
Rank-Based Causal Discovery for Post-Nonlinear Models

Grigor Keropyan

Technical University of Munich

David Strieder

Technical University of Munich
Munich Center for Machine Learning

Mathias Drton

Technical University of Munich
Munich Center for Machine Learning

Abstract

Learning causal relationships from empirical observations is a central task in scientific research. A common method is to employ structural causal models that postulate noisy functional relations among a set of interacting variables. To ensure unique identifiability of causal directions, researchers consider restricted subclasses of structural causal models. Post-nonlinear (PNL) causal models constitute one of the most flexible options for such restricted subclasses, containing in particular the popular additive noise models as a further subclass. However, learning PNL models is not well studied beyond the bivariate case. The existing methods learn non-linear functional relations by minimizing residual dependencies and subsequently test independence from residuals to determine causal orientations. However, these methods can be prone to overfitting and, thus, difficult to tune appropriately in practice. As an alternative, we propose a new approach for PNL causal discovery that uses rank-based methods to estimate the functional parameters. This new approach exploits natural invariances of PNL models and disentangles the estimation of the non-linear functions from the independence tests used to find causal orientations. We prove consistency of our method and validate our results in numerical experiments.

1 INTRODUCTION

Discovering the causal structure of complex systems is an important question in various disciplines such as biology, economics, clinical medicine, or neuroscience (Opgenrhein and Strimmer, 2007; Glymour et al., 2019; Moneta

et al., 2013). The gold standard approach to exploration of causal relations is to perform controlled experiments in which researchers externally intervene in the system and observe the resulting changes to variables of interest. However, in many applications controlled experiments are not feasible due to high cost or for ethical reasons. In such cases, causal discovery based on only observational data can be a useful tool (Spirtes and Zhang, 2016).

A common tool for modeling causal relations are Structural Equation Models (SEMs). In their general form, SEMs postulate noisy functional relationships between a set of interacting variables. In the fully general setting, causal discovery methods such as constraint-based and score-based methods can identify the underlying causal structure only up to Markov equivalence classes (Spirtes et al., 2000). Thus, the literature has also considered many restricted subclasses that enable unique identification (Drton et al., 2011; Hoyer et al., 2008; Peters et al., 2014; Zhang and Hyvärinen, 2009). In this realm, post-nonlinear (PNL) causal models constitute one of the most general approaches. They are identifiable from the joint distribution under mild assumptions (Zhang and Hyvärinen, 2009; Peters et al., 2014) and yet offer a rather flexible framework for modeling complex non-linear causal systems.

Existing methods for bivariate PNL causal discovery are based on estimating the functional relations by minimizing independence criteria (HSIC, mutual information, etc.) between the noise and potential parents in a first step, and performing independence tests to determine the causal structure in a second step (Zhang and Hyvärinen, 2009; Uemura and Shimizu, 2020). However, minimizing dependence to subsequently test for independence leads to potential overfitting and thus limits the PNL approach. Another approach considered by Tu et al. (2022) employs Optimal Transport theory for bivariate post-nonlinear and additive noise causal discovery, but it is not evident how to generalize their method to multivariate models. To our knowledge the only work that deals with multivariate PNL models is Uemura et al. (2022), where the authors generalize the bivariate method from Uemura and Shimizu (2020) based on minimizing dependence and subsequently testing for independence.

In this article we present a new method for multivariate causal discovery in PNL models that disentangles the two tasks of learning the functional relations and learning the causal structure. Our method continues to learn the latter with the help of independence tests, but it employs rank-based methods to learn the functional relations and, in this way, avoids overfitting issues.

The remainder of the paper is organized as follows. In Section 2 we introduce the PNL rank regression methods that we use to learn the functional relations. We study the special case of linearity in the inner function and show consistency of our proposed rank-based functional parameter estimates. This special case includes general nonlinear functions using basis expansions. In Section 3 we discuss the causal order learning routine using our proposed rank-based estimates in a recursive process that finds sink nodes by independence testing. Furthermore, we show consistency of the causal order estimation for PNL models and present the results of a simulation study in Section 4, where we compare our method to existing causal learning methods. Section 5 concludes the paper.

2 PNL RANK REGRESSION

In this section we introduce the rank-based estimators that we use in the first step of our proposed causal learning algorithm. The goal is to employ these rank-based estimators to infer the functional relations among the variables and to obtain estimates of the stochastic noise terms in the model. In the second step, we use the estimated noise terms to test for independence.

Suppose we observe a sample of n independent copies $(X_1, Y_1), \dots, (X_n, Y_n)$ of a random vector (X, Y) . We assume the data generating process follows a PNL model, that is, the response variable $Y \in \mathbb{R}$ is given by

$$h(Y) = g(X) + \varepsilon, \quad (1)$$

where $X \in \mathbb{R}^m$ is a continuous random vector and the stochastic error term ε has mean zero with unknown continuous distribution, independent of X . Furthermore, we assume that the function $h : \mathbb{R} \rightarrow \mathbb{R}$ is continuous and strictly increasing (thus, invertible) whereas $g : \mathbb{R}^m \rightarrow \mathbb{R}$ may be an arbitrary function.

Under similar assumptions, Zhang and Hyvärinen (2009) suggested to estimate the noise $\varepsilon = h(Y) - g(X)$ by representing h and g with Multi-layer Perceptrons (MLPs) and minimizing mutual information with X via gradient-based methods. However, the main drawback of their methodology is that this model can fit perfectly to any data by learning constant functions h and g . The estimated noise will be constant and thus always independent from X .

To overcome this problem Uemura and Shimizu (2020) implemented an additional auto-encoding structure in the

minimization problem that enforces invertibility of the function h . While this circumvents the problem of constant estimation of the function h , there are further challenges that arise from minimizing dependence and subsequent testing for independence. Indeed, the complexity of the function class assumed for g needs to be balanced very carefully with the available sample size. Otherwise, g can be fitted perfectly such that $g(X) = h(Y)$, in which case the functional estimates cancel and the estimated noise is always independent of X . Uemura and Shimizu (2020) used a fixed architecture for the function classes of g and h . As a result, especially for small sample sizes (compared to the complexity of the function class of g), their method is prone to overfitting and canceling the effect of the function h . Such overfitting may then entail erroneous results in independence tests for causal structure learning.

To avoid the noted overfitting issues, we propose the following two-stage method to learn the functional relations. In the first stage, we leverage rank statistics to separately estimate the function g , without any appeal to measures of dependence between the noise and the predictor X . The strictly increasing function h preserves the ranks of Y and thus, using rank-based methods, we can avoid estimating h at this stage. This circumvents the problem of g matching h . In the second step, we estimate the functional relation h at all observed data points to obtain the required estimates of the noise.

In order to simplify the concept and a theoretical analysis of our proposed method, we assume in the following linearity of the function g , i.e. $g(X) = X^T \beta_0$ for $\beta_0 \in \mathbb{R}^m$. This can also be seen as a first order Taylor approximation of an arbitrary functional relation.

Remark 2.1. *Our framework and the idea of disentangling learning and testing by employing rank-based objective functions can be easily extended to the nonlinear case. By employing basis expansions, MLPs or any parametric function class in combination with the proposed rank-based scores to learn the functional relations one can trace the steps of the presented linear case. For instance, consider the basis functions $\{b_l(\cdot) : l = 1, \dots, a_n\}$, where $a_n \rightarrow \infty$ sufficiently slowly, similar to Bühlmann et al. (2014). Then we can represent the (nonlinear) function g by $\sum_{l=1}^{a_n} \alpha_l b_l(\cdot)$, where $\alpha_l \in \mathbb{R}$ for all $l = 1, \dots, a_n$, and employ our proposed framework. The simulations in Section 4 include an example.*

We start by studying the special case of model (1) under the assumption of Gaussian noise and derive a computationally efficient algorithm for estimating the functional parameters. Further, in Subsection 2.2 we consider the general case without restricting the noise distribution. The main idea of our approach is to leverage rank likelihoods, however, using the full marginal rank likelihood is not computationally tractable. A common approach to circumvent calculating the full marginal rank likelihood is to employ ap-

proximate Monte Carlo methods, e.g., considered by Doksum (1987). However, we observed that this approach does not work well in practice for values of β_0 larger than one. Thus, in our proposed framework we employ pairwise rank likelihoods to approximate the full marginal rank likelihood (in the Gaussian case) or the rank correlation function (in the general noise case).

2.1 Gaussian Case

We assume that the data generating process follows the model

$$h(Y) = X^T \beta_0 + \varepsilon, \quad (2)$$

with some unknown $\beta_0 \in \mathbb{R}^m$. Furthermore, we assume that the noise ε is standard normal distributed and propose the following computationally fast algorithm to estimate the functional relations. The idea of this method is based on Yu et al. (2021).

We exploit the fact that h is a strictly increasing function and therefore preserves the ranks of $\{Y_i\}_{i=1}^n$. The normality assumption yields $\varepsilon_i - \varepsilon_j \sim \mathcal{N}(0, 2)$ and we obtain

$$\begin{aligned} \mathbb{P}(Y_j > Y_i | X_j, X_i) &= \mathbb{P}(h(Y_j) > h(Y_i) | X_j, X_i) \\ &= \mathbb{P}(\varepsilon_i - \varepsilon_j < (X_j - X_i)^T \beta_0 | X_j, X_i) \\ &= \Phi\left(\frac{(X_j - X_i)^T \beta_0}{\sqrt{2}}\right), \end{aligned}$$

where Φ is the cumulative distribution function of the standard normal distribution. The normalized log pairwise rank likelihood function is then given by

$$\begin{aligned} \ell_{prl}(\beta) &:= \binom{n}{2}^{-1} \sum_{i < j} \mathbb{1}\{Y_j > Y_i\} \log \Phi\left(\frac{(X_j - X_i)^T \beta}{\sqrt{2}}\right) \\ &\quad + \mathbb{1}\{Y_j \leq Y_i\} \log \Phi\left(\frac{(X_i - X_j)^T \beta}{\sqrt{2}}\right). \end{aligned} \quad (3)$$

We estimate β_0 by maximizing ℓ_{prl} , that is,

$$\hat{\beta}_{prl} := \arg \max_{\beta \in \mathbb{R}^m} \ell_{prl}(\beta).$$

This defines a concave optimization problem, which leads to a computationally fast estimation routine for β_0 without precise knowledge or estimation of the function h .

Proposition 2.1. *The log pairwise rank likelihood function $\ell_{prl}(\beta)$ defined in (3) is concave. Moreover, if we assume that $n > m$, then $\ell_{prl}(\beta)$ is strictly concave.*

The proof can be found in Appendix B.

Furthermore, the proposed estimator is consistent.

Theorem 2.1. *As $n \rightarrow \infty$ (and in particular $n > m$), it holds that*

$$\hat{\beta}_{prl} - \beta_0 = o_P(1).$$

For a proof we refer the reader to Appendix C.

In order to obtain an estimate of the noise, we estimate the transformation function h in a second step. We employ the following computationally fast and consistent estimation routine proposed by Cuzick (1988). In his proposal, Cuzick (1988) considers non-random covariates, however, the results are applicable to our setup conditional on the observed data $\{X_i\}_{i=1}^n$. The method exploits the normality assumption as well as knowledge of the ranks of $\{Y_i\}_{i=1}^n$ and thus of $\{h(Y_i)\}_{i=1}^n$.

Let $\hat{F}(z)$ be the adjusted empirical distribution function of $Z_i := h(Y_i)$, that is

$$\hat{F}(z) := \frac{1}{n+1} \sum_{i=1}^n \mathbb{1}\{Z_i \leq z\}.$$

Remark 2.2. *Note that we only require the ranks of $\{h(Y_i)\}_{i=1}^n$ to obtain the estimate $\hat{F}(h(Y_i))$. Since h is a strictly increasing function, the ranks of $\{h(Y_i)\}_{i=1}^n$ are given by the ranks of $\{Y_i\}_{i=1}^n$.*

We denote the cumulative distribution function of a randomly chosen Z_i by $F_\beta(z)$, that is

$$F_\beta(z) := \frac{1}{n} \sum_{i=1}^n \Phi(z - X_i^T \beta). \quad (4)$$

Then an estimator for the functional relation h at the sample points $\{Y_i\}_{i=1}^n$ is given by

$$\hat{h}_G(Y_i) := F_{\hat{\beta}_{prl}}^{-1}(\hat{F}(h(Y_i))), \quad i = 1, \dots, n. \quad (5)$$

By extending this estimator to a step function on \mathbb{R} , and under some additional assumptions, Cuzick (1988) show that this estimator converges to h almost surely at all continuity points of the function h .

