

Simple Sorting Criteria Help Find the Causal Order in Additive Noise Models

Alexander G. Reisach, Myriam Tami, Christof Seiler, Antoine Chambaz, Sebastian Weichwald

▶ **To cite this version:**

Alexander G. Reisach, Myriam Tami, Christof Seiler, Antoine Chambaz, Sebastian Weichwald. Simple Sorting Criteria Help Find the Causal Order in Additive Noise Models. 2023. hal-04056341v1

HAL Id: hal-04056341

https://hal.science/hal-04056341v1

Preprint submitted on 3 Apr 2023 (v1), last revised 1 Nov 2023 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

Simple Sorting Criteria Help Find the Causal Order in Additive Noise Models

Alexander G. Reisach[1], **Myriam Tami**[2], **Christof Seiler**[3,4], **Antoine Chambaz**[1], **Sebastian Weichwald**[5,6]

[1]Université Paris Cité, CNRS, MAP5, F-75006 Paris, France
[2]CentraleSupélec, Université Paris-Saclay, France
[3]Department of Advanced Computing Sciences, Maastricht University, The Netherlands
[4]Mathematics Centre Maastricht, Maastricht University, The Netherlands
[5]Department of Mathematical Sciences, University of Copenhagen, Denmark
[6]Pioneer Centre for AI, Copenhagen, Denmark

Abstract

Additive Noise Models (ANM) encode a popular functional assumption that enables learning causal structure from observational data. Due to a lack of real-world data meeting the assumptions, synthetic ANM data are often used to evaluate causal discovery algorithms. Reisach et al. (2021) show that, for common simulation parameters, a variable ordering by increasing variance is closely aligned with a causal order and introduce 'var-sortability' to quantify the alignment. Here, we show that not only variance, but also the fraction of a variable's variance explained by all others, as captured by the coefficient of determination R^2, tends to increase along the causal order. Simple baseline algorithms can use 'R^2-sortability' to match the performance of established methods. Since R^2-sortability is invariant under data rescaling, these algorithms perform equally well on standardized or rescaled data, addressing a key limitation of algorithms exploiting var-sortability. We characterize and empirically assess R^2-sortability for different simulation parameters. We show that all simulation parameters can affect R^2-sortability and must be chosen deliberately to control the difficulty of the causal discovery task and the real-world plausibility of the simulated data. We provide an implementation of the sortability measures and sortability-based algorithms in our library CausalDisco.

1 Introduction

Causal Discovery. Understanding the causal relationships between variables is a common goal across the sciences (Imbens et al. 2015) and may facilitate reliable prediction, the identification of goal-directed action, and counterfactual reasoning (Spirtes, Glymour, et al. 2000; Pearl 2009; Peters, Janzing, et al. 2017). Causal reasoning using Structural Causal Models (SCMs) and graphical models consists of two steps. First, we use expert knowledge or causal discovery algorithms to devise a graph that encodes the causal structure between variables. Second, we learn the functions that relate causes to effects in the SCM and define and estimate our causal quantities of interest. Discovering

causal structure requires interventional data or assumptions on the functions and distributions to restrict the class of possible SCMs that describe a given data-generating process (see, for example, Glymour et al. 2019). Since collecting interventional data may be costly, infeasible, or unethical, we are interested in suitable assumptions to learn causal structure from observational data.

Structure Learning Algorithms for Additive Noise Models. Additive Noise Models (ANMs) encode a popular functional assumption that provides identifiability of the causal structure under various assumptions on the noise distributions. We refer, for instance, to Hoyer et al. (2008); Mooij, Janzing, et al. (2009); Peters, Mooij, et al. (2011); Bühlmann et al. (2014); Park (2020) for identifiability results, and to Heinze-Deml et al. (2018); Kitson et al. (2023) for an overview of causal discovery algorithms. Well-known structure learning algorithms include the constraint-based *PC* algorithm (Spirtes and Glymour 1991) and score-based fast greedy equivalence search (*FGES*) (Meek 1997; Chickering 2002). Using a characterization of DAGs as level set of a differentiable function paved the way for causal discovery using gradient-based optimization (Zheng et al. 2018). This approach promises state-of-the-art results on simulated ANM data and has inspired numerous variants (Vowels et al. 2022).

A lack of real-world data. Due to a lack of real-world datasets for which the underlying ANMs are known, it is hard to demonstrate that a causal discovery algorithm works well in practice or that its output can be trusted when exploring new cause-effect relationships in applications. As a result, the use of simulated data is common for the evaluation of causal discovery algorithms. For such benchmark results to be indicative of how causal discovery algorithms may fair in practice, simulated data need to plausibly mimic observations of real-world data generating processes. The challenge of benchmarking with simulated data is also relevant to causality beyond causal discovery, for example for the task of treatment effect estimation (Curth et al. 2021).

Patterns in simulated data. Reisach et al. (2021) demonstrate that data in ANM benchmarks exhibit high var-sortability, a measure quantifying the alignment between the causal order and an ordering of the variables by their marginal variances (see Section 3 for the definition). High var-sortability implies a tendency for variances to increase along the causal order, which results in an easy structure learning task: take the ordering by increasing marginal variance for a causal order to obtain state-of-the-art results that are on par with, or better than, those of continuous structure learning algorithms, the *PC* algorithm, or *FGES*. This also explains the finding by Weichwald et al. (2020) who observe that the magnitude of regression coefficients may contain more information about causal links than their p-values. However, an obvious and strong increase in variances along the causal order may be unrealistic in the real world. Following the findings of Reisach et al. (2021), Kaiser et al. (2022), and Seng et al. (2022), data for benchmarks is often standardized to control the increase of variances (e.g. Rios et al. 2021; Rolland et al. 2022; Mogensen et al. 2022; Lorch et al. 2022; Xu et al. 2022; Li et al. 2023), which deteriorates the performance of algorithms that rely on information in the data scale.

Contribution. We show that on data sampled in common ANM simulation schemes, not only variance but also the fraction of variance explained by the other variables tends to increase along the causal order. We introduce R^2-sortability to quantify the alignment between a causal order and the order of increasing coefficients of determination R^2. We provide two simple algorithms exploiting R^2-sortability that perform on par with established methods on raw as well as standardized data. In R^2 we introduce a sorting criterion for causal discovery that is applicable even when the data scale is unknown, as is often the case in practice. In contrast to var-sortability, one cannot simply rescale the obtained variables to ensure a desired R^2-sortability of the simulated ANMs. Motivated by these results, we conduct an analysis of the drivers of R^2-sortability. We find that all simulation parameters can influence R^2-sortability. The ANM literature tends to focus on assumptions on the

additive noise distributions (see, for example, Shimizu et al. 2006; Peters and Bühlmann 2014; Park 2020), leading to arguably arbitrary choices for the other synthetic data generation parameters not previously considered central to the model class. Our analysis contributes to closing the gap between theoretical results on the identifiability of ANM structure and making deliberate decisions on all ANM simulation parameters to ensure that structure learning results are meaningful and relevant in practice.

2 Additive Noise Models

We follow standard definitions in defining the model class of linear ANMs. Additionally, we detail the sampling process to generate synthetic data.

Let $X = [X_1, ..., X_d]^\top$ be a vector of d random variables. The causal structure between the components of X can be represented by a causal graph over nodes $\{X_1, ..., X_d\}$ and a set of directed edges between nodes. We encode the set of directed edges by a binary adjacency matrix $B \in \{0, 1\}^{d \times d}$ where $B_{s,t} = 1$ if $X_s \to X_t$, that is, if X_s is a direct cause of X_t, and $B_{s,t} = 0$ otherwise. Throughout, we assume the graphs are directed acyclic graphs (DAG). Based on the causal DAG, we define an ANM. Let $\sigma = [\sigma_1, \ldots, \sigma_d]$ be a vector of positive iid random variables. Let $\mathcal{P}_N(\phi)$ be a distribution with parameter ϕ controlling the standard deviation. Given a draw $\boldsymbol{\sigma}$ of noise standard deviations, let $N = [N_1, ..., N_d]^\top$ be the vector of independent noise variables with $N_t \sim \mathcal{P}_N(\boldsymbol{\sigma}_t)$. For each t, let f_t be a measurable function such that $X_t = f_t(\mathbf{Pa}(X_t)) + N_t$ where $\mathbf{Pa}(X_t)$ is the set of parents of X_t in the causal graph. We assume that $f_1, ..., f_d$ and $\mathcal{P}_N(\phi)$ are chosen such that all $X_1, ..., X_d$ have finite second moments and zero mean.[1] Here, we consider linear ANMs and assume that $f_1, \ldots, f_d$ are linear functions. For linear ANMs, we define a weight matrix $W \in \mathbb{R}^{d \times d}$ where $W_{s,t}$ holds the weight of the causal link from X_s to X_t and $W_{s,t} = 0$ if X_s is not a parent of X_t. The structural causal model can be written as

$$X = W^\top X + N. \tag{1}$$

2.1 Synthetic Data Generation

ANMs are commonly used for generating synthetic data and benchmarking causal discovery algorithms (see, for example, Scutari 2010; Ramsey et al. 2018; Kalainathan et al. 2020). We examine patterns that arise in ANM simulation schemes (var-sortability and R^2-sortability, see Section 3.1) and evaluate algorithms designed to exploit these patterns (*SortnRegress* and R^2-*SortnRegress*, see Section 3.2). Here, we describe the steps to sample ANMs for synthetic data generation which requires the following parameters:

d	Number of nodes
$P_{\mathcal{G}}$	Graph structure distribution
γ	Graph density parameter
$\mathcal{P}_N(\phi)$	Parameterized noise distributions
P_σ	Distribution of noise standard deviations
P_W	Distribution of edge weights

1. Generate the graph structure.[2] We generate the causal graph by sampling a DAG adjacency matrix $B \in \{0, 1\}^{d \times d}$ for a given number of nodes d. In any DAG, the variables can be permuted such that its adjacency matrix is upper triangular. We use this to obtain a random DAG adjacency matrix from undirected random graph models by deleting all but the upper triangle of the adjacency matrix

[1] Since we can always subtract empirical means, the assumption of vanishing means does not come with any loss of generality. In our implementations, we subtract empirical means or use regression models with intercept.

[2] We separate sampling the adjacency matrix from sampling the parameters of the functions f_j, although for linear ANMs a weight matrix can be used to encode both the causal structure and the function parameters.

and shuffling the variable order (Zheng et al. 2018). In our simulations, we use the Erdős–Rényi (ER) (Erdős et al. 1960) and Scale-free (SF) (Barabási et al. 1999) random graph models for $P_{\mathcal{G}}$ with a density parameter γ controlling the average number of edges per node. We denote the distribution of Erdős–Rényi random graphs with d nodes and γd edges as $\mathcal{G}_{\text{ER}}(d, \gamma d)$, and the distribution of Scale-free graphs with the same parameters as $\mathcal{G}_{\text{SF}}(d, \gamma d)$.

2. Define noise distributions. In linear ANMs, each variable X_t is a linear function of its parents plus an additive noise variable N_t. We choose a distributional family for $\mathcal{P}_N(\phi)$ (for example, zero-mean Gaussian or Uniform) and independently sample standard deviations $\boldsymbol{\sigma}_1, ..., \boldsymbol{\sigma}_d$ from P_σ. We then set N_t to have distribution $\mathcal{P}_N(\boldsymbol{\sigma}_t)$ with standard deviation $\boldsymbol{\sigma}_t$.

3. Draw weight parameters. We sample $\alpha_{s,t}$ for $s, t = 1, ..., d$ independently from P_W and define the weight matrix via $W = B \odot \left[\alpha_{s,t}\right]_{s,t=1,...,d}$.

4. Sample from the ANM. To sample observations from the ANM with given graph (step 1), edge weights (step 2), and noise distributions (step 3), we use that $\text{Id} - W^\top$ is invertible if W is the adjacency matrix of a DAG to re-arrange Equation (1) and obtain the data generating equation:

$$X = \left(\text{Id} - W^\top\right)^{-1} N. \tag{2}$$

$\mathbf{X} = \left[\mathbf{X}^{(1)}, ..., \mathbf{X}^{(n)}\right]^\top \in \mathbb{R}^{n \times d}$ denotes a dataset of n observations of X. The i-th observation of variable t in $\mathbf{X}$ is $\mathbf{X}_t^{(i)}$, and $\mathbf{X}^{(i)} \in \mathbb{R}^d$ is the i-th observation vector.

