TCD-ARENA: ASSESSING ROBUSTNESS OF TIME SERIES CAUSAL DISCOVERY METHODS AGAINST ASSUMPTION VIOLATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Causal Discovery (CD) is a powerful framework for scientific inquiry. Yet, its practical adoption is hindered by a reliance on strong, often unverifiable assumptions and a lack of robust performance assessment. To address these limitations and advance empirical CD evaluation, we present **TCD-Arena** a modularized and extendable testing kit to assess the robustness of time series CD algorithms against stepwise more severe assumption violations. For demonstration, we conduct an extensive empirical study comprising over 50 million individual CD attempts and reveal nuanced robustness profiles for 27 distinct assumption violations. Further, we investigate CD ensembles and find that they can boost general robustness, which has implications for real-world applications. With this, we strive to ultimately facilitate the development of CD methods that are reliable for a diverse range of synthetic and potentially real-world data conditions.

1 INTRODUCTION

Causal Discovery (CD) holds great potential for addressing scientific hypotheses in fields where randomized control trials are difficult or impossible Glymour et al. (2019). Despite this promise, the widespread adoption of CD methods by practitioners remains limited. Recent works (Brouillard et al., 2024; Yi et al., 2024; Faller et al., 2024) attribute this to mainly two key factors: First, existing CD methods often rely on strong, idealized assumptions (e.g., no hidden confounders or stationarity) that are difficult to validate or are simply unverifiable in real-world scenarios, even if they underpin theoretical guarantees. Second, empirical evaluations of CD methods predominantly use idealized synthetic data, which can overestimate performance and offer limited insight into robustness under imperfect but realistic conditions. Consequently, practitioners hesitate to adopt CD methods where their output reliability is limited (Kaiser & Sipos, 2021; Nastl & Hardt, 2024; Poinsot et al., 2025). To overcome this issue, there has been a recent push towards more benchmarking as it is the de facto golden standard in Machine Learning (Neal et al., 2023; Stein et al., 2024a; Wang, 2024; Mogensen et al., 2024; Herdeanu et al., 2025). However, the scarcity of real-world datasets with known causal ground truth continues to hinder a full reliance on empirical validation of CD methods. As a possible alternative to real-world benchmarks, recent studies investigate CD performance when specific assumptions are violated (Montagna et al., 2023b; Yi et al., 2024; Ferdous et al., 2025). Furthermore, the robustness of CD methods related to hyperparameter selection has been recently highlighted (Machlanski et al., 2024). Building upon and aiming to unify these emerging efforts of empirical evaluation, we present **TCD-Arena**, a modularized testing kit to assess CD robustness against assumption violation. Next to an unprecedented scale TCD-Arena focuses on three so far sporadically addressed aspects: **(1) temporal data**, that introduces additional challenges and opportunities; **(2) stepwise violation intensities**, crucial for capturing nuanced performance degradation rather than binary pass/fail outcomes; and **(3) a focus on violations often encountered in real-world settings.** In this paper, we demonstrate the benefits of TCD-Arena by conducting an extensive empirical study on the robustness of CD algorithms. Specifically, we evaluate eight CD algorithms that cover all four common CD archetypes Assaad et al. (2022). We evaluate these algorithms across 27 different assumption violations, each scaled in intensity. By performing over 50 million individual CD attempts, we find that various methods differ in their ability to cope with assumption violations. Additionally, we investigate hyperparameter sensitivities with respect to

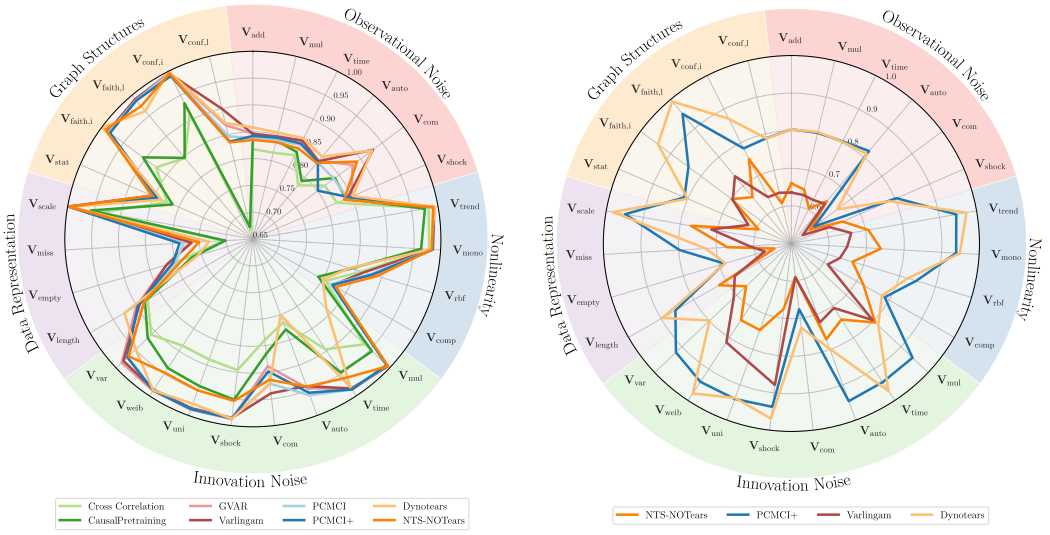


Figure 1: Robustness profiles of eight Causal Discovery algorithms against a multitude of stepwise assumption violations measured as average AUROC over various data regimes. Left: Lagged causal effects, Right: Instantanous causal effects

robustness and model misspecifications, two aspects that we believe to be critical for applications to novel real-world data. Further, we investigate ensembles of CD methods, something that has received little attention in the literature, and conclude that they can boost robustness. Standing with Poinsot et al. (2025) and recognizing the pressing need for more nuanced CD evaluation, we attempt to further establish robustness analysis as an alternative to traditional benchmarking and theoretical analysis. With this, we hope to ultimately facilitate the development of CD methods that are reliably applicable to real-world data. In summary, this paper makes the following contributions:

1. The introduction of TCD-Arena, an open-source and customizable toolkit for quantifying the robustness of CD in diverse time-series data and fosters long-term comparability.
2. A large-scale empirical study that evaluates the robustness of eight time series CD methods against 27 stepwise intensified assumption violations.
3. An investigation into ensembling CD methods with respect to violation robustness.

2 Background and Theoretical Preliminaries

To ground our empirical investigation, we begin by selectively revisiting the relevant theoretical background. Let $\boldsymbol{X} \in \mathbb{R}^{D \times T}$ be a D-variate time series comprising T samples from D interacting variables, generated by an unknown underlying causal process. The objective of time series Causal Discovery (CD) is to infer the causal relationships among D variables from the observed data $\boldsymbol{X}$. These relationships are commonly represented as a Structural Causal Model (SCM) (Peters et al., 2017). For each variable $X_{i,t}$, the SCM contains assignments of the form:

$$X_{i,t} = f_i\left(\mathrm{Pa}(X_{i,t}), \epsilon_{i,t}\right), \tag{1}$$

where $\mathrm{Pa}(X_{i,t})$ is the set of direct causal parents of $X_{i,t}$, f_i is a causal mechanism, and $\epsilon_{i,t}$ is independent innovation noise. The set of assignments within an SCM defines a directed graph $G = (V, E)$. In this work, we distinguish between contemporaneous ($X_{j,t}$ for $j \neq i$) and lagged effects ($X_{j,t-k}$ for $k > 0$) by evaluating the recovery of the following three distinct graph structures: First, the lagged window causal graph (G^{WCG}) provides a lag-specific view of causal dependencies up to a maximum lag L. Here V includes each variable at time step t and at all relevant past lags: $V = \{X_{i,t-l} \mid i \in \{1, \ldots, D\}, l \in \{1, \ldots, L\}\}$. A directed edge $X_{j,t-l} \to X_{i,t}$ exists in G^{WCG} if $X_{j,t-l}$ is in $\mathrm{Pa}(X_{i,t})$. Note that in this representation, edges only connect past variables to variables at step t. Second, the lagged summary graph (G^{SG}) provides a high-level summary of time-lagged

relationships. Its vertices are defined as $V = \{X_1, \ldots, X_D\}$. A directed edge $X_j \to X_i$ exists in G^{SG} if $X_{j,t-l} \in \text{Pa}(X_{i,t})$ for at least one $l > 0$. Third, the instantaneous graph (G^{INST}) captures only contemporaneous relationships. It is a directed graph with vertices $V = \{X_1, \ldots, X_D\}$. A directed edge $X_j \to X_i$ exists in G^{INST} iff $X_{j,t}$ is in $\text{Pa}(X_{i,t})$. While this framework neatly formalizes causal interactions, the general identifiability of any G from $\boldsymbol{X}$ requires a number of assumptions about Eq. (1). For time series, the direction of time Bauer et al. (2016) (effects cannot precede causes) aids in identifying lagged relationships (G^{WCG} and G^{SG}), generally requiring fewer restrictive assumptions. However, the recovery of G^{INST} is more challenging (and not tackled by all CD methods), resembling causal discovery from i.i.d. sample data. We refer to (Pearl, 2009; Peters et al., 2017) for a comprehensive introduction.

Despite the fact that many specific assumptions underpinning CD methods can be relaxed individually, a core set of strong, partly implicit, assumptions generally remains necessary to guarantee the identifiability of any SCM, as the causal hierarchy levels almost never collapse Bareinboim et al. (2022). Furthermore, even if these assumptions can be perfectly met in synthetic data, real-world data will have many assumptions violated, which can lead to performance degradation of CD algorithms Kaiser & Sipos (2021); Nastl & Hardt (2024).

On top, many assumptions are not verifiable without having access to the full SCM, e.g., the appropriate conditional-independence test Shah & Peters (2020). For widespread practical adoption, it is therefore essential to assess method performance under suboptimal conditions Poinsot et al. (2025). In response to these challenges, and mirroring trends in other machine learning domains, there is a growing emphasis on developing standardized benchmarks Cheng et al. (2023); Mogensen et al. (2024); Stein et al. (2024a); Herdeanu et al. (2025), or kits such as Muñoz-Marí et al. (2020) or Zhou et al. (2024) for CD. However, as aggregating extensive real-world causal ground truth is notoriously hard, alternative approaches were introduced to allow for the empirical evaluation of CD method performance. For instance, Schkoda et al. (2024) proposes leave-one-out cross-validation to assess the predictive performance of CD algorithms. Moreover, Machlanski et al. (2024) advocates for evaluating hyperparameter sensitivity, which has implications for method selection in practical applications. Closely related to our work, Yi et al. (2024) and Montagna et al. (2023a) test the performance of i.i.d. sample-based CD methods for fully violated assumptions. Further, Ferdous et al. (2025) provides an insightful study that investigate the impact of five different real-world complications on the performance of CD methods. Notably, robustness has also been explored in a general statistical context as well Rasch & Guiard (2004); Zimmerman (2014). Nevertheless, the impact of such violations remains under-investigated, particularly for time series data and for differences in violation severity. Finally, ensembling strategies, as a practical tool to improve robustness in other machine learning domains Arpit et al. (2022); Mienye & Sun (2022), likewise remain largely unexplored in the CD literature. While recent work has investigated ensembling over variable subsets to recover large graphs Wu et al. (2024), the potential to improve resilience with respect to assumption violations has not yet been studied.

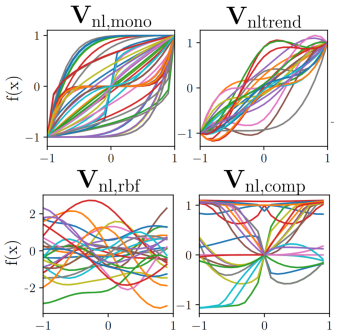
3 Stepwise Increasing Assumption Violations

While the CD literature increasingly explores the relaxation of certain assumptions, identifying any causal structure from $\boldsymbol{X}$ alone typically relies on a core set of assumptions to guarantee identifiability. Although prior work partially analyzes resilience to binary assumption violations Yi et al. (2024); Montagna et al. (2023a); Ferdous et al. (2025), a pertinent question arises: **How robust are certain CD methods against different severities of assumption violation?** This question is critical in applied settings. For instance, the mere existence of observational noise is less informative than understanding the corresponding robustness against its presence. Addressing this requires a framework for varying the severity of these violations. In this study, we introduce TCD-Arena for this purpose. In total, we implement 27 distinct assumption violations, each parameterized to allow for a stepwise increase of its severity. We individually describe these in the sections to come. Generally, we focus on covering commonly made assumptions Runge (2018) along with real-world complications that we believe are practically relevant. Further, we restrict our exploration to the following commonly assumed structure for Eq. (1):

$$X_{i,t} = \sum_{d=1}^{D} \sum_{l=0}^{L} A_{i,d,l} \cdot f_{i,d,l}(X_{d,t-l}) + \epsilon_{t,i}, \tag{2}$$

(a) In our experiments, the sources of randomness for the observational noise variables $\zeta_{i,t}$ are standard normally distributed $\big(\mathcal{N}(0,1)\big)$ random variables $\eta_{i,t}$ and η_t, which are consequently influenced by various factors, e.g., the signal strength ($\mathbf{V}_{\text{obs,mul}}$). Both α and β denote hyperparameters (details in **Apx. B.1**).

Violation	Definition of $\zeta_{i,t}$	Depends On
$\mathbf{V}_{\text{obs,add}}$	$\zeta_{i,t} = \eta_{i,t}$	—
$\mathbf{V}_{\text{obs,mul}}$	$\zeta_{i,t} = X_{i,t} \cdot \eta_{i,t}$	the signal $X_{i,t}$
$\mathbf{V}_{\text{obs,time}}$	$\zeta_{i,t} = \eta_{i,t} \cdot (1 + \alpha t) \cdot \sin(2\pi t/\beta)$	time step t
$\mathbf{V}_{\text{obs,auto}}$	$\zeta_{i,t} = \alpha \cdot \zeta_{i,t-1} + (1-\alpha) \cdot \eta_{i,t}$	autoregressive
$\mathbf{V}_{\text{obs,com}}$	$\zeta_{i,t} = \eta_t$ for all i at each t	—
$\mathbf{V}_{\text{obs,shock}}$	$\zeta_{i,t} \sim \begin{cases} S & \text{with prob. } p_{\text{shock}}, \\ 0 & \text{else.} \end{cases}$	fixed scalar S, shock prob. p_{shock}



(b) Note, coefficients $A_{i,d,l}$ can be negative, resulting in negative trends.

Figure 2: Details for violation types $\mathbf{V}_{\text{obs}}$ and $\mathbf{V}_{\text{nl}}$. Left: Observational noise violations. Right: Functional distributions that we deploy to sample $f_{i,d,l}$ used in Eq. (2).

where A specifies a coefficient matrix and $f_{i,d,l}$ an edge-specific univariate function and $\epsilon_{t,i}$ independent innovation noise. Crucially, any non-zero element in A denotes a corresponding edge in G. Further, for violations not concerning the causal mechanisms, $f_{i,d,l}$ is the identity function, and all interactions are linear. To help with clarity, we mark individual violations as $\mathbf{V}_{\text{type}}$. Finally, we keep the following violation descriptions brief and include a summary table, graphical depictions, specific violation step sizes, and detailed design choices for each violation in **Apx. A** and **Apx. B**.

Observational Noise ($\mathbf{V}_{\text{obs}}$) Many theoretical guarantees in causal discovery assume noise-free measurements, despite the fact that measurement errors are practically unavoidable and can introduce discrepancies that distort true causal relationships Scheines & Ramsey (2016). In an additive form, observation noise can be defined as: $\hat{X}_{i,t} = X_{i,t} + \zeta_{i,t}$, where $\zeta_{i,t}$ denotes observational noise. While standard independent additive noise ($\mathbf{V}_{\text{obs,add}}$) is prevalent, other noise types can occur depending on the measurement process. Due to this, we investigate the impact of five other types of observational noise structures on CD. **Fig. 2a** contains a concrete list. Additionally, details, hyperparameters, and discussions can be found in **Apx. B.1**. In particular, we include multiplicative, signal-dependent noise ($\mathbf{V}_{\text{obs,mul}}$) with real-world examples such as temperature sensors with lower precision at higher values Bentley (1984) or speckle noise in image processing Liu et al. (2014). We include time-dependent noise ($\mathbf{V}_{\text{obs,time}}$), which simulates cycles or linear sensor drift. Further, we model autoregressive noise structures ($\mathbf{V}_{\text{obs,auto}}$), i.e., disturbances of measurements which persist for multiple time steps. Similarly, we include common observational noise ($\mathbf{V}_{\text{obs,com}}$), where multiple variables are affected simultaneously, e.g., by weather events. Finally, we include shock noise ($\mathbf{V}_{\text{obs,shock}}$) to model infrequent events such as measurement failures. To systematically vary the level of intensity for any of these observational noise structures, we adjust the signal power of ζ to control the corresponding Signal-to-Noise Ratio (SNR) with respect to the data $\boldsymbol{X}$. To isolate the influence of the noise structure, we use the same, decreasing SNR levels for all observational noise violations in **Fig. 2a**.

Causal Sufficiency ($\mathbf{V}_{\text{conf}}$) Causal sufficiency posits that for any pair of observed variables X_i and X_j, there are no unobserved common causes (hidden confounders). That is, there is no unmeasured variable U such that $U \to X_i$ and $U \to X_j$. Such latent confounders can induce spurious correlations between observed variables, potentially leading to the inference of incorrect or misleading causal relationships. While some advanced methods aim to address specific types of confounding Trifunov et al. (2019); Chen et al. (2024); Li & Liu (2024), the presence of unmeasured confounders remains a major practical challenge, as it is rarely feasible to measure all relevant variables in complex systems. To simulate varying degrees of confounding and assess its impact, we employ two distinct strategies targeting lagged and contemporaneous confounding: First, concerning instantaneous confounding $\mathbf{V}_{\text{conf,inst}}$, we introduce a set of N exogenous variables $\mathcal{Z} = Z_1, \ldots, Z_N$, where each $Z_{n,t} \sim \mathcal{N}(0,1)$. These exogenous variables are not causally influenced by any variable in $\boldsymbol{X}$ but can act as common causes to multiple variables in step t. The severity of this type of confounding is controlled by progressively increasing the probability that an observed variable $X_{i,t}$ becomes dependent on any

of the exogenous variables Z_n. This, in turn, increases the probability for two variables in X to have a shared parent at t. Second, for lagged confounding, we introduce an additional variable, X_C, designated as the potential confounder ($\mathbf{V}_{\text{conf,lag}}$). This variable X_C is allowed to causally influence, and can be influenced by, other observed variables X_i with lagged effects up to a specified maximum lag L. The severity of confounding is controlled by stepwise increasing the probability that X_C is in the parent set of any other variable in $\boldsymbol{X}$, as well as the related coefficients in A. After sampling, the time series, X_C is removed from the observed data $\mathbf{X}$, rendering it a hidden confounder.

Faithfulness ($\mathbf{V}_{\text{faith}}$) Faithfulness asserts that all conditional independencies observed in the data are precisely those implied by d-separation of the DAG G Scheines (1997). Violations of faithfulness can lead to indistinguishable causal structures as they generate no dependencies in $\boldsymbol{X}$. Some works that examine this assumption and provide alternatives are (Zhang & Spirtes, 2008; Andersen, 2013; Lin & Zhang, 2020; Ng et al., 2021). Typically, unfaithfulness is implemented through causal structures like $X_{j,t} \to X_{i,t} \leftarrow X_{k,t} \leftarrow X_{j,t}$, where effects from $X_{j,t}$ to $X_{i,t}$ cancel out through appropriate parameter configurations in A. We implement this case for instantaneous effects ($\mathbf{V}_{\text{faith,inst}}$) as well as a lagged structure of the form $X_{j,t-2} \to X_{i,t} \leftarrow X_{k,t-1} \leftarrow X_{j,t-2}$ ($\mathbf{V}_{\text{faith,lag}}$). To stepwise scale the intensity of both violations, parameter configurations in A are updated to reach path cancellation. Further, as this is only one case of violating faithfulness Montagna et al. (2023c), we include a discussion concerning this design choice along with visual examples in **Apx. B.3**.

Functional Assumptions ($\mathbf{V}_{\text{nl}}$) While the general SCM in Eq. (2) is agnostic to the functional forms $f_{i,d,l}$ between variables $X_{d,t-l} \to X_{i,t}$, many discovery algorithms assume specific interactions, e.g., linear-additive relationships $X_t = \sum_{l=1}^{L} A_l \cdot X_{t-l} + \epsilon_t$ Hyvärinen et al. (2010); Pamfil et al. (2020). In real-world systems, such assumptions are often violated or are only approximations. Thus, it is crucial to study the consequences of corresponding violations, besides attempting to relax them Runge et al. (2019); Monti et al. (2020); Wu et al. (2022). To better emulate the variety found in practical scenarios and simulate data diversity, we employ a range of function generation techniques with different characteristics. In particular, we sample individual univariate functions $f_{i,d,l}$ from four distinct distributions: (1) Monotonic nonlinear functions ($\mathbf{V}_{\text{nl, mono}}$), (2) Non-monotonic functions with a linear trend ($\mathbf{V}_{\text{nl,trend}}$) (3) Gaussian processes with RBF kernels ($\mathbf{V}_{\text{nl,rbf}}$) following related robustness studies Montagna et al. (2023a); Yi et al. (2024), and (4) Random combinations of a set of base functions, e.g., $\sin(\cdot)$ or $e^{(\cdot)}$ ($\mathbf{V}_{\text{nl,comp}}$). Example functions are depicted in **Fig. 2b**, and we describe the exact distributions from which we sample in **Apx. B.4**. To stepwise increase the violations, we rely on two distinct procedures. First, for $\mathbf{V}_{\text{nl,mono}}$ and $\mathbf{V}_{\text{nl,trend}}$, we stepwise adapt the functional distributions such that sampled functions become on average increasingly nonlinear Emancipator & Kroll (1993) (see **Apx. B.4**). We sample all interactions $f_{i,d,l}$ in Eq. (2) from the corresponding distributions. Second, for $\mathbf{V}_{\text{nl,rbf}}$ and $\mathbf{V}_{\text{nl,comp}}$, we stepwise increase the probability of any $f_{i,d,l}$ to be drawn from the nonlinear distribution instead of being equal to the identity $f_{i,d,l}(\cdot) = \text{id}(\cdot)$.

Independent Innovation Noise ($\mathbf{V}_{\text{inno}}$) Independent additive innovation noise ($\epsilon_{i,t}$ in Eq. (2)) is crucial for causal discovery, as it ensures dependencies are attributed to causal links, not shared noise. However, this assumption is often violated in practice, as noise can incorporate unmeasured, dependent effects, and its true distribution is typically unknown or is fully deterministic Li et al. (2024). To evaluate how alternative innovation noise distributions might affect the performance of CD algorithms, we deploy the same five noise structures that we use for observational noise, i.e., $\mathbf{V}_{\text{inno,mul}}$,$\mathbf{V}_{\text{inno,auto}}$,$\mathbf{V}_{\text{inno,com}}$,$\mathbf{V}_{\text{inno,time}}$, and $\mathbf{V}_{\text{inno,shock}}$. However, for innovation noise scaling, the SNR (compare to the observation noise) is nontrivial as it is part of the signal itself. Therefore, we control the violations by blending standard normal noise with each of the five noise terms to stepwise move away from independent additive conditions (details in **Apx. B.5**). Notably, autoregressive $\mathbf{V}_{\text{inno,auto}}$ and common innovation noise $\mathbf{V}_{\text{inno,com}}$ fundamentally violate the Markov condition Peters et al. (2017).

Additionally, since some identifiability guarantees assume non-Gaussian noise (Shimizu et al., 2006), we test the effect of stepwise moving towards a non-Gaussian distribution. Specifically, we simulate this by starting from a Gaussian distribution (see **Apx. B.5**) and progressively shifting towards either a uniform ($\mathbf{V}_{\text{inno,uni}}$) or a Weibull distribution ($\mathbf{V}_{\text{inno,weib}}$). Finally, as some works rely on the assumption of equal noise variances, e.g., Peters & Bühlmann (2014), we implement a strategy to move away from this condition ($\mathbf{V}_{\text{inno,var}}$). For this, we draw $\epsilon_{i,t}$ from $\mathcal{N}(0, \sigma_i^2)$, where the variance σ_i^2 is individually sampled for each X_i. We then stepwise increase the corresponding ranges.

Stationarity ($\mathbf{V_{stat}}$) CD methods typically aim to uncover a single G from observations $\boldsymbol{X}$. Hence, a common assumption is that the SCM remains unchanged, i.e, stationary, over time or across regions. However, in many real-world scenarios, causal relationships can be heterogeneous across different populations or evolve over time Nastl & Hardt (2024). Here, works such as Huang et al. (2020); Günther et al. (2024); Ahmad et al. (2024) attempt to identify causal relationships in systems where parts of the SCM are changing. To simulate violations of stationarity, we keep the causal skeleton (nonzero elements in A) fixed but redraw the coefficients multiple times throughout the sampling process, violating the idea of causal consistency. Further, to stepwise scale the violation, we increase the number of times we resample A while generating $\boldsymbol{X}$.

Sufficient Sample Sizes ($\mathbf{V_{length}}$) Causal discovery algorithms necessitate a sufficient sample size to reliably detect patterns and estimate relationships Shen et al. (2020); Castelletti & Consonni (2024). For example, statistical tests used to identify conditional independencies may lack power with limited data Spirtes & Zhang (2016). To the best of our knowledge, no work has yet conducted an extensive study on the relationship between CD performance and sample size. To remedy this, we allow for a stepwise reduction of the length of the sampled time series $\boldsymbol{X}$ to model $\mathbf{V}_{\text{length}}$.

Data Quality ($\mathbf{V_q}$) To simulate measurement disturbances beyond observational noise, we introduce two types of quality degradations for $\boldsymbol{X}$. First, we model sensor failures ($\mathbf{V}_{\text{q,empty}}$) by setting all parent sets to $\varnothing$ for short periods, simulating false, zero-information measurements. We then stepwise increase the length of these periods to scale the effect. Second, for missing data ($\mathbf{V}_{\text{q,missing}}$), we remove an increasing number of samples completely at random Heitjan & Basu (1996) and fill the resulting NaNs via linear interpolation, a common approach for practitioners.