Remark 2.3. *In our setup, we assume h is continuous. Furthermore, the additional assumptions that ensure consistency in the setting from Cuzick (1988) are mainly smoothness and moment conditions on the distributions of X and ε . In the considered Gaussian setting most of them are already satisfied. For a detailed list of the assumptions, we refer the reader to Appendix A.*

2.2 General Case

The problem of estimating the parameter β_0 without additional assumptions on the distribution of the noise in model (2) is extensively studied in the literature, see i.e. Doksum (1987); Han (1987); Sherman (1993); Abrevaya (1999a,b, 2003); Cavanagh and Sherman (1998); Zhang (2013). Without any restriction on expectation or variance of the noise the function h is not unique, since it can be replaced by location or scale transformations. Thus, to ensure unique identification, we assume that there exists a known

y_0 such that $h(y_0) = 0$ and we scale the last element of β_0 to 1, that is, $\beta_0 = (\theta_0, 1)$.

To simplify the optimization, we employ a method introduced by Lin and Peng (2013) that utilizes the rank-based objective function

$$S(\beta) := \binom{n}{2}^{-1} \sum_{i < j} \left(\mathbf{1}\{Y_j > Y_i\} \Phi(\sqrt{n}(X_j - X_i)^T \beta) + \mathbf{1}\{Y_j \leq Y_i\} \Phi(\sqrt{n}(X_i - X_j)^T \beta) \right).$$

Remark 2.4. Lin and Peng (2013) used the assumption $\|\beta_0\|_2 = 1$ to ensure unique identification, which is equivalent to our assumption $\beta_0 = (\theta_0, 1)$ up to rescaling.

To obtain a sparse solution the authors focused on a penalized version of $S(\beta)$. However, as we are only interested in estimating the residuals and do not necessarily need sparsity, we adapted their analysis for the following simplified non-penalized estimator $\hat{\beta} := (\hat{\theta}, 1)$, where

$$\hat{\theta} := \arg \max_{\theta \in \mathbb{R}^{m-1}} S(\theta, 1),$$

by setting their penalty term to zero.

Under some additional assumptions that mainly ensure smoothness of the distributions of X and ε , Lin and Peng (2013) prove the existence of a local maximizer $\hat{\theta}$ of $S(\theta, 1)$ with

$$\|\hat{\theta} - \theta_0\|_2 = O_P\left(n^{-\frac{1}{2}}\right)$$

and, thus, $\hat{\beta}$ defines a consistent estimator for β_0 . This estimator uses only the rank information without requiring concrete knowledge of h . A detailed list of the additional assumptions that ensure consistency can be found in Appendix A.

In a second step we estimate the function h in order to subsequently obtain an estimate of the noise. We used the method introduced in Chen (2002) based on the rank correlation. To simplify the complex optimization of discrete objective functions, we employ a smoothed version introduced by Zhang (2013).

The smoothed rank correlation objective function is defined by

$$Q(z, y, \hat{\beta}) := \frac{1}{n(n-1)} \sum_{i \neq j} [(d_{jy} - d_{iy_0}) \times \Phi(\sqrt{n}((X_j - X_i)^T \hat{\beta} - z))],$$

where $d_{jy} := \mathbf{1}(Y_j \geq y)$ and $d_{iy_0} := \mathbf{1}(Y_i \geq y_0)$. Then we define an estimator of the function h at y via

$$\hat{h}(y) := \arg \max_{z \in \Omega_h} Q(z, y, \hat{\beta}), \quad (6)$$

where Ω_h is an appropriate compact set.

In Theorem 4.1, Zhang (2013) establish consistency of the proposed estimator for h under a few assumptions, that include \sqrt{n} -consistency of the involved estimator $\hat{\beta}$ and strict monotonicity of the function h , as well as some additional regularity assumptions. A detailed list of the additional assumptions can be found in Appendix A.

Without restricting the optimization space by Ω_h the problem (6) is ill-posed in the sense that $\hat{h}(y) \rightarrow \infty$ for $y = \max\{Y_i\}_{i=1}^n$ and $\hat{h}(y) \rightarrow -\infty$ for $y = \min\{Y_i\}_{i=1}^n$. To circumvent the issue of choosing a proper compact set Ω_h , we added an L_2 regularization term and optimized over \mathbb{R} , that is,

$$\hat{h}(y) := \arg \max_{z \in \mathbb{R}} \{Q(z, y, \hat{\beta}) - \lambda z^2\}.$$

In the experiments we used the regularization parameter $\lambda = 10^{-3}$, which turned out to be small enough to not affect the estimated values significantly and at the same time bounded the objective function for the observed extremes.

3 LEARNING PNL MODELS

By combining the previously introduced rank-based estimators of the functional relations in PNL models we obtain estimates of the stochastic error terms. Using these rank-based estimated error terms, we propose a routine to learn the underlying causal structure by recursively identifying sink nodes via independence testing. Further, we show that our proposed routine consistently recovers a valid causal ordering under identifiability assumptions.

Suppose we observe data in form of n independent copies X_1, \dots, X_n from a random vector $X := (X^{(1)}, \dots, X^{(m)})$. We assume that X follows a PNL causal model, that is, the data generating process is defined by the structural equations

$$X^{(k)} = f^{(k)}\left(g^{(k)}\left(X^{\mathbf{PA}_k}\right) + \varepsilon^{(k)}\right), \quad k = 1, \dots, m,$$

where \mathbf{PA}_k , called the parents of $X^{(k)}$, are a subset of $\{1, \dots, m\} \setminus \{k\}$. The causal perspective stems from viewing those equations as making assignments. Each variable on the left-hand side is assigned the value specified on the right-hand side, given by the value of its parents and a stochastic error term. The causal structure inherent in such structural equations is naturally represented by a directed graph \mathcal{G}^0 , where edges indicate which other variables each variable causally depends upon. As in related work, we assume the corresponding directed graph to be acyclic (DAG). The noise variables $\{\varepsilon^{(k)}\}_{k=1}^m$ are assumed to be mutually independent and $\varepsilon^{(k)} \perp\!\!\!\perp X^{\mathbf{PA}_k}$ for each $k = 1, \dots, m$. The main ansatz for inferring the causal structure is to leverage the independence structure of the stochastic noise $\varepsilon^{(k)}$ and a correctly specified parent set, that is, $\varepsilon^{(k)}$ is independent of all $X^{(j)}$ that precede $X^{(k)}$ in at least one true causal ordering of the underlying graph.

We focus on inferring the causal ordering to reduce the computational burden, however, the framework can easily be adapted to infer the specific causal graph structure by pruning redundant edges. The causal ordering of a graph \mathcal{G}^0 is given by a permutation π of $\{1, \dots, m\}$, such that, if there exists a directed edge from node $\pi(i)$ to node $\pi(j)$ in the graph then $i < j$. We emphasize that the causal ordering for a given graph is not necessarily unique but each causal ordering π corresponds to a unique, fully connected DAG \mathcal{G}^π , where \mathcal{G}^π has a directed edge from node $\pi(i)$ to node $\pi(j)$ if and only if $i < j$. Thus, similar to Bühlmann et al. (2014), we can define the set of true causal orderings Π^0 for any DAG \mathcal{G}^0 as the set of all causal orderings π that correspond to fully connected DAGs \mathcal{G}^π which contain \mathcal{G}^0 as a sub-graph, that is

$$\Pi^0 := \{\pi : \mathcal{G}^\pi \text{ is a super-graph of } \mathcal{G}^0\}.$$

Remark 3.1. *In general Π^0 contains more than one element and all elements correspond to valid causal orderings of the DAG \mathcal{G}^0 (e.g. in the extreme case of an empty graph, all permutations are true causal orderings).*

In order to apply the previously introduced rank-based regression methods, we assume that all functions $f^{(k)}$ in the data generating PNL causal model are continuous and strictly increasing, and thus, we can define their inverse via $h^{(k)} := (f^{(k)})^{-1}$. Further, we assume that all $g^{(k)}$ are linear and the distribution of every stochastic error $\varepsilon^{(k)}$ is assumed to be continuous. We emphasize again, our method is applicable to nonlinear functional relations by means of basis expansions.

Put together, each structural equation, i.e. each cause-effect relation, corresponds to a PNL regression model (1) as introduced in the previous section. That is, the data generating process follows

$$h^{(k)}(X^{(k)}) = \left(X^{(\mathbf{ND}_k)}\right)^T \beta^{(k)} + \varepsilon^{(k)}, \quad k = 1, \dots, m,$$

where the non-descendants \mathbf{ND}_k are given by all nodes that precede node k in at least one true causal ordering and $\varepsilon^{(k)}$ is independent of $X^{(\mathbf{ND}_k)}$. Note that the entries in $\beta^{(k)}$ which do not correspond to parents of node k are simply zero. To leverage the independence $\varepsilon^{(k)} \perp\!\!\!\perp X^{(\mathbf{ND}_k)}$, we define

$$X^{(-k)} := (X^{(1)}, \dots, X^{(k-1)}, X^{(k+1)}, \dots, X^{(m)}).$$

For every sink node k in the graph, we have $\mathbf{ND}_k = \{1, \dots, m\} \setminus \{k\}$, and, thus, for every sink node k the noise $\varepsilon^{(k)}$ is independent of $X^{(-k)}$. Moreover, if node k is not a sink node in the graph, then the noise $\varepsilon^{(k)}$ is not independent of $X^{(-k)}$, since $X^{(-k)}$ contains at least one child of node k . Thus, we can recursively identify a sink node using the HSIC (Gretton et al., 2005) measure of independence between the estimated noise $\hat{\varepsilon}^{(k)}$ and the remaining nodes $X^{(-k)}$.

We propose the following routine to learn one of the valid causal orderings of the graph. First, we utilize the rank-based estimators $\hat{h}^{(k)}$ and $\hat{\beta}^{(k)}$, introduced in the previous section, and estimate the noise via

$$\hat{\varepsilon}_j^{(k)} := \hat{h}^{(k)}(X_j^{(k)}) - \left(X_j^{(-k)}\right)^T \hat{\beta}^{(k)}, \quad j = 1, \dots, n.$$

We repeat this noise estimation for all remaining nodes $k \in \{1, \dots, m\}$ and subsequently calculate the HSIC test statistic between the estimated noises and the observed values of the remaining nodes $X^{(-k)}$, that is

$$t_k := HSIC(\{X_j^{(-k)}, \hat{\varepsilon}_j^{(k)}\}_{j=1}^n), \quad k = 1, \dots, m.$$

We determine the node which leads to the minimal test statistic as a sink node, that is, our proposed sink node estimator is defined by

$$\hat{\pi}(m) := \arg \min_k \{t_k\}.$$

In the next step we remove $\hat{\pi}(m)$ from the set $\{1, \dots, m\}$ and repeat the sink node identification procedure to estimate $\hat{\pi}(m-1)$. Thus, recursively we obtain an estimate for the causal ordering

$$\hat{\pi} = (\hat{\pi}(1), \dots, \hat{\pi}(m)).$$

In the following Theorem we prove consistency of our proposed estimation routine for the causal ordering. It is clear that if at any step our method fails to correctly identify one of the remaining sink nodes, then it fails to estimate a valid causal ordering. Thus, we must require sink node identifiability from the joint distribution in order to ensure consistency of the estimated causal order. The following assumption **(A)** formalizes this intuition.

(A) *For each $k \in [1, m]$ and $A \subset [1, m] \setminus \{k\}$ that contains at least one child of $X^{(k)}$ as well as for all strictly increasing, continuous functions $h : \mathbb{R} \rightarrow \mathbb{R}$ and for all $\beta \in \mathbb{R}^{|A|}$, there exists a constant $\xi > 0$, such that*

$$HSIC\left(\mathbb{P}^{N, X^{(A)}}\right) > \xi,$$

where

$$N := h\left(X^{(k)}\right) - \left(X^{(A)}\right)^T \beta.$$

Theorem 3.1. *Under assumption **(A)** and consistency of the employed estimators $\hat{h}^{(k)}$ and $\hat{\beta}^{(k)}$ we have*

$$\mathbb{P}(\hat{\pi} \in \Pi^0) \rightarrow 1 \text{ as } n \rightarrow \infty.$$

The proof can be found in Appendix D.

