**Supplementary Information**

Network Accuracy Across Levels of Analysis Using Stochastic Block Models

Alexander P. Christensen & Jeongwon Choi

## SI 1. Edge Weight Generation

To generate edge weights that mirror real-world data, we analyzed the 293 networks from Huth, Haslbeck, Keetelaar, van Holst, & Marsman (2025). We narrowed these networks down to those that most closely represented our simulated conditions: total number of nodes between 8 and 40, and edge densities between 0.20 and 0.80, yielding 191 networks. For the absolute non-zero edge weight distributions of these networks, we fit several candidate distributions (beta, gamma, log-normal, Weibull) that are known for their flexibility (beta, gamma, and Weibull) and appeared to be similar to empirical distributions (log-normal; Wysocki & Rhemtulla, 2021) using the `fitdistr` function in the {fitdistrplus} package (version 1.2.4; Delignette-Muller & Dutang, 2015) in R. The goal was to identify a distribution to generate non-zero edge weights to assign to the edges in the simulated PPM adjacency matrices.

Beta and Weibull distributions most frequently provided the best fit across the 191 networks (33.0% and 27.7%, respectively). Although the beta distribution fit best more frequently overall, a closer examination of the log-likelihood ($L$) differences revealed an asymmetric pattern. In the 78 networks where beta achieved higher log-likelihood than Weibull, the improvement was consistently negligible ($\Delta L < 2$). In contrast, among the 113 networks where Weibull achieved higher log-likelihood than beta, the improvement was substantial ($\Delta L > 2$) for 34 cases. This same pattern held when comparing Weibull to gamma and log-normal distributions: when gamma or log-normal fit better than Weibull, substantial improvements ($\Delta L > 2$) were rare (0 and 10 cases, respectively), but when Weibull fit better than gamma or log-normal, substantial improvements were more common (4 and 124 cases where $\Delta L > 2$, respectively). Based on these patterns, we selected the Weibull distribution to generate edge weights.

The probability density function of a Weibull random variable takes the form:

$$\frac{\alpha}{\eta}\left(\frac{x}{\eta}\right)^{\alpha-1} e^{-(x/\eta)^{\alpha}},$$

where $x \geq 0$, $\alpha$ defines the shape, and $\eta$ defines the scale. The shape parameter has several notable cases: $\alpha = 1$ reflects an exponential distribution, $\alpha = 2$ reflects a Rayleigh distribution, and $\alpha \approx 3.5$ reflects a roughly normal distribution. The scale parameter affects the location and spread of the distribution. For each network, we estimated the shape and scale parameters using Maximum Likelihood Estimation from the {fitdistrplus} package.

Because these distribution parameters likely vary with network features (network density and mean absolute non-zero partial correlation) and sample characteristics (sample size and number of variables), we computed Pearson's correlations to determine how they relate to each parameter (Table 1). One key (and expected) finding was the near identical relationship between scale and mean absolute non-zero partial correlations ($r = 0.985$). Indeed, the mean of the Weibull distribution is defined as:

$$\mu = \eta\ \Gamma(1 + \frac{1}{\alpha}),$$

$\Gamma(z) = \int_0^{\infty} t^{z-1} e^{-t} \mathrm{d}t$. When $\alpha$ is around 1, which described the majority of the empirical distributions ($M = 1.093, SD = 0.188, \text{range} = 0.699 - 1.689$), then the term $\Gamma(1 + \frac{1}{\alpha})$ is centered around 1 (for the full range of empirical $\alpha$: $0.893 - 1.267$). Since this term is centered on 1, $\bar{x} \approx \eta$.

Another key observation was the nonlinear relationships between the estimated Weibull parameters and both sample size and number of variables. Log transforming sample size improved its associations: $r = -0.368$ for shape and $r = -0.452$ for scale (both $p$'s $< 0.001$). The number of variables was transformed using $\frac{1}{\log(\text{variables})}$, a relationship identified by Christensen, Golino, Abad, and Garrido (2025) for mean partial correlations, which improved its association with scale ($r = 0.845$) but not for shape ($r = 0.544$; both $p$'s $< 0.001$).

**Table 1**

*Weibull Parameters Correlated with Network and Sample Descriptive Statistics (N = 191)*

| | **Network Features** | | **Sample Characteristics** | |
|---|---|---|---|---|
| **Parameter** | Density | Partial Correlation | Sample Size | Number of Variables |
| Shape ($\alpha$) | 0.197** | 0.514*** | -0.280*** | -0.564*** |
| Scale ($\eta$) | 0.251*** | 0.985*** | -0.322*** | -0.785*** |

*Note.* * p < 0.05, ** p < 0.01, *** p < 0.001. "Partial Correlation" reflects the mean absolute non-zero partial correlation.

To ensure that simulated edge weights maintained these relationships observed in the empirical networks, we developed a parameter generating procedure to create plausible Weibull parameters based on our simulated network features and sample characteristics. Because the mean partial correlations are not known prior to generation, the scale parameter served as a proxy and therefore was used as the outcome in a linear regression model:

$$\hat{\text{scale}} = -0.061 + 0.021(\text{shape}) - 0.004\log(\text{sample size}) + 0.610\left(\frac{1}{\log(\text{variables})}\right) - 0.090(\text{density}).$$

Before finalizing the model, we identified and removed influential observations using Cook's distance ($D > \frac{4}{N}$), which resulted in the removal of 19 networks, leaving 172 networks for model estimation. The final model achieved strong predictive performance ($R^2 = 0.927, RMSE = 0.008$), suggesting the model was viable. The unstandardized coefficients presented in the equation above facilitate future generation of edge weights using the Weibull distribution. To assess relative predictor importance, we examined standardized coefficients, which revealed that all predictors were significant ($p$'s $< 0.001$) with the strongest being $\frac{1}{\log(\text{variables})}$ ($\beta = 1.041$) followed by density ($\beta = -0.394$), shape ($\beta = 0.129$), and log(sample size) ($\beta = -0.125$).

