





Chain Graph Reduction Into Power Chain Graphs

Vithor Rosa Franco¹ , Guilherme Wang Barros² , Marie Wiberg² ,
Jacob Arie Laros³ 

[1] Post-graduate program of Psychology, Department of Psychology, São Francisco University, Campinas, Brazil.

[2] Department of Statistics, Umeå School of Business, Economics and Statistics, Umeå University, Umeå, Sweden.

[3] Post-graduate program of Social, Work and Organizational Psychology, Institute of Psychology, University of Brasília, Brasília, Brazil.

Quantitative and Computational Methods in Behavioral Sciences, 2022, Article e8383,
<https://doi.org/10.5964/qcmb.8383>

Received: 2022-02-17 • **Accepted:** 2022-10-21 • **Published (VoR):** 2022-12-22 • **Corrected (CVoR):** 2023-03-08

Handling Editor: Manuel Völkle, Humboldt University Berlin, Berlin, Germany

Corresponding Author: Vithor Rosa Franco, Laboratory of Evaluation Methods and Techniques, Darcy Ribeiro Campus, ICC sul, Room A1 - 061/4, University of Brasilia, Asa Norte, 70910900 - Brasília, DF - Brazil. E-mail: vithorfranco@gmail.com

Supplementary Materials: Data [see Index of Supplementary Materials]



Abstract

Reduction of graphs is a class of procedures used to decrease the dimensionality of a given graph in which the properties of the reduced graph are to be induced from the properties of the larger original graph. This paper introduces both a new method for reducing chain graphs to simpler directed acyclic graphs (DAGs), that we call power chain graphs (PCG), as well as a procedure for structure learning of this new type of graph from correlational data of a Gaussian graphical model. A definition for PCGs is given, directly followed by the reduction method. The structure learning procedure is a two-step approach: first, the correlation matrix is used to cluster the variables; and then, the averaged correlation matrix is used to discover the DAGs using the PC-stable algorithm. The results of simulations are provided to illustrate the theoretical proposal, which demonstrate initial evidence for the validity of our procedure to recover the structure of power chain graphs. The paper ends with a discussion regarding suggestions for future studies as well as some practical implications.

Keywords

graph reduction, power chain graph, Monte Carlo simulation, probabilistic graphical models, causal discovery, network models



This is an open access article distributed under the terms of the [Creative Commons Attribution 4.0 International License, CC BY 4.0](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction, provided the original work is properly cited.

A graph is an ordered pair, $G = (V, E)$, consisting of V , a set of nodes (also called vertices), and E , a set of edges, which are formally defined as pairs of nodes. Reduction, or simplification, of graphs is a class of methods used to reduce the dimensionality (defined as the total number of nodes and edges) of a graph. For a smaller graph to be considered a reduction of a larger graph, the properties of the reduced graph have to be induced from the properties of the larger original graph (Aref et al., 2014). In areas such as statistical learning, neuroscience, distributed control of networked dynamical systems, and others that use high-dimensional datasets (Xu et al., 2011), and in which the use of proxy variables is common, these procedures are of particular interest.

It appears from the literature on statistical and machine learning, as well as from the literature on probabilistic graphical models, that motif discovery (Hu et al., 2005), community detection (Javed et al., 2018) and other methods (e.g., Chao et al., 2019; Shah et al., 2017) are used as the main techniques to identify ways of combining variables or simplifying the graphical representation of a high-dimensional set of variables when the true graph structure is unknown. Graph reduction methods are used to reduce directed graphs or undirected graphs. For a third category of graphs, chain graphs, which utilize both directed and undirected edges, there is still no procedure to reduce them effectively.

The main objective of this research is to present a novel type of graph, called a power chain graph (PCG), which is a directed acyclic graph (DAG) derived from the reduction of a chain graph (CG; Cox & Wermuth, 1993), in which all nodes and undirected edges are contracted into a set of power nodes (i.e., nodes that represent groups of nodes), and all the original directed edges are contracted into a set of power edges (i.e., edges that represent groups of edges). Moreover, we propose a PCG structure learning procedure that can be used when the observed undirected graph is a Gaussian graphical model (GGM; Lauritzen, 1996; Uhler, 2017) and it is assumed that the true graph is a chain graph with unknown structure. Lastly, we evaluate the effectiveness of our PCG structure learning procedure by performing three simulation studies and an empirical example with real data.

The remainder of this paper is organized as follows. The next section presents a definition of PCGs and how to reduce CGs into PCGs. In the third section, we present a method for empirically learning the structures of PCGs. Following that, we conduct three simulation studies to test how well our structure learning procedure can recover the structures of simulated PCGs. Next, an empirical example is presented to illustrate how the PCG structure learning procedure can be used to derive causal hypotheses from the data. The paper concludes with a discussion and a few closing remarks.

Probabilistic Graphical Models, Graph Reduction and Power Chain Graphs

Probabilistic graphical models (Koller et al., 2009) are graphical representation of multivariate distributions that satisfy a set of conditional independence relations. In cases where the set of edges V and the set of nodes E are high dimensional, it may be of interest to represent the graph G in a reduced form. A graph H is called a graph minor (i.e., a reduction or an induced minor) of graph G if H can be formed from G by deleting edges and nodes, and by contracting edges (Lovász, 2006). The operation of edge contraction consists in removing an edge from a graph while simultaneously merging the two nodes that it previously connected (Asano & Hirata, 1983).

If G is an undirected graph (i.e., all edges in E are symmetric relations), then a power graph (Royer et al., 2008) $PG = (V', E')$ is a reduction of a graph G , with edges contracted from motifs. Motifs are recurrent or statistically significant sub-graphs or patterns in a graph (Holland & Leinhardt, 1976). For PGs, V' is a set of power nodes (i.e., a set of contracted nodes), and E' is a set of power edges (i.e., a set of contracted edges). PGs do not require all edges to be contracted, and the PG derived from an undirected graph is also an undirected graph.

A similar approach is also used for contraction of directed graphs (i.e., graphs where all edges in E are directional, or causal, relations; Harary et al., 1965). A directed graph is a graph $DG = (V, A)$, where A is a set of directed edges (also called arrows; Scutari & Denis, 2021). A node i is called a parent (or a cause) of j if the arrow from i to j is an element of A and a child (or an effect) of j if the arrow from j to i is an element of A . The reduction of a directed graph by contracting all of the nodes in each strongly connected component (i.e., a subgraph such as every node is reachable from every other node) always results in a directed acyclic graph (DAG; Lauritzen, 1996). A DAG is a type of directed graph such that there is no way to start at any node i and follow a consistently-directed sequence of edges that eventually goes back to i again.

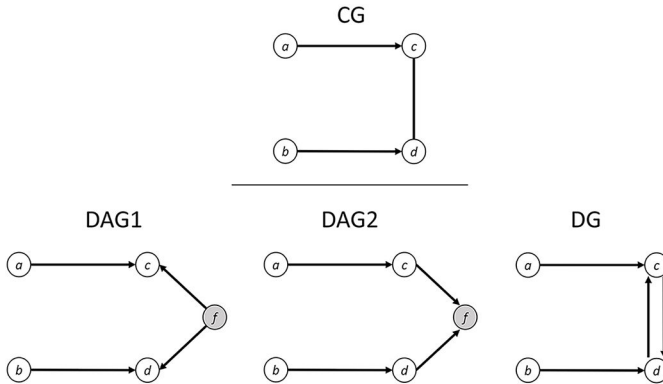
A special type of graph known as chain graph (Lauritzen, 1996), defined as $CG = (V, A, E)$, includes both a set of undirected edges E and a set of directed edges A . In the probabilistic graphical model literature, chain graphs are also known as mixed graphs with no semi-directed cycles and no bidirectional edges (Højsgaard et al., 2012), being thus a generalization of both undirected graphs and DAGs. CGs, like DAGs, can be used to make causal interpretations between variables connected by directed edges. However, the undirected edges in CGs can be interpreted in several different ways, depending on the assumed or real conditional dependencies and the presence or absence of latent variables. For instance, Lauritzen and Richardson (2002) used the example presented in Figure 1 to argue that undirected edges in CGs should not be interpreted as causal relations, unless extra information is presented. At the top of Figure 1 we present an observed CG, with the following factorization of the joint density of variables a , b , c , and d :

$$a \perp\!\!\!\perp b, a \perp\!\!\!\perp d \mid \{b, c\}, b \perp\!\!\!\perp c \mid \{a, d\}. \tag{1}$$

where $\perp\!\!\!\perp$ represents independence.

Figure 1

A CG (top) and Three Different DAGs That Have Different Factorizations



Note. CG = Chain Graph. DAG = Directed Acyclic Graph. DG = Directed Graph. Shaded nodes represent latent variables. Directional edges represent causal relations (from tail to head) and the undirected edge represents a relation with no specific causal representation.

The DAGs and the DG on the bottom are commonly used to interpret, or statistically model, the undirected edge on the CG. However, none of these graphs result in the same conditional distribution, as the joint density of DAG1, DAG2, and DG are, respectively, represented by

$$a \perp\!\!\!\perp \{b, d\}, b \perp\!\!\!\perp \{a, c\}, a \not\perp\!\!\!\perp d \mid \{b, c\}, b \not\perp\!\!\!\perp c \mid \{a, d\}, \tag{2}$$

$$a \perp\!\!\!\perp d \mid \{b, c\}, b \perp\!\!\!\perp c \mid \{a, d\}, a \not\perp\!\!\!\perp b, \tag{3}$$

$$a \perp\!\!\!\perp b, a \perp\!\!\!\perp b \mid \{c, d\}, a \not\perp\!\!\!\perp d \mid \{b, c\}, b \not\perp\!\!\!\perp c \mid \{a, d\}, \tag{4}$$

where $\not\perp\!\!\!\perp$ represents dependence. These relations show that CGs can, and should, be used when correlations have no clear causal interpretation. Despite this fact, it is not rare for causal models to be assumed, for instance, in psychometrics and fMRI studies, where the correlated data (e.g., test responses and voxels) are assumed to have a common latent cause (e.g., psychological construct and original neural signal). In these research areas, statistical modeling usually requires additional assumptions about how observed data relates to the latent causes (e.g., [Mair, 2018](#); [Roberts & Everson, 2001](#)).

In the literature on CGs, when the structure is high dimensional, chain graphs can be divided in a path of blocks of variables, known as dependence chains (Wermuth & Lauritzen, 1990). Dependence chains are used to illustrate and properly condition causal effects in CGs because within blocks the relations are undirected, but between blocks the relations are directed. The interpretation of these dependence chains relies on what interpretation of CGs is assumed in the given study. There are three main interpretations of CGs, each with its own peculiarities (Javidian & Valtorta, 2018): (i) Lauritzen-Wermuth-Frydenberg (LWF) CGs; (ii) Andersson-Madigan-Perlman (AMP) CGs; and (iii) multivariate regression (MVR) CGs.

The first two, AMP and LWF CGs, are characterized by having directed and undirected edges, but differ on how they establish independence relations between parents of vertices that are linked by an undirected edge. MVR CGs, on the other hand, use directed edges together with bidirectional edges. MVR CGs have more in common with AMP CGs, as each node depends only on its own parents according to both interpretations. Both interpretations can also be modelled with linear equations, being the main difference between MVR CGs and AMP CGs the modelling of noise (Sonntag, 2016). The bidirected edges in MVR CGs are used to model latent (or hidden) variables (Javidian & Valtorta, 2021), being related, then, to DAG1 represented in Figure 1. LWF CGs, on the other hand, assume that each node is affected by the parents of the whole block and, therefore, the causal effects propagate bidirectionally between undirected edges, being related, then, to DG represented in Figure 1.

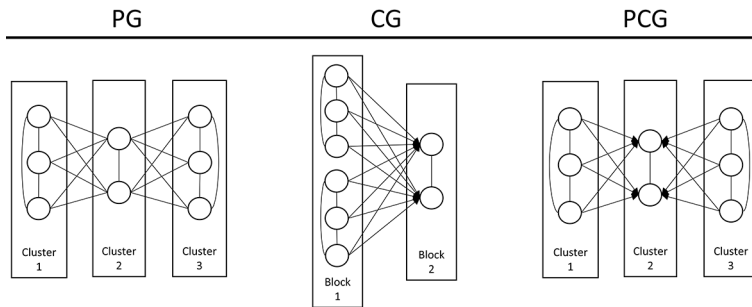
Despite the specific CG interpretation, if all nodes in a subgraph (i.e., a group of variables) have directed relations with all nodes of another subgraph, the nodes within each subgraph can be considered as part of the same block, even if the block represents a disconnected subgraph (i.e., a graph which not all nodes are directly connected). On the other hand, if the blocks are defined by clusters of variables, and if it is possible to test or assume the disjointness condition (i.e., that it is possible to define the direction of the relations between blocks), then we have a special type of CG which can be reduced to a PCG, as illustrated in Figure 2. Despite the fact that the factorization of both the CG and the PCG in Figure 2 are identical, the PCG representation allows further inference about the clustering process of the predictor variables and the causal chains as a whole, without having to assume, a priori, a specific CG interpretation.