3 Defining Sortabilities and Exploiting R^2-sortability

Var-sortability measures a pattern in the variances of variables in a causal graph and takes high values in common ANM simulation schemes (Reisach et al. 2021). We introduce a related pattern in the fractions of explained variance along the causal order that is measured by 'R^2-sortability'. Similarly to var-sortability, R^2-sortability also takes high values in common ANM simulation schemes and can be exploited to learn the causal structure of ANMs via simple algorithms. In contrast to var-sortability, R^2-sortability is invariant under rescaling of the variables, so the presented algorithms recover the causal structure equally well from raw, rescaled, and standardized data.

3.1 From Var-sortability to R^2-sortability

Var-sortability measures the agreement between an ordering by variance and a causal ordering. In common ANM simulation schemes, the marginal variances of variables tend to increase along the causal order leading to high var-sortability. This accumulation of noise along the causal order motivates our investigation of a related pattern. If the marginal variances of variables X_t given as $\text{Var}\left(W_{:,t}{}^\top \mathbf{Pa}(X_t)\right) + \text{Var}\left(N_t\right)$ increase along the causal order and the noise standard deviations are sampled iid, then $\text{Var}\left(W_{:,t}{}^\top \mathbf{Pa}(X_t)\right)$ and in turn the cause-explained variance fractions $\frac{\text{Var}\left(W_{:,t}{}^\top \mathbf{Pa}(X_t)\right)}{\left(\text{Var}\left(W_{:,t}{}^\top \mathbf{Pa}(X_t)\right) + \text{Var}(N_t)\right)}$ are likely to also increase along the causal order.[3] Unlike the variance, we cannot calculate (nor obtain an unbiased estimate of) a variable's cause-explained variance without knowing its parents in the graph.

Instead, we propose to use an upper bound on the cause-explained variance fraction and to assess the fraction of explained variance of a variable X_t given all remaining variables $1 - \frac{\text{Var}(X_t - E[X_t | X_{s \neq t}])}{\text{Var}(X_t)}$ where $s \neq t$ is the set $\{1, \ldots, d\} \setminus \{t\}$. This quantity is known as the coefficient of determination R^2 (see, for example, Glantz et al. 2001). In practice, we need to choose a regression model and regress the variable onto all others to obtain an estimate of this quantity. Here, we choose linear models

[3] We use the shorthand notation $W_{:,t} \in \mathbb{R}^{|\mathbf{Pa}(X_t)|}$ for the vector of non-zero entries in the t^{th} column of W.

$M^{\theta}_{t,S}(X_S)\colon \mathbb{R}^{|S|} \to \mathbb{R}, X_S \mapsto \theta^\top X_S$ for the regression of X_t onto X_S with $S \subseteq \{1, ..., d\} \setminus \{t\}$ and $\theta \in \mathbb{R}^{|S|}$. We denote the least-squares fit by $M^{\theta^*}_{t,S}$ such that the estimate of R^2 is obtained as

$$R^2(M^{\theta^*}_{t,S}, X) = 1 - \frac{\mathrm{Var}\left(X_t - M^{\theta^*}_{t,S}(X_S)\right)}{\mathrm{Var}\left(X_t\right)}.$$

To find a common definition for sortability by different ordering criteria, we introduce a family of criteria $\mathbf{v}$ that assign a scalar in $[0, 1]$ to variables X in graph $\mathcal{G}$ with adjacency matrix $B_{\mathcal{G}}$ for different functions τ:

$$\mathbf{v}_\tau(X, \mathcal{G}) = \frac{\sum_{i=1}^{d} \sum_{(s \to t) \in B^i_{\mathcal{G}}} \mathrm{incr}(\tau(X, s), \tau(X, t))}{\sum_{i=1}^{d} \sum_{(s \to t) \in B^i_{\mathcal{G}}} 1} \text{ where } \mathrm{incr}(a, b) = \begin{cases} 1 & a < b \\ 1/2 & a = b \\ 0 & a > b \end{cases} \tag{3}$$

(where $(s \to t) \in B^i_{\mathcal{G}}$ if and only if a directed path from X_s to X_t of length i exists in $\mathcal{G}$). We obtain the original definition of var-sortability[4] for $\tau(X, t) = \mathrm{Var}\left(X_t\right)$ and denote it as $\mathbf{v}_{\mathrm{Var}}$. We obtain the R^2-sortability for $\tau(X, t) = R^2(M^{\theta^*}_{t,s\neq t}, X)$ and denote it as $\mathbf{v}_{R^2}$.

As with var-sortability, if $\mathbf{v}_{R^2}$ is 1, then the causal order is identified by the variable order of increasing R^2. If $\mathbf{v}_{R^2}$ is 0, the causal order is identified by the variable order of decreasing R^2. A value of $\mathbf{v}_{R^2} = 0.5$ means that ordering the variables by R^2 amounts to a random causal ordering. However, $\mathbf{v}_{\mathrm{Var}}$ and $\mathbf{v}_{R^2}$ need not be equal, and an ANM that may be identifiable under one criterion may not be so under the other. For an example, consider the fully connected graph between the nodes $\{X_1, X_2, X_3\}$ in this causal order. If $\boldsymbol{\sigma} = [1, 1, 1]$ and $W_{1,2} = 1, W_{1,3} = 0.1, W_{2,3} = 1$, we have that $\mathbf{v}_{\mathrm{Var}} = 1$ and $\mathbf{v}_{R^2} = 3/4$. If $\boldsymbol{\sigma} = [1, 1, 0.5]$ and $W_{1,2} = 1, W_{1,3} = 0.5, W_{2,3} = 0.5$, we have that $\mathbf{v}_{\mathrm{Var}} = 3/4$ and $\mathbf{v}_{R^2} = 1$. We present a quantitative comparison across simulation settings in Section 4.3. Although they are not identical, we show in the following subsections that high $\mathbf{v}_{R^2}$ can be exploited similarly to high $\mathbf{v}_{\mathrm{Var}}$.

3.2 Exploiting R^2-sortability

SortnRegress is a simple algorithm that obtains state-of-the-art performance in common causal discovery benchmarks, typically ANMs with high varsortability (Reisach et al. 2021). Here, we show how high R^2-sortability can similarly be used in an ordering-based search algorithm (cf. Teyssier et al. 2005) which we term '*R^2-SortnRegress*' . As before, we denote by $M^{\theta}_{t,S}$ a parameterized model of X_t with covariates X_S and parameters θ. For an example, consider the linear model $M^{\theta}_{t,S}(\mathbf{X}_S)\colon \mathbf{X}_S \mapsto \theta^\top \mathbf{X}_S$ with $\theta_S = \mathbb{R}^{|S|}$. To simplify notation, we define the mean squared error (MSE) of a model as $\mathrm{MSE}(\mathbf{X}_t, M^{\theta}_{t,S}) = n^{-1} \sum_{i=1}^{n} \left(\mathbf{X}^{(i)}_t - M^{\theta}_{t,S}(\mathbf{X}^{(i)}_S)\right)^2$.

Algorithm 1 estimates the coefficient of determination for every variable to obtain a candidate causal order. Then, it performs a regression of each node onto its predecessors in that order to obtain the candidate causal graph. To encourage sparsity, we can use a penalty function $\lambda(\theta)$. In our implementation, we use an L1 penalty with the regularization parameter chosen via the Bayesian Information Criterion (Schwarz 1978).

In addition, we develop a greedy DAG search (GDS) algorithm based on ideas presented, for example, by Chickering (1996). For a greedy DAG search algorithm we iteratively insert edges into a DAG that most improve a score until an improvement is no longer possible. The score criterion for the

[4]This definition of var-sortability follows that of Reisach et al. (2021), which treats all paths of the same length between a given pair of nodes as a single path. We compare it to alternative formulations considering all cause-effect paths separately or considering them all as one, and find that it strikes a balance between these possible alternatives (see Appendix C for details).

Algorithm 1: R^2-SortnRegress

Data: $\mathbf{X} \in \mathbb{R}^{n \times d}$
Input: λ /* Penalty function, such as $\lambda(\theta) = \|\theta\|_1$
Result: weight matrix estimate $\widehat{W} \in \mathbb{R}^{d \times d}$
/* Candidate causal order
$\pi = \mathbf{0} \in \mathbb{R}^d$
/* Estimate R^2-s using regression (with intercept; cf. Footnote 1)
for $t = 1, \ldots, d$ **do**
  $\theta^* = \arg\min_\theta \mathrm{MSE}\left(\mathbf{X}_t, M^\theta_{t,s\neq t}\right)$
  $\pi_t = R^2(M^{\theta^*}_{t,s\neq t}, \mathbf{X})$
Find a permutation σ that sorts π ascending
/* Estimate weights using regression (with intercept)
$\widehat{W} = \mathbf{0} \in \mathbb{R}^{d \times d}$
for $i = 2, \ldots, d$ **do**
  $t = \{j : \sigma(j) = i\}$
  $S = \{j : \sigma(j) < i\}$
  $\widehat{W}_{S,t} = \arg\min_\theta \mathrm{MSE}\left(\mathbf{X}_t, M^\theta_{t,S}\right) + \lambda(\theta)$
return $\widehat{W}$

algorithm we term 'R^2-GDS' is a weighted MSE to prioritise the insertion of edges pointing into variables for which more of the variance can be explained. We iteratively insert edges such that the resulting candidate DAG $\widehat{\mathcal{G}}$ minimizes

$$\sum_{t=1}^{d} \frac{\min_\theta \mathrm{MSE}\left(\mathbf{X}_t, M^\theta_{t,\{s:\, X_s \text{ among } \mathbf{Pa}_{\widehat{\mathcal{G}}}(X_t)\}}\right)}{\min_\theta 1 - R^2(M^\theta_{t,s\neq t}, \mathbf{X})}.$$

Once the score can no longer be improved by edge insertion, we perform a sparse regression of each variable X_t onto its parents $\mathbf{Pa}_{\widehat{\mathcal{G}}}(X_t)$ in the candidate graph $\widehat{\mathcal{G}}$ to possibly prune edges pointing to X_t.

3.3 Empirical Results

We compare R^2-*SortnRegress* and R^2-*GDS* to a representative choice of established structure learning algorithms and evaluate their performance on tasks with var-sortabilities ranging from 0.5 to 1.

3.3.1 Comparison of Causal Discovery Algorithms

We evaluate and compare R^2-*SortnRegress* and R^2-*GDS* on observations obtained from ANMs simulated with the following parameters:

$$\begin{aligned}
P_{\mathcal{G}} &= \mathcal{G}_{\mathrm{ER}}(20, 40) && \text{(Erdős–Rényi graphs, } d = 20, \gamma = 2\text{)} \\
\mathcal{P}_N(\phi) &= \mathcal{N}(0, \phi^2) && \text{(Gaussian noise distributions)} \\
P_\sigma &= \mathrm{Unif}(0.5, 2) && \text{(noise standard deviations)} \\
P_W &= \mathrm{Unif}((-2, -0.5) \cup (0.5, 2)) && \text{(edge weights)}
\end{aligned}$$

We sample 30 ANMs (graph, noise standard deviations, and edge weights) and generate 1000 observations of each. We compare the introduced R^2-based algorithms to the *PC* algorithm (Spirtes and Glymour 1991) and *FGES* (Meek 1997; Chickering 2002). Representative for scale-sensitive

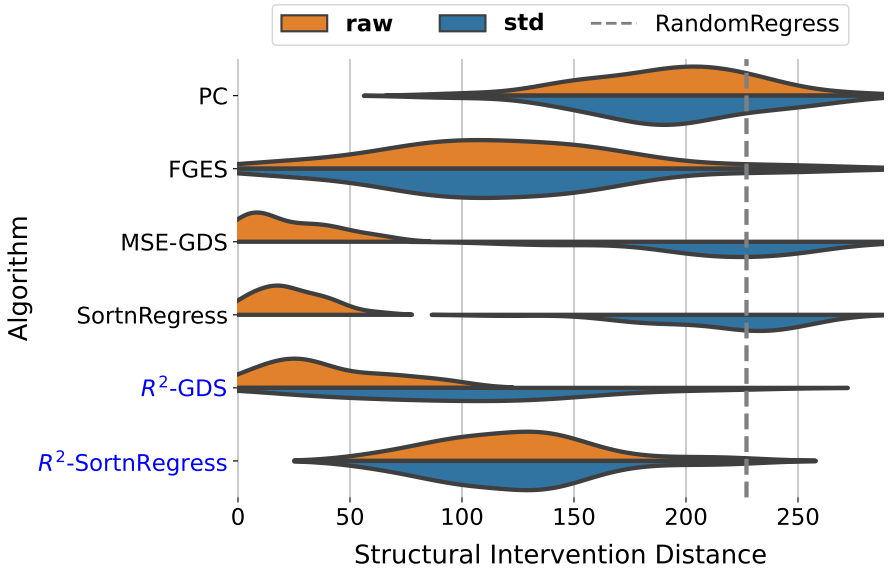


Figure 1: Performance comparison in terms of SID on simulated data. Our algorithms exploiting R^2-sortability (blue) perform well on raw and standardized data with results comparable to those of established methods.

algorithms, we include *MSE-GDS* and *SortnRegress*, introduced by Reisach et al. (2021) to exploit var-sortability, and also provide their *RandomRegress* baseline (we refer to their article for comprehensive comparisons with further algorithms). We describe the algorithms and implementations in Appendix A. We evaluate the structural intervention distance (SID) (Peters and Bühlmann 2015) between the recovered and the ground-truth graphs on raw (orange) and standardized (blue) data. The SID measures the number of interventional distributions between node pairs that would be incorrectly estimated in the recovered graph compared to the true graph (lower values are better). It can be understood as a measure of disagreement between a causal order induced by the recovered graph and a causal order induced by the true graph. The results are shown in Figure 1.