For each simulated adjacency matrix, edge weights were generated using the following procedure. First, we randomly sampled an empirical shape parameter value from the 172 networks and added uniform noise using the standard deviation of shape, $\text{Uniform}(\text{shape} - 0.188, \text{shape} + 0.188)$, to introduce variation while maintaining plausible

values. Second, we used the linear model to obtain a 99% prediction interval for the scale parameter given this generated shape parameter and the simulated condition's network features and sample characteristics. Third, we used this interval to randomly sample a scale parameter, capturing the full range of plausible variation. The aim of this approach was to preserve the empirical relationship between mean absolute non-zero edge weights and both network features and sample characteristics while incorporating variability. Finally, we generated edge weights from a Weibull distribution using the sampled shape and scale parameters. Because empirical networks exhibited a maximum edge weight of 0.845, we capped the maximum edge weight to 0.85 and regenerated edge weights if any generated values exceeded this maximum. All generated edge weights were positive and remained positive for ease of creating positive definite matrices.

This procedure ensured that our simulated edge weights maintained distributional properties consistent with real-world psychological networks (Figure 1). The simulated proportion of edge weights broken down into bins maintained parity (Table 2) with both Huth et al. (2025; 172 networks) and Wysocki and Rhemtulla (2021; p. 293).

**Table 2**
*Binned Proportions of Edge Weights*

| Bins | Simulated | Huth et al. (2025) | Wysocki & Rhemtulla (2021) |
|---|---|---|---|
| 0.00–0.15 | 77.0% | 76.7% | 74.1% |
| 0.15–0.25 | 15.0% | 14.3% | 13.5% |
| 0.25–0.50 | 7.6% | 8.3% | 9.7% |
| 0.50–1.00 | 0.4% | 0.6% | 2.7% |

To provide a more granular demonstration that our simulated edge weight distributions generally mirrored the empirical data, we computed Kullback-Leibler Divergence (KLD) between each shape and scale parameters estimated the empirical networks and each empirical network's estimated shape and scale parameters. Following Bauckhage (2013), the KLD between two Weibull distributions is defined as:

$$D_{KL}(F_1 \| F_2) = \int_0^\infty f_1(x \mid \alpha_1, \eta_1) \log \frac{f_1(x \mid \alpha_1, \eta_1)}{f_2(x \mid \alpha_2, \eta_2)} \, dx =$$

$$\log \frac{\alpha_1}{\eta_1^{\alpha_1}} - \log \frac{\alpha_2}{\eta_2^{\alpha_2}} + (\alpha_1 - \alpha_2) \left[ \log \eta_1 - \frac{\gamma}{\alpha_1} \right] + \left( \frac{\eta_1}{\eta_2} \right)^{\alpha_2} \Gamma\left( \frac{\alpha_2}{\alpha_1} + 1 \right) - 1,$$

where $\gamma$ is the Euler-Mascheroni constant (0.57721) and subscripts '1' and '2' define the target distribution (empirical parameters) and comparison distribution (simulated parameters), respectively. KLD is a relative metric based on fit to the target distribution with Smaller values to zero are better, more similar distributions. The minimum KLD for each simulated network across the empirical parameters was obtained and demonstrated very close similarities: $Mdn = 0.0007, IQR = 0.0003$–$0.002$, range $= 7 \times 10^{-8} - 0.1877$.

