Which Estimation Method to Choose in Network Psychometrics? Deriving Guidelines for Applied Researchers

Adela-Maria Isvoranu[1] & Sacha Epskamp[1,2]

[1]Department of Psychology, Psychological Methods, University of Amsterdam, Amsterdam, The Netherlands

[2]Centre for Urban Mental Health, University of Amsterdam, Amsterdam, the Netherlands

This is a pre-print of a manuscript that has been resubmitted for publication.

Abstract

The Gaussian Graphical Model (GGM) has recently grown popular in psychological research, with a large body of estimation methods being proposed and discussed across various fields of study, and several algorithms being identified and recommend as applicable to psychological datasets. Such high-dimensional model estimation, however, is not trivial, and algorithms tend to perform differently in different settings. In addition, psychological research poses unique challenges, including placing a strong focus on weak edges (e.g., bridge edges), handling data measured on ordered scales, and relatively limited sample sizes. As a result, there is currently no consensus regarding which estimation procedure performs best in which setting. In this large-scale simulation study, we aimed to overcome this gap in the literature by comparing the performance of several estimation algorithms suitable for gaussian and skewed ordered categorical data across a multitude of settings, as to arrive at concrete guidelines from applied researchers. In total, we investigated 60 different metrics across 564,000 simulated datasets. We summarized our findings through a platform that allows for manually exploring simulation results. Overall, we found that an exchange between discovery (e.g., sensitivity, edge weight correlation) and caution (e.g., specificity, precision) should always be expected and achieving both—which is a requirement for perfect replicability—is difficult. Further, we identified that the estimation method is best chosen in light of each research question and highlighted, alongside desirable asymptotic properties and low sample size discovery, results according to most common research questions in the field.

Keywords: GGM, network psychometrics, network models, network analysis

1. Introduction

The Gaussian graphical model (GGM; Epskamp, Waldorp, Mõttus, & Borsboom, 2016; Lauritzen, 1996)—a network structure of variables represented by nodes and linked by edges that are weighted by partial correlation coefficients—has recently grown popular in psychological research, especially in the fields of clinical psychology and psychiatry (Robinaugh et al., 2020). While confirmatory fit of a given GGM is possible (Epskamp, Rhemtulla, et al., 2017; Kan et al., 2019), in most cases a prior theoretical network structure is absent. Many researchers therefore focus on *exploratory* estimation of GGMs: identifying the structure (absence and presence of edges), as well as estimating the edge weights (i.e., partial correlation coefficients). A large body of estimation methods have been proposed and discussed across various fields of study (e.g., Drton & Perlman, 2004; Friedman, Hastie, & Tibshirani, 2008; Meinshausen, Meier, & Bühlmann, 2009), and several algorithms have been identified and recommend as applicable to psychological datasets (e.g., Epskamp & Fried, 2018; Williams, Rhemtulla, Wysocki, & Rast, 2019).

Of note, however, such high-dimensional model estimation is not trivial—especially when the sample size is small relative to the number of potential parameters—and algorithms tend to perform differently in different settings, such as being more or less conservative with balancing the rate of discovering true edges to the rate of discovering false edges. In addition, psychological research poses unique questions and problems, including placing a strong focus on weak edges (e.g., bridge edges) handling data measured on ordered scales, and relatively limited sample sizes. As a result, there is currently no consensus regarding which estimation procedure performs best in which setting. In this paper, we aim to overcome this gap in the literature by comparing the performance of several estimation algorithms suitable for gaussian and skewed ordered categorical data across a multitude of settings in a large-scale simulation study, as to arrive at concrete guidelines from applied researchers.

1.1. GGM estimation from ordered categorical data

A common method for estimating GGM structures widely used in prior literature is the *EBICglasso* algorithm (Epskamp & Fried, 2018), which estimates a regularized GGM based on a correlation matrix as input, by combining the graphical LASSO algorithm (glasso; Friedman et al., 2008) with the extended Bayesian information criterion (EBIC; Chen & Chen, 2008) for tuning parameter selection. For ordered categorical data, often a polychoric correlation matrix (Olsson, 1979) is used as input. This routine is problematic for several reasons: First, recently Williams & Rast (2018) highlighted that regularization is not required for models with large sample sizes compared to the number of nodes, and may lead to poor estimation in some cases. In particular, the tuning parameter selection performs poorly when regularized GGM structures are used in EBIC computation, especially when these networks are dense (i.e., contain many edges). Second, the use of polychoric correlations as input for a likelihood-based estimation method is not the optimal method of estimation in related modeling frameworks, such as structural equation modeling. Within these frameworks, it is recommended to use weighted least squares (WLS) estimation instead, to more properly handle sampling variation in the data (Muthén, 1984). This is particularly important in smaller sample sizes, as the polychoric correlations have been shown to cause large amounts of sampling variation in network structures (Forbes et al., 2019). Finally, it should be noted that very little methodological research has studied the performance of using polychoric correlations as input to the EBICglasso, and the studies that did (Epskamp, 2017; Williams et al., 2019) only investigated variables that were not skewed. This is in stark contrast with reality, where data used in GGM estimation from psychological datasets are often skewed and measured on ordered categorical scales (in psychological data often ranging between 3-point and 5-point scales).

1.2. Empirical questions in psychological research

Empirical questions for GGM estimation in psychological research are unique, and warrant dedicated investigation by themselves. While network models were initially often utilized in an aim to identify links between a wide array of symptoms pertaining to a mental disorder (e.g., Fonseca-Pedrero et al., 2018; Fried & Nesse, 2015; McNally et al., 2015), the field has fast advanced to the study of more complex processes, such as investigating the comorbidity between disorders (e.g., Choi, Batchelder, Ehlinger, Safren, & O'Cleirigh, 2017; Lazarov et al., 2019; Malgaroli, Maccallum, & Bonanno, 2018; Vanzhula, Calebs, Fewell, & Levinson, 2019), and aiming to identify bridging links between environmental and genetic risk factors and symptomatology (Boyette et al., 2020; Fried et al., 2015; Isvoranu, Guloksuz, Epskamp, Van Os, & Borsboom, 2019; Isvoranu et al., 2017; Isvoranu, Borsboom, van Os, & Guloksuz, 2016). Often, such links are weaker than symptom-symptom links and more difficult to stably and soundly identify (Boyette et al., 2020; Isvoranu et al., 2019). However, these are often essential both to the onset and progress of a mental disorder, and they may be key aspects for successful intervention development.

Further, in recent years, replicability has been a central issue in the field of psychology (Open Science Collaboration, 2015) and is currently of growing interest in the field of network psychometrics (Borsboom, Robinaugh, Rhemtulla, & Cramer, 2018; Epskamp, Borsboom, & Fried, 2017; Fried, 2017). While recent research showed heterogeneity is high even when focusing on one particular disorder, and thus aiming to identify a *one* overall network structure replicable across different populations is not feasible (Isvoranu, Epskamp, & Cheung, 2020), choosing an inadequate estimation algorithm may also play an important role in the replicability and generalizability of network structures. Choosing an estimation technique that is best suited for the type of question and data a researcher has is thus critical to estimate an interpretable and reliable network structure. With

the growing number of available GGM estimation algorithms, this is however becoming more challenging and clear guidelines for applied researchers are still lacking.

1.3. Aim of the paper

The aim of the current paper is therefore to investigate the performance of existing GGM estimation algorithms across a multitude of settings commonly encountered by applied researchers. Specifically, we focus on non-skewed and skewed continuous and ordered categorical data, as psychological datasets are often skewed and measured on continuous or ordered categorical scales. We study 13 different estimation algorithms, further described in Table 1. In particular, we study the *EBICglasso* and *ggmModSelect* (two variants) algorithms that are implemented in the *qgraph* R package; Epskamp, Cramer, Waldorp, Schmittmann, & Borsboom, 2012), two variants of full information maximum likelihood (FIML) and weight least squares (WLS) estimation as in the *psychonetrics* R package (Epskamp, 2020a, 2020b), two variants of mixed graphical model estimation as implemented in the *mgm* R package (Haslbeck & Waldorp, 2020), two variants of Bayesian estimation as implemented in the *BGGM* R package (Williams & Mulder, 2019), and two more unregularized estimation procedures as implemented in the *GGMnonreg* R package (Williams et al., 2019).

We assess the performance of these methods by simulating data from three plausible network models (Figure 1): (1) a network model derived from a large empirical dataset of depression, anxiety and stress measures (Lovibond & Lovibond, 1995), (2) a meta-analytic network model on post-traumatic stress disorder (PTSD) symptoms, recently reported by Isvoranu, Epskamp & Cheung (2020), and (3) a network model of personality traits. In the simulation studies, we vary the number of nodes, sample size, and data type (continuous and skewed ordered categorical), and investigate a large array of metrics designed to assess the performance of retrieving edges, the accuracy of the edge weights, the accuracy of centrality

indices, the performance of retrieving bridging edges between clusters of different disorders, and the replicability of these measures.

Finally, in order to summarize the results in a clear and intelligible way, as to aid applied researchers in interpreting the results and guiding them toward choosing suitable estimation algorithms, we created an interactive data visualization tool using the Shiny framework (Winston et al., 2017). The interactive data visualization tool will enable researchers to include information about their study, such as expected sample size and network properties, as well as expected performance attributes, and based on this information they will receive guidelines on which estimation algorithm(s) may be best suited for their study. The current manuscript includes a short example and demonstration of how the interactive data visualization tool can be used, and describes results from the application to answer several research questions.

In addition to the results reported here, all simulation results as well as an example of the simulation code can be found in our online supplementary materials at the Open Science Framework (OSF)[1] and the source code of the interactive data visualization tool can be found in our Github repository.[2]

2. Methods

2.1 Data transformation

To handle non-normal skewed data as well as ordered categorical data, we varied between four different levels of transformations. In addition to studying the performance of estimators without transforming the data, we investigated rank transformations and non-paranormal transformations (Liu et al., 2009). Transforming variables to rank orders is equivalent to

[1] https://osf.io/ycmvf/
[2] https://github.com/AdelaIsvoranu/simulation_exploration_app

using Spearman correlations is input to estimators that use the correlation matrix as input (e.g., *EBICglasso*), but also works for estimators that use raw data as input (e.g., *mgm*). The non-paranormal transformation is a semi-parametric transformation designed to handle non-normality. Finally, for several estimators we also investigated dedicated options for studying ordered categorical variables. These we collectively term "polychoric/categorical" transformation, and is explained below for each estimator separately. The polychoric/categorical condition was only used when data were ordered categorical.

2.2. Network estimation

In total, we investigated thirteen different network estimation methods, also summarized in Table 1. For simplicity, throughout this paper, we will assume no missing data is present. Below we provide a short description of each estimation algorithm investigated here.

qgraph. The *qgraph* package version 1.6.7 was used for two estimators: the *EBICglasso*, used for regularized GGM estimation, and the *ggmModSelect*, used for unregularized model search.

The *EBICglasso algorithm* (Epskamp & Fried, 2018) estimates a regularized GGM based on a correlation matrix as input using the the graphical LASSO (glasso; Friedman et al., 2008) in combination with EBIC model selection (Chen & Chen, 2008; Foygel & Drton, 2011). In regularized estimation, parameters are estimated through the use of *penalized maximum likelihood estimation*: a penalty for model complexity is added to the likelihood, which is subsequently optimized. A tuning parameter controls the amount of penalization, and therefore the sparsity of the resulting network. In the EBICglasso this tuning parameter is varied to return a range (by default 100) of network models, and subsequently the EBIC is used to select which network model fits best. A hypertuningparameter in the EBIC

computation, γ (not to be confused with the LASSO tuning parameter), can be used to further penalize model complexity, and is typically set to 0.5. We also set this hypertuning parameter to 0.5 in the simulation studies reported below, as it is the default used in the software and recommended in prior literature (Epskamp, Borsboom, et al., 2017; Foygel & Drton, 2010). No thresholding is performed on the final GGM by default, as the glasso already performs model selection by shrinking many parameters to exactly equal zero. In the polychoric/categorical condition, we used polychoric correlations (termed `cor_auto` in line with the function used to obtain these correlations) as input to the *EBICglasso* algorithm.

The *ggmModSelect* algorithm is a potentially fast non-regularized algorithm for network estimation, based—like the *EBICglasso*—primarily on the glasso algorithm in combination with BIC model selection. The *ggmModSelect* algorithm was implemented in *qgraph* in response to the work by Williams & Rast (2018) and has been used in prior literature (Isvoranu, Guloksuz, Epskamp, van Os, et al., 2019; Kan et al., 2019), but not yet investigated in detail in published reports. Like the *EBICglasso*, the algorithm generates a range of network structures (by default 100) by varying the LASSO tuning parameter. The key difference with the *EBICglasso* is that subsequently, each network is re-estimated *without* regularization. This is done through maximum likelihood estimation, in which only the edges that were non-zero in the regularized networks are estimated to be non-zero. The *glasso* package is also used for this purpose, by setting the LASSO tuning parameter to zero (no penalization) and supplying a list of edges that are fixed to zero. These non-regularized estimates are subsequently used to optimize the EBIC (by default, $\gamma = 0$ in which case the EBIC reduces to the BIC), which should lead to better model selection as the *EBICglasso* algorithm as asymptotic convergence for the true model to be selected by EBIC is only valid for non-regularized maximum likelihood estimates of the network parameters (Foygel & Drton, 2010). After this process, the *ggmModSelect* algorithm continues optimization by

stepwise adding and removing edges until the EBIC criterion is optimized (this can be slow in larger datasets, and can be disabled using the `stepwise = FALSE` argument). In the simulations, we set $\gamma = 0$ in line with the default used in the software and use polychoric correlations as input in the polychoric/categorical condition.

psychonetrics. The *psychonetrics* package (we used version 0.9) includes GGM estimation through the same estimators as typically used in structural equation modeling (SEM) packages (Epskamp, 2020b, 2020a). In particular, the *psychonetrics* package mimics the often used *lavaan* package (Rosseel, 2012) by including the full information maximum likelihood (FIML) and weighted least squares (WLS) estimators. The FIML estimator treats the data as continuous and assumes normality to estimate the parameters by optimizing the likelihood function without penalties for model complexity, while the WLS estimator uses a weights matrix to minimize the difference between observed and model implied correlational structures. For ordered categorical data, the three-stage WLS estimator (Muthén, 1984) can be used to model polychoric correlations: univariate and pairwise information is used to estimate the thresholds (stage 1) and polychoric correlations (stage 2), and finally a model is fitted to reproduce the thresholds/polychoric correlations (stage 3). This three-stage WLS estimator has not yet been used in GGM estimation, and is evaluated in the polychoric/categorical condition of our simulation studies. For the FIML estimator, no polychoric/categorical condition was available. For optimization for both FIML and WLS estimation the R function nlminb is used.