Data Scaling ($\mathbf{V_{scale}}$) Recent works show that synthetically generated data can introduce artifacts along the causal order that can be abused by CD methods Reisach et al. (2021); Kaiser & Sipos (2021); Ormaniec et al. (2025). By rescaling $\boldsymbol{X}$, these artifacts can be partly removed. To investigate how robust methods are against scaling, we allow for a stepwise scaling of the generated time series. In particular, we blend the original time series with its standardized version, a transformation that is reported to affect CD performance in Reisach et al. (2021); Kaiser & Sipos (2021).

Acyclicity and Sampling Rate Finally, we comment on the foundational DAG modeling assumption of acyclicity, which we deliberately do not address in this work. While central to many algorithms, this assumption can be violated in two primary ways: by genuine feedback loops inherent to the system (e.g., in differential equations), or by apparent cycles emerging as artifacts of temporal aggregation. The latter occurs when a coarse measurement resolution makes a cause and its lagged effect appear as a contemporaneous, bidirectional relationship Runge (2018). This creates a fundamental ambiguity: A system may be acyclic at one temporal scale but cyclic at another, making a single ground truth non-trivial to define.

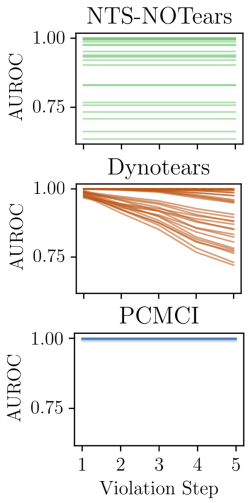
4 Experiments

To evaluate the robustness of CD methods and to showcase the functionality of TCD-Arena, we conducted a large empirical study using synthetic data across all previously described violations. For each violation, we systematically increase its intensity over five discrete levels. Our experiments covered a range of data conditions to ensure the generalizability of our findings. We vary the number of time steps ($T \in \{250, 1000\}$) and the number of variables D, together with the maximum causal lag L in the true SCM $\big((D, L) \in \{(5, 3), (7, 4)\}\big)$. For each setting, we generated datasets with both sparse and dense causal graphs, and both with and without instantaneous effects. This resulted in 8 distinct data-generating conditions which we call "data regimes". For each violation type, severity level, and data regime, we generated 100 independent structural causal models (SCMs) and a corresponding time series. In total, the evaluation for each violation type comprises 8,000 unique time series instances. Further details on the data-generating process are available in **Apx. C.1**.

Concerning CD methods, we run experiments on eight different strategies, including the direct Cross Correlation matrix to predict causal relationships as a baseline strategy. Further, we include the following seven approaches: We leverage Granger-causal ideas and deploy a vector autoregressive model Granger (1969), Varlingam Hyvärinen et al. (2010), PCMCI and PCMCI+ Runge et al. (2019),

(a) By ensembling different CD methods, general robustness can be improved. * marks required knowledge about the underlying SCM. † marks methods that do not discover G^{INST}. We highlight superior performance with **green** .

Method	G^{WCG}	G^{INST}	G^{SG}
Cross Correlation	.852±.07	†	.835±.08
CausalPretraining	.866±.08	†	.860±.09
GVAR	.917±.07	†	.904±.08
Varlingam	**.919±.07**	.674±.08	**.906±.08**
PCMCI	.912±.08	†	.897±.09
PCMCI+	.911±.08	.846±.1	.899±.09
Dynotears	.901±.08	**.847±.1**	.894±.08
NTS-NOTears	.906±.07	.68±.07	.889±.09
$\text{Ensemble}_{\text{Avg.}}$	.923±.07	†	.910±.08
$\text{Ensemble}_{\text{Linear}}$	.928±.07	†	.928±.07
$\text{Ensemble}_{\text{MLP}}$	**.943±.06**	†	**.943±.06**
$\text{Ensemble}_{\text{ConvMixer}}$	.930±.07	†	.930±.07
$\text{Ensemble}_{\text{Pareto}}$	.955*±.05	†	.955*±.05



(b) Each curve depicts AUROC changes of a particular hyperparameter configuration and a particular data regime.

Figure 3: Left: AUROC scoring of the best hyperparameter configuration per CD method and per ensemble for G^{WCG}, G^{INST}, and G^{SG}. Right: Hyperparameter variations with respect to $\mathbf{V}_{\text{scale}}$.

Dynotears (Pamfil et al., 2020), NTS-NOTears (Sun et al., 2023), and CausalPretraining (Stein et al., 2024b). With this, we cover all common CD paradigms Assaad et al. (2022). Under ideal linear conditions with no assumption violations, all included methods are capable of recovering G^{WCG} from Eq. (2). Additional details and a list of assumptions for each method are provided in **Apx. C.2**.

To ensure a fair and rigorous performance comparison, we adopt a evaluation protocol with three key components. First, to mitigate bias from suboptimal parameter choices, we perform an extensive hyperparameter search for each method. The full search spaces are detailed in **Apx. C.3**. This process also enables a secondary analysis of hyperparameter sensitivity. Second, we selected the Area Under the Receiver Operating Characteristic (AUROC) as our primary, threshold-independent performance metric. This allows us to evaluate how well a method distinguishes true causal links from non-dependence, without the need to select a specific decision threshold. For completeness, we also report and discuss alternative metrics in **Apx. D.1**. Third, to measure the robustness for a specific hyperparameter configuration of a method with respect to a violation $\mathbf{V}_{\text{type}}$, we average the AUROC scores for all data regimes and violation levels. This aggregation accounts for potential variations in the optimal decision boundary across different experimental conditions and provides a single, comparable score of robustness. To compare different CD methods, we report the hyperparameter configuration that achieves the highest average robustness over all violations as a representative. We include a visual overview of this experimental protocol in **Apx. A.4**.

Further, the severity levels for each violation were individually calibrated to span a range from negligible impact to a level where the baseline method's performance degrades to chance (if the violation type allows for it). Our analysis therefore focuses on the relative performance differences between methods for a specific violation, rather than comparing performance across different violation types. A complete list of the exact configurations and further discussion of this methodology are provided in **Apx. A.2**. Finally, because all methods were evaluated on the exact same datasets, any potential issues of theoretical non-identifiability affect all algorithms equally. This ensures a fair comparison of their relative robustness. Details on reproducibility can be found in **Apx. C.5**

In this section, we concentrate on three key findings extracted from our empirical study. (1) General robustness, (2) model misclassifications, and (3) hyperparameter sensitivity. To further contextualize these results, we provide additional discussions on specific violations in **Apx. D.3**.

First, we illustrate the robustness of each method against individual violation types for lagged effects and for instantaneous effects in **Fig. 1**. Furthermore, **Fig. 3a** summarizes the average robustness scores

across all assessed violations. Considering the discovery of lagged effects (G^{WCG}), we find that simple Granger-based approaches (GVAR and Varlingam) have slightly increased robustness. Interestingly, this is consistent with results reported in CausalRivers Stein et al. (2024a), a large-scale real-world benchmark. Further, we find that deep learning approaches (CausalPretraining and NTS-NOTears) lag behind approaches with much fewer parameters. Especially, CausalPretraining often performs close to our baseline, suggesting that it largely relies on correlational patterns. Concerning uncovering G^{INST} (**Fig. 1**), we find that Dynotears and PCMCI+ show comparable robustness (**Fig. 3a**) while Varlingam lags behind. Further, we find that there are larger differences in robustness. We denote this to the fact it is generally harder to uncover G^{INST}. Next, as we do not explicitly generate data with non-Gaussian additive noise, these results are consistent with theoretical constraints. Further, the large performance gap between G^{WCG} and G^{INST} can be explained by the fact that the non-gaussian assumption is specifically made to uncover the instantanous links and is not necessary for uncovering lagged effects. Interestingly, even for innovation noise violations where we stepwise increase a non-Gaussian additive noise component, i.e., ($\mathbf{V}_{\text{weib}}$ and $\mathbf{V}_{\text{uni}}$), Varlingam shows no superior robustness.

Second, as we previously assumed a known maximum lag L, we further investigate the performance of all tested algorithms under two additional scenarios: (i) The model is allowed to search for causes up to a lag greater than the true maximum lag ($L \in \{3, 4\}$ while $L_{\text{model}} \in \{5, 6\}$. (ii) The model's search space is restricted to lags shorter than the true maximum lag ($L \in \{3, 4\}$ while $L_{\text{model}} \in \{1, 2\}$). We denote these cases with $\uparrow L$ and $\downarrow L$, and report the effect of these misspecifications in **Table 1** (additional details in **Apx. D.4**). For the $\downarrow L$ condition, we observe strongly reduced performance across the board. However, we also find that methods able to discover instantaneous effects (with the exception of PCMCI+) have a noticeably lower performance deterioration. While we have no direct explanation for this phenomenon, we hypothesize that the estimation of instantaneous links might help with catching additional lagged effects that should be attributed to G^{WCG}. Importantly, a formal theoretical analysis is needed to further understand this phenomenon and its implications. For the $\uparrow L$ regime, we find that performance is rather robust. While such results on synthetic data should be treated with caution, this observation suggests that in practice, choosing a larger-than-necessary max lag L_{model} can be beneficial. Finally, we note that CausalPretraining has the distinct advantage that L_{model} does not have to be specified, which explains the robustness in the $\downarrow L$ regimes.

Third, note the fact that correct hyperparameter specifications cannot be directly selected in real-world applications, we report the average robustness over all hyperparameters in **Table 1** and examine a particular interesting example of hyperparameter influence with **Fig. 2b**. While the robustness of all CD methods reduces when reporting the average over all hyperparameters, we find that the methods with more hyperparameters (Dynotears and NTS-NOTears) have a noticeably higher reduction and standard deviation between configurations, but also show no superior robustness under optimal hyperparameters (**Table 1**). Additionally, **Fig. 2b** shows that various CD methods can have drastically differing hyperparameter sensitivities with respect to a violation. While we find that the optimal hyperparameters have a high robustness against $\mathbf{V}_{\text{scale}}$ in all cases (**Fig. 1**), other configurations that show similar performance at the beginning, degrade much faster with violation strength (Dynotears) or show high robustness but also high variance between hyperparameter performances (NTS-NOTears). For completeness, we include the visualizations of the remaining methods and violations in **Apx. D.5**.

To conclude, we find various empirical differences between CD methods that raise the question of whether a combination of multiple CD methods can improve general robustness. We investigate this question in the next section.

4.1 Ensembling CD to Improve Robustness

Given the observed variability in robustness across different individual CD methods, we investigate the potential of ensembling techniques to achieve improved general robustness. While ensembling is a cornerstone of modern machine learning Arpit et al. (2022); Mienye & Sun (2022), its potential to enhance the robustness of time series CD methods remains unexplored in the literature. To remedy this, we learn a meta model that predicts the causal graph G based on the collection of predicted graphs $\{\hat{G}_1, \ldots, \hat{G}_M\}$ from M individual base CD methods. Specifically, we investigate a linear combination $\text{Ensemble}_{\text{Linear}}$, a simple MLP $\text{Ensemble}_{\text{MLP}}$, and a ConvMixer architecture Trockman & Zico Kolter (2022) $\text{Ensemble}_{\text{ConvMixer}}$. To train these meta-learners, we generate an additional, independent training dataset containing time series samples for all violations and data regimes. The exact training procedure is contained in **Apx. C.4**. Additionally, we report the performance of a

Table 1: Average robustness under wrongly specified L ($\downarrow L$ denotes too low, $\uparrow L$ denotes too high) for G^{WCG} and G^{SG}. In parentheses, we include the change from a correctly specified L. Further, we report the average hyperparameter performance. As CausalPretraining (*) does not require the specification of a max lag, its performance for the $\downarrow L$ regime is superior. As Cross Corr. has no hyperparameters, it has no standard deviation (†). We mark superior performance with **green** .

Method	G^{WCG}		G^{SG}		HP
	$\downarrow L$	$\uparrow L$	$\downarrow L$	$\uparrow L$	Avg.
Cross Corr.	.571(-.28)	.855(+.00)	.686(-.15)	.827(-.01)	.852± †
CausalPretr.	.866*(+.00)	.866(+.00)	.860*(+.00)	.860(+.00)	.863±.00
GVAR	.582(-.33)	.914(-.00)	**.727(-.18)**	.892(-.01)	**.913±.01**
Varlingam	**.649(-.27)**	**.917(-.00)**	**.727(-.18)**	**.895(-.01)**	.907±.02
PCMCI	.583(-.33)	**.913(+.00)**	.708(-.19)	.890(-.01)	.911±.00
PCMCI+	.583(-.33)	.912(+.00)	.708(-.19)	.892(-.01)	.898±.01
Dynotears	.637(-.26)	.902(+.00)	.711(-.18)	.889(-.01)	.855±.05
NTS-NOTears	.620(-.29)	.907(+.00)	.710(-.18)	.878(-.01)	.827±.09

simple unweighted averaging of all $\hat{G}_m$ ($\text{Ensemble}_{\text{Avg.}}$) and an oracle strategy that comprises the Pareto front, we call $\text{Ensemble}_{\text{Pareto}}$. Explicitly, for any given assumption violation, $\text{Ensemble}_{\text{Pareto}}$ selects the output from the CD method that achieves the highest measured robustness on that specific violation. While not practically attainable, it serves as a baseline, indicating the maximum potential performance gain achievable by perfectly selecting among the outputs of the base methods. All ensembling strategies are evaluated on the original test datasets used for all other experiments in this paper, ensuring a fair comparison. We report the average performance of these ensembling approaches in **Fig. 3a** and provide additional analysis of performance gains in **Apx. D.6**.

We find that all ensembling approaches are able to improve the robustness over any individual method. Specifically, $\text{Ensemble}_{\text{MLP}}$ leads to notable increases while also reducing the standard deviation between different violations. Further, our oracle $\text{Ensemble}_{\text{Pareto}}$ achieves the highest performance, highlighting the fact that various CD methods do have distinct advantages concerning robustness against particular assumption violations. Note, as transitioning this approach to real-world scenarios will require addressing challenges such as domain adaptation and disttributional shifts, we present these results as a theoretical proof-of-concept. They however suggest that ensembling is a promising strategy for enhancing the robustness and reliability of CD methods in complicated data settings. Especially, when considering that the here presented ensembles have no direct access to $\boldsymbol{X}$.

5 Conclusion

This study demonstrates the first extensive empirical investigation into the robustness with respect to assumption violations of Causal Discovery (CD) methods for time series data. We implement 27 distinct assumption violation scenarios, inspired by real-world data complexities, to evaluate eight distinct CD algorithms. Our large-scale study revealed notable variability in how different methods respond to these violations. In particular, we first quantify general robustness over all violations, then analyze how model misspecification affects performance, and finally investigate general hyperparameter sensitivities. Motivated by these differences, we investigate the ensembling of CD methods and conclude that they can improve general robustness. Our study is supported by TCD-Arena, an empirical framework and testing kit for time series CD that we developed to conduct all of our experiments. While our investigation aimed to cover a broad spectrum of frequently made assumptions and practical challenges, the landscape of potential data-generating processes is vast. Hence, we are releasing TCD-Arena as an open-source, modular package to facilitate future extensions and foster long-term comparability. Further, to encourage community engagement, we maintain a live list of all implemented violations on our project page[1] and provide a reproducability statement in **Apx. C.5**. With this, we ultimately aim to support a deeper understanding of causal discovery methods with respect to their strengths and weaknesses under various synthetic but diverse conditions, paving the way for more robust real-world applications.

[1]TCD-Arena(anonymized Git)

REFERENCES

Wasim Ahmad, Maha Shadaydeh, and Joachim Denzler. Regime Identification for Improving Causal Analysis in Non-stationary Timeseries, April 2024. URL http://arxiv.org/abs/2405.02315. arXiv:2405.02315 [stat].

Stanley Ainsworth. Michaelis-Menten Kinetics. In Stanley Ainsworth (ed.), *Steady-State Enzyme Kinetics*, pp. 43–73. Macmillan Education UK, London, 1977. ISBN 9781349019595. doi: 10.1007/978-1-349-01959-5_3. URL https://doi.org/10.1007/978-1-349-01959-5_3.

Holly Andersen. When to Expect Violations of Causal Faithfulness and Why It Matters. *Philosophy of Science*, 80(5):672–683, 2013. doi: 10.1086/673937.

Devansh Arpit, Huan Wang, Yingbo Zhou, and Caiming Xiong. Ensemble of Averages: Improving Model Selection and Boosting Performance in Domain Generalization. *Advances in Neural Information Processing Systems*, 35:8265–8277, December 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/372cb7805eaccb2b7eed641271a30eec-Abstract-Conference.html.

Charles K. Assaad, Emilie Devijver, and Eric Gaussier. Survey and Evaluation of Causal Discovery Methods for Time Series. *Journal of Artificial Intelligence Research*, 73:767–819, February 2022. ISSN 1076-9757. doi: 10.1613/jair.1.13428. URL https://www.jair.org/index.php/jair/article/view/13428.

Elias Bareinboim, Juan D. Correa, Duligur Ibeling, and Thomas Icard. On Pearl's Hierarchy and the Foundations of Causal Inference. In *Probabilistic and Causal Inference: The Works of Judea Pearl*, volume 36, pp. 507–556. Association for Computing Machinery, New York, NY, USA, 1 edition, March 2022. ISBN 9781450395861. URL https://doi.org/10.1145/3501714.3501743.

Stefan Bauer, Bernhard Schölkopf, and Jonas Peters. The Arrow of Time in Multivariate Time Series. In *Proceedings of The 33rd International Conference on Machine Learning*, pp. 2043–2051. PMLR, June 2016. URL https://proceedings.mlr.press/v48/bauer16.html.

J. P. Bentley. Temperature sensor characteristics and measurement system design. *Journal of Physics E: Scientific Instruments*, 17(6):430, June 1984. ISSN 0022-3735. doi: 10.1088/0022-3735/17/6/002. URL https://dx.doi.org/10.1088/0022-3735/17/6/002.

Philippe Brouillard, Chandler Squires, Jonas Wahl, Konrad P. Kording, Karen Sachs, Alexandre Drouin, and Dhanya Sridhar. The Landscape of Causal Discovery Data: Grounding Causal Discovery in Real-World Applications, December 2024. URL http://arxiv.org/abs/2412.01953. arXiv:2412.01953 [cs].

Federico Castelletti and Guido Consonni. Bayesian Sample Size Determination for Causal Discovery. *Statistical Science*, 39(2):305–321, May 2024. ISSN 0883-4237, 2168-8745. doi: 10.1214/23-STS905. URL https://projecteuclid.org/journals/statistical-science/volume-39/issue-2/Bayesian-Sample-Size-Determination-for-Causal-Discovery/10.1214/23-STS905.full.

Li Chen, Chunlin Li, Xiaotong Shen, and Wei Pan. Discovery and Inference of a Causal Network with Hidden Confounding. *Journal of the American Statistical Association*, 119(548):2572–2584, October 2024. ISSN 0162-1459. doi: 10.1080/01621459.2023.2261658. URL https://doi.org/10.1080/01621459.2023.2261658.

Yuxiao Cheng, Ziqian Wang, Tingxiong Xiao, Qin Zhong, Jinli Suo, and Kunlun He. CausalTime: Realistically Generated Time-series for Benchmarking of Causal Discovery. October 2023. URL https://openreview.net/forum?id=iad1yyyGme.

Carl de Boor. *A Practical Guide to Splines*, volume 27 of *Applied Mathematical Sciences*. Springer, 2001. Revised Edition.

Kenneth Emancipator and Martin H. Kroll. A quantitative measure of nonlinearity. *Clinical Chemistry*, 39(5):766–772, 05 1993. ISSN 0009-9147. doi: 10.1093/clinchem/39.5.766. URL https://doi.org/10.1093/clinchem/39.5.766.

Philipp M. Faller, Leena C. Vankadara, Atalanti A. Mastakouri, Francesco Locatello, and Dominik Janzing. Self-Compatibility: Evaluating Causal Discovery without Ground Truth. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, pp. 4132–4140. PMLR, April 2024. URL https://proceedings.mlr.press/v238/faller24a.html.

Muhammad Hasan Ferdous, Emam Hossain, and Md Osman Gani. TimeGraph: Synthetic Benchmark Datasets for Robust Time-Series Causal Discovery. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pp. 5425–5435, August 2025. doi: 10.1145/3711896.3737439. URL http://arxiv.org/abs/2506.01361. arXiv:2506.01361 [cs].

Clark Glymour, Kun Zhang, and Peter Spirtes. Review of Causal Discovery Methods Based on Graphical Models. *Frontiers in Genetics*, 10, June 2019. ISSN 1664-8021. doi: 10.3389/fgene.2019.00524. URL https://www.frontiersin.org/journals/genetics/articles/10.3389/fgene.2019.00524/full.

C. W. J. Granger. Investigating Causal Relations by Econometric Models and Cross-spectral Methods. *Econometrica*, 37(3):424–438, 1969. ISSN 0012-9682. doi: 10.2307/1912791. URL https://www.jstor.org/stable/1912791.

Wiebke Günther, Oana-Iuliana Popescu, Martin Rabel, Urmi Ninad, Andreas Gerhardus, and Jakob Runge. Causal discovery with endogenous context variables. *Advances in Neural Information Processing Systems*, 37:36243–36284, December 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/3fcce87e6df22b2ab6f0be68af3ec714-Abstract-Conference.html.

Daniel F. Heitjan and Srabashi Basu. Distinguishing "Missing at Random" and "Missing Completely at Random". *The American Statistician*, 50(3):207–213, 1996. doi: 10.1080/00031305.1996.10474381. URL https://doi.org/10.1080/00031305.1996.10474381.

Benjamin Herdeanu, Juan Nathaniel, Carla Roesch, Jatan Buch, Gregor Ramien, Johannes Haux, and Pierre Gentine. CausalDynamics: A large-scale benchmark for structural discovery of dynamical causal models, May 2025. URL http://arxiv.org/abs/2505.16620. arXiv:2505.16620 [cs].

Biwei Huang, Kun Zhang, Jiji Zhang, Joseph Ramsey, Ruben Sanchez-Romero, Clark Glymour, and Bernhard Schölkopf. Causal Discovery from Heterogeneous/Nonstationary Data. *Journal of Machine Learning Research*, 21(89):1–53, 2020. ISSN 1533-7928. URL http://jmlr.org/papers/v21/19-232.html.

Aapo Hyvärinen, Kun Zhang, Shohei Shimizu, and Patrik O. Hoyer. Estimation of a Structural Vector Autoregression Model Using Non-Gaussianity. *Journal of Machine Learning Research*, 11(56):1709–1731, 2010. ISSN 1533-7928. URL http://jmlr.org/papers/v11/hyvarinen10a.html.

Marcus Kaiser and Maksim Sipos. Unsuitability of NOTEARS for Causal Graph Discovery, June 2021. URL http://arxiv.org/abs/2104.05441. arXiv:2104.05441 [cs, math, stat].

William Ogilvy Kermack, A. G. McKendrick, and Gilbert Thomas Walker. A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 115(772):700–721, January 1997. doi: 10.1098/rspa.1927.0118. URL https://royalsocietypublishing.org/doi/10.1098/rspa.1927.0118.

Loka Li, Haoyue Dai, Hanin Al Ghothani, Biwei Huang, Jiji Zhang, Shahar Harel, Isaac Bentwich, Guangyi Chen, and Kun Zhang. On Causal Discovery in the Presence of Deterministic Relations. *Advances in Neural Information Processing Systems*, 37:130920–130952, December 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/ec52572b9e16b91edff5dc70e2642240-Abstract-Conference.html.

Xiu-Chuan Li and Tongliang Liu. Efficient and Trustworthy Causal Discovery with Latent Variables and Complex Relations. October 2024. URL https://openreview.net/forum?id=BZYIEw4mcY.

Hanti Lin and Jiji Zhang. On Learning Causal Structures from Non-Experimental Data without Any Faithfulness Assumption. In *Proceedings of the 31st International Conference on Algorithmic Learning Theory*, pp. 554–582. PMLR, January 2020. URL https://proceedings.mlr.press/v117/lin20a.html.

Xinhao Liu, Masayuki Tanaka, and Masatoshi Okutomi. Practical Signal-Dependent Noise Parameter Estimation From a Single Noisy Image. *IEEE Transactions on Image Processing*, 23(10):4361–4371, October 2014. ISSN 1941-0042. doi: 10.1109/TIP.2014.2347204. URL https://ieeexplore.ieee.org/document/6876183.

Damian Machlanski, Spyridon Samothrakis, and Paul S. Clarke. Robustness of Algorithms for Causal Structure Learning to Hyperparameter Choice. In *Proceedings of the Third Conference on Causal Learning and Reasoning*, pp. 703–739. PMLR, March 2024. URL https://proceedings.mlr.press/v236/machlanski24a.html.

Ibomoiye Domor Mienye and Yanxia Sun. A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects. *IEEE Access*, 10:99129–99149, 2022. ISSN 2169-3536. doi: 10.1109/ACCESS.2022.3207287. URL https://ieeexplore.ieee.org/abstract/document/9893798/citations.

Søren Wengel Mogensen, Karin Rathsman, and Per Nilsson. Causal discovery in a complex industrial system: A time series benchmark. In *Proceedings of the Third Conference on Causal Learning and Reasoning*, pp. 1218–1236. PMLR, March 2024. URL https://proceedings.mlr.press/v236/mogensen24a.html.

Francesco Montagna, Atalanti Mastakouri, Elias Eulig, Nicoletta Noceti, Lorenzo Rosasco, Dominik Janzing, Bryon Aragam, and Francesco Locatello. Assumption violations in causal discovery and the robustness of score matching. *Advances in Neural Information Processing Systems*, 36:47339–47378, December 2023a. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/93ed74938a54a73b5e4c52bbaf42ca8e-Abstract-Conference.html.

Francesco Montagna, Atalanti Mastakouri, Elias Eulig, Nicoletta Noceti, Lorenzo Rosasco, Dominik Janzing, Bryon Aragam, and Francesco Locatello. Assumption violations in causal discovery and the robustness of score matching. *Advances in Neural Information Processing Systems*, 36:47339–47378, December 2023b. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/93ed74938a54a73b5e4c52bbaf42ca8e-Abstract-Conference.html.

