausalNF	0.0029	0	0.59 s	0.02 s
Linear-60-Hidden	DCM	0.0000	0	8.38 s	3.45 s
Linear-60-Hidden	DeCaFlow	0.0108	0	4.53 s	1.27 s