Both estimators can use *pruning* to estimate a network model. This process starts with a GGM in which all edges are included, removes all edges that are not significant at alpha = 0.01, and re-estimates the remaining edges while keeping the removed edge weights fixed to zero. After pruning, the model can be refined with further stepwise model estimation. In

WLS estimator, a stepwise routine is used that adds edges with the largest modification index until no modification index significant at alpha = 0.01 can be found. For the ML estimator, the more sophisticated *modelsearch* algorithm can be used (Epskamp, 2020a), which is similar to the stepwise estimator in *ggmModSelect*, but utilizes modification indices and significance of edges to speed up the model search: in each step, only absent edges with significant modification indices are considered to be added, and only included edges that are not significant are considered to be removed. This makes the *modelsearch* algorithm in principle faster than the *ggmModSelect* algorithm. In practice, however, in the current implementations we studied the *modelsearch* algorithm is much slower than the *ggmModSelect* algorithm due to the slower estimation through *nlminb* compared to *glasso* and the relatively slow computation of the standard errors and modification indices used in each step.

MGM. A modeling framework closely related to the GGM is *mixed graphical modeling* (MGM), implemented in the *mgm* software packages (Haslbeck & Waldorp, 2020), of which we used version 1.2.11. The *mgm* package makes use of node-wise penalized generalized linear models for estimating edges connected to each node. That is, the package loops over each variable included in the model, performs a regularized generalized linear regression to predict the variable of interest from all other variables in the model, and finally combines and averages all estimated regression weights into a network model. This method allows for continuous, categorical and count variables to be used. When continuous variables are used, the model is very similar to the GGM, differing only slightly in the parameterization used (regression coefficients between standardized variables rather than partial correlation coefficients). Like the EBICglasso method, the LASSO tuning parameter can be varied to obtain a range of networks. The *mgm* package subsequently allows for model selection in two

ways: choosing a model that optimizes the EBIC, and choosing a model that optimize cross-validation (CV) prediction accuracy. We assess both variants in our simulation studies below, with the EBIC hypertuningparameter γ set to 0.5 to be comparable to the *EBICglasso* simulations and the number of cross-validation folds set to 10 in line with previous literature (Costantini et al., 2015) and the software default.. While *mgm* does not handle ordered categorical data directly, it does allow for variables to be modeled as categorical variables. In our simulations, we assess the performance of treating ordered categorical variables as categorical.

BGGM. All estimation methods discussed so far are *frequentist*, in that these return point estimates obtained by maximizing the (pseudo) likelihood (including possible a penalty for model complexity) of the model given the data. An alternative to frequentist estimation is Bayesian estimation, in which Bayes' rule is used to obtain a posterior distribution of the parameter given the data. This distribution can subsequently be used for detailed inference on the parameter values. The *BGGM* package contains several methods for estimating GGM structures using Bayesian sampling methods (Williams & Mulder, 2020). In our simulations, we used version 2.0.2 of the BGGM package. The first method we include utilizes the `estimate` function, which makes use of a Wishart prior, as further detailed by (Williams, 2021a). The posterior samples are then used to form credibility intervals on the parameter values, allowing to test if a parameter is non-zero at a given α level (here set to 0.05) by checking if zero is not in the credibility interval. The second method we include utilizes the `explore` function, which uses a matrix-F prior distribution (Mulder & Pericchi, 2018), as further detailed by Williams & Mulder (2020). This prior allows for the construction of Bayes factors to determine evidence for the presence (and absence) of edges. In our simulation studies, we use a Bayes factor of 3 to determine the presence of an edge in the

selected GGM structure, which is the default in the *BGGM* package. Both methods allow for handing ordinal data through the use of a semi-parametric Gaussian copula model (Hoff, 2007).

GGMnonreg. The final package we used in the simulation studies is the *GGMnonreg* package for non-regularized GGM estimation (Williams et al., 2019). At the time we performed the simulation study, the R package was not on CRAN. We used the developmental version published on Github with commit reference "58965a3".[3] This package includes two different estimation methods. The first, *GGM_bootstrap*, estimates a saturated partial correlation network, and subsequently uses non-parametric bootstrapping to assess significance of edges in order to threshold non-significant edges at some level of α. To this end, the method is very similar to the *pruning* used in the *psychonetrics* estimators, except that non-analytic significance values are used, and parameters are not re-estimated with non-significant edges fixed to zero. The second method, *GGM_regression*, uses step-up model search for nodewise regression models to optimize some information criterion (in our simulations: BIC). This method is much faster than other stepwise estimators assessed in our simulations, as the stepwise search is performed for the predictors of each variable separately rather than in the full multivariate model. Both these estimators are further described by Williams et al. (2019).

2.3. Network model construction

We simulated data under three network models based on large sample sizes that can be expected to represent target network structures in psychological research. The network structures used in the simulation study are shown in Figure 1, with some more details

[3] https://github.com/donaldRwilliams/GGMnonreg

presented in Table 2. We made use of non-regularized estimation procedures in each of these datasets as to obtain parameter values that are in line with what can be expected in a psychological setting (regularized networks would lead to biased parameter estimates that are shrunk to zero).

==========

========== FIGURE 1 ABOUT HERE ==========

==========

First, we analyzed the short version of the Depression Anxiety Stress Scales (DASS21; Lovibond & Lovibond, 1995), which contains 21 items aimed at measuring depression, anxiety and stress. We obtained the data from the Open Source Psychometrics Project (openpsychometrics.org), which contained n = 39,775 full responses on 21 items measured on a 4-point scale. We analyzed this data using the *ggmModSelect* algorithm with stepwise estimation and a rank-order transformation (Spearman correlations). We termed this the *DASS21 network*.

Second, we used a meta-analytic network model on PTSD symptoms reported by Isvoranu, Epskamp & Cheung (2020), estimated using *meta-analytic Gaussian network aggregation* (MAGNA; Epskamp et al., 2020). In this meta-analysis, 52 samples, with a total sample size of n = 29,561, used in published papers for estimating PTSD symptom networks were re-analyzed to obtain a single aggregated GGM structure. This structure includes 17 nodes representing the DSM-IV-TR PTSD symptoms. To obtain a network with edge weights of exactly zero (i.e., absent edges), we removed all edges that were not significant at α = 0.05. We termed this model the *MAGNA network*.

Finally, we analyzed the `bfi` dataset from the *psychTools* package (Revelle, 2015) designed to measure the Big 5 personality traits(Benet-Martínez & John, 1998; Digman, 1989; Goldberg, 1990, 1993; McCrae & Costa, 1997). This dataset consists of 2,800

observations of 25 personality inventory items. We estimated a network model using the *ggmModSelect* algorithm with stepwise estimation. We termed this model the *BFI network*.

2.4. Data Generation

We generated four types of data: normally distributed continuous data, skewed continuous data, uniformly distributed ordered-categorical data with four levels, and skewed ordered-categorical data with four levels. Figure 2 shows the intended target distributions for each of these types of data. Data were first generated as normally continuous data, after which data were transformed for the skewed and ordered-categorical conditions. In the skewed condition, we used the exponential function to transform data, indicating that the data were log-normally distributed. For the ordered-categorical conditions, we used thresholds models (Epskamp & Fried, 2018), such that data with four levels were generated that were either uniformly distributed or skewed distributed.

==========

========== FIGURE 2 ABOUT HERE ==========

==========

2.5. Quantifying network estimation accuracy

We investigated a total of 60 different measures to quantify the accuracy of network estimation, which we describe further below, with the number in brackets indicating the order in which the metrics appear in the web application we use to study the results

Sensitivity: ability to identify true edges. In line with previous studies (Epskamp & Fried, 2018), we investigated the (1) *sensitivity* of the network structure, which represents the proportion of edges in the true network that were also included in the estimated network:

$$Sensitivity = \frac{\text{Number of true edges in the estimated network}}{\text{Total number of edges in the true network}}$$

Unlike prior research, we also included four additional variants of the sensitivity. The (2) *signed sensitivity* is the sensitivity as defined above, except that the numerator only counts edges that were estimated with the same sign as edges in the true network as 'true edges.' To investigate if an estimation method is capable of detecting the strongest edges, we also investigated the sensitivity of the top (3) 50%, (4) 25%, and (5) 10% absolute edge weights from the true network. For example, the *top* 25% *sensitivity* gives the proportion of the 25% strongest edges in the true network that were also included in the estimated network. Sensitivity should increase with sample size—when sensitivity is low, there is a risk that not all edges are detected, which can lead to poorer replicability and inflated visual heterogeneity across estimated networks (Hoekstra et al., 2020; Mansueto et al., 2020), as well as a potentially false conclusion that the generating network model was sparse (Epskamp, Kruis, et al., 2017).

Specificity & precision: ability to not include false edges. Similar to prior studies, we investigated the (6) *specificity* of the network structure, which represents the proportion of missing edges in the true network that were correctly not included in the estimated network:

$$Specificity = \frac{\text{Number of true absent edges in the estimated network}}{\text{Total number of absent edges in the true network}}$$

The specificity can also be seen as 1 minus the false positive rate (α; (Williams & Rast, 2020). Specificity should always be high, and any trend in specificity as a function of sample size can indicate problems with the estimation method. In addition, we also investigated the (7) *precision*, which is the proportion of included edges in the estimated networks that were also included in the true network:

$$Precision = \frac{\text{Number of true edges in the estimated network}}{\text{Total number of edges in the estimated network}}$$

Like specificity, precision will be low if many edges are included in the network that are actually not in the true network (false edges). Of note, specificity and precision can differ. For instance, suppose that a true network contains 50 edges and 50 absent edges, and suppose that we estimate a network with only two edges: a true and a false edge. Then, specificity is high (out of 50 potential false edges, we only included one), but precision is low (out of 2 identified edges, 50% were false edges). To investigate the prominence of falsely detected edges, we also computed the precision in estimated edges with the top (8) 50%, (9) 25% and (10) 10% absolute edge weight. This led to three more variants of precision. For example, the *top* 25% *precision* gives the proportion of the 25% strongest edges in the estimated network that were also true edges. A low *top* 25% *precision* would indicate that false edges are very prominent in the network structure and visualization.

Edge weight accuracy: ability to estimate precise edge weights. All measures above (sensitivity, specificity, and precision) only investigate if an edge is included or not, not if the edge weight is estimated accurately (Epskamp, Borsboom, et al., 2017). This is however important as the edge weight directly controls the network visualization. A related topic is on the topic of how prominent false edges are, which should be estimated to be zero but are included as non-zero edges in the network. We used five measures to quantify these topics:

- (11) The Pearson *correlation* between the full vectorized edge weight matrices, as previously described by Epskamp & Fried (2018).
- (12) The Pearson correlation between the absolute edge weights, as the sign of edges may greatly inflate the correlation otherwise.

- (13) The average absolute deviation (*bias*) between the true edge weight and the estimated edge weight all edges (included and not included).
- (14) The average absolute deviation between the true edge weight and the estimated edge weight for true edges that were included in the estimated network.

(15) The *maximum fading of false edges*. To compute this, we first compute the *opacity* of the *strongest false edge* in a network: the absolute weight of that edge divided by the absolute weight of the strongest edge in the network. Next, we computed the *maximum fading of false edges* as one minus the opacity of the strongest false edge. This measure is directly comparable to the *fading* used of edges in the *qgraph* package (Epskamp et al., 2012), which is often used to visualize psychological network structures, and quantifies how prominently displayed a false edge can be in the estimated network. For example, a *maximum fading* of 0.9 indicates that the strongest falsely included edge was faded to white for 90%. **Centrality index accuracy: ability to identify important nodes.** To investigate the accuracy of centrality indices, we computed the (16, 21, 26) Pearson correlation and the (17, 22, 27) Kendall correlation between the obtained centrality indices in line with prior literature (Borsboom et al., 2017). In addition, we investigated the ability of detecting nodes with the strongest centrality indices, by investigating how many of the same nodes were correctly placed in the (18, 23, 28) top 1, (19, 24, 29) top 3 and (20, 25, 30) top 5 of most central nodes in the network. We investigated these five measures for node strength, closeness and betweenness[4].

Bridge edges detection: ability to detect edges that connect clusters. To assess the ability of methods to detect edges connecting clusters (which are often relatively weak), we investigated a subset of the measures above exclusively for (potential) edges connecting

[4]We did not investigate expected influence (Robinaugh et al., 2016), as most edges in the network were positive and we did not expect any difference with node strength)

clusters in the DASS21 and BFI networks (the MAGNA networks did not feature clusters). For these edges we computed the (31) sensitivity, (32) signed sensitivity, (33) top 50% sensitivity, (34) top 25% sensitivity, (35) top 10% sensitivity, (36) specificity, (37) precision, (38) top 50% precision, (39) top 25% precision, and the (40) top 10% precision.

Network replicability: ability to replicate features in an independent dataset. To assess replicability, we simulated a second dataset for each condition, and estimated a second GGM model using the same procedure as the first GGM model. The second network is thus intended as a replication of the first. We compared these two networks on several metrics:

- (41) The correlation between the edge weights from the first network and edge weights from the second network.
- (42) The proportion of edges from the first network that were also included (replicated) in the second network. This was also estimated for bridge edges.
- The proportion of (43) top 50% / (44) 25% / (45) 10% edges (based on the strength of their absolute weight) that were also included in the second network (regardless of strength). This was also estimated for bridge edges.
- (46) The proportion of absent edges from the first network that were also classified as absent in the second network. This was also estimated for bridge edges.
- For the DASS21 and BFI networks, we also investigated the proportions of (47) replicated bridge edges, (48) replicated top 50% bridge edges, (49) replicated top 25% bridge edges, (50) replicated top 10% bridge edges, and (51) replicated absent edges between clusters.
- The (52, 55, 58) Pearson and (53, 56, 59) Kendall correlations between node strength, closeness and betweenness centrality indices from both networks.

- (54, 57, 60) Whether or not the most central edge according to node strength, closeness and betweenness was the same.

2.6 Simulation setup summary. In sum, we simulated six different levels of sample size (150, 400, 600, 1000, 2,500, and 5,000), four different kinds of data (normal continuous, skewed continuous, uniform ordered categorical, and skewed ordered categorical), three different types of true networks (MAGNA, DASS21 and BFI), four different transformations (no transformation, rank/Spearman, non-paranormal and polychoric/categorical), and 13 different estimators. The polychoric/categorical transformation was only used for nine out of 13 estimation methods, and only for data that was ordered categorical. Furthermore, the stepwise estimators implemented in *psychonetrics* (WLS stepup estimation and ML model search) were too computationally challenging to assess for the two larger network models (DASS21 & BFI), and therefore only included for the MAGNA network simulations. Every condition was repeated 100 times, leading to a total of 282,000 simulation conditions. In each condition, two datasets were generated: one for the main estimated network for most metrics, and a second to assess replicability. As such, a total of 564,000 datasets were generated.