Remark 3.2. *Location and scale transformations of the noise variables can be matched by transformations in the functions $h^{(k)}$, however, these transformations do not change the dependence structure and thus without loss of generality, we can assume that the location and scale assumptions in Section 2 are satisfied.*

4 SIMULATIONS

In this section we present the results of a simulation study with the aim to compare the performance of our algorithm to existing causal learning methods on various simulated data sets and validate the consistency results experimentally. If necessary in the specific application, our method can be easily extended to infer the full causal graph structure, however, this vastly increases the computation time, similar to the related existing methods. Since the main differences of the algorithms already come into play during the causal order estimation procedure, we compared the performance on this task alone. Our experiments were designed as follows. We randomly sampled a causal structure with m ($m = 4$ and $m = 7$) nodes from Erdős–Rényi directed acyclic graphs with edge probability $\frac{2}{m-1}$. Thus, the expected total number of edges in the graph is m . We generated data according to the corresponding post-nonlinear model using the following structural relations

$$X^{(k)} = \left(\sum_{j \in \text{PA}_k} \beta_{1j} X^{(j)} + \beta_{2j} (X^{(j)})^2 + \varepsilon_k \right)^{1/3}$$

for $k = 1, \dots, m$, with different noise distributions (standard Normal $\mathcal{N}(0, 1)$, standard Gumbel $\text{Gumbel}(0, 1)$ or standard Logistic $\text{Logistic}(0, 1)$). Further, β_{1j} and β_{2j} are sampled from a uniform distribution with either a low range from -10 to 10 , representing a weak signal setting (low signal-to-noise ratio (SNR)), or a higher range from -100 to 100 , representing a strong signal setting (high SNR).

Remark 4.1. *We highlight that the inner function $g^{(k)}$ is quadratic in the parents of $X^{(k)}$. In the experiments, we used polynomial basis expansions of order two to linearly model the inner function by specifying not only parents but also their squares.*

Using this process, we generated 100 independent data sets and estimated the causal ordering. We compared our results with the classical RESIT method for additive noise models (Peters et al., 2014) and the AbPNL method for post-nonlinear models (Uemura et al., 2022) restricted to the respective causal order estimation parts. The causal ordering of a given DAG is not necessarily unique, thus, as a measure of performance for an estimated causal ordering $\hat{\pi}$ we report the number of directed edges $\hat{\pi}(i) \rightarrow \hat{\pi}(j)$ in the true graph \mathcal{G} with $j < i$, that is

$$\#\{(i, j) : \hat{\pi}(i) \rightarrow \hat{\pi}(j) \in \mathcal{G} \text{ and } j < i\}.$$

This measure equals zero when $\hat{\pi}$ is a valid causal ordering and achieves its maximum, the number of edges in \mathcal{G} , when $\hat{\pi}$ is a reversed causal ordering.

Figure 1 shows the performance of the Gaussian method introduced in Subsection 2.1, named RankG, compared to RESIT and AbPNL on 4- and 7-dimensional causal graphs in settings with Gaussian, Gumbel (standard extreme value

distribution) and Logistic noise. Dashed lines indicate the weak signal setting and solid lines depict the strong signal setting. We plot the mean of our performance measure, the number of wrongly oriented edges in the fully connected DAG corresponding to the estimated causal ordering, over all 100 data sets against the sample size.

Our proposed RankG method outperforms the competition in almost all considered settings, especially in the low SNR setting. We emphasize that it might seem counterintuitive that the RankG method performs better in a low SNR setting than in a high SNR setting, however, the noise drives the identification in the rank-based learning procedure in PNL models. Thus, higher noise in comparison to the signal strengths induces more changes in the ranks that propagate through the graph, and thus, better performance of the rank-based estimation methods.

Furthermore, in the low SNR setting our computational results support the theoretical consistency results and our method seems to recover a valid causal ordering even in moderate sample sizes.

The results in Figure 1 display that even under noise misspecification, that is, in the Gumbel and Logistic noise cases, the RankG method performs best. This might indicate some robustness of our proposed method for causal order estimation, even though we do not recover the true noise under misspecification (see Figure 6).

We conducted similar experiments to compare the performance of our proposed general method introduced in Subsection 2.2, named RankS, however with lower sample sizes for computational reasons. Figure 2 shows the results of the different competitors RankS, RankG, RESIT and AbPNL for 4-dimensional graphs with Gaussian, Gumbel and Logistic noise. Our proposed method RankS performs best in all considered sample sizes and all noise cases, except for the weak signal settings where RankG performs better. This might stem from the fact that the pairwise rank likelihood used in RankG better approximates the marginal rank likelihood.

In additional experiments, we investigated the behavior of the introduced methods for more complex functional relations g , namely a polynomial of degree 4. Similar to the experiments before, we sampled data from Erdős–Rényi DAGs, however, with the structural relations

$$X^{(k)} = \left(\sum_{j \in \text{PA}_k} \beta_{1j} X^{(j)} + \beta_{2j} (X^{(j)})^2 + \beta_{3j} (X^{(j)})^3 + \beta_{4j} (X^{(j)})^4 + \varepsilon_k \right)^{1/3}$$

for $k = 1, \dots, m$, with different noise distributions and $\beta_{1j}, \beta_{2j}, \beta_{3j}$ and β_{4j} sampled with low or high range. Figure 3 shows the resulting mean performance measures for the different methods RankS, RankG, RESIT and AbPNL over 100 data sets. As expected, the task becomes more

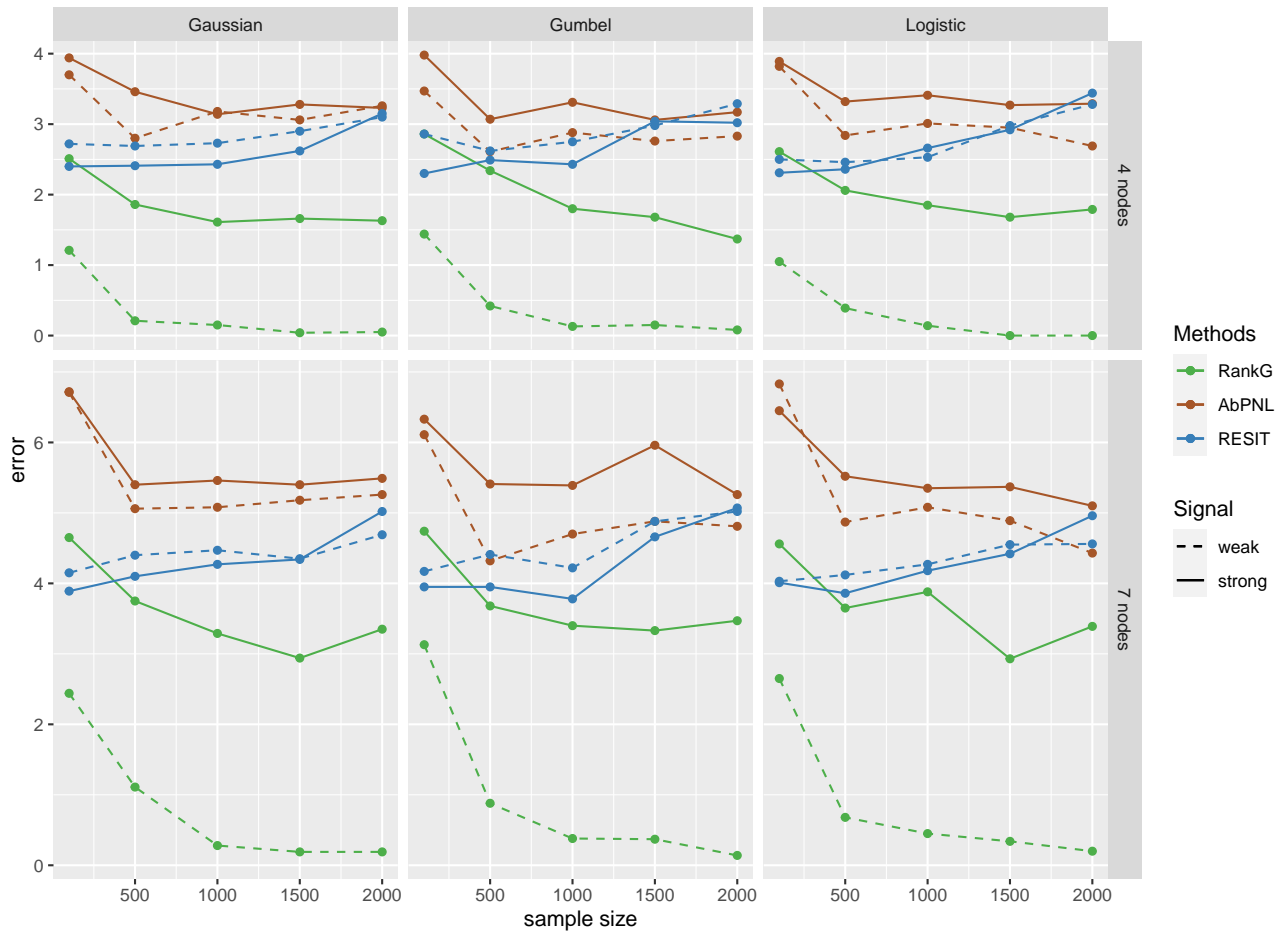


Figure 1: Performance of RankG, AbPNL and RESIT causal order estimation methods in different dimensions (4,7) against sample size.

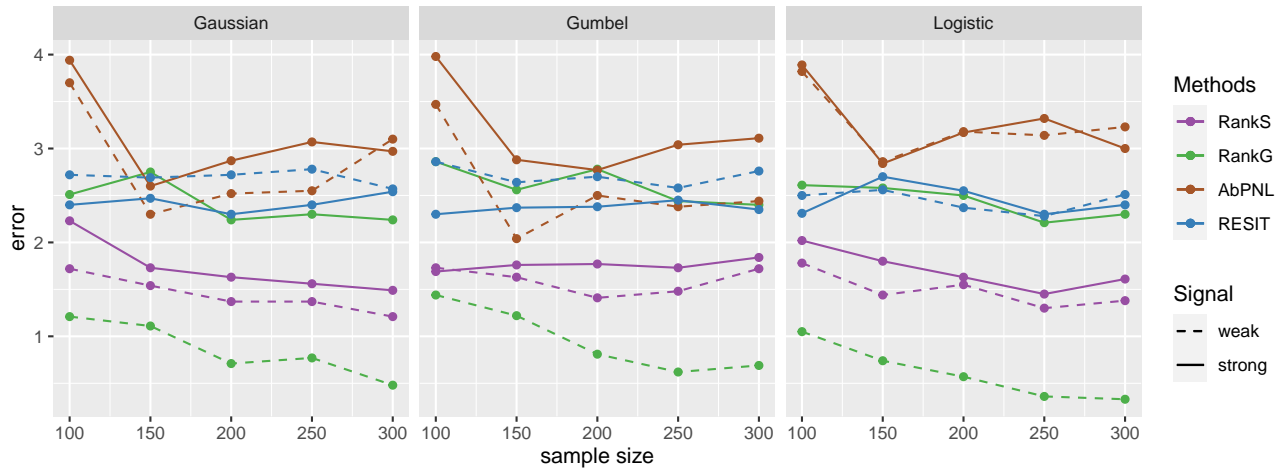


Figure 2: Performance of RankS, RankG, AbPNL and RESIT causal order estimation methods in dimension 4 against sample size.

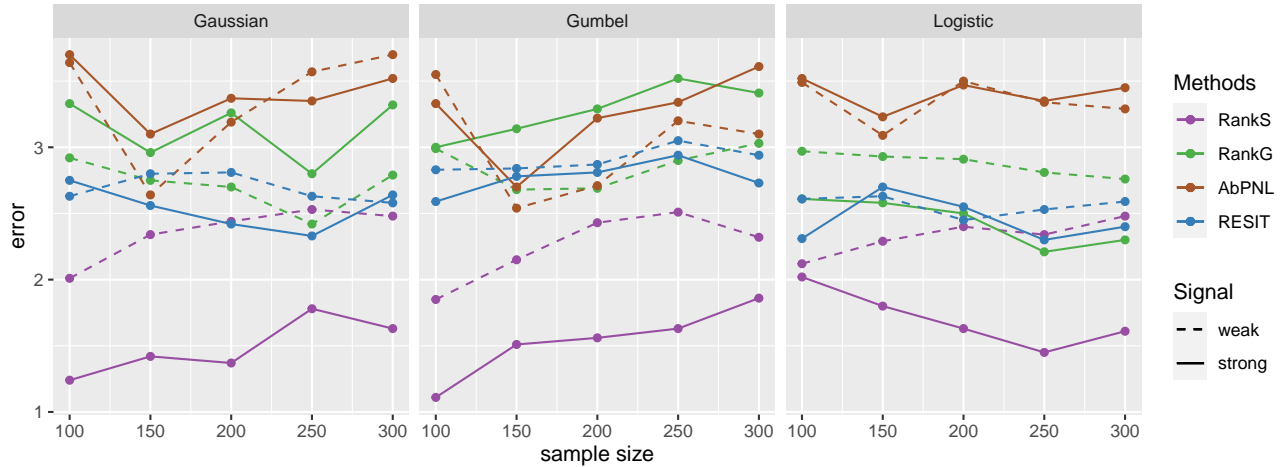


Figure 3: Performance of RankS, RankG, AbPNL and RESIT causal order estimation methods in dimension 4 against sample size, where function g is 4 degree polynomial.

challenging by increasing the complexity of the functional parameters, since it is difficult to estimate the functional relations in the first place. However, even in the considered low sample sizes, our proposed RankS method seems to detect some causal structure and outperform the competition.