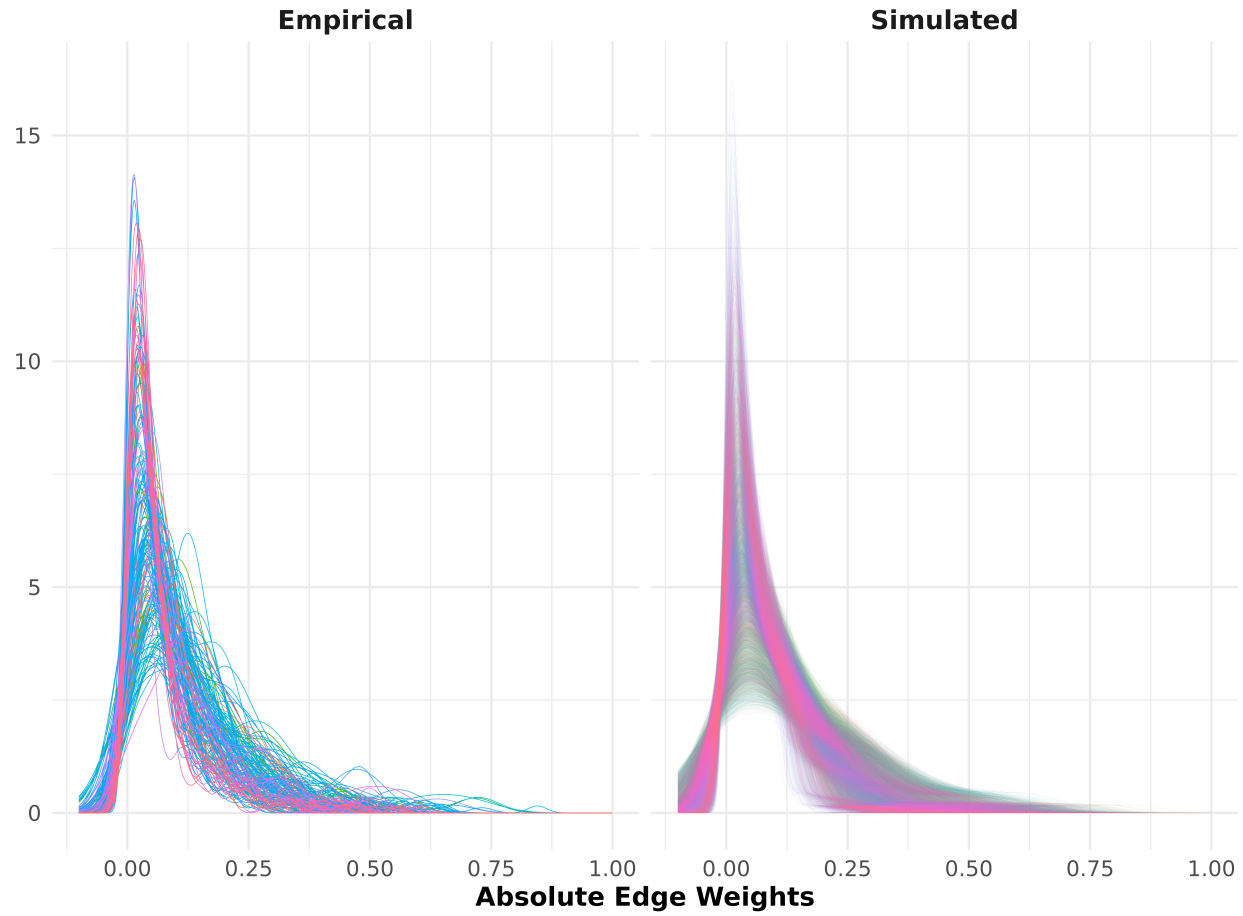



**Figure 1**

*Absolute non-zero edge weight distributions for the subsampled 172 networks from Huth et al. (2025; left) and the 14,400 simulated networks (right)*

The correlations between network features and sample characteristics were largely comparable between the empirical and simulated networks (all $p$'s $< 0.001$. The total number of variables was negatively correlated with density in both empirical ($r = -0.657$) and simulated ($r = -0.484$) networks as well as mean absolute non-zero edge weights: $r = -0.799$ and $r = -0.873$, respectively. Edge density was positively correlated with mean absolute non-zero edge weights: $r = 0.285$ and $r = 0.149$, respectively.

Once edge weights were generated, the adjacency matrices were populated by assigning edge weights to within- and between-community edges separately. First, all edge weights were sorted in descending order. The top $N$ edge weights (where $N$ equals the total number of within-community edges across all communities) were randomly shuffled and assigned to the within-community edges. The remaining edge weights were randomly shuffled and assigned to the between-community edges. This procedure ensured that within-community edges were proportionately stronger than between-community edges.

## SI 2. Network Estimation Methods

For all network estimation methods, Pearson's, polychoric, and tetrachoric correlations were computed for continuous, polytomous, and dichotomous data (respectively), producing a $p \times p$ correlation matrix, $\hat{\mathbf{R}}$. The following briefly describes each network estimation method and the defaults that are typically used in their software packages.

### *Full Partial Correlation Matrix*

As a baseline, the full partial correlation matrix was used. To obtain the partial correlation matrix from the zero-order correlation matrix, $\hat{\mathbf{R}}$, the following was performed,

$$\hat{\mathbf{K}} = \hat{\mathbf{R}}^{-1},$$

The precision matrix is then converted to the partial correlation matrix ($\hat{\mathbf{\Omega}}$) by:

$$\hat{\mathbf{\Omega}}_{ij} = -\frac{\hat{\mathbf{K}}_{ij}}{\sqrt{\hat{\mathbf{K}}_{ii}\hat{\mathbf{K}}_{jj}}}.$$

In the main text, the full partial correlation matrix is referred to as "Full".

### *GLASSO*

Two versions of the GLASSO with different parameter values were employed. The GLASSO algorithm applies block-wise coordinate descent to maximize the penalized log-likelihood (Friedman, Hastie, & Tibshirani, 2008),

$$\log\det(\hat{\mathbf{K}}) - \text{tr}(\hat{\mathbf{R}}\hat{\mathbf{K}}) - \lambda \sum_{<i,j>} |\hat{\mathbf{K}}_{ij}|,$$

where the last term, $\sum_{<i,j>} |\hat{\mathbf{K}}_{ij}|$, represents the $\ell_1$ norm. In this process, estimates of the precision matrix, $\hat{\mathbf{K}}$, are often (but not always) made smaller and estimates smaller than the lambda ($\lambda$) are set to zero (known as "soft-thresholding"). $\lambda$ controls the level of sparsity and must be set. The standard in the network psychometric literature has been to estimate many networks using 100 $\lambda$ values and select the network that minimizes the extended Bayesian information criterion (Chen & Chen, 2008), which also has a parameter gamma ($\gamma$).

The Gaussian log-likelihood and EBIC are defined as,

$$L = \frac{N}{2} \log \det(\hat{\mathbf{K}}) - \text{tr}(\hat{\mathbf{R}}\hat{\mathbf{K}})$$

and

$$-2L + E \log(N) + 4\gamma E \log(p),$$

where $N$ is the sample size and tr is the trace or sum of the diagonal of a matrix. The standard in the literature has been to set $\gamma = 0.50$ and search over a logarithmically spaced range of $\lambda$ values. The minimum and maximum of this range is defined by:

$$\lambda_{\text{max}} = \max |\hat{\mathbf{R}} - \mathbf{I}|$$

and

$$\lambda_{\text{min}} = 0.01\lambda_{\text{max}}.$$

A sequence of 100 $\lambda$ values are obtained by generated in an evenly spaced increment between $\lambda_{min}$ and $\lambda_{\text{max}}$. The exponent of these values is then computed. Afterwards, these $\lambda$ values are used in the GLASSO algorithm to estimate a set of 100 networks. The network that minimizes the EBIC is selected (Epskamp & Fried, 2018). This approach describes the EBICglasso algorithm as implemented by the {qgraph} package (version 1.9.8; Epskamp, Cramer, Waldorp, Schmittmann, & Borsboom, 2012) in R. In the main text, this variant is referred to as "EBICglasso {qgraph}".