3. Results

Out of 282,000 simulation conditions, 1,070 resulted in an error (e.g., especially in low sample sizes some estimators ran into convergence issues), leading to a total of 280,930 conditions and thus 561,860 simulated datasets to be included in the results. As these were assessed on 45 (MAGNA) to 60 (DASS21 & BFI) metrics, there were far too many results to report that could be fitted in the bounds of this paper. We therefore chose to summarize our findings through a platform that allows for manually exploring simulation results, discussed in more detail below. We use this app in deriving results for more concrete research questions below. A subset of the results—highlighting some of the most important metrics assessed

only in the case of Gaussian data without the use of a transformation—can be found in the tables in Appendix B.

3.1 Simulation exploration app

We make results from our simulation study available through the *simulation explorer app* (SEA), which is available on Github[5]. The app can be run locally by using the following R commands:

```
library("shiny")
runGitHub("AdelaIsvoranu/simulation_exploration_app")
```

This code will install several required R packages when run for the first time. Alternatively, the app is also hosted online and accessible via psychonetrics.org/simulations. Figure 3 shows a screenshot of SEA. Included in the app are (1) an overview of network models used in the simulation study, (2) an overview of estimators used in the simulation study, (3) a *recommender system*, that can be used to recommend estimators based on certain required conditions (e.g., find an estimator that has an average specificity above 0.9 and an average sensitivity of 0.5 or higher), (4) radar plots allowing to compare methods on several metrics for a given setting (such as sample size), and (5) line plots for investigating performance as a function of sample size. All results discussed below can be visualized using SEA. To this end, we do not include further figures or tables in this manuscript, as doing so would mean selectively highlighting some results in favor of others.

==========

========== FIGURE 3 ABOUT HERE ==========

==========

[5] https://github.com/AdelaIsvoranu/simulation_exploration_app

3.2. Overall results

Primarily and as a general summary, an important finding of the current simulation study is that an exchange between discovery (e.g., sensitivity, edge weight correlation) and caution (e.g., specificity, precision) should always be expected and achieving both—which is a requirement for perfect replicability (i.e., both very good edge inclusion and no false positive edges)—is difficult. It is therefore important to consider in light of each research question whether there is a favor for one or the other: some methods perform better at retrieving a lot of true edges with an appropriate edge weight, but may include also false edges. Other methods may be very conservative, but fail to properly retrieve the global picture.

Further, findings in terms of transformations and data types: when data are Gaussian, applying a non-paranormal on rank-transformation (Spearman correlations as input) did not seem to impact performance of the estimators. When data were skewed, a non-paranormal or rank-transformation improved the performance in the majority of estimators across most datasets. For ordered categorical data, there was hardly a difference between using a transformation or not. Surprisingly, dedicated methods for handling ordered categorical (polychoric/categorical 'transformation') data did not perform much better than other methods, and actually performed worse in many cases. Notably, in ordered categorical data, the *mgm* estimators that model responses as categorical had very poor performance on replicated zeroes, likely related to the poorer specificity of these estimators described below. For other estimators, there weren't strong differences between transformations and methods of modeling ordered categorical data. As rank-transformations (Spearman correlations as input) worked well on all data types and comparably to the more complicated nonparanormal transformation, we will discuss results specific to rank-transformations below in more detail.

The remainder of this section will highlight findings in specific settings across the three networks. We will start by investigating desirable asymptotic (i.e., high sample size)

properties, for (skewed) continuous and ordinal data, followed by desirable low sample size properties. Table 3 summarizes the operationalizations used for each of these settings.

Desirable asymptotic properties. First, we investigate asymptotic properties of a network structure. We have defined asymptotic properties as low false inclusion rate of edges, high rate of identifying (the strongest) edges in the network structure, high correlations between the estimated network structure and the true network structure, in the context of a very large sample size. That is, we were interested in achieving best performance at a large sample size. We specifically operationalized this question by looking at the following properties in SEA radar plots: *n = 5,000*, *specificity* and *precision*, to investigate if the edges included in the network structures were true edges, *sensitivity*, as well as *sensitivity (top 50%)*, to investigate if the (strongest) edges were detected, *absolute correlation*, and *1-bias* to detect if similar differences between strong and weak edges in the true network were estimated. Overall, as expected, most estimators reached desirable asymptotic properties with a high sample size. Exceptions were the *EBICglasso* algorithm, followed by the *mgm (CV; 10 folds)* algorithm and, especially in the DASS21 network, the *ggmModSelect* algorithm without stepwise estimation, both performing poorer in terms of *specificity* and *precision*.

We conclude that at a high sample size (*n=5,000*), most estimators work well, but unregularized estimators (especially *ggmModselect stepwise*, *FIML* estimators, and *BGGM estimate*) work best in retrieving a network structure with a low false inclusion rate of edges, high rate of identifying (the strongest edges) in the network structure, and high correlations between the estimated network structure and the true network structure. In addition, as detailed above, in the case of skewed normal data, rank-transformations (Spearman correlations as input) improve the performance of estimators, while in the case of (skewed) ordered categorical data transformations don't make a considerable difference.

Low sample size discovery. Further, we investigate low sample size discovery. That is, we focus on desirable results obtained at low sample size, which is often a common encounter in applied research (e.g., in mental health research where patients are a central research concern). We were thus interested in whether we can discover the most important edges and the overall network structure at low sample size. In this setting we allow for a method to include false edges, as long as these are hardly visible in default faded edge weight visualizations. In addition, we focused on *precision,* rather than *specificity*, as we were interested in whether the edges included in the network structures were true edges. We specifically operationalized this question by looking at the following properties in SEA radar plots: *n = 300*, *sensitivity (top 25%)*, to investigate if the strongest edges were included the network structures, *precision*, to investigate if the edges included in the network structures were true edges, and *fade maximum false edge*, to investigate whether the false positives edges in the network structures were hardly visible.

While many estimators performed well in one or two of the three predefined properties, the three regularized estimators, *EBICglasso* and the two *mgm* estimators (EBIC and CV selection), worked best in terms of the *fade maximum false edge* (faded edges were at most around 25% visible). Of these, *EBICglasso* and *mgm* (CV) worked best in terms of *sensitivity (top 25%)*, while the EBIC selection variant of *mgm* performed poorer in sensitivity (top 25%) but better in precision. Overall, these estimators performed well in discovering the strongest edges in the network structures, while also ensuring that false edges were not prominently visible. Of note, however, there was a clear inverse relationship between the *sensitivity (top 25%)* and *precision*, indicating that edges discovered by these methods are also more likely to be false edges than edges discovered by other methods. Choosing an unregularized estimator, on the other hand, will lead to a higher precision, but at the cost of both losing power to detect strong edges, as well as the risk of more prominently

featured strong edges. Using a transformation improved the performance of most estimators for continuous skewed data, but had little impact in the case of skewed ordered categorical data, with the exception of the *polychoric correlations*, which considerably improved the *sensitivity (top 25%)* in the case of the *EBICglasso* algorithm in the MAGNA network.

Therefore, at these sample sizes we recommend even more so than in other cases to let the choice of estimator be guided by the research question(s). If the goal of the study is to investigate the presence of each individual discovered edge, and if these will be displayed without fading, then regularized estimators should be avoided. However, if the goal is to discover a structure that resembles a true network and to discover the strongest edges, regularized estimators are preferred in combination with a fading rule used to draw the edges, as it is default for example in the *qgraph* package.

3.3. Specific research questions

This section will highlight findings based on common research questions of applied researchers. We will start by investigating visual network alignment, followed by identification of central nodes (i.e., in terms of strength), identification of bridge edges, and finally replicability. Table 3 also summarizes the operationalizations used for each of these research questions.

Visual network alignment. Next, we investigate how well the visual network picture aligns with the true network structure at a medium sample size (n = 1,000). That is, we were interested in if the estimated network picture resembles the true network picture (e.g., are the same edges strong and visually present in the estimated and the true networks). In this setting we allow for a method to include false edges, as long as they are hardly visible in default faded edge weight visualizations. We specifically operationalized this question by looking at

the following properties in SEA radar plots: *sensitivity (top 50%)* to investigate if the strongest edges were detected, *correlation (absolute)* and *1 – bias (true edges)* to detect in similar differences between strong and weak edges in the true network were estimated, and *fade maximum false edge* to detect how pronounced the strongest false edge was in the estimated network.

Across all three empirical network structures used in simulations, the *EBICglasso* algorithm performed particularly well on these benchmarks: the algorithm consistently scored very high on all four measures, indicating that the most important edges were routinely included and false edges were not prominent in the network. The *mgm* methods also performed well in this setting, with the EBIC variant performing better on *fade maximum false edge* criterium (not showing false edges), while the cross-validation variant performed better on *sensitivity (top 50%)*. Unregularized estimators, such as the *WLS estimators, FIML, GGMnonreg* and *BGGM* variants, performed a bit worse, although not very badly (only the DASS21 network showed top 50% sensitivity levels under 0.75 on average). These estimators were more conservative, leading to a lower *sensitivity top 50%*, and tended to include false edges a bit more prominent in the network compared to regularized estimators.

We conclude that at a relatively high, though attainable, sample size of n = 1,000, regularized network estimators work best when the interest solely lies on the visual representation of the network using standard fading of edges. Using regularized estimators, strong edges are routinely included with an appropriate strength, and false edges are not very prominent in the network.

Centrality. Further, we investigated how well the three most common centrality indices (i.e., strength, closeness, and betweenness) of the estimated network structures align with the true network structure, at a low, but common sample size in social science (n = 600). Specifically,

we were interested in how well the estimated centrality measures correlate with the true centrality measures, as well as how well the centrality based on one dataset correlates with the centrality based on another dataset. We investigated in particular the metrics in SEA radar plots: *node strength correlation*, *closeness correlation*, and *betweenness correlations*, followed by the metrics *node strength correlation replication*, *closeness correlation replication*, and *betweenness correlation replication*.

Overall, the pattern of results indicates that *strength* as a centrality measure is the most likely measure to show both a good correlation coefficient between the true and simulated network structure, as well as good correlation replication. This holds both for unregularized methods, specifically for *ggmModselect stepwise*, as well as for regularized methods, specifically for the *EBICglasso* and *mgm (EBIC; gamma = 0.25)*. In the case of ordered categorical data in the BFI network, the performance of all estimators dropped for the node strength correlation, suggesting that for dense large network structures, ordered categorical data may pose more problematic centrality results. Measures for *closeness* performed worse than measures for *node strength*, and measures for *betweenness* performed even worse than those for *closeness*, displaying very low correlation and correlation replications across all estimators and all networks. The estimators that worked best were *ggmModselect stepwise* and *EBICglasso*. In the case of skewed ordered categorical data, estimator performance varied based on the chosen network.

We conclude that, if centrality is of interest with a relatively low sample size, using the *EBICglasso* estimator or unregularized *ggmModSelect stepwise* will likely give most confidence interpreting centrality indices. *Strength* as a centrality measure showed highest correlation values between the true and estimated network structures, as well as highest replication correlation, followed by *closeness*. In the case of skewed ordered categorical data, the correlations however dropped, indicating more caution is indeed when interpreting

centrality measures in the case of ordered categorical data. In addition, the performance of estimators varied more across networks for the case of centrality, thus we recommend checking for the network structure of interest. For the case of *betweenness*, our results showed that global properties of betweenness are likely hard to estimate, and as such we would not recommend anymore the use and interpretation of betweenness in network studies. Nonetheless, if *betweenness* is central to the research question of the applied researchers, taking an approach where investigating local properties of betweenness (e.g., by bootstrapping betweenness to see which particular betweenness results are more likely to come up often) should be common practice. Finally, and important, while here we interpret results for centrality in the case of MAGNA, the PTSD meta-analysis on which the MAGNA network is based showed little differences in centrality between nodes, especially for strength and closeness. As such, focusing on the latter two networks for the case of centrality may be warranted.

Bridge edges. Further, we focus on edges bridging different domains in a network structure. That is, we focus on desirable results obtained at frequent, but slightly higher sample sizes (n = 1000). This is a common question in applied research, especially in cases in which environmental (e.g., trauma) or genetic risk scores aim to be integrated into network models. We were thus interested in whether we can steadily discover bridging edges between different domains. In this setting we specifically focused on the *(top 25%) sensitivity*, *precision*, and *specificity* of *bridge* edges. We specifically operationalized this question by looking at the following properties in SEA radar plots: *n = 1000*, *sensitivity bridge*, *sensitivity bridge (top 25%), precision bridge*, and *specificity bridge*, to get information on correct detection of (strong) edges, as well as on whether the edges identified as bridging edges were true edges.

Of note, in the current case we only focused on two network structures, the DASS21 and the BFI, as these were the only ones including different domains (i.e., in the MAGNA network does not include bridge edges). Here the performance of estimators varied more across the two network structures, indicating that the topology of the network may play an important role in how the estimators perform, with slightly better performance for many of these in the BFI network. The main results indicate a strong exchange between sensitivity and specificity/precision in the case of bridging edges. Most estimators that perform well on one side, will perform less well on the other. The *EBICglasso* for instance shows good sensitivity of bridge edges, as well as extremely good sensitivity of strongest bridge edges, but lower precision and specificity, suggesting the presence of false positive edges in the identified bridges. *Mgm* on the other hand, especially using *EBIC (gamma = 0.25)*, as well as *ggmModSelect (stepwise = TRUE, gamma = 0)* show very good precision and specificity, but lower sensitivity, indicating that some bridging edges may be missed. In the case of the BFI network, the *ggmModSelect (stepwise = TRUE, gamma = 0)* algorithm perform very well on all counts, with very good precision and specificity of bridge edges, as well as good sensitivity (~.75) and extremely good sensitivity of strongest edges. While in the case of ordered categorical data the pattern of results is similar, the performance of the estimators is poorer than in the case of continuous data.