Further, we analysed the behaviour of the used functional estimators introduced in Section 2 by performing the following experiments. We generated 100 independent data sets of sample size 500 according to model (2) with a 3-dimensional predictor X , standard Gaussian or Gumbel noise, cubic function $h(y) = y^3$ and linear parameters $\beta_0 = (10, 5, 1)$. Then we estimated the functional parameter β_0 and the function h with the introduced rank-based methods.

Figure 4 shows the estimation of the function h for one representative result across the 100 replications. The red lines indicate the true value of the function h , while the black dots indicate the pointwise estimates. We notice that the estimation of the function h with the rank-based method that relies on the normality assumption (used in RankG) fails to correctly estimate the functional relation at extremes under Gumbel noise. This is due to the misspecified tail probability structure. In contrast, the general estimation method employed in RankS is not influenced by the specific underlying noise distribution. However, in the Gaussian noise case we notice small estimation bias, which can be regulated with the hyperparameter λ in the estimation procedure. Further, from the results across all 100 data sets, we noticed that the variance of the functional estimate across the data sets is higher using the general estimation methods in RankS.

Figure 5 shows the estimation of the first two entries of β_0 .

Recall that in the general RankS method we fixed the last entry in our estimation of β_0 . The box plots show the estimated values of β_0 across the 100 data sets and red dots indicate the true values. We see again that RankG estimates the parameter β_0 with a bias in the misspecified Gumbel noise setting, similar to the estimation of the function h . However, in the Gaussian setting the RankG method estimation of β_0 has a lower variance than RankS and in all other cases the median estimate corresponds to the true value.

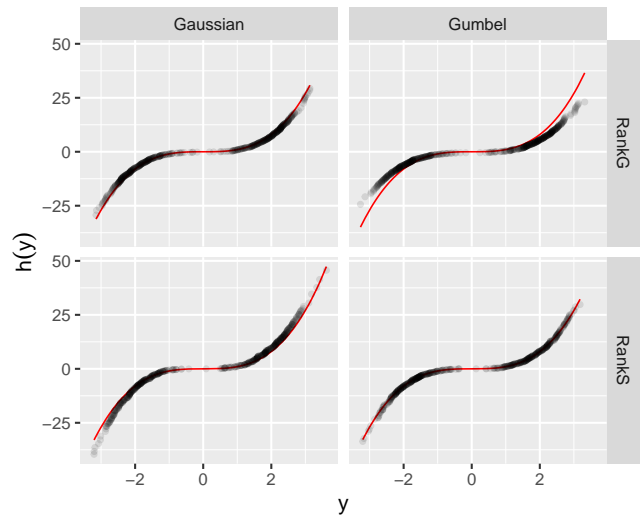


Figure 4: Estimation of the function h .

Figure 6 shows the estimated noise by combining both estimation results. We plot the estimated noise against the true values for one representative data set (similar to Fig-

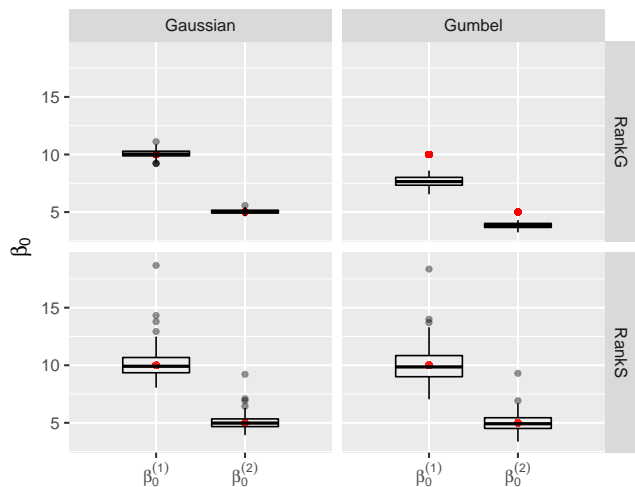


Figure 5: Estimation of β_0 .

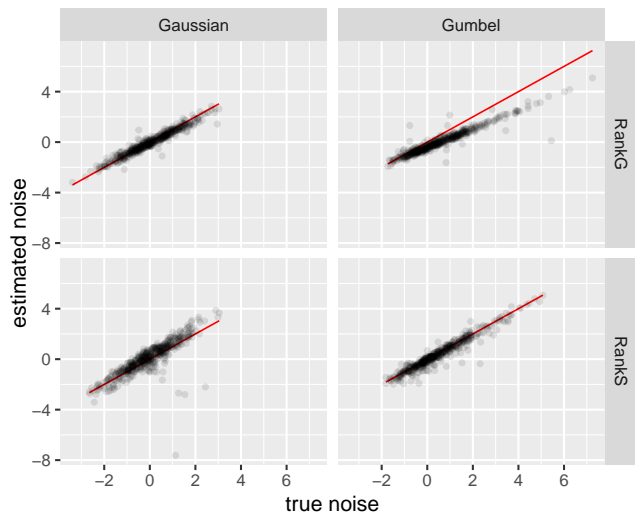


Figure 6: Estimation of the noise ε .

ure 4). Red lines correspond to a perfect estimation. The RankG method inherits the behaviour from both estimation parts and fails to correctly estimate the extreme noise cases. However, it outperforms the RankS method in the Gaussian setting.

5 CONCLUSION

We proposed a new routine for causal discovery in multivariate post-nonlinear structural equation models. Our method disentangles the two tasks of estimating the functional relations and learning the causal structure by employing rank-based methods for the first task. Thus, our

proposed routine is less susceptible to overfitting issues exhibited by the existing methods that rely on minimizing dependence and subsequently testing for independence.

We introduced PNL rank regression methods to learn the functional relations in PNL models and subsequently estimate the residuals. As a special case, we first considered Gaussian noise and used pairwise rank likelihoods in a computationally fast algorithm, whereas, for the general unknown noise case, we employed a smoothed version of rank correlations to obtain estimates of the functional relations. While our presentation focused on linearity in the inner functional relation to simplify the theoretical analysis, the framework includes nonlinear relations by means of basis expansions. Employing the introduced estimators of the functional relations, we proposed a causal learning routine to recursively identify sink nodes based on independence tests with the estimated residuals. Further, we prove consistent causal order recovery of our proposed routine under identifiability assumptions and consistency of the employed functional estimators.

We validated our theoretical findings in a simulation study that showed that our proposed routine outperforms the competition and is able to recover a valid causal ordering even in moderate sample sizes.

Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No. 83818), the German Federal Ministry of Education and Research, and the Bavarian State Ministry for Science and the Arts. The authors of this work take full responsibility for its content.

References

Abrevaya, J. (1999a). Computation of the maximum rank correlation estimator. *Economics Letters*, 62(3):279–285.

Abrevaya, J. (1999b). Leapfrog estimation of a fixed-effects model with unknown transformation of the dependent variable. *Journal of Econometrics*, 93(2):203–228.

Abrevaya, J. (2003). Pairwise-difference rank estimation of the transformation model. *Journal of Business & Economic Statistics*, 21(3):437–447.

Bühlmann, P., Peters, J., and Ernest, J. (2014). CAM: Causal additive models, high-dimensional order search and penalized regression. *The Annals of Statistics*, 42(6):2526 – 2556.

Cavanagh, C. L. and Sherman, R. P. (1998). Rank estimators for monotonic index models. *Journal of Econometrics*, 84:351–381.

- Chen, S. (2002). Rank estimation of transformation models. *Econometrica*, 70(4):1683–1697.
- Cuzick, J. (1988). Rank regression. *The Annals of Statistics*, 16(4):1369–1389.
- Doksum, K. A. (1987). An Extension of Partial Likelihood Methods for Proportional Hazard Models to General Transformation Models. *The Annals of Statistics*, 15(1):325 – 345.
- Drton, M., Foygel, R., and Sullivant, S. (2011). Global identifiability of linear structural equation models. *The Annals of Statistics*, 39(2):865 – 886.
- Glymour, C., Zhang, K., , and Spirtes, P. (2019). Review of causal discovery methods based on graphical models. *Front. Genet.*
- Gretton, A., Bousquet, O., Smola, A., and Schölkopf, B. (2005). Measuring statistical dependence with hilbertschmidt norms. In *Proceedings of the 16th International Conference on Algorithmic Learning Theory*, ALT’05, page 63–77, Berlin, Heidelberg. Springer-Verlag.
- Han, A. K. (1987). Non-parametric analysis of a generalized regression model: The maximum rank correlation estimator. *Journal of Econometrics*, 35(2):303–316.
- Hoyer, P., Janzing, D., Mooij, J. M., Peters, J., and Schölkopf, B. (2008). Nonlinear causal discovery with additive noise models. In *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc.
- Lin, H. and Peng, H. (2013). Smoothed rank correlation of the linear transformation regression model. *Computational Statistics & Data Analysis*, 57(1):615–630.
- Moneta, A., Entner, D., Hoyer, P. O., and Coad, A. (2013). Causal Inference by Independent Component Analysis: Theory and Applications. *Oxford Bulletin of Economics and Statistics*, 75(5):705–730.
- Okamoto, M. (1973). Distinctness of the eigenvalues of a quadratic form in a multivariate sample. *The Annals of Statistics*, 1(4):763–765.
- Opgen-Rhein, R. and Strimmer, K. (2007). From correlation to causation networks: a simple approximate learning algorithm and its application to high-dimensional plant gene expression data. *BMC Systems Biology*.
- Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. (2014). Causal discovery with continuous additive noise models. *Journal of Machine Learning Research*, 15(58):2009–2053.
- Serfling, R. J. (1980). *Approximation Theorems of Mathematical Statistics*. John Wiley and Sons.
- Sherman, R. P. (1993). The limiting distribution of the maximum rank correlation estimator. *Econometrica*, 61(1):123–137.
- Spirtes, P., Glymour, C., and Scheines, R. (2000). *Causation, Prediction, and Search*. MIT press, 2nd edition.
- Spirtes, P. and Zhang, K. (2016). Causal discovery and inference: concepts and recent methodological advances. *Applied Informatics*.
- Teran Hidalgo, S. J., Wu, M. C., Engel, S. M., and Kosorok, M. R. (2018). Goodness-of-fit test for nonparametric regression models: Smoothing spline anova models as example. *Computational Statistics & Data Analysis*, 122:135–155.
- Tu, R., Zhang, K., Kjellstrom, H., and Zhang, C. (2022). Optimal transport for causal discovery. In *International Conference on Learning Representations*.
- Uemura, K. and Shimizu, S. (2020). Estimation of post-nonlinear causal models using autoencoding structure. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3312–3316.
- Uemura, K., Takagi, T., Takayuki, K., Yoshida, H., and Shimizu, S. (2022). A multivariate causal discovery based on post-nonlinear model. In *First Conference on Causal Learning and Reasoning*.
- Yu, T., Li, P., Chen, B., Yuan, A., and Qin, J. (2021). Maximum pairwise-rank-likelihood-based inference for the semiparametric transformation model. *arXiv e-prints*, page arXiv:2103.13435.
- Zhang, J. (2013). Estimation and testing methods for monotone transformation models. PhD thesis, Columbia University.
- Zhang, K. and Hyvärinen, A. (2009). On the identifiability of the post-nonlinear causal model. *UAI’09*, page 647–655.

A ADDITIONAL ASSUMPTIONS

In this section, we list all additional assumptions that are mentioned in the main paper in detail.

Cuzick (1988) employs the following 4 (additional) assumptions to ensure the consistency of \hat{h}_G defined in (5). Thus, the following assumptions together with the modelling assumptions and assumption **(A)** in the main paper ensure the consistency of the proposed RankG method.

AG1 Let $G_n(x) := \frac{1}{n} \sum_{j=1}^n \mathbb{1}\{X_j \leq x\}$. Assume Z_n has distribution G_n , then $|Z_n|^t$ is uniformly integrable for some t which is specified in the next assumption (at least $t > 4$).

AG2 For $F_\beta(z)$ defined in (4), there exists finite K such that

$$F_\beta^{-1}(u) + u(1-u)(F_\beta^{-1}(u))' \leq \frac{K}{((u(1-u)))^\alpha}, \quad \forall u \in (0, 1),$$

for all n and $\beta \in B$, where B is a neighborhood of β_0 and $\alpha + t^{-1} < \frac{1}{2}$.