The other variant of the EBICglasso algorithm is implemented in the {EGAnet} package (version 2.0.7; Golino & Christensen, 2025) in R. This variant applies two changes to the above description. First, the 0.01 value used to obtain the $\lambda_{min}$ value is set to 0.10 instead. This adjustment increases $\lambda_{min}$ which can lead to the selection of sparser networks than the EBICglasso {qgraph} default. Second, the $\gamma$ parameter of the EBIC is adjusted

based on whether all nodes in the network are connected. If any node is disconnected from the rest of the network (i.e., all edges equal zero), then the $\gamma$ parameter is decreased by 0.25 and the algorithm restarts. If there is an unconnected node at $\gamma = 0.25$, then $\gamma$ is further dropped to zero, setting the EBIC equal to the Bayesian information criterion (BIC). Regardless of connectivity at $\gamma = 0.00$, the resulting network is retained. In the main text, this variant is referred to as "EBICglasso {EGAnet}".

***Non-convex GLASSO***

Non-convex penalties for the GLASSO algorithm were developed to address some of the limitations of the $\ell_1$ regularization such as increased false positive edges (Williams, Rhemtulla, Wysocki, & Rast, 2019) and no guarantee to converge to the *true* model as $N \to \infty$ (Williams & Rast, 2020). Several penalties were introduced by Fan and Li (2001) and implemented in the easy-to-use {GGMncv} package (version 2.1.1; Williams, 2020) in R.

{GGMncv} provides several non-convex options. The default options were tested in this study. Specifically, the non-convex penalty of Atan, which replaces the $\ell_1$ norm in the penalized likelihood above with the arctan penalty defined as,

$$\lambda \left( \nu + \frac{2}{\pi} \right) \arctan \left( \frac{|\hat{\mathbf{K}}|}{\nu} \right),$$

where $\nu$ represents a parameter that adjusts the penalty between the $\ell_0$ norm ($\nu \to 0$) and $\ell_1$ norm ($\nu \to \infty$).[1] By default, and recommended by Wang and Zhu (2016), $\nu = 0.01$. $\nu$ is selected similarly to the EBICglasso by searching over many values (50) and selecting the network that minimizes the EBIC ($\gamma = 0.50$). To ensure robust results in high dimensional settings (e.g., 4 factors with 8 variables per factor), the local linear approximation (LLA; Fan, Feng, & Wu, 2009) was used (setting `LLA = TRUE`). Because this model is the only non-convex regularization model, it is referred to as "GGMncv" in the main text.

[1]Traditionally, gamma ($\gamma$) is used for the $\nu$ parameter. To avoid confusion with the EBIC $\gamma$, we use $\nu$ instead.

### *Non-regularized GLASSO Selection*

The non-regularized GLASSO approach, implemented as `ggmModSelect` in the {qgraph} package, follows the same estimation steps as the EBICglasso {qgraph} algorithm with the exception that each network is re-estimated without regularization using maximum likelihood estimation Isvoranu & Epskamp (2023). By default, the $\gamma$ parameter of EBIC is set to zero (i.e., BIC). There is the potential for an additional procedure to add and remove edges in a stepwise manner, however, this procedure was not used in this study (the argument `"stepwise"` was set to `FALSE`). In the main text, this model is referred to as "GGMmod".

### *Non-regularized Non-convex Penalty*

Another non-regularized method was implemented using a non-convex regularization estimator that applies the seamless $\ell_0$-penalty or SELO (Williams, 2020). Like the Atan penalty, the $\ell_0$-penalty replaces the $\ell_1$ norm in the penalized log-likelihood using ($\nu = 0.01$):

$$\frac{\nu}{\log(2)} \log\left(\frac{|\hat{\mathbf{K}}|}{|\hat{\mathbf{K}}| + \nu} + 1\right).$$

Although this approach uses regularization, the SELO penalty approximates non-regularized best subset model selection. By default in the {GGMnonreg} package (version 1.0.0; Williams, 2019), this method is applied using the {GGMncv} package with `unreg = TRUE`, which estimates the network using regularization and then refits the unregularized model following the same approach as GGMmod (Foygel & Drton, 2010). Similar to Atan, the LLA was used (`LLA = TRUE`). Because this model is estimated using the defaults set in the {GGMnonreg} package, it is referred to as "GGMnonreg" in the main text.

## SI 3. Benchmarking $s_F$ and Jensen-Shannon Similarity

Four conditions were simulated to establish interpretation guidelines for $s_F$ and Jensen-Shannon Similarity (JSS). The first two conditions aim to establish baseline expectations through maximal disruption of network structure (Condition 1) and edge weights (Condition 2). The latter two conditions provide incremental disruptions to edge weights (Condition 3) and network structure (Condition 4). Together, these conditions provide insight into how different magnitudes of change in structure versus edge weights affect each similarity metric.