The undirected part of the chain graphs can be reduced by graph minor operations, while the directed part can be reduced by the contraction of an undirected equivalent of the strongly connected component to achieve a power chain graph (PCG). The PCG can be defined as the graph $P = (V', A')$, where all original nodes and undirected edges are contracted into a set of power nodes, V' , and all the original directed edges are contracted into a set of directed power edges, A' . Figure 3 depicts this reduction process, where it is possible to see that a PCG is equivalent to a DAG that implies an underlying CG. MVR CGs can, at this point, again be contrasted with PCGs, in the sense that they also

operate a type of graph reduction. However, MVR CGs assume that strongly connected subgraphs result from a latent common cause, which have to be marginalized from the original CG before causal discovery can occur (Javidian & Valtorta, 2018; Sonntag & Peña, 2015). PCG does not require this additional assumption.

Figure 2

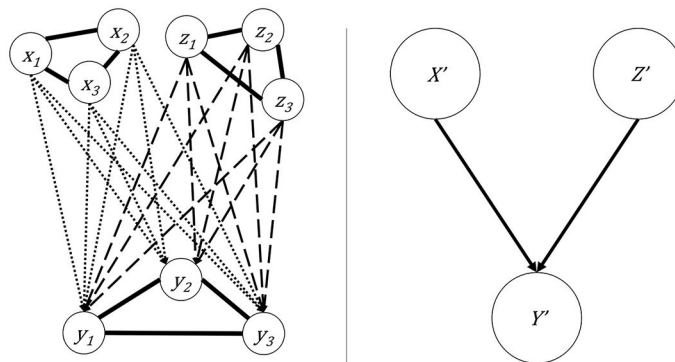
Comparison Between Dependencies Represented With PG, CG, and PCG



Note. PG = Power Graph. CG = Chain Graph. PCG = Power Chain Graph.

Figure 3

An Example of CG (to the left) and the PCG (to the right) Attained by Contracting the CG's Edges, Arrows, and Nodes



Note. X' is the power node representing the cluster (or community) of nodes x_1 , x_2 , and x_3 . Similar interpretation is given for power nodes Z' and Y' .

In the case of CGs, it has been shown that the structure of CGs, just like DAGs, can be, under the correct circumstances and assumptions, recovered from observational data (Peña & Gómez-Olmedo, 2016). A possible method for CGs is a PC-like algorithm, based on the Peter-Clark (PC) algorithm (Spirtes et al., 2000), which uses partial correlations

and theorems of causality theory (Pearl, 2009) to infer a causal structure from the data. However, this PC-like approach for CGs, just as for DAGs, will recover, in most cases, only a CG which is in the same equivalent class as the graph that generated the data (Peña & Gómez-Olmedo, 2016). On top of that, as CGs are more complex and general than DAGs, these equivalence classes tend to contain more options than for DAGs, so it would be in interest of a researcher if some CG structures could be reduced to DAG structures by PCG reduction.

If the CG is given or assumed, then reducing it to a PCG requires only three steps. First, one must define or choose a measurement function for assessing the conditional dependencies between the nodes. Then, the strongly connected undirected subgraphs are theoretically defined or empirically estimated from data. In the final step, a method for ensembling the weights of the directed edges of similar nodes in A into directed power edges in A' is applied to the graph. However, more usually than not, the CG is not given nor should be assumed to be known. In this more realistic scenario, structure learning procedure (also known as causal discovery algorithm) can be used to first estimate the PCG and then the relations in the CG can be implied from the PCG.

Structure Learning of Power Chain Graphs

We propose a structure learning procedure for PCGs where the dependencies between variables are assumed to be linear and, therefore, the variables are assumed to follow a multivariate normal distribution. Gaussian graphical models (GGMs; Lauritzen, 1996; Uhler, 2017) are a usual decision for modeling network models in this scenario (e.g., Drton & Perlman, 2008; Epskamp et al., 2018; Yin & Li, 2011). GGMs are mathematically convenient for modeling linear dependencies and conditional linear dependencies between variables, as GGMs present maximal entropy configuration and the possibility for intuitive maximum likelihood estimation.

The first step of our structure learning procedure for learning the structure of PCGs is the identification of the groups of similar nodes by some data-driven procedure. In a chain graph that can be represented by a GGM, any node v_a is more similar to any node v_b than to any node v_c if and only if $\lambda_{\rho_a, \rho_b} > \lambda_{\rho_a, \rho_c}$. The measure λ_{ρ_a, ρ_b} assesses the similarity between the vectors of dependencies ρ_a and ρ_b . The vector of dependencies ρ_a has as its elements the dependencies involving the v_a node. The vector of dependencies ρ_b is interpreted similarly to ρ_a in respect to v_b , and λ_{ρ_a, ρ_c} is interpreted similarly to λ_{ρ_a, ρ_b} in respect to ρ_a and ρ_c . With GGMs, Pearson correlation coefficient and regularized partial correlation are the most commonly used measures of dependency between nodes (Epskamp et al., 2018). Clusters (i.e., groups or communities) can then be found by conventional methods, such as exploratory factor analysis with parallel analysis (Horn, 1965), exploratory graph analysis (Golino & Epskamp, 2017), and motif discovery.

In the second step of our structure learning procedure, after finding the clusters of variables, the directions of the relations between power nodes can be estimated using causal discovery algorithms (Malinsky & Danks, 2018). More specifically, we propose that correlations between different clusters can be averaged and the resulting averaged correlation matrix can be fed to a causal discovery algorithm, resulting in the PCG. Because the steps in our structure learning procedure are complex and dependent on a series of intricate assumptions, we summarize the steps as well as the main assumptions in Table 1.

Table 1

Assumptions of Our Structure Learning Procedure for Power Chain Graphs

Step	Assumption	Description
Clustering and Contraction	Multivariate Gaussianity of variables	Variables in the dataset can be correctly described by a multivariate normal distribution. This assumption is necessary so the data can be modelled as a GGM.
	Multivariate Gaussianity of dependencies	Correlations estimated by maximum likelihood from the same sample asymptotically follow a multivariate normal distribution. This assumption is necessary so the weights of the edges estimated to be directed can be contracted by taking the arithmetic mean of the correlations between clusters.
Causal Discovery	Causal Markov condition	A variable x is independent of every other variable (except x 's effects) conditional on all of its direct causes. This assumption means that, in the absence of measurement error, all data generated from the same process will result in the same set of conditional dependencies.
	Faithfulness	A graph is faithful to the data generating process if no conditional independence relations other than the ones entailed by the Markov condition are present. This assumption means that correlations are not spurious and lack of correlations are not due to suppression effects, but rather to true independence.
	Causal sufficiency	The dataset includes all of the common causes of pairs of dependent variables in the dataset. This assumption means that there are no confounders (i.e., latent causes) that explain away the correlations between pairs of variables.
	Multivariate Gaussianity of variables	Same as above. But in this case, this assumption makes it possible to test the significance of the conditional dependencies of triplets of variables.

Finding Similar Nodes and Contracting Nodes

The clustering procedure that we propose departs from three main elements: (i) our definition of similar nodes; (ii) the fact proved by Rao (1979) that correlations estimated by maximum likelihood from the same sample asymptotically follow a multivariate normal distribution; and (iii) the use of model-based clustering as an extension of block modeling (Breiger et al., 1975). From these elements, we propose that model-based clustering, parameterized with finite Gaussian mixture models, can be used to find the clusters of similar nodes from the correlation matrix estimated from data. In our application of model-based block modeling, which we call correlation Gaussian mixture model (CGMM), the estimated correlations $p = (p_1, \dots, p_k)$ are assumed to be generated by a mixture model with C clusters:

$$f(p) = \prod_{i=1}^k \sum_{l=1}^C \tau_l f_l(p_i | \theta_l), \quad (5)$$

where $f_l(p_i | \theta_l)$ is the normal probability distribution with parameter θ_l , and τ_l is the probability of belonging to the l^{th} cluster. The parameters of the model can be estimated by maximum likelihood using the Expectation-Maximization algorithm (Dempster et al., 1977), with model comparison methods used to find the value of C which better fits the estimated correlation matrix. In multivariate settings, the volume, shape, and orientation of the covariances can be constrained to be equal or to vary across clusters, resulting in 14 possible models with different geometric characteristics. A modified version of the Bayesian information criterion can be used to choose the best solution (Fraley & Raftery, 2007).

The clusters identified by our procedure are the strongly connected undirected subgraphs of the CG. By our definition of a PCG, nodes from different clusters can be connected only by directed edges and, if a node from a cluster is connected to a node in another cluster, all nodes in the first cluster are connected to all nodes in the second cluster, with the edges directed in the same direction. Based on the normality assumption, contraction of edges, resulting in the weighted power edges E'_{lhgj} , can be conducted by calculating

$$E'_{lhgj} = r \left(\frac{1}{(N_l + N_h)} \sum_{g=1}^{N_l} \sum_{j=1}^{N_h} z_{gj} \right), \quad (6)$$

where N_l and N_h represents the number of nodes included in the clusters l and h , respectively, z_{gj} is the Fisher-transformed correlation of the g^{th} variable of a cluster l with the j^{th} variable of a cluster h , and $r()$ is equal to

$$r \left(z' \right) = \frac{\exp(2z') - 1}{\exp(2z') + 1}. \quad (7)$$

where z' is equal to $\frac{1}{(N_i + N_h)} \sum_{g=1}^{N_i} \sum_{j=1}^{N_h} z_{gj}$. This procedure will generate an $C \times C$ averaged correlation matrix that, we propose, can be used to learn the direction of the directed power edges of a PCG using a causal discovery algorithm.

Finding the Direction of the Relations Between Power Nodes

After calculating the weights of the directed power edges (i.e., after contracting what was estimated to be directed relations), one can use causal discovery methods to estimate causal relations from correlational data (Pearl, 2009). Causal discovery from correlational data is based on the fact that different causal paths (i.e., causal data generating processes) result in different conditional distributions on a set of variables. This can be shown by the factorization of the three fundamental connections, represented in Figure 4: serial connection; divergent connection; and convergent connection (also known as collider or v-structure). The factorization of the probability distributions of these connections can be expressed, respectively, as serial connection

$$\Pr(X_i)\Pr(X_j|X_i)\Pr(X_k|X_j); \quad (8)$$

divergent connection

$$\Pr(X_i|X_j)\Pr(X_j)\Pr(X_k|X_j); \quad (9)$$

and v-structure

$$\Pr(X_i)\Pr(X_j|X_i, X_k)\Pr(X_k); \quad (10)$$

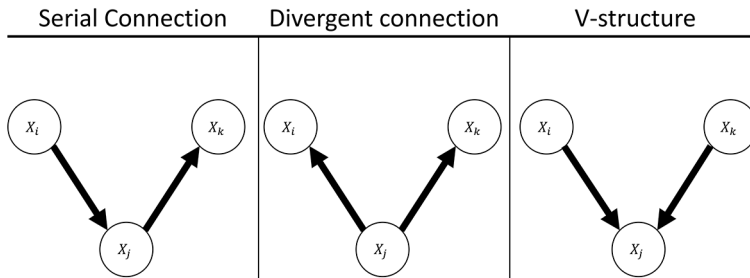
it being easy to see that Equation 8 is equivalent to Equation 9.

The fact that v-structures have a different factorization from the other types of connections implies that—assuming non-confounders and no cycles (i.e., mutual causation)—it is possible to identify causal effects even if the data is correlational, as long as v-structures, as expressed in Figure 4, can be found in the graph (Rohrer, 2018). However, it is important to note that the direction of some relations may not be identifiable due to the fact that some data generating processes entail the same dependence structure—i.e., are part of the same equivalence class—and, therefore, result in the same set of conditional dependencies. For instance, serial and divergent connections are part of the same equivalence class and, therefore, it is impossible to distinguish these types of dependencies from correlational data alone. Therefore, causal discovery algorithms will usually return complete partially DAGs instead of DAGs, given that a complete partially

DAG express an equivalence class consisting of the same directed and undirected edges (Verma & Pearl, 1990).

Figure 4

Three Fundamental Connections Between Three Variables



Note. X_i , X_j , and X_k represent any three nodes, or power nodes, and the data generating processes, in terms of causal relations.

There exist three classes of approaches for learning causal graphs (i.e., structure learning) from data (Scutari & Denis, 2021): constraint-based; score-based; and hybrid. One of the first constraint-based algorithms to be implemented was the PC algorithm, proposed by Spirtes et al., (2000), which we use to learn the directions of the edges in the PCG. The output of the original PC algorithm depends on the order in which the possible v-structures are tested. A simple modification proposed by Colombo and Maathuis (2014), called PC-stable, yields order-independent adjacencies in the skeleton, which is the partial correlations generalized from a DAG. This means that PC-stable finds a partial correlations graph before trying to direct the edges, reducing the computational expense of testing the possible v-structures due to the size of the graph.

Score-based algorithms, on the other hand, apply heuristic optimization techniques to the problem of structure learning (Russell & Norvig, 2009). This class of algorithms assigns network scores (i.e., a goodness-of-fit index) to possible structures, which are then selected based on how well they fit the data. The greedy equivalence search (GES; Chickering, 2003) and the hill climbing (HC; Daly & Shen, 2007) algorithms use locally optimal choices at several iterations, until a solution is found. They have shown good performance when compared to the PC algorithm or others score-based algorithms (e.g., Russell & Norvig, 2009). However, they do not evaluate the existence of v-structures and, therefore, may be less conservative than constraint-based algorithms.