Francesco Montagna, Nicoletta Noceti, Lorenzo Rosasco, Kun Zhang, and Francesco Locatello. Causal Discovery with Score Matching on Additive Models with Arbitrary Noise. March 2023c. URL https://openreview.net/forum?id=rVO0Bx90deu.

Ricardo Pio Monti, Kun Zhang, and Aapo Hyvärinen. Causal Discovery with General Non-Linear Relationships using Non-Linear ICA. In *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference*, pp. 186–195. PMLR, August 2020. URL https://proceedings.mlr.press/v115/monti20a.html.

J. Muñoz-Marí, G. Mateo, J. Runge, and G. Camps-Valls. CauseMe: An online system for benchmarking causal discovery methods., 2020. In preparation (2020).

Vivian Y. Nastl and Moritz Hardt. Do causal predictors generalize better to new domains? *Advances in Neural Information Processing Systems*, 37:31202–31315, December 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/3792ddbf94b68ff4369f510f7a3e1777-Abstract-Conference.html.

Brady Neal, Chin-Wei Huang, and Sunand Raghupathi. RealCause: Realistic Causal Inference Benchmarking. May 2023. URL https://openreview.net/forum?id=m28E5RN64hi.

Ignavier Ng, Yujia Zheng, Jiji Zhang, and Kun Zhang. Reliable Causal Discovery with Improved Exact Search and Weaker Assumptions. In *Advances in Neural Information Processing Systems*, volume 34, pp. 20308–20320. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/hash/a9b4ec2eb4ab7b1b9c3392bb5388119d-Abstract.html.

Weronika Ormaniec, Scott Sussex, Lars Lorch, Bernhard Schölkopf, and Andreas Krause. Standardizing Structural Causal Models, March 2025. URL http://arxiv.org/abs/2406.11601. arXiv:2406.11601 [cs].

Roxana Pamfil, Nisara Sriwattanaworachai, Shaan Desai, Philip Pilgerstorfer, Konstantinos Georgatzis, Paul Beaumont, and Bryon Aragam. DYNOTEARS: Structure Learning from Time-Series Data. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, pp. 1595–1605. PMLR, June 2020. URL https://proceedings.mlr.press/v108/pamfil20a.html.

Judea Pearl. Causal inference in statistics: An overview. *Statistics Surveys*, 3 (none):96–146, January 2009. ISSN 1935-7516. doi: 10.1214/09-SS057. URL https://projecteuclid.org/journals/statistics-surveys/volume-3/issue-none/Causal-inference-in-statistics-An-overview/10.1214/09-SS057.full.

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

J. Peters and P. Bühlmann. Identifiability of Gaussian structural equation models with equal error variances. *Biometrika*, 101(1):219–228, March 2014. ISSN 0006-3444. doi: 10.1093/biomet/ast043. URL https://doi.org/10.1093/biomet/ast043.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

Audrey Poinsot, Panayiotis Panayiotou, Alessandro Leite, Nicolas CHESNEAU, Özgür Şimşek, and Marc Schoenauer. Position: Causal machine learning requires rigorous synthetic experiments for broader adoption. In *Forty-second International Conference on Machine Learning Position Paper Track*, 2025.

Murray H Protter and Charles B Morrey. *Intermediate Calculus*. Springer, 1985.

J.O. Ramsay and B.W. Silverman. *Functional Data Analysis*. Springer, 2005.

Dieter Rasch and Volker Guiard. The robustness of parametric statistical methods. *Psychology Science*, 46:175–208, 2004.

Alexander Reisach, Christof Seiler, and Sebastian Weichwald. Beware of the Simulated DAG! Causal Discovery Benchmarks May Be Easy to Game. In *Advances in Neural Information Processing Systems*, volume 34, pp. 27772–27784. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/hash/e987eff4a7c7b7e580d659feb6f60c1a-Abstract.html.

J. Runge. Causal network reconstruction from time series: From theoretical assumptions to practical estimation. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 28(7):075310, July 2018. ISSN 1054-1500. doi: 10.1063/1.5025050. URL https://doi.org/10.1063/1.5025050.

Jakob Runge, Peer Nowack, Marlene Kretschmer, Seth Flaxman, and Dino Sejdinovic. Detecting causal associations in large nonlinear time series datasets. *Science Advances*, 5(11):eaau4996, November 2019. ISSN 2375-2548. doi: 10.1126/sciadv.aau4996. URL http://arxiv.org/abs/1702.07007. arXiv:1702.07007 [physics, stat].

Richard Scheines. An introduction to causal inference. 1997.

Richard Scheines and Joseph Ramsey. Measurement Error and Causal Discovery. *CEUR workshop proceedings*, 1792:1–7, June 2016. ISSN 1613-0073. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5340263/.

Daniela Schkoda, Philipp Faller, Patrick Blöbaum, and Dominik Janzing. Cross-validating causal discovery via Leave-One-Variable-Out, November 2024. URL http://arxiv.org/abs/2411.05625. arXiv:2411.05625 [stat].

Rajen D. Shah and Jonas Peters. The Hardness of Conditional Independence Testing and the Generalised Covariance Measure. *The Annals of Statistics*, 48(3), June 2020. ISSN 0090-5364. doi: 10.1214/19-AOS1857. URL http://arxiv.org/abs/1804.07203. arXiv:1804.07203 [math, stat].

Xinpeng Shen, Sisi Ma, Prashanthi Vemuri, and Gyorgy Simon. Challenges and Opportunities with Causal Discovery Algorithms: Application to Alzheimer's Pathophysiology. *Scientific Reports*, 10(1):2975, February 2020. ISSN 2045-2322. doi: 10.1038/s41598-020-59669-x. URL https://www.nature.com/articles/s41598-020-59669-x.

Shohei Shimizu, Patrik O. Hoyer, Aapo Hyv&#228, rinen, and Antti Kerminen. A Linear Non-Gaussian Acyclic Model for Causal Discovery. *Journal of Machine Learning Research*, 7(72): 2003–2030, 2006. ISSN 1533-7928. URL http://jmlr.org/papers/v7/shimizu06a.html.

Peter Spirtes and Kun Zhang. Causal discovery and inference: concepts and recent methodological advances. *Applied Informatics*, 3(1):3, February 2016. ISSN 2196-0089. doi: 10.1186/s40535-016-0018-x. URL https://doi.org/10.1186/s40535-016-0018-x.

Gideon Stein, Maha Shadaydeh, Jan Blunk, Niklas Penzel, and Joachim Denzler. CausalRivers - Scaling up benchmarking of causal discovery for real-world time-series. In *Proceedings of the Thirteenth International Conference on Learning Representations*, October 2024a. URL https://openreview.net/forum?id=wmV4cIbgl6.

Gideon Stein, Maha Shadaydeh, and Joachim Denzler. Embracing the black box: Heading towards foundation models for causal discovery from time series data, February 2024b. URL http://arxiv.org/abs/2402.09305. arXiv:2402.09305 [cs].

Xiangyu Sun, Oliver Schulte, Guiliang Liu, and Pascal Poupart. NTS-NOTEARS: Learning Nonparametric DBNs With Prior Knowledge. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, pp. 1942–1964. PMLR, April 2023. URL https://proceedings.mlr.press/v206/sun23c.html.

Gionatan Torricelli, Fabrizio Argenti, and Luciano Alparone. Modelling and assessment of signal-dependent noise for image de-noising. In *2002 11th European Signal Processing Conference*, pp. 1–4, 2002.

Violeta Teodora Trifunov, Maha Shadaydeh, Jakob Runge, Veronika Eyring, Markus Reichstein, and Joachim Denzler. Nonlinear Causal Link Estimation Under Hidden Confounding with an Application to Time Series Anomaly Detection. In Gernot A. Fink, Simone Frintrop, and Xiaoyi Jiang (eds.), *Pattern Recognition*, pp. 261–273, Cham, 2019. Springer International Publishing. ISBN 9783030336769. doi: 10.1007/978-3-030-33676-9_18.

Asher Trockman and J. Zico Kolter. Patches Are All You Need?, January 2022. URL https://ui.adsabs.harvard.edu/abs/2022arXiv220109792T. ADS Bibcode: 2022arXiv220109792T.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Zeyu Wang. CausalBench: A Comprehensive Benchmark for Evaluating Causal Reasoning Capabilities of Large Language Models. pp. 143–151, August 2024. URL https://aclanthology.org/2024.sighan-1.17/.

W Weibull. Astatistical theory ofthe strength of materials. *Proc. Royal 4cademy Engrg Science*, 15 (1):1, 1939.

Menghua Wu, Yujia Bao, Regina Barzilay, and Tommi Jaakkola. Sample, estimate, aggregate: A recipe for causal discovery foundation models, May 2024. URL http://arxiv.org/abs/2402.01929. arXiv:2402.01929 [cs].

Tianhao Wu, Xingyu Wu, Xin Wang, Shikang Liu, and Huanhuan Chen. Nonlinear Causal Discovery in Time Series. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, CIKM '22, pp. 4575–4579, New York, NY, USA, October 2022. Association for Computing Machinery. ISBN 9781450392365. doi: 10.1145/3511808.3557660. URL https://dl.acm.org/doi/10.1145/3511808.3557660.

Huiyang Yi, Yanyan He, Duxin Chen, Mingyu Kang, He Wang, and Wenwu Yu. THE ROBUSTNESS OF DIFFERENTIABLE CAUSAL DISCOVERY IN MISSPECIFIED SCENARIOS. October 2024. URL https://openreview.net/forum?id=iaP7yHRq1l.

Jiji Zhang and Peter Spirtes. Detection of Unfaithfulness and Robust Causal Inference. *Minds and Machines*, 18(2):239–271, June 2008. ISSN 1572-8641. doi: 10.1007/s11023-008-9096-4. URL https://doi.org/10.1007/s11023-008-9096-4.

Wei Zhou, Hong Huang, Guowen Zhang, Ruize Shi, Kehan Yin, Yuanyuan Lin, and Bang Liu. OCDB: Revisiting Causal Discovery with a Comprehensive Benchmark and Evaluation Framework, June 2024. URL http://arxiv.org/abs/2406.04598. arXiv:2406.04598 [cs].

Donald W. Zimmerman. Robust Statistical Tests. In Alex C. Michalos (ed.), *Encyclopedia of Quality of Life and Well-Being Research*, pp. 5574–5587. Springer Netherlands, Dordrecht, 2014. ISBN 9789400707535. doi: 10.1007/978-94-007-0753-5_2529. URL https://doi.org/10.1007/978-94-007-0753-5_2529.

Contents

A Appendix — High Level Overview

A.1 Violation List

Table 2 contains a brief overview of all 27 violations contained in TCD-Arena and investigated empirically in our main paper. Further details on the implementation and evaluated severity levels can be found in the following chapters.

Violation	Short Description
$\mathbf{V}_{\text{obs,add}}$	Additive measurement noise.
$\mathbf{V}_{\text{obs,mul}}$	Signal dependent measurement noise.
$\mathbf{V}_{\text{obs,time}}$	Time-varying measurement noise.
$\mathbf{V}_{\text{obs,auto}}$	Autoregressive measurement noise.
$\mathbf{V}_{\text{obs,com}}$	Common source measurement noise.
$\mathbf{V}_{\text{obs,shock}}$	Spike measurement noise.
$\mathbf{V}_{\text{conf,inst}}$	Unseen internal common cause.
$\mathbf{V}_{\text{conf,lag}}$	Unseen external common causes.
$\mathbf{V}_{\text{faith,inst}}$	Instantaneous effects cancel out.
$\mathbf{V}_{\text{faith,lag}}$	Lagged effects cancel out.
$\mathbf{V}_{\text{nl,mono}}$	Monotonic functions.
$\mathbf{V}_{\text{nl,trend}}$	B-spline functions with a linear trend.
$\mathbf{V}_{\text{nl,rbf}}$	GP-RBF functions.
$\mathbf{V}_{\text{nl,comp}}$	Composite functions.
$\mathbf{V}_{\text{inno,mul}}$	Signal dependent innovation noise.
$\mathbf{V}_{\text{inno,time}}$	Time-dependent innovation noise.
$\mathbf{V}_{\text{inno,auto}}$	Autoregressive innovation noise.
$\mathbf{V}_{\text{inno,com}}$	Common source innovation noise.
$\mathbf{V}_{\text{inno,shock}}$	Spike innovation noise.
$\mathbf{V}_{\text{inno,uni}}$	Uniform additive innovation noise.
$\mathbf{V}_{\text{inno,weib}}$	Weibull additive innovation noise.
$\mathbf{V}_{\text{inno,var}}$	Unequal variances in innovation noise.
$\mathbf{V}_{\text{stat}}$	Causal link strengths change over time.
$\mathbf{V}_{\text{length}}$	Reduced time series length.
$\mathbf{V}_{\text{q,empty}}$	Temporary complete loss of causal signal.
$\mathbf{V}_{\text{q,missing}}$	Missing data points (interpolated).
$\mathbf{V}_{\text{scale}}$	Data standardization.

Table 2: List of all 27 violations contained in TCD-Arena with corresponding short descriptions.

A.2 Violation Steps

Table 3 contains a list of parameter values used to intensify all 27 violations contained in TCD-Arena and included in our study. Additionally, we note a short description of how each violation is scaled. We refer to Apx. B for more in-depth descriptions. The experimental violations were individually configured to establish a relevant performance range for evaluating Causal Discovery (CD) methods. Recognizing that the disruptive impact of each violation type varies considerably, a standardized approach was not employed. Instead, for each violation, the parameters were calibrated according to a three-step procedure. First, a parameter range was identified that induced a significant performance degradation for a baseline Cross-Correlation (CC) method. Second, where the violation type allowed, the maximum intensity was set to reduce the CC's performance to an Area Under the Receiver Operating Characteristic (AUROC) of approximately 0.5. Third, discrete levels of the violation were established by equally spacing them between a minimal-effect level and the determined maximum. This methodology ensures that various CD methods can be effectively compared against each other

within a challenging and relevant operational range for each specific violation, although it precludes direct performance comparisons between different violation types. We document this process in TCD-Arena for maximum clarity. Finally, as we only evaluate five violation levels in this study it is worth discussing when this estimation, and general our robusteness metric, might fail. We discuss this in Apx. D.2.

A.3 Violation Depictions

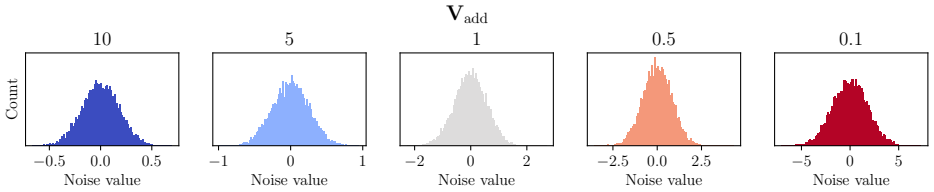
We include a graphic depiction of each described violation along with the concrete steps that we evaluated throughout our experiments in Fig. 4 - Fig. 8.

Violation	Increasing Intensity via	Parameter Values
$\mathbf{V}_{\text{obs,add}}$	Reducing the SNR	$\{10, 5, 1, 1/2, 1/10\}$
$\mathbf{V}_{\text{obs,mul}}$	Reducing the SNR	$\{10, 5, 1, 1/2, 1/10\}$
$\mathbf{V}_{\text{obs,time}}$	Reducing the SNR	$\{10, 5, 1, 1/2, 1/10\}$
$\mathbf{V}_{\text{obs,auto}}$	Reducing the SNR	$\{10, 5, 1, 1/2, 1/10\}$
$\mathbf{V}_{\text{obs,com}}$	Reducing the SNR	$\{10, 5, 1, 1/2, 1/10\}$
$\mathbf{V}_{\text{obs,shock}}$	Reducing the SNR	$\{10, 5, 1, 1/2, 1/10\}$
$\mathbf{V}_{\text{conf,inst}}$	Increased link probability from hidden confounders $\mathcal{Z}$.	$\{0.2, 0.4, 0.6, 0.8, 1.0\}$
$\mathbf{V}_{\text{conf,lag}}$	Increased link probability to/from hidden confounder X_c.	$\{0.1, 0.2, 0.5, 0.7, 0.9\}$
$\mathbf{V}_{\text{faith,lag}}$	Lagged effects increasingly cancel out.	$\{0.2, 0.15, 0.1, 0.05, 0.0\}$
$\mathbf{V}_{\text{faith,inst}}$	Instantaneous effects increasingly cancel out.	$\{0.2, 0.15, 0.1, 0.05, 0.0\}$
$\mathbf{V}_{\text{nl,mono}}$	Increasing the nonlinearity of sampled monotonic functions.	See Eq. (21) in Apx. B.4
$\mathbf{V}_{\text{nl,trend}}$	Reducing number of interpolation points for B-spline functions.	$\{25, 15, 10, 6, 4\}$
$\mathbf{V}_{\text{nl,rbf}}$	Higher probability of nonlinear links in the SCM (GP-RBF functions).	$\{0.2, 0.4, 0.6, 0.8, 1.0\}$
$\mathbf{V}_{\text{nl,comp}}$	Higher probability of nonlinear links in the SCM (composite functions).	$\{0.2, 0.4, 0.6, 0.8, 1.0\}$
$\mathbf{V}_{\text{inno,mul}}$	Signal dependent innovation noise portion.	$\{0.1, 0.25, 0.5, 0.75, 0.85\}$
$\mathbf{V}_{\text{inno,time}}$	Time-dependent innovation noise portion.	$\{0.1, 0.25, 0.5, 0.75, 0.85\}$
$\mathbf{V}_{\text{inno,auto}}$	Autoregressive innovation noise portion.	$\{0.1, 0.25, 0.5, 0.75, 0.85\}$
$\mathbf{V}_{\text{inno,com}}$	Common source innovation noise portion.	$\{0.1, 0.25, 0.5, 0.75, 0.85\}$
$\mathbf{V}_{\text{inno,shock}}$	Spike innovation noise portion.	$\{0.1, 0.25, 0.5, 0.75, 0.85\}$
$\mathbf{V}_{\text{inno,uni}}$	Uniform additive innovation noise scale.	$\{0.05, 0.25, 0.5, 0.75, 1.0\}$
$\mathbf{V}_{\text{inno,weib}}$	Weibull additive innovation noise scale.	$\{0.05, 0.25, 0.5, 0.75, 1.0\}$
$\mathbf{V}_{\text{inno,var}}$	Changing interval from which the σ_i^2 are uniformly sampled.	$[0.5, 1], [0.1, 1], [0.1, 2], [0.1, 4], [0.1, 8]$
$\mathbf{V}_{\text{stat}}$	Increasing number of SCM change points during generation of $\boldsymbol{X}$.	$\{1, 3, 5, 7, 9\}$
$\mathbf{V}_{\text{length}}$	Reducing number of observed steps T	$\{200, 100, 50, 25, 12\}$
$\mathbf{V}_{\text{q,empty}}$	Lengthening periods (ratios) of temporary loss of causal signal.	$2 \times \text{ratio} \in \{0.25, 0.345, 0.4, 0.425, 0.455\}$
$\mathbf{V}_{\text{q,missing}}$	Increasing probability of missing data points (interpolated).	$\{0.2, 0.35, 0.5, 0.65, 0.8\}$
$\mathbf{V}_{\text{scale}}$	Mixing factor of the standardization.	$\{0.0, 0.5, 0.7, 0.9, 1.0\}$

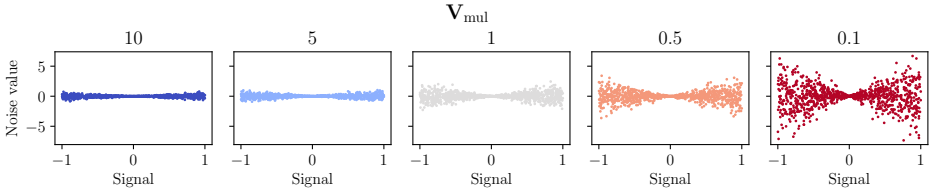
Table 3: A short description of how we scale all 27 violations contained in TCD-Arena. We include also a list of the specific parameter values to reproduce our empirical study.

A.4 Experimental protocol depiction

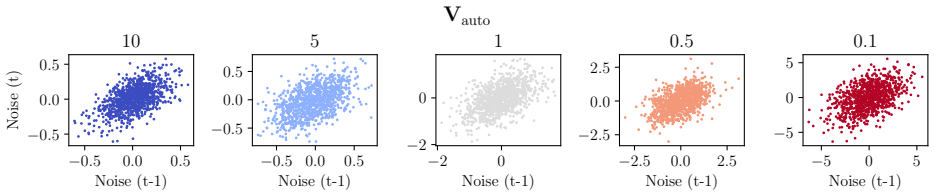
To clarify the protocol we used to asses the robustness of various CD methods against assumption violations, we depict the process in Fig. 9. The process can be divided into three aspects. Data generation, CD method evaluation and the extraction of robustness profiles.



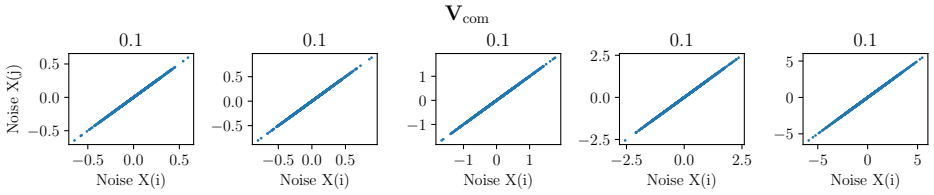
(a) Additive innovation noise distributions for all violation levels



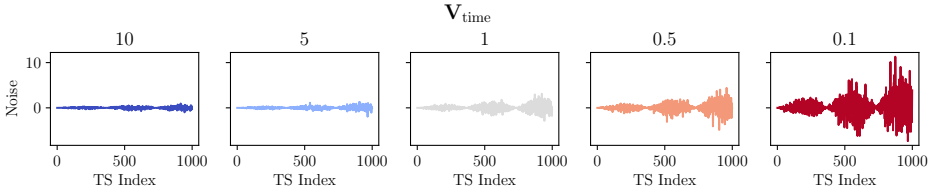
(b) Relationship between multiplicative innovation and the signal itself.



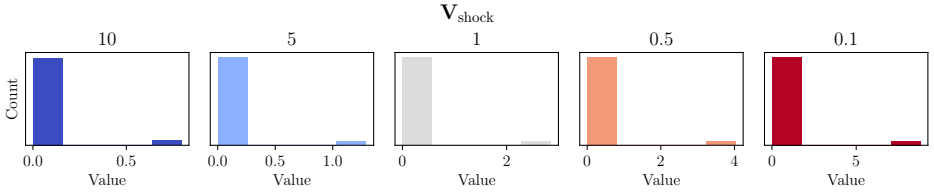
(c) Correlation of autoregressive observational noise between t and $t-1$



(d) Relationship between two observational noise variables in X_t

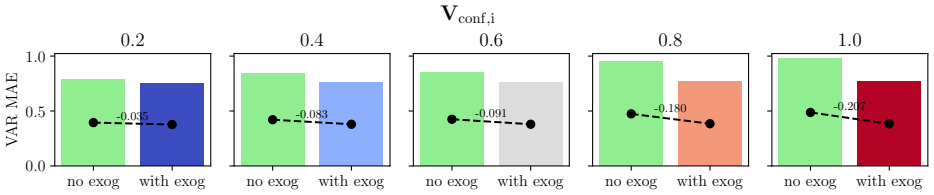


(e) Development of time-dependent observational noise

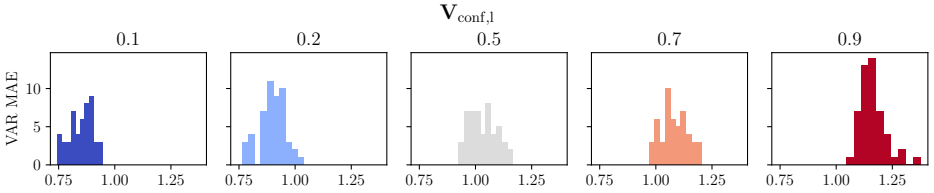


(f) Distributions over noise values when sampled from a shock distribution.

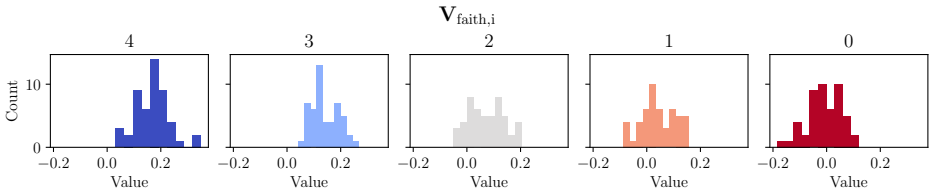
Figure 4: Various depictions of different violations of observational noise. We depict the severity of the violation from left to right and denote the SNR above the figure.



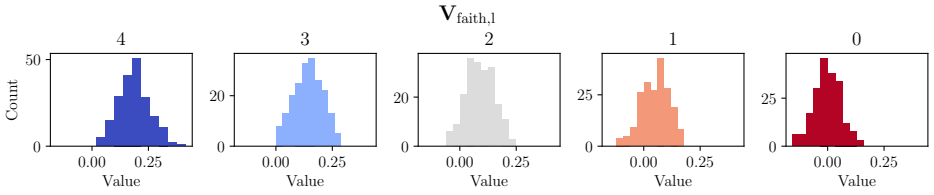
(a) Mean Absolute error between a VAR model that has or has no access to the exogenous variables Z.



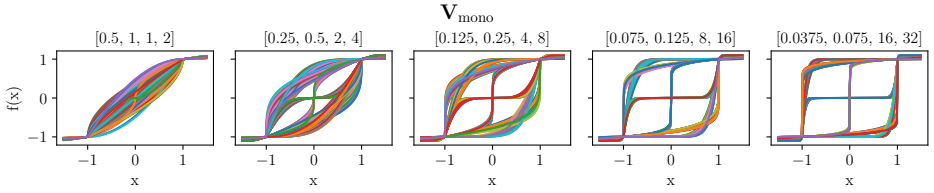
(b) Mean Absolute error when modeling X with a VAR process with no access to the confounder.



