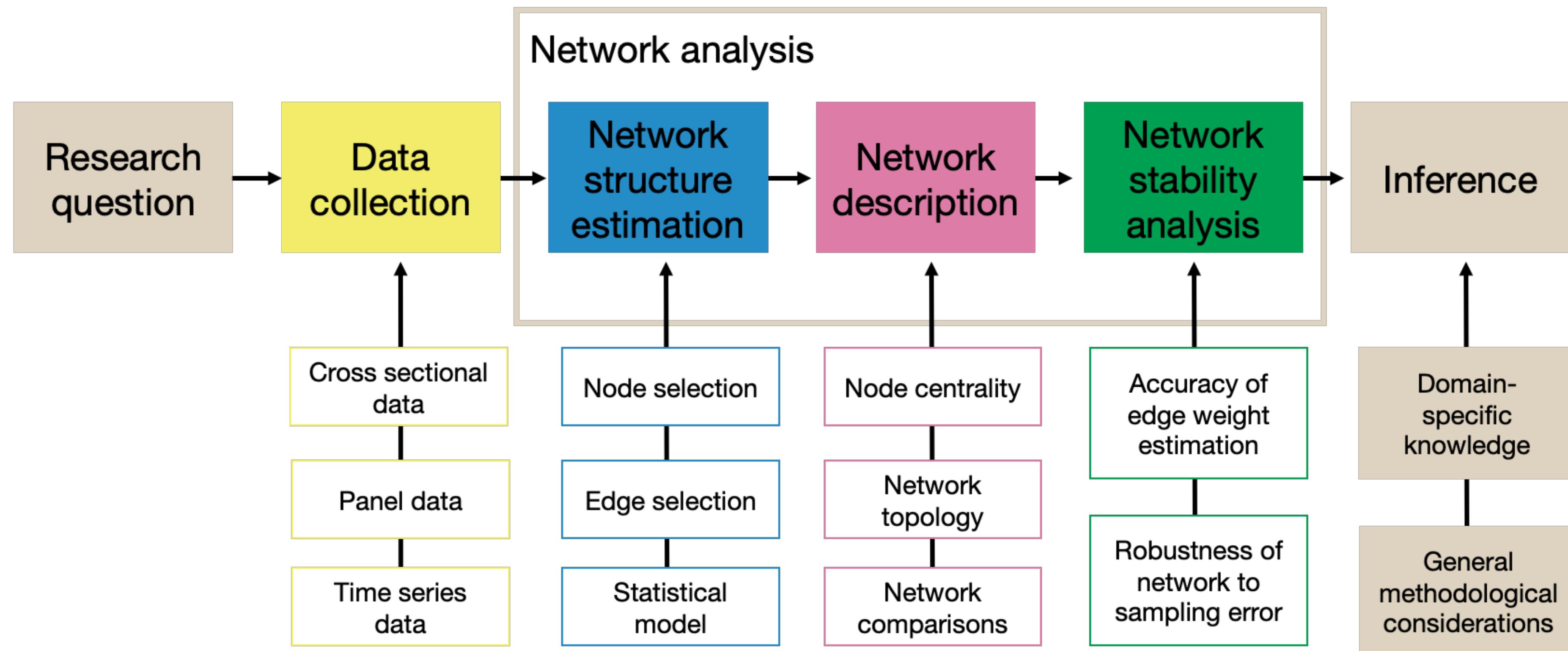


Estimation & model selection Stability & replicability Comparison & inferences

Tessa Blanken
Network Analysis Course
November 17

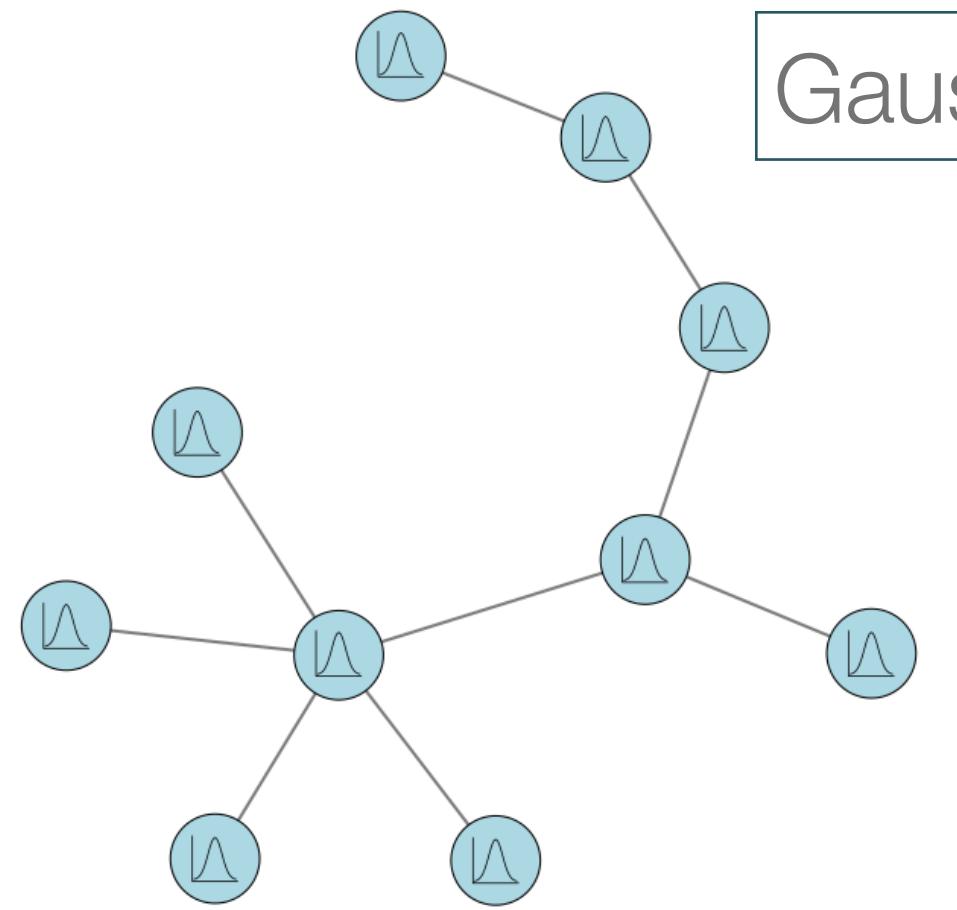






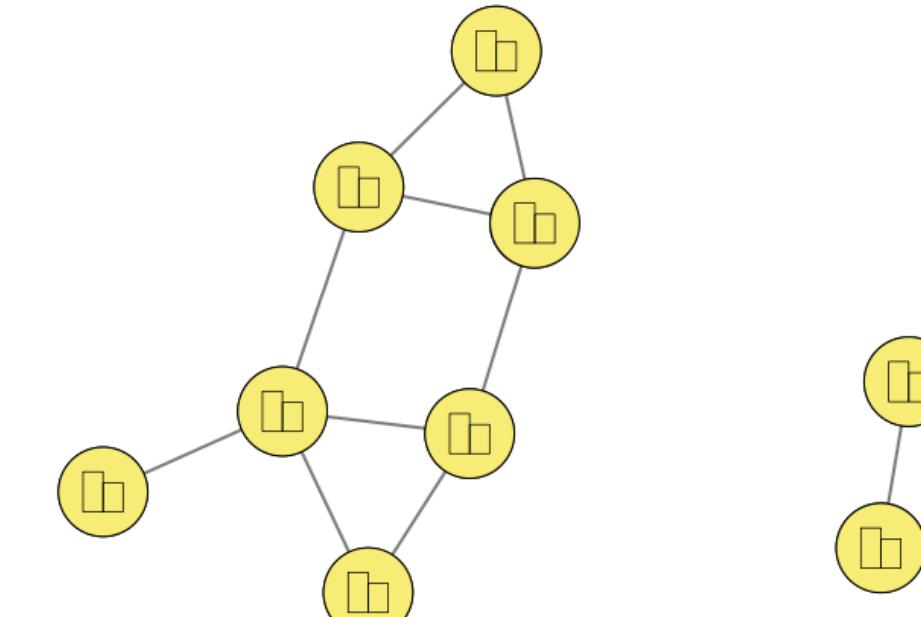
Estimation

Models

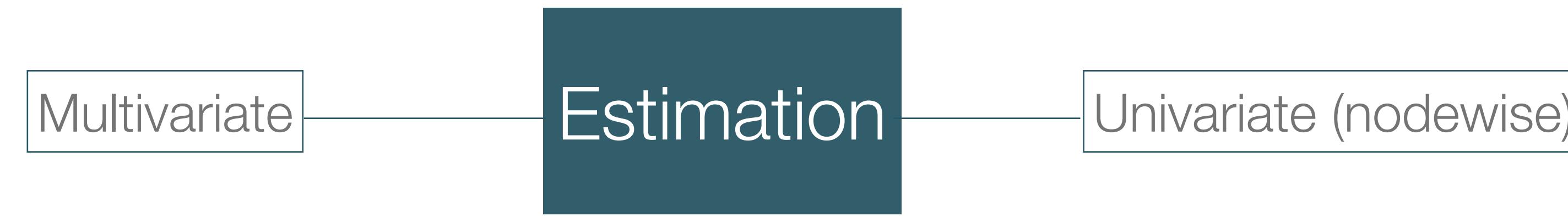
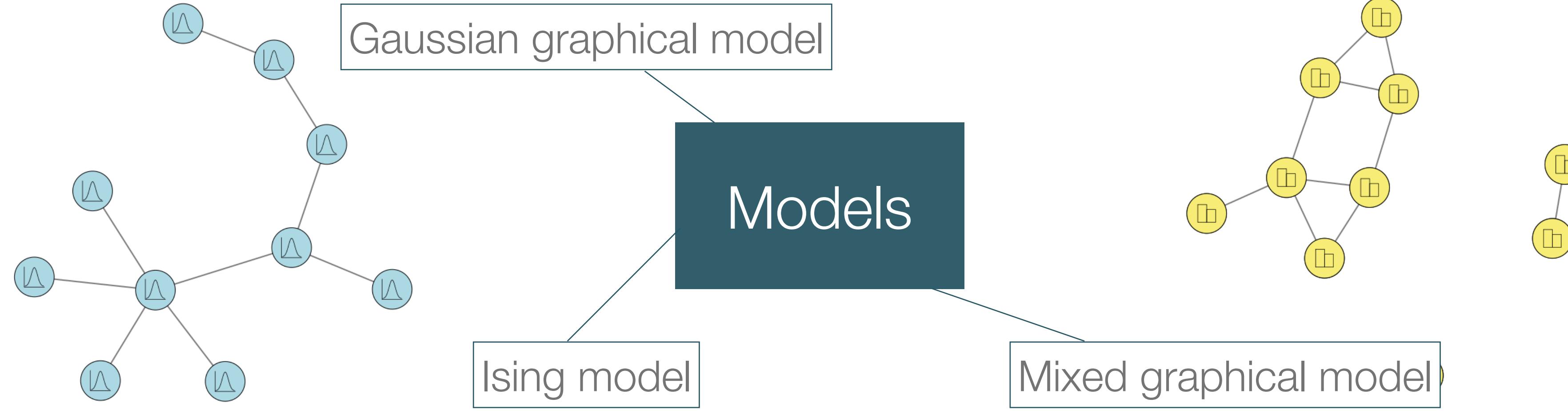


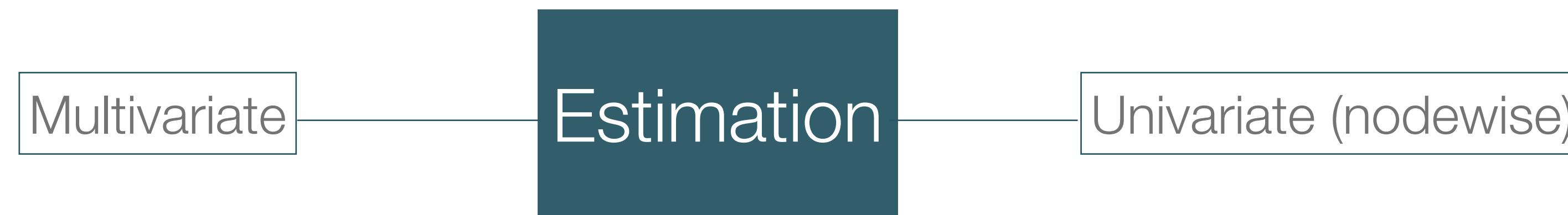
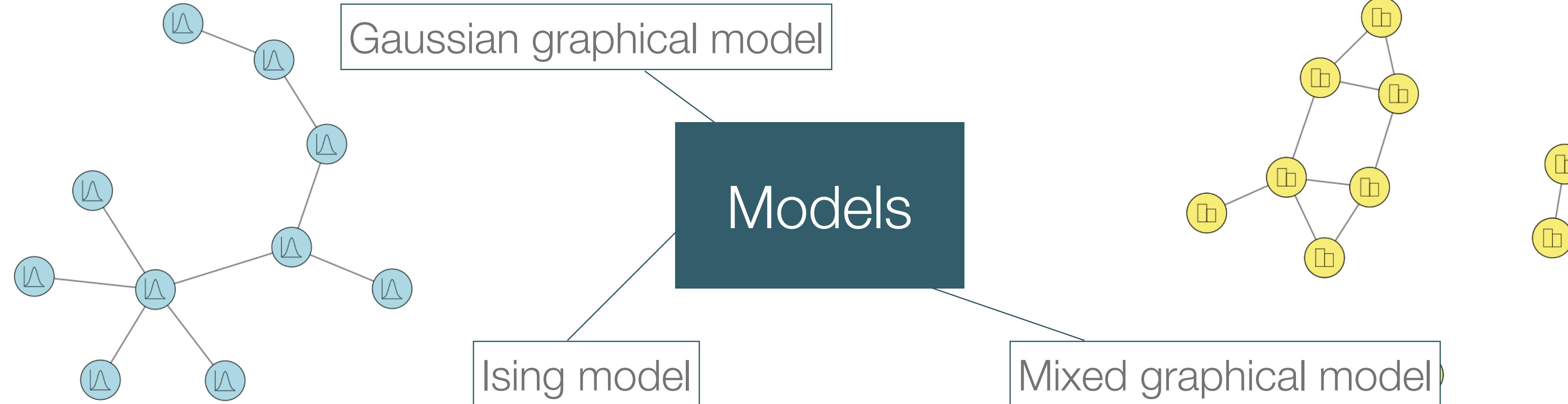
Gaussian graphical model