Hybrid learning algorithms, as the name may suggest, combine both constraint-based and score-based algorithms to trying to overcome the limitations of single class algorithms (Friedman et al., 1999; Tsamardinou et al., 2006). One simple approach for a hybrid learning procedure would be, for example, to learn the skeleton of a DAG by means of

any constraint-based algorithm (what is called a restriction phase), and then to direct the edges accordingly to a score-based algorithm (called maximization phase). The Max–Min Hill Climbing (MMHC; [Tsamardinos et al., 2003](#)) algorithm performs this exact procedure by combining both the PC-stable and the HC algorithms.

Despite the fact that the PC-stable is the most adequate algorithm to learn the directions of the edges in a PCG, as it can be applied using the averaged correlation matrix estimated from the first step of our structure learning procedure, the other algorithms can be used for “fine-tuning” the CG implied by the PCG. This means that other algorithms, or even the PC-stable itself, can be used, after learning the structure of PCG and generating the implied CG, to remove directed edges that may not hold true causal relations between nodes from different power nodes. Therefore, this step of “fine-tuning”, despite not essential for learning the structure of a PCG, can shed a light on how to improve structure learning of CGs ([Peña, 2018](#)).

Finally, it should be noted that current procedures used for learning the structure of CGs, as well as the structure of DAGs, can only recover an equivalence class of directed, or partially directed, graphs. This fact will hold true even when the underlying distribution (i.e., the data generating process) is known, as any two CGs (or DAGs) that belong to the same equivalence class are causally indistinguishable ([Ma et al., 2008](#); [Peña & Gómez-Olmedo, 2016](#)). This characteristic is not a limitation of these procedures, but a fundamental restriction due to the nature of the relations between causality and observed correlations ([Pearl, 2009](#)). Therefore, our procedure will also return complete partially DAGs whenever the underlying distribution results in a causally indistinguishable set of conditional dependences.

Simulation Studies

Method

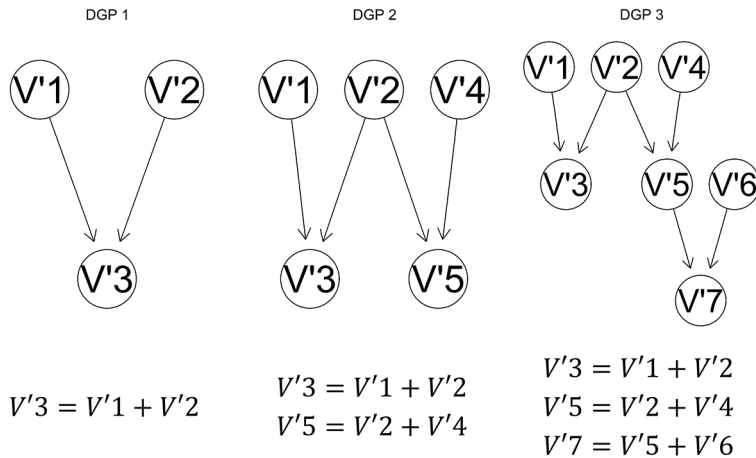
Data Generating Process (DGP)

The DGP demanded three characteristics to properly address our aim: variables should be normally distributed in groups of similar nodes; different groups should be causally related; and the simulated PCG should be the only element on its equivalence class (so it is causally distinguishable). The first two characteristics are simply characteristics of PCGs. The third characteristic guarantees that, in the absence of both unmeasured confounders and measurement error, there is only one best causal model to explain the dependencies in the data ([Verma & Pearl, 1988](#)).

The DGPs were based on three different GGM chain graphs, with their corresponding PCGs represented in [Figure 5](#). GGMs are usually difficult to simulate as they require the correlation matrix to be positive definite ([Williams & Mulder, 2020](#)). To simplify the simulation of these DGPs, we used a factor analytical model (i.e., an MVR CG). For doing

Figure 5

DGPs of the PCGs in the Present Study



Note. DGP = Data Generating Process. $V'1$ to $V'7$ represent power nodes.

this, for each l^{th} independent power node ($V'1$, $V'2$, $V'4$, and $V'6$), a vector f_l of size n was randomly drawn from a normal distribution with mean 0 and standard deviation 1. The dependent power nodes ($V'3$, $V'5$, and $V'7$) were created by simply summing the vectors of the parents of the given power node (e.g., $f_3 = f_1 + f_2$). Observations x_{ij} for the variables for each power node were created using a linear factorial model:

$$x_{ij} = u_j + \lambda_{jl}f_{il} + \varepsilon_{ij}\sqrt{1 - \lambda_{jl}^2}, \tag{11}$$

where u_j is the intercept for variable j (which was drawn from a normal distribution with mean 0 and standard deviation 1), λ_{jl} is the correlation of variable j with the latent variable l (and these correlations were randomly drawn from a uniform distribution with lower bound .4 and upper bound .8), and ε_{ij} is the random component for simulated individual i on variable j , which was also drawn from a normal distribution with mean 0 and standard deviation 1.

DGP1 was chosen because it represents a v-structure, the simplest possible model to be unique in its equivalence class. Both DGP2 and DGP3 were built upon DGP1, increasing the number of nodes, but keeping the same nodes and directed edges, as well as the characteristic of being unique in its equivalence class. For instance, $V'2$ causes $V'3$ in all the DGPs. On the other hand, $V'2$ only causes $V'5$ in DGP2 and DGP3. This strategy reflects an important aspect of DAGs (Pearl, 2009): as long as confounders do not change the dependence relations in the graph, they can be estimated with efficiency by causal

discovery algorithms. Therefore, even in the absence of some nodes, the algorithm applied to DGP1 should perform at least as well as when compared to its applications to DGP2 and DGP3. Two other conditions were used: the number of variables per power node (or cluster size; $v = 5$ or $v = 10$) and the sample size ($n = 100$, $n = 250$ or $n = 500$). Each triplet—composed by a DGP, a cluster size, and a sample size—of these conditions was simulated 1,000 times.

Performance Indices

Our proposed clustering procedure was compared to two other procedures: Exploratory Factor Analysis (EFA; [Goretzko et al., 2021](#)) with Parallel Analysis (PA; [Horn, 1965](#)), hereafter called PA-EFA; and the Exploratory Graph Analysis (EGA; [Golino & Epskamp, 2017](#)). The PA-EFA is used in many fields, such as test development in psychology and dimension reduction in machine learning. EGA is a promising new procedure that combines two procedures from the probabilistic graphical models' framework for identifying the correct number of graph communities (i.e., clusters of variables): the regularized partial correlations GGM, and the walktrap algorithm for community detection.

For assessing the quality of the clustering procedures, we used cluster performance indices ([Hennig & Liao, 2013](#)). For comparing the structure learning algorithms, we used accuracy performance indices ([Glaros & Kline, 1988](#)). The accuracy indices were accuracy (Acc), positive predictive value (PPV), and true positive and true negative rates (TPR and TNR, respectively). The cluster indices were the graph adjusted Rand index (GARI; [Terada & von Luxburg, 2014](#)), variation of information (VI; [Meilă, 2007](#)), normalized mutual information (NMI; [Haghighat et al., 2011](#)), and whether the clustering procedure identified the correct number of dimensions (HitND; also known as cluster accuracy). The formulas for calculating these indices, their numerical ranges, and theoretical interpretations are presented in [Table 2](#).

Table 2

Performance Indices Used in This Study for Comparing Algorithms Effectiveness

Index	Formula	Range	Interpretation
Acc	$\frac{TP + TN}{TP + FP + FN + TN}$	[0, 1]	Values closer to 1 are preferred as they mean that all edges and all lack of edges were correctly identified
PPV	$\frac{TP}{TP + FP}$	[0, 1]	Values closer to 1 are preferred as they mean that the proportion of correctly identified edges is larger than the proportion of incorrectly identified edges
TPR	$\frac{TP}{TP + FN}$	[0, 1]	Values closer to 1 are preferred as they mean that the proportion of correctly identified edges is larger than the proportion of incorrectly identified lack of edges
TNR	$\frac{TN}{TN + FP}$	[0, 1]	Values closer to 1 are preferred as they mean that the proportion of correctly identified lack of edges is larger than the proportion of incorrectly identified edges
NMI	$\frac{I(\hat{\vartheta}; \vartheta)}{\sqrt{H(\hat{\vartheta})H(\vartheta)}}$	[0, 1]	Values closer to 1 are preferred as they mean that the estimated cluster is dependent of the true cluster
HitND	$\frac{1}{R} \sum_{i=1}^R Cor_i$	[0, 1]	Can be interpreted as the power of the clustering procedures, with values closer to 1 preferred as they mean that the procedure has always found the correct number of clusters
GARI	$\frac{\sum_{j=1}^n (\widehat{M}_j - D[M_j])}{\sum_{j=1}^n (\max \widehat{M}_j - D[M_j])}$	$(-\infty, 1]$	Negative values mean that the clusters being compared have completely different structures and the value of 1 means that they have exactly the same structure
VI	$2H(\hat{\vartheta}, \vartheta) - H(\hat{\vartheta}) - H(\vartheta)$	$VI \leq 2\log(K^*)$	Smaller values are preferred as they mean all variables that should be clustered together were properly clustered together

Note. TP = the number of correctly identified edges that existed in the true graph. TN = the number of correctly identified lack of edges that existed in the true graph. FP = the number of incorrectly identified edges that existed in the true graph. TN = the number of incorrectly identified lack of edges that existed in the true graph. I = the mutual information function. $\hat{\vartheta}$ = the vector of the estimated clusters for each node. ϑ = the vector of the true clusters for each node. H = the entropy function. R = the number of simulation iterations. Cor_i = a binary indicator for each i^{th} simulation indicating if the number of identified clusters was correct. \widehat{M}_j = the number of nodes estimated to be clustered together. D = the hypergeometric function. K^* = the maximum number of clusters.

Implementation of the Simulation

All simulations and data analyses were conducted in R version 4.2.0 (R Core Team, 2021). The simulations start by choosing a DGP scenario (i.e., a $3 \times 2 \times 3$ factorial design with three possible DGPs, two possible cluster sizes, and three possible sample sizes, in a total of 18 conditions) and generating data accordingly. With these data, first the nodes/variables are clustered using one of the three clustering procedures (PA-EFA, EGA, or CGMM). Then, the performance of each procedure is evaluated by all the performance indices. Finally, using the true clustering (which is known by design) of the simulated observational data, the averaged correlation matrix is used in combination with the PC-algorithm to discover the direction of the edges. The performance of the PC-algorithm is then evaluated by all the accuracy-related performance indices.

For the cluster detection part of the simulation, the PA used was the one implemented in the paran package version 1.5.2 (Dinno, 2018) and the EFA, with oblimin rotation (to allow the latent variables to be correlated), implemented in the psych package version 2.2.5 (Revelle, 2019). EGA was estimated using the implementation in the EGAnet package, version 1.2.3 (Golino et al., 2022). For CGMM, the Gaussian mixture model for clustering analysis was the one implemented in the mclust R package version 5.4.10 (Scrucca et al., 2016). The PC-algorithm used was the one implemented on the pcalg package version 2.7-6 (Kalisch et al., 2021). VI and NMI were calculated by the implementation in the fpc package version 2.2-9 (Hennig, 2018) and GARI by the implementation in the loe package version 1.1 (Terada & von Luxburg, 2016). All the other computations were made with authors' own code. The R codes with the full simulation are available at Franco et al. (2022).

The Results section is divided in three subsections. In the first subsection, we investigate the performance of clustering algorithms for finding clusters (i.e., the subgroups of undirected edges) in a PCG setting. In the second subsection, we test the performance of algorithms in finding the directed edges of the PCG. We start with the assumption that the clusters are all known so the PC-stable performance is evaluated independently from the performance of the clustering procedure. The last table in the second subsection presents the results when we also consider an experimental setting in which the clusters are not assumed to be known and were first estimated with the clustering algorithms. At last, the application of fine-tuning is illustrated to evaluate the average change of density (i.e., the quantity of arrows kept) after the application of each of the alternative algorithms.

Results

Comparison Between Clustering Procedures

Table 3 presents the results for all the 18 conditions. Bold values represent the method that had the best performance in the specific index and condition. It is possible to see

that EGA performed the best in 6 conditions, while CGMM performed the best in the other 12. The PA-EFA procedure was never the best procedure, showing performance indices considerably worse than the other two methods.