AG3 The function $Z(\beta) := \mathbb{E}[X|X^T(\beta - \beta_0) = 0]$ is L_2 continuous as $\beta \rightarrow \beta_0$.

AG4 The following inequality holds

$$\mathbb{E}[X(X - Z(\beta_0))] > 0.$$

The main assumptions above essentially correspond to moment conditions on the distribution of X .

Following Lin and Peng (2013) **(AS1-AS4)** and Zhang (2013) **(AS5-AS9)**, we list the additional assumptions that are required to ensure consistency of the estimators in Section 2.2. Thus, in combination with the modelling assumptions and assumption **(A)** in the main paper, the RankS method is consistent.

AS1 Let g be the density function of $(X_j - X_i)^T \beta_0$ and F the distribution function of $\varepsilon_j - \varepsilon_i$. Define the functions $\Gamma(s) := \mathbb{E}[(X_j - X_i)^T | (X_j - X_i)_0^\beta = s]$ and $\Omega(s) := F(s)\Gamma(s)g(s)$. Then $\Omega'(0)$ is nonsingular.

AS2 The density g is positive with a continuous second derivative on its corresponding compact support.

AS3 The function F has a continuous second derivative on its corresponding support.

AS4 The random variable X is bounded with a compact support.

AS5 The true parameter θ_0 is an interior point of a compact subset $\Theta \subset \mathbb{R}^{m-1}$.

AS6 The support of X is not contained in a linear subspace of \mathbb{R}^m . Moreover, conditional on the first $m - 1$ components of X , the last component of X has a density function with respect to the Lebesgue measure.

AS7 Define

$$\tau(y, x, \theta) := \mathbb{E}[\mathbb{1}\{y > Y\} \mathbb{1}\{(x - X)^T(\theta^T, 1)^T > 0\} + \mathbb{1}\{y < Y\} \mathbb{1}\{(x - X)^T(\theta^T, 1)^T < 0\}]$$

and let

$$|\nabla_m| \tau(y, x, \theta) = \sum_{i_1, \dots, i_m} \left| \frac{\partial^m \tau(y, x, \theta)}{\partial \theta_{i_1} \dots \partial \theta_{i_m}} \right|.$$

There exists a neighborhood \mathcal{N} of θ_0 such that for each pair of (y, x) in the support of (Y, X) the following hold

- The second derivatives of $\tau(y, x, \theta)$ with respect to θ exist in \mathcal{N} .
- There exists an integrable function $M(y, x)$ such that for all θ in \mathcal{N}

$$\|\nabla_2 \tau(y, x, \theta) - \nabla_2 \tau(y, x, \theta_0)\| \leq M(y, x) |\theta - \theta_0|.$$

- $\mathbb{E}[(|\nabla_1| \tau(Y, X, \theta_0))^2] < \infty$.
- $\mathbb{E}[|\nabla_2| \tau(Y, X, \theta_0)] < \infty$.
- The matrix $\mathbb{E}[\nabla_2 \tau(Y, X, \theta_0)]$ is strictly negative definite.

AS8 There exists $\epsilon^* > 0$ and y_1, y_2 in the support of Y such that $[h(y_1 - \epsilon^*), h(y_2 + \epsilon^*)]$ is contained in a compact interval.

AS9 For $\omega_1 = (x^1, y^1), \omega_2 = (x^2, y^2)$ and $W = (X, Y)$ we define

$$f^z(\omega_1, \omega_2, z, y, \beta) := (\mathbb{1}\{y^1 \geq y\} - \mathbb{1}\{y^2 \geq y\})\mathbb{1}\{(x^1 - x^2)^T \beta \geq z\}.$$

Further, we define

$$\tau(\omega, z, y, \beta) := \mathbb{E}[f^z(\omega, W, z, y, \beta) + f^z(W, \omega, z, y, \beta)],$$

then

$$V^z(y) := \mathbb{E}\left[\frac{\partial^2 \tau(W, h(y), y, \beta_0)}{\partial z^2}\right]$$

is negative for each $y \in [y_1, y_2]$ and uniformly bounded away from 0.

Zhang (2013) show uniform consistency on the interval $[y_1, y_2]$ defined in the assumptions **AS8-AS9**.

B PROOF OF PROPOSITION 2.1

To prove Proposition 2.1 we employ the following two Lemmas.

Lemma B.1. For all $z \in \mathbb{R}$ we have

$$\phi'(z)\Phi(z) - (\phi(z))^2 < 0,$$

where Φ and ϕ denote the CDF and PDF of the standard normal distribution.

Proof. With $h(z) := \phi'(z)\Phi(z) - (\phi(z))^2$, we show that $h(z) < 0$. Substituting the derivative of ϕ we have

$$h(z) = -z\phi(z)\Phi(z) - (\phi(z))^2 = \phi(z)(-z\Phi(z) - \phi(z)).$$

Since $\phi(z) > 0$ for all $z \in \mathbb{R}$ it remains to show that $g(z) := -z\Phi(z) - \phi(z) < 0$. For $z \geq 0$ this is clear. Thus we consider the case $z < 0$. We have for the derivative of g

$$g'(z) = -\Phi(z) - z\phi(z) + z\phi(z) = -\Phi(z) < 0.$$

Therefore, g is a strictly decreasing function. The limit of $g(z)$ for $z \rightarrow -\infty$ is given by

$$\begin{aligned} \lim_{z \rightarrow -\infty} g(z) &= \lim_{z \rightarrow -\infty} (-z\Phi(z) - \phi(z)) = - \lim_{z \rightarrow -\infty} \frac{\Phi(z)}{\frac{1}{z}} \\ &= \lim_{z \rightarrow -\infty} \frac{\phi(z)}{\frac{1}{z^2}} = \lim_{z \rightarrow -\infty} \frac{1}{\sqrt{2\pi}} \frac{z^2}{e^{z^2/2}} = 0, \end{aligned}$$

and the claim follows. \square

Lemma B.2. The function

$$f(x) := \log \Phi(c^T x), \quad x \in \mathbb{R}^m$$

is concave, where Φ is the CDF of the standard normal distribution and $c \in \mathbb{R}^m$ is a nonzero constant. Moreover, $v^T \nabla^2 f(x) v = 0$ for a vector $v \in \mathbb{R}^m$ and Hessian matrix $\nabla^2 f(x)$ if and only if $v^T c = 0$.

Proof. The function f is twice differentiable, and thus for the first part of the Lemma it is enough to show that the Hessian of $f(x)$ is negative semi-definite. The gradient of f is given by

$$\nabla f(x) = \frac{\phi(c^T x)}{\Phi(c^T x)} c,$$

where ϕ is the PDF of the standard normal distribution. Thus, the Hessian is

$$\nabla^2 f(x) = \frac{\phi'(c^T x)\Phi(c^T x) - (\phi(c^T x))^2}{(\Phi(c^T x))^2} \cdot cc^T.$$

So, for any $v \in \mathbb{R}^m$ we have

$$\begin{aligned} v^T \nabla^2 f(x) v &= v^T \frac{\phi'(c^T x) \Phi(c^T x) - (\phi(c^T x))^2}{(\Phi(c^T x))^2} \cdot c c^T v \\ &= \frac{\phi'(c^T x) \Phi(c^T x) - (\phi(c^T x))^2}{(\Phi(c^T x))^2} \cdot v^T c c^T v \\ &= \frac{\phi'(c^T x) \Phi(c^T x) - (\phi(c^T x))^2}{(\Phi(c^T x))^2} \cdot (v^T c)^2 \leq 0, \end{aligned}$$

where the last step follows from Lemma B.1 and the fact that $(v^T c)^2 \geq 0$.

Moreover, $v^T \nabla^2 f(x) v = 0$ if and only if $v^T c = 0$, which completes the proof. \square

Employing the Lemmas we can prove Proposition 2.1.

Proof. From (3) we have

$$\begin{aligned} \ell_{prl}(\beta) &= \binom{n}{2}^{-1} \sum_{i < j} \mathbf{1}(Y_j > Y_i) \log \Phi \left(\frac{(X_j - X_i)^T \beta}{\sqrt{2}} \right) \\ &\quad + \mathbf{1}(Y_j \leq Y_i) \log \Phi \left(\frac{(X_i - X_j)^T \beta}{\sqrt{2}} \right). \end{aligned}$$

Thus, with Lemma B.2 we know that $\ell_{prl}(\beta)$ is a sum of concave functions. Since sums preserve the concavity $\ell_{prl}(\beta)$ is concave.

We show strict concavity by contradiction and thus assume that $\ell_{prl}(\beta)$ is not strictly concave. This implies that there exists a vector v such that $v^T \nabla^2 \ell_{prl}(\beta) v = 0$ for the Hessian matrix $\nabla^2 \ell_{prl}(\beta)$ of $\ell_{prl}(\beta)$. The Hessian operator is linear, thus, $\nabla^2 \ell_{prl}(\beta)$ is a sum of Hessians, that is

$$\begin{aligned} \nabla^2 \ell_{prl}(\beta) &= \binom{n}{2}^{-1} \sum_{i < j} \mathbf{1}(Y_j > Y_i) \nabla^2 \log \Phi \left(\frac{(X_j - X_i)^T \beta}{\sqrt{2}} \right) \\ &\quad + \mathbf{1}(Y_j \leq Y_i) \nabla^2 \log \Phi \left(\frac{(X_i - X_j)^T \beta}{\sqrt{2}} \right). \end{aligned}$$

Lemma B.2 gives that $v^T \nabla^2 \log \Phi \left(\frac{(X_j - X_i)^T \beta}{\sqrt{2}} \right) v = 0$ if and only if $v^T (X_j - X_i) = 0$. Since for each i, j , either one of $\mathbf{1}(Y_j > Y_i)$ or $\mathbf{1}(Y_j \leq Y_i)$ is 1, $v^T \nabla^2 \ell_{prl}(\beta) v = 0$ implies that $v^T (X_j - X_i) = 0$ for all i and j . Therefore, $X_j^T v = c$ for a constant $c \in \mathbb{R}$ for all j . Let $X^{(i)} = (X_1^{(i)}, \dots, X_n^{(i)})$ be the sample vector of the i -th component of X and define the matrix $\mathbf{X} := [\mathbf{1}, X^{(1)}, \dots, X^{(m)}] \in \mathbb{R}^{n \times m+1}$. Then \mathbf{X} does not have full column rank, i.e. taking $u = (-c, v^T)^T$ implies $\mathbf{X}u = \mathbf{0}$.

However, if we take any arbitrary square sub-matrix in \mathbf{X} and compute the determinant, we obtain a non-zero polynomial of some $X^{(1)}, \dots, X^{(m)}$. The Lemma in (Okamoto, 1973) states that such a polynomial is zero only on the Lebesgue measure zero. Therefore, the rank of \mathbf{X} is $\min\{n, m+1\} = m+1$, which contradicts the equality $\mathbf{X}u = \mathbf{0}$ and, thus, completes the proof. \square

C PROOF OF THEOREM 2.1

To keep the formulas readable, we define $U_{ij} := \frac{X_i - X_j}{\sqrt{2}}$. This gives

$$\ell_{prl}(\beta) = \binom{n}{2}^{-1} \sum_{i < j} \mathbf{1}(Y_j > Y_i) \log \Phi (U_{ji}^T \beta) + \mathbf{1}(Y_j \leq Y_i) \log \Phi (U_{ij}^T \beta).$$

For the proof, we use the Taylor expansion of $\ell_{prl}(\hat{\beta}_{PRL})$ around β_0 and use properties of the gradient of $\ell_{prl}(\beta)$ at β_0 , which are established in the following Lemmas.

Lemma C.1. *The gradient $\nabla_{\beta} \ell_{prl}(\beta_0)$ converges to zero almost surely, that is*

$$\nabla_{\beta} \ell_{prl}(\beta_0) \xrightarrow{a.s.} 0.$$

Proof. The gradient

$$\nabla_{\beta} \ell_{prl}(\beta_0) = \binom{n}{2}^{-1} \sum_{i < j} \mathbf{1}(Y_j > Y_i) \frac{\phi(U_{ji}^T \beta_0)}{\Phi(U_{ji}^T \beta_0)} \cdot U_{ji} + \mathbf{1}(Y_j \leq Y_i) \frac{\phi(U_{ij}^T \beta_0)}{\Phi(U_{ij}^T \beta_0)} \cdot U_{ij},$$

is a U-statistic with kernel

$$\psi((X_1, Y_1), (X_2, Y_2)) := \mathbf{1}(Y_2 > Y_1) \frac{\phi(U_{21}^T \beta_0)}{\Phi(U_{21}^T \beta_0)} \cdot U_{21} + \mathbf{1}(Y_2 \leq Y_1) \frac{\phi(U_{12}^T \beta_0)}{\Phi(U_{12}^T \beta_0)} \cdot U_{12}.$$

Note that $\psi((X_1, Y_1), (X_2, Y_2))$ is symmetric as there are no ties in Y_i 's (Y has a continuous distribution).