We conclude that, if bridging edges between different domains are of interest, first and foremost the network structure and topology may play an important role in the choice of estimators and the expectations on performance. In the case of the DASS21 network, this is a large and very dense network structure, with a high number of bridging edges between the different domains, which is an important consideration. In the case of the BFI, while still dense, given that all measures are related to personality traits, the bridging edges are stronger. The estimators perform better in the latter case, while in the case of the DASS21 network the

exchange between specificity and sensitivity is more prominent. Overall, we identified that the *mgm (EBIC; gamma = 0.25)* algorithm and the *ggmModSelect (stepwise = TRUE, gamma = 0)* algorithm performed consistently well on all conditions, while the *ggmModSelect (stepwise = TRUE, gamma = 0)* algorithm performed extremely good in the case of the DASS21 network with continuous data. The *ggmModSelect (stepwise = TRUE, gamma = 0)* is focused on obtaining a local optimum, meaning that no individual edges can be added or removed to improved fit. In this case, every bridge edge is also individually evaluated, this being an extra property of the estimator. As such, we recommend the use of the *ggmModSelect (stepwise = TRUE, gamma = 0)*, or *mgm (EBIC; gamma = 0.25)* for denser network structures when the expectation is that many and less strong bridging edges exist. Of note, if the interest is only on the strongest bridging edges, or on either side of specificity or sensitivity, most other estimators we evaluated performed very well.

Network replicability. Finally, we investigate how well network models estimated from one dataset replicate in a second dataset drawn from exactly the same distribution. To answer this, we simulated two datasets, and estimated a set of replicability measures that summarize the overlap between networks estimated from both datasets. We focus our discussion here on the sample sizes n = 600 (relatively low, similar to the sample size studied by Forbes et al., 2019), and n = 5,000 (high, similar to the sample size studied by Forbes et al., 2017). We specifically operationalized this question by looking at the following metrics in SEA radar plots: *replicated edges*, summarizing the proportion of edges that replicated in the second network, *replicated zeroes*, summarizing the proportion of absent edges that were also absent in the second network, *correlation replication*, summarizing the correlation between edge weights of both networks, and *replicated edges (top 25%)*, summarizing if the strongest 25% edges from the first network were also included in the second network (regardless of strength).

Across all data types and transformations, most estimators performed very similarly to one-another, with the exception of *EBICglasso* and *mgm (CV; 10 folds)*. These algorithms tended to perform worse than other methods on *replicated zeroes* and at, to a lesser extend, *replicated edges* (especially in the BFI high sample size condition). This performance is likely related to the poorer specificity of these algorithms compared to other algorithms discussed earlier: due to a larger number of falsely included edges less zeroes replicate, and the falsely included edges themselves also do not replicate in other networks. Further, across all types and transformations, the *replicated edges (top 25%)* metric was very high across the board, indicating that the strongest edges estimated in one network were very likely to also be included in a network estimated from a second dataset.

Across the conditions, there were some striking differences between the two *ggmModSelect* methods. At high sample size, the variant with stepwise estimation performed well (in combination with a transformation for skewed normal data) on all measures (although the least in the DASS21 network, which featured the most edges), while the variant without stepwise estimation performed worse on *replicated zeroes* in the DASS21 and MAGNA networks and both *replicated zeroes* and *replicated edges* in the BFI network. This performance may be related to the difference in specificity discussed earlier. At the lower sample size of $N = 600$, the variant without stepwise estimation performed slightly better than the stepwise variant on the *replicated edges* metric.

Using a non-paranormal or rank transformation, the *mgm (EBIC; gamma = 0.25)* algorithm performed consistently well on all conditions at the lower sample size of $N = 600$, although other metrics (in particular the *ggmModSelect* and *BGGM* variants) also performed well. Of note, performance on some of the measures was not very high, and often ranged around 0.5 - 0.75. It is arguable if, say, a rate of 60% replicated edges is good or not. At larger sample sizes, unregularized methods all performed quite well on all metrics.

We conclude that researchers should take expected replicability in the light of sensitivity and specificity (Williams, 2020), and not expect near-perfect replication of the network structure at low to medium sample sizes. Nonetheless, with most methods most edges should be replicated in a second dataset, and the overall structure should be similar. If the goal is to replicate individual edges, the *EBICglasso* and *mgm* with cross-validation algorithms should be avoided. At low sample sizes, the *mgm* with EBIC model selection performs well, as does *ggmModSelect*. At high sample sizes, one of the unregularized methods with desirable asymptotic properties should be used to optimize replicability.

4. Discussion

The current paper is, to our knowledge, the only large-scale simulation study designed to compare the performance of a wide array of estimation algorithms suitable for Gaussian and skewed ordered categorical data across a multitude of settings, as to arrive at concrete guidelines from applied researchers. All the results from our simulation study are made available through the *simulation explorer app* (SEA), accessible on Github[6]. Overall, we advise that the estimation method is best chosen in light of the research question(s) of the applied researcher, as some algorithms perform better in retrieving a lot of true edges, but may include false positive edges, while others may be more conservative and have a very low rate of false positive results, but concurrently fail in retrieving all edges and the global picture. Thus, the current paper highlighted results according to most common research questions in the field, alongside desirable asymptotic properties and low sample size discovery.

In particular, at high sample size ($n = 5,000$), we identified most estimators to work well, but unregularized estimators to work best in retrieving a network structure with a low

[6] https://github.com/AdelaIsvoranu/simulation_exploration_app

false inclusion rate of edges, high rate of identifying (the strongest) edges in the network structure, as well as high correlations between the estimated network structure and the true network structure. At low sample size (*n = 300*), if the goal is to discover a network structure that resembles a true network and to discover the strongest edges, regularized estimators should be preferred; if the goal however is to focus on each individual discovered edge, regularized estimators should be avoided.

In terms of specific research questions, we found that at an attainable sample size of *n=1,000*, if interested in the visual network alignment (i.e., investigating overall structure and strong edges), regularized network estimators work best. If interested in bridge edges between different domains, the topology of the network was important, with estimators performing better when strong bridging edges were present. When the expectation was that many and less strong bridging edges exist, which is often the case of denser network structures, we found the *ggmModSelect (stepwise = TRUE, gamma = 0)* and *mgm (EBIC; gamma =0.25)* estimators to perform best. Of note, if the interest lies only on the strongest bridging edges, or on either side of specificity or sensitivity, most other estimators we evaluated performed very well. When investigating centrality, at a low but common sample size in social science (*n = 600*), using the *EBICglasso* estimator or the unregularized *ggmModSelect stepwise* estimators gave most confidence interpreting centrality indices. *Strength* as a centrality measure showed best results, followed by *closeness*. Of note, in the case of skewed ordered categorical data, more caution is indeed when interpreting centrality measures. For the case of *betweenness*, our results showed that global properties of betweenness are likely hard to estimate, and as such we would not recommend the use and interpretation of betweenness in network studies. Alternatively, investigating local properties of betweenness should be common practice.

Finally, when data were Gaussian, applying a non-paranormal on rank-transformation (Spearman correlations as input) did not impact performance of the estimators. When data were skewed, a non-paranormal or rank-transformation improved the performance in the majority of estimators across most datasets. Since the latter worked well on all data types and comparably to the more complicated nonparanormal transformation, we recommend Spearman correlations as input in the case of skewed data. Surprisingly, for ordered categorical data, data transformation did not make a substantial difference.

4.1 Limitations

The current study should be considered in light of several limitations. First, the results of the simulations discussed in this paper rely critically on the network models used to generate data. We chose these three models to be representative of network models we may expect to find in psychological research (all three are based on a large sample of variables that would suit analysis using a psychological network model). Nonetheless, other generating network structures with different network characteristics may lead to different recommendations. To this end, we make an example code of the simulation script used in this study available in the online supplementary materials on the OSF.

Second, we generated data using a GGM with Gaussian variables, and subsequently transformed data to be ordered categorical (using thresholds) or skewed (using an exponential function). This introduces a critical assumption in the simulations, namely that the true underlying model of all simulated datasets *is* a multivariate Gaussian, possibly combined with a transformation function. These transformations may not preserve the variance-covariance structure, making it vital to apply also some form of transformation before estimating a GGM. This may be a reason why, especially in the skewed continuous conditions, monotone transformations (rank/Spearman and non-paranormal) work well, and

estimating a GGM on the untransformed data does not work well. Data can also be ordered categorical or skewed in different ways. For example, the data could follow different marginal distributions than the log-normal distributions used in skewed continuous cases, and ordered categorical data could also follow a network structure without relying on an underlying Gaussian distribution. As such, our results do not generalize to all ways in which data can be skewed and/or ordered categorical.

Third, when evaluating different estimation algorithms, we specifically tied the results to the software implementations of these algorithms. We can expect that our results will generalize to other software packages that implement the same algorithms, but cannot be sure as there may be small differences in the implementations in different software packages that will lead to different results (Epskamp, 2019). For example, regularized estimation can be implemented with and without an additional threshold for small edges to be included or not (the *mgm* package has such a threshold but the *qgraph* package does not), which will impact the specificity.

Fourth, the current simulation study relied on the default tuning parameters of the software used. Of note, however, the choice of tuning parameter may further impact the results and performance of these models. While further extending the already very large simulation study to vary differing tuning parameters was not feasible, future research may further address this issue.

Finally, network psychometrics is a rapidly developing field, and new estimation methods are proposed routinely. Future studies could include more novel estimation methods (e.g., Lafit et al., 2019; Williams, 2021), which could perform differently than the estimators currently assessed.

4.2 Conclusion

To conclude, regardless of the setting chosen, we found that an exchange between discovery (e.g., sensitivity, edge weight correlation) and caution (e.g., specificity, precision) should always be expected and achieving both—which is a requirement for perfect replicability—is difficult, even more so when bridging edges are of interest. Researchers should take expected replicability in the light of sensitivity and specificity and not expect near-perfect replication of the network structure at low to medium sample sizes. Nonetheless, with most methods most edges should be replicated in a second dataset, and the overall structure should be similar. Finally, the estimation method should be chosen in light of the research question(s) of the applied researcher, as highlighted above. Further simulation conditions, which can be used for investigating performance of estimators for additional research questions, are available through the SEA.

Acknowledgments

We would like to thank Denny Borsboom for helpful advice throughout the set-up of the project, as well as useful feedback on this manuscript.

Funding

This work was supported by the Netherlands Organisation for Scientific Research (NWO; A.M. I., grant number 406.16.516; S.E., grant number 016-195-261).

Conflict of interest

The authors have declared that there are no conflicts of interest in relation to the subject of this study.

References

Benet-Martínez, V., & John, O. P. (1998). Los Cinco Grandes across cultures and ethnic groups: Multitrait multimethod analyses of the Big Five in Spanish and English. *Journal of Personality and Social Psychology*, *75*(3), 729–750. https://doi.org/10.1037/0022-3514.75.3.729

Borsboom, D., Fried, E. I., Epskamp, S., Waldorp, L., Van Borkulo, C., Van Der Maas, H., & Cramer, A. (2017). False alarm? A comprehensive reanalysis of "Evidence that psychopathology symptom networks have limited replicability" by Forbes, Wright, Markon, and Krueger. *Journal of Abnormal Psychology*. https://doi.org/10.17605/OSF.IO/TGEZ8

Borsboom, D., Robinaugh, D. J., Rhemtulla, M., & Cramer, A. O. J. (2018). Robustness and replicability of psychopathology networks. *World Psychiatry*, *17*(2), 143–144. https://doi.org/10.1002/wps.20515

Boyette, L.-L., Isvoranu, A.-M., Schirmbeck, F., Velthorst, E., Simons, C. J. P., Barrantes-Vidal, N., Bressan, R., Kempton, M. J., Krebs, M.-O., McGuire, P., Nelson, B., Nordentoft, M., Riecher-Rössler, A., Ruhrmann, S., Rutten, B. P., Sachs, G., Valmaggia, L. R., van der Gaag, M., Borsboom, D., … van Os, J. (2020). From speech illusions to onset of psychotic disorder: Applying network analysis to an experimental measure of aberrant experiences. *Schizophrenia Bulletin Open*, *1*(1), sgaa025. https://doi.org/10.1093/schizbullopen/sgaa025

Chen, J., & Chen, Z. (2008). Extended Bayesian information criteria for model selection with large model spaces. *Biometrika*, *95*(3), 759–771. https://doi.org/10.1093/biomet/asn034

Choi, K. W., Batchelder, A. W., Ehlinger, P. P., Safren, S. A., & O'Cleirigh, C. (2017). Applying network analysis to psychological comorbidity and health behavior: Depression, PTSD, and sexual risk in sexual minority men with trauma histories.

Journal of Consulting and Clinical Psychology, *85*(12), 1158. https://doi.org/10.1037/ccp0000241

Costantini, G., Epskamp, S., Borsboom, D., Perugini, M., Mõttus, R., Waldorp, L. J., & Cramer, A. O. J. (2015). State of the aRt personality research: A tutorial on network analysis of personality data in R. *Journal of Research in Personality*, *54*, 13–29. https://doi.org/10.1016/j.jrp.2014.07.003

Digman, J. M. (1989). Five robust trait dimensions: Development, stability, and utility. *Journal of Personality*, *57*(2), 195–214. https://doi.org/10.1111/j.1467-6494.1989.tb00480.x

Drton, M., & Perlman, M. D. (2004). Model selection for Gaussian concentration graphs. *Biometrika*, *91*(3), 591–602. https://doi.org/10.1093/biomet/91.3.591

Epskamp, S. (2017). *Network Psychometrics, PhD dessertation*.