Ising model



Mixed graphical model





Selection

Spoiler

- Actual estimation in R is quite simple...

```
# Estimate a network:  
library("bootnet")  
Network <- estimateNetwork(Data, default = "...")  
  
# Perform bootstraps:  
Boots1 <- bootnet(Network, nCores = 8)  
Boots2 <- bootnet(Network, type = "case", nCores = 8)  
  
# Compare networks:  
library("NetworkComparisonTest")  
Comparison <- NCT(Network, Network2)
```

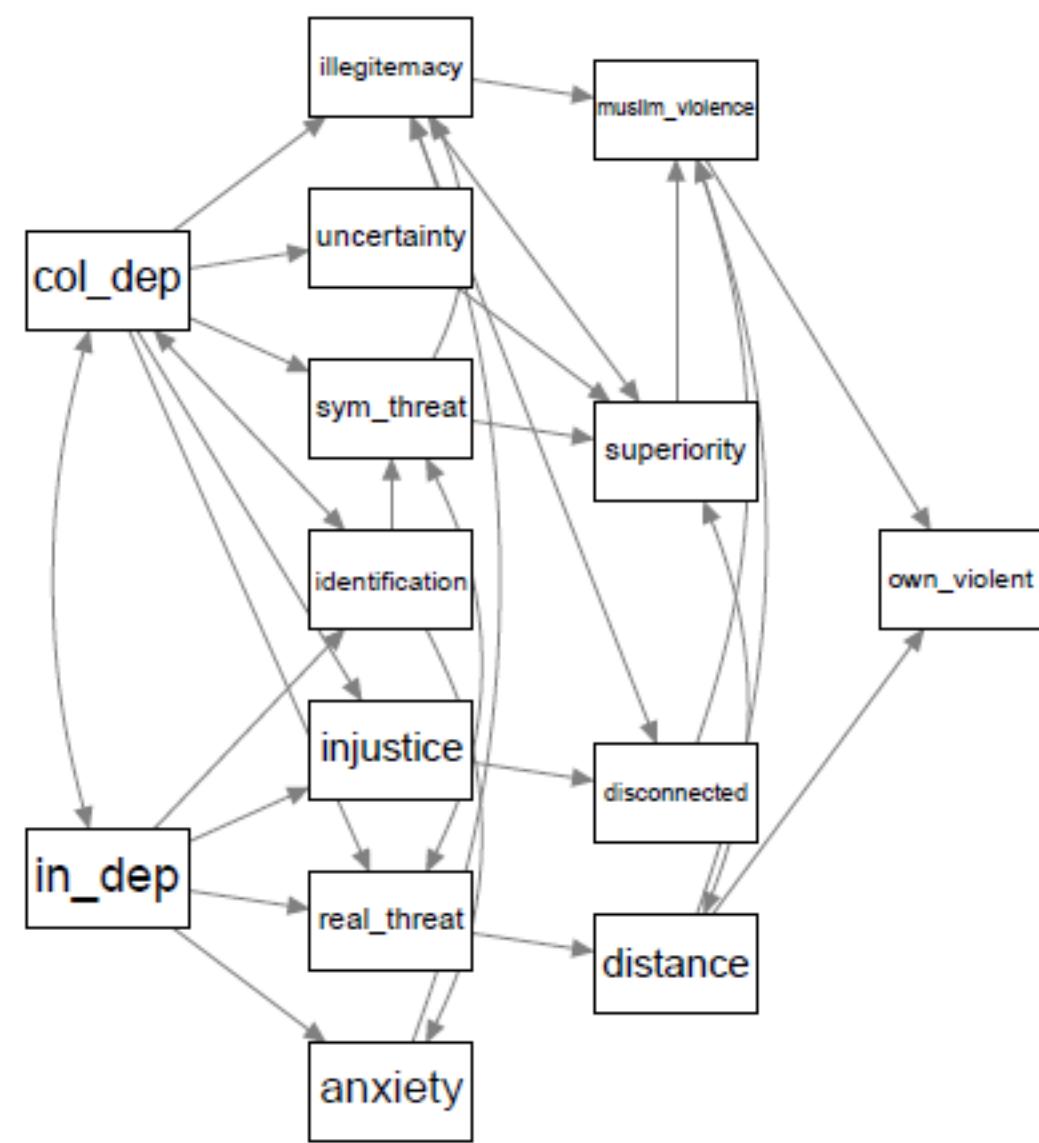
Network modelling: recap

- “Structure learning”: find optimal network structure within data
- Our focus: Pairwise Markov Random Fields
 - Undirected graphs
 - Conditional association between two variables after controlling or all other variables in the network
- Why Pairwise Markov Random Fields?

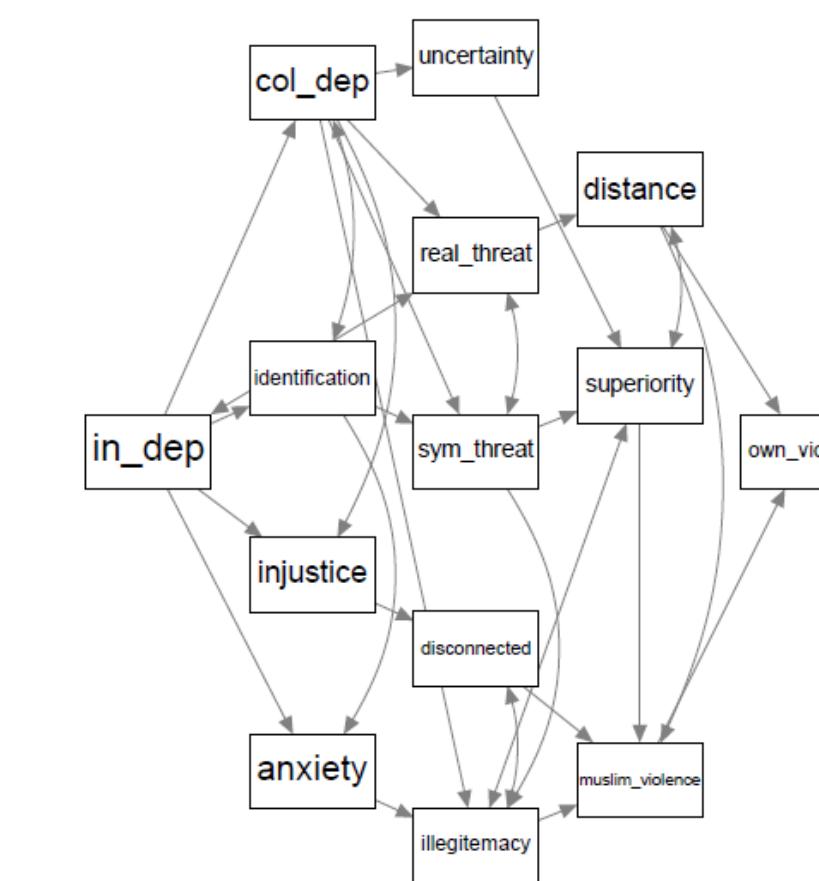
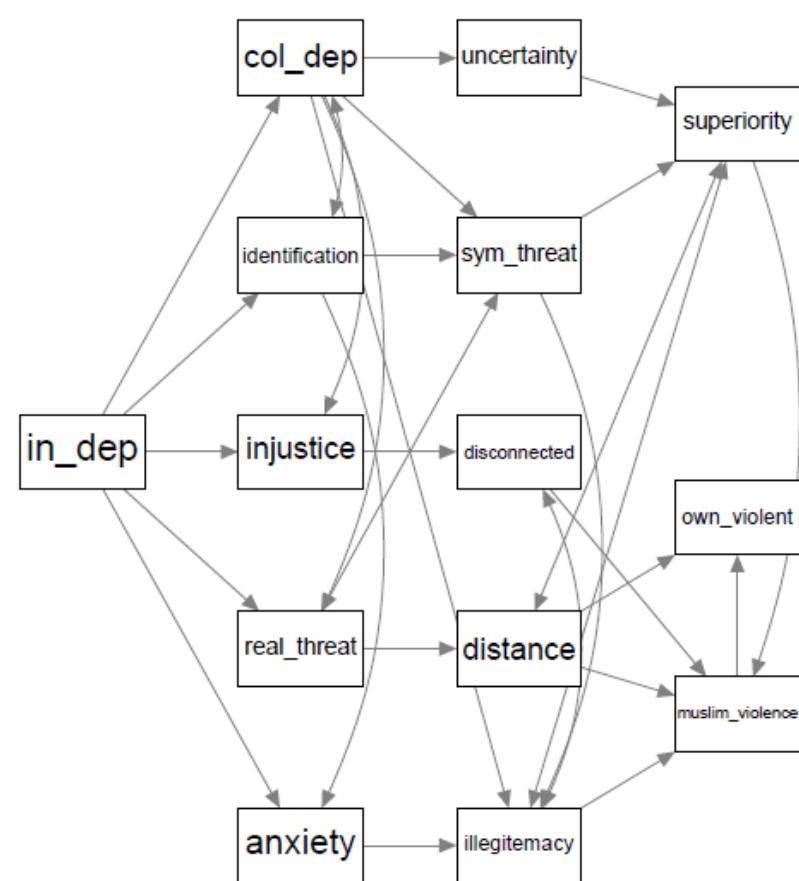
Network modelling: recap

- Directed networks (like structural equation models) are suited to test **confirmatory** theory
- Finding such directed models in **exploratory** analyses is hard without stringent assumptions
 - Acyclicity
 - Many equivalent models, all differently parameterised