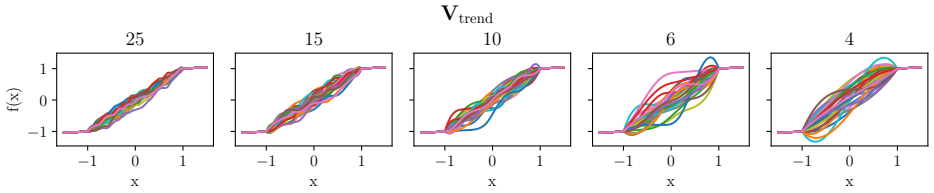
(c) Total sum of all parameters in the B.



(d) Total sum of all parameters in the A.

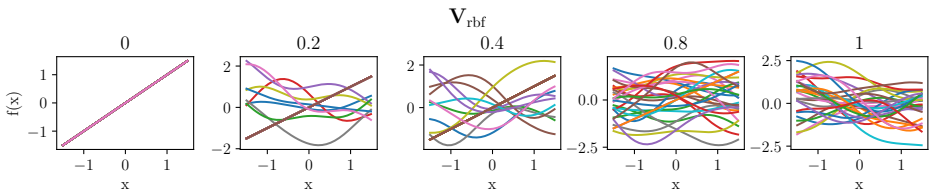


(e) Distributions from which we draw the nonlinear functions of the monotonic family.

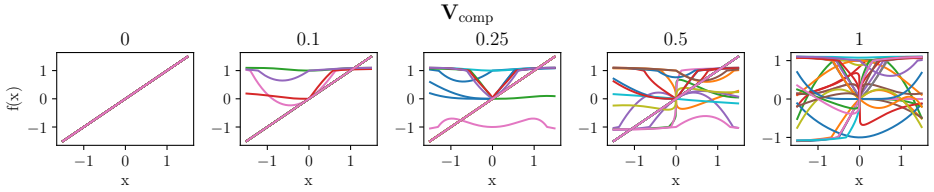


(f) Distributions from which we draw the nonlinear functions of the trend family.

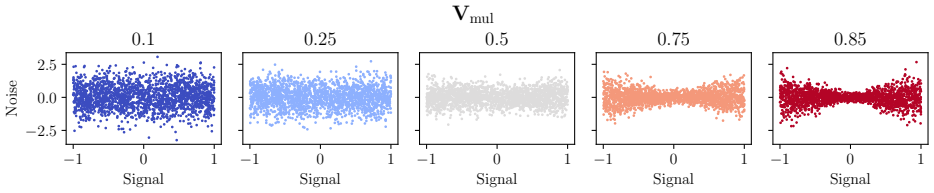
Figure 5: Graphical depictions of different violations and their intensities.



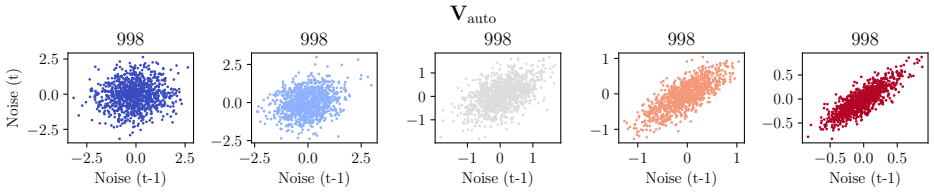
(a) Distributions from which we draw the nonlinear functions of the RBF family.



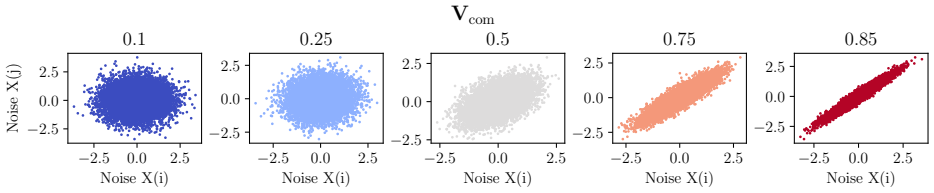
(b) Distributions from which we draw the nonlinear functions of the composite family.



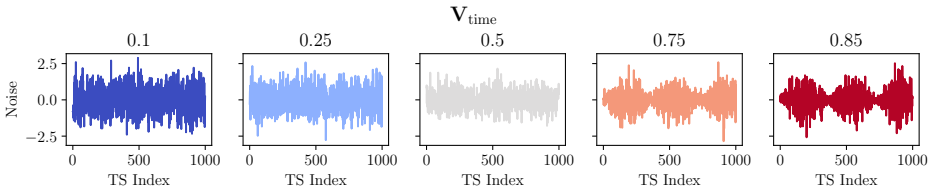
(c) Relationship between signal strength and innovation noise.



(d) Relationship between innovation noise of X_{t-1} and X_t.

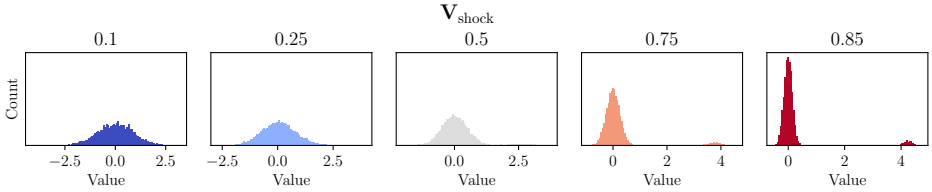


(e) Relationship between different innovation noise terms in X_t.

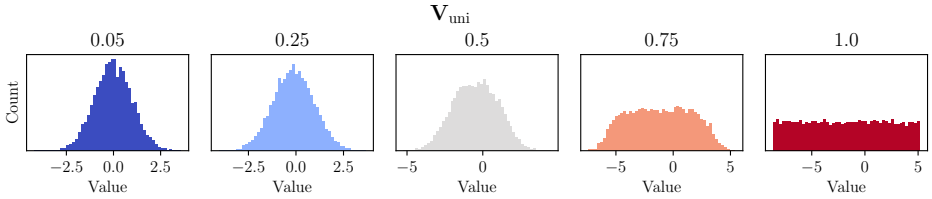


(f) Depiction of the development of time-dependent innovation noise.

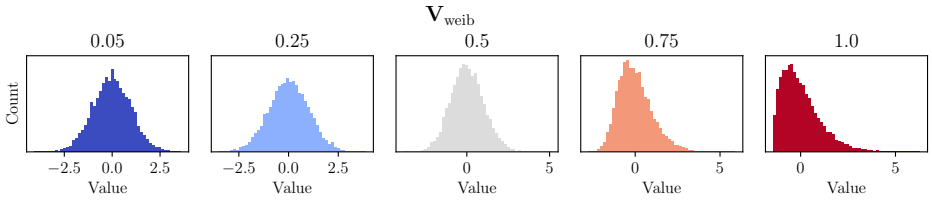
Figure 6: Graphical depictions of different violations and their intensities.



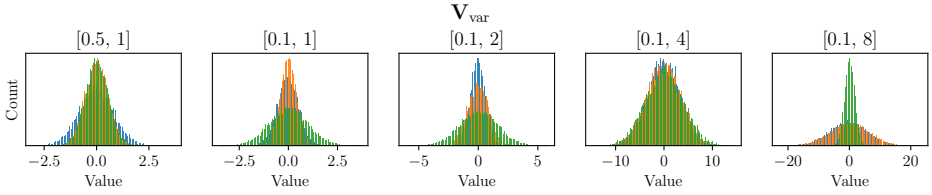
(a) Distributions from which we draw the shock innovation noise.



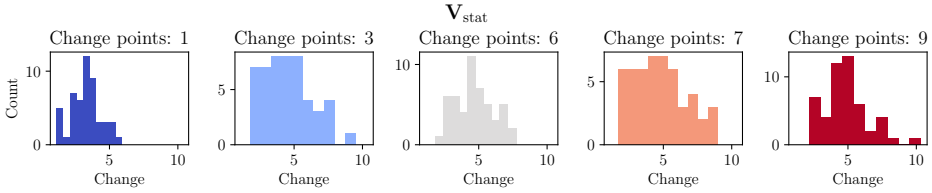
(b) Distributions from which we draw nongaussian innovation noise.



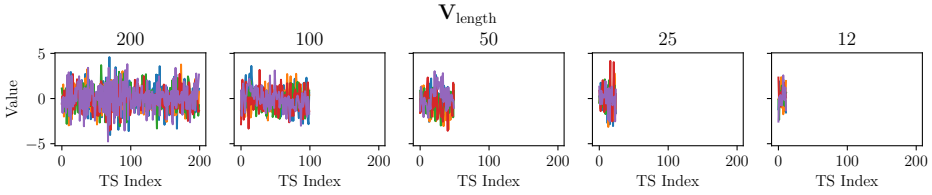
(c) Distributions from which we draw nongaussian innovation noise.



(d) Distributions from which we draw innovation noise with unequal variance for each variable in X.

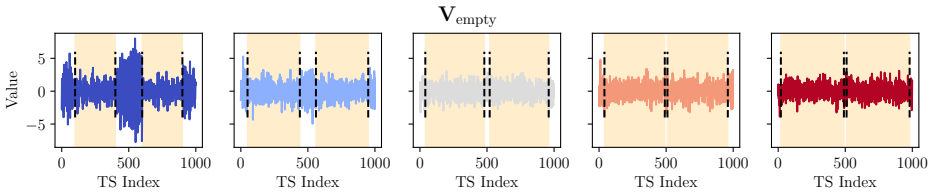


(e) Causal Coefficient absolute changes (elements in A) between first and last section of X.

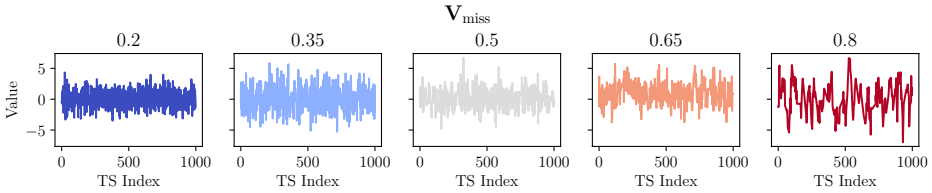


(f) Length of the time generated time series.

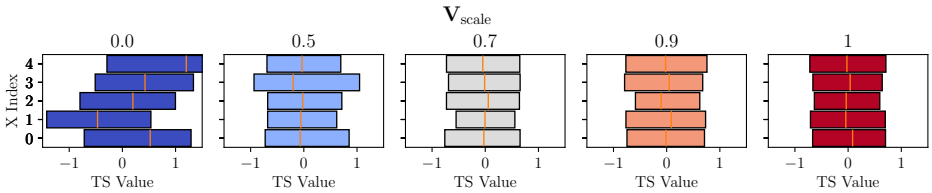
Figure 7: Graphical depictions of different violations and their intensities.



(a) Timeseries with partially empty A. Sections where A is empty are marked as yellow areas.



(b) Time series with increasingly missing and afterwards interpolated values.



(c) Mean and standard deviations of individual variables in X.

Figure 8: Graphical depictions of different violations and their intensities.

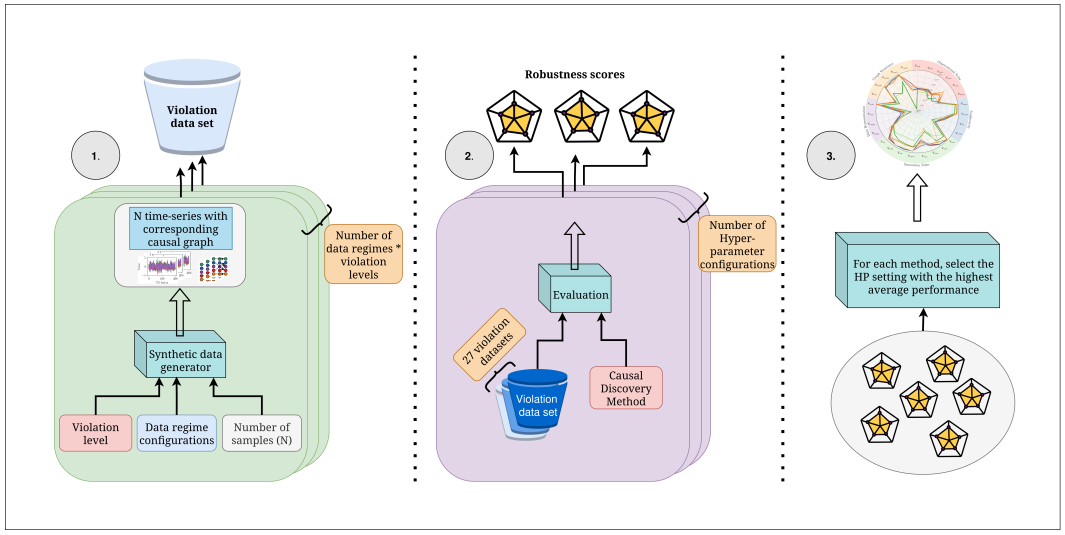


Figure 9: Experimental protocol that was used to create robustness profiles for various Causal Discovery methods. The process can be divided into three steps: 1. Data generation, 2. CD method evaluation and 3. Extraction of results.

B Appendix — Violation Details

Fundamentally, the base SCM that we use for each violation is a linear causal process with additive Gaussian noise:

$$X_{i,t} = \sum_{d=1}^{D} \sum_{l=0}^{L} A_{i,d,l} \cdot f_{i,d,l}(X_{d,t-l}) + \epsilon_{t,i}, \tag{3}$$

where f is the identify function, A a coefficient tensor, X the time series and ϵ additive gaussian noise. To bring this into a more compact form, omitting f:

$$X_{i,t} = \sum_{d=1}^{D} \sum_{l=0}^{L} A_{i,d,l} \cdot X_{d,t-l} + \epsilon_{t,i}, \tag{4}$$

Further, we can bring this into matrix notation:

$$X_t = AX_{t\ldots t-L} + \mathcal{E} \tag{5}$$

where $\mathcal{E}$ is a vector of independent innovation noise variables. Crucially, the non-zero entries in A correspond to links in the causal graph G. While sampling A in the base linear process, we control the density of links using a corresponding probability that determines whether entries $A_{i,d,l}$ are equal to zero.

Finally, we can separate instantaneous effects as they are not always present and are implemented in a different manner:

$$X_t = BX_t + AX_{t-1\ldots t-L} + \mathcal{E} \tag{6}$$

where A is a coefficient matrix and B is a coefficient vector.

To implement all our violations, we alter this basic linear additive process.

B.1 Additional Details $\mathbf{V}_{\text{OBS}}$

To briefly recap, when we violate the no observational noise assumption, then we do not directly observe the measurements $X_{i,t}$. Instead, we measure noisy versions $\hat{X}_{i,t} = X_{i,t} + \zeta_{i,t}$. We define a concrete list of observational noise variables and structures in Fig. 2a. In this section, we provide additional details for the specific design choices of the various implemented $\zeta_{i,t}$. Afterward, we include the concrete formula that we use to control the signal-to-noise ratios when increasing the respective observational noise violations.

First, we consider independent additive noise ($\mathbf{V}_{\text{obs,add}}$), where we model the noise as standard normal $\zeta_{i,t} \sim \mathcal{N}(0, 1)$.

Second, we consider multiplicative noise ($\mathbf{V}_{\text{obs,mul}}$), which in the signal-dependent noise model is an additive noise scaled by a function of the signal strength Torricelli et al. (2002); Liu et al. (2014). Here, we use $\zeta_{i,t} \sim \mathcal{N}\left(0, (X_{i,t})^2\right)$, i.e., a multiplication with the identity of the signal $X_{i,t}$ (see Fig. 2a). Real-world examples include temperature sensors whose precision degrades at high signal values Bentley (1984) and speckle noise in image processing Liu et al. (2014).

Third, $\mathbf{V}_{\text{obs,time}}$, specifies noise with distribution characteristics changing over time. Real-world examples of such a noise source would be sensor drifts or interference with periodic environmental factors. We model this by scaling the variance by a periodic signal, i.e., $\zeta_{i,t} \sim \mathcal{N}\big(0, ((1 + \alpha t) \cdot \sin(2\pi t/\beta))^2\big)$, where α and β are hyperparameters. In our experiments, we fix them to simulate an annual cycle and a small linear trend to simulate sensor degradation. Specifically, we use $\alpha = 0.01$ and $\beta = 2 \cdot 365 = 730$.

Fourth, $\mathbf{V}_{\text{obs,auto}}$ indicates an autoregressive noise structure ($\zeta_{i,t} \not\perp\!\!\!\perp \zeta_{i,t-1}$) that can be found when disturbances of the measurement process persist over multiple timesteps. Here, one could imagine a sensor that is overshadowed by a cloud for multiple consecutive time steps. We model this term as $\zeta_{i,t} \sim \mathcal{N}\left(\alpha\zeta_{i,t-1}, (1-\alpha)^2\right)$ where α is the weighting coefficient which is a hyperparameter.

Intuitively, the mean of the distribution depends on the last sampled noise, similarly to a random walk. In our implementation, we equally mix the previous step with the random source, i.e., $\alpha = 1/2$. This design choice ensures that the overall process, while dependent on previous noise, is still nondeterministic.

Fifth, noise sources across different variables can be dependent ($\mathbf{V}_{\text{obs,com}}$), i.e., $\zeta_{i,t} \not\perp\!\!\!\perp \zeta_{j,t}$ for $i \neq j$. Such a scenario can occur if multiple sensors are affected by a shared, unmeasured environmental factor (e.g., temperature, power fluctuations). We model this by sampling from a single noise source $\zeta_t \sim \mathcal{N}(0, 1)$ that is shared for all variables in $\boldsymbol{X}$, i.e., $\forall i \in \{1, \dots, D\} : \zeta_{i,t} = \zeta_t$ for a timestep t.

Finally, observed data might be subject to infrequent, large disturbances or measurement failures ($\mathbf{V}_{\text{obs,shock}}$). Using a shock probability p_{shock}, we model $\zeta_{i,t} = \begin{cases} S & \text{with probability } p_{\text{shock}} \\ 0 & \text{else} \end{cases}$, where S is a fixed scalar. In our experiments, we set $= 5$ and $p_{\text{shock}} = 0.05$.

As specified in our main paper, we isolate the influence of the noise structure by using five discrete, decreasing SNR levels. Inparticular, we are using $\{10, 5, 1, 1/2, 1/10\}$ for all observational noise violations in Fig. 2a.

To rescale the noise vector, to achieve a desired Signal-to-Noise Ratio ($\text{SNR}_{\text{target}}$) with respect to X, we first compute the average power of the signal and the unscaled base noise ζ_{base}.

The signal power, P_X, is defined as:

$$P_X = \frac{1}{T * D} \sum_{d=1}^{D} \sum_{t=1}^{T} X_{d,t}^2$$

The power of ζ_{base}, is:

$$P_{\zeta,base} = \frac{1}{T * D} \sum_{d=1}^{D} \sum_{t=1}^{T} \zeta_{\text{base,d},t}^2$$

Given a target $\text{SNR}_{\text{target}}$, the desired power for the final noise, $P_{N,\text{target}}$, is calculated as:

$$P_{N,\text{target}} = \frac{P_X}{\text{SNR}_{\text{target}}}$$

We find a scaling factor, α, that transforms the base noise power to the target noise power.

$$\alpha = \sqrt{\frac{P_X / \text{SNR}_{\text{target}}}{P_{\zeta,base}}}$$

The final noise vector ζ is then obtained by scaling the base noise:

$$\zeta = \alpha \cdot \zeta_{base}.$$

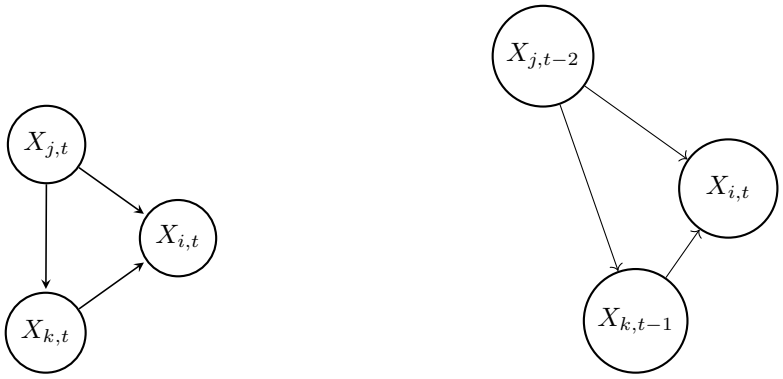
B.2 Additional Details $\mathbf{V}_{\text{CONF}}$

In our main paper, we specify two possible techniques to introduce confounding in a sampled time series. Specifically, we separate instantaneous effects ($\mathbf{V}_{\text{conf,inst}}$) and and lagged confounding ($\mathbf{V}_{\text{conf,lag}}$). We model the former by generating a set of N independent potential parent variables $Z_1, ..., Z_N$. In all time steps t, the Z_n are standard normally distributed and can act as common causes for any variable in $\boldsymbol{X}$. Hence, the causal assignment (Eq. (1)) for an observed variable $X_{i,t}$ becomes: $X_{i,t} = f_i\left(\text{Pa}_{\boldsymbol{X}}(X_{i,t}) \cup \text{Pa}_{\mathcal{Z}}(X_{i,t}), \epsilon_{i,t}\right)$. We scale $\mathbf{V}_{\text{conf,inst}}$ by increasing the probability of links from $\mathcal{Z}$ to variables in $\boldsymbol{X}$. In particular, we use the probabilities $\{0.2, 0.4, 0.6, 0.8, 1.0\}$.

To model $\mathbf{V}_{\text{conf,lag}}$, generate the time series with an additional confounding variable X_C. This variable is part of the normal process, while iteratively generating the time series, and can influence any variable X_i. Further, it can also be influenced by all other variables. After generating the complete time series, we create the observed data $\boldsymbol{X}$ by removing X_C, rendering it a hidden confounder. To scale $\mathbf{V}_{\text{conf,lag}}$, we again increase the probability of links from and to X_C. Specifically, we use the probabilities $\{0.1, 0.2, 0.5, 0.7, 0.9\}$.

B.3 Additional Details $\mathbf{V}_{\text{FAITH}}$

To violate faithfulness, we have to ensure that there are variables with a connection in the causal graph G, which have no measurable dependency, i.e., cancel each other out. Again, we separate instantaneous effects ($\mathbf{X}_{\text{faith,inst}}$) and lagged effects ($\mathbf{X}_{\text{faith,lag}}$) and visualize the structures we implement in Fig. 10.



(a) Structure for $\mathbf{V}_{\text{faith,inst}}$ (b) Structure for $\mathbf{V}_{\text{faith,lag}}$

Figure 10: The two structures, we enforce to violate faithfulness. Note that in Fig. 10b, some of the variables are lagged. In both cases, the connection from X_j to X_i cancels out in part by the connection over X_k, meaning X_i and X_j become more and more independent during the violations.

Now, to scale the violations of the faithfulness assumption, we adapt the parameters of the causal graph to cancel out information of X_j from X_i, using the path over X_k (see Fig. 10). Specifically, we implement the following assignments for $\mathbf{V}_{\text{faith,inst}}$ and $\mathbf{V}_{\text{faith,lag}}$ respectively:

$$
\begin{aligned}
v &\sim \text{Uniform}(0.3, 0.5) \\
X_{k,t} &:= 2vX_{j,t} + \epsilon_{i,t} \\
X_{i,t} &:= (-v + d)X_{j,t} + 0.5X_{k,t} + \epsilon_{i,t}
\end{aligned}
$$

$$
\begin{aligned}
v &\sim \text{Uniform}(0.3, 0.5) \\
X_{k,t-1} &:= 2vX_{j,t-2} + \epsilon_{i,t} \\
X_{i,t} &:= (-v + d)X_{j,t-2} + 0.5X_{k,t-1} + \epsilon_{i,t}
\end{aligned}
$$

Here d denotes the distortion parameter, which we decrease along the violation severity. We choose the levels $\{0.2, 0.15, 0.1, 0.05, 0.0\}$ for the violation severity in both cases. Further, we note that this implementation is only one way, arguably the most straightforward, to generate Unfaithfulness. However, as other ways of generating Unfaithfulness, such as deterministic relationships, conditional links (XOR), or specific mixtures of causal models, are possible, we plan on extending TCD-Arena in these directions in the future to gain additional insights.

B.4 Additional Details $\mathbf{V}_{\text{NL}}$

To introduce nonlinearities, studies on CD robustness sample functions from Gaussian Processes (GPs) with Radial Basis Function (RBF) kernels Yi et al. (2024); Montagna et al. (2023a). The smooth and oscillatory nature of these functions provides a difficult test case for algorithms that assume linear relationships, motivating the development of more robust nonlinear methods.

However, they may not fully represent the diverse spectrum of nonlinearities encountered in real-world applications. In many domains, such as those governed by physical constraints, nonlinearities often

adhere to specific characteristics like monotonicity or saturation, rather than arbitrary nonlinearity (e.g., SIR-models Kermack et al. (1997) or the Michaelis-Menten Kinetics Ainsworth (1977)).

In this section, we detail our four specific choices in nonlinear function distributions and the respective design paradigms. Further, a critical consideration in generating synthetic time series from such structural equations is ensuring the stability of the process (i.e., preventing divergence) because the functions are applied iteratively. This often requires constraining the output range or characteristics of the sampled functions $f_{i,d,l}$. As the specific constraints depend on the functional class, we first detail the normalization or bounding procedures possible during the generation process. Then we note the respective choices for the four families of functions we investigate. We also formalize what we mean by increasing the nonlinearity of the structural causal model (Eq. (2)).

B.4.1 Details on Boundary Conditions

For sampling nonlinear functions used to iteratively generate observations $\boldsymbol{X}$ using Eq. (6), it is important to consider the boundary behavior when inputs are either very large $x >> 0$ or very small $x << 0$. This is because if we leave the interval $[-1, 1]$ coupled with possibly high coefficients in A, it is possible for any of the D time series to diverge towards positive or negative infinity. This behavior could lead to numerically unstable values even in our finite simulated time steps T. While we will discuss specific checks to test for such conditions in Apx. C.1, we consider it here explicitly for the set of violations $\mathbf{V}_{\text{nl}}$ concerning the functional relationships $f_{i,d,l}$.

In our implementation, we focus on the input interval $x \in [-1, 1]$. Again, this specific setup is motivated by the time series sampling process, where we apply these functions iteratively as described in our main paper. To enforce saturation for $x \to \pm\infty$, we wrap sampled univariate functions f using hyperbolic tangents to roughly enforce value ranges of $[-1, 1]$. Consequently, values contained in the generated time series $\boldsymbol{X}$ stay close to $[-1, 1]$.