SortnRegress and *MSE-GDS* successfully exploit the high var-sortability on raw data to achieve the best results. Of the other algorithms, only R^2*-GDS* performs on par as its score criterion also includes the variance-sensitive mean squared error. Notably, R^2*-SortnRegress* performs almost as well as *FGES* and outperforms *PC*, indicating an exploitable degree of R^2-sortability in the data.

As expected, the performance of scale-sensitive algorithms is worse on standardized data while *PC*, *FGES*, and R^2*-SortnRegress* show similar performance on raw and standardized data. While R^2*-GDS* is scale-sensitive, its performance is still comparable to *PC* and *FGES* on standardized data. Our results indicate that R^2*-SortnRegress* is a competitive baseline on raw data with high var-sortability and on standardized data with high pre-standardization var-sortability.

We observe similar qualitative results across several simulation settings and evaluation metrics, for both the recovery of the DAG and its Markov Equivalence Class (see Appendix A). R^2*-SortnRegress* is scale-invariant since the fraction of explained variance is unaffected by any (non-zero) multiplicative rescaling of the variables, which is in line with our empirical results across raw and standardized data.

3.3.2 Performance Sensitivity to Pre-Standardization Var-Sortability and R^2-Sortability

We investigate the sensitivity of R^2*-SortnRegress* to pre-standardization var-sortability and R^2-sortability, and compare it to *SortnRegress*. This analysis serves to illuminate the performance that

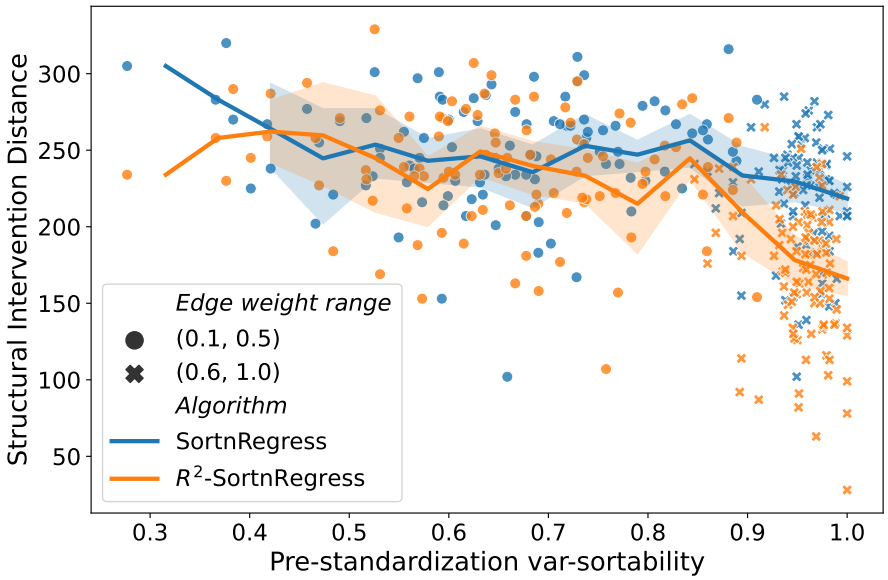


Figure 2: Algorithm performance in SID obtained on standardized data for different pre-standardization var-sortability. The performance of R^2-*SortnRegress* increases with pre-standardization var-sortability.

may be achievable by R^2-based algorithms on data with different (pre-standardization) var-sortability and R^2-sortability values. We use Erdős–Rényi (ER) graphs $P_{\mathcal{G}} = \mathcal{G}_{\text{ER}}(10, 20)$ ($d = 10, \gamma = 2$) and Gaussian noise $\mathcal{P}_N(\phi) = \mathcal{N}(0, \phi^2)$ with standard deviations drawn from $P_\sigma = \text{Unif}(0.5, 2)$. We sample 100 ANMs with edge weight distribution $P_W = \text{Unif}((-0.5, -0.1) \cup (0.1, 0.5))$ and 100 ANMs with edge weight distribution $P_W = \text{Unif}((-1, -0.6) \cup (0.6, 1))$ to ensure that we obtain a wide range of var-sortabilities before standardization and a wide range of R^2-sortabilities. We generate 1000 observations per ANM, standardize all data, and again use SID to assess structure recovery performance. The results for different pre-standardization var-sortabilities are shown in Figure 2.

As expected *SortnRegress*, which performs sparse regressions for parent selection after ordering the variables by increasing variance, is unaffected by pre-standardization var-sortability since an ordering by variance amounts to a randomized ordering on standardized data. By contrast, R^2-*SortnRegress* achieves lower SID on average for ANM data with pre-standardization var-sortability $\gtrsim 0.9$. For context, Reisach et al. (2021) report that (pre-standardization) var-sortability averages above 0.94 for DAGs and edge weights sampled as in common benchmark settings. In Appendix B we present analogous results for R^2-sortability and observe a virtually identical trend to that in Figure 2 with lower SID for higher values of R^2-sortability ($\gtrsim 0.9$).

4 Drivers of Sortability Patterns in ANM Simulations

In this section, we characterize parameter choices for the synthetic data generation (see Section 2.1) that drive high var-sortability of the simulated ANMs. We provide a lower bound for var-sortability in causal chains that depends on the chosen weight distribution P_W and explain the connection to R^2-sortability. We empirically assess R^2-sortability for different ANM simulation parameters.

4.1 A Lower Bound On Var-Sortability in Causal Chains

When generating synthetic ANM data for a given graph $\mathcal{G}$ and noise distributions $\mathcal{P}_N(\phi)$, var-sortability is determined by P_W and P_σ from which we independently sample the edge weights

$W_{s,t} \sim P_W$ for all s, t with $B_{s,t} = 1$ and the noise standard deviations $\sigma_t \sim P_\sigma$ as described in Section 2.1. We can reason about the expected var-sortability of a given graph where $\mathrm{Var}(X_t)$ is the conditional variance of X_t given everything but the noise vector N. For example, given the 2-chain graph denoted $\mathcal{G}_{2\text{-chain}}: X_1 \to X_2$ the probability that $\mathrm{Var}(X_1) < \mathrm{Var}(X_2)$ is lower bounded by $P(1 \leq |W_{1,2}|)$, hence the probability for a var-sortability of 1 is bounded as

$$P_{W,\sigma}(\mathbf{v}_{\mathrm{Var}}(X, \mathcal{G}_{2\text{-chain}}) = 1) \;\geq\; P\left(1 \leq |W_{1,2}|\right) \tag{4}$$

(see Appendix D and Reisach et al. (2021), Section 3.3, who introduce this lower bound).

We generalize Equation (4) for causal chains to characterize the emergence of var-sortability, which also provides intuition about var-sortability in general graphs. The idea is to lower bound the probability of a causal ancestor having lower variance than its causal descendant in terms of the edge weight products along all paths between them. The bound further simplifies and depends only on the weight distribution P_W if only one directed path between two connected nodes exists, as is the case in chain graphs.

We derive the following two generalizations of Equation (4) and provide details in Appendix D.

Pair-wise lower bound in chains. For a causal chain of length p between a source node X_0 and a target node X_p such that X_0 and X_p have no common ancestors and with a unique directed path

$$X_0 \xrightarrow{W_{0,1}} X_1 \xrightarrow{W_{1,2}} X_2 \longrightarrow \cdots \xrightarrow{W_{p-1,p}} X_p$$

the probability that the variance of X_0 is smaller than that of X_p when sampling edge weights iid from P_W is lower bounded as

$$P_{W,\sigma}\left(\mathrm{Var}(X_0) < \mathrm{Var}\left(X_p\right)\right) \geq P\left(0 < \sum_{s=0}^{p-1} \ln |W_{s,s+1}|\right). \tag{5}$$

The sum is over iid random variables $\ln |W_{s,t}|$ with $W_{s,t} \sim P_W$. By the law of large numbers, for long paths the lower bound in Equation (5) is driven by the geometric weight mean $E(\ln |V|)$ for $V \sim P_W$, which we write as $E(\ln |P_W|)$.

Pair-wise lower bound in chains with $E(\ln |P_W|) > 0$. If the edge weight distribution P_W is chosen such that $E(\ln |P_W|) > 0$, we can further lower bound Equation (5) using Cantelli's inequality (Glantz et al. 2001). The probability that $\mathrm{Var}(X_0) < \mathrm{Var}\left(X_p\right)$ when sampling edge weights iid from P_W with $E(\ln |V|) > 0$ is lower bounded as

$$P_{W,\sigma}\left(\mathrm{Var}(X_0) < \mathrm{Var}\left(X_p\right)\right) \geq 1 - \frac{\mathrm{Var}(\ln |V|)}{\mathrm{Var}(\ln |V|) + pE(\ln |V|)^2}. \tag{6}$$

The lower bound on the probability of X_0 and X_p being correctly sorted by variance when sampling the edge weights from P_W increases to 1 as the length p of the chain from X_0 to X_p goes to infinity. The bound highlights the importance of the weight distribution P_W for var-sortability in simulated ANMs.

Comparison of bound and empirical var-sortability in chains. We compare the empirical var-sortability of chain graphs to the lower bound on the expected var-sortability obtained via Equation (6).[5] We simulate chain graphs with $P_N(\sigma) = \mathcal{N}(0, \sigma^2)$ and $P_\sigma = \mathrm{Unif}(0.5, 2)$. Edge weights

[5] For a causal chain $\mathcal{G}$ and if $E(\ln |P_W|) > 0$, we use the linearity of expectation to lower bound the expectation of $\sum_{i=1}^{d} \sum_{(s \to t) \in B^i_{\mathcal{G}}} \mathrm{incr}(\mathrm{Var}(X_s), \mathrm{Var}(X_t))$ (the numerator in the definition of var-sortability, cf. Equation (3)) via bounding the expectation of each summand using Equation (6). For a chain of length l and $h(p) = 1 - \frac{\mathrm{Var}(\ln |P_W|)}{\mathrm{Var}(\ln |P_W|) + pE(\ln |P_W|)^2}$, we obtain the bound as $\sum_{p=1}^{l} (l + 1 - p) h(p)$.

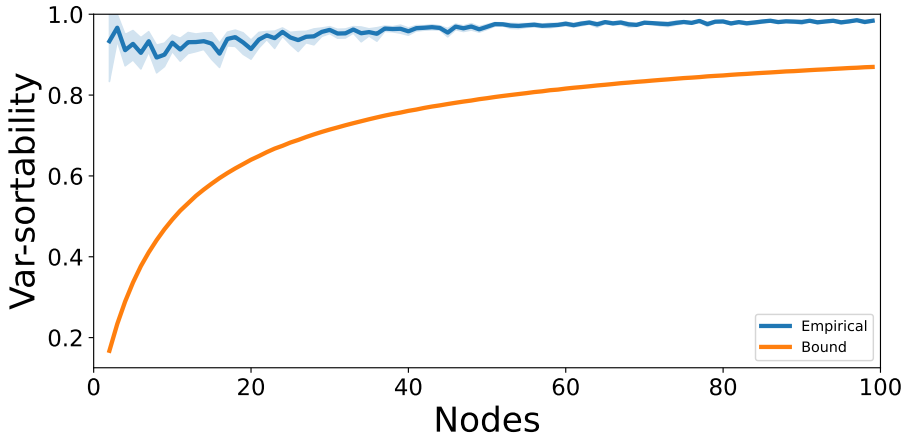


Figure 3: Comparison of empirical var-sortability and the lower bound based on Equation (6) (cf. Footnote 5) for chains with $P_N(\sigma) = \mathcal{N}(0, \sigma^2)$, $P_\sigma = \text{Unif}(0.5, 2)$, and edge weights sampled from $\text{Unif}((-2, -0.5) \cup (0.5, 2))$.

are sampled from $P_W = \text{Unif}((-2, -0.5) \cup (0.5, 2))$. For each number of nodes $d = 2, ..., 100$ we independently draw 30 chains with different edge weights connecting them and generate 10^3 iid observations for each of the chains. To calculate the bound, we estimate $E(\ln |P_W|)$ and $\text{Var}(\ln |P_W|)$ on 10^5 iid samples from P_W, which yields ≈ 0.152 and ≈ 0.146 respectively in our setting.