Table 3

Performance Comparison Over All Possible Combinations of Experimental Conditions

DGP	Cluster Size	Sample Size	Index	PA-EFA		EGA		CGMM	
1	5	100	GARI	0.425	(0.0001)	0.818	(0.0002)	0.927	(0.0002)
			VI	0.701	(0.0002)	0.238	(0.0003)	0.144	(0.0003)
			NMI	0.716	(0.0001)	0.918	(0.0002)	0.952	(0.0001)
			HitND	0.000	(0)	0.589	(0.0004)	0.718	(0.0005)
		250	GARI	0.442	(0.0001)	0.955	(0.0001)	0.938	(0.0001)
			VI	0.585	(0.0002)	0.061	(0.0002)	0.124	(0.0002)
			NMI	0.667	(0.0001)	0.968	(0.0001)	0.953	(0.0001)
			HitND	0.000	(0)	0.963	(0.0002)	0.750	(0.0004)
		500	GARI	0.444	(0.0001)	0.985	(0.0001)	0.950	(0.0001)
			VI	0.581	(0.0002)	0.018	(0.0001)	0.101	(0.0002)
			NMI	0.669	(0.0001)	0.990	(0.0001)	0.963	(0.0001)
			HitND	0.000	(0)	0.981	(0.0001)	0.777	(0.0004)
	10	100	GARI	0.436	(0.0001)	0.860	(0.0002)	0.873	(0.0001)
			VI	0.649	(0.0002)	0.287	(0.0003)	0.275	(0.0002)
			NMI	0.632	(0.0001)	0.870	(0.0001)	0.886	(0.0001)
			HitND	0.000	(0)	0.893	(0.0003)	0.539	(0.0005)
		250	GARI	0.439	(0.0001)	0.983	(0.0001)	0.906	(0.0001)
			VI	0.644	(0.0002)	0.027	(0.0001)	0.193	(0.0002)
			NMI	0.635	(0.0001)	0.985	(0.0001)	0.923	(0.0001)
			HitND	0.000	(0)	0.995	(0)	0.463	(0.0005)
		500	GARI	0.444	(0.0001)	0.995	(0.0001)	0.916	(0.0001)
			VI	0.624	(0.0002)	0.004	(0.0001)	0.166	(0.0002)
			NMI	0.645	(0.0001)	0.996	(0.0001)	0.933	(0.0001)
			HitND	0.000	(0)	0.997	(0.0001)	0.427	(0.0005)
2	5	100	GARI	0.414	(0.0001)	0.568	(0.0002)	0.873	(0.0001)
			VI	0.751	(0.0002)	0.550	(0.0002)	0.250	(0.0002)
			NMI	0.718	(0.0001)	0.803	(0.0001)	0.921	(0.0001)
			HitND	0.000	(0)	0.124	(0.0003)	0.713	(0.0004)
		250	GARI	0.446	(0.0001)	0.704	(0.0002)	0.982	(0)
			VI	0.687	(0.0002)	0.334	(0.0002)	0.040	(0.0001)
			NMI	0.743	(0.0001)	0.879	(0.0001)	0.988	(0)
			HitND	0.000	(0)	0.256	(0.0004)	0.929	(0.0003)
		500	GARI	0.458	(0.0001)	0.685	(0.0002)	0.996	(0)
			VI	0.656	(0.0001)	0.346	(0.0002)	0.008	(0)
			NMI	0.754	(0.0001)	0.873	(0.0001)	0.998	(0)
			HitND	0.000	(0)	0.220	(0.0004)	0.970	(0.0002)
	10	100	GARI	0.424	(0.0001)	0.801	(0.0002)	0.916	(0.0001)
			VI	0.769	(0.0002)	0.328	(0.0002)	0.216	(0.0002)
			NMI	0.713	(0.0001)	0.891	(0.0001)	0.933	(0.0001)
			HitND	0.000	(0)	0.577	(0.0005)	0.909	(0.0003)
		250	GARI	0.454	(0.0001)	0.969	(0.0001)	0.974	(0)

DGP	Cluster Size	Sample Size	Index	PA-EFA		EGA		CGMM		
		500	VI	0.697	(0.0001)	0.047	(0.0001)	0.069	(0.0001)	
			NMI	0.740	(0.0001)	0.985	(0)	0.979	(0)	
			HitND	0.000	(0)	0.909	(0.0003)	0.866	(0.0003)	
			GARI	0.464	(0)	1.000	(0)	0.991	(0)	
			VI	0.660	(0.0001)	0.000	(0)	0.021	(0.0001)	
			NMI	0.753	(0)	1.000	(0)	0.994	(0)	
			HitND	0.000	(0)	1.000	(0.0001)	0.875	(0.0003)	
3	5	100	GARI	0.348	(0.0001)	0.508	(0.0002)	0.772	(0.0001)	
			VI	0.848	(0.0001)	0.611	(0.0002)	0.349	(0.0002)	
			NMI	0.741	(0)	0.819	(0.0001)	0.904	(0.0001)	
		HitND	0.000	(0)	0.019	(0.0001)	0.229	(0.0004)		
		250	GARI	0.392	(0.0001)	0.655	(0.0002)	0.910	(0.0001)	
			VI	0.769	(0.0001)	0.401	(0.0002)	0.122	(0.0001)	
	NMI		0.766	(0)	0.884	(0)	0.967	(0)		
	500	HitND	0.000	(0)	0.045	(0.0002)	0.519	(0.0005)		
		GARI	0.409	(0.0001)	0.668	(0.0001)	0.956	(0.0001)		
		VI	0.724	(0.0001)	0.386	(0.0001)	0.056	(0.0001)		
	10	100	NMI	0.779	(0)	0.889	(0)	0.985	(0)	
			HitND	0.000	(0)	0.027	(0.0002)	0.751	(0.0004)	
			GARI	0.360	(0.0001)	0.718	(0.0001)	0.915	(0.0001)	
		250	VI	0.868	(0.0001)	0.417	(0.0002)	0.218	(0.0002)	
			NMI	0.735	(0)	0.883	(0.0001)	0.944	(0)	
			HitND	0.000	(0)	0.217	(0.0004)	0.697	(0.0005)	
		500	GARI	0.409	(0.0001)	0.907	(0.0001)	0.985	(0)	
			VI	0.776	(0.0001)	0.120	(0.0002)	0.040	(0.0001)	
			NMI	0.764	(0)	0.967	(0)	0.990	(0)	
			HitND	0.000	(0)	0.656	(0.0005)	0.966	(0.0002)	
			GARI	0.433	(0.0001)	0.981	(0.0001)	0.997	(0)	
VI			0.725	(0.0001)	0.023	(0.0001)	0.010	(0)		
				NMI	0.780	(0)	0.994	(0)	0.997	(0)
				HitND	0.000	(0)	0.916	(0.0003)	0.976	(0.0002)

Note. DGP = data generating process; PA-EFA = parallel analysis with exploratory factor analysis; EGA = exploratory graph analysis; CGMM = correlation Gaussian mixture model; GARI = graph adjusted Rand index; VI = variation of information; NMI = normalized mutual information; HitND = simulation rate of hits of number of clusters. Bold values indicate the best value per row.

Table 4 presents the results for the comparison between the clustering procedures averaged over the three different DGPs' conditions represented in Figure 5. We used bold values to emphasize the better performing procedure at each condition. It is possible to see that, overall, CGMM performed better in the most complex DGPs in almost all indices, but EGA performed better in the simplest DGP. It is also possible to see that PA-EFA and CGMM had somewhat stable performances across the different DGPs, while the performance of EGA seemed to decay with the complexity of the DGP. This pattern is also shown in Figure 6, with the size of the points proportional to the standard errors.

Table 4

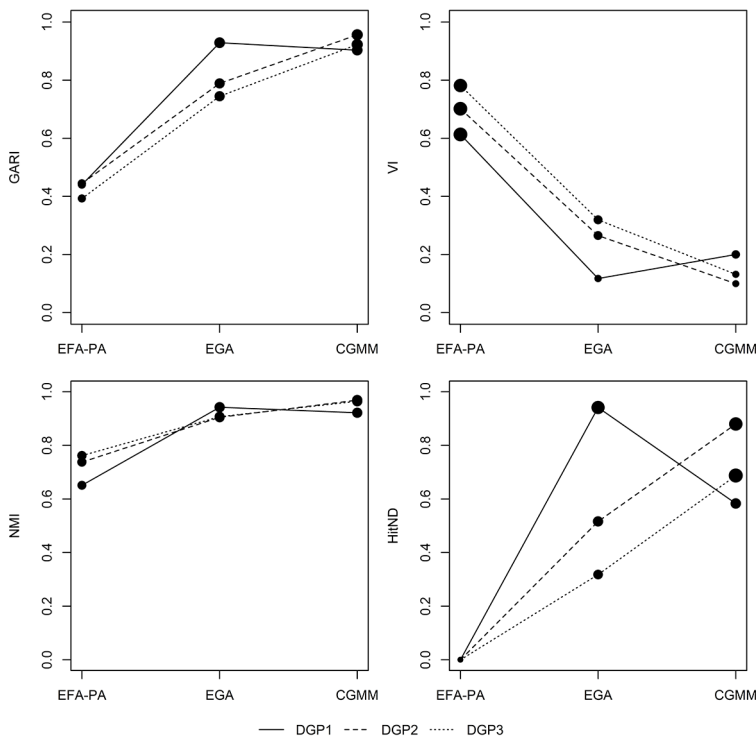
Performance Comparison Between Different DGPs

Index	DGP1			DGP2			DGP3		
	PA-EFA	EGA	CGMM	PA-EFA	EGA	CGMM	PA-EFA	EGA	CGMM
GARI	0.441	0.922	0.899	0.446	0.782	0.954	0.392	0.735	0.923
VI	0.614	0.129	0.211	0.699	0.274	0.105	0.777	0.329	0.132
NMI	0.651	0.941	0.918	0.739	0.903	0.968	0.763	0.906	0.965
HitND	0.000	0.918	0.570	0.000	0.485	0.873	0.000	0.303	0.698

Note. DGP = data generating process; PA-EFA = parallel analysis with exploratory factor analysis; EGA = exploratory graph analysis; CGMM = correlation Gaussian mixture model; GARI = graph adjusted Rand index; VI = variation of information; NMI = normalized mutual information; HitND = simulation rate of hits of number of clusters.

Figure 6

Performance Comparison Between Different DGPs



Note. DGP = data generating process; PA-EFA = parallel analysis with exploratory factor analysis; EGA = exploratory graph analysis; CGMM = correlation Gaussian mixture model; GARI = graph adjusted Rand index;

VI = variation of information; NMI = normalized mutual information; HitND = simulation rate of hits of number of clusters.

Table 5 shows the effect of the sample size (n , the number of simulated cases) on the performance of the clustering procedures. Again, CGMM had the best performance over all conditions and over all indices. PA-EFA is again the worst performer over all conditions and indices. There also seems to be an influence of the sample size on the performance of the procedures: increasing from 100 cases to 250 cases will increase the performance notably. Increasing the sample from 250 to 500 also increases the performance, but by a lesser magnitude. These results are also shown in Figure 7.

Table 5

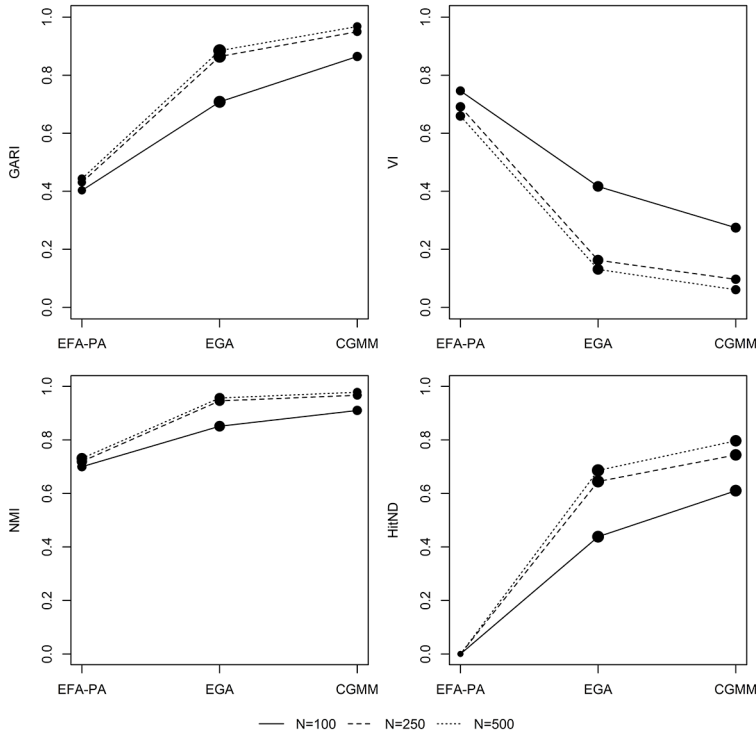
Performance Comparison Between Different Sample Sizes

Index	$n = 100$			$n = 250$			$n = 500$		
	PA-EFA	EGA	CGMM	PA-EFA	EGA	CGMM	PA-EFA	EGA	CGMM
GARI	0.404	0.696	0.859	0.431	0.862	0.948	0.443	0.881	0.969
VI	0.745	0.428	0.287	0.684	0.168	0.101	0.662	0.137	0.059
NMI	0.701	0.850	0.906	0.722	0.945	0.965	0.730	0.955	0.979
HitND	0.000	0.398	0.607	0.000	0.630	0.733	0.000	0.678	0.802

Notes: PA-EFA = parallel analysis with exploratory factor analysis; EGA = exploratory graph analysis; CGMM = correlation Gaussian mixture model; GARI = graph adjusted Rand index; VI = variation of information; NMI = normalized mutual information; HitND = simulation rate of hits of number of clusters.

Figure 7

Performance Comparison Between Different Sample Sizes



Notes: N = sample size; PA-EFA = parallel analysis with exploratory factor analysis; EGA = exploratory graph analysis; CGMM = correlation Gaussian mixture model; GARI = graph adjusted Rand index; VI = variation of information; NMI = normalized mutual information; HitND = simulation rate of hits of number of clusters.