Specifically, we either wrap a function f based on the input x or the output $f(x)$ to ensure saturation. In particular, we employ

$$s_x(f(x)) = \begin{cases} f(x) & \text{if } x \in [-\alpha, \alpha] \\ \tanh(x) & \text{else} \end{cases}, \tag{7}$$

and

$$s_y(f(x)) = \begin{cases} f(x) & \text{if } |f(x)| <= \alpha \\ \tanh(f(x)) & \text{else} \end{cases}, \tag{8}$$

respectively. In Eq. (7) and Eq. (8), α defines the symmetric intervals around zero, which in our experiments is set to one. When detailing the specific functional families used for the violations $\mathbf{V}_{\text{nl,mono}}$, $\mathbf{V}_{\text{nl,trend}}$, $\mathbf{V}_{\text{nl,rbf}}$, and $\mathbf{V}_{\text{nl,comp}}$, we will specify which concrete wrapper s_x or s_y, we use to ensure saturation of the sampled $f_{i,d,l}$.

B.4.2 Quantifying and Increasing Nonlinearity

To formalize the concept of nonlinearity of a univariate function f, multiple scores were proposed in the literature. For instance, roughness penalties for spline smoothing, e.g., (Ramsay & Silverman, 2005, Sec. 5.2.2), quantify it using the squared curvature of a function over a specified interval. In particular, they calculate the deviation from a linear function as

$$\mathscr{D}_{\text{curv}}(f) = \int_{-\alpha}^{\alpha} (f''(x))^2 dx, \tag{9}$$

where f'' denotes the second derivative. If and only if the second derivative is zero over the complete interval $[-\alpha, \alpha]$, then f is linear in said interval. Further, given the squared integrand, $\mathscr{D}_{\text{curv}}$ is strictly nonnegative with minima exactly when f is a linear function. Intuitively, this score quantifies changes in the derivative of f, which is constant only for linear functions.

In contrast, in Emancipator & Kroll (1993), the authors measure the minimum possible mean squared error of f to any linear function in the interval of interest. Specifically, for a given f, we follow their approach and define the nonlinearity $\mathscr{D}_{\text{MSE}}$ in an interval $[-\alpha, \alpha]$ as

$$\mathscr{D}_{\text{MSE}}(f) = \min_{a,b \in \mathbb{R}} \left(\frac{1}{2\alpha} \int_{-\alpha}^{\alpha} \big(f(x) - (ax + b)\big)^2 dx \right). \tag{10}$$

Intuitively, $\mathscr{D}_{\text{MSE}}$ measures the minimum possible mean squared error to any linear function in $[-\alpha, \alpha]$ and is greater than or equal to zero for any f. Further, if f can be expressed as a linear function, then $\mathscr{D}_{\text{MSE}}$ is exactly zero. To compute $\mathscr{D}_{\text{MSE}}$, we have to consider the optimal $a^*, b^* \in \mathbb{R}$, which are necessary to minimize the mean squared error. In Emancipator & Kroll (1993), the authors give general solutions for arbitrary interval boundaries. In our case of a boundary symmetric around $x = 0$ of $[-\alpha, \alpha]$, the optimal solutions that minimize the error for a function f are given by

$$\begin{aligned} a^* &= \frac{3}{2\alpha^3} \int_{-\alpha}^{\alpha} x f(x) dx, \quad \text{and} \\ b^* &= \frac{1}{2\alpha} \int_{-\alpha}^{\alpha} f(x) dx. \end{aligned} \tag{11}$$

Hence, the measure for nonlinearity becomes

$$\mathscr{D}_{\text{MSE}}(f) = \frac{1}{2} \int_{-1}^{1} \big(f(x) - (a^* x + b^*)\big)^2 dx. \tag{12}$$

Both $\mathscr{D}_{\text{curv}}$ and $\mathscr{D}_{\text{MSE}}$ behave differently and are not always aligned. Specifically, $\mathscr{D}_{\text{MSE}}$ considers the absolute distance to a line, meaning it can change if we multiply f with a constant factor, while $\mathscr{D}_{\text{curv}}$ would not change. However, $\mathscr{D}_{\text{curv}}$ necessitates that the function is twice differentiable, i.e., in C^2, in the interval of interest $[-\alpha, \alpha]$. Hence, it cannot distinguish nonlinearity between step functions or absolute values, even if they closely follow a linear function in absolute deviation. In contrast, $\mathscr{D}_{\text{MSE}}$ is finite in such cases but includes an optimization process. However, using linear regression, we can empirically estimate $\mathscr{D}_{\text{MSE}}$ for any given function.

In our work, we randomly generate time series. Hence, we are interested in the approximate behavior of the resulting processes. We formalize this by describing families of distributions $\mathcal{F}$ of a function f with stepwise varying nonlinearity. Specifically, we ensure that sampled functions $f \sim \mathcal{F}$ have controllable expected nonlinearity

$$\mathbb{E}_{f \sim \mathcal{F}}[\mathscr{D}(f))], \tag{13}$$

where $\mathscr{D}$ is a measure of nonlinearity. Thus, to increase the nonlinearity of sampled processes, we stepwise change $\mathcal{F}$ from which we sample the functions $f_{i,d,l}$ in Eq. (2).

Lastly, consider that in our formulation of the general structural causal model (Eq. (2)), the functions in the causal graph G are univariate and connect two, possibly lagged variables. Hence, another approach to increase the nonlinearity in a stepwise manner is to only sample a subset of the $f_{i,d,l}$ from a distribution of nonlinear functions while keeping the rest linear. This leaves us with a third possibility to increase the nonlinearity of the overall SCM.

In the following, we specify four families of distributions of nonlinear functions and establish how specifically we change the nonlinearity of the resulting sampled time series processes.

B.4.3 1. MONOTONIC FAMILY:

For the first family, we sample uniformly one of three univariate functions, where a parameter $\beta > 0$ determines the nonlinearity. Before we formally analyze the influence of this parameter, we detail our specific functions and motivate our design choices. We use

$$\begin{aligned} f_1(x; \beta) &= \text{sgn}(x) \cdot |x|^{\beta}, \\ f_2(x; \beta) &= c \left| \frac{(x+1)}{c} \right|^{\beta} - 1, \quad \text{and} \\ f_3(x; \beta) &= -c \left| \frac{(x-1)}{c} \right|^{\beta} + 1, \end{aligned} \tag{14}$$

where c is a hyperparameter which we set to 2 in our experiments. The specific choice of the added and subtracted 1 in f_2 and f_3 depends on our interval of interest $[-\alpha, \alpha]$, where we want to change the non-linearity using β.

Note that for the following analysis and in our implementation, we focus on the interval $[-1, 1]$. This specific setup is motivated by the timeseries sampling process, where we apply these functions

iteratively as described in our main paper. Hence, functions that scale values outside of this interval likely lead to divergent processes. Furthermore, we wrap each function using a hyperbolic tangent of the input (Eq. (7)) to ensure that our functions are saturated. Additionally, this ensures that values stay roughly in the interval of interest [-1,1].

Our intention in designing this family of functions is to ensure monotonicity in the specified interval, hypothesizing that this property is important for CD methods. We investigate this hypothesis empirically in our main paper. As a first step, we now show that all three of our functions are monotonically increasing in $[-1, 1]$ and for $\beta > 0$.

To prove this statement, it is enough to show that the first derivative is greater than or equal to zero for all $x \in [-1, 1]$. Note that given Eq. (2) in our main paper, monotonically decreasing functions are possible because the coefficient matrix A can have negative entries. Hence, without loss of generality, we focus in the following on the monotonically increasing nature. Specifically, consider our three functions (Eq. (14)) only in the interval of interest $[-1, 1]$

$$\begin{aligned} f_1(x;\beta) &= \text{sgn}(x) \cdot |x|^\beta \\ f_2(x;\beta) &= c\left(\frac{(x+1)}{c}\right)^\beta - 1, \\ f_3(x;\beta) &= -c\left(-\frac{(x-1)}{c}\right)^\beta + 1, \end{aligned} \tag{15}$$

where the absolute value can be removed from f_2 and f_3 because $x \pm 1$ is always positive or negative, respectively.

We start our analysis with the derivative of f_1, which has three cases, i.e., $x < 0$, $x > 0$, and $x = 0$. We start with the first two:

Case 1, $-1 \leq x < 0$**:** In this case, $\text{sgn}(x) = -1$ and $|x| = -x$ apply, leading to $f_1(x;\beta) = (-1)(-x)^\beta$. Using the chain rule, we find

$$f_1'(x;\beta) = (-1)\beta(-x)^{\beta-1} \cdot (-1) = \beta(-x)^{\beta-1} \tag{16}$$

$$= \beta|x|^{\beta-1}. \tag{17}$$

Case 2, $0 < x \leq 1$**:** Here, the absolute value becomes the identity and $\text{sgn}(x) = 1$. Thus, we have $f_1(x;\beta) =$, and

$$f_1'(x;\beta) = \beta \cdot x^{\beta-1} = \beta|x|^{\beta-1}. \tag{18}$$

In both cases, we can see that the derivative is equal to $f_1'(x;\beta) = \beta|x|^{\beta-1}$. For all $x \in [-1, 1] \setminus \{0\}$ and $\beta > 0$ this is strictly nonnegative. Lastly, to prove that f_1 is monotonically increasing in the interval of interest, it is left to show that the derivative is also larger than or equal to zero for $x = 0$. Here, the value depends on the specific setting of β. For $\beta > 0$, we have three cases and we study the limits of f_1' from both directions

Case 3.1, $\beta = 1$**:** In the linear case, we find

$$\begin{aligned} \lim_{x\to 0^+} f_1'(x;1) &= \lim_{x\to 0^+} 1 \cdot |x|^0 = \lim_{x\to 0^+} 1 = 1, \text{ and} \\ \lim_{x\to 0^-} f_1'(x;1) &= \lim_{x\to 0^-} 1 \cdot |x|^0 = \lim_{x\to 0^-} 1 = 1. \end{aligned}$$

In other words, the derivative is constant and larger than zero.

Case 3.2, $\beta > 1$**:** Here we find the exponent $\beta - 1 > 0$ leading to the following two limits

$$\begin{aligned} \lim_{x\to 0^+} f_1'(x;1) &= \lim_{x\to 0^+} \beta \cdot |x|^{\beta-1} = \beta \cdot |0|^{\beta-1} = 0, \text{ and} \\ \lim_{x\to 0^-} f_1'(x;1) &= \lim_{x\to 0^-} \beta \cdot |x|^{\beta-1} = \beta \cdot |0|^{\beta-1} = 0. \end{aligned}$$

Hence, we find a saddle point, where the rate of change is exactly zero when $x = 0$.

Case 3.3, $0 < \beta < 1$**:** In this case, the exponent $\beta - 1$ becomes negative meaning $|x|^{\beta-1} = 1/|x|^{1-\beta}$. Consequently, limits from both sides diverge towards

$$\lim_{x\to 0^+} f_1'(x;1) = \lim_{x\to 0^+} \beta \cdot \frac{1}{|x|^{1-\beta}} = +\infty, \text{ and}$$
$$\lim_{x\to 0^-} f_1'(x;1) = \lim_{x\to 0^-} \beta \cdot \frac{1}{|x|^{1-\beta}} = +\infty.$$

Crucially, in all three cases, the limits of the derivative from both sides are equal and strictly nonnegative. Hence, f_1 is monotonically increasing in $[-1, 1]$ for all $\beta > 0$.

Next, we investigate the derivatives of f_2 and f_3. Following the observation that for $x \in [-1, 1]$ the absolute values can be rewritten as in Eq. (15), we calculate f_2' and f_3' using the chain rule as

$$f_2'(x;\beta) = \beta \left(\frac{x+1}{c}\right)^{\beta-1}, \quad \text{and}$$
$$f_3'(x;\beta) = \beta \left(\frac{1-x}{c}\right)^{\beta-1}. \tag{19}$$

For both functions, we have a strictly positive number ($\beta > 0$) which is multiplied by a base raised to a real power. Remember that in our experiments, we set $c = 2$, meaning both $(x-1)/2$ and $(1-x)/2$ vary in $[0, 1]$, i.e., are strictly nonnegative. Therefore, raising it by a real power ($\beta - 1$) leads for both f_2' and f_3' to a positive factor times a nonnegative factor. Hence, for all $x \in [-1, 1]$ and $\beta > 0$, we find $f_2'(x;\beta) \geq 0$ and $f_3'(x;\beta) \geq 0$. Note that in both cases, when $0 < \beta < 1$, we again find limits for both derivatives, where they become infinite. Specifically, for f_2, we observe a vertical tangent when $x = -1$, and for f_3, we similarly observe one for $x = 1$ (compare to case 3.3 of f_1). Nevertheless, the derivatives of all three functions f_1, f_2, and f_3 are strictly nonnegative in the specified interval. Hence, the functions themselves are monotonically increasing in $[-1, 1]$ for all $\beta > 0$. Next, we discuss how we can increase the nonlinearity of all three functions.

Monotonic Family Increasing Nonlinearity As specified above, for a given distribution of functions, we can quantify the linearity by considering the corresponding expectation. For the monotonic family of functions, the distribution we consider is a uniform choice of $\{f_i, f_2, f_3\}$. Hence, for a fixed β, we are interested in

$$\mathbb{E}_{f_j \sim \text{Uniform}\{f_1,f_2,f_3\}}[\mathscr{D}_{\text{MSE}}(f_j(\,\cdot\,;\beta))]. \tag{20}$$

In the case of this uniform distribution, all three cases are equally likely. Thus, the expectation for a fixed β is equal to the average of $\mathscr{D}_{\text{MSE}}$ for the three functions. We specifically choose $\mathscr{D}_{\text{MSE}}$ because the integral over the squared second derivative ($\mathscr{D}_{\text{curv}}$) of f_1 diverges for $1 < \beta < 1.5$. Further, we are interested in measuring the squared deviation from any possible line in $[-1, 1]$.

Consider that the parameter β directly controls the nonlinearity of f_1, f_2, and f_3. In particular, all three functions are equal and linear in $[-1, 1]$ if $\beta = 1$

$$f_1(x;1) = \text{sgn}(x) \cdot |x| = x,$$
$$f_2(x;1) = c\left(\frac{x+1}{c}\right) - 1 = x,$$
$$f_3(x;1) = -c\left(-\frac{x-1}{c}\right) + 1 = x.$$

Hence, the expectation in Eq. (20) becomes zero for $\beta = 1$.

Now, by changing β away from 1, all three functions become nonlinear in the sense of $\mathscr{D}_{\text{MSE}}$. Specifically, we construct five discrete levels $\ell \in \{1, 2, 3, 4, 5\}$ to scale $\mathbf{V}_{\text{nl,mono}}$ and sample β with an equal chance from either of the following intervals

$$\begin{aligned}
&\ell = 1 \to \beta \in [1/2, 1] \text{ or } \beta \in [1, 2],\\
&\ell = 2 \to \beta \in [1/4, 1/2] \text{ or } \beta \in [2, 4],\\
&\ell = 3 \to \beta \in [1/8, 1/4] \text{ or } \beta \in [4, 8],\\
&\ell = 4 \to \beta \in [1/12, 1/8] \text{ or } \beta \in [8, 12],\\
&\ell = 5 \to \beta \in [1/20, 1/12] \text{ or } \beta \in [12, 20].
\end{aligned} \tag{21}$$

For a concrete level ℓ, we denote the lower and upper boundaries of the two intervals with $[\beta_L^{(\ell\downarrow)}, \beta_U^{(\ell\downarrow)}]$ and $[\beta_L^{(\ell\uparrow)}, \beta_U^{(\ell\uparrow)}]$, respectively. Fig. 11 visualizes examples of functions drawn from the five levels of the resulting distributions. Crucially, the intervals of the distinct levels only overlap at a maximum of two concrete boundary points with any of the other intervals.

To analyze the non-linearity of our functions $f_j(\cdot;\beta)$ in the interval $x \in [-1, 1]$, we consider a second expectation over β distributed uniformly from either of the two intervals given by a level ℓ, i.e.,

$$\mathbb{E}_\beta[\mathbb{E}_{f_j}[\mathscr{D}_{\text{MSE}}(f_j(\,\cdot\,;\beta))]], \tag{22}$$

where we omit the specific distributions for brevity.

Here, both the function f_j and β are sampled independently. Hence, Eq. (22) is equal to

$$\begin{aligned}&\int_{\beta_L^{(\ell\downarrow)}}^{\beta_U^{(\ell\downarrow)}} \frac{1}{2(\beta_U^{(\ell\downarrow)} - \beta_L^{(\ell\downarrow)})} \sum_{j=1}^{3} \frac{1}{3} \mathscr{D}_{\text{MSE}}(f_j(\,\cdot\,;\beta)) d\beta \\ &+ \int_{\beta_L^{(\ell\uparrow)}}^{\beta_U^{(\ell\uparrow)}} \frac{1}{2(\beta_U^{(\ell\uparrow)} - \beta_L^{(\ell\uparrow)})} \sum_{j=1}^{3} \frac{1}{3} \mathscr{D}_{\text{MSE}}(f_j(\,\cdot\,;\beta)) d\beta,\end{aligned} \tag{23}$$

where both of the integrals describe one of the two equally likely and symmetrical intervals from which β is sampled, respectively. Further, for a fixed level ℓ, the factors contained in the intervals are a fixed normalization given by the probability density of the corresponding uniform distributions over the intervals $[\beta_L^{(\ell\downarrow)}, \beta_U^{(\ell\downarrow)}]$ and $[\beta_L^{(\ell\uparrow)}, \beta_U^{(\ell\uparrow)}]$.

By linearity of expectation, we can reorder Eq. (23) into

$$\begin{aligned}\frac{1}{3} \sum_{j=1}^{3} \Bigg(&\int_{\beta_L^{(\ell\downarrow)}}^{\beta_U^{(\ell\downarrow)}} \frac{\mathscr{D}_{\text{MSE}}(f_j(\,\cdot\,;\beta))}{2(\beta_U^{(\ell\downarrow)} - \beta_L^{(\ell\downarrow)})} d\beta \\ &+ \int_{\beta_L^{(\ell\uparrow)}}^{\beta_U^{(\ell\uparrow)}} \frac{\mathscr{D}_{\text{MSE}}(f_j(\,\cdot\,;\beta))}{2(\beta_U^{(\ell\uparrow)} - \beta_L^{(\ell\uparrow)})} d\beta \Bigg).\end{aligned} \tag{24}$$

As stated above, the only point in all intervals we consider where the f_j are linear is for $\beta = 1$. In any other case, $\mathscr{D}_{\text{MSE}}(f_j(\,\cdot\,;\beta)) > 0$ applies.

To now show that the expectation increases with the level ℓ, consider that the integrals in Eq. (24) calculate averages over all values of $\mathscr{D}_{\text{MSE}}$ for β in the corresponding intervals. Hence, it is enough to show that for $\beta > 0$, $\mathscr{D}_{\text{MSE}}$ is smooth and increases when moving away from the global minimum at $\beta = 1$ in our specified intervals. In all cases, we focus our analysis on the set of functions $\{f_1, f_2, f_3\}$ we defined above. Consider that for the interval $[-1, 1]$, the optimal parameter for the MSE minimizing line (Eq. (11)), become

$$\begin{aligned}a^* &= \frac{3}{2} \int_{-1}^{1} x f(x) dx, \quad \text{and} \\ b^* &= \frac{1}{2} \int_{-1}^{1} f(x) dx.\end{aligned} \tag{25}$$

We visualize examples for f_1, f_2, and f_3 and the corresponding optimal lines in Fig. 11.

Now to determine whether $\mathscr{D}_{\text{MSE}}$ is smooth with respect to changes in β, we have to consider the three terms in Eq. (12) that are functions of β: f_j, a^*, and b^*, where the last two also depend on the specific function f_j (Eq. (11), Eq. (25)).

For all three of our functions f_j, the critical part is of the form $|g(x)|^\beta$, where $g(\cdot)$ is defined in Eq. (14). Hence, the following equality holds

$$|g(x)|^\beta = e^{\beta \ln |g(x)|}. \tag{26}$$

In particular, the function has an exponential form, which is infinitely differentiable (C^∞) with respect to β for any fixed x (where $g(x) \neq 0$). In other words, f_1, f_2, and f_3 are smooth with respect

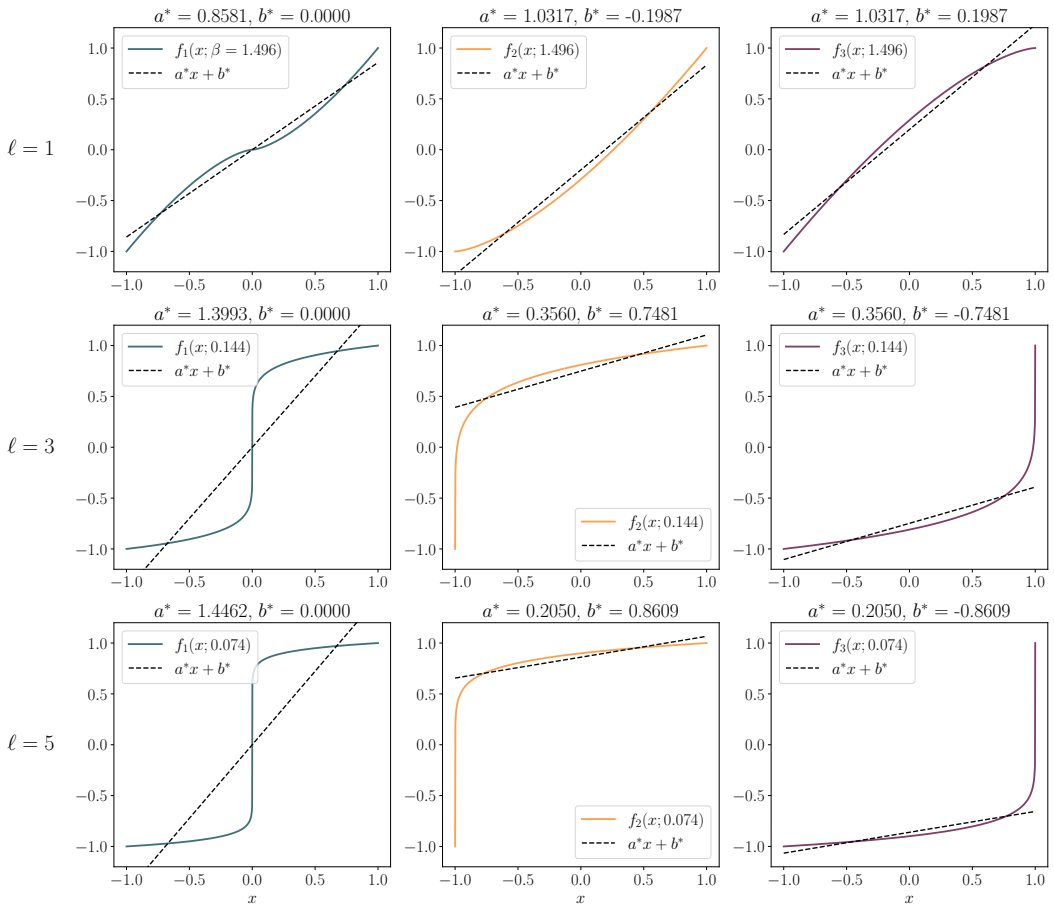


Figure 11: Examples of f_1, f_2, and f_3 with randomly sampled β from the respective level ℓ. We also visualize the optimal line and denote the corresponding parameters.

to β in $[-1, 1]$. Further, as a direct consequence of the Leipnitz integral rule, e.g., (Protter & Morrey, 1985, Chap. 8), integrals of the form $\int_{-1}^{1} f_j(x; \beta)dx$ and $\int_{-1}^{1} x f_j(x; \beta)dx$ are also smooth functions with respect to β. Finally, consider that $\mathscr{D}_{\text{MSE}}$ is again an integral with respect to x of a square of the sum of three functions that are smooth with respect to β. Hence, using the Leipnitz integral rule and the chain rule for differentiation, we can conclude that $\mathscr{D}_{\text{MSE}}$ is also smooth in β for the functions f_1, f_2, and f_3.

To show that the nonlinearity increases if we shift β away from the linear case of $\beta = 1$, we now study $\frac{\partial}{\partial \beta}\mathscr{D}_{\text{MSE}}$. In particular, using the Leibniz rule and the chain rule, we have

$$\frac{\partial \mathscr{D}_{\text{MSE}}(f_j)}{\partial \beta} = \int_{-1}^{1} \left(f_j(x; \beta) - (a^* x + b^*)\right) \cdot \left(\frac{\partial f_j(x; \beta)}{\partial \beta} - x\frac{\partial a^*}{\partial \beta} - \frac{\partial b^*}{\partial \beta}\right) dx. \tag{27}$$

Expanding the product in the integral leaves us with three separate terms

$$\begin{aligned}\frac{\partial \mathscr{D}_{\text{MSE}}(f_j)}{\partial \beta} = &\int_{-1}^{1} \big(f_j(x;\beta) - (a^*x + b^*)\big)\frac{\partial f_j(x;\beta)}{\partial \beta}dx \\ &-\frac{\partial a^*}{\partial \beta}\int_{-1}^{1} x(f_j(x;\beta) - a^*x - b^*)dx \\ &-\frac{\partial b^*}{\partial \beta}\int_{-1}^{1} (f_j(x;\beta) - a^*x - b^*)dx.\end{aligned} \tag{28}$$

Crucially, note that it is possible to move the partial derivatives $\frac{\partial a^*}{\partial \beta}$ and $\frac{\partial b^*}{\partial \beta}$ outside of the integral because they are independent of x. This is important because the integrals in the second and third terms are exactly the first-order optimality conditions of a^* and b^*, respectively Emancipator & Kroll (1993). Hence, both of these integrals vanish, and we are left with

$$\frac{\partial \mathscr{D}_{\text{MSE}}(f_j)}{\partial \beta} = \int_{-1}^{1} \big(f_j(x;\beta) - (a^*x + b^*)\big)\frac{\partial f_j(x;\beta)}{\partial \beta}dx. \tag{29}$$

In Eq. (29), we have two factors: the residual error to the MSE optimal line and the sensitivity of f_j with respect to changes in β. Given that the residual is a constant zero at $\beta = 1$ when our f_j become linear, we again confirm that this is a minimum of $\mathscr{D}_{\text{MSE}}$. Hence, $\frac{\partial \mathscr{D}_{\text{MSE}}(f_j)}{\partial \beta} = 0$ if $\beta = 1$. Consider now that for all values $\beta > 0$ which are not $\beta = 1$, all our functions f_1, f_2, and f_3 are nonlinear, we know that $\mathscr{D}_{\text{MSE}}$ has to be strictly larger than zero. This implies that $\beta = 1$ is a unique global minimum. Given this observation and the previous insight that $\mathscr{D}_{\text{MSE}}$ is smooth with respect to β, we conclude that the averages over $\mathscr{D}_{\text{MSE}}$ have to increase locally in the neighborhood of the global minimum at $\beta = 1$. However, this does not necessarily imply that the only critical point is at $\beta = 1$. To test whether our claim that the expected nonlinearity increases for an increase in level ℓ, we use Eq. (11) and Eq. (12) to simulate the nonlinearity.