As shown in Figure 3, we observe var-sortability > 0.8 in all simulated causal chains and > 0.9 in all simulated causal chains of length > 25. The bound is loose but for an increasing number of nodes approaches the empirically observed var-sortability from below.

4.2 R^2-Sortability in Random DAGs

We do not have a lower bound on the expected var-sortability when sampling general DAGs instead of causal chains. Therefore, we analyze var-sortability of synthetically generated ANMs for different simulation parameters (cf. Section 2.1).

4.2.1 R^2-Sortability and Graph Size

We analyze R^2-sortability for different graph sizes with d nodes. We use $P_\mathcal{G} = \mathcal{G}_{\text{ER}}(d, 2d)$ when $d > 4$ and fully connected graphs when $d \leq 4$, that is, when $2d$ exceeds the maximum number of possible edges. We test a range of different edge weight distributions $P_W = \text{Unif}((-\beta, -0.1) \cup (0.1, \beta))$ with $\beta > 0.1$ chosen such that the $E(\ln |P_W|)$ are approximately evenly spaced in $(-1, 2.5)$. We otherwise use the same settings as in Section 3.3.1. For each weight distribution and node number we independently simulate 30 ANMs (graph, noise standard deviations, and edge weights) to obtain error bars. The results are shown in Figure 4. We observe that R^2-sortability is higher for edge weight distributions P_W with higher $E(\ln |P_W|)$. R^2-sortability fluctuates most for small graphs and appears to stabilize for increasing graph size.

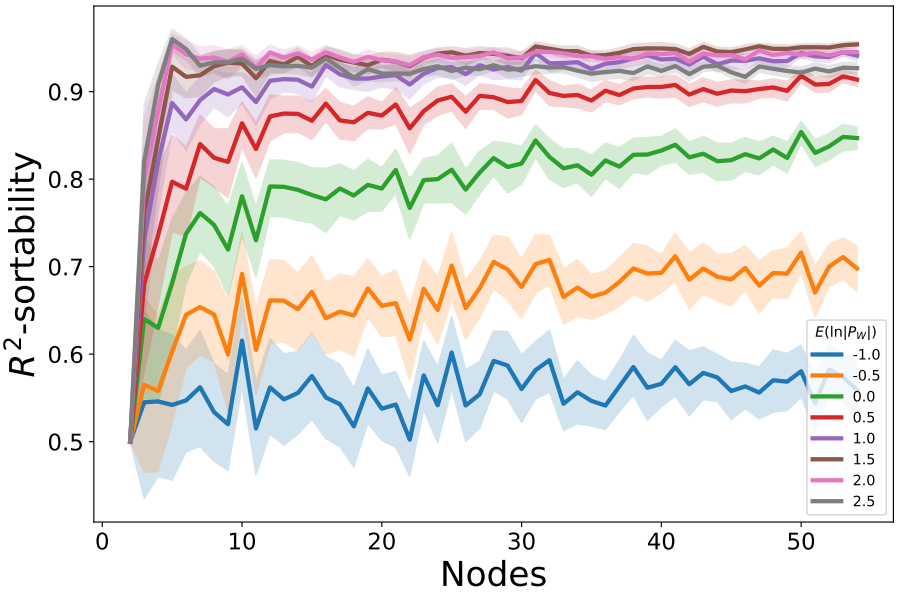


Figure 4: R^2-sortability at different graph sizes for a range of weight distributions with different geometric weight mean $E(\ln|P_W|)$.

4.2.2 R^2-Sortability and ANM Simulation Parameters

We empirically assess which simulation parameters drive R^2-sortability and how we may control R^2-sortability in synthetic data generation schemes. To this end, we sample ANMs under different values of the simulation parameters described in Section 2.1. We show illustrative examples here and comprehensive simulation results in Appendix E.

We use a fixed graph size $d = 50$ since we expect trends in R^2-sortability for changing graph sizes to have levelled off (cf. Figure 4). We use the Erdős–Rényi (Erdős et al. 1960) and Scale Free (SF) (Barabási et al. 1999) random graph models for $P_{\mathcal{G}}$. We vary the graph density parameter γ, not normally a focus of theoretical ANM analyses, for each simulation setting as it affects the average number of incoming edges for a node and thereby may affect its fraction of cause-explained variance. We choose a wide range of values for γ, from the subcritical to the connected regime (Erdős et al. 1960; Erdős et al. 1961). We use zero-mean Gaussian and Uniform noise distributions $\mathcal{P}_N$ and sample the noise standard deviations either from Uniform or Exponential distributions. We obtain edge weight distributions with different $E(\ln|P_W|)$ as described in Section 4.2.1. We average results for 10 independent runs of each setting.

The results for a simulation setting as in Section 3.3.1 are shown in Figure 5. We find that R^2-sortability $\mathbf{v}_{R^2}$ is generally high and sensitive to $E(\ln|P_W|)$ and γ. We observe the highest R^2-sortabilities when neither $E(\ln|P_W|)$ nor γ are very low or very high, and the lowest ones when both are comparatively low. The absolute values of the edge weights for $E(\ln|P_W|) = -1$ are in the range $[0.1, 0.73]$ while the ones for $E(\ln|P_W|) = 1$ are in the range $[0.1, 32.53]$. Weights close to zero complicate the statistical problem of detecting dependences between variables while weights with large magnitude may lead to very large variances on the raw data scale. We conclude that R^2-sortability is sensitive to the choice of edge weight distribution and graph density, and we observe high R^2-sortabilities for what can be considered moderate parameter choices.

We observe qualitatively similar results across all simulations shown in Appendix E with slightly lower values of R^2-sortability for noise standard deviations sampled from an Exponential distribution. For Scale-free graphs, which are inspired by the topology of real-world networks, the base level of R^2-sortability is extremely high (Barabási et al. 1999). Our empirical results highlight that

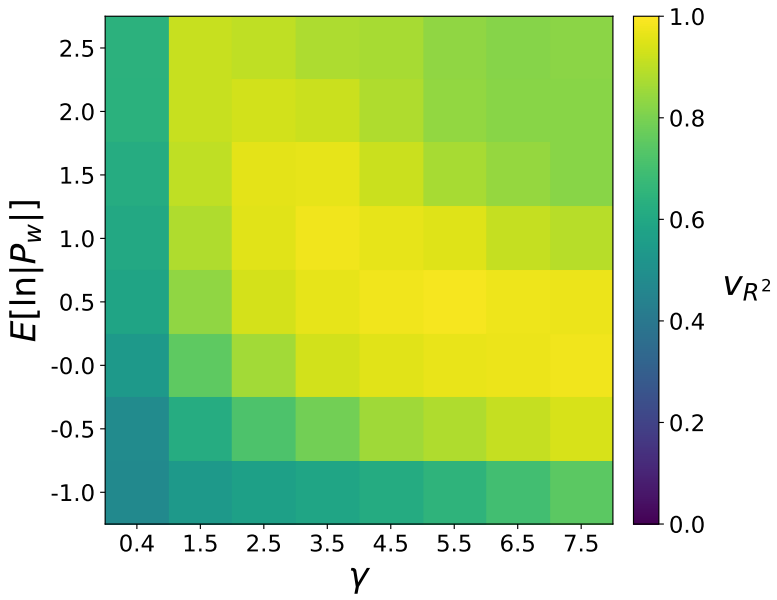


Figure 5: R^2-sortability of $\mathcal{G}_{ER}(50, \gamma 50)$ graphs with $\mathcal{P}_N = \mathcal{N}(0, \sigma^2)$ and $P_\sigma = \text{Unif}(0.5, 2)$ for different values of graph density γ and geometric weight mean $E(\ln|P_W|)$.

ANM simulation parameters that may not appear central when investigating ANMs often have a dominant impact on the difficulty of the causal discovery task, and that our sortability criteria may be exploitable for a wide range of ANM simulation parameters.

4.3 The Relationship between R^2-sortability and Var-sortability

We analyze the relative magnitude of var-sortability $\mathbf{v}_{\text{Var}}$ and R^2-sortability $\mathbf{v}_{R^2}$ and compute their ratio for the simulation settings from Section 4.2.2.

Figure 6 shows the average value of the ratio $\mathbf{v}_{R^2}/\mathbf{v}_{\text{Var}}$ of 10 independent runs for each parameter combination. We observe that $\mathbf{v}_{R^2}$ tracks $\mathbf{v}_{\text{Var}}$ well in most settings. Their ratio $\mathbf{v}_{R^2}/\mathbf{v}_{\text{Var}}$ is smaller than 1, meaning that on raw data R^2-sortability takes lower values than var-sortability. Nonetheless, R^2-sortability is important because it remains unchanged under standardization or other data rescaling. The ratio is closest to 1 in the settings for which we observe the highest R^2-sortability in Figure 5, that is, around the main diagonal, and is lowest for sparse graphs (small γ). We refer to Appendix F for an analogous experiment on Scale-free graphs and an analysis of the relationship of both measures to the cause-explained variance fraction.

5 Discussion

We introduce R^2 as a simple sorting criterion that helps find the causal order in linear ANMs and that is scale-invariant. We show that algorithms exploiting R^2-sortability achieve competitive performance on data simulated from ANMs with common parameters. This raises the question if, and to what extent, other causal discovery algorithms may also be sensitive to R^2-patterns, which requires further research and the comparison of a wider class of algorithms on data with different R^2-sortabilities. To provide context on the scale-invariant difficulty of a causal discovery task, we recommend the comparison to the baseline algorithms R^2-*SortnRegress* or R^2-*GDS* when evaluating structure learning algorithms on simulated ANM data. We emphasize that the algorithms exploiting var- and R^2-sortability exploit a pattern in ANM data that is, to a large extent, driven by parameters not previously thought to be central to the model class. Given the success of sortability-based algorithms for causal discovery on simulated data, a promising topic for future research is to

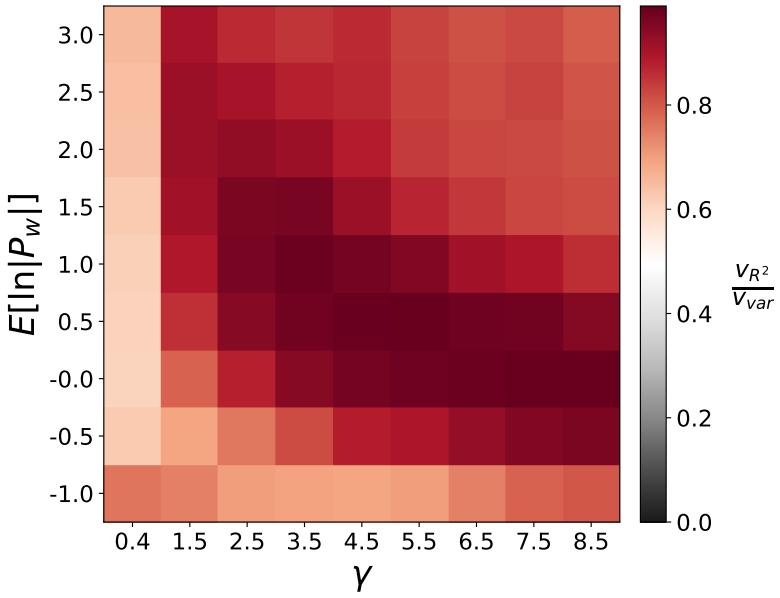


Figure 6: Relationship between $\mathbf{v}_{R^2}$ and $\mathbf{v}_{\text{Var}}$ as ratio $\mathbf{v}_{R^2}/\mathbf{v}_{\text{Var}}$ in $\mathcal{G}_{ER}(50, \gamma 50)$ graphs with $\mathcal{P}_N = \mathcal{N}(0, \sigma^2)$ and $P_\sigma = \text{Unif}(0.5, 2)$ for different graph densities γ and geometric weight means $E(\ln|P_W|)$.

evaluate these algorithms in applications, and to combine them with other approaches. Additionally, since var-sortability is high in many nonlinear ANMs (Reisach et al. 2021), R^2-sortability may also be high and thus useful for finding the causal structure in nonlinear ANMs.