Epskamp, S. (2019). Reproducibility and replicability in a fast-paced methodological world. *Advances in Methods and Practices in Psychological Science*, *2*(2), 145–155. https://doi.org/10.1177/2515245919847421

Epskamp, S. (2020a). Psychometric network models from time-series and panel data. *Psychometrika*, *85*(1), 206–231. https://doi.org/10.1007/s11336-020-09697-3

Epskamp, S. (2020b). Psychonetrics. In *CRAN* (0.6). https://cran.r-project.org/web/packages/psychonetrics/index.html

Epskamp, S., Borsboom, D., & Fried, E. I. (2017). Estimating psychological networks and their accuracy: A tutorial paper. *Behavior Research Methods*, *50*(1), 195–212. https://doi.org/10.3758/s13428-017-0862-1

Epskamp, S., Cramer, A. O. J., Waldorp, L. J., Schmittmann, V. D., & Borsboom, D. (2012). qgraph : Network Visualizations of Relationships in Psychometric Data. *Journal of Statistical Software*, *48*(4), 1–18. https://doi.org/10.18637/jss.v048.i04

Epskamp, S., & Fried, E. I. (2018). A tutorial on regularized partial correlation networks. *Psychological Methods*, *23*(4), 617–634. https://doi.org/10.1037/met0000167

Epskamp, S., Isvoranu, A.-M., & Cheung, M. W. L. (2020). Meta-analytic Gaussian Network Aggregation. *PsyArXiv (Preprint)*, https://psyarxiv.com/236w8/. https://doi.org/10.31234/osf.io/236w8

Epskamp, S., Kruis, J., & Marsman, M. (2017). Estimating psychopathological networks: Be careful what you wish for. *PLoS ONE*, *12*(6), E0179891. https://doi.org/10.1371/journal.pone.0179891

Epskamp, S., Rhemtulla, M., & Borsboom, D. (2017). Generalized Network Psychometrics: Combining network and latent variable models. *Psychometrika*, *82*(4), 904–927. https://doi.org/10.1007/s11336-017-9557-x

Epskamp, S., Waldorp, L. J., Mõttus, 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/doi.org/10.1080/00273171.2018.1454823

Fonseca-Pedrero, E., Ortuño, J., Debbané, M., Chan, R. C. K., Cicero, D., Zhang, L. C., Brenner, C., Barkus, E., Linscott, R. J., Kwapil, T., Barrantes-Vidal, N., Cohen, A., Raine, A., Compton, M. T., Tone, E. B., Suhr, J., Inchausti, F., Bobes, J., Fumero, A., … Fried, E. I. (2018). The network structure of schizotypal personality traits. *Schizophrenia Bulletin*, *44*(suppl_2), S468–S479. https://doi.org/10.1093/schbul/sby044

Forbes, M. K., Wright, A. G. C., Markon, K. E., & Krueger, R. F. (2017). Evidence that psychopathology symptom networks have limited replicability. *Journal of Abnormal Psychology*, *126*(7), 969. https://doi.org/10.1037/abn0000276

Forbes, M. K., Wright, A. G. C., Markon, K. E., & Krueger, R. F. (2019). The network approach to psychopathology: promise versus reality. *World Psychiatry*, *18*(3), 272–273. https://doi.org/10.1002/wps.20659

Foygel, R., & Drton, M. (2011). Bayesian model choice and information criteria in sparse generalized linear models. *ArXiv (Preprint)*, 1–37. http://arxiv.org/abs/1112.5635

Foygel, R., & Drton, M. (2010). Extended Bayesian information criteria for Gaussian graphical models. *Advances in Neural Information Processing Systems*, 604–612. https://doi.org/10.1.1.231

Fried, E. I., Bockting, C., Arjadi, R., Borsboom, D., Amshoff, M., Cramer, A. O. J., Epskamp, S., Tuerlinckx, F., Carr, D., & Stroebe, M. (2015). From loss to loneliness: The relationship between bereavement and depressive symptoms. *Journal of Abnormal Psychology*, *124*(2), 256–265. https://doi.org/10.1037/abn0000028

Fried, E. I., Eidhof, M. B., Palic, S., Costantini, G., Huisman-van Dijk, H. M., Bockting, C. L. H., Engelhard, I., Armour, C., Nielsen, A. B. S., & Karstoft, K. I. (2018). Replicability and generalizability of posttraumatic stress disorder (PTSD) networks: A cross-cultural multisite study of PTSD symptoms in four trauma patient samples. *Clinical Psychological Science*, *6*(3), 335–351. https://doi.org/10.1177/2167702617745092

Fried, E. I., & Nesse, R. M. (2015). Depression is not a consistent syndrome: An investigation of unique symptom patterns in the STAR∗D study. *Journal of Affective Disorders*, *172*, 96–102. https://doi.org/10.1016/j.jad.2014.10.010

Friedman, J., Hastie, T., & Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, *9*(3), 432–441. https://doi.org/10.1093/biostatistics/kxm045

Goldberg, L. R. (1990). An alternative "description of personality": The Big-Five factor structure. *Journal of Personality and Social Psychology*, *59*(6), 1216–1229. https://doi.org/10.1037/0022-3514.59.6.1216

Goldberg, L. R. (1993). The Structure of phenotypic personality traits. *American*

Psychologist, *48*(1), 26–34. https://doi.org/10.1037/0003-066X.48.1.26

Haslbeck, J. M. B. B., & Waldorp, L. J. (2020). mgm: Structure Estimation for Time-Varying Mixed Graphical Models in High-dimensional Data. *Journal of Statistical Software*, *93*(8), 1–46. https://doi.org/10.18637/jss.v093.i08

Hoekstra, R. H. A., Epskamp, S., & Borsboom, D. (2020). Heterogeneity in individual network analysis: Reality or illusion? *Masuscript Submitted for Publication*, Department of Psychology, University of Amsterdam.

Hoff, P. (2007). Extending the rank likelihood for semiparametric copula estimation. *The Annals of Applied Statistics*, *1*(1), 265–283. https://doi.org/10.1214/07-aoas107

Isvoranu, A.-M., Borsboom, D., van Os, J., & Guloksuz, S. (2016). A network approach to environmental impact in psychotic disorder: Brief theoretical framework. *Schizophrenia Bulletin*, *42*(4), 870–873. https://doi.org/10.1093/schbul/sbw049

Isvoranu, A.-M., Epskamp, S., & Cheung, M. W. L. (2020). Network models of post-traumatic stress disorder: A meta-analysis. *PsyArXiv (Preprint)*, https://psyarxiv.com/8k4u6. https://doi.org/10.31234/osf.io/8k4u6

Isvoranu, A.-M., Guloksuz, S., Epskamp, S., van Os, J., Borsboom, D., & GROUP. (2019). Toward incorporating genetic risk scores into symptom networks of psychosis. *Psychological Medicine*, *50*(4), 636–643.

Isvoranu, A.-M., van Borkulo, C. D., Boyette, L.-L. L., Wigman, J. T. W. T. W., Vinkers, C. H. H., Borsboom, D., Kahn, R., De Haan, L., Van Os, J., Wiersma, D., Bruggeman, R., Cahn, W., Meijer, C., & Myin-Germeys, I. (2017). A network approach to psychosis: Pathways between childhood trauma and psychotic symptoms. *Schizophrenia Bulletin*, *43*(1), 187–196. https://doi.org/10.1093/schbul/sbw055

Kan, K. J., van der Maas, H. L. J., & Levine, S. Z. (2019). Extending psychometric network analysis: Empirical evidence against g in favor of mutualism? *Intelligence*, *73*, 52–62.

https://doi.org/10.1016/j.intell.2018.12.004

Lafit, G., Tuerlinckx, F., Myin-Germeys, I., & Ceulemans, E. (2019). A partial correlation screening approach for controlling the false positive rate in sparse Gaussian graphical models. *Scientific Reports*, *9*(1), 1–24. https://doi.org/10.1038/s41598-019-53795-x

Lauritzen, S. L. (1996). *Graphical models* (17th ed.). Clarendon.

Lazarov, A., Suarez-Jimenez, B., Levy, O., Coppersmith, D. D. L., Lubin, G., Pine, D. S., Bar-Haim, Y., Abend, R., & Neria, Y. (2019). Symptom structure of PTSD and co-morbid depressive symptoms - A network analysis of combat veteran patients. *Psychological Medicine*, *50*(13), 2154–2170. https://doi.org/10.1017/S0033291719002034

Liu, H., Lafferty, J., & Wasserman, L. (2009). The nonparanormal: semiparametric estimation of high dimensional undirected graphs. *The Journal of Machine Learning Research*, *10*, 2295–2328. https://doi.org/10.1016/0006-291X(91)91267-G

Lovibond, P. F., & Lovibond, S. H. (1995). The structure of negative emotional states: Comparison of the Depression Anxiety Stress Scales (DASS) with the Beck Depression and Anxiety Inventories. *Behaviour Research and Therapy*, *33*(3), 335–343. https://doi.org/10.1016/0005-7967(94)00075-U

Malgaroli, M., Maccallum, F., & Bonanno, G. A. (2018). Symptoms of persistent complex bereavement disorder, depression, and PTSD in a conjugally bereaved sample: A network analysis. *Psychological Medicine*, *48*(14), 2439–2448. https://doi.org/10.1017/S0033291718001769

Mansueto, A. C., Wiers, W. R., van Weert, J. C. M., Schouten, B. C., & Epskamp, S. (2020). Investigating the feasibility of idiographic network models. *PsyArXiv (Preprint)*, https://psyarxiv.com/hgcz6/. https://doi.org/10.31234/osf.io/hgcz6

McCrae, R. R., & Costa, P. T. (1997). Personality trait structure as a human universal. *The*

American Psychologist, *52*(5), 509–516. https://doi.org/10.1037/0003-066X.52.5.509

McNally, R. J., Robinaugh, D. J., Wu, G. W. Y., Wang, L., Deserno, M., & Borsboom, D. (2015). Mental disorders as causal systems: A network approach to posttraumatic stress disorder. *Clinical Psychological Science*, *3*(6), 836–849. https://doi.org/10.1177/2167702614553230

Meinshausen, N., Meier, L., & Bühlmann, P. (2009). p-Values for high-dimensional regression. *Journal of the American Statistical Association*, *104*(488), 1671–1681. https://doi.org/10.1198/jasa.2009.tm08647

Mulder, J., & Pericchi, L. R. (2018). The matrix-F prior for estimating and testing covariance matrices. *Bayesian Analysis*, *13*(4), 1193–1214. https://doi.org/10.1214/17-BA1092

Muthén, B. (1984). A general structural equation model with dichotomous, ordered categorical, and continuous latent variable indicators. *Psychometrika*, *49*(1), 115–132.

Olsson, U. (1979). Maximum likelihood estimation of the polychoric correlation coefficient. *Psychometrika*, *44*(4), 441–460. https://doi.org/10.1007/BF02296207

Open Science Collaboration. (2015). Estimating the reproducibility of psychological science. *Science*, *349*(6251), 943–951. https://doi.org/10.1126/science.aac4716

Revelle, W. (2015). Package “psych” - Procedures for psychological, psychometric and personality research. In *CRAN*. https://cran.r-project.org/web/packages/psych/

Robinaugh, D. J., Hoekstra, R. H. A., Toner, E. R., & Borsboom, D. (2020). The network approach to psychopathology: A review of the literature 2008-2018 and an agenda for future research. *Psychological Medicine*, *50*(3), 353–366. https://doi.org/10.1017/S0033291719003404

Robinaugh, D. J., Millner, A. J., & McNally, R. J. (2016). Identifying highly influential nodes in the complicated grief network. *Journal of Abnormal Psychology*, *125*(6), 747. https://doi.org/10.1037/abn0000181

Rosseel, Y. (2012). lavaan: An R package for structural equation modeling. *Journal of Statistical Software*, *48*(2), 1–36.

Vanzhula, I. A., Calebs, B., Fewell, L., & Levinson, C. A. (2019). Illness pathways between eating disorder and post-traumatic stress disorder symptoms: Understanding comorbidity with network analysis. *European Eating Disorders Review*, *27*(2), 147–160. https://doi.org/10.1002/erv.2634

Williams, D. R. (2020). Learning to live with sampling variability: Expected replicability in partial correlation networks. *PsyArXiv (Preprint)*, https://psyarxiv.com/fb4sa/. https://doi.org/10.31234/osf.io/fb4sa

Williams, D. R. (2021a). Bayesian estimation for Gaussian graphical models: Structure learning, predictability, and network comparisons. *Multivariate Behavioral Research*, 1–17.

Williams, D. R. (2021b). Beyond Lasso: A survey of nonconvex regularization in Gaussian graphical models. *PsyArXiv (Preprint)*, https://psyarxiv.com/ad57p/. https://doi.org/10.31234/osf.io/ad57p

Williams, D. R., & Mulder, J. (2019). BGGM : A R Package for Bayesian Gaussian graphical models. *PsyArXiv (Preprint)*, https://psyarxiv.com/3b5hf/. https://doi.org/10.31234/osf.io/3b5hf

Williams, D. R., & Mulder, J. (2020). BGGM: Bayesian Gaussian graphical models in R. *Journal of Open Source Software*, *5*(51), 2111. https://doi.org/10.31234/osf.io/t2cn7

Williams, D. R., & Rast, P. (2020). Back to the basics: Rethinking partial correlation network methodology. *Journal of Mathematical and Statistical Psychology*, *73*(2), 187–212. https://doi.org/10.17605/OSF.IO/FNDRU

Williams, D. R., Rhemtulla, M., Wysocki, A. C., & Rast, P. (2019). On Nonregularized Estimation of Psychological Networks. *Multivariate Behavioral Research*, *54*(5), 719–

750. https://doi.org/10.1080/00273171.2019.1575716

Winston, C., Cheng, J., Allaire, J., Xie, Y., & McPherson, J. (2017). Shiny: Web application framework for R. In *CRAN*. https://cran.r-project.org/package=shiny

Table 1. Estimation methods used in the simulation study

Method	R package	Functions	Settings	Ordinal/Categorical
EBICglasso	*qgraph*	EBICglasso	$\gamma = 0.5$	polychoric correlations as input
ggmModSelect	*qgraph*	ggmModSelect	stepwise = TRUE	polychoric correlations as input
ggmModSelect_stepwise	*qgraph*	ggmModSelect	stepwise = FALSE	polychoric correlations as input
FIML_prune	*psychonetrics*	ggm %>% prune	$\alpha = 0.01$	N/A
FIML_prune_modelsearch	*psychonetrics*	ggm %>% prune %>% modelsearch	$\alpha = 0.01$	N/A
WLS_prune	*psychonetrics*	ggm %>% prune	$\alpha = 0.01$	three-stage WLS (Muthén, 1984)
WLS_prune_stepup	*psychonetrics*	ggm %>% prune %>% stepup	$\alpha = 0.01$	three-stage WLS (Muthén, 1984)
mgm_CV	*mgm*	mgm	Selection via 10-fold cross-validation	variables treated categorical
mgm_EBIC	*mgm*	mgm	Selection via EBC ($\gamma = 0.5$)	variables treated categorical
BGGM_explore	BGGM	explore %>% select	BF cutoff = 3	variables treated as ordinal
BGGM_estimate	BGGM	estimate %>% select	95% credibility interval	variables treated as ordinal
GGM_bootstrap	GMMnonreg	GGM_bootstrap	$\alpha = 0.01$	N/A
GGM_regression	GMMnonreg	GGM_regression	BIC optimization	N/A

Table 2. Characteristics of the three network models used in the simulation study.

Graph	# Nodes	# Clusters	Sparsity (proportion of zeroes)	Average absolute edge-weight	smallword index	Average path length
DASS21	21	3	0.41	0.78	1.17	1.39
PTSD	17	1	0.46	0.09	1.10	1.43
BFI	25	5	0.64	0.13	1.10	1.63

Table 3. Operationalizations of different settings discussed in more detail in the paper. The column "Operationalization" shows the settings that can be used in the SEA app to obtain the simulation results that support our conclusions.