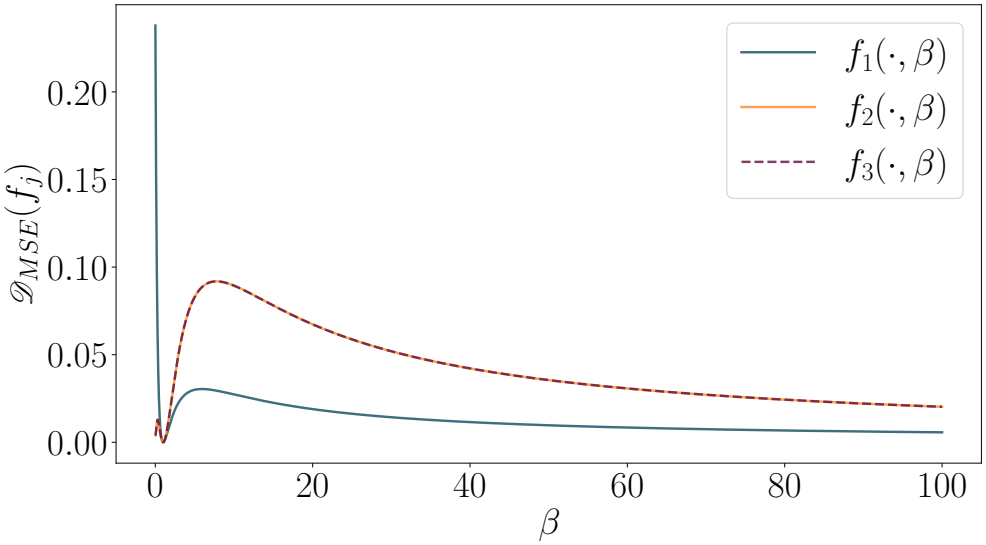


Figure 12: Nonlinearity measured with $\mathscr{D}_{\text{MSE}}$ for the three functions f_1, f_2, and f_3 for increasing values of $\beta > 0$.

We visualize $\mathscr{D}_{\text{MSE}}$ in Fig. 12 and confirm that the nonlinearity increases when we move away from the global minimum $\beta = 1$. However, we observe local maxima in the interval $(0, 20]$. Hence, it is unclear how the expected nonlinearity for randomly sampled f_j and β according to the defined levels ℓ behaves.

ℓ	f_1	f_2	f_3	$\mathbb{E}[\mathscr{D}_{\text{MSE}}(f_j)]$
1	0.005178	0.006049	0.005783	0.005670
2	0.032103	0.030878	0.029993	0.030991
3	0.066364	0.046946	0.048134	0.053815
4	0.087101	0.046346	0.044872	0.059440
5	0.103752	0.039548	0.037819	0.060373

Table 4: Approximated nonlinearity scores for the three functions f_1, f_2, and f_3 and different levels ℓ. The last column contains the accumulated $\mathscr{D}_{\text{MSE}}$ over all f_j for all β sampled in the respective level.

Thus, we estimate the expected nonlinearity (Eq. (24)) per level ℓ. Specifically, we sample 1000 β values for each level and use the theoretically optimal line parameters a^* and b^*. We list the approximated expected nonlinearity in Table 4. We find that the expected $\mathscr{D}_{\text{MSE}}$ does stepwise increase for f_1 while it decreases slightly for f_2 and f_3 again after $\ell = 3$. However, the accumulated expectation over all functions (Eq. (22)) does grow for $\ell = 1, ..., 5$. Therefore, we conclude that the empirical nonlinearity does increase stepwise for our defined violation levels.

B.4.4 2. B-Splines Following a Trend:

Next, we investigate univariate functions f that have an overall increasing trend but are not necessarily monotonic in nature. To do this, we rely on B-spline interpolations, e.g, de Boor (2001). Specifically, we sample sample N_P scalar values (interpolation points) $\{v_1, v_2, ..., v_{N_P}\}$ from a uniform distribution Uniform$(-1, 1)$. Next, we sort the values v_j and set them as targets for $f(x)$ at equidistant abscissae in the range $x \in [-1, 1]$. Consequently, a B-spline $f(x) = \sum_{j=1}^{N_P} c_k B_{j,k}(x)$ of degree $k = 3$ is constructed to smoothly interpolate these points. The corresponding B-spline basis elements are given by

$$B_{j,0}(x) = \begin{cases} 1 & \text{if } \tau_j \leq x < \tau_{j+1}, \\ 0 & \text{else} \end{cases}$$

$$B_{j,k}(x) = \frac{x - \tau_j}{\tau_{j+k} - \tau_j} B_{j,k-1}(x) + \frac{\tau_{j+k+1} - x}{\tau_{j+k+1} - \tau_{j+1}} B_{j+1,k-1}(x),$$

where we determine the entries of the knot vector τ as described in de Boor (2001) using the implementation provided in Virtanen et al. (2020). Given that the basis functions are piecewise cubic, the fitted curve is not monotonic, as visualized in Fig. 5f. To ensure saturation, we use the procedure described in Eq. (7). For $\mathbf{V}_{\text{nl,trend}}$, we sample different $f_{i,d,l}$ for all non-zero interactions in Eq. (2) by sampling different v_j. Next, we discuss how we stepwise increase in nonlinearity of the sampled functions.

Number of Interpolation Points	$\mathbb{E}_{f_{\text{spline}}}[\mathscr{D}_{\text{MSE}}(f_{\text{spline}})]$
25	0.003792
15	0.006517
10	0.010198
6	0.017728
4	0.019475

Table 5: Estimated expected $\mathscr{D}_{\text{MSE}}$ for the spline functions sampled for the five decreasing number of interpolation points.

***B*-Splines Increasing Nonlinearity** To stepwise scale the nonlinearity of the sampled functions $f_{i,d,l}$, we decrease the number N_P of interpolation points. Specifically, we use the following values: $\{25, 15, 10, 6, 4\}$. Intuitively, a higher number of interpolation points indicates more values v_j that

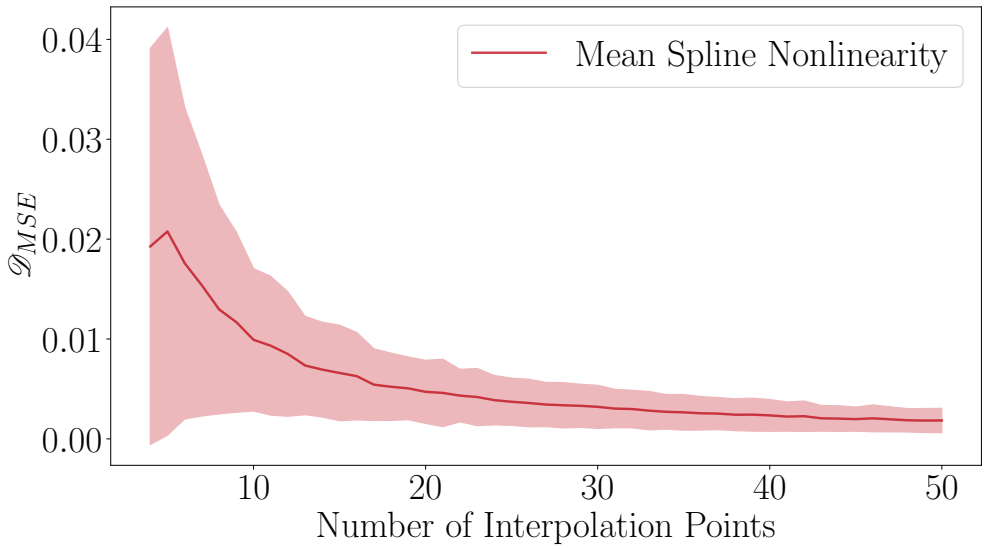


Figure 13: Nonlinearity measured with $\mathscr{D}_{\text{MSE}}$ for the spline functions for an increasing number of interpolation points. We report the average and the standard deviations.

have to be interpolated and which are strictly increasing. To empirically show that a larger N_P leads to more linear functions in the sense of $\mathscr{D}_{\text{MSE}}$ (Eq. (12)), we use Eq. (11) to approximate the average nonlinearity in Fig. 13. Further, following the uniform distribution of independently sampled v_j, we approximate the expected $\mathscr{D}_{\text{MSE}}$ of our concrete $\mathbf{V}_{\text{nl,trend}}$ in Table 5. Similar to the monotonic family of functions, we observe that the nonlinearity of sampled $f_{i,d,l}$ increases on average.

B.4.5 3. Gaussian Processes with RBF Kernels:

Related studies focusing on robustness of CD methods for i.i.d. sample data Montagna et al. (2023a); Yi et al. (2024) primarily change the standard linear relationships to interactions modeled by Gaussian processes using Radial Basis Function (RBF) kernels. In our third family of functions for the violation $\mathbf{V}_{\text{nl,rbf}}$, we follow the same approach. Specifically, we use a Gaussian process prior $f(x) \sim \mathcal{GP}(0, \kappa_{\text{RBF}}(x, x'))$ with the RBF kernel $\kappa_{\text{RBF}}(x, x') = \exp\left(-\frac{(x-x')^2}{2\lambda^2}\right)$, where λ is the length scale which we set to one. In particular, we use the implementation of Pedregosa et al. (2011).

Further, we do not employ wrapping in this case as the mean zero of the Gaussian process ensures that the sampled functions do not diverge. Next, we describe how we stepwise increase the nonlinearity of the resulting SCM.

Stepwise Increasing Nonlinearity To stepwise increase the nonlinearity, we use a different approach for $\mathbf{V}_{\text{nl,rbf}}$. Specifically, we sample nonlinear links $f_{i,d,l}$ with an increasing probability from the Gaussian process. Here, we use the following probabilities: $\{0.2, 0.4, 0.6, 0.8, 1.0\}$. All remaining links in the causal graph G are linear and use the identity function in Eq. (2). Hence, we increase the nonlinearity of the overall SCM by increasing the likelihood of sampling nonlinear links. To be specific, for our two empirical scenarios $(D, L) = (5, 3)$ and $(D, L) = (7, 4)$, we have a maximum of $5 \times 5 \times (3 + 1)$ and $7 \times 7 \times (4 + 1)$ links, respectively, where D is the number of variables and L the number of lags (+1 for instantaneous). For lagged links, we employ the link probabilities $p_{\text{lag}} = 0.075$ or $p_{\text{lag}} = 0.15$, while for instantaneous links, we use $p_{\text{inst}} = 0.0$ and $p_{\text{inst}} = 0.1$ (see Eq. (6)).

We list the number of expected links for the eight combinations in Table 6. For each of these eight combinations, we can calculate the number of expected nonlinear links for the increasing intensities $\{0.2, 0.4, 0.6, 0.8, 1.0\}$ via a standard multiplication because they are sampled independently. This

	(D, L)	
	(5,3)	(7,4)
$(p_{\text{lag}}, p_{\text{inst}}) = (0.075, 0.0)$	5.625	14.7
$(p_{\text{lag}}, p_{\text{inst}}) = (0.075, 0.1)$	8.125	19.6
$(p_{\text{lag}}, p_{\text{inst}}) = (0.15, 0.0)$	11.25	29.4
$(p_{\text{lag}}, p_{\text{inst}}) = (0.15, 0.1)$	13.75	34.3

Table 6: Expected number of links in sampled SCMs. Here, D denotes the number of variables in $\boldsymbol{X}$ and L denotes the number of lags. The probabilities $(p_{\text{lag}}, p_{\text{inst}})$, correspond to the likelihood of lagged and instantaneous connections in the causal graphs G (i.e., nonzero elements in A and B, Eq. (6)).

leads to the following increasing numbers of expected nonlinear links for the smallest and sparsest scenario ($D = 5, L = 3, p_{\text{lag}} = 0.075, p_{\text{inst}} = 0.0$): $\{1.125, 2.25, 3.375, 4.5, 5.625\}$. Conversely, we get the following expected nonlinear links in the largest scenario ($D = 7, L = 4, p_{\text{lag}} = 0.15, p_{\text{inst}} = 0.1$): $\{6.86, 13.72, 20.58, 27.44, 34.3\}$. Hence, we conclude that sampling the SCM in this stepwise manner progressively increases the amount of nonlinear interactions.

4. Composite Functions Lastly, and inspired by symbolic regression, we sample the $f_{i,d,l}$ through a random hierarchical composition. First, we define a set of base functions $\mathfrak{B}$. Specifically, we implement $\{x^{1/3}, \tanh(x), \sinh^{-1}(x), \max(x, 0), x, x^2, |x|, \cosh(x), \sin(x), cos(x)\}$. Then, m independent chains $h^{(j)}(x)$ are formed, each by randomly selecting and sequentially composing N_β functions from $\mathfrak{B}$, i.e., $h^{(j)}(x) = b_{N_\beta}(\dots b_2(b_1(x))\dots)$. Finally, the results of the independent chains get multiplied by -1 with a probability of $1/2$, i.e., $c_{\text{flip}}^{(j)} \sim \text{Uniform}\{-1, 1\}$, before all chains get summed up to

$$f(x) = \sum_{j=1}^{m} c_{\text{flip}}^{(j)} \cdot h^{(j)}(x).$$

This construction allows for a wide range of possible, potentially highly nonlinear functions. Hence, we apply Eq. (8) to enforce stable behavior. In our empirical evaluation, we use two chains, each composed of two base functions uniformly sampled from $\mathfrak{B}$ to model $\mathbf{V}_{\text{nl,comp}}$. Next, we describe how we stepwise increase the nonlinearity of the resulting SCM.

Stepwise Increasing Nonlinearity For $\mathbf{V}_{\text{nl,comp}}$, we strictly follow the procedure also employed for $\mathbf{V}_{\text{nl,rbf}}$. Specifically, we increase the probability of sampling nonlinear interactions when generating the SCM. Again, we use the following probabilities: $\{0.2, 0.4, 0.6, 0.8, 1.0\}$ to scale the violation intensity. Table 6 summarizes the number of links for the various scenarios in our experiments, and the same calculations as for $\mathbf{V}_{\text{nl,rbf}}$ apply to estimate the expected number of nonlinear links. Hence, we conclude that sampling the SCM in this stepwise manner progressively increases the amount of nonlinear interactions.

B.5 Additional Details $\mathbf{V}_{\text{INNO}}$

Since the standard assumption of independent additive innovation noise is often violated in practice. Here, we detail our setup for evaluating the impact of such violations. Specifically, we consider three different paradigms: First, we discuss direct changes to the noise structures by introducing dependencies. Second, we shift the distribution from Gaussian to non-Gaussian variations. Lastly, we consider widely different variances leading to stronger variations between the variables X_i, which can be problematic Peters & Bühlmann (2014).

In the first set of violations, we test robustness against five alternative noise structures following our discussion of observational noise in Apx. B.1. To be specific, implement $\mathbf{V}_{\text{inno,mul}}$, $\mathbf{V}_{\text{inno,auto}}$, $\mathbf{V}_{\text{inno,com}}$, $\mathbf{V}_{\text{inno,time}}$, and $\mathbf{V}_{\text{inno,shock}}$.

Violation	Definition of $\epsilon_{i,t}$	Depends On
$\mathbf{V}_{\text{obs,mul}}$	$\epsilon_{i,t} = X_{i,t} \cdot \eta_{i,t}$	the signal $X_{i,t}$
$\mathbf{V}_{\text{obs,time}}$	$\epsilon_{i,t} = \eta_{i,t} \cdot (1 + \alpha t) \cdot \sin(2\pi t/\beta)$	time step t
$\mathbf{V}_{\text{obs,auto}}$	$\epsilon_{i,t} = \alpha \cdot \epsilon_{i,t-1} + (1 - \alpha) \cdot \eta_{i,t}$	autoregressive
$\mathbf{V}_{\text{obs,com}}$	$\forall i : \epsilon_{i,t} = \eta_t$	—
$\mathbf{V}_{\text{obs,shock}}$	$\epsilon_{i,t} \sim \begin{cases} S & \text{with prob. } p_{\text{shock}}, \\ 0 & \text{else.} \end{cases}$	fixed scalar S, shock prob. p_{shock}

Table 7: First set of innovation noise violations. In our experiments, the sources of randomness for the variables X_i are routed in the innovation noise $\epsilon_{i,t}$, which are typically additive and standard normal. Here, we include various dependencies by using random variables $\eta_{i,t}$ and η_t $\big($standard normal $\mathcal{N}(0, 1)\big)$, which are subsequently influenced by various factors, e.g., the signal strength ($\mathbf{V}_{\text{obs,mul}}$). Both α and β denote hyperparameters.

All of these structures change the distributions of the independent additive noise $\epsilon_{i,t}$ in Eq. (2). We list the specific distributions in Table 7. Regarding the corresponding hyperparameters, i.e., α, β, S, and p_{shock}, we use the same setting as for the observational variants (see Apx. B.1)

In contrast to scaling the SNR, we blend the different $\epsilon_{i,t}$ distributions with a decreasing amount of standard normal noise to intensify the five violations. This is important because the innovation noise is part of the signal (Eq. (2)). In particular, we use alpha blending, i.e.,

$$\alpha\epsilon_{i,t} + (1 - \alpha)\varepsilon_{i,t}, \tag{30}$$

where $\varepsilon_{i,t} \sim \mathcal{N}(0, 1)$. We use the following increasing α values in our experiments: $\{0.1, 0.25, 0.5, 0.75, 0.85\}$

For the second set of innovation noise violations, i.e., $\mathbf{V}_{\text{inno,uni}}$ and $\mathbf{V}_{\text{inno,weib}}$, we progressively blend non-Gaussian distributed noise with standard normal noise. Let $\omega \sim \Omega_{ng}$ be a non-Gaussian random variable and let $\psi \sim \mathcal{N}(0, 1)$ be standard normal noise. Then, we define $\epsilon_{i,t}$ as

$$\epsilon_{i,t} = \frac{(1 - \alpha)(\omega - \mathbb{E}[\omega]) + \alpha\psi}{\sqrt{\text{var}(\omega)(1 - 2\alpha) + \alpha^2(\text{var}(\omega) + 1)}}, \tag{31}$$

where α is a blending parameter. To stepwise scale the intensity, we use the following blending values for α: $\{0.95, 0.75, 0.5, 0.25, 0.0\}$.

The denominator in Eq. (31) and substracting $\mathbb{E}[\omega]$ in the numerator ensure that $\mathbb{E}[\epsilon_{i,t}] = 0$ and $\text{var}(\epsilon_{i,t}) = 1$. To verify this, note that the expectation is linear and the denominator is a constant factor for a fixed ω. Hence, it is enough to analyze the numerator. Here, $\mathbb{E}[\omega] - \mathbb{E}[\mathbb{E}[\omega]] = \mathbb{E}[\omega] - \mathbb{E}[\omega] = 0$ and $\mathbb{E}[\psi]$, which directly implies $\mathbb{E}[\epsilon_{i,t}] = 0$. Further, to show that the denominator scales the mixture to unit variance, we have to show that it is equivalent to the standard deviation of the numerator. Given that the standard deviation is the square root of the variance, it is enough to show that the variance of the numerator is equal to the squared denominator. Crucially, both ω and ψ are independent random variables, which means their covariance is zero. Hence, $\text{var}((1 - \alpha)(\omega - \mathbb{E}[\omega]) + \alpha\psi)$

$$\begin{aligned} &= (1 - \alpha)^2\text{var}(\omega) + \alpha^2\text{var}(\psi) \\ &= (1 - 2\alpha + \alpha^2)\text{var}(\omega) + \alpha^2 1 \\ &= \text{var}(\omega)(1 - 2\alpha) + \alpha^2(\text{var}(\omega) + 1), \end{aligned}$$

i.e., the squared denominator in Eq. (31). A direct consequence of this is that $\text{var}(\epsilon_{i,t}) = 1$.

In our experiments, we use two different non-Gaussian distributions to model the violations $\mathbf{V}_{\text{inno,uni}}$ and $\mathbf{V}_{\text{inno,weib}}$. In the first case, for $\mathbf{V}_{\text{inno,uni}}$, we use a uniform distribution over the interval $[-2, 2]$ with the corresponding density

$$p_{\text{Uniform}}(x) = \mathbf{1}_{[-2,2]}(x) \cdot \frac{1}{4}, \tag{32}$$

where $\mathbf{1}_{[-2,2]}$ is a unit function that is equal to one iff $x \in [-2, 2]$.

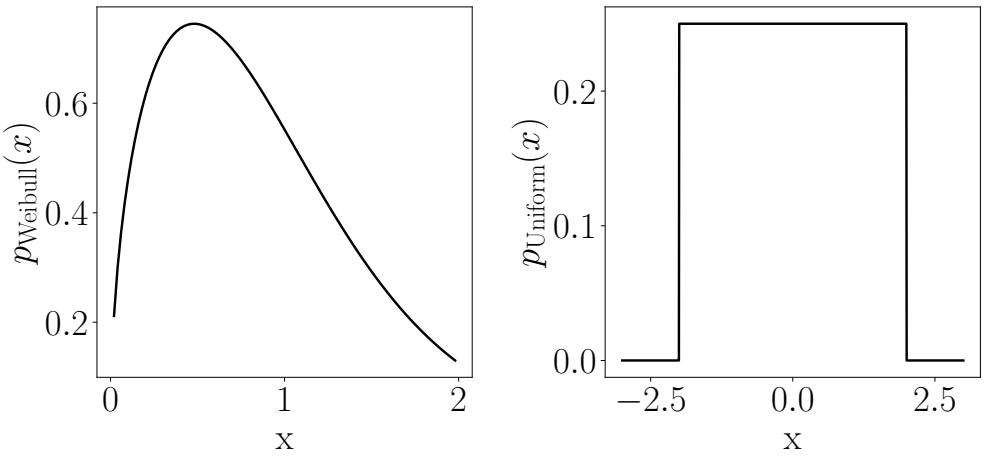


Figure 14: The densities of the two non-Gaussian distributions, we employ to violate standard normal Gaussian innovation noise. Specifically, we employ a Weibull distribution Weibull (1939) with scale $\lambda = 1$ and shape $a = 1.5$ and a uniform distribution over the interval $[-2, 2]$.

In the second case, for $\mathbf{V}_{\text{inno,weib}}$, we employ a Weibull distribution Weibull (1939). Such a distribution is described by two parameters: a scale λ, which we set to one, and shape a, where we use 1.5. The corresponding density is defined as

$$p_{\text{Weibull}}(x) = \frac{a}{\lambda}\left(\frac{x}{\lambda}\right)^{a-1} e^{-(x/\lambda)^a}. \tag{33}$$

We visualize the densities of both non-Gaussian distributions in Fig. 14

Lastly, as the final, separate violation concerning innovation noise, we model a high variance for the different variables $\mathbf{V}_{\text{inno,var}}$, which can be problematic Peters & Bühlmann (2014). We implement this violation by sampling one variance σ_i^2 for each variable X_i at the beginning of the sampling process, which is then used in each timestep to draw $\epsilon_{i,t} \sim \mathcal{N}(0, \sigma_i^2)$. In particular, we sample the σ_i^2 uniformly from an interval. Now, to stepwise intensify $\mathbf{V}_{\text{inno,var}}$, we increase the length of these intervals. Specifically, we use the following order of intervals: $[0.5, 1]$, $[0.1, 1]$, $[0.1, 2]$, $[0.1, 4]$, and $[0.1, 8]$ to model the widening uniform distributions.

B.6 Additional Details $\mathbf{V}_{\text{STAT}}$

Stationarity assumes that the structural assignments in Eq. (6) do not change during the generation/measurement process. To violate this assumption, we keep the nonzero entries in A, but resample the coefficients during the generation of the time series. This approach models changes in the SCM, and we scale $\mathbf{V}_{\text{stat}}$ by increasing the number of change points N_{change}. Specifically, we introduce one, three, five, seven, or nine changes to the structural assignments. To change the nonzero coefficients $A_{i,d,l}$, i.e., the causal skeleton, we uniformly sample an additive change from Uniform$(-0.6, 0.6)$.

In our experiment, we set the number of time steps T either to 250 or 1000. Hence, to introduce $N_{\text{change}} \in \{1, 3, 5, 7, 9\}$ change points, we use the time steps denoted in Table 8, respectively.

B.7 Additional Details $\mathbf{V}_{\text{LENGTH}}$

To reliably estimate relationships and identify patterns, CD algorithms need a sufficient number of samples. To violate this necessity ($\mathbf{V}_{\text{length}}$), we reduce the number of time steps T we sample from Eq. (6). Specifically, we employ the following five discrete levels $T \in \{200, 100, 50, 25, 12\}$.

N_{change}	Selected Time Steps $T = 250$	$T = 1000$
1	125	500
3	100, 125, 150	400, 500, 600
5	75, 100, 125, 150, 175	300, 400, 500, 600, 700, 800
7	50, 75, 100, 125, 150, 175, 200	200, 300, 400, 500, 600, 700, 800
9	25, 50, 75, 100, 125, 150, 175, 200, 225	100, 200, 300, 400, 500, 600, 700, 800, 900

Table 8: The specific time steps we use for the respective number of change points to violate stationarity ($\mathbf{V}_{\text{stat}}$). We separate the time steps for the two settings $T = 250$ and $T = 1000$ in our experiments.