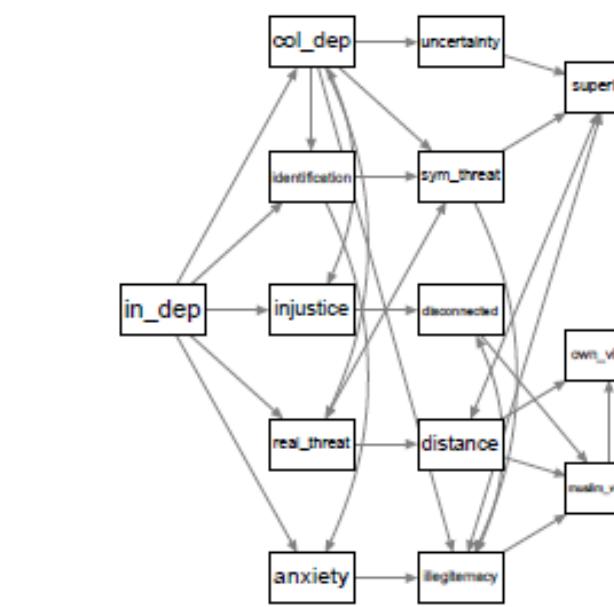
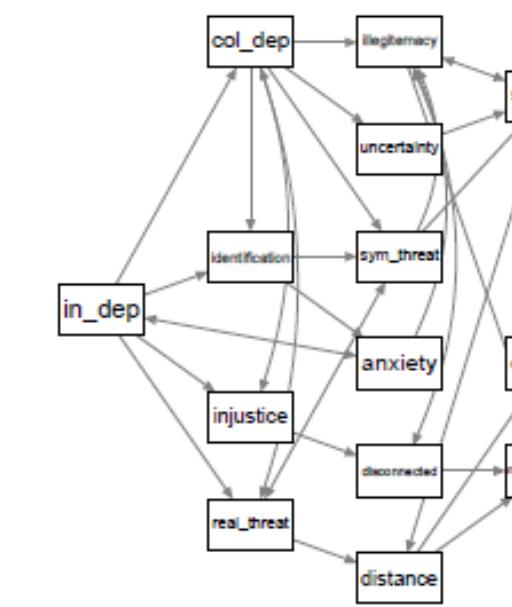
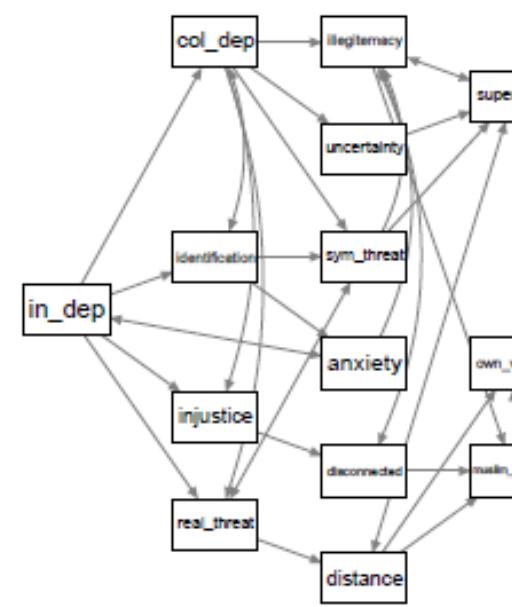
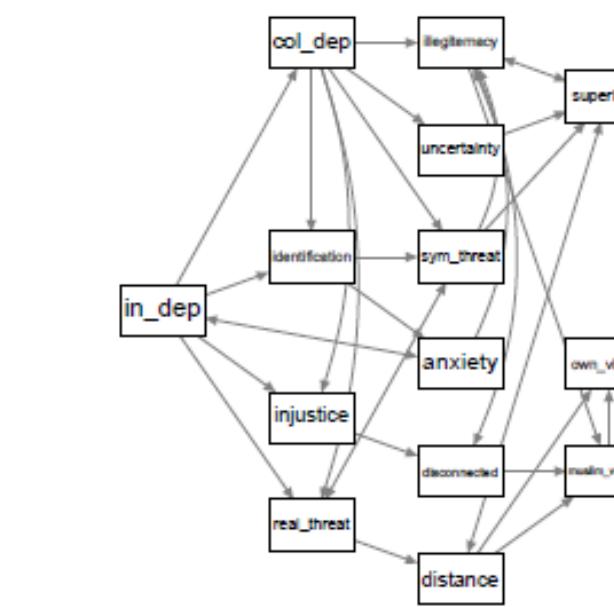
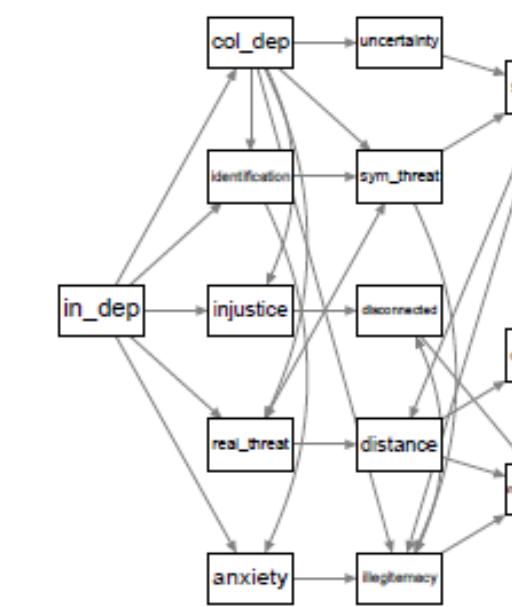
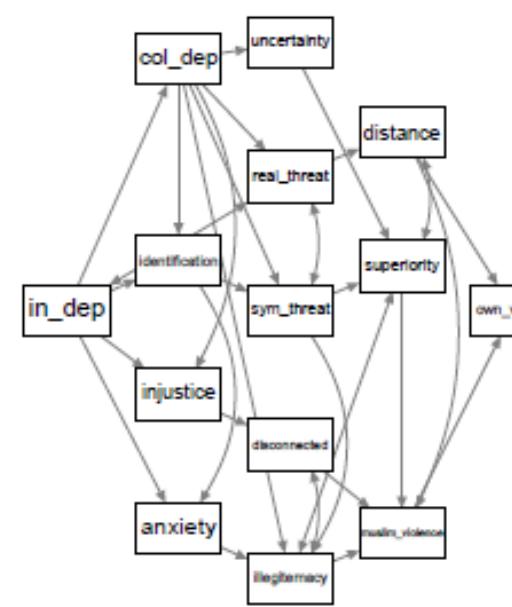
Model equivalence