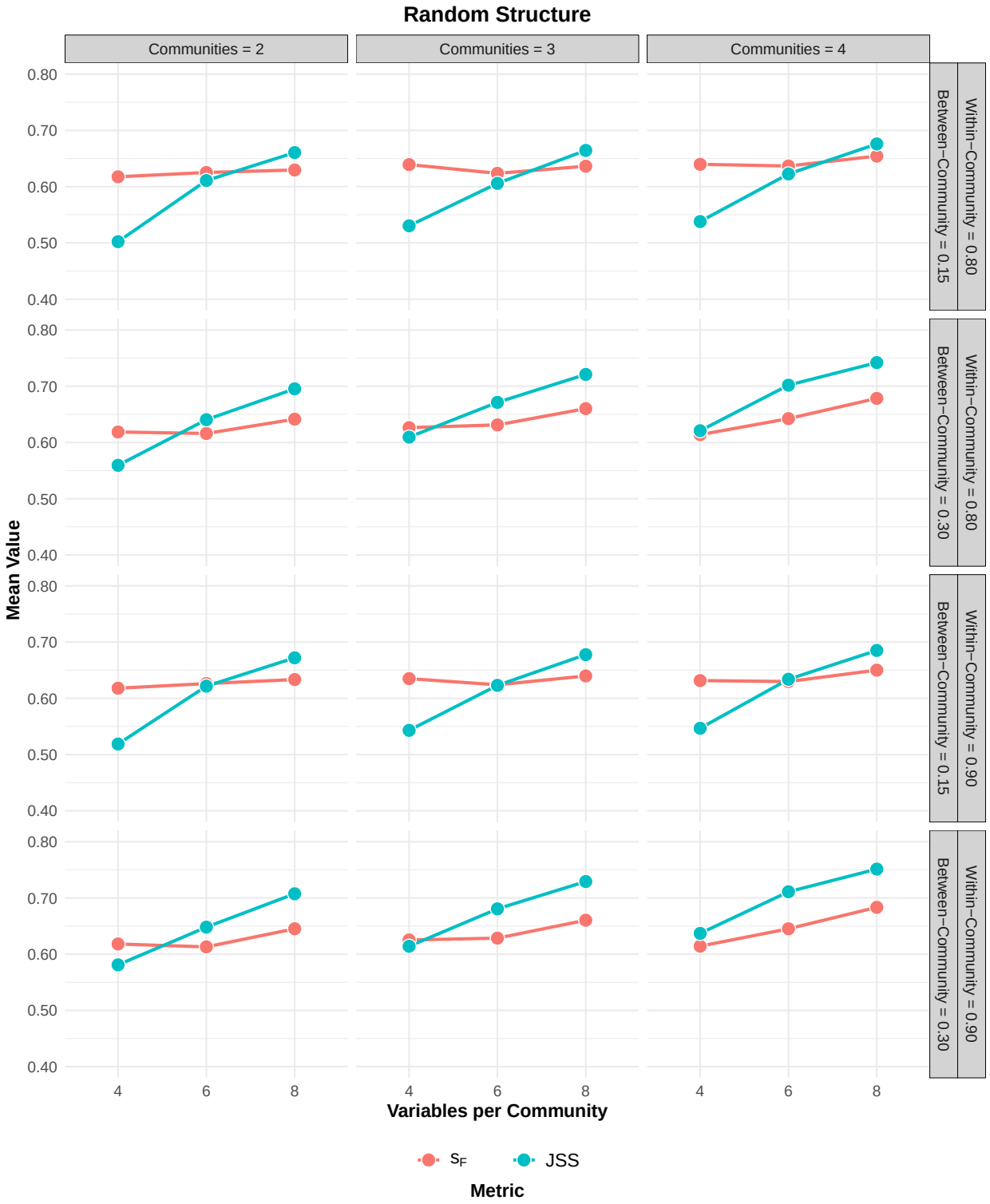
Each of the 14,400 population networks was used as the base network for manipulations. Comparing the effects of the manipulations over all population networks across all conditions allowed for any condition-specific effects to emerge (Ulitzsch, Khanna, Rhemtulla, & Domingue, 2023). This within-network design provides controlled comparisons, as each population network serves as its own ground-truth baseline. For all conditions (including each increment), 1000 iterations were performed. The mean values across these iterations serve as a more robust estimate of the change in the metrics such that changes are not driven purely by random chance.

### *Condition 1: Random Structure, Same Edge Weights*

The first condition evaluated the extent to which completely randomizing network structure affects the similarity measures while holding edge weights constant. Edge weights remained the same values but were randomly redistributed to randomly selected pairwise connections (maintaining the same number of edges as the original network). This condition ensured that the edge weights were consistent with the original network but the network structure bore no relationship to the original. This condition provides a lower bound on the values for how much the metrics can change due to structural differences alone (Figure 2).

**Figure 2**

*$s_F$ and Jensen-Shannon Similariy (JSS) when the structure of the ground truth networks have been randomized*