For evaluating the influence of the clusters' sizes (v , the number of nodes per cluster), results in Table 6 show, again, that overall CGMM outperforms the other procedures. But this time, EGA has the largest power (HitND) when the cluster size is equal to 10 variables per cluster. PA-EFA is, once more, the worst performing procedure in all the cases. Finally, increasing the cluster size will, in general, increase the performance of EGA and CGMM, but will decrease the performance of PA-EFA when measured by NMI. These results are also represented in Figure 8.

Table 6

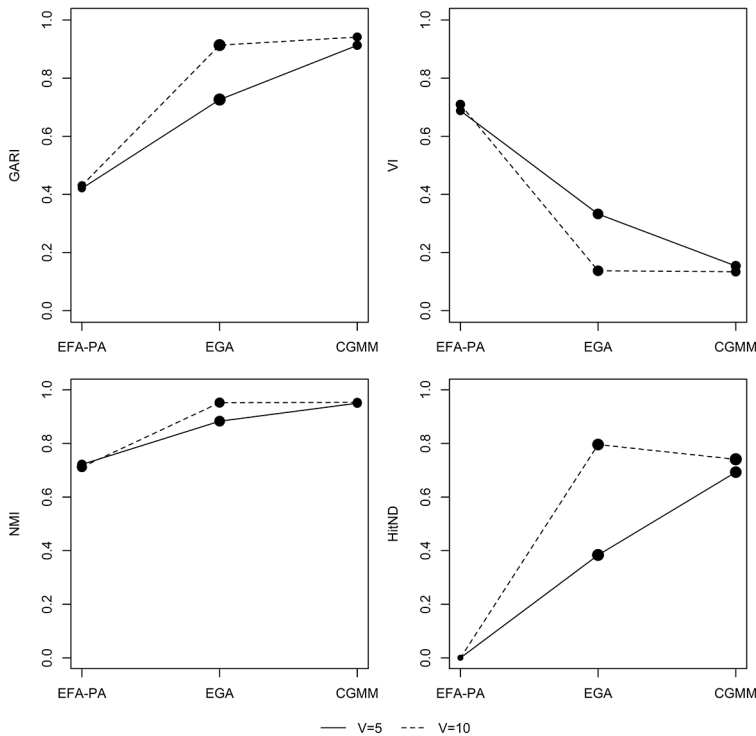
Performance Comparison Between Different Cluster Sizes

Index	v = 5			v = 10		
	PA-EFA	EGA	CGMM	PA-EFA	EGA	CGMM
GARI	0.420	0.715	0.911	0.432	0.910	0.940
VI	0.688	0.346	0.160	0.706	0.142	0.138
NMI	0.722	0.881	0.949	0.713	0.951	0.952
HitND	0.000	0.356	0.703	0.000	0.782	0.724

Note. PA-EFA = parallel analysis with exploratory factor analysis; EGA = exploratory graph analysis; CGMM = correlation Gaussian mixture model; GARI = graph adjusted Rand index; VI = variation of information; NMI = normalized mutual information; HitND = simulation rate of hits of number of clusters.

Figure 8

Performance Comparison Between Different Cluster Sizes



Note. V = cluster size; PA-EFA = parallel analysis with exploratory factor analysis; EGA = exploratory graph analysis; CGMM = correlation Gaussian mixture model; GARI = graph adjusted Rand index; VI = variation of information; NMI = normalized mutual information; HitND = simulation rate of hits of number of clusters.

Finally, analyses of variance were conducted in order to assess how each condition affect the performance of the methods. We report only the partial eta squared effect size in Table 7, as our aim with this analysis is only to verify the tendency of effect of the conditions on the performance of each procedure. As the independent variables, we included the sample size (SS), the cluster size (CS), the data generating process (DGP), and all the higher order interactions. The results show that, in general, the conditions had only very small effects on the methods' performance. Only the DGP affected the performance in some of the indices, and only for the EGA and the CGMM.

Table 7

ANOVA's Partial Eta Squared Effect Sizes for Clustering Procedures

Index	Factor	PA-EFA	EGA	CGMM
GARI	SS	0.0000	0.0003	0.0001
	CS	0.0000	0.0004	0.0000
	DGP	0.0001	0.0014	0.0044
	SS:CS	0.0000	0.0001	0.0000
	SS:DGP	0.0000	0.0002	0.0006
	CS:DGP	0.0000	0.0002	0.0005
	SS:CS:DGP	0.0000	0.0002	0.0003
HitND	SS	0.0000	0.0005	0.0000
	CS	0.0000	0.0007	0.0001
	DGP	0.0002	0.0656	0.0282
	SS:CS	0.0000	0.0004	0.0001
	SS:DGP	0.0000	0.0011	0.0003
	CS:DGP	0.0000	0.0009	0.0006
	SS:CS:DGP	0.0000	0.0006	0.0005
NMI	SS	0.0000	0.0010	0.0004
	CS	0.0000	0.0023	0.0007
	DGP	0.0001	0.0473	0.0053
	SS:CS	0.0000	0.0004	0.0001
	SS:DGP	0.0000	0.0013	0.0003
	CS:DGP	0.0000	0.0017	0.0005
	SS:CS:DGP	0.0000	0.0013	0.0004
VI	SS	0.0000	0.0000	0.0000
	CS	0.0000	0.0000	0.0000
	DGP	0.0040	0.0666	0.0170
	SS:CS	0.0000	0.0001	0.0001
	SS:DGP	0.0000	0.0002	0.0002
	CS:DGP	0.0000	0.0002	0.0002
	SS:CS:DGP	0.0000	0.0001	0.0002

Note. SS = sample size; CS = cluster size; DGP = data generating process; PA-EFA = parallel analysis with exploratory factor analysis; EGA = exploratory graph analysis; CGMM = correlation Gaussian mixture model; GARI = graph adjusted Rand index; VI = variation of information; NMI = normalized mutual information; HitND = simulation rate of hits of number of clusters.

Comparison Between Structure Learning Algorithms

Using the PC-stable algorithm with the averaged correlation matrix of the clusters and assuming the true clusters are known we generated the results shown in Table 8. The overall results, averaged over all conditions and shown in Table 9, show that the PC-stable algorithm will recover most of the arrows correctly. This means that all edges that should be identified as directed was identified as such, as well as their correct directions were also identified. We see that the DGP affects the efficiency of the algorithm, as the results are almost perfect when there are only three clusters of variables with a simple v-structure between them; i.e., DGP 1.

Table 8

Performance Comparison Over All Possible Combinations of Experimental Conditions, When the Cluster Are Assumed to Be Known

DGP	Cluster Size	Sample Size	Acc	PPV	TPR	TNR	
1	5	100	0.982 (0.0001)	0.988 (0)	0.972 (0.0001)	0.993 (0)	
		250	1.000 (0)	1.000 (0)	1.000 (0)	1.000 (0)	
		500	1.000 (0)	1.000 (0)	1.000 (0)	1.000 (0)	
	10	100	0.994 (0)	0.996 (0)	0.991 (0.0001)	0.997 (0)	
		250	1.000 (0)	1.000 (0)	1.000 (0)	1.000 (0)	
		500	1.000 (0)	1.000 (0)	1.000 (0)	1.000 (0)	
	2	5	100	0.934 (0.0001)	0.964 (0)	0.899 (0.0001)	0.970 (0)
			250	0.982 (0)	0.966 (0)	1.000 (0)	0.964 (0)
			500	0.980 (0)	0.962 (0)	1.000 (0)	0.960 (0)
10		100	0.957 (0.0001)	0.973 (0)	0.938 (0.0001)	0.976 (0)	
		250	0.984 (0)	0.970 (0)	1.000 (0)	0.968 (0)	
		500	0.983 (0)	0.968 (0)	1.000 (0)	0.967 (0)	
3	5	100	0.882 (0.0001)	0.965 (0)	0.791 (0.0002)	0.974 (0)	
		250	0.980 (0)	0.978 (0)	0.982 (0.0001)	0.978 (0)	
		500	0.984 (0)	0.978 (0)	0.990 (0)	0.978 (0)	
	10	100	0.902 (0.0001)	0.972 (0)	0.825 (0.0001)	0.978 (0)	
		250	0.981 (0)	0.985 (0)	0.977 (0.0001)	0.985 (0)	
		500	0.983 (0)	0.987 (0)	0.979 (0.0001)	0.987 (0)	

Note. DGP = data generating process; Acc = accuracy; PPV = positive predictive value; TPR = true positive rate; TNR = true negative rate.

Table 9

Performance of the PC-Stable Algorithm Applied to Learning the Power Arrows of the PCC

Index	Overall	Data Generating Process			Sample Size			Cluster size	
		DGP1	DGP2	DGP3	100	250	500	5	10
Acc	.962	0.997	0.945	0.943	0.937	0.973	0.975	0.956	0.968
PPV	.965	1.000	0.928	0.967	0.976	0.959	0.960	0.959	0.971
TPR	.961	0.993	0.970	0.919	0.898	0.992	0.994	0.955	0.967
TNR	.962	1.000	0.921	0.967	0.977	0.955	0.955	0.957	0.968

Note. DGP = data generating process; Acc = accuracy; PPV = positive predictive value; TPR = true positive rate; TNR = true negative rate.

Regarding the effects of the sample size, the results are somewhat ambiguous: both the conditions with the largest sample and the condition with the smallest sample outperform the other in two indices. We also found that increasing the cluster size will improve the average performance of the algorithm.

Analyses of variance were conducted in order to assess how each condition affect the performance of the PC-stable algorithm when the true clusters are known. Once again, we report only the partial eta squared effect size in Table 10, and the independent variables were the sample size (SS), the cluster size (CS), the data generating process (DGP), and all the higher order interactions. The results show that, again, the DGP affected the performance of the procedure, with larger effect sizes. The sample size and the two-way interaction between the sample size and the DGP also affected the performances, but less consistently.

Table 10

ANOVA's Partial Eta Squared Effect Sizes for Causal Discovering With Known True Clusters

Factor	Acc	PPV	TPR	TNR
SS	0.1896	0.0155	0.2315	0.0006
CS	0.0057	0.0133	0.0031	0.0150
DGP	0.1395	0.2114	0.1222	0.2948
SS:CS	0.0078	0.0016	0.0099	0.0005
SS:DGP	0.1051	0.0158	0.1287	0.0250
CS:DGP	0.0007	0.0021	0.0004	0.0035
SS:CS:DGP	0.0004	0.0011	0.0010	0.0021

Note. SS = sample size; CS = cluster size; DGP = data generating process; Acc = accuracy; PPV = positive predictive value; TPR = true positive rate; TNR = true negative rate.

We performed the same simulation again, but this time without assuming the true clusters are known. After applying the PC-stable algorithm, we recovered the adjacency matrix for the induced chain graphs. From those, we calculated the accuracy for all the conditions, as shown in Table 11.

Table 11

Performance Comparison Over All Possible Combinations of Experimental Conditions, When the Cluster Are Estimated

DGP	Cluster Size	Sample Size	Cluster Procedure	Acc	PPV	TPR	TNR
1	5	100	CGMM	0.861 (0.0001)	0.950 (0.0001)	0.751 (0.0003)	0.971 (0)
			EGA	0.844 (0.0002)	0.925 (0.0001)	0.734 (0.0003)	0.953 (0.0001)
		250	CGMM	0.954 (0.0001)	0.978 (0)	0.925 (0.0002)	0.982 (0)
			EGA	0.975 (0.0001)	0.983 (0.0001)	0.965 (0.0001)	0.985 (0.0001)
		500	CGMM	0.969 (0.0001)	0.985 (0)	0.951 (0.0002)	0.987 (0)
			EGA	0.990 (0.0001)	0.992 (0.0001)	0.987 (0.0001)	0.993 (0.0001)
	10	100	CGMM	0.899 (0.0001)	0.961 (0.0001)	0.825 (0.0002)	0.974 (0)
			EGA	0.918 (0.0001)	0.954 (0.0001)	0.869 (0.0002)	0.966 (0.0001)
		250	CGMM	0.920 (0.0001)	0.946 (0.0001)	0.883 (0.0002)	0.957 (0)
			EGA	0.995 (0)	0.997 (0)	0.993 (0)	0.997 (0)
		500	CGMM	0.954 (0.0001)	0.954 (0.0001)	0.952 (0.0001)	0.957 (0)
			EGA	1.000 (0)	1.000 (0)	1.000 (0)	1.000 (0)
2	5	100	CGMM	0.847 (0.0001)	0.929 (0.0001)	0.743 (0.0002)	0.950 (0)
			EGA	0.605 (0.0001)	0.793 (0.0002)	0.269 (0.0003)	0.941 (0.0001)
		250	CGMM	0.971 (0)	0.963 (0)	0.979 (0.0001)	0.962 (0)
			EGA	0.681 (0.0002)	0.678 (0.0003)	0.495 (0.0003)	0.867 (0.0001)
		500	CGMM	0.978 (0)	0.961 (0)	0.996 (0)	0.959 (0)
			EGA	0.639 (0.0002)	0.518 (0.0004)	0.446 (0.0004)	0.832 (0.0001)
	10	100	CGMM	0.900 (0.0001)	0.950 (0)	0.841 (0.0001)	0.959 (0)
			EGA	0.759 (0.0001)	0.887 (0.0001)	0.586 (0.0003)	0.931 (0.0001)
		250	CGMM	0.968 (0)	0.966 (0)	0.971 (0.0001)	0.966 (0)
			EGA	0.909 (0.0001)	0.909 (0.0001)	0.890 (0.0002)	0.927 (0.0001)
		500	CGMM	0.978 (0)	0.968 (0)	0.989 (0)	0.967 (0)
			EGA	0.928 (0.0001)	0.919 (0.0001)	0.927 (0.0002)	0.929 (0.0001)
3	5	100	CGMM	0.743 (0.0001)	0.874 (0.0001)	0.553 (0.0002)	0.934 (0)
			EGA	0.598 (0.0001)	0.767 (0.0002)	0.272 (0.0002)	0.925 (0.0001)
		250	CGMM	0.907 (0.0001)	0.940 (0)	0.865 (0.0001)	0.949 (0)
			EGA	0.666 (0.0001)	0.746 (0.0002)	0.460 (0.0002)	0.872 (0.0001)
		500	CGMM	0.955 (0.0001)	0.962 (0)	0.946 (0.0001)	0.963 (0)
			EGA	0.666 (0.0001)	0.738 (0.0001)	0.486 (0.0002)	0.845 (0)
	10	100	CGMM	0.851 (0.0001)	0.953 (0)	0.735 (0.0002)	0.967 (0)
			EGA	0.683 (0.0001)	0.859 (0.0001)	0.431 (0.0002)	0.936 (0)
		250	CGMM	0.970 (0)	0.981 (0)	0.959 (0.0001)	0.982 (0)
			EGA	0.812 (0.0001)	0.879 (0.0001)	0.707 (0.0002)	0.916 (0)
		500	CGMM	0.981 (0)	0.987 (0)	0.976 (0.0001)	0.987 (0)
			EGA	0.848 (0.0001)	0.900 (0.0001)	0.772 (0.0002)	0.924 (0)

Note. EGA = exploratory graph analysis; CGMM = correlation Gaussian mixture model; DGP = data generating process; Acc = accuracy; PPV = positive predictive value; TPR = true positive rate; TNR = true negative rate.