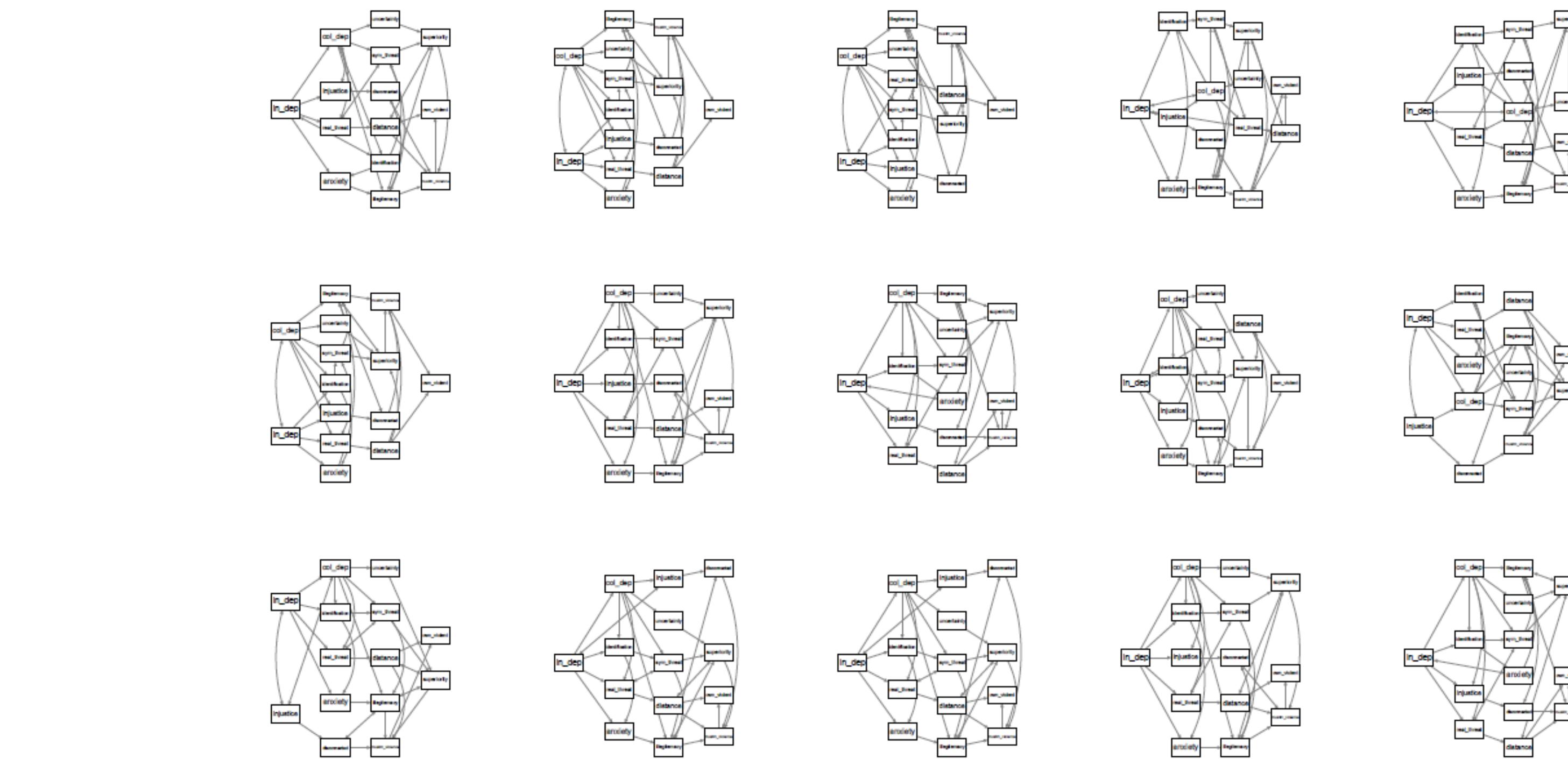
Model equivalence



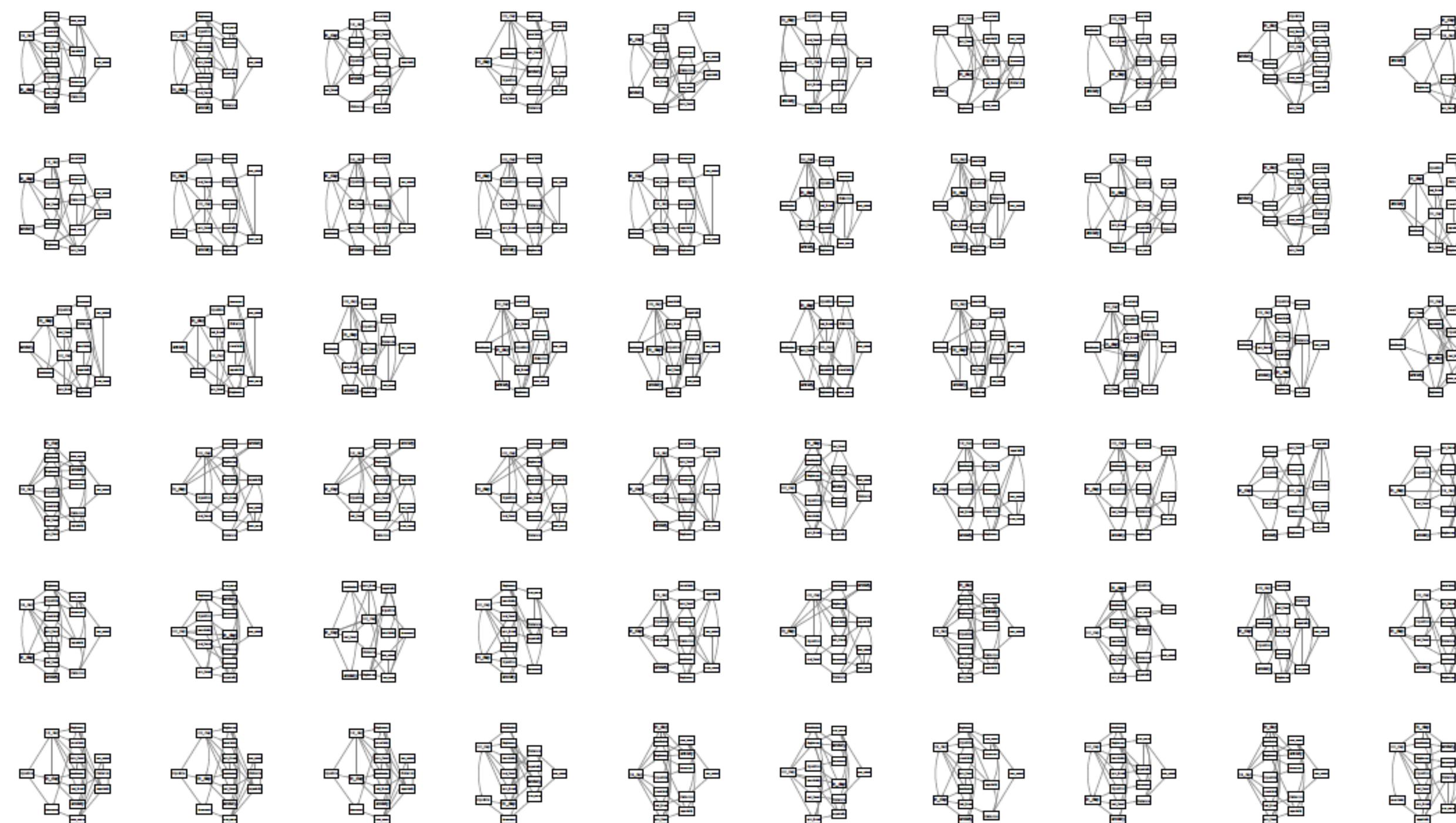
Model equivalence



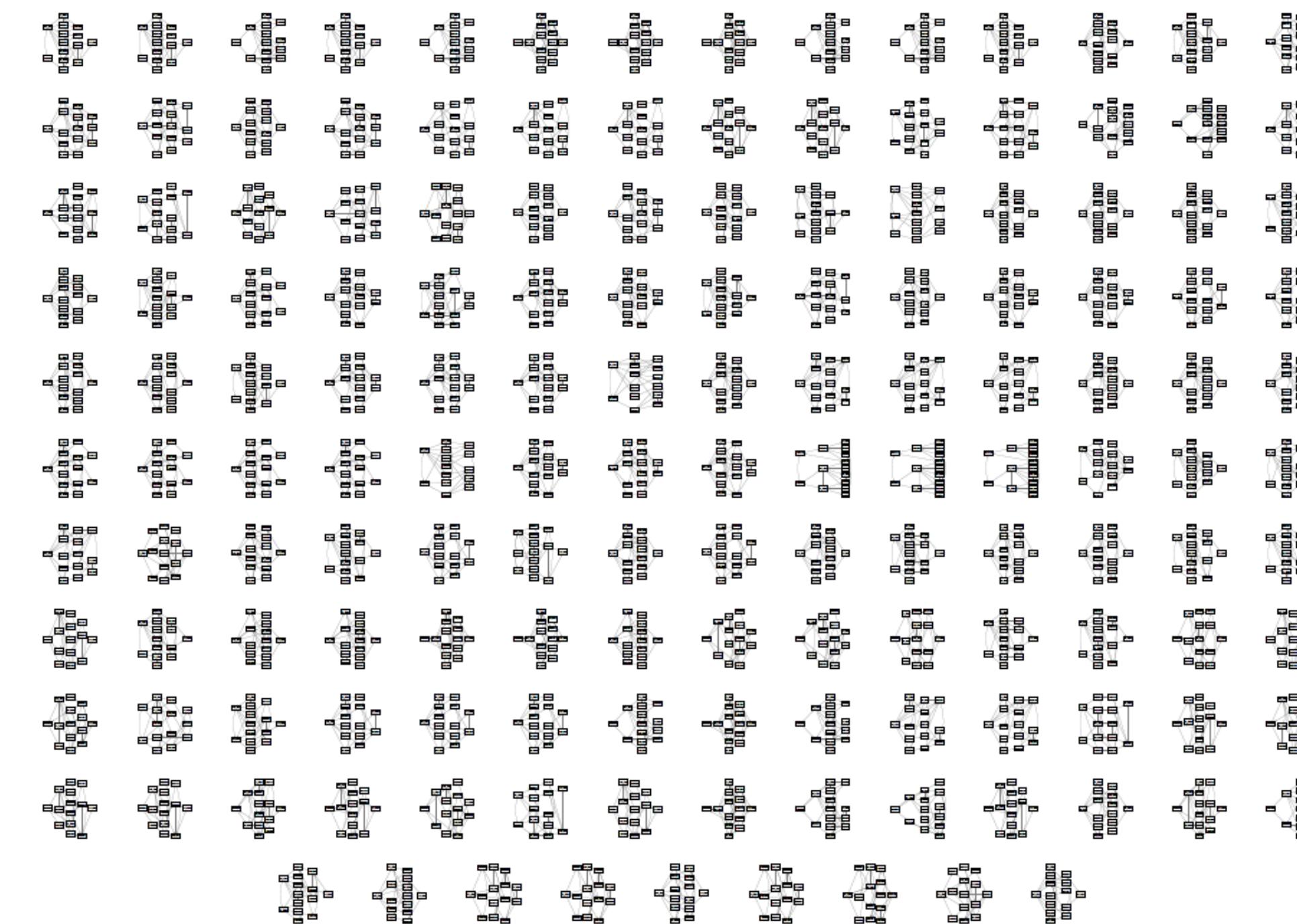
Model equivalence



Model equivalence



Model equivalence

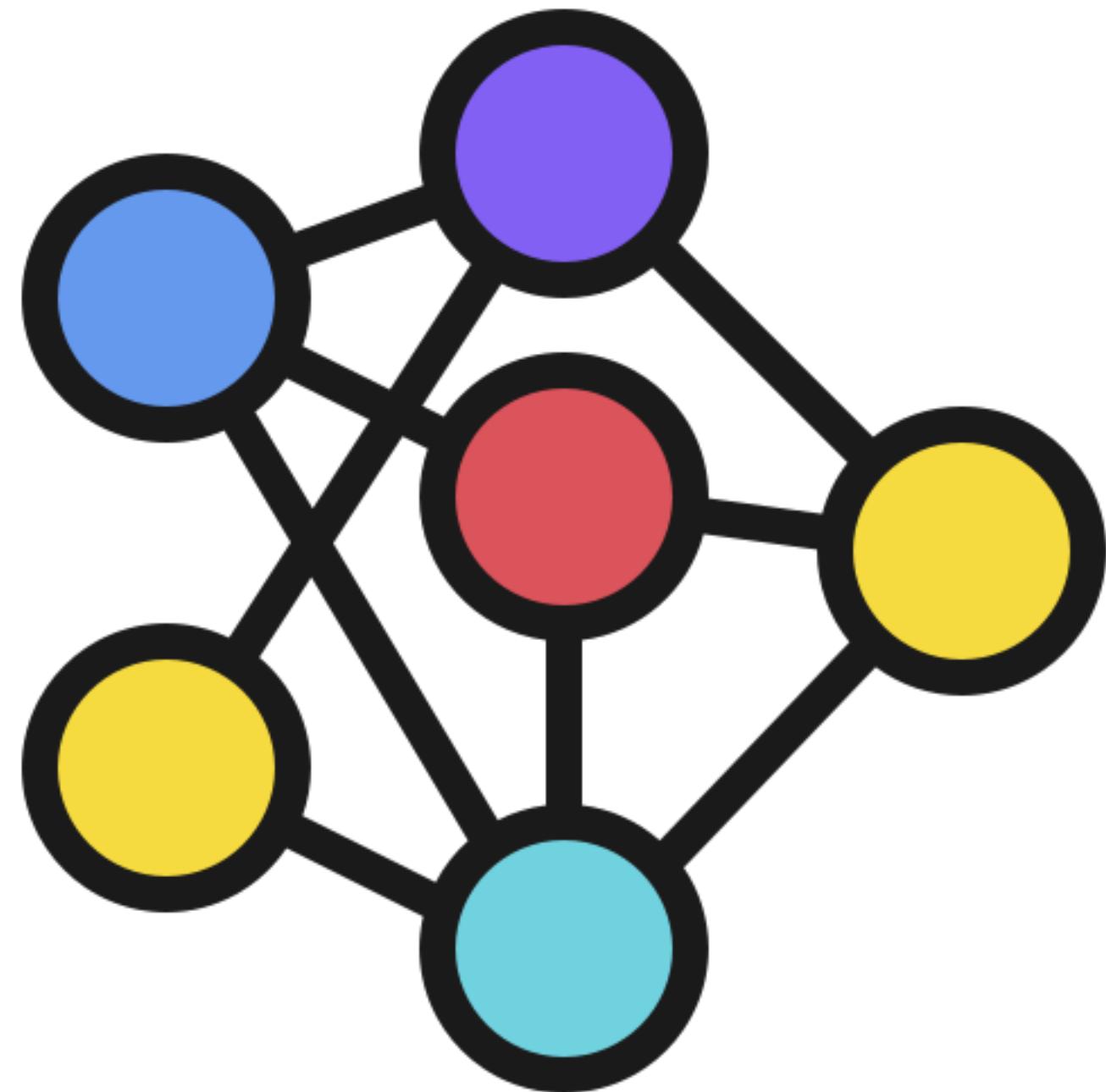


Network modelling

- Directed networks (like structural equation models) are suited to test **confirmatory** theory
- Finding such directed models in **exploratory** analyses is hard without stringent assumptions
 - Acyclicity
 - Many equivalent models, all differently parameterised
- Exploratory estimation of undirected networks is an attractive alternative:
 - No equivalent models
 - No assumption of acyclicity
 - Straightforward to estimate
 - Well parameterised
- These models are **pairwise Markov random fields**

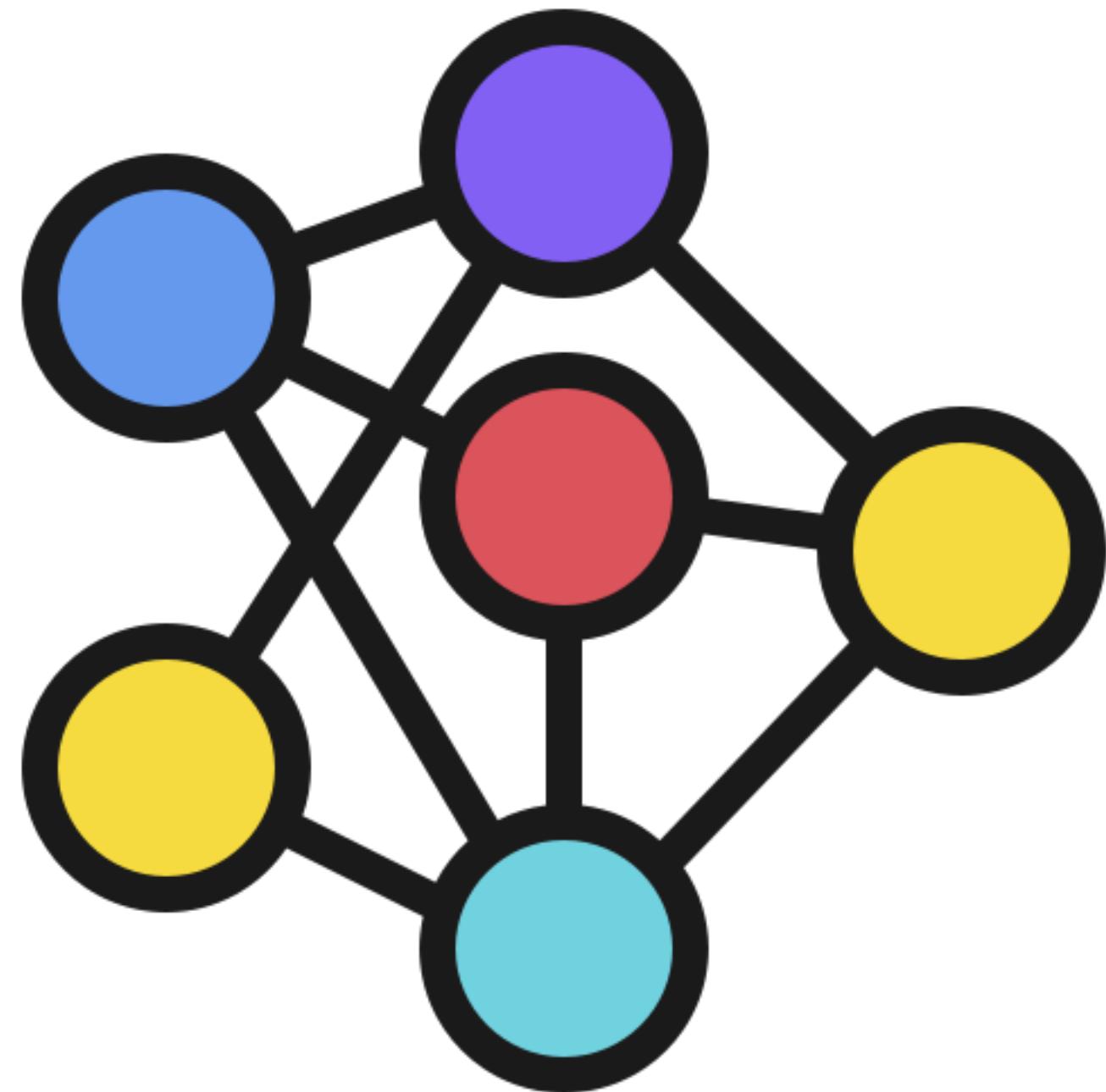
Pairwise Markov Random Fields

- Undirected network that encodes the *conditional association* between two variables after conditioning on all other variables in the network:
 - Two nodes are **connected** if they are *not independent conditional* on all other nodes in the network
 - Two nodes are **not connected** if they are *independent conditional* on all other nodes in the network
- Each connection (edge) is represented by a parameter
- The graph structure is encoded in a matrix of parameters



Pairwise Markov Random Fields

1. Statistical modelling framework
2. Showcasing predictive relationships
3. Generating causal hypotheses
4. Data generation model
5. Exploratory tool in factor analysis



Pairwise Markov Random Fields

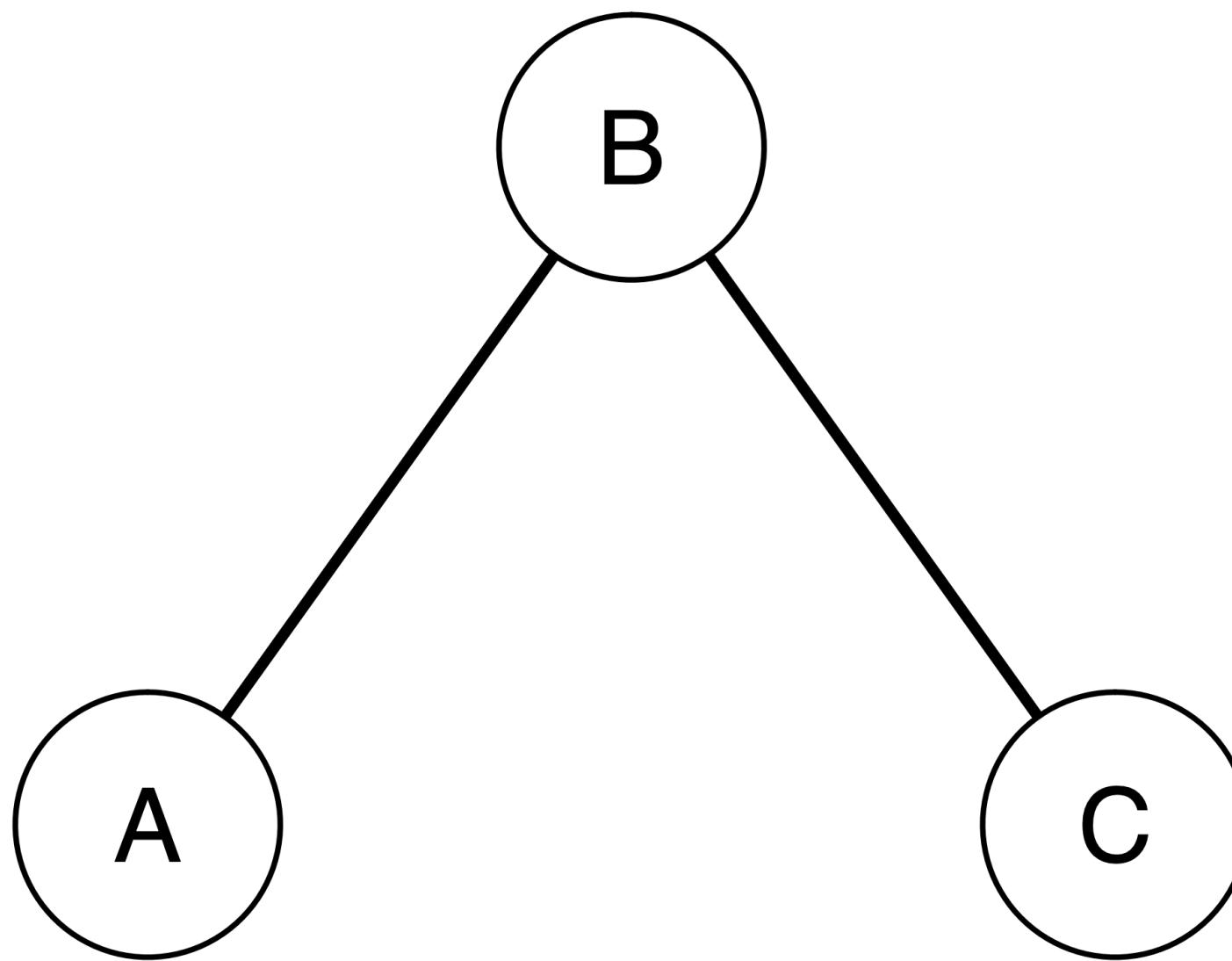
- 1. Statistical modelling framework**
2. Showcasing predictive relationships
3. Generating causal hypotheses
4. Data generation model
5. Exploratory tool in factor analysis