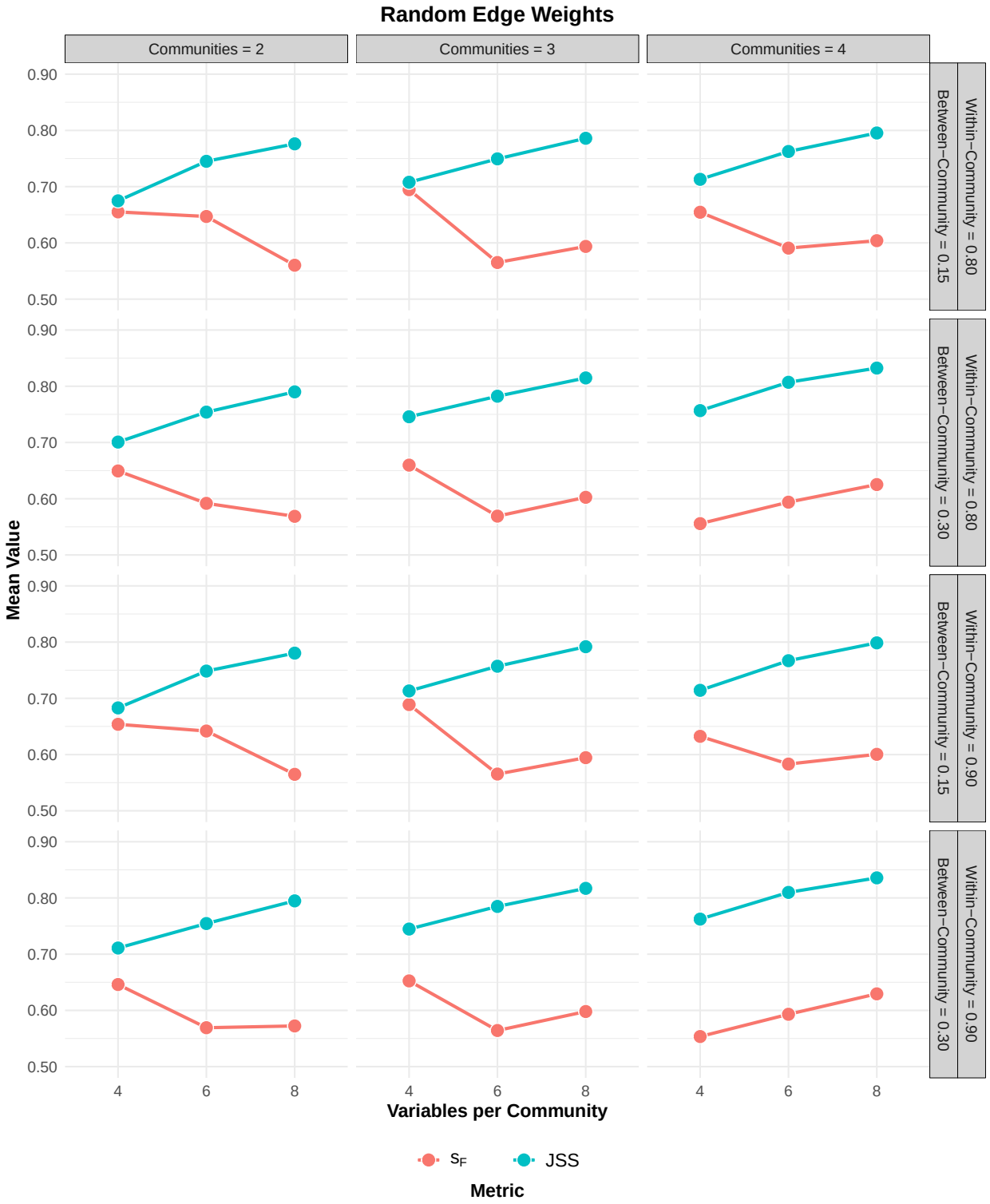


### *Condition 2: Same Structure, Random Edge Weights*

The second condition evaluated the extent to which completely randomizing edge weights affects the similarity measures while holding network structure constant. The network structure was maintained but edge weight values were replaced with random values drawn from a uniform distribution ranging from the minimum to the maximum edge weight in the original network. This condition provides a lower bound on the values for how much the metrics can change due to edge weight differences alone (Figure 3).

**Figure 3**

*$s_F$ and Jensen-Shannon Similariy (JSS) when the edge weight values of the ground truth networks have been randomized*



### *Condition 3: Same Structure, But Incrementally Noisy Edges*

The third condition examined how incremental changes to edge weights affect the similarity measures. The network structure was retained and incremental amounts of random noise were added to the original edge weights. The random noise added to the edge weights were drawn from a uniform distribution centered around the values 0.00, 0.025, 0.050, 0.075, and 0.100 with ± 0.025 for the minimum and maximum of the range,

$$\pm(c + U),$$

where the ± sign indicates that the noise value is randomly added to or subtracted from each edge weight,