Because PA-EFA had the worst performance in the previous analyses, it was not included in this analysis. As it was expected due to the results shown in Table 4, EGA performed better than CGMM in all the DGP1 conditions. However, CGMM performed better in all the other conditions.

The results shown in Table 12 demonstrate that, in average, the CGMM performs better than the EGA procedure. Increasing both the cluster (averaged over sample sizes) and the sample (averaged over cluster sizes) size seems to improve the performance of both algorithms. However, caution should be taken as the cluster size increases, given that the sparsity of the chain graph also increases, which can result in overestimated effects.

Table 12

Accuracy of the PC-Stable Algorithm Applied to Learning the Power Arrows of the PCG When the True Clusters Are not Known, but Estimated With CGMM or EGA

Sample	Index	EGA	CGMM
Full Sample	Acc	0.816	0.925
	PPV	0.865	0.958
	TPR	0.699	0.884
	TNR	0.933	0.966
N = 100	Acc	0.734	0.850
	PPV	0.864	0.936
	TPR	0.527	0.741
	TNR	0.942	0.959
N = 250	Acc	0.840	0.948
	PPV	0.866	0.962
	TPR	0.752	0.930
	TNR	0.927	0.966
N = 500	Acc	0.845	0.969
	PPV	0.844	0.969
	TPR	0.770	0.969
	TNR	0.921	0.970
V = 5	Acc	0.740	0.909
	PPV	0.793	0.949
	TPR	0.568	0.857
	TNR	0.912	0.962
V = 10	Acc	0.872	0.936
	PPV	0.923	0.963
	TPR	0.797	0.903
	TNR	0.947	0.968
DGP = 1	Acc	0.959	0.932
	PPV	0.978	0.966
	TPR	0.934	0.891

Sample	Index	EGA	CGMM
DGP = 2	TNR	0.984	0.973
	Acc	0.753	0.940
	PPV	0.784	0.956
	TPR	0.602	0.920
DGP = 3	TNR	0.905	0.960
	Acc	0.712	0.901
	PPV	0.815	0.949
	TPR	0.521	0.839
	TNR	0.903	0.964

Note. EGA = exploratory graph analysis; CGMM = correlation Gaussian mixture model; DGP = data generating process; Acc = accuracy; PPV = positive predictive value; TPR = true positive rate; TNR = true negative rate.

We conducted the final analyses of variance in order to assess how each condition affect the performance of the PC-stable algorithm when the true clusters are unknown. As usual, we report only the partial eta squared effect size in Table 13, and the independent variables were the sample size (SS), the cluster size (CS), the data generating process (DGP), and all the higher order interactions. The results show that the CGMM is strongly affected by the sample size. The EGA, on the other hand, seems to be affect by the sample size, the cluster size, and the DGP. The effects of the higher order interactions are also somewhat relevant, but comparably less meaningful.

Table 13

ANOVA's Partial Eta Squared Effect Sizes for Causal Discovering With Unknown True Clusters

Method	Factor	Acc	PPV	TPR	TNR
CGMM	SS	0.2995	0.0706	0.3311	0.0189
	CS	0.0270	0.0172	0.0265	0.0090
	DGP	0.0393	0.0105	0.0517	0.0188
	SS:CS	0.0308	0.0250	0.0310	0.0090
	SS:DGP	0.0457	0.0282	0.0444	0.0187
	CS:DGP	0.0324	0.0634	0.0192	0.0814
	SS:CS:DGP	0.0037	0.0052	0.0043	0.0094
EGA	SS	0.1408	0.0059	0.1938	0.0232
	CS	0.2252	0.1534	0.2131	0.0863
	DGP	0.4215	0.2356	0.3810	0.2937
	SS:CS	0.0086	0.0221	0.0028	0.0385
	SS:DGP	0.0017	0.0382	0.0069	0.0846
	CS:DGP	0.0907	0.0886	0.0905	0.0223
	SS:CS:DGP	0.0206	0.0303	0.0143	0.0271

Note. SS = sample size; CS = cluster size; DGP = data generating process; Acc = accuracy; PPV = positive predictive value; TPR = true positive rate; TNR = true negative rate.

Network Density After Fine-Tuning

The final analysis was used to evaluate how applying “fine-tuning” causal discovery algorithms would change the total number of arrows in the CG implied by the PCG. Table 14 shows our findings in terms of the sparsity of the CGs discovered by the algorithms, where sparsity is simply the false negative rates (i.e., $1 - \text{TPR}$). The first algorithm column, named CG, is simply the CG implied by the PCG. Because of how our simulations were conducted, this column should always be the one with the lowest sparsity and, therefore, it is used as a benchmark for the other procedures. It is possible to see that the HC algorithm is always the one that keeps most of the arrows, followed by the PC-stable algorithm and finally, the MMHC algorithm. No best–worst comparison is adequate in this case. However, these results show that further tuning of the CG implied by the PCG will be considerably sparser and, therefore, causal analysis of CGs estimated by these procedures will need to be sensible to causal assumptions.

Table 14

Sparsity (1-TPR) Comparison Between Tuning Algorithms

Condition	Algorithms			
	CG	PC	HC	MMHC
Overall	0.039	0.818	0.599	0.928
DGP				
DGP1	0.007	0.794	0.584	0.920
DGP2	0.030	0.817	0.603	0.926
DGP3	0.081	0.842	0.611	0.938
Sample size				
100	0.103	0.913	0.744	0.959
250	0.008	0.820	0.592	0.928
500	0.006	0.721	0.463	0.897
Cluster size				
5	0.045	0.754	0.505	0.893
10	0.033	0.882	0.694	0.963

Note. CG = chain graph; PC = PC-stable algorithm; HC = hill-climbing algorithm; MMHC = min-max hill-climbing algorithm; DGP = data generating process.

Empirical Example

For illustration purposes, the present empirical example will apply PCG to derive causal hypotheses related to the empathy construct. Davis (1980) proposed a multidimensional measurement of empathy, operationalized in a questionnaire composed by four seven-

item subscales: perspective-taking; fantasy; empathic concern; and personal distress. Fantasy is the tendency to get involved in the actions of fictional characters from diverse media. Perspective-taking is the tendency to comprehend others' point of view. Empathic concern is the tendency of feeling concern and sympathy for people in distress. Personal distress is the tendency of feeling unease in difficult, tense or emotional situations.

As usual in the psychological literature, the psychometric properties of the questionnaire were evaluated by [Davis \(1980\)](#) by means of exploratory factor analysis, without explicit assumptions regarding causal relations between the latent factors. This analysis resulted in what is usually considered as good evidence of validity (i.e., items theorized to be together were indeed clustered by the same latent factor) and reliability (i.e., Cronbach's alpha above 0.70). [Briganti et al. \(2018\)](#) applied the EGA procedure and found the same structure as the original study by [Davis \(1980\)](#). In the present example, we will apply our PCG procedure to the open dataset, originally from [Braun et al. \(2015\)](#), provided by [Briganti et al. \(2018\)](#).

Method

Participants

The dataset is composed of 1,973 French-speaking students from different colleges, from a diverse set of courses. The age of the participants ranged from 17 to 25 years ($M = 19.6$ years, $SD = 1.6$ years), with 57% of these participants identifying themselves as females and the other 43% as males. From the original sample of 1,973 individuals, only 1,270 answered all the items of the questionnaire. Missing data was imputed by [Briganti et al. \(2018\)](#) using a maximum likelihood method based on the Gaussian graphical model that they fitted to the data. The items of the empathy measurement are displayed in [Table 15](#), as well as the factors identified by the EGA procedure.

Analysis

Because the structure of the instrument was validated extensively in the literature, we proceeded by assuming that the true clusters are known. In the next step, the averaged correlation matrix was calculated and then the PC-stable algorithm was applied for discovering the directions of the relations in the PCG. Finally, the tuning algorithms were applied to decrease the density of the CG implied by the PCG, with the results displayed both graphically and with the percentage of removed relations textually described. The R code for doing these analyses is also included in the [Supplementary Material](#).

Results

We calculated the averaged correlation matrix based on the clusters identified by [Briganti et al. \(2018\)](#). Then, we applied the PC-stable algorithm to this averaged correlation matrix, resulting in the PCG shown to the left of [Figure 9](#). It is possible to see that

all the edges are directed, indicating that the PC-stable found a graph that is unique in its equivalence class. From a theoretical perspective, it is possible to infer that Personal distress (V'1) is the cause for both Empathic concern (V'4) and Fantasy (V'2). It is possible to infer as well that Perspective-taking (V'3) causes Empathic concerns which in turn causes Fantasy. The chain graph implied by the PCG is shown to the right of Figure 9. In this graph, the edges were weighted by the regularized partial correlation estimated with the EBICglasso procedure.

Table 15

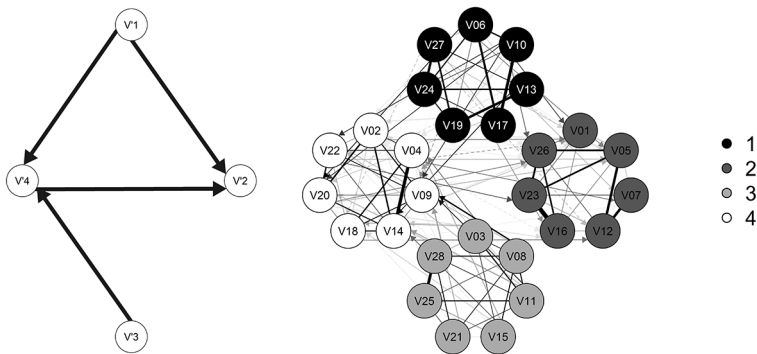
Items' Descriptions and Original (Davis, 1980) Factors' Assignment

Item	Factor	Item description
1	Fantasy	I daydream and fantasize, with some regularity, about things that might happen to me.
2	Empathic concern	I often have tender, concerned feelings for people less fortunate than me.
3R	Perspective-taking	I sometimes find it difficult to see things from the "other guy's" point of view.
4R	Empathic concern	Sometimes I don't feel very sorry for other people when they are having problems.
5	Fantasy	I really get involved with the feelings of the characters in a novel.
6	Personal distress	In emergency situations, I feel apprehensive and ill-at-ease.
7R	Fantasy	I am usually objective when I watch a movie or play, and I don't often get completely caught up in it.
8	Perspective-taking	I try to look at everybody's side of a disagreement before I make a decision.
9	Empathic concern	When I see someone being taken advantage of, I feel kind of protective towards them.
10	Personal distress	I sometimes feel helpless when I am in the middle of a very emotional situation.
11	Perspective-taking	I sometimes try to understand my friends better by imagining how things look from their perspective.
12R	Fantasy	Becoming extremely involved in a good book or movie is somewhat rare for me.
13R	Personal distress	When I see someone get hurt, I tend to remain calm.
14R	Empathic concern	Other people's misfortunes do not usually disturb me a great deal.
15R	Perspective-taking	If I'm sure I'm right about something, I don't waste much time listening to other people's arguments.
16	Fantasy	After seeing a play or movie, I have felt as though I were one of the characters.
17	Personal distress	Being in a tense emotional situation scares me.
18R	Empathic concern	When I see someone being treated unfairly, I sometimes don't feel very much pity for them.
19R	Personal distress	I am usually pretty effective in dealing with emergencies.
20	Fantasy	I am often quite touched by things that I see happen.
21	Perspective-taking	I believe that there are two sides to every question and try to look at them both.
22	Empathic concern	I would describe myself as a pretty soft-hearted person.
23	Fantasy	When I watch a good movie, I can very easily put myself in the place of a leading character.
24	Personal distress	I tend to lose control during emergencies.
25	Perspective-taking	When I'm upset at someone, I usually try to "put myself in his shoes" for a while.
26	Fantasy	When I am reading an interesting story or novel, I imagine how I would feel if the events in the story were happening to me.
27	Personal distress	When I see someone who badly needs help in an emergency, I go to pieces.
28	Perspective-taking	Before criticizing somebody, I try to imagine how I would feel if I were in their place.