Pairwise Markov Random Fields

1. Statistical modelling framework

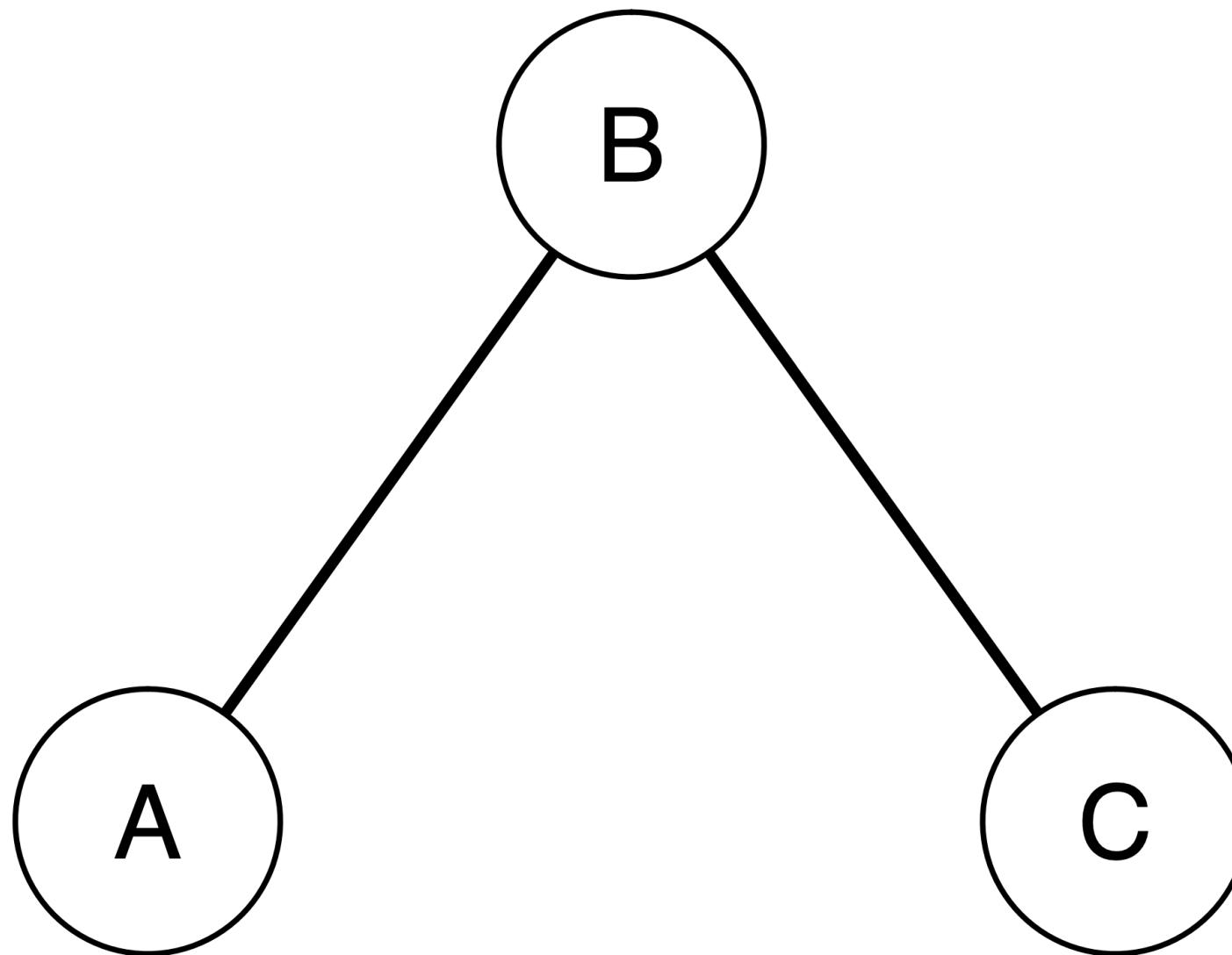
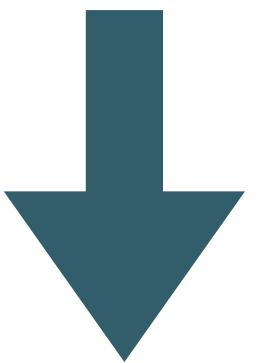
- If A and B interact
- and B and C interact
- then A and C are expected to be correlated



Pairwise Markov Random Fields

1. Statistical modelling framework

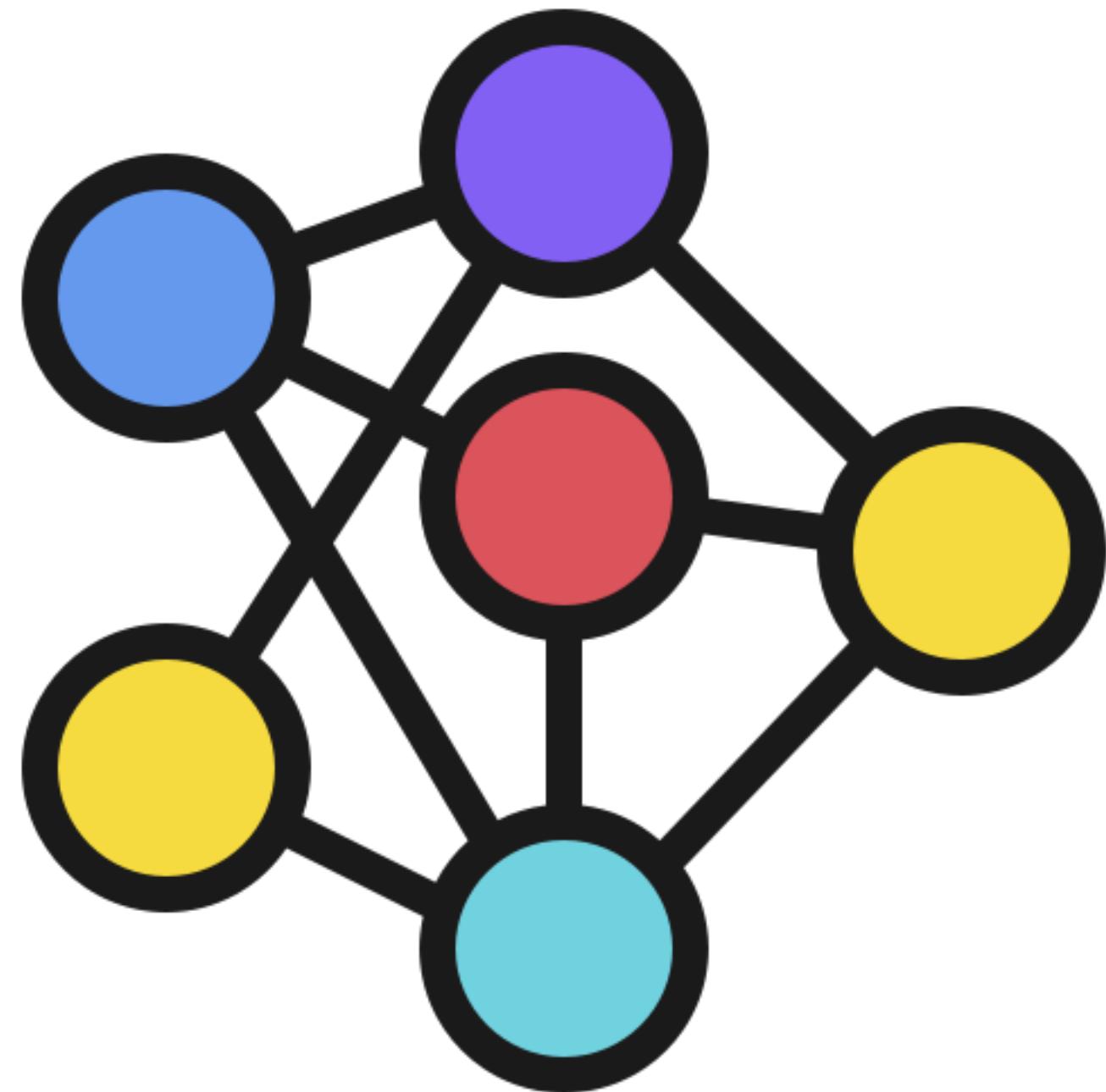
- If A and B interact
- and B and C interact
- then A and C are expected to be correlated



PMRFs allow to model *three* covariances using only *two* parameters → 1 degree of freedom

Pairwise Markov Random Fields

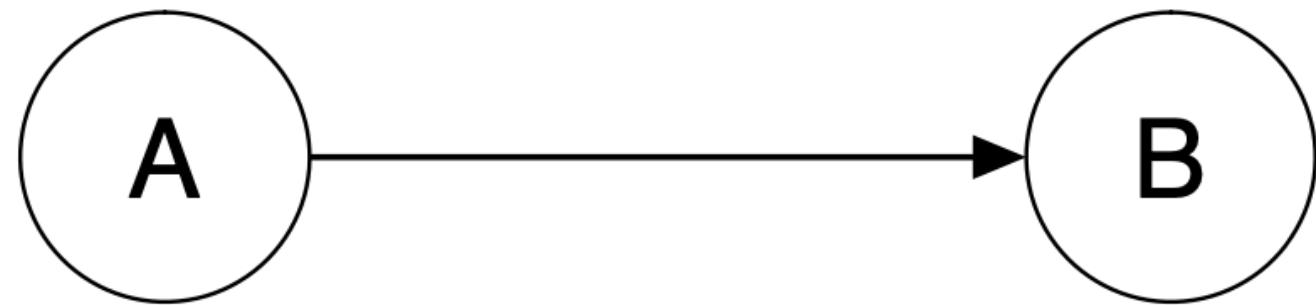
1. Statistical modelling framework
- 2. Showcasing predictive relationships**
3. Generating causal hypotheses
4. Data generation model
5. Exploratory tool in factor analysis



Pairwise Markov Random Fields

2. Showcasing predictive relationships

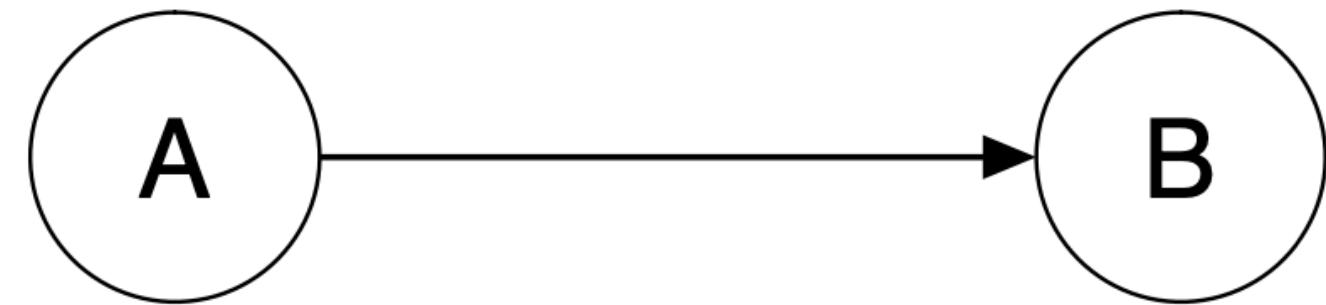
- If this is the true generating model, does:
 - A predict B?



Pairwise Markov Random Fields

2. Showcasing predictive relationships

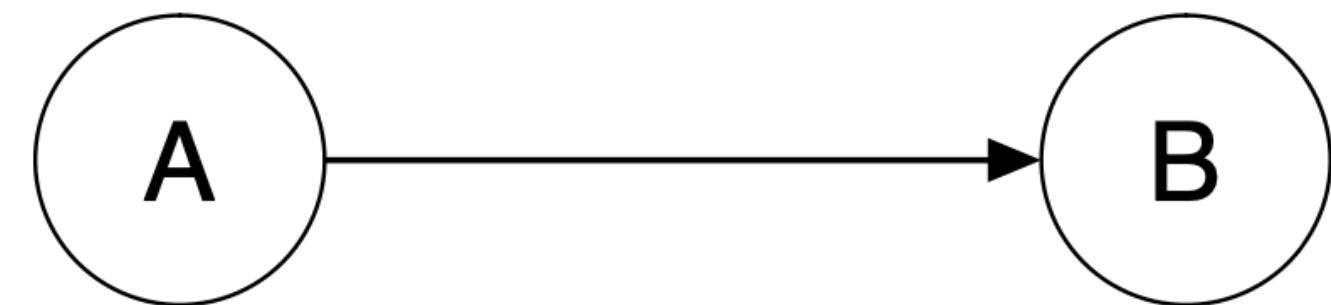
- If this is the true generating model, does:
 - A predict B? ✓



Pairwise Markov Random Fields

2. Showcasing predictive relationships

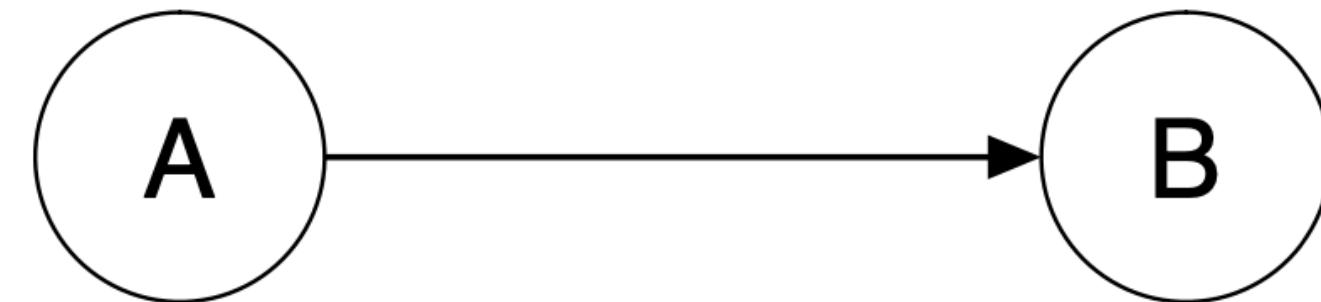
- If this is the true generating model, does:
 - A predict B? ✓
 - B predict A?