$$c \in \{0.00, 0.025, 0.050, 0.075, 0.100\}$$

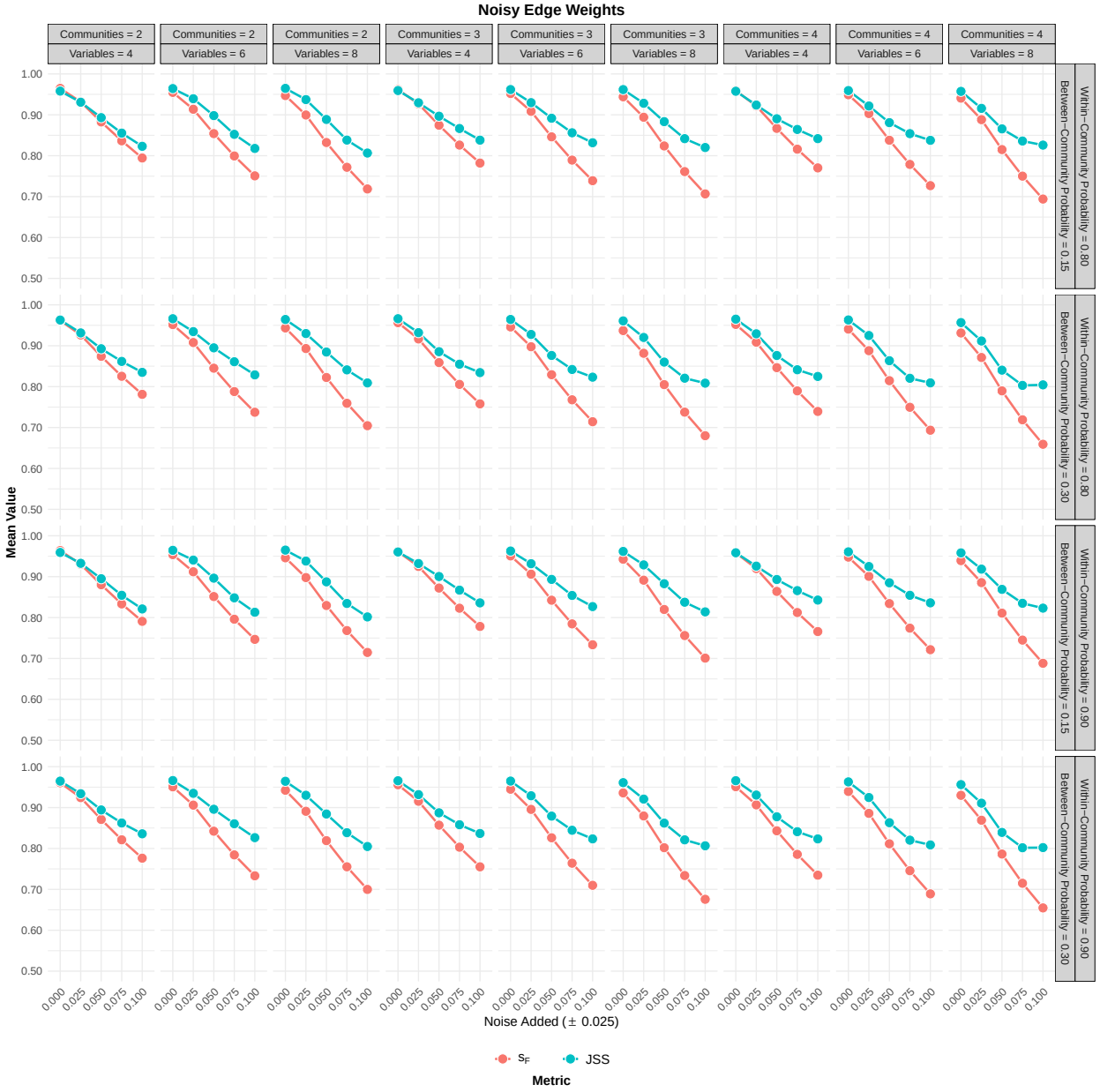
and

$$U \sim \text{Uniform}(-0.025, 0.025).$$

This procedure maintains the same structure but incrementally increases deviation in the magnitude of the edge weights between the networks. A metric more sensitive to changes in edge weights should show steeper declines across increments than one that is less sensitive (Figure 4). This condition is specifically targeted toward the $s_F$ metric and serves as the primary condition to interpret how $s_F$ changes with respect to different magnitudes of edge weight perturbation.

**Figure 4**

*$s_F$ and Jensen-Shannon Similariy (JSS) when the edge weight values of the ground truth networks have had noise added to them*