B.8 Additional Details $\mathbf{V}_{\text{Q}}$

In practical applications, data quality cannot always be controlled, leading to various degradations beyond observational noise. We investigate two quality violations that are common in various domains, i.e., sensor failures $\mathbf{V}_{\text{q,empty}}$ and missing values $\mathbf{V}_{\text{q,missing}}$. They differ because in the former scenario, we change the parent set of variables to $\varnothing$ during the generation of $\boldsymbol{X}$, while in the latter, we remove and linearly interpolate periods of measurements after generation.

To scale $\mathbf{V}_{\text{q,missing}}$, we increase the probability to delete observations $X_{i,t}$ completely at random Heitjan & Basu (1996). Specifically, we use $p_{\text{remove}} \in \{0.2, 0.35, 0.5, 0.65, 0.8\}$ before we use linear interpolation to fill $\boldsymbol{X}$ again.

$T = 250$			$T = 1000$			
Length	Ratio	Empty Periods	Length	Ratio	Empty Periods	Avg. Ratio
50	2× 0.2	(50, 100) and (150, 200)	300	2× 0.3	(100, 400) and (600, 900)	0.25
75	2× 0.3	(25, 100) and (150, 225)	390	2× 0.39	(50, 440) and (560, 950)	0.345
90	2× 0.36	(20, 110) and (140, 230)	440	2× 0.44	(40, 480) and (520, 960)	0.4
100	2× 0.4	(20, 120) and (130, 230)	450	2× 0.45	(40, 490) and (510, 960)	0.425
110	2× 0.44	(10, 120) and (130, 240)	470	2× 0.47	(20, 490) and (510, 980)	0.455

Table 9: The specific time periods, denoted by (start, end), where we set the parent sets to $\varnothing$ during generation of $\boldsymbol{X}$, i.e., $\mathbf{V}_{\text{q,empty}}$. We separate the time steps for the two settings $T = 250$ and $T = 1000$ in our experiments. In all cases, we introduce two periods with no causal signal and scale the length to increase intensity. We denote the average ratio of empty periods for sampled time series in the last column. Note that this ratio, i.e., the violation intensity, increases with each row.

To control the intensity of $\mathbf{V}_{\text{q,empty}}$, we increase the length of periods, where we remove the causal signal during the generation of $\boldsymbol{X}$. Specifically, we introduce two such periods per sampled time series and list the concrete intervals in Table 9. Crucially, the average ratio of each of the empty periods during the $T = 250$ or $T = 1000$ time steps increases as follows: $\{0.25, 0.345, 0.4, 0.425, 0.455\}$.

B.9 Additional Details $\mathbf{V}_{\text{SCALE}}$

Related works suggest that synthetically generating data introduces artifacts beneficial for identifying causal order, e.g., Ormaniec et al. (2025). This phenomenon is problematic because it can lead to an overestimation of a CD method's efficacy. Because it can be remedied using standardization to mean zero and variance one Reisach et al. (2021); Kaiser & Sipos (2021), we violate this condition by mixing the original observations $\boldsymbol{X}$ with its standardized version $\overline{\boldsymbol{X}}$ after generating the time series.

In particular, in $\overline{\boldsymbol{X}}$ all variables X_i are standardized over the time steps t. Then, the observations $\hat{\boldsymbol{X}} = \alpha\overline{\boldsymbol{X}} + (1 - \alpha)\boldsymbol{X}$, where α determines the intensity of $\mathbf{V}_{\text{scale}}$. Specifically, in our investigation, we use $\alpha \in \{0.0, 0.5, 0.7, 0.9, 1.0\}$.

	Cross Correlation	CausalPretraining	GVAR	Varlingam	PCMCI	PCMCI+	Dynotears	NTS-NOTears
Temporal Dynamics								
Lagged Effects	✓	✓	✓	✓	✓	✓	✓	✓
Instantaneous Effects	✗	✗	✗	✓	✗	✓	✓	✓
Observational noise	✗	✗	✗	✗	✗	✗	✗	✗
Hidden confounding	✗	✗	✗	✗	✗	✗	✗	✗
Unfaithfulness	✓	†	✓	✓	✗	✗	✓	✓
Nonlinearity	✗	✓	✗	✗	✓*	✓*	✗	✓
Innovation noise								
Non-Additive noise	✗	✗	✗	✗	✗	✗	✗	✗
Gaussian additive noise	✓	✓	✓	✗	✓	✓	✓	✓
NonStationarity	✗	✗	✗	✗	✗	✗	✗	✗

Table 10: Comparison of core assumptions and capabilities of selected causal discovery algorithms. Note, PCMCI and PCMCI+ can handle both linearity and nonlinearity. We, however, only test these methods with a linear conditional independence test in this study (partial correlation and robust partial correlation). Therefore, we mark their ability to handle nonlinearity with a *. We found no information on the faithfulness assumption for CausalPretraining and therefore mark it with †. Finally, we omit data quality issues such as V_{length}, V_q, or V_{scale} from this table as they are typically not mentioned as explicit assumptions.

C Appendix — Experimental Setup

C.1 Sampling Details

In this section, we describe the data generation process that we use throughout our experiments and for all violations. Generally, we base all of our experiments on Eq. (6) and alter it according to the violations described in Apx. B. We employ two combinations of the number of variables D and the maximum lags L, resulting in a small and a big scenario. Specifically, we set $(D, L) = (5, 3)$ or $(D, L) = (7, 4)$, respectively. Then, with a probability $p_{\text{lag}} \in \{0.075, 0.15\}$ and probability $p_{\text{inst}} \in \{0.0, 0.1\}$ links in A and B are being selected to be nonzero. This leads to eight "data regime" combinations, and we list the expected number of links in Table 6. Next, each nonzero element in A and B receives a value that is uniformly sampled from the joint interval $[-0.5, -0.3] \cup [0.3, 0.5]$. Notably, we explicitly exclude coefficients close to 0 to render causal relationships detectable. If f is not the identity function (i.e., for $\mathbf{V}_{\text{nl}}$), a univariate function is drawn from the corresponding distribution. We then generate $\boldsymbol{X}$ iteratively using Eq. (6). To initialize, we sample every variable from $\big(\mathcal{N}(0, 1)\big)$.

Before we start to generate $\boldsymbol{X}$, we need to evaluate the following two conditions: First, concerning instantaneous coefficients B, we guarantee the sample graph to be acyclic by checking the following necessary and sufficient condition for acyclicity:

$$\operatorname{tr}(e^B) \stackrel{!}{=} D, \tag{34}$$

where tr is the trace operator. If this condition is not met, we resample B until it passes. To account for potential divergence of the SCM, we test the VAR stability of A. In particular, we investigate whether it is stationary by evaluating the eigenvalues of its companion matrix F:

$$F = \begin{bmatrix} A_{t-1} & A_{t-2} & \dots & A_{t-3} & A_{t-4} \\ I_D & 0 & \dots & 0 & 0 \\ 0 & I_D & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & I_D & 0 \end{bmatrix} \tag{35}$$

where I_D is a $D \times D$ identity matrix. To guarantee the stationarity of the corresponding process, all eigenvalues of F have to lie in the complex unit circle. This condition applies if

$$\max_i |\lambda_i(F)| < 1. \tag{36}$$

If it does not hold for the sampled A, we resample the coefficients.

However, if the $f_{i,d,l}$ are nonlinear, then the VAR stability test does not apply. Hence, we additionally check for divergence with the following two tests: First, while generating $\boldsymbol{X}$ we continuously check

whether any variable in X is monotonically increasing over the last $\mathcal{T}$ time steps by testing:

$$\exists i \in \{1, \dots, d\}, \exists t \quad \text{s.t.} \quad \forall k \in \{0, \dots, \mathcal{T}\}, \\ |X_{i,t-k-1}| < |X_{i,t-k}| \tag{37}$$

If this condition is met, we halt the generation process and resample a new SCM. In our experiments, we set $\mathcal{T} = 10$.

Second, we test whether any time series in X holds values higher than a maximum value (likely indicating divergent processes). In our experiments, this value is set to ± 25. Again, if this condition is met, we halt the generation process and resample a new SCM.

For each violation intensity and data regime, we sample 100 random SCMs along with a corresponding X to calculate a single AUROC score. As discussed in our main paper, a "data regime" is a combination of D, L, p_{lag}, and p_{inst} (compare Table 6). Further, we vary the length of the time series $T \in \{250, 1000\}$. In summary, this results in $2 \times 2 \times 2 \times 2 \times 100 = 1600$ SCMs per individual violation intensity. Considering that we evaluate 5 stepwise violation intensities, we sample 8000 SCMs to evaluate the robustness of the eight CD methods for each of the 27 violations contained in Table 2, 4000 each for the small ($(D, L) = (5, 3)$) and big scenarios ($(D, L) = (7, 4)$).

C.2 Causal Discovery Methods Details

We include details on the method assumptions of all causal discovery methods involved in Table 10. Note, many of the data quality assumptions that we test, such as $\mathbf{V}_{\text{length}}$, $\mathbf{V}_{\text{q}}$, or $\mathbf{V}_{\text{scale}}$, are not explicitly assumed by most methods, however they are nonetheless often implicitly modeled in the synthetic data that is used for testing algorithm performance.

C.3 Hyperparameter Search Spaces

In Table 11, we include a list of the full hyperparameter space that we evaluated for each causal discovery method used throughout this paper.

C.4 Ensemble Training

To train our examples, we generate a separate training dataset that holds SCMs and corresponding X from all violations, respective intensities, and data regimes. These samples are combined into a single joint training dataset that we use to train Ensembles with trainable parameters. In the most general sense, all our ensembles take a tensor of the shape $B \times D \times D \times L_{model} \times M$, where M denotes the number of individual CD methods, D the number of variables, L_{model} the model order and B the batch size. This tensor is then, if necessary, reshaped to match the first layer of the respective network architectures ($\text{Ensemble}_{\text{Linear}}$, $\text{Ensemble}_{\text{MLP}}$, and $\text{Ensemble}_{\text{MLP}}$). All network architectures return a $B \times D \times D \times L_{model}$ tensor that is directly used as the final predicted graph G. Notably, for the $\text{Ensemble}_{\text{Mean}}$ and $\text{Ensemble}_{\text{Pareto}}$ we directly recombine elements in the input tensor by either taking the average over the model dimension M or selecting the optimal element. $\text{Ensemble}_{\text{Linear}}$ is implemented as a single fully-connected layer without an activation function. For $\text{Ensemble}_{\text{MLP}}$, we use a 3-layer MLP with RELU activation functions. For $\text{Ensemble}_{\text{ConvMixer}}$ we use a standard ConvMixer architecture Trockman & Zico Kolter (2022) where we set the input channels to D and the hidden dimension to $D \times D \times L_{model}$. Further, we evaluate the hyperparameters specified in Table 12 to select the best model that we report in Fig. 3a:

C.5 Reproducability and Computational resources

To facilitate the reproduction of our results, we have made all code and necessary resources available in the TCD-Arena repository. This repository includes seeded functionality to generate the datasets used in this paper, along with hashing functions to verify their integrity. The code for all evaluated Causal Discovery methods, ensembling approaches, and the scripts to generate the figures presented are also included. Experiments were conducted on a 7-node Slurm cluster using CPU cores, with the exception of ensemble training, which was performed on a single Nvidia RTX 3090 GPU. While most individual experiments are not resource-intensive, reproducing the complete set of approximately 50 million causal discovery attempts will require a multi-day runtime on a comparable setup. However,

Method (Combos)	**Parameters**	**Values**
Cross Correlation (3)	L_{model}	$L-2, L, L+2$
CausalPretraining (2)	Architecture	TRF, GRU
Varlingam (6)	L_{model}	$L-2, L, L+2$
	Prune	True, False
GVAR	L_{model}	$L-2, L, L+2$
	Use	coeff, p-val
PCMCI (6)	L_{model}	$L-2, L, L+2$
	CI test	ParC, RParC
PCMCI+ (12)	L_{model}	$L-2, L, L+2$
	CI test	ParC, RParC
	RLL	True, False
NTS-NOTears (48)	L_{model}	$L-2, L, L+2$
	h-tol	1e-60, 1e-10
	Rho-max	1e+16, 1e+18
	Lambda1	0.005, 0.001
	Lambda2	0.01, 0.001
Dynotears (48)	L_{model}	$L-2, L, L+2$
	Lambda-w	0.1, 0.3
	Lambda-a	0.1, 0.3
	Max iter	100, 40
	H-tol	1e-8, 1e-5

Table 11: Hyperparameter space and number of combinations in the hyperparameter grid. For NTS-NOTears and Dynotears, we use default parameters (first value) and an alternative value per HP. ParC and RParC denote the Partial Correlation conditional independence test and the Robust Partial Correlation conditional independence test, respectively. RLL denotes the resetting of the lagged links before calculating instantaneous effects. We refer to the implementations of all methods in TCD-Arena for further details.

as many scripts can be executed in parallel on a Slurm cluster, the total runtime may vary depending on the specific hardware and configuration.

Category	Hyperparameter	Evaluated Values
Common to all architectures		
General	Batch Size	{16, 128}
	Loss Function	{BCE, MSE}
Optimizer	Learning Rate	{1e-4, 1e-2}
Base Model: Linear		
Base Model: MLP		
Architecture	Hidden Layer 1	{264, 512, 686, 1360}
	Hidden Layer 2	{128, 264, 392, 792}
Base Model: ConvM		
Regularization	Dropout Rate	{0.0, 0.1}
Architecture	Depth	{3, 6}
	Kernel Size	{4, 8}
	Patch Size	{1, 3}

Table 12: Summary of hyperparameters that were evaluated for CD ensembling.

D Appendix — Additional Results

D.1 Additional Metrics

To extend our empirical evaluation, we include additional metrics of robustness quantification in Fig. 15 - Fig. 18. As potential metrics are vast, we additionally include the raw AUROC scores in our repository. Generally, we find that most metrics show a similar picture (e.g., the ordering of methods for when looking at performance metrics of G^{INST} or the generally low performance of Cross Correlation and CausalPretraining. However, some small differences are notable, e.g., the robustness evaluation of PCMCI and PCMCI+ against $\mathbf{V}_{\text{obs,shock}}$. Here, the average F1-max score suggests a superior performance of PCMCI, which is not perceptible under average AUROC scores. We view this observation as evidence that there are more fine-grained differences in robustness that still need to be uncovered.

D.2 Discussion on Metric Failure Cases

In this study, we fundamentally quantify the robustness of a method against an assumption violation based on a limited number of samples (five violation intensities). As the underlying robustness is often continuous, we deem it reasonable to discuss under which conditions our methodological approach provides potentially misleading results. For this, we depict three simplified scenarios in Fig. 19.

In the first scenario (green box), we find a consistent separation between the robustness of different methods. In this ideal scenario, the green curve consistently maintains a higher performance than the blue curve throughout the violation range. The discrete measurements (marked by stars) accurately capture this relationship, leading to a faithful robustness score. In the second scenario (blue box), the curves cross. The discrete sampling points suggest that the performance of both curves is roughly equal across the measured violation levels, resulting in a very similar robustness score. However, this discrete evaluation fails to account for the precise dynamics of the robustness curves. Notably, depending on the application, either the blue or the green curve could be preferable. In the third scenario, a non-monotonic and highly volatile curve highlights the most critical risk. At the discrete evaluation points, the blue curve is preferred even though its robustness is disputable.

D.3 Analysis of Individual Violations

To contextualize the results in our main paper, we highlight a couple of individual results that we find worth mentioning. Generally, we find that a few violations favor distinct method archetypes

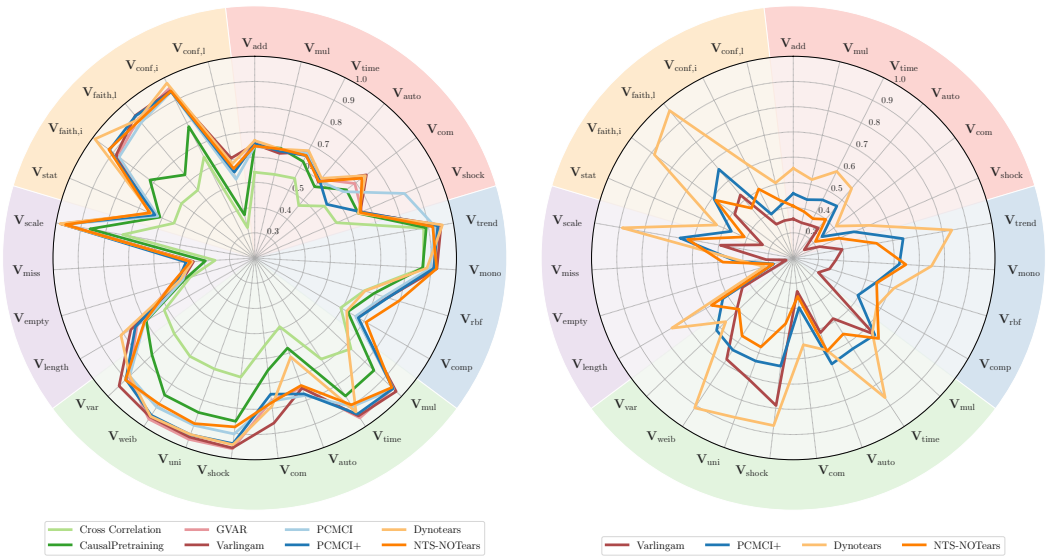


Figure 15: Robustness profiles of eight Causal Discovery algorithms against a multitude of stepwise assumption violations measured as **average maximum F1** over various data regimes. TOP: results for G^{WCG}. Bottom: results for G^{INST}. Colors specify: **Observational noise:** , **Nonlinearity:** , **Innovation noise:** , **Graph structures:** , **Data representation:** .

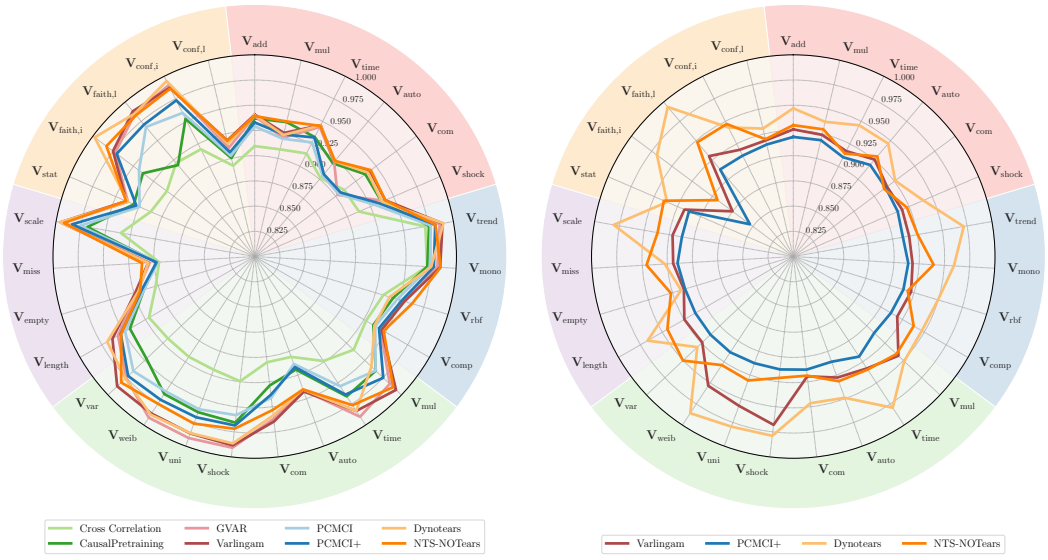


Figure 16: Robustness profiles of eight Causal Discovery algorithms against a multitude of stepwise assumption violations measured as **average maximum Accuracy** over various data regimes. TOP: results for G^{WCG}. Bottom: results for G^{INST}. Colors specify: **Observational noise:** , **Nonlinearity:** , **Innovation noise:** , **Graph structures:** , **Data representation:** .

when uncovering G^{WCG} (Fig. 1). Concerning $\mathbf{V}_{\text{obs,com}}$, score-based approaches (PCMCI,PCMCI+) seem to be less robust. On the other hand, for $\mathbf{V}_{\text{q,missing}}$ they seem to be favored. Further, Dynotears seems to struggle heavily with $\mathbf{V}_{\text{inno,auto}}$ and drops even below baseline performance Finally, while CausalPretraining and Cross Correlation are generally not as robust to various violations, their difference towards all other methods is especially pronounced for $\mathbf{V}_{\text{conf,lag}}$. Concerning the uncovering of G^{INST} (Fig. 2b), we find that Dynotears and PCMCI+ are quite robust, while the other methods are not able to reach similar violation robustness. With this, we want to conclude again that the most

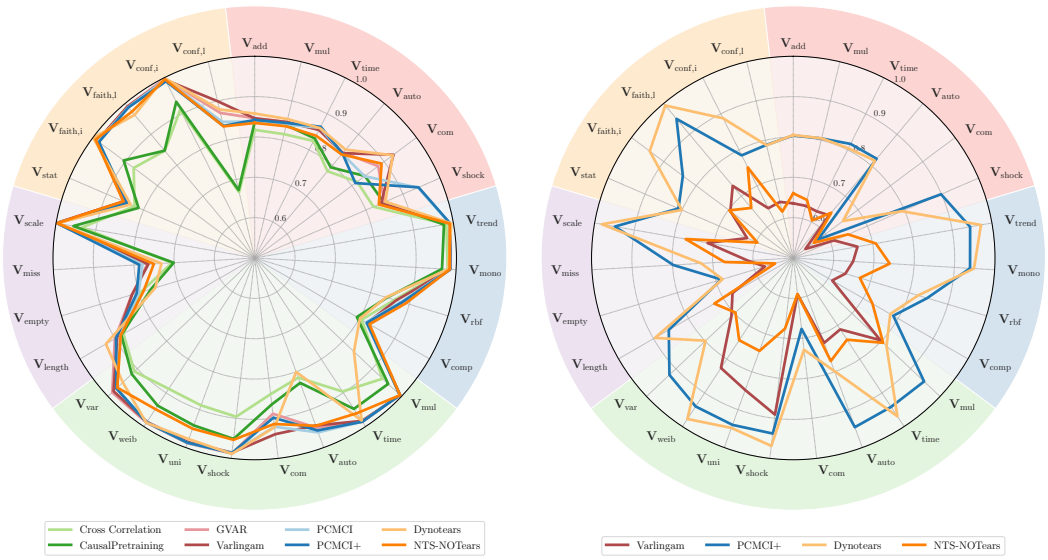


Figure 17: Robustness profiles of eight Causal Discovery algorithms against a multitude of stepwise assumption violations measured as **average AUROC** over various data regimes. **For each method, we here report the optimal hyperparameters individually selected for each violation.** TOP: results for G^{WCG}. Bottom: results for G^{INST}. Colors specify: **Observational noise:** , **Nonlinearity:** , **Innovation noise:** , **Graph structures:** , **Data representation:** .

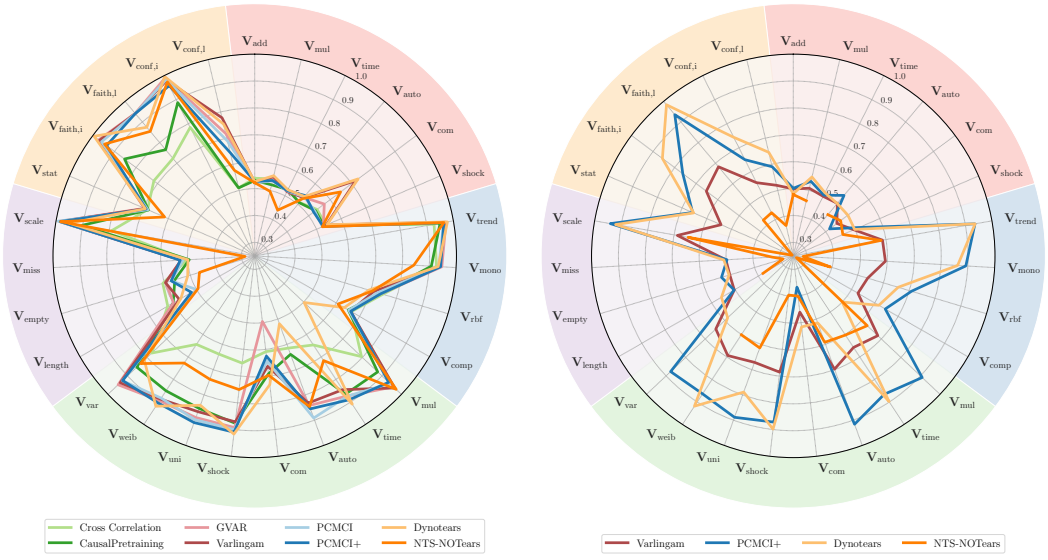


Figure 18: Robustness profiles of eight Causal Discovery algorithms against a multitude of stepwise assumption violations measured as **mean AUROC over the highest violation level**. TOP: results for G^{WCG}. Bottom: results for G^{INST}. Colors specify: **Observational noise:** , **Nonlinearity:** , **Innovation noise:** , **Graph structures:** , **Data representation:** .

important insight of our empirical investigation is that different causal discovery methods are to be preferred under specific assumption violation scenarios. This has direct implications concerning real-world applications as it suggests that a clever selection of CD methods, be it by hand or through ensembling, can improve the confidence in uncovering a latent SCM.

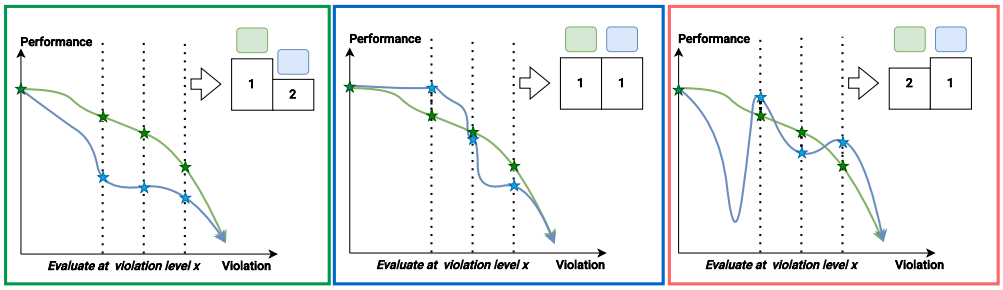


Figure 19: Depiction of problematic relationships between violation property and robustness measured as average AUROC for a single data regime and a single method configuration. Left: Optimal case in which all performance curves are monotonically decreasing and one curve is Pareto superior. Middle: While both curves are monotonically decreasing, our metric does not directly distinguish between them. Right: If any performance curve is highly non-monotonic, the comparison can be misleading.