Pairwise Markov Random Fields

2. Showcasing predictive relationships

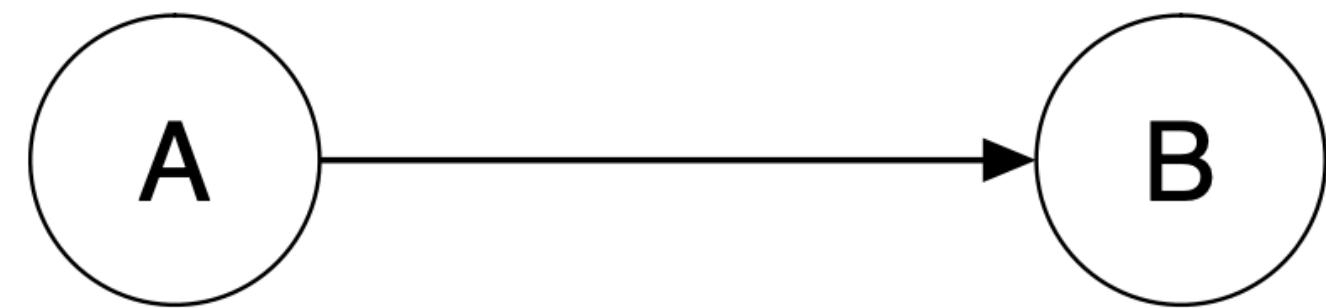
- If this is the true generating model, does:
 - A predict B? ✓
 - B predict A? ✓



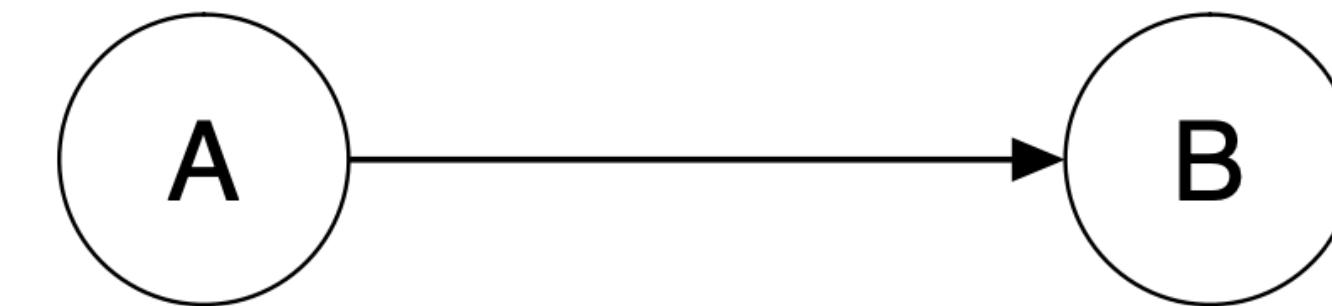
Pairwise Markov Random Fields

2. Showcasing predictive relationships

- If this is the true generating model, does:
 - A predict B? ✓
 - B predict A? ✓
- A predicts B *just as well* as B predicts A



Pairwise Markov Random Fields



2. Showcasing predictive relationships

- If this is the true generating model, does:
 - A predict B? ✓
 - B predict A? ✓
 - A predicts B *just as well* as B predicts A

```
A <- sample(c(0,1), 10000, replace = TRUE)  
B <- 1 * (runif(10000) < ifelse(A==1, 0.8, 0.2))
```

```
# Predict A from B (logistic regression):  
coef(glm(A ~ B, family = "binomial"))
```

```
## (Intercept) B  
## -1.359440 2.735295
```

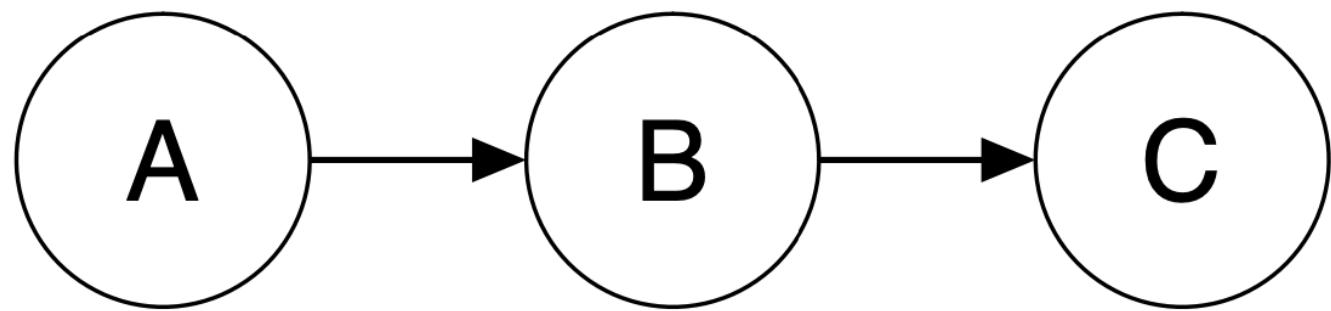
```
# Predict B from A (logistic regression):  
coef(glm(B ~ A, family = "binomial"))
```

```
## (Intercept) A  
## -1.367322 2.735295
```

Pairwise Markov Random Fields

2. Showcasing predictive relationships

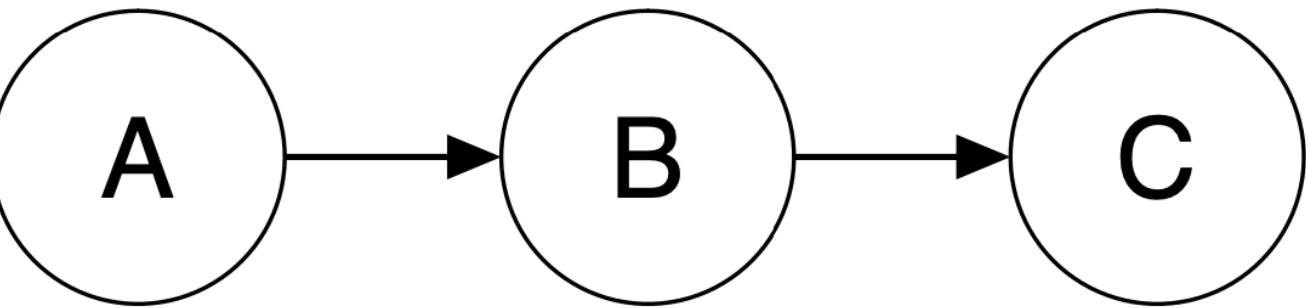
- If this is the true generating model, does:
 - A predict C, or vice versa?



Pairwise Markov Random Fields

2. Showcasing predictive relationships

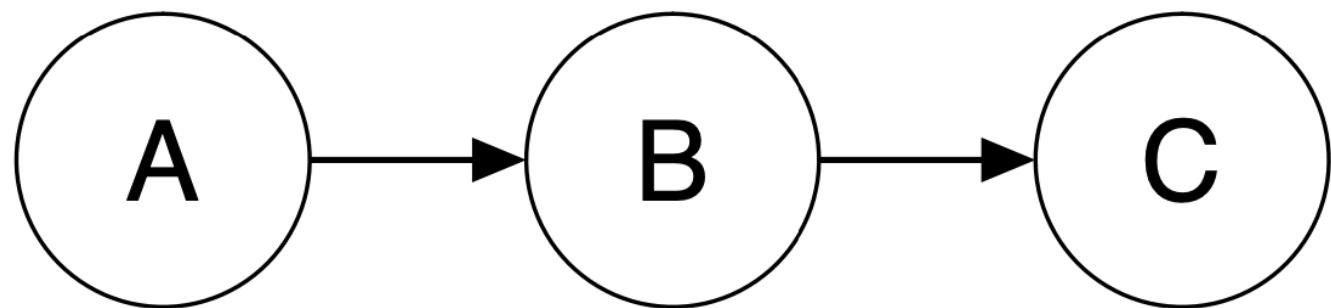
- If this is the true generating model, does:
 - A predict C, or vice versa? ✓



Pairwise Markov Random Fields

2. Showcasing predictive relationships

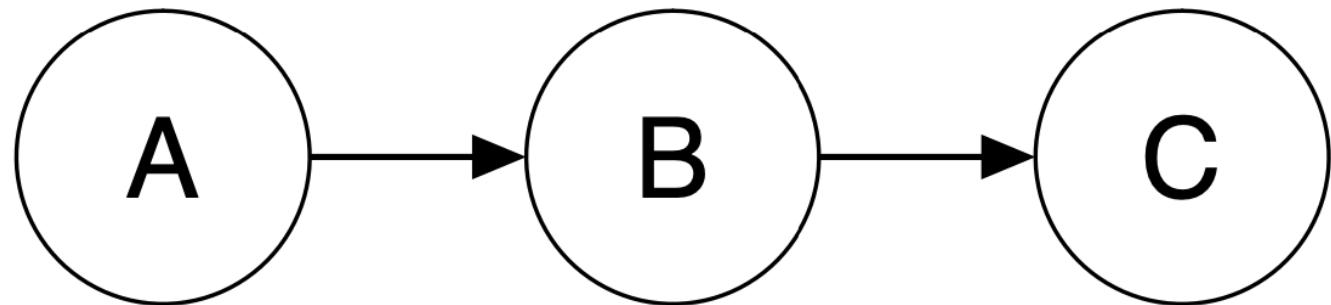
- If this is the true generating model, does:
 - A predict C, or vice versa? ✓
 - A predict C (or vice versa) when B is taken into account?



Pairwise Markov Random Fields

2. Showcasing predictive relationships

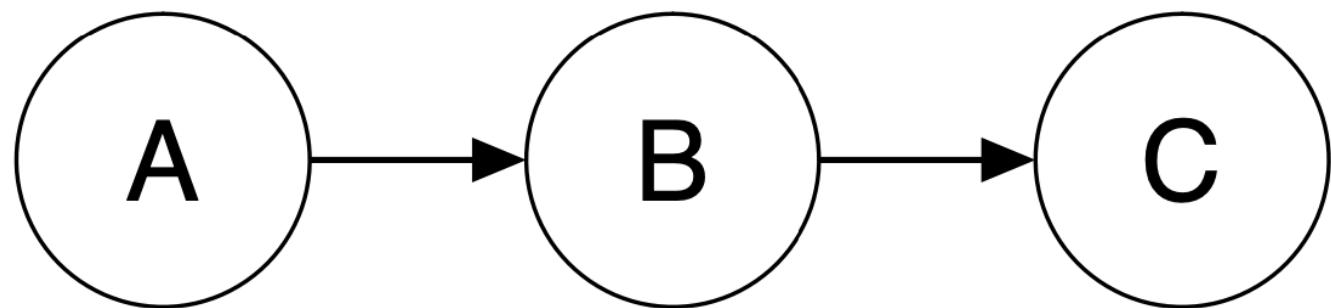
- If this is the true generating model, does:
 - A predict C, or vice versa? ✓
 - A predict C (or vice versa) when B is taken into account? ✗



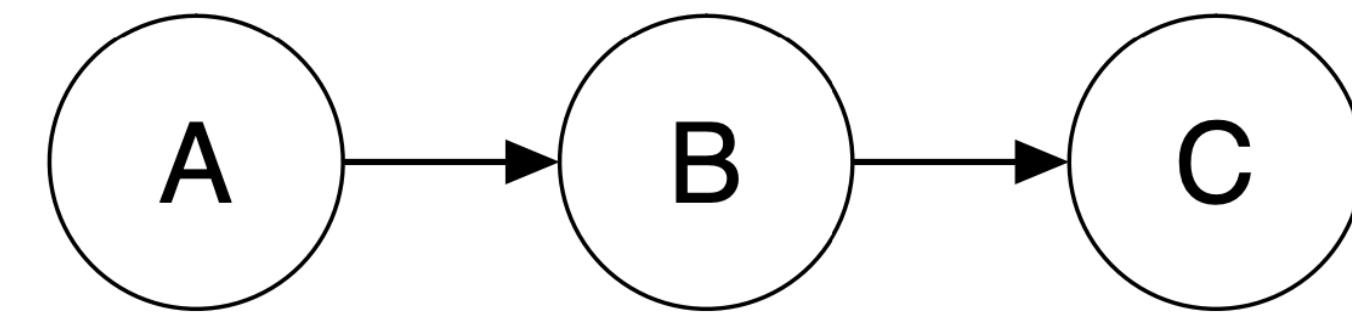
Pairwise Markov Random Fields

2. Showcasing predictive relationships

- If this is the true generating model, does:
 - A predict C, or vice versa? ✓
 - A predict C (or vice versa) when B is taken into account? ✗
- In a multiple (logistic) regression, A should not predict C (and vice versa) when B is also taken into account as predictor