For many ANM simulation parameters, R^2-sortability closely tracks var-sortability, and both take high values. Nonetheless, R^2- and var-sortability are not equally close for all parameters, and their trends for more extreme choices are yet to be explored. The big gap in the level of R^2-sortability between Erdős–Rényi and Scale-Free graphs hints at the possibility of designing graph models with specified target R^2-sortabilities. An alternative way to steer R^2-sortability and attenuate the accumulation of cause-explained variance along the causal order could be a coupling of edge weights or noise parameters, instead of drawing them independently. Our findings provide a basis for choosing parameters in existing simulation schemes to control var- and R^2-sortability. Whenever possible, we advocate choosing the parameters carefully with respect to the given application context. An open and challenging problem is to determine what R^2-sortabilities can be expected in real-world data. On simulated data, we therefore recommend reporting R^2-sortability, or pre-standardization var-sortability as a simple and computationally inexpensive proxy of R^2-sortability.

Conclusion In R^2-sortability, we show the existence of another pattern closely linked to var-sortability that can also be exploited by simple algorithms. Due to the scale-invariance of R^2, algorithms exploiting the pattern do not rely on knowing the data scale and are therefore more widely applicable than scale-sensitive algorithms. The two presented algorithms achieve competitive results and provide a new baseline for the causal discovery performance achievable by exploiting simple sorting criteria. We recommend their use in future simulations to indicate the difficulty of the causal discovery task and the relative performance gains of other algorithms. This is important since R^2-sortability is high (that is, much closer to 1 than to 0.5) across a wide range of simulation settings. All ANM simulation parameters need to be chosen deliberately and, ideally, to match the expected prevalence of the R^2 pattern in real-world data generating processes. In Erdős–Rényi graphs, high R^2-sortability arises for common parameter choices. Low edge weights, low graph density, and an unbounded noise standard deviation distribution can lead to lower R^2-sortability. In Scale-free graphs, R^2-sortabilities are consistently high for a wide range of parameter settings.

Acknowledgements

We thank Brice Hannebicque for helpful discussions. AGR received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 945332 .

References

[1] Albert-László Barabási and Réka Albert. "Emergence of Scaling in Random Networks." In: *Science* 286.5439 (1999), pp. 509–512 (cit. on pp. 4, 11).

[2] Peter Bühlmann, Jonas Peters, and Jan Ernest. ""CAM: Causal Additive Models, High-Dimensional Order Search and Penalized Regression." In: *The Annals of Statistics* 42.6 (2014), pp. 2526–2556 (cit. on p. 2).

[3] David Maxwell Chickering. "Learning Bayesian Networks is NP-Complete." In: *Learning from data: Artificial intelligence and statistics V* (1996), pp. 121–130 (cit. on p. 5).

[4] David Maxwell Chickering. "Optimal Structure Identification With Greedy Search." In: *Journal of Machine Learning Research* 3 (2002), pp. 507–554 (cit. on pp. 2, 6).

[5] Gabor Csardi, Tamas Nepusz, et al. "The igraph software package for complex network research." In: *InterJournal Complex Systems* 1695.5 (2006), pp. 1–9 (cit. on p. 18).

[6] Alicia Curth et al. "Really Doing Great at Estimating CATE? A Critical Look at ML Benchmarking Practices in Treatment Effect Estimation." In: *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*. Vol. 1. 2021 (cit. on p. 2).

[7] Paul Erdős, Alfréd Rényi, et al. "On the Evolution of Random Graphs." In: *Publ. Math. Inst. Hung. Acad. Sci* 5.1 (1960), pp. 17–60 (cit. on pp. 4, 11).

[8] Paul Erdős and Alfréd Rényi. "On the Strength of Connectedness of a Random Graph." In: *Acta Mathematica Hungarica* 12.1 (1961), pp. 261–267 (cit. on p. 11).

[9] Stanton A Glantz, Bryan K Slinker, and Torsten B Neilands. *Primer of Applied Regression & Analysis of Variance*. Vol. 654. McGraw-Hill Inc., 2001 (cit. on pp. 4, 9, 23).

[10] Clark Glymour, Kun Zhang, and Peter Spirtes. "Review of Causal Discovery Methods Based on Graphical Models." In: *Frontiers in Genetics* 10 (2019), p. 524 (cit. on p. 2).

[11] Christina Heinze-Deml, Marloes H Maathuis, and Nicolai Meinshausen. "Causal Structure Learning." In: *Annual Review of Statistics and Its Application* 5 (2018), pp. 371–391 (cit. on p. 2).

[12] Patrik O Hoyer et al. "Nonlinear causal discovery with additive noise models." In: *Advances in Neural Information Processing Systems*. Vol. 21. Citeseer. 2008, pp. 689–696 (cit. on p. 2).

[13] Guido W Imbens and Donald B Rubin. *Causal Inference for Statistics, Social, and Biomedical Sciences*. Cambridge University Press, 2015 (cit. on p. 1).

[14] Marcus Kaiser and Maksim Sipos. "Unsuitability of NOTEARS for Causal Graph Discovery when Dealing with Dimensional Quantities." In: *Neural Processing Letters* 54.3 (2022), pp. 1587–1595 (cit. on p. 2).

[15] Diviyan Kalainathan, Olivier Goudet, and Ritik Dutta. "Causal Discovery Toolbox: Uncovering causal relationships in Python." In: *Journal of Machine Learning Research* 21.1 (2020), pp. 1406–1410 (cit. on p. 3).

[16] Neville Kenneth Kitson et al. "A survey of Bayesian Network structure learning." In: *Artificial Intelligence Review* (2023), pp. 1–94 (cit. on p. 2).

[17] Chunlin Li, Xiaotong Shen, and Wei Pan. "Nonlinear Causal Discovery with Confounders." In: *Journal of the American Statistical Association* (2023) (cit. on p. 2).

[18] Lars Lorch et al. "Amortized Inference for Causal Structure Learning." In: *NeurIPS 2022 Workshop on Causality for Real-world Impact*. 2022 (cit. on p. 2).

[19] Christopher Meek. "Graphical Models: Selecting causal and statistical models." PhD thesis. Carnegie Mellon University, 1997 (cit. on pp. 2, 6).

[20] Phillip B Mogensen, Nikolaj Thams, and Jonas Peters. "Invariant Ancestry Search." In: *International Conference on Machine Learning*. PMLR. 2022, pp. 15832–15857 (cit. on p. 2).

[21] Joris M Mooij, Dominik Janzing, et al. "Regression by dependence minimization and its application to causal inference in additive noise models." In: *Proceedings of the 26th annual international conference on machine learning*. 2009, pp. 745–752 (cit. on p. 2).

[22] Joris M Mooij, Sara Magliacane, and Tom Claassen. "Joint Causal Inference from Multiple Contexts." In: *Journal of Machine Learning Research* 21.1 (2020), pp. 3919–4026 (cit. on p. 19).

[23] Gunwoong Park. "Identifiability of Additive Noise Models Using Conditional Variances." In: *Journal of Machine Learning Research* 21.75 (2020), pp. 1–34 (cit. on pp. 2, 3).

[24] Judea Pearl. *Causality*. Cambridge University Press, 2009 (cit. on p. 1).

[25] Fabian Pedregosa et al. "Scikit-learn: Machine learning in Python." In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830 (cit. on p. 18).

[26] Jonas Peters and Peter Bühlmann. "Identifiability of Gaussian structural equation models with equal error variances." In: *Biometrika* 101.1 (2014), pp. 219–228 (cit. on p. 3).

[27] Jonas Peters and Peter Bühlmann. "Structural Intervention Distance for Evaluating Causal Graphs." In: *Neural computation* 27.3 (2015), pp. 771–799 (cit. on p. 7).

[28] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. The MIT Press, 2017 (cit. on p. 1).

[29] Jonas Peters, Joris M Mooij, et al. "Identifiability of Causal Graphs using Functional Models." In: *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 2011, pp. 589–598 (cit. on p. 2).

[30] Joseph D Ramsey et al. "TETRAD—A Toolbox for Causal Discovery." In: *8th International Workshop on Climate Informatics*. 2018 (cit. on pp. 3, 18).

[31] Alexander G 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*. Vol. 34. 2021, pp. 27772–27784 (cit. on pp. 1, 2, 4, 5, 7–9, 13, 18, 20, 22).

[32] Felix L. Rios, Giusi Moffa, and Jack Kuipers. "Benchpress: A Scalable and Versatile Workflow for Benchmarking Structure Learning Algorithms." In: *arXiv* (July 2021) (cit. on p. 2).

[33] Paul Rolland et al. "Score Matching Enables Causal Discovery of Nonlinear Additive Noise Models." In: *International Conference on Machine Learning*. PMLR. 2022, pp. 18741–18753 (cit. on p. 2).

[34] Gideon Schwarz. "Estimating the Dimension of a Model." In: *The Annals of Statistics* 6.2 (1978), pp. 461–464 (cit. on pp. 5, 18).

[35] Marco Scutari. "Learning Bayesian Networks with the bnlearn R Package." In: *Journal of Statistical Software* 35.3 (2010) (cit. on p. 3).

[36] Skipper Seabold and Josef Perktold. "Statsmodels: Econometric and Statistical Modeling with Python." In: *Proceedings of the 9th Python in Science Conference*. Vol. 57. 61. 2010, pp. 10–25080 (cit. on p. 18).

[37] Jonas Seng et al. "Tearing Apart NOTEARS: Controlling the Graph Prediction via Variance Manipulation." In: *arXiv preprint arXiv:2206.07195* (2022) (cit. on p. 2).

[38] Shohei Shimizu et al. "Finding a causal ordering via independent component analysis." In: *Computational Statistics & Data Analysis* 50.11 (2006), pp. 3278–3293 (cit. on p. 3).

[39] Peter Spirtes and Clark Glymour. "An Algorithm for Fast Recovery of Sparse Causal Graphs." In: *Social Science Computer Review* 9.1 (1991), pp. 62–72 (cit. on pp. 2, 6).

[40] Peter Spirtes, Clark N Glymour, et al. *Causation, Prediction, and Search*. The MIT Press, 2000 (cit. on p. 1).

[41] Marc Teyssier and Daphne Koller. "Ordering-Based Search: A Simple and Effective Algorithm for Learning Bayesian Networks." In: *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 2005, pp. 584–590 (cit. on p. 5).

[42] Matthew J. Vowels, Necati Cihan Camgoz, and Richard Bowden. "D'Ya Like DAGs? A Survey on Structure Learning and Causal Discovery." In: *ACM Computing Surveys* 55.4 (2022) (cit. on p. 2).

[43] Sebastian Weichwald et al. "Causal structure learning from time series: Large regression coefficients may predict causal links better in practice than small p-values." In: *Proceedings of the NeurIPS 2019 Competition and Demonstration Track, Proceedings of Machine Learning Research (PMLR)*. Vol. 123. 2020, pp. 27–36 (cit. on p. 2).

[44] Sewall Wright. "The Method of Path Coefficients." In: *The Annals of Mathematical Statistics* 5.3 (1934), pp. 161–215 (cit. on p. 22).

[45] Sascha Xu et al. "Inferring Cause and Effect in the Presence of Heteroscedastic Noise." In: *International Conference on Machine Learning*. PMLR, 2022, pp. 24615–24630 (cit. on p. 2).

[46] Xun Zheng et al. "DAGs with NO TEARS: Continuous Optimization for Structure Learning." In: *Advances in Neural Information Processing Systems*. Vol. 32. 2018, pp. 9472–9483 (cit. on pp. 2, 4).

A Causal Discovery Experiments

We provide an open-source implementation of R^2-sortability and var-sortability as well as the R^2-*SortnRegress* and R^2-*GDS* algorithms in our library CausalDisco.

In our experiments we make use of the *scikit-learn* (Pedregosa et al. 2011) and *statsmodels* (Seabold et al. 2010) Python packages as well as the Python interface to the *igraph* (Csardi et al. 2006) package. We use the implementation by Ramsey et al. (2018) of the *PC* and *FGES* algorithms. For *sortnregress*, we rely on the implementation provided by Reisach et al. (2021). For *MSE-GDS*, we use an implementation following Reisach et al. (2021) augmented by a regulatization based on the Bayesian Information Criterion (Schwarz 1978) as described in Section 3.2.