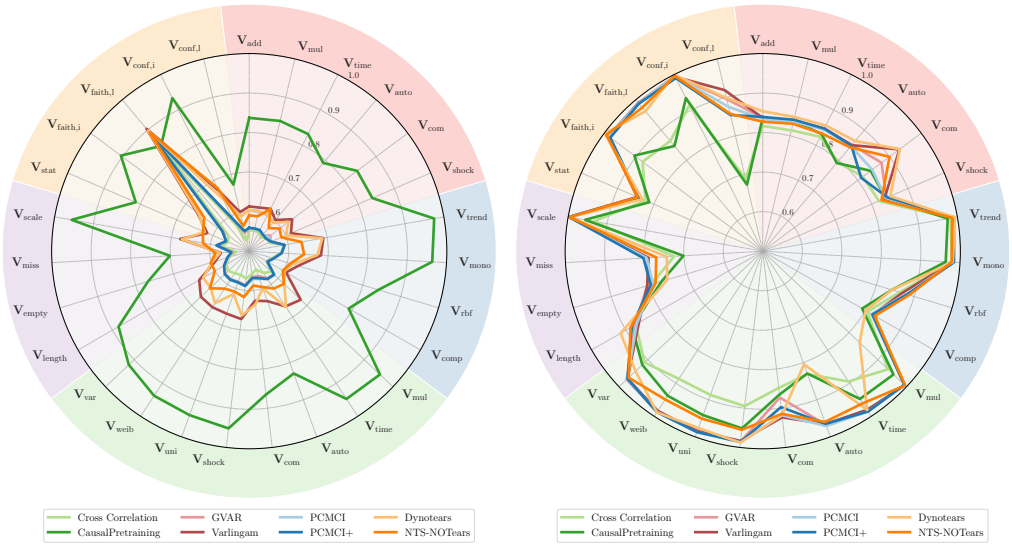


Figure 20: Robustness profiles of eight Causal Discovery algorithms against a multitude of stepwise assumption violations measured as average AUROC over various data regimes and under model misspecification ($L_{model} \neq L$). Left: $\downarrow L$ regime, Right: $\uparrow L$ regime. Colors specify: **Observational noise:** , **Nonlinearity:** , **Innovation noise:** , **Graph structures:** , **Data representation:** .

D.4 Visualizations of Misspecified Models

In Fig. 20 we visualize robustness profiles for misspecified modelling parameters, i.e., $L_{model} \neq L$. While under $\downarrow L$, the robustness of all methods decreases visibly, under $\uparrow L$, changes are negligible.

D.5 Hyperparameter Sensitivity

To provide a full picture of the hyperparameter sensitivity of all CD methods, we include graphics that depict the relationship between violation severity and performance for all combinations of hyperparameter and data regime per method. Fig. 21-Fig. 23 contain the sensitivities for lagged effects, while Fig. 24 and Fig. 25 depict methods estimating instantaneous links.

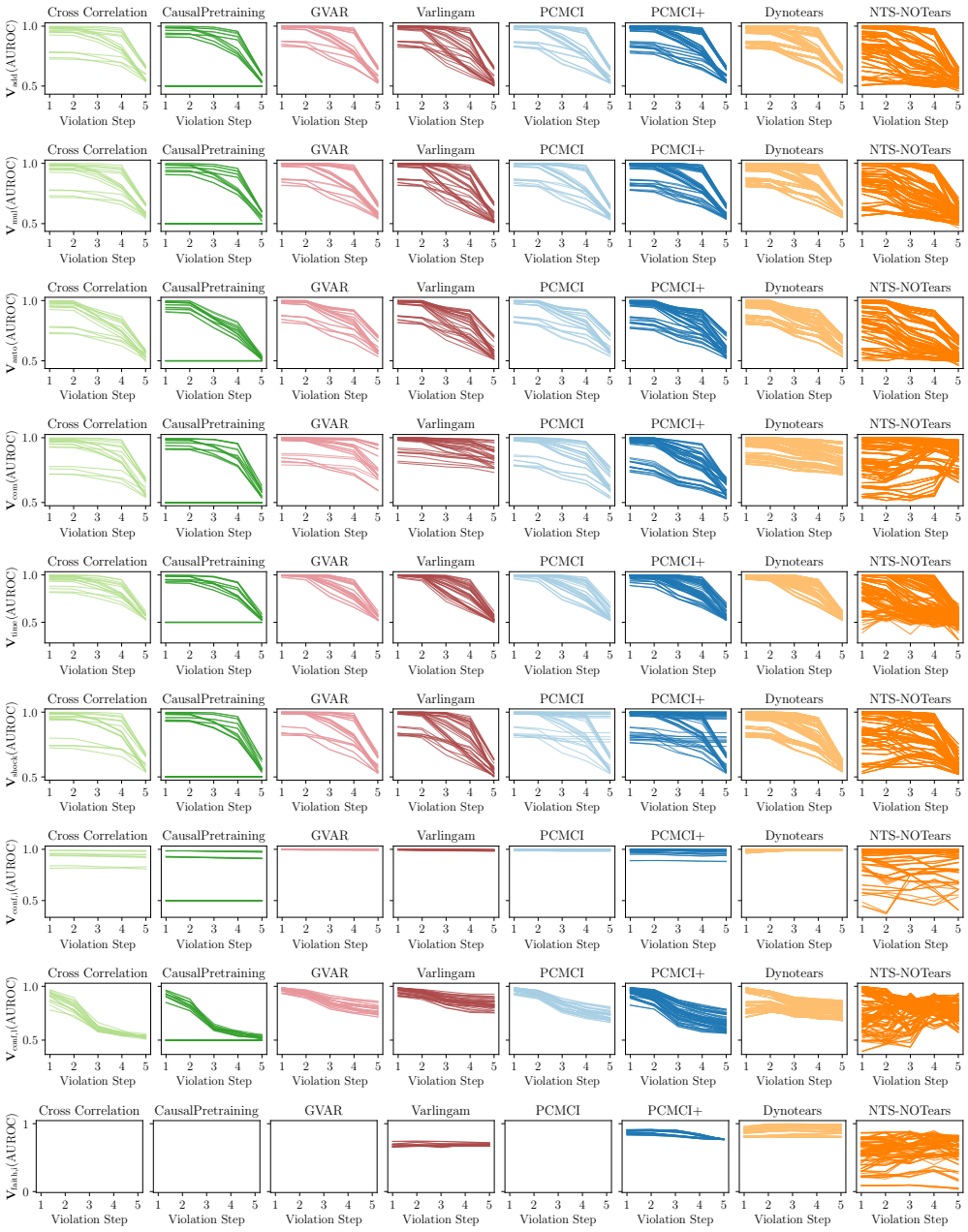


Figure 21: Hyperparameter sensitivities for $\mathbf{V}_{\text{obs,add}}$, $\mathbf{V}_{\text{obs,mul}}$, $\mathbf{V}_{\text{obs,auto}}$, $\mathbf{V}_{\text{obs,com}}$, $\mathbf{V}_{\text{obs,time}}$, $\mathbf{V}_{\text{obs,shock}}$, $\mathbf{V}_{\text{conf,inst}}$, $\mathbf{V}_{\text{conf,lag}}$, and $\mathbf{V}_{\text{faith,inst}}$.

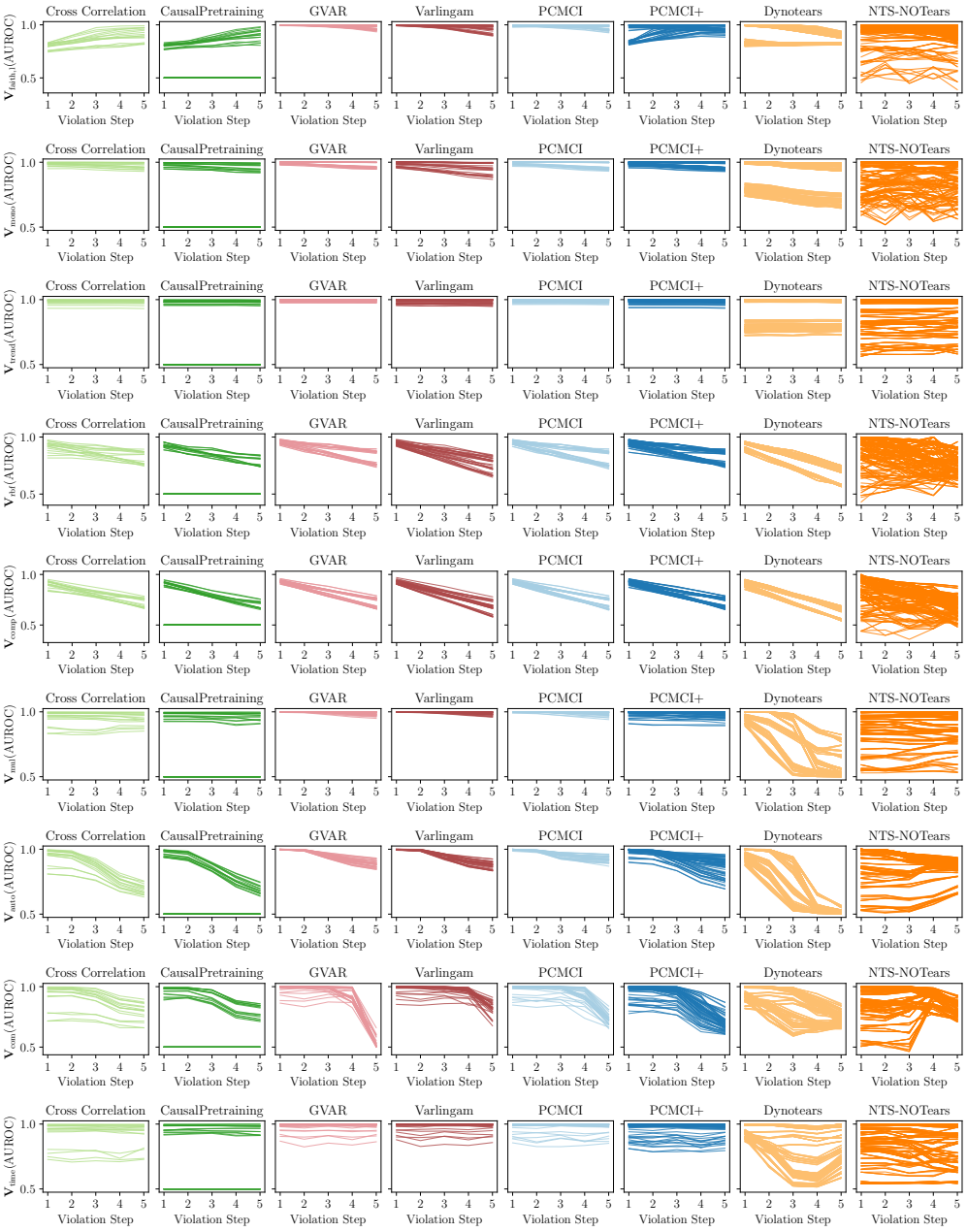


Figure 22: Hyperparameter sensitivities for $\mathbf{V}_{\text{faith,lag}}$, $\mathbf{V}_{\text{nl,mono}}$, $\mathbf{V}_{\text{nl,trend}}$, $\mathbf{V}_{\text{nl,rbf}}$, $\mathbf{V}_{\text{nl,comp}}$, $\mathbf{V}_{\text{inno,mul}}$, $\mathbf{V}_{\text{inno,auto}}$, $\mathbf{V}_{\text{inno,com}}$, and $\mathbf{V}_{\text{inno,time}}$.

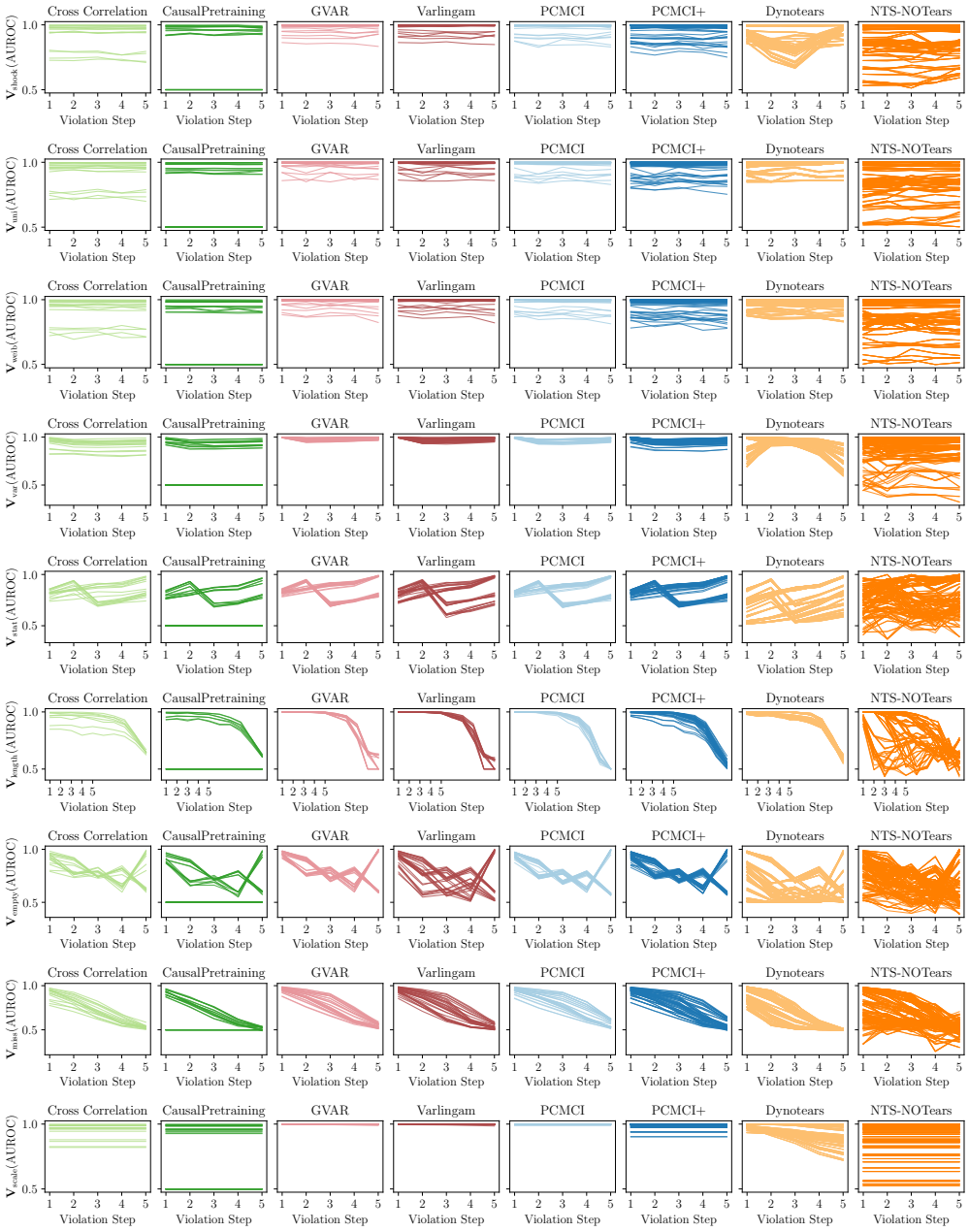


Figure 23: Hyperparameter sensitivities for $\mathbf{V}_{\text{inno,shock}}$, $\mathbf{V}_{\text{inno,uni}}$, $\mathbf{V}_{\text{inno,weib}}$, $\mathbf{V}_{\text{inno,var}}$, $\mathbf{V}_{\text{stat}}$, $\mathbf{V}_{\text{length}}$, $\mathbf{V}_{\text{q,missing}}$, $\mathbf{V}_{\text{q,empty}}$, and $\mathbf{V}_{\text{scale}}$.

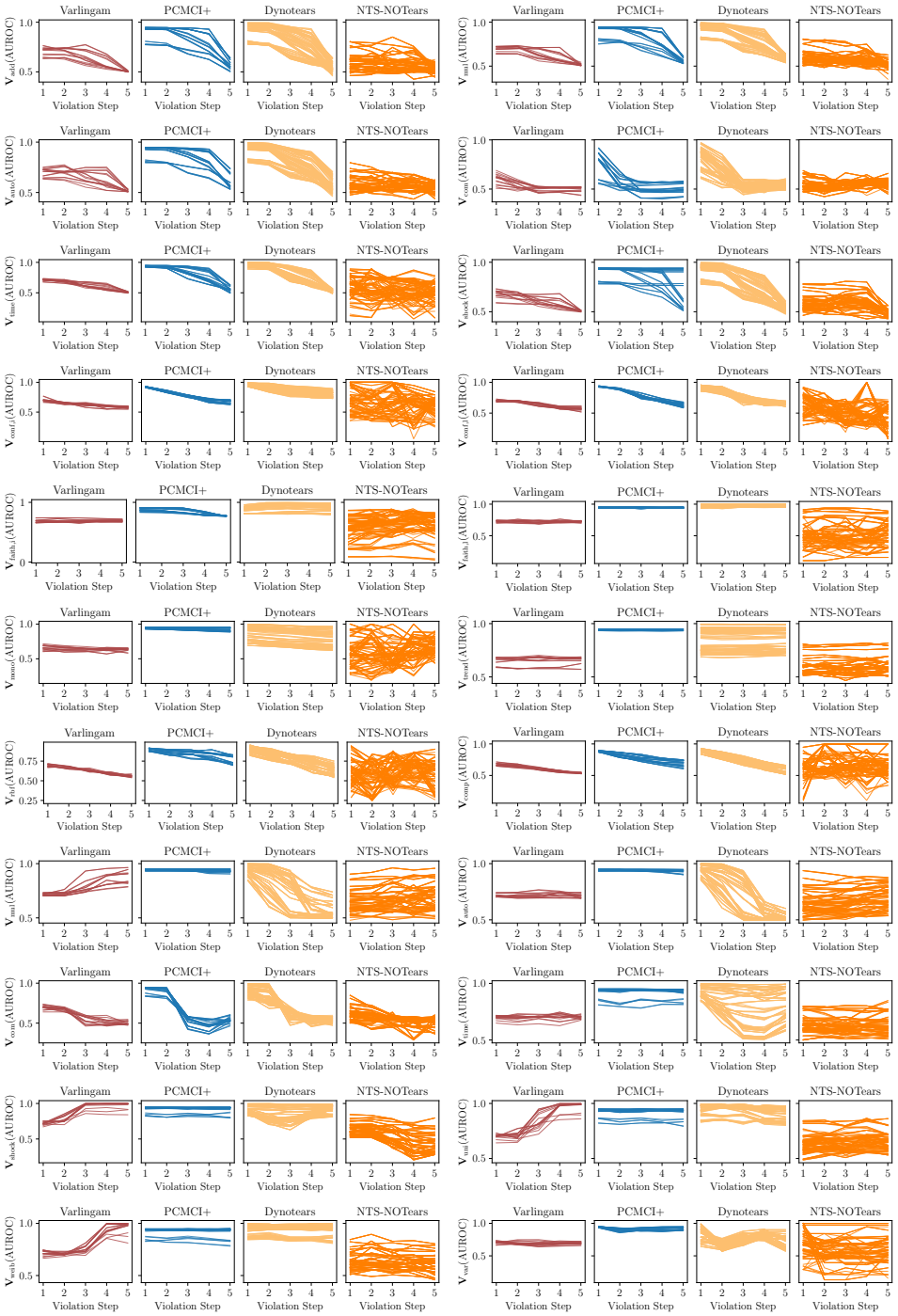


Figure 24: Hyperparameter sensitivities for all methods estimating instantaneous effects. The order from left to right, top to bottom is $\mathbf{V}_{\text{obs,add}}$, $\mathbf{V}_{\text{obs,mul}}$, $\mathbf{V}_{\text{obs,auto}}$, $\mathbf{V}_{\text{obs,com}}$, $\mathbf{V}_{\text{obs,time}}$, $\mathbf{V}_{\text{obs,shock}}$, $\mathbf{V}_{\text{conf,inst}}$, $\mathbf{V}_{\text{conf,lag}}$, $\mathbf{V}_{\text{faith,inst}}$, $\mathbf{V}_{\text{faith,lag}}$, $\mathbf{V}_{\text{nl,mono}}$, $\mathbf{V}_{\text{nl,trend}}$, $\mathbf{V}_{\text{nl,rbf}}$, $\mathbf{V}_{\text{nl,comp}}$, $\mathbf{V}_{\text{inno,mul}}$, $\mathbf{V}_{\text{inno,auto}}$, $\mathbf{V}_{\text{inno,com}}$, and $\mathbf{V}_{\text{inno,time}}$, $\mathbf{V}_{\text{inno,shock}}$, $\mathbf{V}_{\text{inno,uni}}$, and $\mathbf{V}_{\text{inno,weib}}$.

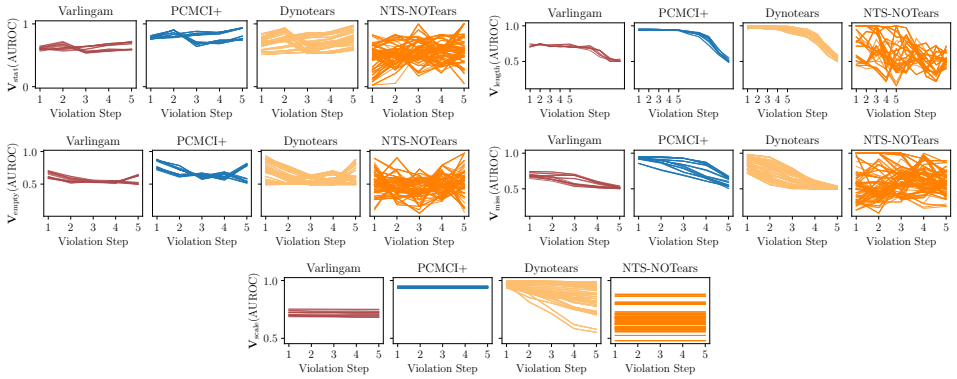


Figure 25: Hyperparameter sensitivities for all methods estimating instantaneous effects. The order from left to right, top to bottom is $\mathbf{V}_{\text{inno,var}}$, $\mathbf{V}_{\text{stat}}$, $\mathbf{V}_{\text{length}}$, $\mathbf{V}_{\text{q,missing}}$, $\mathbf{V}_{\text{q,empty}}$, and $\mathbf{V}_{\text{scale}}$.

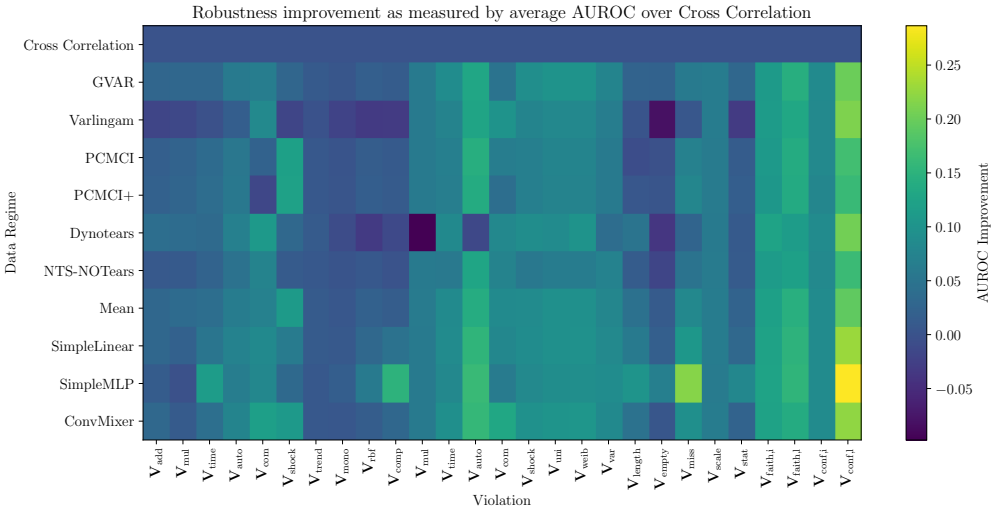


Figure 26: Robustness improvements in comparison to the performance of Cross Correlation per violation. Interestingly, the highest improvements by the best ensembling strategy are achieved on $\mathbf{V}_{\text{q,missing}}$ and $\mathbf{V}_{\text{conf,lag}}$.

D.6 Robustness Gains of Ensembles

To gain deeper insights into the improvements of our ensemble strategies, we plot the gains in robustness in comparison to the Cross Correlation baseline for each violation in Fig. 26 and split by smaller and bigger data regimes in Fig. 27. We find that the largest improvements can be found for the violations $\mathbf{V}_{\text{q,missing}}$ and $\mathbf{V}_{\text{conf,lag}}$ as well as for the bigger SCMs ($D = 7, L = 4$).

E Appendix — LLM usage

During the preparation of this manuscript, we used Gemini 2.5 Pro to refine sentence structure and correct grammatical errors. We reviewed and edited all AI-generated suggestions and take full responsibility for the final content of the publication.

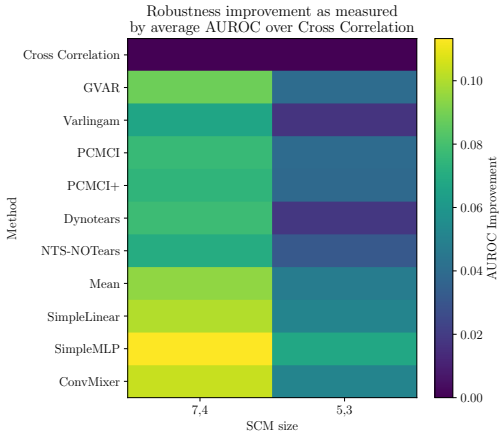


Figure 27: Robustness improvements in comparison to the performance of Cross Correlation for the smaller data regimes ($D = 5, L = 3$) and the bigger data regimes ($D = 7, L = 4$). We find that the highest performance gains can be achieved on the bigger data regimes, independent of the CD method or ensemble strategy. However, the best ensemble strategy (SimpleMLP) is achieving the highest improvements.