Pairwise Markov Random Fields



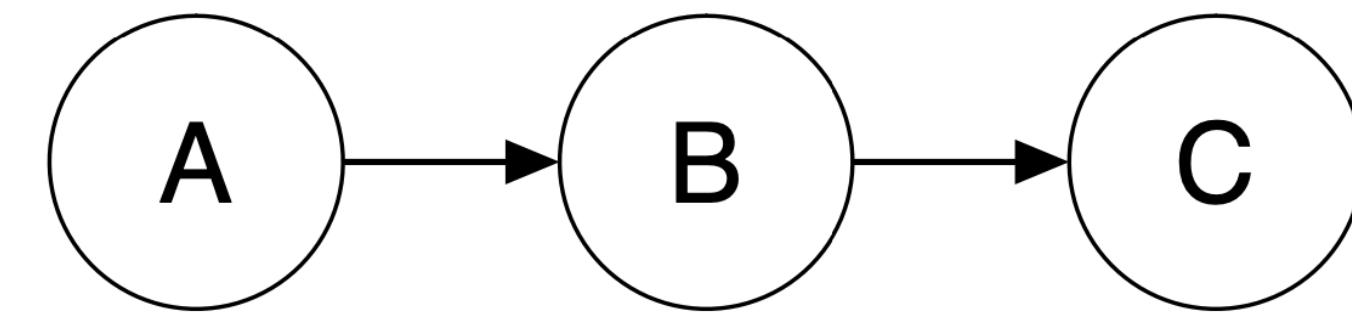
2. Showcasing predictive relationships

- If this is the true generating model, does:
 - A predict C, or vice versa? ✓
 - A predict C (or vice versa) when B is taken into account? ✗
- In a multiple (logistic) regression, A should not predict C (and vice versa) when B is also taken into account as predictor

```
# Generate data (Gaussian):  
A <- rnorm(10000)  
B <- A + rnorm(10000)  
C <- B + 2*rnorm(10000)
```

```
# Predict A from C:  
AonC <- lm(A ~ C)  
summary(AonC)  
  
##  
## Call:  
## lm(formula = A ~ C)  
##  
## Residuals:  
##     Min      1Q  Median      3Q     Max  
## -3.6790 -0.6168 -0.0087  0.6279  3.6256  
##  
## Coefficients:  
##                               Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -0.013331   0.009155  -1.456   0.145  
## C           0.172838   0.003718  46.491  <2e-16 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.'
```

Pairwise Markov Random Fields



2. Showcasing predictive relationships

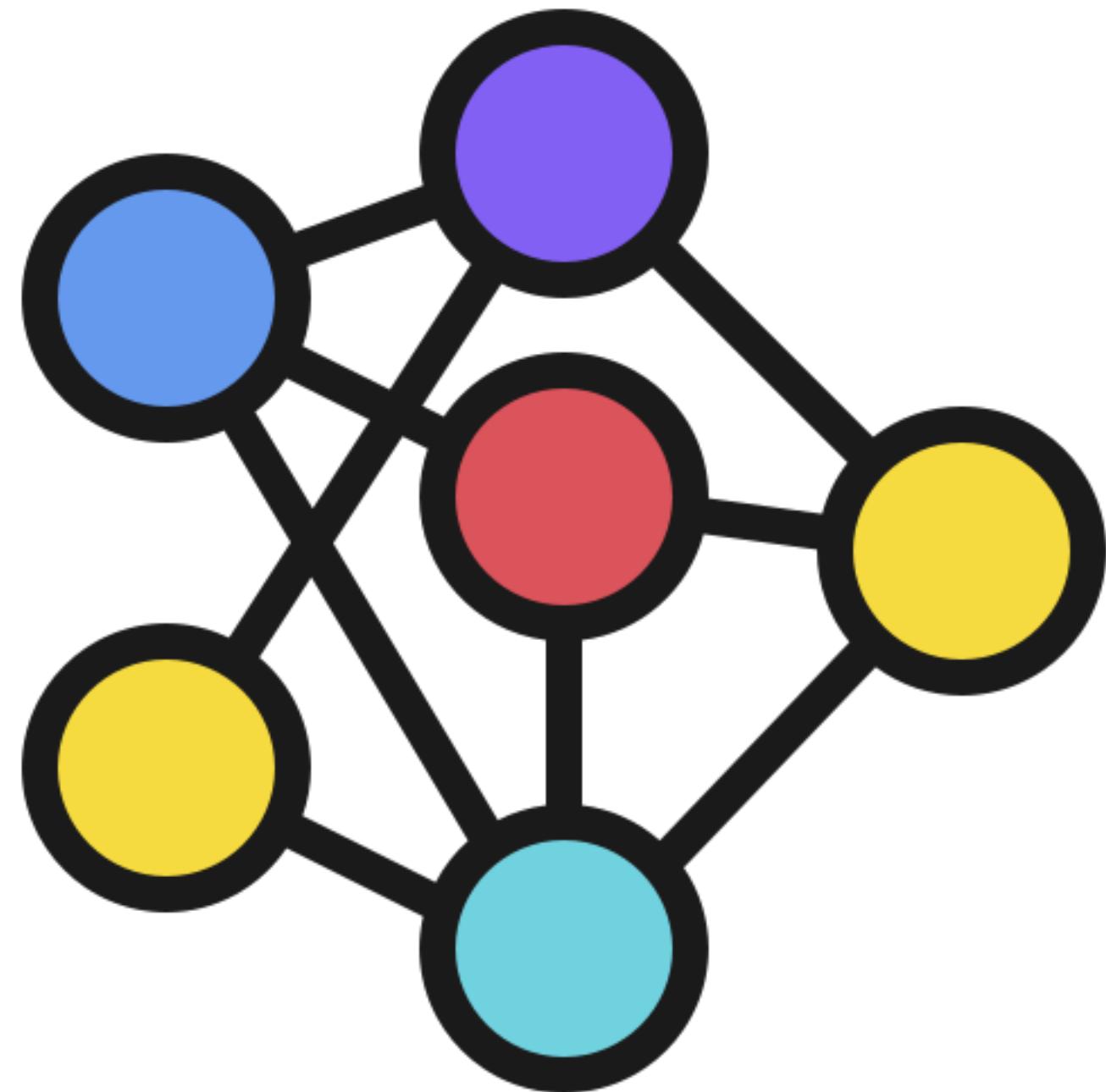
- If this is the true generating model, does:
 - A predict C, or vice versa? ✓
 - A predict C (or vice versa) when B is taken into account? ✗
- In a multiple (logistic) regression, A should not predict C (and vice versa) when B is also taken into account as predictor

```
# Generate data (Gaussian):  
A <- rnorm(10000)  
B <- A + rnorm(10000)  
C <- B + 2*rnorm(10000)
```

```
# Predict A from B and C:  
AonBC <- lm(A ~ B + C)  
summary(AonBC)  
  
##  
## Call:  
## lm(formula = A ~ B + C)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max  
## -2.95242 -0.47027 -0.00241  0.47033  2.61127  
##  
## Coefficients:  
##                               Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -0.009806    0.007061  -1.389   0.165  
## B            0.502434    0.006089  82.515 <2e-16 ***  
## C            0.005489    0.003512   1.563   0.118  
## ---
```

Pairwise Markov Random Fields

1. Statistical modelling framework
2. Showcasing predictive relationships
- 3. Generating causal hypotheses**
4. Data generation model
5. Exploratory tool in factor analysis

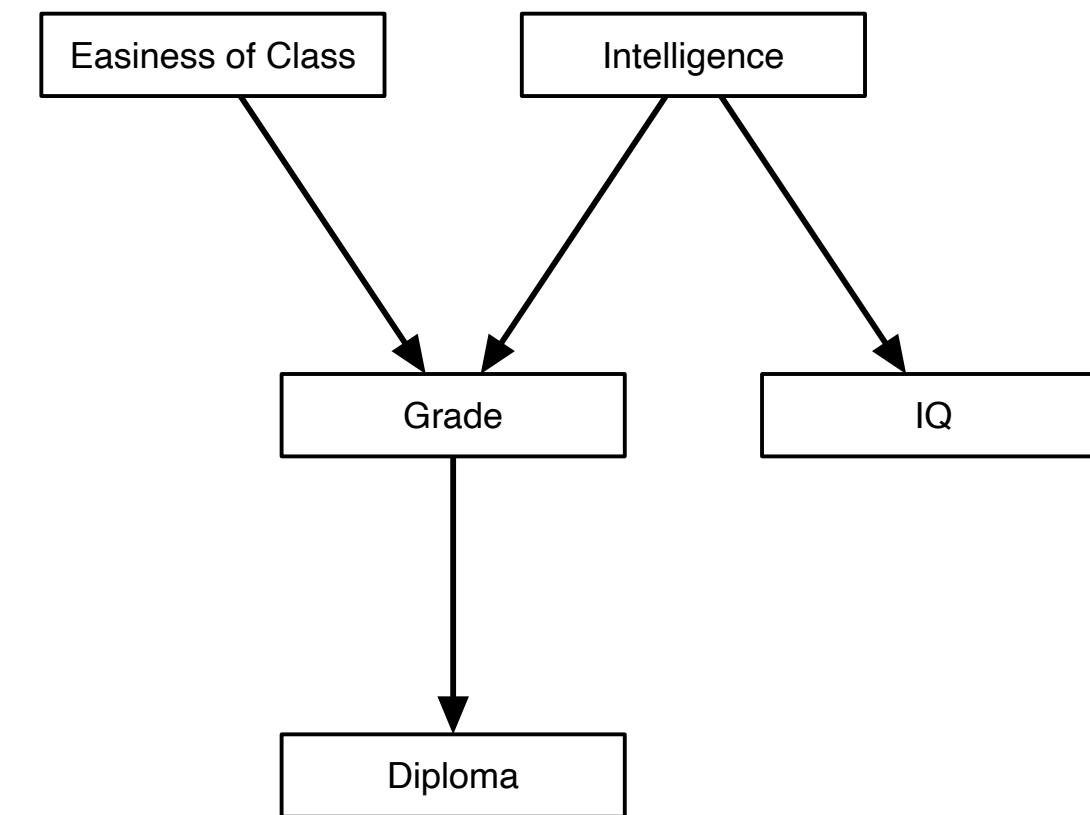


Pairwise Markov Random Fields

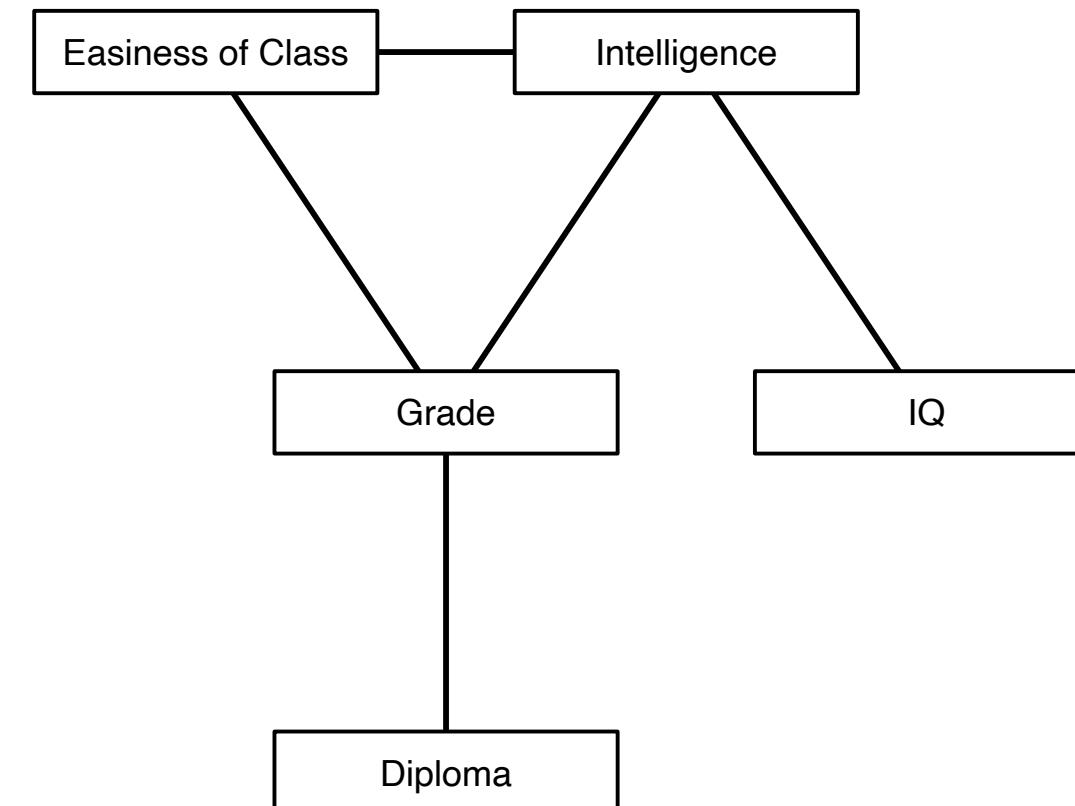
2. Generating causal hypotheses

- $A - B - C$
- Is equivalent to three causal structures:
 1. $A \rightarrow B \rightarrow C$ (chain)
 2. $A \leftarrow B \leftarrow C$ (chain)
 3. $A \leftarrow B \rightarrow C$ (common cause structure)