The algorithms *PC* and *FGES* return a completed partially directed acyclic graph (CPDAG), while the other algorithms return a DAG. For evaluation in terms of Structural Intervention Distance (SID) and Structural Hamming Distance (SHD), we direct the CPDAGs favorably by orienting all undirected edges in the right direction if the nodes are connected in the true graph. All algorithms except for *PC* (which uses the Fisher z-test) are regularized using the Bayesian Information Criterion. The sparsity induced by the regularization effectively strikes a balance between SID and SHD performance because the former does not penalize wrongfully inserted edges while the latter does. An unregularized version would therefore likely perform better in terms of SID but worse in terms of SHD. For the evaluation of the recovery of the Markov Equivalence Class (MEC), we transform all true and recovered DAGs into CPDAGs before calculating the SID and SHD between them. For all settings, we independently sample 30 graphs to obtain error bars and run our algorithms on 1000 iid observations from each graph. Across simulation regimes, we use the edge weight distribution $P_W = \text{Unif}((-2, -0.5) \cup (0.5, 2))$.

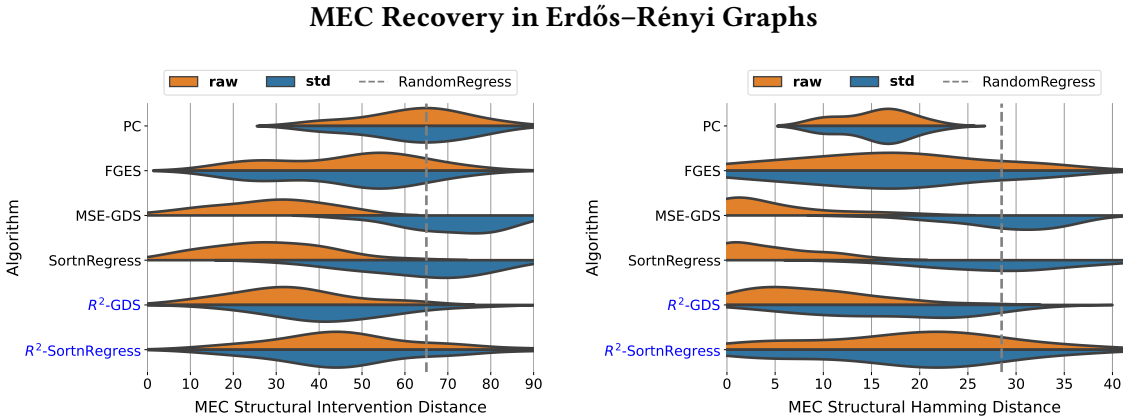
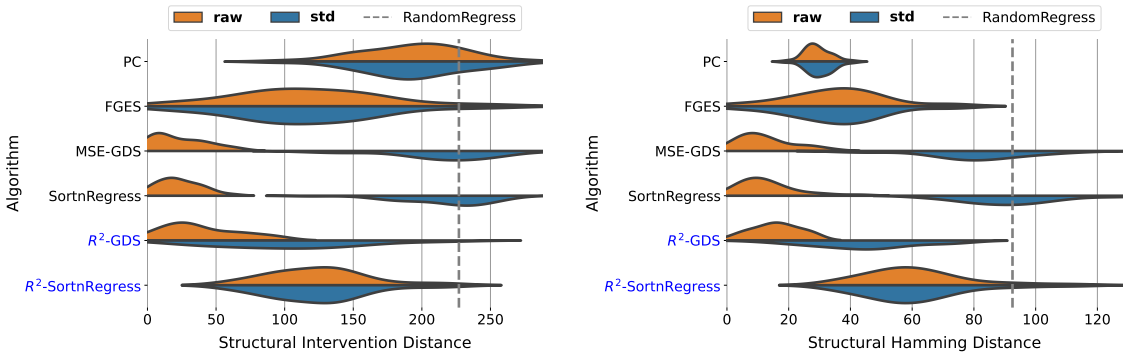
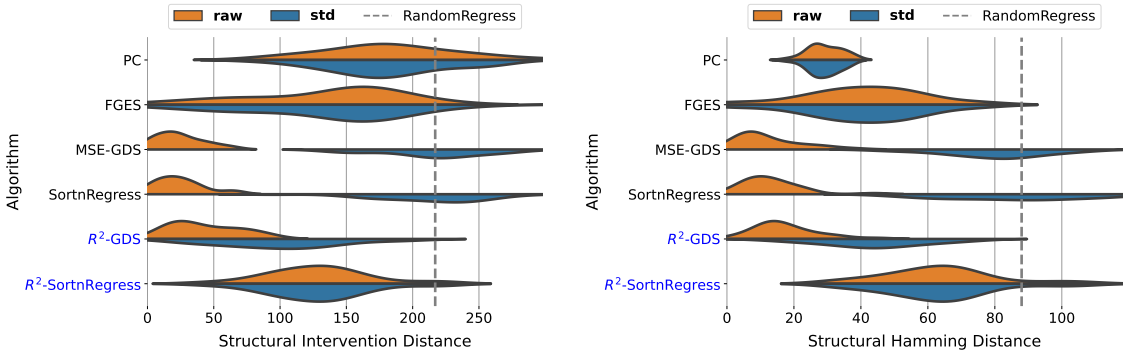


Figure 7: ANM parameters $P_{\mathcal{G}} = \mathcal{G}_{\text{ER}}(10, 20)$, $\mathcal{P}_N(\phi) = \mathcal{N}(0, \phi^2)$, and $P_\sigma = \text{Unif}(0.5, 2)$.



ANM parameters $P_{\mathcal{G}} = \mathcal{G}_{\text{ER}}(20, 40)$, $\mathcal{P}_N(\phi) = \mathcal{N}(0, \phi^2)$, and $P_\sigma = \textbf{Unif}(0.5, 2)$.



ANM parameters $P_{\mathcal{G}} = \mathcal{G}_{\text{ER}}(20, 40)$, $\mathcal{P}_N(\phi) = \mathcal{N}(0, \phi^2)$, and $P_\sigma = \textbf{Exp}(1.54)$.

Figure 8: DAG recovery in $\mathcal{G}_{\text{ER}}(20, 40)$ graphs with noise standard deviations drawn from uniform (first row) and exponential (second row) distributions.

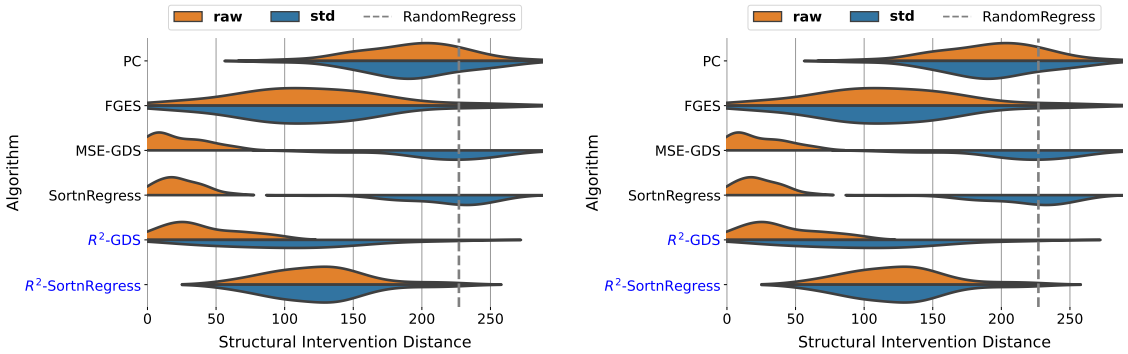


Figure 9: ANM parameters $P_{\mathcal{G}} = \mathcal{G}_{\text{ER}}(20, 40)$, $\mathcal{P}_N(\phi) = \mathcal{N}(0, \phi^2)$, and $P_\sigma = \text{Unif}(0.5, 2)$. Using the weight harmonization scheme by Mooij, Magliacane, et al. (2020).

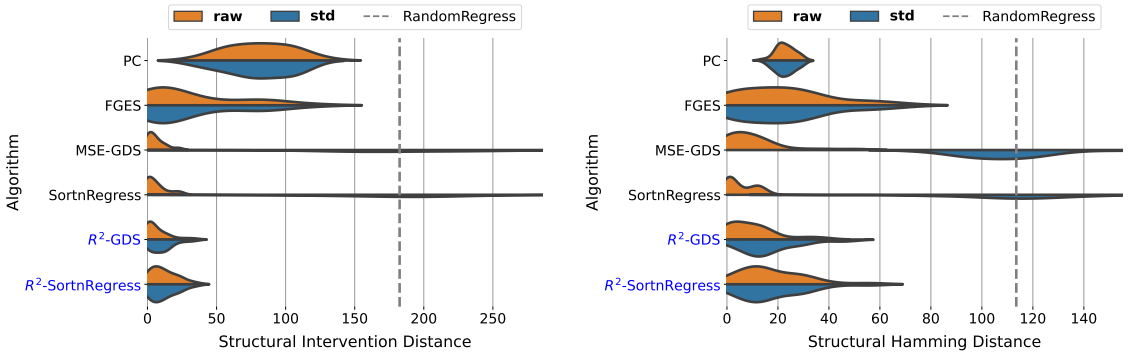


Figure 10: ANM parameters $P_{\mathcal{G}} = \mathcal{G}_{\text{SF}}(10, 20)$, $\mathcal{P}_N(\phi) = \mathcal{N}(0, \phi^2)$, and $P_\sigma = \text{Unif}(0.5, 2)$.

B Performance Sensitivity to R^2-sortability

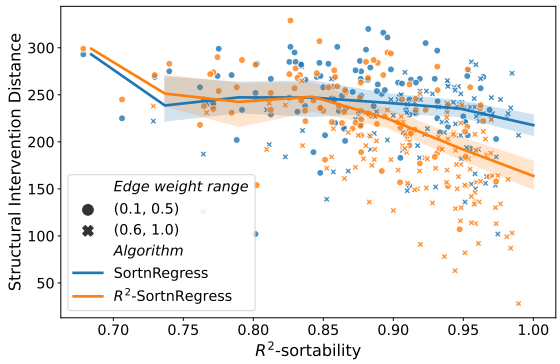


Figure 11: Algorithm performance in SID for different values of R^2-sortability on standardized data.

Figure 11 shows the performance of R^2-*SortnRegress* and *SortnRegress* for a range of R^2-sortabilities in data from ANMs sampled as described in Section 3.3.2. As with pre-standardization var-sortability (Figure 2), the performance of R^2-*SortnRegress* improves for higher R^2-sortability .

C Comparison of Var-sortability Definitions

The original definition of var-sortability by Reisach et al. (2021) measures the fraction of all paths of different length between each cause and effect pair where cause and effect are correctly ordered by variance. We propose two alternative definitions to explore the impact of the distinction between paths of the same or different length between a given pair. The first alternative measures the fraction of node pairs connected directly by a single edge that are correctly sorted by variance. We refer to this version as 'pair-wise' var-sortability. The second alternative measures the fraction of all cause-effect paths, regardless of length, for which cause and effect are correctly sorted by variance. We refer to this version as 'path-wise' var-sortability. Using the same definition of $\text{incr}(a, b)$ as in

Equation (3), we define them as

$$\mathbf{v}_\tau^{(\text{pair-wise})}(X, \mathcal{G}) = \frac{\sum_{(s \to t) \in B_\mathcal{G}} \text{incr}(\tau(X, s), \tau(X, t))}{\sum_{(s \to t) \in B_\mathcal{G}} 1},$$

$$\mathbf{v}_\tau^{(\text{path-wise})}(X, \mathcal{G}) = \frac{\sum_{i=1}^{d} \sum_{(s \to t) \in B_\mathcal{G}^i} \left(B_\mathcal{G}^i\right)_{s,t} \text{incr}(\tau(X, s), \tau(X, t))}{\sum_{i=1}^{d} \sum_{(s \to t) \in B_\mathcal{G}^i} \left(B_\mathcal{G}^i\right)_{s,t}}$$

where $\left(B_\mathcal{G}^i\right)_{s,t}$ is the number of distinct directed paths from X_s to X_t of length i in $\mathcal{G}$. We obtain our measures for $\tau(x) = \text{Var}(x)$. We independently sample 500 ANMs with the parameters $P_\mathcal{G} = \mathcal{G}_{\text{ER}}(10, 20)$, $\mathcal{P}_N(\phi) = \mathcal{N}(0, \phi^2)$, $P_\sigma = \text{Unif}(0.5, 2)$, and $P_W = \text{Unif}((-2, -0.5) \cup (0.5, 2))$. From each ANM, we sample 1000 iid observations. To obtain a wide spectrum of var-sortabilities, we multiply the observations of $\mathbf{X}_t$ by e^{100/σ_t} for all t. On these data, we run *SortnRegress* and *RandomRegress* (a version of *SortnRegress* that uses a random regression order).