Setting	Operationalization
Desirable asymptotic properties	*n = 5,000* *specificity* *precision* *sensitivity* *sensitivity (top 50%)* *absolute correlation* *1-bias*
Low sample size discovery	*n = 300* *sensitivity (top 25%)* *precision* *fade maximum false edge*
Visual network alignment	*n* = 1,000 *sensitivity (top 50%)* *correlation (absolute)* *1 – bias (true edges)* *fade maximum false edge*
Centrality	*n* = 600 *node strength correlation* *closeness correlation* *betweenness correlations* *node strength correlation replication* *closeness correlation replication* *betweenness correlation replication*
Bridge edges	*n = 1,000* *sensitivity bridge* *sensitivity bridge (top 25%)* *precision bridge* *specificity bridge*
Network replicability	*n* = 600 & *n* = 5,000 *replicated edges* *replicated zeroes* *correlation replication* *replicated edges (top 25%)*

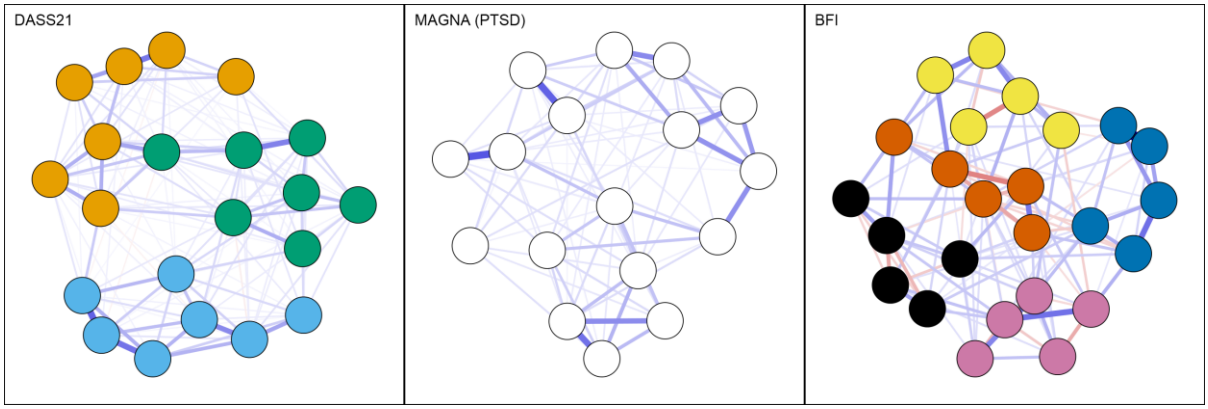

Figure 1: Network structures under which data were generated.

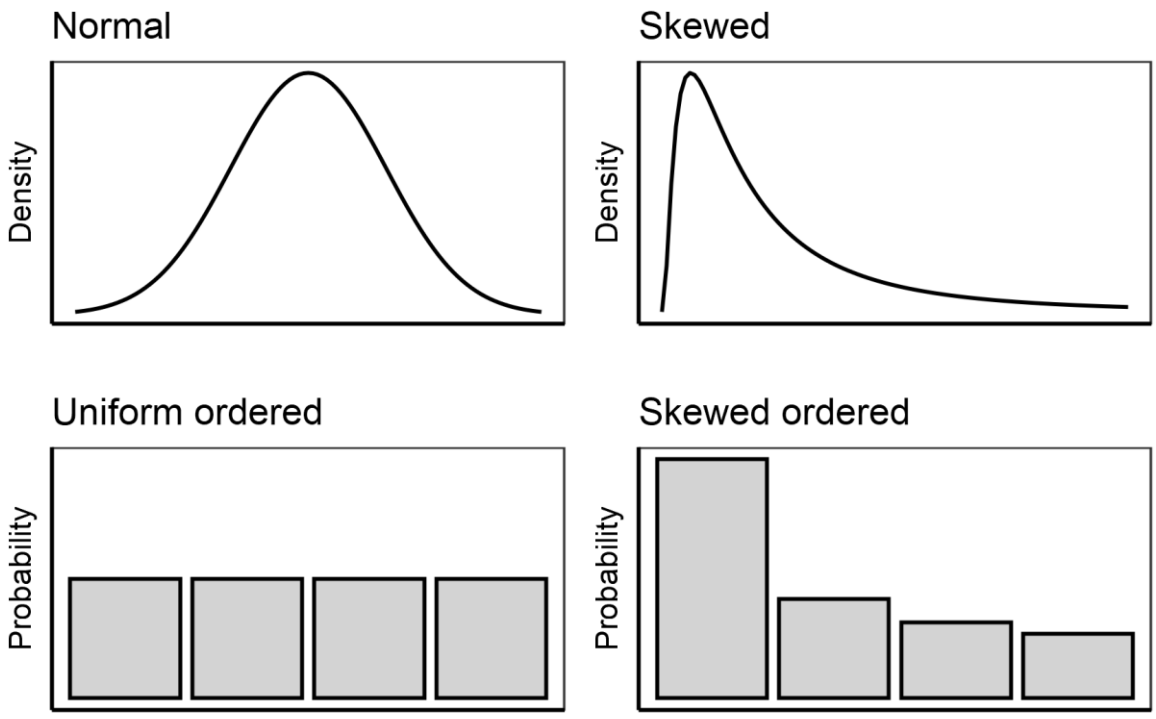

Figure 2: Types of data generated in the simulation studies. All data were first generated as continues normally distributed data, after which the data was transformed to skewed data through the exponential function and to ordered-categorical data through threshold models.

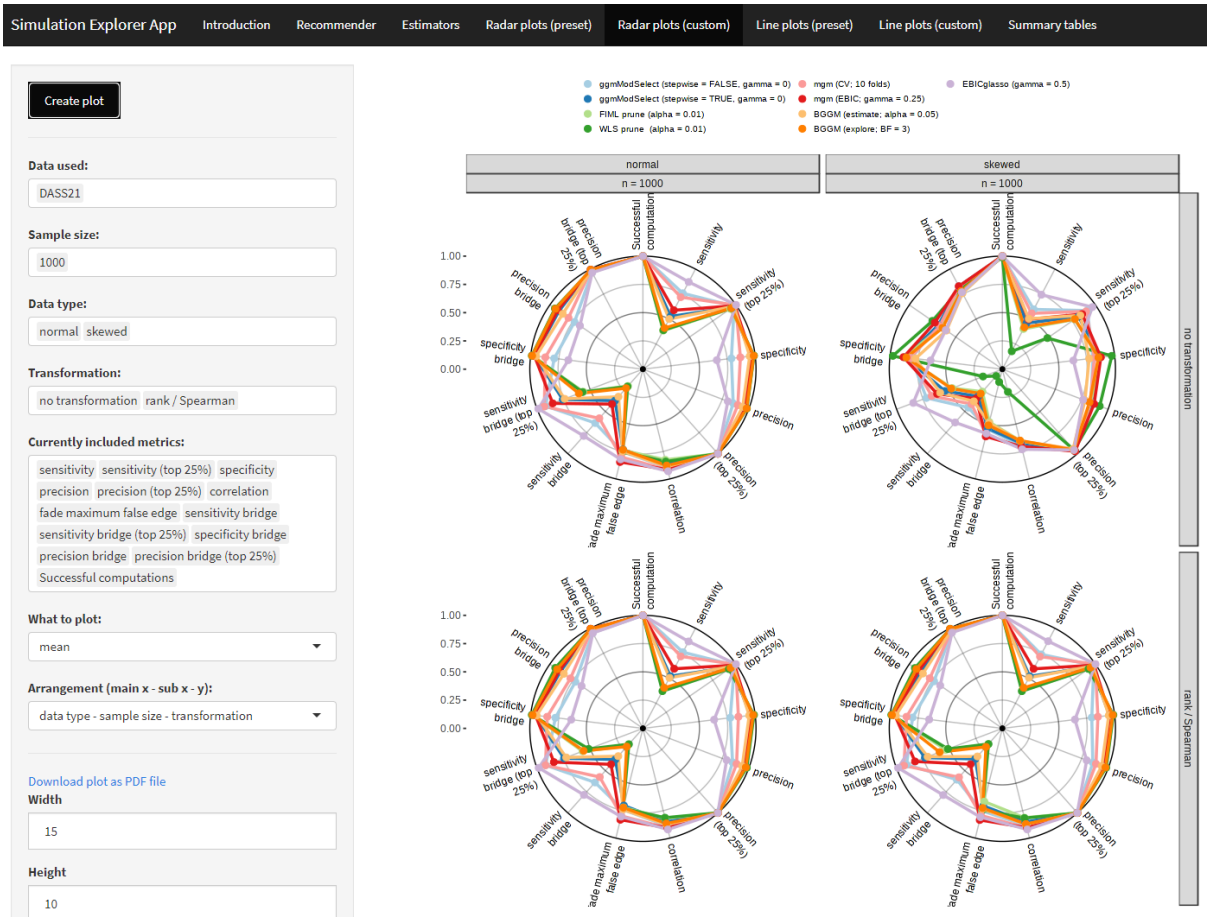

Figure 3. Example of output provided by the simulation exploration app (SEA) available at psychonetrics.org/simulations (source code: github.com/AdelaIsvoranu/simulation_exploration_app).

Appendix A. Data generating model

This section describes how we generated data used in the simulation study. Given a GGM structure encoded in a square matrix $\boldsymbol{\Omega}$ (a matrix with zeroes on the diagonal and partial correlation coefficients on the off-diagonal elements), we first transformed this matrix into an expected correlation matrix $\boldsymbol{P}$ using the following expression (Epskamp et al., 2020; Epskamp, Rhemtulla, et al., 2017):

$$\boldsymbol{P} = \boldsymbol{\Delta}(\boldsymbol{I} - \boldsymbol{\Omega})^{-1}\boldsymbol{\Delta},$$

with $\boldsymbol{\Delta}$ chosen such that the diagonal elements of $\boldsymbol{P}$ are all 1. Next, for case c we generated a latent response-vector $\boldsymbol{\theta_c}$ from a multivariate normal distribution with zero mean-vector and variance–covariance matrix $\boldsymbol{\Sigma}$. The latent response-vector $\boldsymbol{\theta_c}$ was subsequently transformed into observed score-vector $\boldsymbol{y_c}$ depending on the data generation condition. For the normal continuous data condition, we simply used an identity link:

$$\boldsymbol{y_c} = \boldsymbol{\theta_c}.$$

For the skewed continuous data condition, we used the exponential function:

$$\boldsymbol{y_c} = exp(\boldsymbol{\theta_c}),$$

which indicates a log-normal distribution.

For the ordered categorical data conditions, we made use of threshold models (Muthén, 1984). We used a threshold model using thresholds $\tau_0 = -\infty, \tau_1, \tau_2, \tau_3, \tau_4 = \infty$, such that:

$$y_{ci} = \text{if } y_{ci} \geq \tau_{i-1} \text{ and } y_{ci} \leq \tau_i$$

As such, using a model with these thresholds will lead to ordered categorical data with levels 1, 2, 3, and 4. We used the following thresholds for the uniform ordered categorical data condition:

$$\tau_1 = -0.67, \tau_2 = 0, \tau_3 = 0.67.$$

And the following thresholds for the skewed categorical condition:

$$\tau_1 = 0, \tau_2 = 0.55, \tau_3 = 1.10.$$

Appendix B. Subset of simulation results

Table A1. A subset of simulation results for the MAGNA network, based on Gaussian data estimated without using a transformation. Shown are the averages of the following metrics: sensitivity (1), specificity (6), precision (7), correlation between absolute true and estimated edge-weights (12), fade maximum false edge (15), the correlation between true and estimated node strength (16), closeness (21) and betweenness (26), specificity of bridge edges (36), precision of bridge edges (37), correlation of edge weights between networks estimated from two datasets (41), the proportion of replicated edges in a second dataset (42), and the proportion of replicated zeroes in a second dataset (46).

	N	(1)	(6)	(7)	(12)	(15)	(16)	(21)	(26)	(36)	(37)	(41)	(42)	(46)
ggmModSelect (stepwise = FALSE, gamma = 0)	n = 150	0.39	0.93	0.89	0.72	0.54	0.69	0.15	0.24	.	.	0.55	0.54	0.86
	n = 300	0.49	0.92	0.90	0.82	0.65	0.82	0.26	0.38	.	.	0.70	0.66	0.85
	n = 600	0.60	0.90	0.90	0.87	0.73	0.88	0.39	0.46	.	.	0.80	0.73	0.84
	n = 1000	0.70	0.89	0.90	0.91	0.80	0.90	0.65	0.55	.	.	0.84	0.78	0.81
	n = 2500	0.86	0.85	0.89	0.96	0.88	0.96	0.81	0.71	.	.	0.93	0.86	0.81
	n = 5000	0.93	0.82	0.87	0.98	0.92	0.98	0.90	0.82	.	.	0.97	0.91	0.83
ggmModSelect (stepwise = TRUE, gamma = 0)	n = 150	0.34	0.94	0.88	0.68	0.50	0.65	0.47	0.26	.	.	0.49	0.48	0.85
	n = 300	0.43	0.94	0.91	0.77	0.59	0.77	0.52	0.34	.	.	0.62	0.57	0.84
	n = 600	0.52	0.95	0.94	0.85	0.69	0.86	0.63	0.47	.	.	0.74	0.64	0.84
	n = 1000	0.60	0.96	0.95	0.90	0.74	0.90	0.68	0.57	.	.	0.82	0.72	0.85
	n = 2500	0.75	0.98	0.98	0.95	0.83	0.96	0.83	0.71	.	.	0.91	0.83	0.87
	n = 5000	0.86	0.99	0.99	0.98	0.88	0.98	0.91	0.80	.	.	0.96	0.90	0.90
FIML prune (alpha = 0.01)	n = 150	0.16	0.98	0.92	0.66	0.59	0.50	0.00	0.18	.	.	0.51	0.45	0.94
	n = 300	0.23	0.99	0.96	0.76	0.65	0.62	0.12	0.32	.	.	0.64	0.59	0.93
	n = 600	0.35	0.99	0.97	0.82	0.74	0.73	0.41	0.41	.	.	0.71	0.66	0.91
	n = 1000	0.46	0.99	0.98	0.87	0.78	0.81	0.59	0.45	.	.	0.79	0.73	0.89
	n = 2500	0.68	0.99	0.99	0.94	0.84	0.92	0.78	0.63	.	.	0.90	0.81	0.88
	n = 5000	0.85	0.99	0.99	0.98	0.89	0.97	0.90	0.78	.	.	0.96	0.90	0.90
FIML prune -> modelsearch (alpha = 0.01)	n = 150	0.34	0.94	0.88	0.67	0.50	0.63	0.37	0.23	.	.	0.48	0.47	0.85
	n = 300	0.42	0.95	0.92	0.77	0.59	0.75	0.50	0.35	.	.	0.62	0.57	0.85
	n = 600	0.51	0.95	0.94	0.85	0.68	0.86	0.63	0.46	.	.	0.73	0.65	0.84
	n = 1000	0.60	0.96	0.96	0.90	0.76	0.90	0.70	0.55	.	.	0.82	0.72	0.85
	n = 2500	0.75	0.97	0.97	0.95	0.83	0.96	0.83	0.70	.	.	0.91	0.82	0.86
	n = 5000	0.87	0.98	0.99	0.98	0.88	0.98	0.91	0.83	.	.	0.96	0.90	0.90
WLS prune (alpha = 0.01)	n = 150	0.21	0.96	0.88	0.55	0.46	0.35	0.05	0.21	.	.	0.33	0.44	0.91
	n = 300	0.25	0.98	0.96	0.73	0.52	0.49	0.15	0.32	.	.	0.61	0.58	0.93
	n = 600	0.35	0.99	0.97	0.83	0.70	0.65	0.36	0.35	.	.	0.75	0.67	0.91
	n = 1000	0.45	0.99	0.98	0.88	0.78	0.75	0.58	0.44	.	.	0.80	0.73	0.90
	n = 2500	0.69	0.99	0.99	0.94	0.86	0.92	0.78	0.61	.	.	0.90	0.82	0.88
	n = 5000	0.85	0.99	0.99	0.98	0.89	0.97	0.89	0.77	.	.	0.96	0.90	0.90
WLS prune -> stepup (alpha = 0.01)	n = 150	0.30	0.89	0.79	0.52	0.32	0.38	0.19	0.18	.	.	0.32	0.39	0.84
	n = 300	0.38	0.92	0.86	0.72	0.53	0.56	0.36	0.22	.	.	0.55	0.50	0.83
	n = 600	0.51	0.93	0.91	0.83	0.68	0.78	0.58	0.41	.	.	0.71	0.62	0.82
	n = 1000	0.60	0.95	0.94	0.89	0.75	0.87	0.68	0.53	.	.	0.80	0.69	0.83