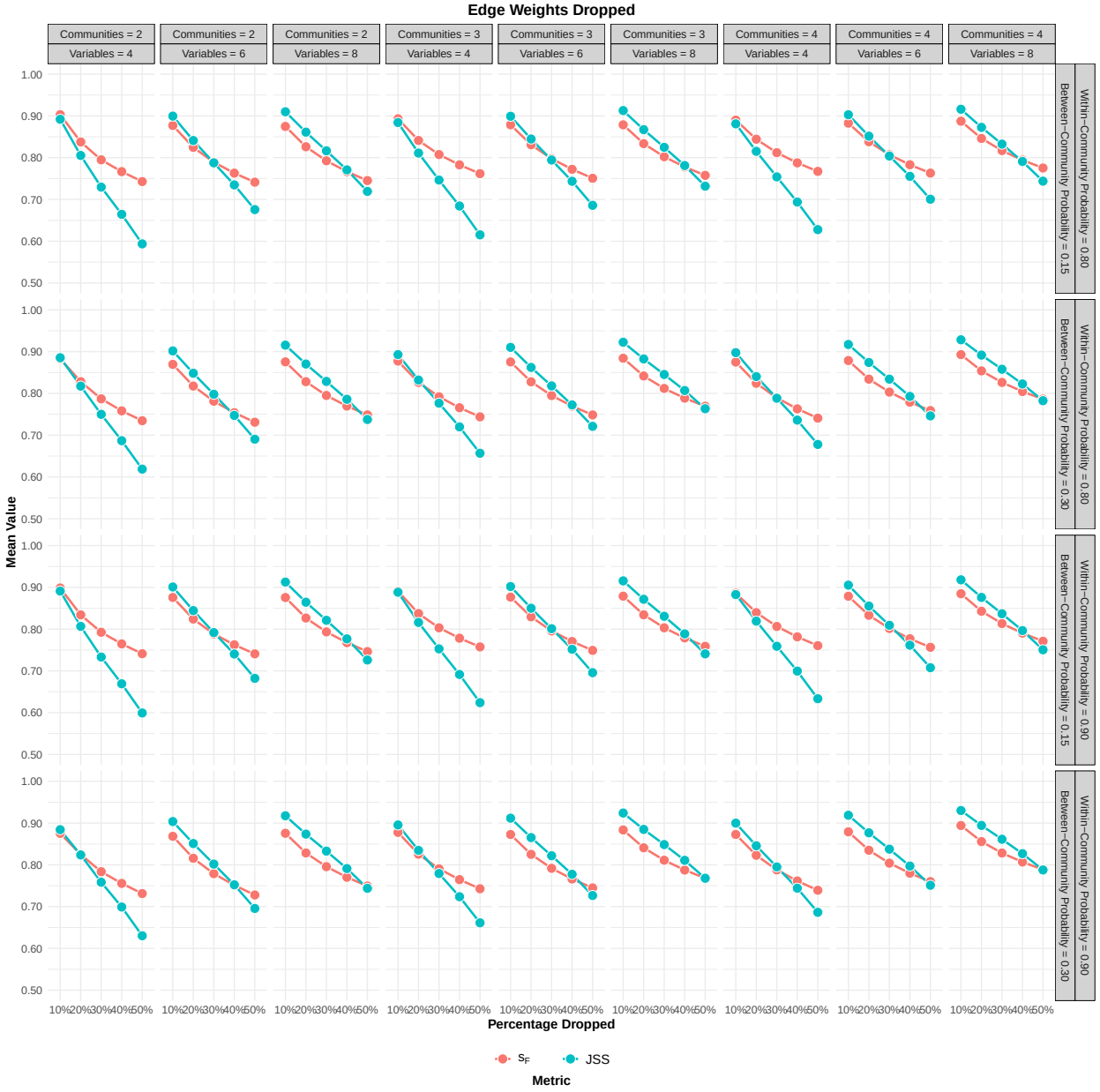

### *Condition 4: Same Weights, But Incrementally Dropped Edges*

The fourth condition examined how incremental degradation of the network structure affects the similarity measures. Starting with the original network, edges were incrementally dropped at rates of 0.10, 0.20, 0.30, 0.40, and 0.50, progressively degrading the network

structure while maintaining the weights of retained edges. A metric more sensitive to structural disruption should show steeper declines across increments than one that is less sensitive (Figure 5). Although $s_F$ is expected to be affected (dropped edge weights contribute to edge weight differences), the retained edges will have a difference of zero, which should attenuate the overall effect of edge weight differences. This condition is specifically targeted toward the JSS metric and serves as the primary condition to interpret how JSS changes with respect to different magnitudes of structural degradation.

**Figure 5**

*$s_F$ and Jensen-Shannon Similariy (JSS) when the edges of the ground truth networks have been dropped*



## References

Bauckhage, C. (2013). Computing the Kullback-Leibler Divergence between two Weibull distributions. *arXiv*. https://doi.org/10.48550/arXiv.1310.3713

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

Christensen, A. P., Golino, H., Abad, F. J., & Garrido, L. E. (2025). Revised network loadings. *Behavior Research Methods*, *57*(4), 1–18. https://doi.org/10.3758/s13428-025-02640-3

Delignette-Muller, M. L., & Dutang, C. (2015). fitdistrplus: An R package for fitting distributions. *Journal of Statistical Software*, *64*(4), 1–34. https://doi.org/10.18637/jss.v064.i04

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*, 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*, 617–634. https://doi.org/10.1037/met0000167

Fan, J., Feng, Y., & Wu, Y. (2009). Network exploration via the adaptive LASSO and SCAD penalties. *The Annals of Applied Statistics*, *3*(2), 521–541. https://doi.org/10.1214/08-AOAS215

Fan, J., & Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, *96*(456), 1348–1360. https://doi.org/10.1198/016214501753382273

Foygel, R., & Drton, M. (2010). Extended Bayesian information criteria for Gaussian graphical models. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, & A. Culotta (Eds.), *Advances in neural information processing systems* (pp. 604–612). Retrieved from https://papers.nips.cc/paper/4087-extended-bayesian-information-criteria-for-gaussian-graphical-models

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

Golino, H., & Christensen, A. P. (2025). *EGAnet: Exploratory Graph Analysis– a framework for estimating the number of dimensions in multivariate data using network psychometrics*. Retrieved from https://r-ega.net

Huth, K. B. S., Haslbeck, J. M. B., Keetelaar, S., van Holst, R. J., & Marsman, M. (2025). Statistical evidence in psychological networks. *Nature Human Behaviour*, 1–14. https://doi.org/10.1038/s41562-025-02314-2

Isvoranu, A.-M., & Epskamp, S. (2023). Which estimation method to choose in network psychometrics? Deriving guidelines for applied researchers. *Psychological Methods*, 925–946. https://doi.org/10.1037/met0000439

Ulitzsch, E., Khanna, S., Rhemtulla, M., & Domingue, B. W. (2023). A graph theory based similarity metric enables comparison of subpopulation psychometric networks. *Psychological Methods*. https://doi.org/10.1037/met0000625

Wang, Y., & Zhu, L. (2016). Variable selection and parameter estimation with the Atan regularization method. *Journal of Probability and Statistics*, *2016*(1), 6495417. https://doi.org/10.1155/2016/6495417

Williams, D. R. (2019). *GGMnonreg: Estimate non-regularized Gaussian graphical models*. Retrieved from https://github.com/donaldRwilliams/GGMnonreg

Williams, D. R. (2020). Beyond lasso: A survey of nonconvex regularization in Gaussian graphical models. *PsyArXiv*. https://doi.org/10.31234/osf.io/ad57p

Williams, D. R., & Rast, P. (2020). Back to the basics: Rethinking partial correlation network methodology. *British Journal of Mathematical and Statistical Psychology*, *73*(2), 187–212. https://doi.org/10.1111/bmsp.12173

Williams, D. R., Rhemtulla, M., Wysocki, A. C., & Rast, P. (2019). On nonregularized estimation of psychological networks. *Multivariate Behavioral Research*, *54*, 719–750.