We use the generalization of the classical Strong Law of Large Numbers (SLLN) for U-statistics (e.g. Theorem A in (Serfling, 1980) Section 5.4). The expectation of the kernel ψ is

$$\begin{aligned} & \mathbb{E}[\psi((X_1, Y_1), (X_2, Y_2))] \\ &= \mathbb{E} \left[\mathbf{1}(Y_2 > Y_1) \frac{\phi(U_{21}^T \beta_0)}{\Phi(U_{21}^T \beta_0)} \cdot U_{21} + \mathbf{1}(Y_2 \leq Y_1) \frac{\phi(U_{12}^T \beta_0)}{\Phi(U_{12}^T \beta_0)} \cdot U_{12} \right] \\ &= \mathbb{E}_X \left[\mathbb{E}_Y [\mathbf{1}(Y_j > Y_i) | X] \frac{\phi(U_{21}^T \beta_0)}{\Phi(U_{21}^T \beta_0)} \cdot U_{21} + \mathbb{E}_Y [\mathbf{1}(Y_j \leq Y_i) | X] \frac{\phi(U_{12}^T \beta_0)}{\Phi(U_{12}^T \beta_0)} \cdot U_{12} \right] \\ &= \mathbb{E}_X \left[\mathbb{P}(Y_j > Y_i | X) \frac{\phi(U_{21}^T \beta_0)}{\Phi(U_{21}^T \beta_0)} \cdot U_{21} + \mathbb{P}(Y_j \leq Y_i | X) \frac{\phi(U_{12}^T \beta_0)}{\Phi(U_{12}^T \beta_0)} \cdot U_{12} \right] \\ &= \mathbb{E}_X \left[\Phi(U_{21}^T \beta_0) \frac{\phi(U_{21}^T \beta_0)}{\Phi(U_{21}^T \beta_0)} \cdot U_{21} + \Phi(U_{12}^T \beta_0) \frac{\phi(U_{12}^T \beta_0)}{\Phi(U_{12}^T \beta_0)} \cdot U_{12} \right] \\ &= \mathbb{E}_X [\phi(U_{21}^T \beta_0) \cdot U_{21} + \phi(U_{12}^T \beta_0) \cdot U_{12}] = \mathbb{E}_X [\phi(U_{21}^T \beta_0) \cdot [U_{21} + U_{12}]] \\ &= \mathbb{E}_X \left[\phi(U_{21}^T \beta_0) \cdot \left[\frac{X_j - X_i}{\sqrt{2}} + \frac{X_i - X_j}{\sqrt{2}} \right] \right] = \mathbb{E}_X [0] = 0, \end{aligned}$$

where the first equalities follow from the linearity of the expectation and the tower rule of the expectation, i.e. $\mathbb{E}[Q(X, Y)] = \mathbb{E}[\mathbb{E}[Q(X, Y) | X]]$ for any function Q . The third equality is a classical property of the indicator function. The fourth equality uses the monotonicity of the function h . The fifth step is just a cancellation of the equal members in the fractions. Finally, the sixth equality follows from the fact that $\phi(x) = \phi(-x)$ for the probability density function ϕ of the standard normal distribution.

We show that the absolute value of the kernel ψ has a finite expectation, since

$$\begin{aligned} & \mathbb{E}[|\psi((X_1, Y_1), (X_2, Y_2))|] \\ & \leq \mathbb{E} \left[\mathbf{1}(Y_2 > Y_1) \frac{\phi(U_{21}^T \beta_0)}{\Phi(U_{21}^T \beta_0)} \cdot |U_{21}| + \mathbf{1}(Y_2 \leq Y_1) \frac{\phi(U_{12}^T \beta_0)}{\Phi(U_{12}^T \beta_0)} \cdot |U_{12}| \right] \\ & = \mathbb{E}_X \left[\Phi(U_{21}^T \beta_0) \frac{\phi(U_{21}^T \beta_0)}{\Phi(U_{21}^T \beta_0)} \cdot |U_{21}| + \Phi(U_{12}^T \beta_0) \frac{\phi(U_{12}^T \beta_0)}{\Phi(U_{12}^T \beta_0)} \cdot |U_{12}| \right] \\ & = \phi(U_{12}^T \beta_0) \cdot \frac{2 \cdot |X_i - X_j|}{\sqrt{2}} < \infty, \end{aligned}$$

where we used the triangle inequality for the absolute value in the first step and the other steps are similar to the calculation for the expectation of the kernel ψ . The last quantity is finite since the probability density function of the standard normal distribution is bounded.

Thus, the SLLN yields

$$\nabla_{\beta} \ell_{prl}(\beta_0) \xrightarrow{a.s.} 0,$$

which completes the proof of the Lemma. \square

Lemma C.2. Let $a > 0$. For all $\beta \in \mathbb{R}^m$, such that $\|\beta - \beta_0\|_2 = a$, and $n > m$ we have

$$\mathbb{P}(\ell_{prl}(\beta) < \ell_{prl}(\beta_0)) \rightarrow 1.$$

Proof. The Taylor expansion of $\ell_{prl}(\beta)$ around β_0 gives

$$\ell_{prl}(\beta) - \ell_{prl}(\beta_0) = (\beta - \beta_0)^T \nabla_{\beta} \ell_{prl}(\beta_0) + \frac{1}{2} (\beta - \beta_0)^T \nabla_{\beta}^2 \ell_{prl}(\beta^*) (\beta - \beta_0),$$

where $\|\beta^* - \beta_0\|_2 \leq \|\beta - \beta_0\|_2 = a$, i.e., β^* is in the closed ball around β_0 with radius a . Clearly, $\nabla_{\beta}^2 \ell_{prl}(\beta^*)$ is continuous with respect to β^* . Moreover, from Proposition 2.1 we know that the maximum eigenvalue of $\nabla_{\beta}^2 \ell_{prl}$ is negative. Thus, there exist a maximum eigenvalue $\lambda_{max} < 0$ of $\nabla_{\beta}^2 \ell_{prl}$ within the closed ball with center β_0 and radius a .

Therefore, we have

$$\frac{1}{2} (\beta - \beta_0)^T \nabla_{\beta}^2 \ell_{prl}(\beta^*) (\beta - \beta_0) \leq \frac{\lambda_{max}}{2} \|\beta - \beta_0\|_2^2.$$

Using the previous Lemma C.1 the remaining term $(\beta - \beta_0)^T \nabla_{\beta} \ell_{prl}(\beta_0)$ asymptotically vanishes in probability and thus the claim follows. \square

In the following we prove Theorem 2.1., i.e. $\hat{\beta}_{prl} - \beta_0 = o_P(1)$.

Proof. From Lemma C.2 we know that for any fixed $a > 0$, there exists a maximum of $\ell_{prl}(\beta)$ within the closed ball around β_0 with radius a with probability tends to 1. However, $\hat{\beta}_{prl}$ maximizes $\ell_{prl}(\beta)$, and thus

$$\mathbb{P}\left(\left\|\hat{\beta}_{prl} - \beta_0\right\|_2 < a\right) \rightarrow 1 \text{ as } n \rightarrow \infty,$$

which completes the proof. \square

D PROOF OF THEOREM 3.1

To prove the Theorem we employ the following Lemma.

Lemma D.1. Under the assumption (A) we have

- If node k is not a sink node, then

$$t_k > \xi + o_P(1).$$

- If node k is a sink node, then

$$t_k \rightarrow 0 \text{ as } n \rightarrow \infty.$$

Proof. First, assume that k is not a sink node. Denoting $\hat{\varepsilon} := \hat{h}(X^{(k)}) - (X^{(-k)})^T \hat{\beta}^{(k)}$, assumption (A) gives that $HSIC(\mathbb{P}^{X^{(-k)}, \hat{\varepsilon}}) > \xi$. On the other hand, Theorem 3 in Gretton et al. (2005) gives that for all $\delta > 0$, with probability at least $1 - \delta$ we have

$$|HSIC(\{X_j^{(-k)}, \hat{\varepsilon}_j^{(k)}\}_{j=1}^n) - HSIC(\mathbb{P}^{X^{(-k)}, \hat{\varepsilon}})| \leq \sqrt{\frac{\log 6/\delta}{\alpha^2 n}} + \frac{C}{n},$$

where α and C are constants. Thus,

$$\begin{aligned} t_k - \xi &= HSIC(\{X_j^{(-k)}, \hat{\varepsilon}_j^{(k)}\}_{j=1}^n) - \xi \\ &> HSIC(\{X_j^{(-k)}, \hat{\varepsilon}_j^{(k)}\}_{j=1}^n) - HSIC(\mathbb{P}^{X^{(-k)}, \hat{\varepsilon}}) = o_P(1). \end{aligned}$$

Second, assume that k is a sink node. Then $\{X_j^{(k)}, X_j^{(-k)}\}_{j=1}^n$ is an i.i.d. sample from the PNL regression model (1). Using the (point-wise) consistency of the estimators, we have

$$\hat{h}_k(X_1^{(k)}) - h(X_1^{(k)}) = o_P(1) \text{ and } \left(X_1^{(-k)}\right)^T \hat{\beta}^{(k)} - \left(X_1^{(-k)}\right)^T \beta^{(k)} = o_P(1).$$

Thus, we obtain

$$\hat{\varepsilon}_1^{(k)} - \varepsilon_1^{(k)} = \hat{h}_k(X_1^{(k)}) - h(X_1^{(k)}) - \left((X_1^{(-k)})^T \hat{\beta}^{(k)} - (X^{(-k)})^T \beta^{(k)} \right) = o_P(1) - o_P(1) = o_P(1). \quad (7)$$

From the definition of t_k and the decomposition of the HSIC, we have

$$t_k := HSIC(\{X_j^{(-k)}, \hat{\varepsilon}_j^{(k)}\}_{j=1}^n) = \frac{1}{n^2} \sum_{i,j=1}^n K_{ij} \hat{L}_{ij} + \frac{1}{n^4} \sum_{i,j,q,r=1}^n K_{ij} \hat{L}_{ij} - \frac{2}{n^3} \sum_{i,j,q=1}^n K_{ij} \hat{L}_{ij} = \hat{Q}_1 + \hat{Q}_2 - \hat{Q}_3,$$

where

$$K_{ij} := \exp\left(-\left\|X_i^{(-k)} - X_j^{(-k)}\right\|_2^2\right), \quad \hat{L}_{ij} := \exp(-(\hat{\varepsilon}_i^{(k)} - \hat{\varepsilon}_j^{(k)})^2),$$

$$\text{and } \hat{Q}_1 := \frac{1}{n^2} \sum_{i,j=1}^n K_{ij} \hat{L}_{ij}, \quad \hat{Q}_2 := \frac{1}{n^4} \sum_{i,j,q,r=1}^n K_{ij} \hat{L}_{ij}, \quad \hat{Q}_3 := \frac{2}{n^3} \sum_{i,j,q=1}^n K_{ij} \hat{L}_{ij}.$$

Similar to the proof of Theorem 2 in Teran Hidalgo et al. (2018) we will show that $t_k - HSIC(\mathbb{P}^{X^{(-k)}, \varepsilon^k}) = o_P(1)$. From Lemma 1 in Gretton et al. (2005) we have

$$HSIC(\mathbb{P}^{X^{(-k)}, \varepsilon^k}) = Q_1 + Q_2 - Q_3,$$

where $L_{ij} := \exp(-(\varepsilon_i^{(k)} - \varepsilon_j^{(k)})^2)$, $Q_1 := \mathbb{E}_{X_1^{(-k)}, \varepsilon_1^{(k)}, X_2^{(-k)}, \varepsilon_2^{(k)}}[K_{12}L_{12}]$, $Q_2 := \mathbb{E}_{X_1^{(-k)}, X_2^{(-k)}}[K_{12}]\mathbb{E}_{\varepsilon_1^{(k)}, \varepsilon_2^{(k)}}[L_{12}]$ and $Q_3 := 2\mathbb{E}_{X_1^{(-k)}, \varepsilon_1^{(k)}}[\mathbb{E}_{X_1^{(-k)}}[K_{12}]\mathbb{E}_{\varepsilon_1^{(k)}}[L_{12}]]$.