	n = 2500	0.77	0.96	0.96	0.95	0.84	0.95	0.83	0.70	.	.	0.92	0.82	0.85
	n = 5000	0.90	0.98	0.98	0.98	0.89	0.98	0.91	0.82	.	.	0.96	0.91	0.90
mgm (CV; 10 folds)	n = 150	0.44	0.89	0.85	0.73	0.59	0.60	0.36	0.24	.	.	0.55	0.53	0.79
	n = 300	0.58	0.85	0.84	0.82	0.68	0.68	0.51	0.35	.	.	0.70	0.61	0.75
	n = 600	0.72	0.83	0.85	0.90	0.77	0.81	0.66	0.48	.	.	0.82	0.70	0.72
	n = 1000	0.81	0.82	0.86	0.93	0.81	0.87	0.76	0.58	.	.	0.88	0.76	0.72
	n = 2500	0.93	0.81	0.87	0.97	0.88	0.94	0.88	0.76	.	.	0.95	0.85	0.74
	n = 5000	0.98	0.80	0.87	0.99	0.91	0.97	0.93	0.84	.	.	0.97	0.88	0.78
mgm (EBIC; gamma = 0.25)	n = 150	0.18	0.99	0.97	0.69	0.65	0.51	0.00	0.17	.	.	0.59	0.53	0.94
	n = 300	0.34	0.98	0.95	0.82	0.75	0.68	0.10	0.32	.	.	0.75	0.63	0.89
	n = 600	0.52	0.96	0.95	0.89	0.81	0.83	0.43	0.48	.	.	0.85	0.71	0.87
	n = 1000	0.65	0.95	0.95	0.93	0.86	0.86	0.76	0.63	.	.	0.89	0.76	0.83
	n = 2500	0.87	0.93	0.94	0.97	0.90	0.94	0.88	0.77	.	.	0.95	0.87	0.85
	n = 5000	0.96	0.92	0.94	0.99	0.93	0.97	0.92	0.83	.	.	0.98	0.92	0.88
GGM_bootstrap (alpha = 0.01)	n = 150	0.13	0.99	0.94	0.62	0.39	0.34	0.01	0.16	.	.	0.49	0.45	0.95
	n = 300	0.23	0.99	0.96	0.77	0.55	0.54	0.04	0.26	.	.	0.67	0.57	0.93
	n = 600	0.34	0.99	0.97	0.85	0.68	0.66	0.44	0.41	.	.	0.78	0.66	0.91
	n = 1000	0.46	0.99	0.98	0.90	0.76	0.76	0.68	0.53	.	.	0.84	0.71	0.90
	n = 2500	0.69	0.99	0.99	0.95	0.84	0.88	0.86	0.72	.	.	0.92	0.82	0.88
	n = 5000	0.85	0.99	0.99	0.98	0.89	0.95	0.92	0.83	.	.	0.96	0.90	0.90
GGM_regression	n = 150	0.24	0.98	0.94	0.69	0.49	0.55	0.15	0.24	.	.	0.53	0.48	0.91
	n = 300	0.32	0.98	0.96	0.79	0.58	0.69	0.42	0.36	.	.	0.65	0.57	0.90
	n = 600	0.43	0.99	0.98	0.86	0.68	0.80	0.58	0.44	.	.	0.77	0.67	0.89
	n = 1000	0.51	0.99	0.98	0.90	0.74	0.85	0.68	0.54	.	.	0.83	0.73	0.88
	n = 2500	0.68	0.99	0.99	0.95	0.83	0.92	0.85	0.73	.	.	0.92	0.83	0.88
	n = 5000	0.82	1.00	1.00	0.98	0.87	0.96	0.92	0.81	.	.	0.96	0.89	0.90
BGGM (estimate; alpha = 0.05)	n = 150	0.21	0.96	0.87	0.63	0.44	0.40	0.05	0.22	.	.	0.49	0.46	0.91
	n = 300	0.33	0.95	0.90	0.77	0.58	0.57	0.33	0.30	.	.	0.65	0.55	0.89
	n = 600	0.46	0.95	0.93	0.86	0.70	0.69	0.57	0.41	.	.	0.78	0.64	0.86
	n = 1000	0.59	0.95	0.94	0.91	0.78	0.79	0.72	0.54	.	.	0.84	0.71	0.84
	n = 2500	0.80	0.95	0.96	0.96	0.85	0.89	0.86	0.74	.	.	0.93	0.83	0.84
	n = 5000	0.92	0.94	0.96	0.98	0.89	0.95	0.92	0.82	.	.	0.97	0.90	0.88
BGGM (explore; BF = 3)	n = 150	0.21	0.96	0.87	0.62	0.42	0.34	0.05	0.16	.	.	0.48	0.45	0.91
	n = 300	0.29	0.97	0.94	0.77	0.59	0.53	0.24	0.30	.	.	0.66	0.55	0.90
	n = 600	0.39	0.98	0.97	0.86	0.69	0.69	0.50	0.47	.	.	0.78	0.67	0.90
	n = 1000	0.48	0.98	0.97	0.90	0.75	0.77	0.70	0.53	.	.	0.85	0.72	0.90
	n = 2500	0.67	0.99	0.99	0.95	0.84	0.87	0.86	0.72	.	.	0.92	0.81	0.88
	n = 5000	0.83	0.99	0.99	0.98	0.88	0.94	0.92	0.83	.	.	0.96	0.90	0.90
EBICglasso (gamma = 0.5)	n = 150	0.20	0.93	0.88	0.33	0.68	0.28	0.09	0.09	.	.	0.11	0.23	0.92
	n = 300	0.82	0.63	0.75	0.85	0.71	0.82	0.58	0.39	.	.	0.74	0.72	0.56
	n = 600	0.89	0.62	0.76	0.91	0.78	0.89	0.72	0.57	.	.	0.85	0.78	0.55
	n = 1000	0.94	0.62	0.77	0.94	0.82	0.93	0.77	0.62	.	.	0.90	0.82	0.57
	n = 2500	0.98	0.60	0.77	0.97	0.89	0.97	0.88	0.77	.	.	0.95	0.85	0.59
	n = 5000	0.99	0.60	0.77	0.99	0.92	0.98	0.94	0.85	.	.	0.98	0.87	0.59

Table A2. A subset of simulation results for the DASS21 network, based on Gaussian data estimated without using a transformation. Shown are the averages of the following metrics: sensitivity (1), specificity (6), precision (7), correlation between absolute true and estimated edge-weights (12), fade maximum false edge (15), the correlation between true and estimated node strength (16), closeness (21) and betweenness (26), specificity of bridge edges (36), precision of bridge edges (37), correlation of edge weights between networks estimated from two datasets (41), the proportion of replicated edges in a second dataset (42), and the proportion of replicated zeroes in a second dataset (46).

	N	(1)	(6)	(7)	(12)	(15)	(16)	(21)	(26)	(36)	(37)	(41)	(42)	(46)
ggmModSelect (stepwise = FALSE, gamma = 0)	n = 150	0.41	0.92	0.90	0.68	0.57	0.77	0.13	0.24	0.92	0.75	0.51	0.63	0.85
	n = 300	0.54	0.90	0.90	0.78	0.62	0.83	0.29	0.29	0.90	0.78	0.63	0.71	0.85
	n = 600	0.67	0.85	0.88	0.85	0.70	0.87	0.50	0.38	0.85	0.76	0.75	0.78	0.83
	n = 1000	0.76	0.79	0.85	0.89	0.75	0.90	0.76	0.53	0.79	0.73	0.82	0.82	0.79
	n = 2500	0.86	0.73	0.84	0.95	0.84	0.93	0.89	0.74	0.73	0.72	0.92	0.86	0.72
	n = 5000	0.95	0.68	0.83	0.98	0.90	0.96	0.95	0.89	0.68	0.72	0.96	0.86	0.68
ggmModSelect (stepwise = TRUE, gamma = 0)	n = 150	0.30	0.95	0.90	0.63	0.49	0.68	0.54	0.26	0.95	0.77	0.41	0.43	0.85
	n = 300	0.37	0.95	0.92	0.72	0.58	0.79	0.62	0.28	0.95	0.80	0.54	0.52	0.84
	n = 600	0.46	0.96	0.95	0.81	0.66	0.87	0.69	0.36	0.96	0.86	0.68	0.63	0.84
	n = 1000	0.53	0.96	0.96	0.87	0.73	0.90	0.74	0.46	0.96	0.90	0.76	0.69	0.84
	n = 2500	0.66	0.98	0.98	0.94	0.82	0.94	0.86	0.71	0.98	0.95	0.89	0.81	0.86
	n = 5000	0.75	0.98	0.98	0.96	0.87	0.96	0.92	0.83	0.98	0.96	0.94	0.86	0.88
FIML prune (alpha = 0.01)	n = 150	0.12	0.98	0.92	0.54	0.58	0.43	0.00	0.10	0.98	0.72	0.36	0.35	0.94
	n = 300	0.19	0.98	0.95	0.67	0.62	0.57	0.13	0.29	0.98	0.83	0.51	0.50	0.93
	n = 600	0.30	0.99	0.98	0.77	0.66	0.72	0.56	0.36	0.99	0.92	0.65	0.65	0.92
	n = 1000	0.38	0.99	0.98	0.83	0.74	0.78	0.70	0.46	0.99	0.94	0.73	0.73	0.91
	n = 2500	0.58	0.99	0.99	0.92	0.85	0.92	0.80	0.53	0.99	0.98	0.87	0.81	0.90
	n = 5000	0.72	0.99	0.99	0.96	0.89	0.96	0.88	0.72	0.99	0.98	0.93	0.86	0.89
WLS prune (alpha = 0.01)	n = 150	0.13	0.98	0.91	0.43	0.30	0.32	0.00	0.11	0.98	0.72	0.13	0.34	0.94
	n = 300	0.19	0.98	0.95	0.62	0.55	0.43	0.04	0.18	0.98	0.81	0.41	0.49	0.93
	n = 600	0.30	0.99	0.97	0.78	0.75	0.60	0.43	0.33	0.99	0.90	0.68	0.64	0.92
	n = 1000	0.39	0.99	0.98	0.84	0.76	0.70	0.66	0.48	0.99	0.94	0.74	0.72	0.91
	n = 2500	0.58	0.99	0.99	0.91	0.85	0.90	0.79	0.55	0.99	0.97	0.85	0.81	0.90
	n = 5000	0.73	0.99	0.99	0.96	0.90	0.96	0.87	0.72	0.99	0.98	0.93	0.86	0.89
mgm (CV; 10 folds)	n = 150	0.39	0.91	0.87	0.67	0.57	0.60	0.36	0.18	0.91	0.72	0.49	0.49	0.82
	n = 300	0.51	0.90	0.89	0.80	0.68	0.72	0.54	0.30	0.90	0.77	0.66	0.62	0.79
	n = 600	0.64	0.88	0.89	0.88	0.76	0.82	0.72	0.47	0.88	0.79	0.78	0.71	0.76
	n = 1000	0.72	0.87	0.90	0.91	0.80	0.86	0.79	0.59	0.87	0.80	0.85	0.76	0.76
	n = 2500	0.85	0.85	0.90	0.96	0.87	0.93	0.91	0.80	0.85	0.82	0.93	0.83	0.77
	n = 5000	0.92	0.83	0.90	0.98	0.91	0.96	0.95	0.90	0.83	0.82	0.96	0.87	0.77
mgm (EBIC; gamma = 0.25)	n = 150	0.20	0.99	0.98	0.67	0.67	0.64	0.01	0.22	0.99	0.92	0.54	0.50	0.93
	n = 300	0.34	0.98	0.96	0.80	0.75	0.77	0.31	0.26	0.98	0.89	0.70	0.62	0.90
	n = 600	0.49	0.97	0.97	0.87	0.82	0.83	0.68	0.44	0.97	0.92	0.81	0.71	0.87
	n = 1000	0.59	0.97	0.96	0.92	0.85	0.87	0.80	0.62	0.97	0.92	0.87	0.77	0.85
	n = 2500	0.76	0.96	0.96	0.96	0.90	0.93	0.91	0.80	0.96	0.93	0.94	0.85	0.86
	n = 5000	0.85	0.95	0.97	0.98	0.93	0.95	0.95	0.90	0.95	0.94	0.97	0.90	0.88
GGM_bootstrap (alpha = 0.01)	n = 150	0.09	0.99	0.93	0.49	0.36	0.33	0.00	0.09	0.99	0.77	0.34	0.31	0.96
	n = 300	0.18	0.99	0.95	0.68	0.52	0.51	0.07	0.21	0.99	0.83	0.56	0.49	0.93
	n = 600	0.29	0.99	0.97	0.81	0.66	0.66	0.46	0.42	0.99	0.91	0.73	0.65	0.92
	n = 1000	0.39	0.99	0.98	0.87	0.73	0.75	0.71	0.54	0.99	0.93	0.81	0.72	0.91
	n = 2500	0.58	0.99	0.99	0.94	0.83	0.86	0.90	0.78	0.99	0.97	0.91	0.81	0.89
	n = 5000	0.73	0.99	0.99	0.97	0.88	0.93	0.94	0.88	0.99	0.98	0.95	0.86	0.89
GGM_regression	n = 150	0.20	0.97	0.93	0.61	0.45	0.55	0.18	0.20	0.97	0.79	0.41	0.41	0.91
	n = 300	0.28	0.98	0.96	0.73	0.59	0.66	0.43	0.27	0.98	0.88	0.56	0.53	0.89
	n = 600	0.37	0.98	0.97	0.83	0.67	0.79	0.69	0.48	0.98	0.92	0.70	0.64	0.89
	n = 1000	0.45	0.99	0.98	0.88	0.71	0.83	0.78	0.56	0.99	0.95	0.79	0.71	0.89
	n = 2500	0.59	0.99	0.99	0.94	0.81	0.90	0.89	0.77	0.99	0.98	0.90	0.81	0.89
	n = 5000	0.70	1.00	1.00	0.97	0.86	0.94	0.94	0.87	1.00	1.00	0.94	0.87	0.90