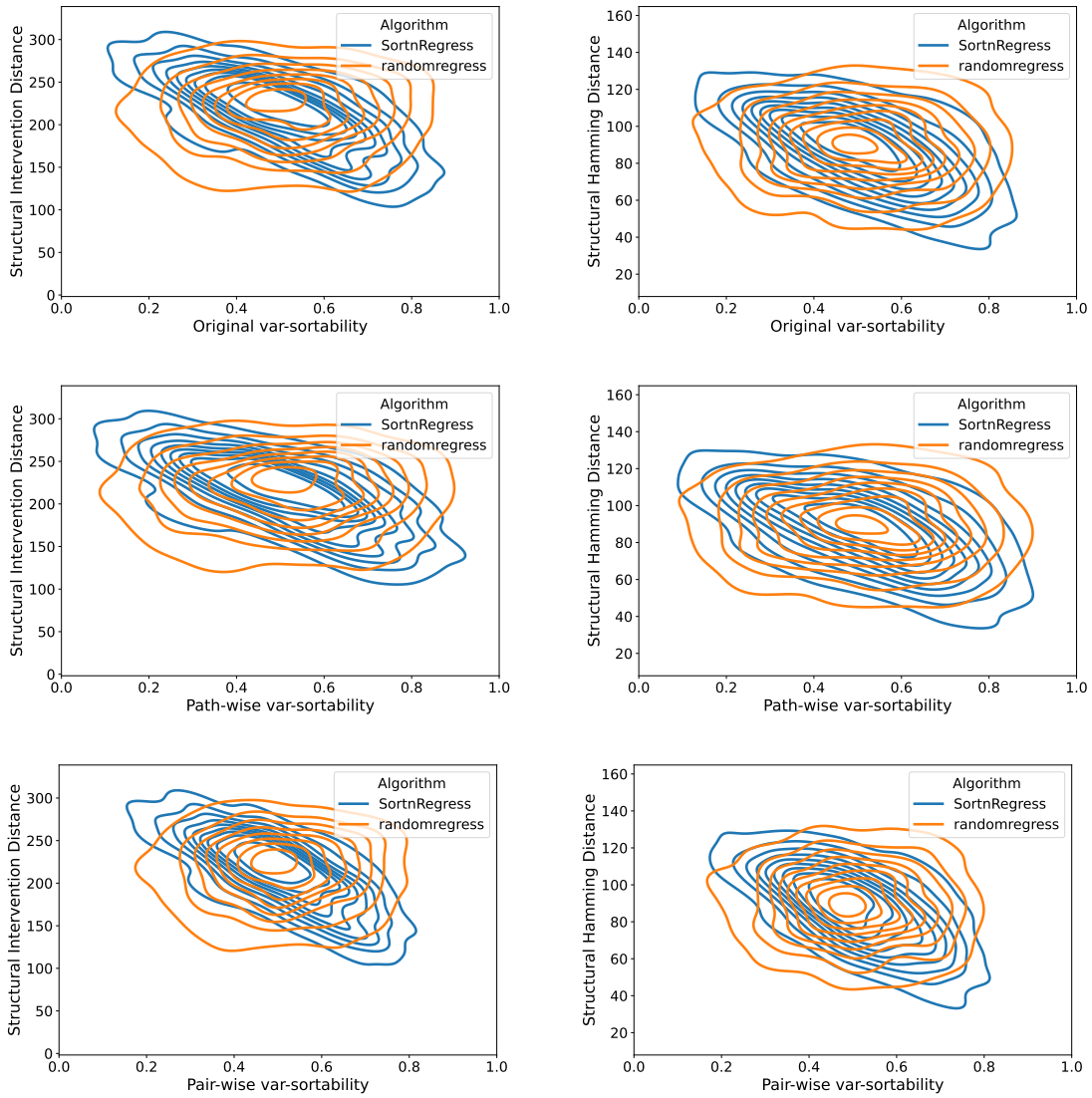


Figure 12: Comparison of SID (left column) and SHD (right column) performances between alternative definitions of var-sortability. We use contour-lines to show the results of the *SortnRegress* and R^2-*SortnRegress* algorithms.

A visualization of the results of our experiment can be seen in Figure 12. Pair-wise var-sortabilities appear to take values in a somewhat narrower range, while path-wise var-sortability makes the greatest use of the possible range $(0, 1)$. We detect no qualitative difference in the performance of the algorithms regardless of the measure and conclude that the original definition of var-sortability appears to strike a balance between these two alternative definitions.

D Var-sortability Lower Bound

We first consider the 2-chain $X_1 \to X_2$ and denote this causal graph by $\mathcal{G}_{\text{2-chain}}$. The expected var-sortability of $\mathcal{G}_{\text{2-chain}}$ when generating synthetic ANM data is determined by P_W and P_σ from which we independently sample the only non-zero edge weight $W_{1,2} \sim P_W$ and the noise standard deviations $\sigma_1, \sigma_2 \sim P_\sigma$. We can express and lower-bound the expected var-sortability for the 2-chain as follows:[6]

$$\begin{aligned} E_{W,\sigma}[\mathbf{v}_{\text{Var}}(X, \mathcal{G})] =& P_{W,\sigma}[\text{Var}(X_2) > \text{Var}(X_1)] \\ =& P\left(W_{1,2}^2\sigma_1^2 + \sigma_2^2 > \sigma_1^2\right) \\ =& P\left(W_{1,2}^2 + \frac{\sigma_2^2}{\sigma_1^2} > 1\right) \\ \geq& P\left(|W_{1,2}| \geq 1\right). \end{aligned} \tag{7}$$

We generalize Equation (7) for causal chains to characterise the emergence of var-sortability in causal chains, which also provides intuition about var-sortability in general graphs. Let source node X_s and target node X_t be two nodes such that X_s is a causal ancestor of X_t and that X_s and X_t have no common ancestors. We define $\mathcal{P}_{X_s \to X_t}$ as the set of all directed paths from X_s to X_t. A single path $p \in \mathcal{P}$ of length $|p|$ is represented by a p-tuple of edge weights along that directed path, for example, the path $X_s \to X_{s+1} \to \cdots \to X_t$ is represented by $(W_{s,s+1}, ..., W_{s+|p|-1,t})$. We can lower-bound the probability of X_s and X_t being correctly sorted by variance in terms of the square of the sum of all path products between them (cf. Wright (1934)). In a causal chain there is only a single path between the pair, so we can further simplify to obtain a lower bound analogous to Equation (7):

$$\begin{aligned} & P_{W,\sigma}[\text{Var}(X_t) > \text{Var}(X_s)] \\ & \geq P\left(\left(\sum_{p \in \mathcal{P}_{X_s \to X_t}} \prod_{w \in p} w\right)^2 > 1\right) && \text{(no common ancestors)} \\ & = P\left(\left(\prod_{w \in p^*} w\right)^2 > 1\right) && \text{(single path } p^* \in \mathcal{P}_{X_s \to X_t}\text{)} \\ & = P\left(\sum_{w \in p^*} \ln|w| > 0\right). \end{aligned} \tag{8}$$

The sum over $w \in p^*$ is a sum of iid random variables $W_{i,j} \sim P_W$. Equation (8) gives a sufficient criterion for var-sortability based only on the weight distribution P_W. For long paths, the lower bound in Equation (8) is driven by $E(\ln|V|)$ for $V \sim P_W$ which, in slight abuse of notation, we succinctly write as $E(\ln|P_W|)$.

[6] See Reisach et al. (2021), Section 3.3, which we adapt to our notation. For $W_{1,2} \sim P_W$, $\sigma_1 \sim P_\sigma$, $\sigma_2 \sim P_\sigma$, $N_1 \mid \sigma_1 = \boldsymbol{\sigma}_1 \sim \mathcal{P}_N(\boldsymbol{\sigma}_1)$, $N_2 \mid \sigma_2 = \boldsymbol{\sigma}_2 \sim \mathcal{P}_N(\boldsymbol{\sigma}_2)$, and $X = W_{1,2}N_1 + N_2$ we have $\text{Var}\left(X \mid W_{1,2} = w, \sigma = \boldsymbol{\sigma}\right) = w^2\boldsymbol{\sigma}_1^2 + \boldsymbol{\sigma}_2^2$ and write $P_{W,\sigma}[\text{Var}(X) > 0] = P[W_{1,2}^2\sigma_1^2 + \sigma_2^2 > 0]$.

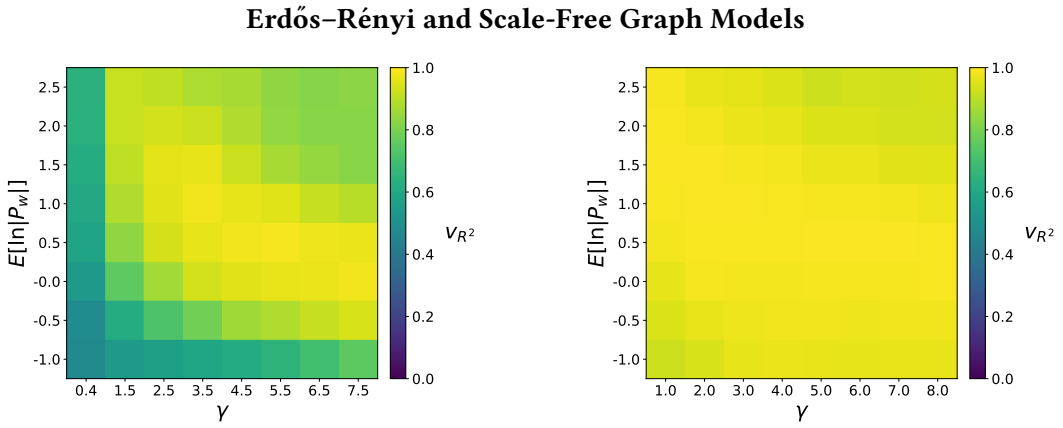
Var-sortability lower bound in chains. If the edge weight distribution P_W is chosen such that $E(\ln|V|) > 0$ for $V \sim P_W$, we can further lower bound Equation (8) using Cantelli's inequality (Glantz et al. 2001). Consider a chain graph and let X_s and X_t be two nodes such that X_s is a causal ancestor of X_t. Since we consider a chain graph, X_s and X_t have no common ancestors and there is a unique directed path from X_s to X_t which we denote by p^*. For $Z = \sum_{w \in p^*} \ln|w|$ with $E(Z) = |p^*|E(\ln|V|)$ and $\mathrm{Var}(Z) = |p^*|\mathrm{Var}(\ln|V|)$, we have that

$$
\begin{aligned}
& P_{W,\sigma}[\mathrm{Var}(X_t) > \mathrm{Var}(X_s)] && (9)\\
& \geq P\left(\sum_{w \in p^*} \ln|w| > 0\right) \\
& = 1 - P(Z - E(Z) \leq -E(Z)) \\
& \geq 1 - \frac{\mathrm{Var}(Z)}{\mathrm{Var}(Z) + (E(Z))^2} \\
& = 1 - \frac{\mathrm{Var}(\ln|V|)}{\mathrm{Var}(\ln|V|) + |p^*|E(\ln|V|)^2}. && (10)
\end{aligned}
$$

The lower bound on the probability of X_s and X_t being correctly sorted by variance when sampling the edge weights from P_W increases with the length of the path from X_s to X_t and converges as the distance between the node pair increases. The bound highlights the importance of the weight distribution P_W for var-sortability in the simulated ANMs.

E Var-sortability for Different Simulation Parameter Choices

In this section, we present the var-sortability obtained in graphs with a range of graph density parameters γ and geometric weight means $E(\log|P_W|)$ for different random graph models, noise distributions, and noise standard deviation distributions. For each combination of parameters, we compute the mean value of $\mathbf{v}_{R^2}$ for 10 independently sampled graphs with $d = 50$ nodes and 1000 iid observations per graph.