Note. An R after the number of the item indicates that it is a reversed item.

Figure 9

Estimated Power Chain Graph (on the Left) and the Implied Chain Graph (on the Right)

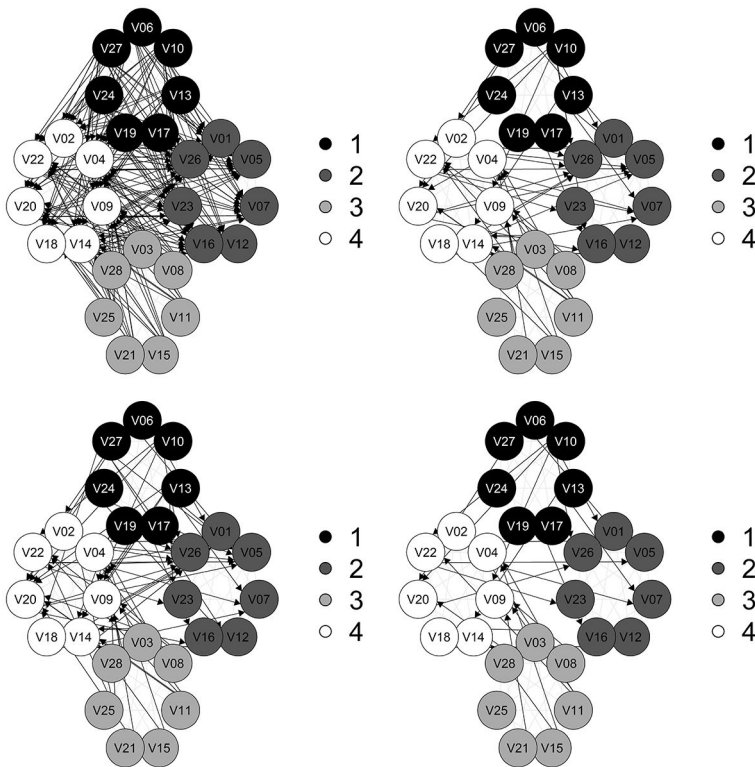


Note. $V'1$ = First power node, Personal distress. $V'2$ = Second power node, Fantasy. $V'3$ = Third power node, Perspective-taking. $V'4$ = Fourth power node, Empathic concern. $V01$ to $V28$ are the 28 variables according to Table 15.

For the final analysis, we proceeded to the fine-tuning procedure with the CG implied by the PCG. For doing that, first, we fixed the direction of the edges according to the estimated PCG. We also fixed the absence of relations between power nodes (e.g., no direct relations between the variables in $V'3$ and $V'2$). After that, the PC-stable, the HC, and the MMHC algorithms were applied. Fine-tuning with the PC-stable algorithm resulted in 74.49% of the edges between clusters removed. With the HC algorithm, 65.31% of the edges between clusters were removed. With the MMHC algorithm, 86.22% of the edges between clusters were removed. Each corresponding unweighted CG is displayed in Figure 10. From this point on, an empathy researcher could test any of the fine-tuned models by means of path analysis, a strategy we are not approaching here as it is beyond our scope.

Figure 10

The Original CG Implied by the Estimated PCG (Top-Left), a CG Tuned by the PC-Stable Algorithm (Top-Right), a CG Tuned by the HC Algorithm (Bottom-Left) and a CG Tuned by the MMHC Algorithm (Bottom-Right)



Discussion

The main aim of the present study was to propose a procedure of structure learning of PCGs. The secondary aim of the study was to compare clustering algorithms and causal discovery algorithms that can be used for learning the structure of the PCGs. To achieve these aims we used simulated data. To evaluate the practical value of our procedure, we did a full analysis of an empirical example. Our simulations showed that our procedure can properly recover both the clusters of variables, by means of the CGMM procedure, and the correct direction of the underlying causal relations, by means of averaged correlation matrix and the PC-stable algorithm. Our simulations have also illustrated how PCGs can be used to further investigate the structure of CGs, through tuning of the CG implied by the PCG using three different classes of causal discovery algorithms.

To assess the performance of the CGMM procedure in finding the correct number of clusters, we used a number of accuracy related indices and cluster comparison indices. We also compared the CGMM performance with two other more traditional procedures in statistical analysis: PA-EFA and EGA. Our results show that, in general, CGMM outperformed both procedures in a number of different conditions of sample sizes, number of variables per cluster, and the total number of clusters. These results are somewhat unexpected. Previous studies (e.g., [Golino et al, 2020](#)) have shown high accuracy for both PA-EFA and EGA procedures. However, PA-EFA performed very poorly and EGA performed poorer than expected (despite the fact that it performed better than the CGMM in some conditions). One possible reason for this result is the differences in design between our study and the previous studies. In our study, the simulation has an underlying causal structure that forces independence between some power nodes. Previous studies used correlated underlying structures. This difference causes data to be factorized in a different manner ([Lauritzen, 1996](#)), probably resulting in the differences of performance found in our study. This design difference also means that more studies are necessary to evaluate how well CGMM will perform when there are dependencies between all power nodes.

The PC-stable algorithm performed well in a number of different conditions, even when the clustering was not known a priori. Changing the sample sizes, number of variables per cluster and the total number of clusters had little to no clear effect in the accuracy related indices. The second part of the causal discovery analysis, which compared different tuning algorithms, has some implications for structure learning procedures of CGs (e.g., [Drton & Perlman, 2008](#); [Ma et al., 2008](#); [Peña et al., 2014](#)). Most of these procedures try to find the best fitting CG by brute force ([Peña, 2018](#)): fitting a large number of different models and choosing the best fitting one. Our results indicate that learning the structure using the full PCG procedure and then using its implied CG as an initial guess for the estimated true CG can enhance CG's structure learning procedures.

Regarding the empirical example, although it was used more as an illustration of our procedure than a true investigation of the causal structures of empathy as proposed by [Davis \(1980\)](#), our findings look promising. In sum, the results indicate that it is reasonable to infer that our own feelings of uneasiness in difficult situations and our capacity to comprehend others' point of view causes how we feel for people in distress. Also, how we feel for people in distress and our own feelings of uneasiness in difficult situations causes how we feel for fictional characters in distress. These inferences may have implications that might be of interest for replication and extension from researchers of empathy. For instance, longitudinal (e.g., [Zhou et al., 2002](#)) studies could be used to test if the causal structure holds in a developmental setting. Neurological (e.g., [Singer et al., 2004](#)) studies, on the other hand, could help to validate the estimated causal structure by identifying which areas of the brain are related to which factor, as well as if neural relations reflect the causal structure found in the present study. For researchers from

other areas of study can also benefit from these results as they provide insight that maybe constructs first hypothesized to be only associational can actually present some identifiable underlying causal structure.

There are two main limitations of the present study. The first is the CGMM procedure. Despite its performance in the simulation study, the results may be limited to the particular setting of the data generating process. Theoretical studies are necessary to better understand in what conditions CGMM may perform the best so the present results regarding CGMM can be properly generalized. However, this has little impact on the PCG procedure as a whole, as any clustering procedure can be used in place of the CGMM, as long as it respects the “similarity” condition for clustering the variables. The second main limitation is related to the averaging of the correlation matrix. Despite its relation with the definitions of a power edge and similarity between nodes, there is neither logical implication nor model constraint that requires for the power edges to be estimated like this. Therefore, it is also necessary for future studies to evaluate which procedures for estimating or calculating power edges can improve the performance of the PCG procedure.

Future studies on PCG can focus on both theoretical aspects of CGMM and how causal discovery algorithms can be extended for learning the causal structure from the averaged correlation matrix. For instance, what would happen if instead of correlations, measures of dependence (Paul & Shill, 2017; Zhao et al., 2016), such as the maximal information coefficient or distance correlations, were used in CGMM? What would be the impact of using partial distance correlations’ tests (Székely & Rizzo, 2014) instead of partial correlations’ tests in the PC-stable algorithm? Is it possible to incorporate a clustering procedure such as CGMM directly as a step in the score-based algorithms? These questions help to set the environment for prolific research about PCGs and all its constituent elements: clustering; causal discovery; and CG structure learning.

Funding: Vithor Rosa Franco received a research grant from the Coordination for the Improvement of Higher Education Personnel grant number 88881.190469/2018-01.

Acknowledgments: The authors would like to thank the reviewers for their valuable comments and suggestions. We also would like to thank the editor for his decision to publish the manuscript.

Competing Interests: The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Publisher Note: Due to a technical issue in article production, all figures in the original version of this article, published December 22, 2022, were produced without graphical content. Therefore, the faulty Version of Record was replaced by the current Corrected Version of Record (CVoR) on March 8, 2023.

Data Availability: Data are freely available, see Franco, Barros, Wiberg, & Laros (2022) , and Franco, Barros, Wiberg, & Laros (2021).

Supplementary Materials

The supplementary materials provided are the data that support the findings of this study and a manuscript preprint from the first author's PhD dissertation (for access see [Index of Supplementary Materials](#) below).

Index of Supplementary Materials

Franco, V. R., Barros, G. W., Wiberg, M., & Laros, J. A. (2022). *Supplementary materials to "Chain graph reduction into power chain graphs"* [Data, R code]. OSF. <https://osf.io/e27fj/>

Franco, V. R., Barros, G. W., Wiberg, M., & Laros, J. A. (2021). *Chain graph reduction into Power Chain Graphs*. [Manuscript preprint]. PsyArXiv. <https://psyarxiv.com/rw95u/>

References

- Aref, M., Moawad, I., & Mahmoud, M. (2014). A survey on graph reduction methods and applications. *Egyptian Journal of Language Engineering*, 1(2), 50–57. <https://doi.org/10.21608/ejle.2014.60011>
- Asano, T., & Hirata, T. (1983). Edge-contraction problems. *Journal of Computer and System Sciences*, 26(2), 197–208. [https://doi.org/10.1016/0022-0000\(83\)90012-0](https://doi.org/10.1016/0022-0000(83)90012-0)
- Braun, S., Rosseel, Y., Kempnaers, C., Loas, G., & Linkowski, P. (2015). Self-report of empathy: A shortened French adaptation of the Interpersonal Reactivity Index (IRI) using two large Belgian samples. *Psychological Reports*, 117(3), 735–753. <https://doi.org/10.2466/08.02.PR0.117c23z6>

- Breiger, R. L., Boorman, S. A., & Arabie, P. (1975). An algorithm for clustering relational data with applications to social network analysis and comparison with multidimensional scaling. *Journal of Mathematical Psychology*, 12(3), 328–383. [https://doi.org/10.1016/0022-2496\(75\)90028-0](https://doi.org/10.1016/0022-2496(75)90028-0)
- Briganti, G., Kempnaers, C., Braun, S., Fried, E. I., & Linkowski, P. (2018). Network analysis of empathy items from the Interpersonal Reactivity Index in 1973 young adults. *Psychiatry Research*, 265, 87–92. <https://doi.org/10.1016/j.psychres.2018.03.082>
- Chao, G., Luo, Y., & Ding, W. (2019). Recent advances in supervised dimension reduction: A survey. *Machine Learning and Knowledge Extraction*, 1(1), 341–358. <https://doi.org/10.3390/make1010020>
- Chickering, D. M. (2003). Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3, 507–554. <https://dl.acm.org/doi/10.1162/153244303321897717>
- Colombo, D., & Maathuis, M. H. (2014). Order-independent constraint-based causal structure learning. *Journal of Machine Learning Research*, 15(116), 3921–3962. <https://www.jmlr.org/beta/papers/v15/colombo14a.html>
- Cox, D. R., & Wermuth, N. (1993). Linear dependencies represented by chain graphs. *Statistical Science*, 8(3), 204–218. <https://doi.org/10.1214/ss/1177010887>
- Daly, R., & Shen, Q. (2007). Methods to accelerate the learning of Bayesian network structures. In *Proceedings of the 2007 UK Workshop on Computational Intelligence*. <https://www.semanticscholar.org/paper/Methods-to-accelerate-the-learning-of-bayesian-Shen-Daly/e7d3029e84a1775bb12e7e67541beaf2367f7a88>
- Davis, M. H. (1980). A multidimensional approach to individual differences in empathy. *Catalog of Selected Documents in Psychology*, 10, 1–18. https://www.uv.es/friasnav/Davis_1980.pdf
- Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1), 1–22. <https://doi.org/10.1111/j.2517-6161.1977.tb01600.x>
- Dinno, A. (2018). *paran: Horn's test of principal components/factors* [Computer software]. R Foundation for Statistical Computing. <https://cran.r-project.org/web/packages/paran/index.html>
- Drton, M., & Perlman, M. D. (2008). A SINful approach to Gaussian graphical model selection. *Journal of Statistical Planning and Inference*, 138(4), 1179–1200. <https://doi.org/10.1016/j.jspi.2007.05.035>
- Epskamp, S., Waldorp, L. J., Möttus, R., & Borsboom, D. (2018). The Gaussian graphical model in cross-sectional and time-series data. *Multivariate Behavioral Research*, 53(4), 453–480. <https://doi.org/10.1080/00273171.2018.1454823>
- Fraley, C., & Raftery, A. E. (2007). Bayesian regularization for normal mixture estimation and model-based clustering. *Journal of Classification*, 24(2), 155–181. <https://doi.org/10.1007/s00357-007-0004-5>
- Franco, V. R., Barros, G. W., Wiberg, M., & Laros, J. A. (2022). *Supplemental materials for preprint: "Chain graph reduction into Power Chain Graphs"*. OSF. <https://doi.org/10.17605/OSF.IO/E27FJ>