BGGM (estimate; alpha = 0.05)	n = 150	0.16	0.96	0.86	0.52	0.38	0.35	0.01	0.13	0.96	0.64	0.34	0.34	0.91
	n = 300	0.27	0.95	0.90	0.70	0.56	0.53	0.33	0.29	0.95	0.73	0.55	0.49	0.88
	n = 600	0.40	0.95	0.93	0.82	0.67	0.69	0.54	0.33	0.95	0.82	0.71	0.62	0.87
	n = 1000	0.50	0.95	0.94	0.88	0.75	0.75	0.68	0.47	0.95	0.86	0.81	0.70	0.85
	n = 2500	0.69	0.95	0.96	0.95	0.84	0.89	0.88	0.75	0.95	0.91	0.91	0.79	0.85
	n = 5000	0.82	0.95	0.96	0.97	0.88	0.94	0.94	0.86	0.95	0.93	0.95	0.86	0.85
BGGM (explore; BF = 3)	n = 150	0.17	0.96	0.86	0.52	0.37	0.34	0.02	0.12	0.96	0.65	0.35	0.33	0.91
	n = 300	0.24	0.97	0.93	0.70	0.55	0.55	0.22	0.20	0.97	0.78	0.56	0.51	0.91
	n = 600	0.33	0.98	0.96	0.81	0.66	0.65	0.48	0.34	0.98	0.87	0.72	0.63	0.90
	n = 1000	0.41	0.99	0.98	0.88	0.73	0.75	0.70	0.50	0.99	0.94	0.81	0.71	0.90
	n = 2500	0.57	0.99	0.99	0.94	0.82	0.86	0.89	0.76	0.99	0.97	0.90	0.81	0.89
	n = 5000	0.70	0.99	0.99	0.97	0.87	0.92	0.94	0.87	0.99	0.99	0.95	0.86	0.89
EBICglasso (gamma = 0.5)	n = 150	0.69	0.69	0.78	0.74	0.59	0.76	0.51	0.26	0.69	0.62	0.58	0.66	0.61
	n = 300	0.76	0.69	0.79	0.83	0.70	0.85	0.65	0.34	0.69	0.65	0.71	0.73	0.61
	n = 600	0.83	0.67	0.80	0.90	0.78	0.90	0.76	0.48	0.67	0.67	0.82	0.78	0.61
	n = 1000	0.87	0.66	0.80	0.93	0.82	0.93	0.83	0.64	0.66	0.68	0.88	0.80	0.62
	n = 2500	0.93	0.67	0.82	0.97	0.89	0.96	0.92	0.82	0.67	0.71	0.94	0.85	0.64
	n = 5000	0.96	0.67	0.82	0.98	0.92	0.97	0.95	0.89	0.67	0.72	0.97	0.87	0.67

Table A3. A subset of simulation results for the BFI network, based on Gaussian data estimated without using a transformation. Shown are the averages of the following metrics: sensitivity (1), specificity (6), precision (7), correlation between absolute true and estimated edge-weights (12), fade maximum false edge (15), the correlation between true and estimated node strength (16), closeness (21) and betweenness (26), specificity of bridge edges (36), precision of bridge edges (37), correlation of edge weights between networks estimated from two datasets (41), the proportion of replicated edges in a second dataset (42), and the proportion of replicated zeroes in a second dataset (46).

	N	(1)	(6)	(7)	(12)	(15)	(16)	(21)	(26)	(36)	(37)	(41)	(42)	(46)
ggmModSelect (stepwise = FALSE, gamma = 0)	n = 150	0.38	0.97	0.88	0.74	0.67	0.42	0.18	0.39	0.97	0.75	0.73	0.67	0.94
	n = 300	0.48	0.96	0.89	0.82	0.76	0.49	0.40	0.45	0.97	0.79	0.83	0.76	0.94
	n = 600	0.68	0.92	0.85	0.88	0.81	0.62	0.58	0.54	0.93	0.77	0.88	0.80	0.89
	n = 1000	0.87	0.88	0.81	0.94	0.85	0.82	0.73	0.69	0.89	0.74	0.92	0.81	0.87
	n = 2500	0.97	0.89	0.84	0.98	0.91	0.95	0.90	0.84	0.90	0.79	0.97	0.88	0.90
	n = 5000	0.99	0.90	0.86	0.99	0.94	0.98	0.95	0.90	0.92	0.83	0.99	0.90	0.93
ggmModSelect (stepwise = TRUE, gamma = 0)	n = 150	0.38	0.96	0.86	0.71	0.59	0.47	0.46	0.42	0.96	0.73	0.63	0.51	0.90
	n = 300	0.49	0.97	0.90	0.80	0.69	0.57	0.57	0.51	0.97	0.82	0.73	0.60	0.90
	n = 600	0.67	0.98	0.94	0.88	0.78	0.74	0.70	0.62	0.98	0.90	0.85	0.74	0.91
	n = 1000	0.80	0.98	0.97	0.94	0.83	0.86	0.81	0.74	0.98	0.94	0.91	0.83	0.92
	n = 2500	0.96	0.99	0.99	0.98	0.90	0.96	0.92	0.86	0.99	0.98	0.97	0.95	0.97
	n = 5000	1.00	1.00	0.99	0.99	0.93	0.99	0.97	0.91	1.00	0.99	0.99	0.99	0.99
FIML prune (alpha = 0.01)	n = 150	0.23	0.98	0.87	0.63	0.63	0.30	0.03	0.26	0.98	0.72	0.54	0.38	0.93
	n = 300	0.37	0.98	0.94	0.76	0.74	0.47	0.46	0.45	0.98	0.86	0.71	0.56	0.92
	n = 600	0.56	0.99	0.97	0.87	0.84	0.66	0.70	0.61	0.99	0.94	0.83	0.71	0.92
	n = 1000	0.71	0.99	0.97	0.92	0.84	0.81	0.77	0.69	0.99	0.95	0.89	0.80	0.92
	n = 2500	0.93	0.99	0.98	0.98	0.91	0.95	0.91	0.84	0.99	0.97	0.97	0.92	0.96
	n = 5000	0.99	0.99	0.98	0.99	0.94	0.98	0.96	0.90	0.99	0.97	0.99	0.97	0.98
WLS prune (alpha = 0.01)	n = 150	0.24	0.97	0.84	0.51	0.33	0.23	0.05	0.21	0.97	0.68	0.28	0.36	0.92
	n = 300	0.37	0.98	0.93	0.68	0.59	0.35	0.32	0.29	0.98	0.85	0.53	0.55	0.92
	n = 600	0.57	0.99	0.96	0.84	0.78	0.62	0.60	0.50	0.99	0.92	0.78	0.71	0.92
	n = 1000	0.71	0.99	0.97	0.91	0.82	0.77	0.76	0.65	0.99	0.94	0.88	0.80	0.92
	n = 2500	0.93	0.99	0.98	0.98	0.91	0.95	0.91	0.83	0.99	0.98	0.97	0.93	0.96
	n = 5000	0.99	0.99	0.98	0.99	0.94	0.99	0.96	0.90	0.99	0.98	0.99	0.98	0.98
mgm (CV; 10 folds)	n = 150	0.45	0.95	0.84	0.73	0.67	0.41	0.45	0.36	0.95	0.70	0.66	0.54	0.88
	n = 300	0.63	0.93	0.85	0.84	0.77	0.60	0.59	0.52	0.93	0.74	0.79	0.64	0.87
	n = 600	0.80	0.92	0.86	0.91	0.84	0.76	0.73	0.67	0.92	0.78	0.89	0.75	0.87
	n = 1000	0.90	0.91	0.86	0.95	0.87	0.85	0.80	0.73	0.91	0.79	0.93	0.81	0.87
	n = 2500	0.98	0.91	0.86	0.98	0.92	0.95	0.91	0.84	0.91	0.80	0.97	0.86	0.90
	n = 5000	1.00	0.90	0.86	0.99	0.94	0.97	0.95	0.89	0.91	0.81	0.99	0.88	0.91
mgm (EBIC; gamma = 0.25)	n = 150	0.17	1.00	0.98	0.66	0.75	0.22	0.00	0.31	1.00	0.94	0.70	0.57	0.97
	n = 300	0.34	1.00	0.98	0.78	0.86	0.33	0.07	0.36	1.00	0.94	0.83	0.70	0.96
	n = 600	0.51	0.99	0.98	0.86	0.89	0.45	0.55	0.52	0.99	0.95	0.90	0.79	0.95
	n = 1000	0.63	0.99	0.98	0.90	0.91	0.58	0.70	0.64	0.99	0.97	0.93	0.84	0.95
	n = 2500	0.88	0.99	0.98	0.97	0.95	0.87	0.87	0.80	0.99	0.97	0.97	0.93	0.97
	n = 5000	0.96	0.99	0.98	0.98	0.96	0.94	0.93	0.87	0.99	0.97	0.99	0.96	0.98
GGM_bootstrap (alpha = 0.01)	n = 150	0.16	0.99	0.90	0.56	0.51	0.32	0.00	0.20	0.99	0.75	0.45	0.35	0.95
	n = 300	0.34	0.99	0.94	0.74	0.67	0.43	0.31	0.40	0.99	0.87	0.66	0.55	0.93
	n = 600	0.55	0.99	0.96	0.86	0.77	0.62	0.64	0.58	0.99	0.92	0.81	0.71	0.92
	n = 1000	0.70	0.99	0.97	0.92	0.82	0.77	0.77	0.69	0.99	0.95	0.89	0.80	0.92
	n = 2500	0.93	0.99	0.98	0.98	0.89	0.94	0.90	0.82	0.99	0.97	0.97	0.93	0.96
	n = 5000	0.99	0.99	0.98	0.99	0.92	0.98	0.95	0.88	0.99	0.97	0.99	0.97	0.98
GGM_regression	n = 150	0.29	0.98	0.91	0.68	0.58	0.38	0.24	0.34	0.98	0.79	0.58	0.48	0.93
	n = 300	0.40	0.99	0.96	0.78	0.70	0.51	0.49	0.48	0.99	0.91	0.72	0.61	0.93
	n = 600	0.56	0.99	0.98	0.87	0.78	0.67	0.69	0.63	0.99	0.96	0.83	0.73	0.92
	n = 1000	0.69	1.00	0.99	0.92	0.83	0.80	0.80	0.72	1.00	0.98	0.89	0.80	0.93
	n = 2500	0.91	1.00	1.00	0.98	0.88	0.95	0.92	0.85	1.00	0.99	0.97	0.93	0.96
	n = 5000	0.99	1.00	1.00	0.99	0.92	0.99	0.96	0.90	1.00	1.00	0.99	0.99	0.99

BGGM (estimate; alpha = 0.05)	n = 150	0.28	0.96	0.82	0.60	0.52	0.31	0.11	0.22	0.96	0.64	0.47	0.39	0.91
	n = 300	0.48	0.96	0.87	0.76	0.68	0.51	0.47	0.45	0.96	0.76	0.68	0.57	0.89
	n = 600	0.69	0.96	0.90	0.88	0.77	0.70	0.67	0.62	0.96	0.84	0.83	0.72	0.89
	n = 1000	0.82	0.95	0.91	0.93	0.83	0.83	0.79	0.71	0.95	0.86	0.90	0.80	0.90
	n = 2500	0.97	0.95	0.92	0.98	0.89	0.94	0.90	0.83	0.95	0.88	0.97	0.90	0.93
	n = 5000	1.00	0.95	0.92	0.99	0.92	0.97	0.95	0.88	0.95	0.88	0.98	0.92	0.95
BGGM (explore; BF = 3)	n = 150	0.28	0.96	0.81	0.59	0.51	0.27	0.13	0.24	0.96	0.64	0.47	0.38	0.91
	n = 300	0.43	0.97	0.90	0.76	0.67	0.49	0.44	0.42	0.97	0.81	0.67	0.56	0.90
	n = 600	0.60	0.98	0.94	0.87	0.77	0.68	0.65	0.60	0.98	0.90	0.82	0.71	0.91
	n = 1000	0.73	0.99	0.97	0.92	0.82	0.80	0.78	0.71	0.99	0.94	0.89	0.80	0.92
	n = 2500	0.92	0.99	0.98	0.98	0.89	0.94	0.90	0.83	0.99	0.97	0.97	0.92	0.96
	n = 5000	0.99	0.99	0.99	0.99	0.92	0.98	0.95	0.89	0.99	0.98	0.99	0.98	0.99
EBICglasso (gamma = 0.5)	n = 150	0.25	0.98	0.93	0.65	0.81	0.25	0.06	0.30	0.99	0.83	0.69	0.59	0.95
	n = 300	0.64	0.90	0.80	0.84	0.80	0.49	0.54	0.48	0.91	0.68	0.88	0.73	0.88
	n = 600	0.79	0.87	0.79	0.90	0.84	0.61	0.65	0.59	0.89	0.70	0.93	0.78	0.87
	n = 1000	0.88	0.87	0.80	0.93	0.86	0.70	0.71	0.63	0.88	0.73	0.95	0.81	0.86
	n = 2500	0.99	0.82	0.77	0.97	0.92	0.91	0.86	0.79	0.83	0.69	0.98	0.83	0.83
	n = 5000	1.00	0.82	0.77	0.99	0.94	0.95	0.92	0.87	0.83	0.70	0.99	0.84	0.84