We show that $\hat{Q}_1 - Q_1 = o_P(1)$. We have

$$\begin{aligned} \hat{Q}_1 - Q_1 &= \frac{1}{n^2} \sum_{i,j=1}^n K_{ij} \hat{L}_{ij} - \mathbb{E}[K_{12}L_{12}] = \frac{1}{n^2} \sum_{i \neq j}^n K_{ij} \hat{L}_{ij} - \mathbb{E}[K_{12}L_{12}] + \frac{1}{n} \\ &= \frac{1}{n(n-1)} \sum_{i \neq j}^n K_{ij} \hat{L}_{ij} - \mathbb{E}[K_{12}L_{12}] + \frac{1}{n} - \frac{1}{n^2(n-1)} \sum_{i \neq j}^n K_{ij} \hat{L}_{ij} \\ &= \frac{1}{n(n-1)} \sum_{i \neq j}^n K_{ij} \hat{L}_{ij} - \mathbb{E}[K_{12}\hat{L}_{12}] + \mathbb{E}[K_{12}\hat{L}_{12}] - \mathbb{E}[K_{12}L_{12}] + \frac{1}{n} - \frac{1}{n^2(n-1)} \sum_{i \neq j}^n K_{ij} \hat{L}_{ij}, \end{aligned} \quad (8)$$

where we used $K_{jj} = \hat{L}_{jj} = 1$ in the second equality. For $\delta > 0$, the Markov inequality gives

$$\begin{aligned} \mathbb{P}\left(\left|\frac{1}{n(n-1)} \sum_{i \neq j}^n K_{ij} \hat{L}_{ij} - \mathbb{E}[K_{12}\hat{L}_{12}]\right| \geq \delta\right) &\leq \frac{\mathbb{E}\left[\left(\frac{1}{n(n-1)} \sum_{i \neq j}^n K_{ij} \hat{L}_{ij} - \mathbb{E}[K_{12}\hat{L}_{12}]\right)^2\right]}{\delta^2} \\ &= \frac{1}{n(n-1)\delta^2} \text{Var}(K_{12}\hat{L}_{12}) + \frac{1}{n^2(n-1)^2\delta^2} \sum_{i \neq j} \sum_{p \neq q} \text{Cov}(K_{ij}\hat{L}_{ij}, K_{pq}\hat{L}_{pq}) \\ &= O(1)\frac{1}{n(n-1)} + O(1)\frac{1}{n} \text{Cov}(K_{12}\hat{L}_{12}, K_{13}\hat{L}_{13}) + O(1)\text{Cov}(K_{12}\hat{L}_{12}, K_{34}\hat{L}_{34}). \end{aligned} \quad (9)$$

Here, we used that K_{ij} and \hat{L}_{ij} are bounded by 1, thus, their variances are bounded. The number of terms in the quadruple sum that have exactly three different indices are of order n^3 , but the denominator is of order n^4 , which leads to the second $O(1)$ term. The last $O(1)$ comes from the fact that the number of terms in the quadruple sum that have exactly four different indices are of order n^4 .

Using (7) and the continuous mapping theorem we obtain

$$K_{12}\hat{L}_{12} - K_{12}L_{12} = o_P(1),$$

and since \hat{L}_{12} is bounded, it is uniformly integrable and we obtain

$$\mathbb{E}[K_{12}\hat{L}_{12}] - \mathbb{E}[K_{12}L_{12}] = o(1).$$

In a similar way, we obtain

$$\mathbb{E}[K_{12}\hat{L}_{12}K_{34}\hat{L}_{34}] - \mathbb{E}[K_{12}L_{12}K_{34}L_{34}] = o(1).$$

Using the fact that $K_{12}L_{12}$ is independent from $K_{34}L_{34}$ and employing the two equalities above, we have

$$\begin{aligned} \text{Cov}(K_{12}\hat{L}_{12}, K_{34}\hat{L}_{34}) &= \mathbb{E}[K_{12}\hat{L}_{12}K_{34}\hat{L}_{34}] - \mathbb{E}[K_{12}\hat{L}_{12}]\mathbb{E}[K_{34}\hat{L}_{34}] \\ &= \mathbb{E}[K_{12}L_{12}K_{34}L_{34}] - \mathbb{E}[K_{12}L_{12}]\mathbb{E}[K_{34}L_{34}] + o_P(1) = o_P(1). \end{aligned}$$

Thus, all terms in (9) are $o_P(1)$. Moreover, from the above arguments we have $\mathbb{E}[K_{12}\hat{L}_{12}] - \mathbb{E}[K_{12}L_{12}] = o(1)$ and $\frac{1}{n^2(n-1)} \sum_{i \neq j} K_{ij}\hat{L}_{ij} = o(1)$ as $K_{ij}\hat{L}_{ij}$ is bounded and the denominator is of order n^3 . So, in (8) all the terms are $o_P(1)$, which gives

$$\hat{Q}_1 - Q_1 = o_P(1).$$

In the same fashion, one can show that $\hat{Q}_2 - Q_2 = o_P(1)$ and $\hat{Q}_3 - Q_3 = o_P(1)$, which put together proves that $t_k - HSIC(\mathbb{P}^{X^{(-k)}, \varepsilon^k}) = o_P(1)$. Moreover, $HSIC(\mathbb{P}^{X^{(-k)}, \varepsilon^k}) = 0$, since $X^{(k)}$ is a sink node and thus the noise ε^k is independent from the remaining nodes $X^{(-k)}$. Thus,

$$t_k \rightarrow 0 \text{ as } n \rightarrow \infty \text{ if } k \text{ is a sink node,}$$

which completes the proof of the Lemma. \square

Now we can prove Theorem 3.1.

For any set $A \subseteq \{1, 2, \dots, m\}$ we denote the sub-graph of \mathcal{G}^0 over the nodes A as \mathcal{G}_A^0 . Then

$$\begin{aligned} \mathbb{P}(\hat{\pi} \in \Pi^0) &= \mathbb{P}(\forall k \in \{1, 2, \dots, m\} : \hat{\pi}(k) \text{ is a sink node in } \mathcal{G}_{\{\hat{\pi}(1), \dots, \hat{\pi}(k)\}}^0) \\ &= 1 - \mathbb{P}(\exists k \in \{1, 2, \dots, m\} : \hat{\pi}(k) \text{ is not a sink node in } \mathcal{G}_{\{\hat{\pi}(1), \dots, \hat{\pi}(k)\}}^0) \\ &\geq 1 - \sum_{k=1}^m \mathbb{P}(\hat{\pi}(k) \text{ is not a sink node in } \mathcal{G}_{\{\hat{\pi}(1), \dots, \hat{\pi}(k)\}}^0). \end{aligned}$$

The proof of Theorem 3.1 is completed if we show that

$$\mathbb{P}(\hat{\pi}(k) \text{ is a sink node in } \mathcal{G}_{\{\hat{\pi}(1), \dots, \hat{\pi}(k)\}}^0) \rightarrow 1 \text{ as } n \rightarrow \infty \quad \forall k \in \{1, 2, \dots, m\}, \quad (10)$$

since this implies

$$\mathbb{P}(\hat{\pi}(k) \text{ is not a sink node in } \mathcal{G}_{\{\hat{\pi}(1), \dots, \hat{\pi}(k)\}}^0) \rightarrow 0 \text{ as } n \rightarrow \infty \quad \forall k \in \{1, 2, \dots, m\}.$$

Using assumption (A) and the recursive construction of $\hat{\pi}$ it is enough to prove (10) for $k = m$, that is

$$\mathbb{P}(\hat{\pi}(m) \text{ is a sink node in } \mathcal{G}^0) \rightarrow 1 \text{ as } n \rightarrow \infty.$$

By Lemma D.1 we know that t_k goes to zero in probability for sink nodes and is at least ξ for other nodes. Thus,

$$\mathbb{P}(\hat{\pi}(m) \text{ is a sink node in } \mathcal{G}^0) = \mathbb{P}(\arg \min \{t_k\} \text{ is a sink node in } \mathcal{G}^0) \rightarrow 1 \text{ as } n \rightarrow \infty.$$

This completes the proof of the Theorem.

E NUMERICAL RESULTS

In this Section we provide the concrete simulation results in tabular form and additionally provide the empirical standard deviations if the results.

Tables 1, 2, 3 and 4, 5, 6 show the results of RankG, AbPNL and RESIT methods for 4-dimensional graphs with strong and weak signal settings, for Gaussian, Gumbel and Logistic noises, respectively. Tables 7, 8, 9 and 10, 11, 12 show the results for 7-dimensional graphs. In both cases the sample sizes are 100, 500, 1000, 1500, and 2000, respectively.

Moreover, Tables 13, 14, 15 and 16, 17, 18 show the results of RankS, RankG, AbPNL and RESIT methods for 4-dimensional graphs with strong and weak signal settings, respectively. Tables 19, 20, 21 and 22, 23, 24 show the results of RankS, RankG, AbPNL and RESIT methods for 4-dimensional graphs where the function g is polynomial of degree 4 with strong and weak signal settings, respectively. In both cases the sample sizes are 100, 150, 200, 250, and 300.

-	Gaussian noise		
-	RankG	AbPNL	RESIT
100	2.51 ± 1.26	3.94 ± 1.09	2.4 ± 1.16
500	1.86 ± 1.31	3.46 ± 1.08	2.41 ± 1.23
1000	1.61 ± 1.15	3.14 ± 1.14	2.43 ± 1.16
1500	1.66 ± 1.35	3.28 ± 1.18	2.62 ± 1.36
2000	1.63 ± 1.17	3.23 ± 1.12	3.15 ± 1.51

Table 1: Results of RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-100, 100)$ for Gaussian noise (100 repetitions).

-	Gumbel noise		
-	RankG	AbPNL	RESIT
100	2.86 ± 1.36	3.98 ± 1.36	2.3 ± 1.18
500	2.34 ± 1.45	3.07 ± 1.24	2.49 ± 1.2
1000	1.8 ± 1.28	3.31 ± 1.11	2.43 ± 1.28
1500	1.68 ± 1.1	3.06 ± 1.03	3.04 ± 1.31
2000	1.37 ± 1.14	3.17 ± 1.18	3.02 ± 1.41

Table 2: Results of RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-100, 100)$ for Gumbel noise (100 repetitions).

-	Logistic noise		
-	RankG	AbPNL	RESIT
100	2.61 ± 1.35	3.89 ± 1.11	2.31 ± 1.1
500	2.06 ± 1.34	3.32 ± 1.24	2.36 ± 1.13
1000	1.85 ± 1.37	3.41 ± 1.1	2.66 ± 1.24
1500	1.68 ± 1.38	3.27 ± 1.2	2.92 ± 1.32
2000	1.79 ± 1.18	3.29 ± 1.16	3.44 ± 1.38

Table 3: Results of RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-100, 100)$ for Logistic noise (100 repetitions).

-	Gaussian noise		
-	RankG	AbPNL	RESIT
100	1.21 ± 1.23	3.7 ± 1.28	2.72 ± 1.2
500	0.21 ± 0.56	2.8 ± 1.01	2.69 ± 1.12
1000	0.15 ± 0.59	3.18 ± 1.08	2.73 ± 1.08
1500	0.04 ± 0.24	3.06 ± 1.08	2.9 ± 1.06
2000	0.05 ± 0.3	3.26 ± 1.17	3.1 ± 1.29

Table 4: Results of RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-10, 10)$ for Gaussian noise (100 repetitions).

-	Gumbel noise		
-	RankG	AbPNL	RESIT
100	1.44 ± 1.26	3.47 ± 1.13	2.86 ± 1.26
500	0.42 ± 0.91	2.61 ± 1.1	2.62 ± 1.19
1000	0.13 ± 0.39	2.88 ± 1.15	2.75 ± 1.15
1500	0.15 ± 0.54	2.76 ± 1.14	2.98 ± 1.2
2000	0.08 ± 0.37	2.83 ± 1.21	3.29 ± 1.26

Table 5: Results of RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-10, 10)$ for Gumbel noise (100 repetitions).

-	Logistic noise		
-	RankG	AbPNL	RESIT
100	1.05 ± 1.3	3.82 ± 1.15	2.5 ± 1.24
500	0.39 ± 0.79	2.84 ± 1.02	2.46 ± 1
1000	0.14 ± 0.47	3.01 ± 1.14	2.53 ± 1.11
1500	0 ± 0	2.95 ± 1.18	2.98 ± 1.05
2000	0 ± 0	2.69 ± 1.17	3.28 ± 1.21

Table 6: Results of RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-10, 10)$ for Logistic noise (100 repetitions).

-	Gaussian noise		
-	RankG	AbPNL	RESIT
100	4.65 ± 2.21	6.72 ± 2.13	3.89 ± 1.85
500	3.75 ± 2.11	5.4 ± 1.88	4.1 ± 1.76
1000	3.29 ± 2.03	5.46 ± 1.96	4.27 ± 1.95
1500	2.94 ± 2.13	5.4 ± 1.87	4.34 ± 1.88
2000	3.35 ± 2.09	5.49 ± 1.84	5.02 ± 2.09

Table 7: Results of RankG, RESIT and AbPNL methods on 7 nodes with $\beta \sim U(-100, 100)$ for Gaussian noise (100 repetitions).