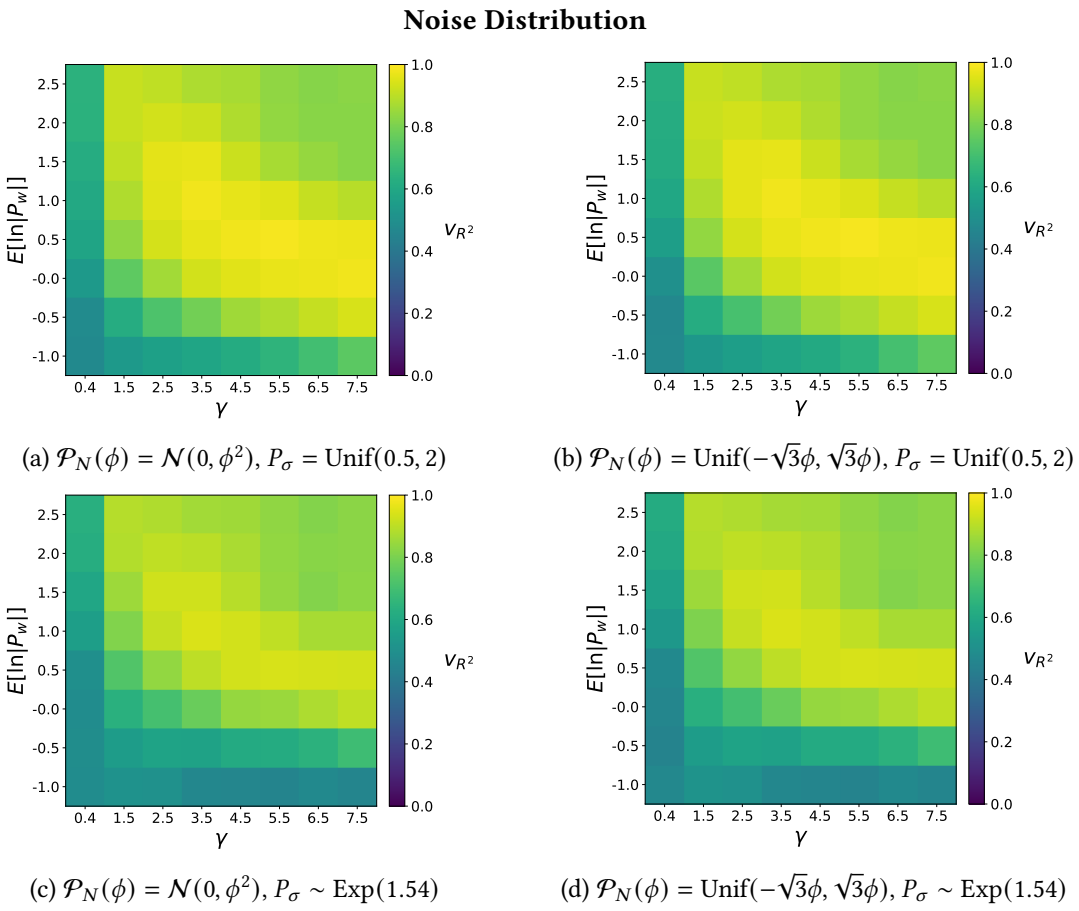


(a) Erdős–Rényi graphs

(b) Scale-free graphs

Figure 13: Sensitivity of R^2-sortability to parameters in 50-node graphs with $\mathcal{P}_N(\phi) = \mathcal{N}(0, \phi^2)$, $P_\sigma \sim \mathrm{Unif}(0.5, 2)$ to γ and $E[\ln|P_W|]$ for Erdős–Rényi and Scale-free random graphs.

Figure 13 shows a comparison of R^2-sortability in Erdős–Rényi and Scale-free graphs. We choose slightly different values of λ for the two settings because the Scale-free graph generating mechanism requires integer values. We see that R^2-sortability is much higher across the board for Scale-free graphs. The relationship between the parameters and R^2-sortability observed for Erdős–Rényi graphs remains faintly visible for Scale-free graphs.



(a) $\mathcal{P}_N(\phi) = \mathcal{N}(0, \phi^2)$, $P_\sigma = \text{Unif}(0.5, 2)$

(b) $\mathcal{P}_N(\phi) = \text{Unif}(-\sqrt{3}\phi, \sqrt{3}\phi)$, $P_\sigma = \text{Unif}(0.5, 2)$

(c) $\mathcal{P}_N(\phi) = \mathcal{N}(0, \phi^2)$, $P_\sigma \sim \text{Exp}(1.54)$

(d) $\mathcal{P}_N(\phi) = \text{Unif}(-\sqrt{3}\phi, \sqrt{3}\phi)$, $P_\sigma \sim \text{Exp}(1.54)$

Figure 14: Sensitivity of var-sortability in $\mathcal{G}_{\text{ER}}(50, \gamma 50)$ graphs to γ and $E[\ln|P_W|]$ for different noise distributions $\mathcal{P}_N$ and noise standard deviation distributions P_σ.

Figure 14 shows the impact of the noise and noise parameter distribution on R^2-sortability. We observe a similar trend of high R^2-sortability for moderate choices of $E[\ln|P_w|]$ and γ. R^2-sortability is lowest when either of these lakes very low values. In the setting of $P_\sigma = \text{Exp}(1.54)$ we observe lower values of R^2-sortability including values close to randomness ($\mathbf{v}_{R^2} \approx 0.5$) even for dense graphs given sufficiently low edge weight magnitudes. This could be explained by the higher skewness and positive unboundedness of the exponential distribution which may allow it to disrupt R^2 patterns.

Note that parameters to the noise standard deviation P_σ do not impact var-sortability for many noise distributions $\mathcal{P}_N(\phi)$, including ours. Since the variable variances are linear transformations of the noise variances, see Equation (2), the condition for all paths between a pair of variables X_s, X_t being correctly ordered by variance according to Equation (3) is of the form

$$\frac{\text{Var}(X_s)}{\text{Var}(X_t)} = \frac{\sum_{s=1}^{K} \frac{\sigma_s^2}{\sigma} a_s}{\sum_{t=1}^{L} \frac{\sigma_t^2}{\sigma} b_t} < 1$$

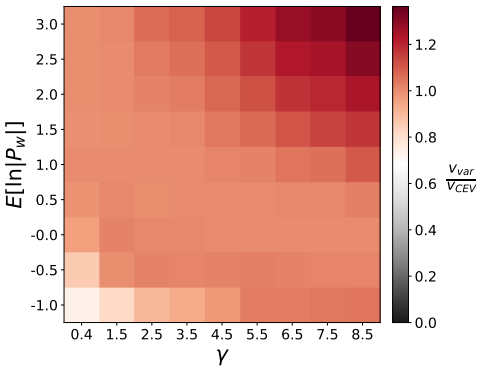
for some $a_1, \ldots, a_K$ and $b_1, \ldots, b_L$. For any parametrized distribution $P_\sigma(\psi)$ such that $Y \sim P_\sigma(\psi)$ with $\frac{Y}{\text{std}(Y)} \sim P_\sigma(1)$, the parameter of the noise standard deviation distribution does therefore not affect var-sortability, although it may still affect R^2-sortability.

F Relationship Between the Different Sortabilities

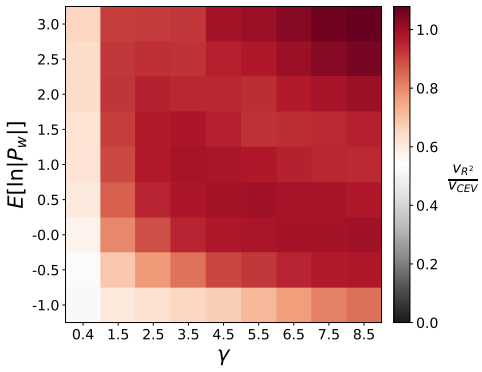
Our use of R^2 is motivated by our reasoning about an increase in the fraction of cause-explained variance (CEV) along the causal order for data with high var-sortability (see Section 3.1). On real

data we do not know the causal parents, so we cannot compute the CEV. However, it may be instructive to analyze the relationship between var-sortability, R^2-sortability, and CEV-sortability in our main simulation setting. We obtain CEV-sortability using the definition of τ-sortability (Equation (3)) for $\tau(X, t) = R^2(M^{\theta^*}_{t,\mathbf{Pa}(X_t)}, X)$ and denote it as $\mathbf{v}_{\text{CEV}}$.

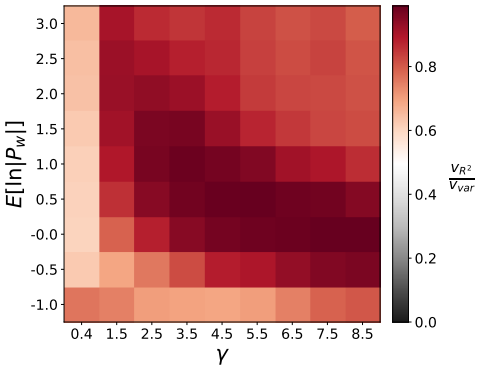
F.1 Erdős–Rényi Graphs



(a) Var-sortability and CEV-sortability



(b) R^2-sortability and CEV-sortability

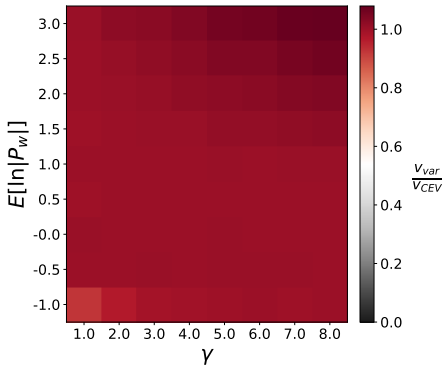


(c) R^2-sortability and Var-sortability

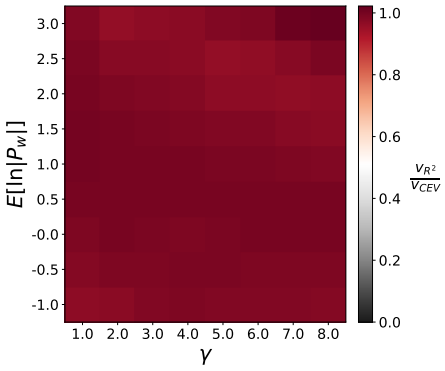
Figure 15: Ratio between sortabilities for different $E[\ln|P_W|]$ and γ in $\mathcal{G}_{\text{ER}}(50, \gamma 50)$ graphs with $\mathcal{P}_N(\phi) = (0, \phi^2)$ and $P_\sigma = \text{Unif}(0.5, 2)$.

Our algorithms are motivated by the connection between var-sortability and CEV-sortability. They make use of R^2-sortability as a proxy for CEV-sortability. We can see in Figure 15a that CEV-sortability indeed tracks var-sortability closely in most settings as hypothesized and indicated by our analysis of the weight distribution. The two measures disagree when both parameters take extreme values. CEV-sortability takes higher values compared to var-sortability for very low values of $E[\ln|P_W|]$ and γ, and lower values when both parameters are high. It appears that higher weights and denser graphs lead to a stronger pattern in variance, but not necessarily in CEV. In Figure 15b, we see that R^2-sortability is a good proxy for CEV-sortability in all settings except when weights are extremely low or the graphs are extremely sparse. As a result, R^2-sortability can be seen to track var-sortability well across most settings, unless one of the parameters is very low or they are both very high.

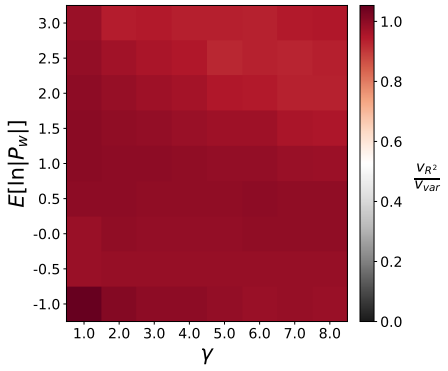
F.2 Scale-Free Graphs



(a) Var-sortability and CEV-sortability



(b) R^2-sortability and CEV-sortability



(c) R^2-sortability and Var-sortability

Figure 16: Ratio between sortabilities for different $E[\ln|P_W|]$ and γ in $\mathcal{G}_{SF}(50, \gamma 50)$ graphs with $\mathcal{P}_N(\phi) = (0, \phi^2)$ and $P_\sigma = \text{Unif}(0.5, 2)$.

We choose slightly different values of λ compared to Figure 15 because the Scale-free graph generating mechanism requires integer values. As can be seen in Figure 16, all three of $\mathbf{v}_{\text{Var}}$, $\mathbf{v}_{R^2}$, and $\mathbf{v}_{\text{CEV}}$ are almost perfectly aligned in all settings. While the alignment between $\mathbf{v}_{R^2}$ and $\mathbf{v}_{\text{Var}}$ does diminish somewhat when $E[\ln|P_W|]$ and γ are large and $\mathbf{v}_{\text{CEV}}$ matches or exceeds $\mathbf{v}_{\text{Var}}$, it is substantially stronger still than the already good alignment in Erdős–Rényi graphs (see Figure 15).