- Friedman, N., Peér, D., & Nachman, I. (1999). Learning Bayesian network structure from massive datasets: The “sparse candidate” algorithm. In *Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence* (pp. 206–221). Morgan Kaufmann.
- Glaros, A. G., & Kline, R. B. (1988). Understanding the accuracy of tests with cutting scores: The sensitivity, specificity, and predictive value model. *Journal of Clinical Psychology*, *44*(6), 1013–1023. [https://doi.org/10.1002/1097-4679\(198811\)44:6<1013::AID-JCLP2270440627>3.0.CO;2-Z](https://doi.org/10.1002/1097-4679(198811)44:6<1013::AID-JCLP2270440627>3.0.CO;2-Z)
- Golino, H., Christensen, A., Moulder, R., Garrido, L. E., & Jamiso, L. (2022). *EGAnet: Exploratory graph analysis. A framework for estimating the number of dimensions in multivariate data using network psychometrics* [Computer software]. R Foundation for Statistical Computing. <https://cran.r-project.org/web/packages/EGAnet/index.html>
- Golino, H. F., & Epskamp, S. (2017). Exploratory graph analysis: A new approach for estimating the number of dimensions in psychological research. *PLoS One*, *12*(6), Article e0174035. <https://doi.org/10.1371/journal.pone.0174035>
- Golino, H., Shi, D., Christensen, A. P., Garrido, L. E., Nieto, M. D., Sadana, R., Thiagarajan, J. A., & Martinez-Molina, A. (2020). Investigating the performance of exploratory graph analysis and traditional techniques to identify the number of latent factors: A simulation and tutorial. *Psychological Methods*, *25*(3), 292–320. <https://doi.org/10.1037/met0000255>
- Goretzko, D., Pham, T. T. H., & Bühner, M. (2021). Exploratory factor analysis: Current use, methodological developments and recommendations for good practice. *Current Psychology*, *40*(7), 3510–3521. <https://doi.org/10.1007/s12144-019-00300-2>
- Haghighat, M. B. A., Aghagolzadeh, A., & Seyedarabi, H. (2011). A non-reference image fusion metric based on mutual information of image features. *Computers & Electrical Engineering*, *37*(5), 744–756. <https://doi.org/10.1016/j.compeleceng.2011.07.012>
- Harary, F., Norman, R. Z., & Cartwright, D. (1965). *Structural models: An introduction to the theory of directed graphs*. Wiley.
- Hennig, C. (2018). *fpc: Flexible procedures for clustering* [Computer software]. R Foundation for Statistical Computing. <https://cran.r-project.org/web/packages/fpc/index.html>
- Hennig, C., & Liao, T. F. (2013). How to find an appropriate clustering for mixed-type variables with application to socio-economic stratification. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, *62*(3), 309–369. <https://doi.org/10.1111/j.1467-9876.2012.01066.x>
- Holland, P. W., & Leinhardt, S. (1976). Local structure in social networks. *Sociological Methodology*, *7*, 1–45. <https://doi.org/10.2307/270703>
- Horn, J. L. (1965). A rationale and test for the number of factors in factor analysis. *Psychometrika*, *30*(2), 179–185. <https://doi.org/10.1007/BF02289447>
- Højsgaard, S., Edwards, D., & Lauritzen, S. (2012). *Graphical models with R*. Springer.
- Hu, J., Li, B., & Kihara, D. (2005). Limitations and potentials of current motif discovery algorithms. *Nucleic Acids Research*, *33*(15), 4899–4913. <https://doi.org/10.1093/nar/gki791>
- Javed, M. A., Younis, M. S., Latif, S., Qadir, J., & Baig, A. (2018). Community detection in networks: A multidisciplinary review. *Journal of Network and Computer Applications*, *108*, 87–111. <https://doi.org/10.1016/j.jnca.2018.02.011>

- Javidian, M. A., & Valtorta, M. (2018). *On the properties of MVR chain graphs*. arXiv. <https://arxiv.org/abs/1803.04262>
- Javidian, M. A., & Valtorta, M. (2021). A decomposition-based algorithm for learning the structure of multivariate regression chain graphs. *International Journal of Approximate Reasoning*, 136(September), 66–85. <https://doi.org/10.1016/j.ijar.2021.05.005>
- Kalisch, M., Hauser, A., & Maechler, M. (2021). pcalg: Methods for graphical models and causal inference. R Foundation for Statistical Computing. <https://CRAN.R-project.org/package=pcalg>
- Koller, D., Friedman, N., & Bach, F. (2009). *Probabilistic graphical models: Principles and techniques*. MIT Press.
- Lauritzen, S. L. (1996). *Graphical models*. Clarendon Press.
- Lauritzen, S. L., & Richardson, T. (2002). Chain graph models and their causal interpretations. *Journal of the Royal Statistical Society. Series B (Methodological)*, 64, 321–348. <https://doi.org/10.1111/1467-9868.00340>
- Lovász, L. (2006). Graph minor theory. *Bulletin of the American Mathematical Society*, 43(1), 75–86. <https://doi.org/10.1090/S0273-0979-05-01088-8>
- Ma, Z., Xie, X., & Geng, Z. (2008). Structural learning of chain graphs via decomposition. *Journal of Machine Learning Research*, 9(95), 2847–2880. <https://www.jmlr.org/beta/papers/v9/ma08a.html>
- Mair, P. (2018). *Modern psychometrics with R*. Springer International Publishing.
- Malinsky, D., & Danks, D. (2018). Causal discovery algorithms: A practical guide. *Philosophy Compass*, 13(1), Article e12470. <https://doi.org/10.1111/phc3.12470>
- Meilă, M. (2007). Comparing clusterings: An information based distance. *Journal of Multivariate Analysis*, 98(5), 873–895. <https://doi.org/10.1016/j.jmva.2006.11.013>
- Paul, A. K., & Shill, P. C. (2017). Reconstruction of gene network through backward elimination based information-theoretic inference with maximal information coefficient. In *2017 IEEE International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, (pp. 1–5). IEEE.
- Pearl, J. (2009). *Causality*. Cambridge University Press.
- Peña, J. M. (2018). *Unifying Gaussian LWF and AMP chain graphs to model interference*. arXiv. <https://arxiv.org/abs/1811.04477>
- Peña, J. M., & Gómez-Olmedo, M. (2016). Learning marginal AMP chain graphs under faithfulness revisited. *International Journal of Approximate Reasoning*, 68, 108–126. <https://doi.org/10.1016/j.ijar.2015.09.004>
- Peña, J., Sonntag, D., & Nielsen, J. (2014). An inclusion optimal algorithm for chain graph structure learning. In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics. PMLR*, 33, 778–786. <https://proceedings.mlr.press/v33/pena14.html>
- R Core Team. (2021). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. <http://www.R-project.org/>
- Rao, D. C. (1979). Joint distribution of z transformations estimated from the same sample. *Human Heredity*, 29(6), 334–336. <https://doi.org/10.1159/000153068>

- Revelle, W. (2019). psych: Procedures for psychological, psychometric, and personality research. R Foundation for Statistical Computing.
<https://cran.r-project.org/web/packages/psych/index.html>
- Roberts, S., & Everson, R. (2001). *Independent component analysis: Principles and practice*. Cambridge University Press.
- Rohrer, J. M. (2018). Thinking clearly about correlations and causation: Graphical causal models for observational data. *Advances in Methods and Practices in Psychological Science*, 1(1), 27–42.
<https://doi.org/10.1177/2515245917745629>
- Royer, L., Reimann, M., Andreopoulos, B., & Schroeder, M. (2008). Unraveling protein networks with power graph analysis. *PLoS Computational Biology*, 4(7), Article e1000108.
<https://doi.org/10.1371/journal.pcbi.1000108>
- Russell, S. J., & Norvig, P. (2009). *Artificial intelligence: A modern approach*. Prentice Hall.
- Scrucca, L., Fop, M., Murphy, T. B., & Raftery, A. E. (2016). mclust 5: Clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal*, 8(1), 289–317.
<https://doi.org/10.32614/RJ-2016-021>
- Scutari, M., & Denis, J. B. (2021). *Bayesian networks: With examples in R*. CRC Press.
- Shah, P., Patel, M., Shah, K. H., Kagathara, V., & Sujata, Z. (2017). Dimensionality reduction: A brief survey on its myriad techniques. *Research & Reviews: Journal of Space Science & Technology*, 6(2), 46–51. <https://sciencejournals.stmjournals.in/index.php/RRJoSST/article/view/2003>
- Singer, T., Seymour, B., O’doherly, J., Kaube, H., Dolan, R. J., & Frith, C. D. (2004). Empathy for pain involves the affective but not sensory components of pain. *Science*, 303(5661), 1157–1162.
<https://doi.org/10.1126/science.1093535>
- Sonntag, D. (2016) *Chain graphs: Interpretations, expressiveness and learning algorithms* [Doctoral dissertation, Linköping University]. Linköping University Electronic Press.
- Sonntag, D., & Peña, J. M. (2015). Chain graph interpretations and their relations revisited. *International Journal of Approximate Reasoning*, 58, 39–56.
<https://doi.org/10.1016/j.ijar.2014.12.001>
- Spirtes, P., Glymour, C., & Scheines, R. (2000). *Causation, prediction, and search*. MIT Press.
- Székely, G. J., & Rizzo, M. L. (2014). Partial distance correlation with methods for dissimilarities. *Annals of Statistics*, 42(6), 2382–2412. <https://doi.org/10.1214/14-AOS1255>
- Terada, Y., & von Luxburg, U. (2014). Local ordinal embedding. In *International Conference on Machine Learning (ICML)*, PMLR 32(2), 847–855.
<https://proceedings.mlr.press/v32/terada14.html>
- Terada, Y., & von Luxburg, U. (2016). loe: Local ordinal embedding. R Foundation for Statistical Computing. <https://cran.r-project.org/web/packages/loe/index.html>
- Tsamardinos, I., Aliferis, C. F., & Statnikov, A. (2003). Time and sample efficient discovery of Markov blankets and direct causal relations. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 673–678). ACM.
<https://doi.org/10.1145/956750.956838>

- Tsamardinos, I., Brown, L. E., & Aliferis, C. F. (2006). The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, *65*(1), 31–78. <https://doi.org/10.1007/s10994-006-6889-7>
- Uhler, C. (2017). Gaussian graphical models: An algebraic and geometric perspective. arXiv. <https://doi.org/10.48550/arXiv.1707.04345>
- Verma, T., & Pearl, J. (1988). Causal networks: Semantics and expressiveness. In R. Schachter, T. S. Levitt, & L. N., Kanal (Eds.), *Uncertainty in artificial intelligence* (pp. 69–76). Elsevier.
- Verma, T. S., & Pearl, J. (1990). Equivalence and synthesis of causal models. *Proceedings of the Conference on Uncertainty in Artificial Intelligence* (pp. 220–227). <https://scirp.org/reference/referencespapers.aspx?referenceid=1691254>
- Wermuth, N., & Lauritzen, S. L. (1990). On substantive research hypotheses, conditional independence graphs and graphical chain models. *Journal of the Royal Statistical Society: Series B (Methodological)*, *52*(1), 21–50. <https://doi.org/10.1111/j.2517-6161.1990.tb01771.x>
- Williams, D. R., & Mulder, J. (2020). Bayesian hypothesis testing for Gaussian graphical models: Conditional independence and order constraints. *Journal of Mathematical Psychology*, *99*, Article e102441. <https://doi.org/10.1016/j.jmp.2020.102441>
- Xu, Y., Salapaka, S. M., & Beck, C. L. (2011). On reduction of graphs and Markov chain models. In *2011 50th IEEE Conference on Decision and Control and European Control Conference* (pp. 2317–2322). IEEE. <https://doi.org/10.1109/CDC.2011.6160882>
- Yin, J., & Li, H. (2011). A sparse conditional Gaussian graphical model for analysis of genetical genomics data. *Annals of Applied Statistics*, *5*(4), 2630–2650. <https://doi.org/10.1214/11-AOAS494>
- Zhao, J., Zhou, Y., Zhang, X., & Chen, L. (2016). Part mutual information for quantifying direct associations in networks. *Proceedings of the National Academy of Sciences of the United States of America*, *113*(18), 5130–5135. <https://doi.org/10.1073/pnas.1522586113>
- Zhou, Q., Eisenberg, N., Losoya, S. H., Fabes, R. A., Reiser, M., Guthrie, I. K., Murphy, B. C., Cumberland, A. J., & Shepard, S. A. (2002). The relations of parental warmth and positive expressiveness to children's empathy-related responding and social functioning: A longitudinal study. *Child Development*, *73*(3), 893–915. <https://doi.org/10.1111/1467-8624.00446>