-	Gumbel noise		
-	RankG	AbPNL	RESIT
100	4.74 ± 2.12	6.33 ± 2.26	3.95 ± 2.09
500	3.68 ± 1.97	5.41 ± 1.83	3.95 ± 1.75
1000	3.4 ± 1.98	5.39 ± 1.92	3.78 ± 1.92
1500	3.33 ± 2.05	5.96 ± 2.12	4.66 ± 1.99
2000	3.47 ± 2.61	5.26 ± 1.7	5.07 ± 2.41

Table 8: Results of RankG, RESIT and AbPNL methods on 7 nodes with $\beta \sim U(-100, 100)$ for Gumbel noise (100 repetitions).

-	Logistic noise		
-	RankG	AbPNL	RESIT
100	4.56 ± 2.1	6.45 ± 2.08	4.01 ± 1.96
500	3.65 ± 2.13	5.52 ± 1.49	3.86 ± 1.85
1000	3.88 ± 2.42	5.35 ± 1.96	4.18 ± 1.88
1500	2.93 ± 1.98	5.37 ± 1.8	4.42 ± 2.01
2000	3.39 ± 1.97	5.1 ± 1.64	4.96 ± 2.17

Table 9: Results of RankG, RESIT and AbPNL methods on 7 nodes with $\beta \sim U(-100, 100)$ for Logistic noise(100 repetitions).

-	Gaussian noise		
-	RankG	AbPNL	RESIT
100	2.44 ± 1.69	6.71 ± 2.47	4.15 ± 1.75
500	1.11 ± 1.46	5.06 ± 1.75	4.4 ± 1.84
1000	0.28 ± 0.75	5.08 ± 1.87	4.47 ± 1.88
1500	0.19 ± 0.6	5.18 ± 2	4.35 ± 1.79
2000	0.19 ± 0.69	5.26 ± 1.55	4.69 ± 2

Table 10: Results of RankG, RESIT and AbPNL methods on 7 nodes with $\beta \sim U(-10, 10)$ for Gaussian noise (100 repetitions).

-	Gumbel noise		
-	RankG	AbPNL	RESIT
100	3.13 ± 2.02	6.11 ± 2.22	4.17 ± 1.76
500	0.88 ± 1.4	4.32 ± 1.65	4.41 ± 1.74
1000	0.38 ± 0.93	4.7 ± 1.87	4.22 ± 1.97
1500	0.37 ± 0.97	4.88 ± 1.78	4.88 ± 1.87
2000	0.14 ± 0.55	4.81 ± 1.94	5.02 ± 2.06

Table 11: Results of RankG, RESIT and AbPNL methods on 7 nodes with $\beta \sim U(-10, 10)$ for Gumbel noise (100 repetitions).

-	Logistic noise		
-	RankG	AbPNL	RESIT
100	2.65 ± 2	6.83 ± 2.2	4.03 ± 1.54
500	0.68 ± 1.09	4.87 ± 1.82	4.12 ± 1.77
1000	0.45 ± 1.01	5.08 ± 1.79	4.27 ± 1.8
1500	0.34 ± 0.98	4.89 ± 1.87	4.55 ± 1.76
2000	0.2 ± 0.68	4.43 ± 1.69	4.56 ± 1.81

Table 12: Results of RankG, RESIT and AbPNL methods on 7 nodes with $\beta \sim U(-10, 10)$ for Logistic noise (100 repetitions).

-	Gaussian noise			
-	RankS	RankG	AbPNL	RESIT
100	2.23 ± 1.42	2.51 ± 1.26	3.94 ± 1.09	2.4 ± 1.16
150	1.73 ± 1.49	2.75 ± 1.25	2.6 ± 1.2	2.47 ± 1.13
200	1.63 ± 1.34	2.24 ± 1.32	2.87 ± 1.19	2.3 ± 1.02
250	1.56 ± 1.27	2.3 ± 1.16	3.07 ± 1.15	2.4 ± 1.22
300	1.49 ± 1.29	2.24 ± 1.35	2.97 ± 1.02	2.54 ± 1.1

Table 13: Results of RankS, RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-100, 100)$ for Gaussian noise (100 repetitions).

-	Gumbel noise			
-	RankS	RankG	AbPNL	RESIT
100	1.69 ± 1.36	2.86 ± 1.36	3.98 ± 1.36	2.3 ± 1.18
150	1.76 ± 1.28	2.56 ± 1.4	2.88 ± 1.34	2.37 ± 1.1
200	1.77 ± 1.38	2.78 ± 1.3	2.77 ± 1.17	2.38 ± 1.25
250	1.73 ± 1.27	2.44 ± 1.18	3.04 ± 1.13	2.45 ± 1.16
300	1.84 ± 1.23	2.4 ± 1.43	3.11 ± 1.12	2.35 ± 1.1

Table 14: Results of RankS, RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-100, 100)$ for Gumbel noise (100 repetitions).

-	Logistic noise			
-	RankS	RankG	AbPNL	RESIT
100	2.02 ± 1.32	2.61 ± 1.35	3.89 ± 1.11	2.31 ± 1.1
150	1.8 ± 1.26	2.58 ± 1.39	2.84 ± 1.26	2.7 ± 1.23
200	1.63 ± 1.28	2.5 ± 1.4	3.17 ± 1.25	2.55 ± 1.25
250	1.45 ± 1.23	2.21 ± 1.3	3.32 ± 1.23	2.3 ± 1.25
300	1.61 ± 1.32	2.3 ± 1.23	3 ± 1.04	2.4 ± 1.17

Table 15: Results of RankS, RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-100, 100)$ for Logistic noise (100 repetitions).

-	Gaussian noise			
-	RankS	RankG	AbPNL	RESIT
100	1.72 ± 1.45	1.21 ± 1.23	3.7 ± 1.28	2.72 ± 1.2
150	1.54 ± 1.36	1.11 ± 1.48	2.3 ± 1.41	2.69 ± 1.2
200	1.37 ± 1.37	0.71 ± 1.09	2.52 ± 1.29	2.72 ± 1.05
250	1.37 ± 1.28	0.77 ± 1.14	2.55 ± 1.07	2.78 ± 1.22
300	1.21 ± 1.27	0.48 ± 0.89	3.1 ± 1.28	2.57 ± 1.17

Table 16: Results of RankS, RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-10, 10)$ for Gaussian noise (100 repetitions).

-	Gumbel noise			
-	RankS	RankG	AbPNL	RESIT
100	1.73 ± 1.43	1.44 ± 1.26	3.47 ± 1.13	2.86 ± 1.26
150	1.63 ± 1.37	1.22 ± 1.33	2.04 ± 1.19	2.64 ± 1.28
200	1.41 ± 1.31	0.81 ± 1.02	2.5 ± 1.08	2.7 ± 1.32
250	1.48 ± 1.26	0.62 ± 1.08	2.38 ± 1.25	2.58 ± 1.24
300	1.72 ± 1.42	0.69 ± 1.05	2.44 ± 1.19	2.76 ± 1.12

Table 17: Results of RankS, RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-10, 10)$ for Gumbel noise (100 repetitions).

-	Logistic noise			
-	RankS	RankG	AbPNL	RESIT
100	1.78 ± 1.4	1.05 ± 1.3	3.82 ± 1.15	2.5 ± 1.24
150	1.44 ± 1.34	0.74 ± 1.02	2.86 ± 1.25	2.56 ± 1.16
200	1.55 ± 1.27	0.57 ± 1.02	3.18 ± 1.24	2.37 ± 1.17
250	1.3 ± 1.14	0.36 ± 0.72	3.14 ± 1.25	2.28 ± 1.09
300	1.38 ± 1.43	0.33 ± 0.75	3.23 ± 1.15	2.51 ± 1.11

Table 18: Results of RankS, RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-10, 10)$ for Logistic noise (100 repetitions).

-	Gaussian noise			
-	RankS	RankG	AbPNL	RESIT
100	1.24 ± 0.92	3.33 ± 1.41	3.7 ± 1.32	2.75 ± 1.31
150	1.42 ± 1.07	2.96 ± 1.5	3.1 ± 1.49	2.56 ± 1.21
200	1.37 ± 1.1	3.26 ± 1.54	3.37 ± 1.34	2.42 ± 1.32
250	1.78 ± 1.18	2.8 ± 1.55	3.35 ± 1.23	2.33 ± 1.12
300	1.63 ± 1	3.32 ± 1.59	3.52 ± 1.23	2.64 ± 1.31

Table 19: Results of RankS, RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-100, 100)$ for Gaussian noise and quartic polynomial g (100 repetitions).

-	Gumbel noise			
-	RankS	RankG	AbPNL	RESIT
100	1.11 ± 0.92	3 ± 1.37	3.33 ± 1.36	2.59 ± 1.23
150	1.51 ± 1.01	3.14 ± 1.37	2.7 ± 1.27	2.78 ± 1.2
200	1.56 ± 0.96	3.29 ± 1.41	3.22 ± 1.22	2.81 ± 1.26
250	1.63 ± 0.93	3.52 ± 1.41	3.34 ± 1.08	2.94 ± 1.29
300	1.86 ± 0.98	3.41 ± 1.44	3.61 ± 1.15	2.73 ± 1.25

Table 20: Results of RankS, RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-100, 100)$ for Gumbel noise and quartic polynomial g (100 repetitions).

-	Logistic noise			
-	RankS	RankG	AbPNL	RESIT
100	2.02 ± 1.32	2.61 ± 1.35	3.52 ± 1.24	2.31 ± 1.1
150	1.8 ± 1.26	2.58 ± 1.39	3.23 ± 1.18	2.7 ± 1.23
200	1.63 ± 1.28	2.5 ± 1.4	3.47 ± 1.09	2.55 ± 1.25
250	1.45 ± 1.23	2.21 ± 1.3	3.35 ± 1.07	2.3 ± 1.25
300	1.61 ± 1.32	2.3 ± 1.23	3.45 ± 1.41	2.4 ± 1.17

Table 21: Results of RankS, RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-100, 100)$ for Logistic noise and quartic polynomial g (100 repetitions).

-	Gaussian noise			
-	RankS	RankG	AbPNL	RESIT
100	2.01 ± 1.38	2.92 ± 1.66	3.64 ± 1.28	2.63 ± 1.17
150	2.34 ± 1.29	2.75 ± 1.72	2.64 ± 1.43	2.8 ± 1.29
200	2.44 ± 1.42	2.7 ± 1.62	3.19 ± 1.35	2.81 ± 1.32
250	2.53 ± 1.49	2.42 ± 1.65	3.57 ± 1.27	2.63 ± 1.24
300	2.48 ± 1.57	2.79 ± 1.8	3.7 ± 1.2	2.58 ± 1.34

Table 22: Results of RankS, RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-10, 10)$ for Gaussian noise and quartic polynomial g (100 repetitions).

-	Gumbel noise			
-	RankS	RankG	AbPNL	RESIT
100	1.85 ± 1.17	2.99 ± 1.49	3.55 ± 1.13	2.83 ± 1.23
150	2.15 ± 1.21	2.68 ± 1.56	2.54 ± 1.43	2.84 ± 1.39
200	2.43 ± 1.37	2.69 ± 1.59	2.71 ± 1.27	2.87 ± 1.3
250	2.51 ± 1.33	2.9 ± 1.61	3.2 ± 1.26	3.05 ± 1.45
300	2.32 ± 1.35	3.03 ± 1.38	3.1 ± 1.18	2.94 ± 1.32

Table 23: Results of RankS, RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-10, 10)$ for Gumbel noise and quartic polynomial g (100 repetitions).

-	Logistic noise			
-	RankS	RankG	AbPNL	RESIT
100	2.12 ± 1.17	2.97 ± 1.45	3.49 ± 1.36	2.61 ± 1.38
150	2.29 ± 1.39	2.93 ± 1.42	3.09 ± 1.39	2.63 ± 1.19
200	2.4 ± 1.36	2.91 ± 1.54	3.5 ± 1.17	2.45 ± 1.2
250	2.34 ± 1.34	2.81 ± 1.5	3.34 ± 1.23	2.53 ± 1.15
300	2.48 ± 1.44	2.76 ± 1.72	3.29 ± 1.19	2.59 ± 1.24

Table 24: Results of RankS, RankG, RESIT and AbPNL methods on 4 nodes with $\beta \sim U(-10, 10)$ for Logistic noise and quartic polynomial g (100 repetitions).