Causal model

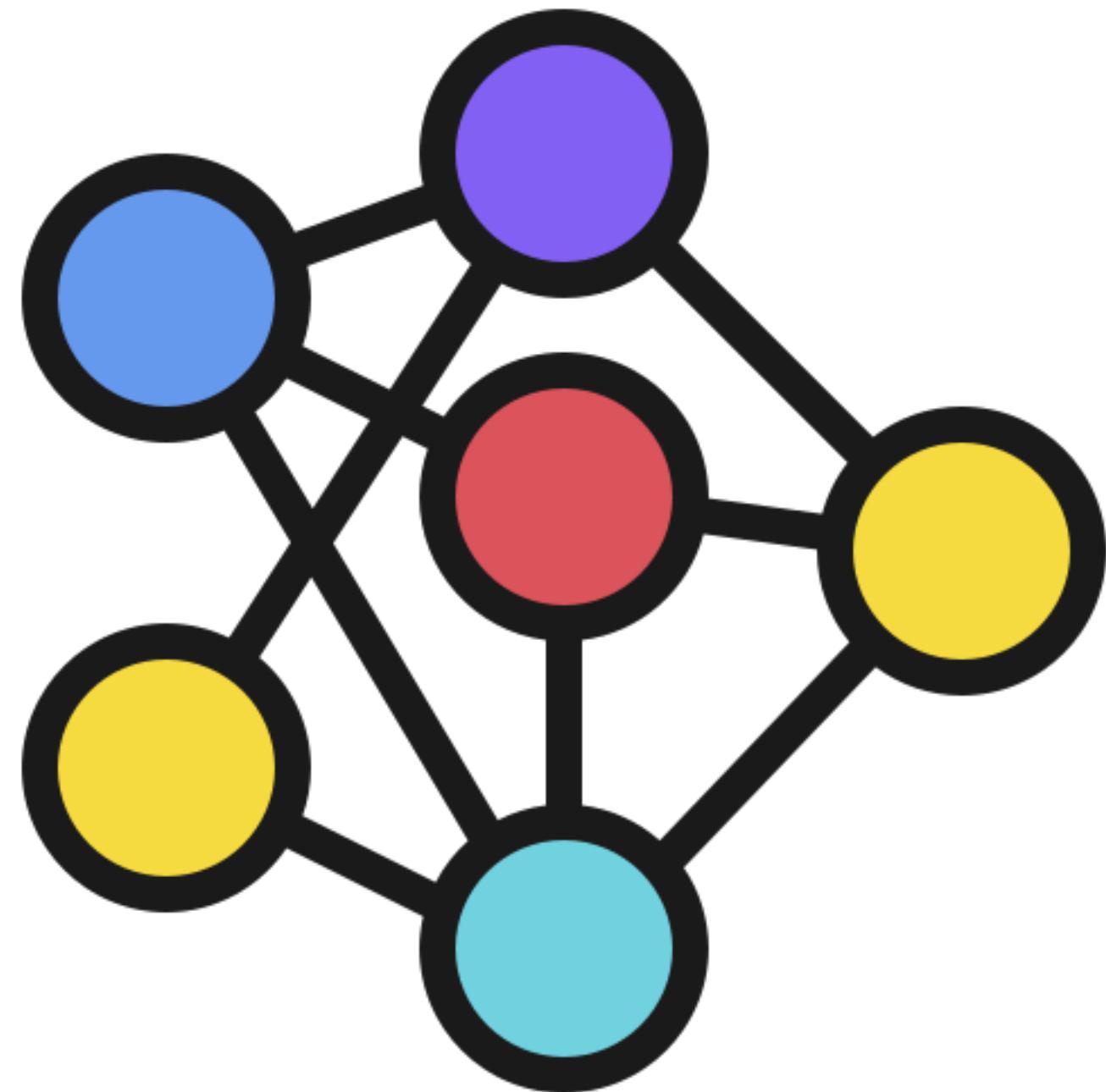


Markov Random Field



Pairwise Markov Random Fields

1. Statistical modelling framework
2. Showcasing predictive relationships
3. Generating causal hypotheses
- 4. Data generation model**
5. Exploratory tool in factor analysis



Pairwise Markov Random Fields

4. Data generation model

- Undirected network models as generating structures
 - Assumption of bi-directional causal relations (i.e., symmetrical effects)
 - For example: when you expect symptoms to mutually reinforce each other and intervening on symptom A would lead to a change in B, and intervening on symptom B would lead to a change in A



RESEARCH ARTICLE

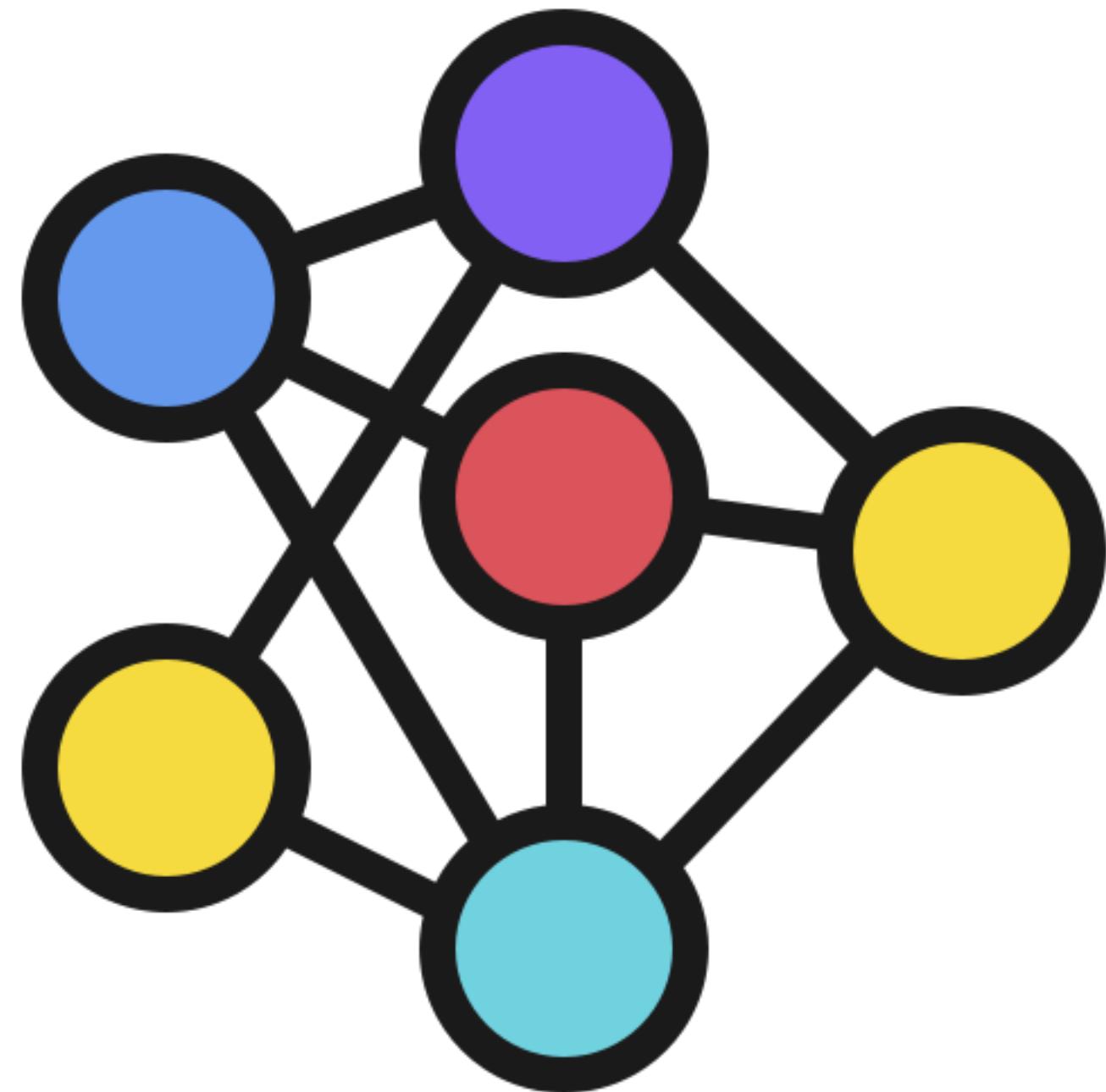
Major Depression as a Complex Dynamic System

Angélique O. J. Cramer^{1*}, Claudia D. van Borkulo¹, Erik J. Giltay², Han L. J. van der Maas¹, Kenneth S. Kendler³, Marten Scheffer⁴, Denny Borsboom¹

¹ Psychological Methods, University of Amsterdam, Amsterdam, the Netherlands, ² Department of Psychiatry, Leids Universitair Medisch Centrum, Leiden, the Netherlands, ³ Virginia Institute of Psychiatric and Behavioral Genetics, Virginia Commonwealth University, Richmond, Virginia, United States of America, ⁴ Department of Aquatic Ecology, Wageningen University, Wageningen, the Netherlands

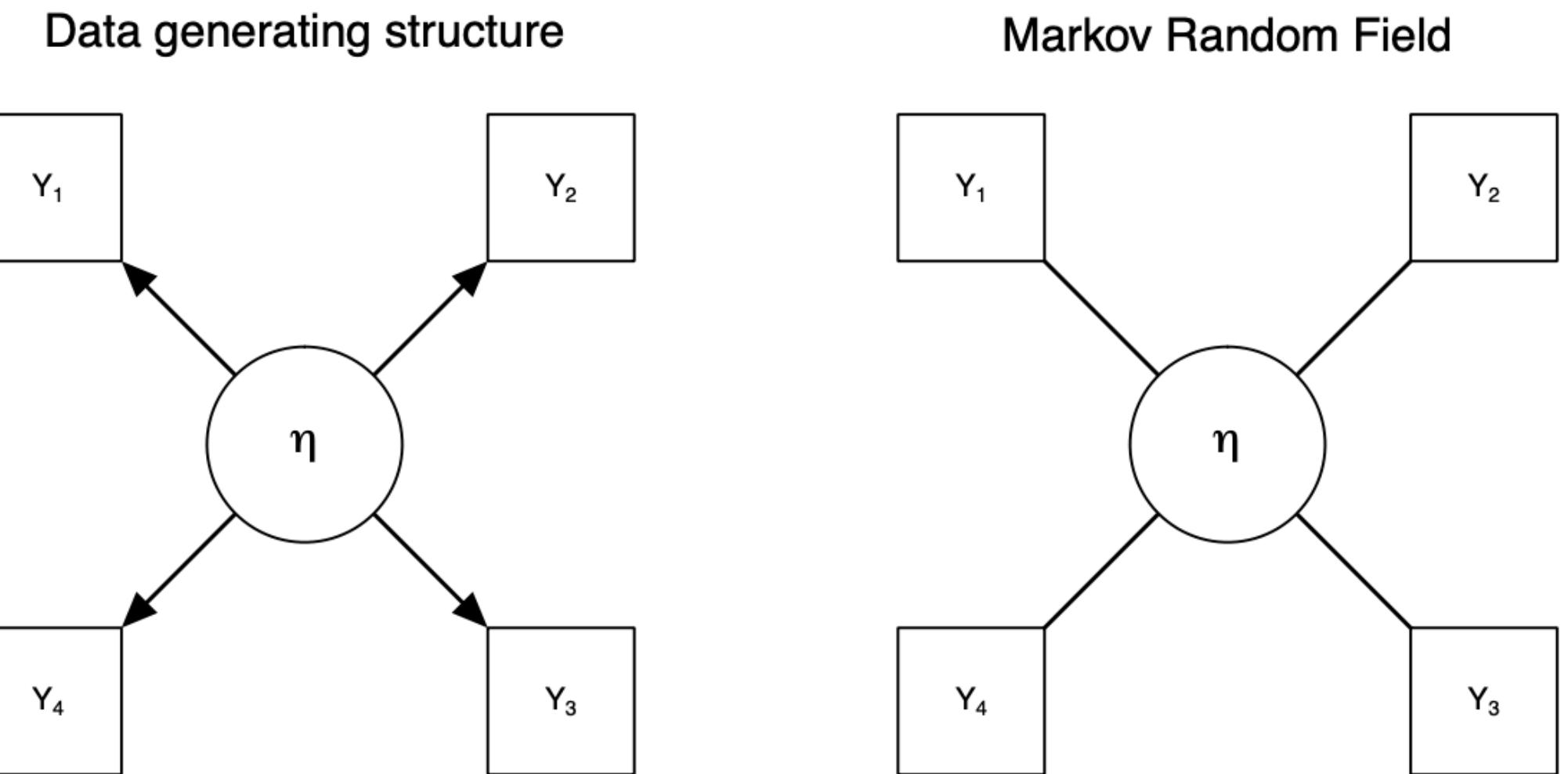
Pairwise Markov Random Fields

1. Statistical modelling framework
2. Showcasing predictive relationships
3. Generating causal hypotheses
4. Data generation model
5. **Exploratory tool in factor analysis**



Pairwise Markov Random Fields

5. Exploratory tool in factor analysis

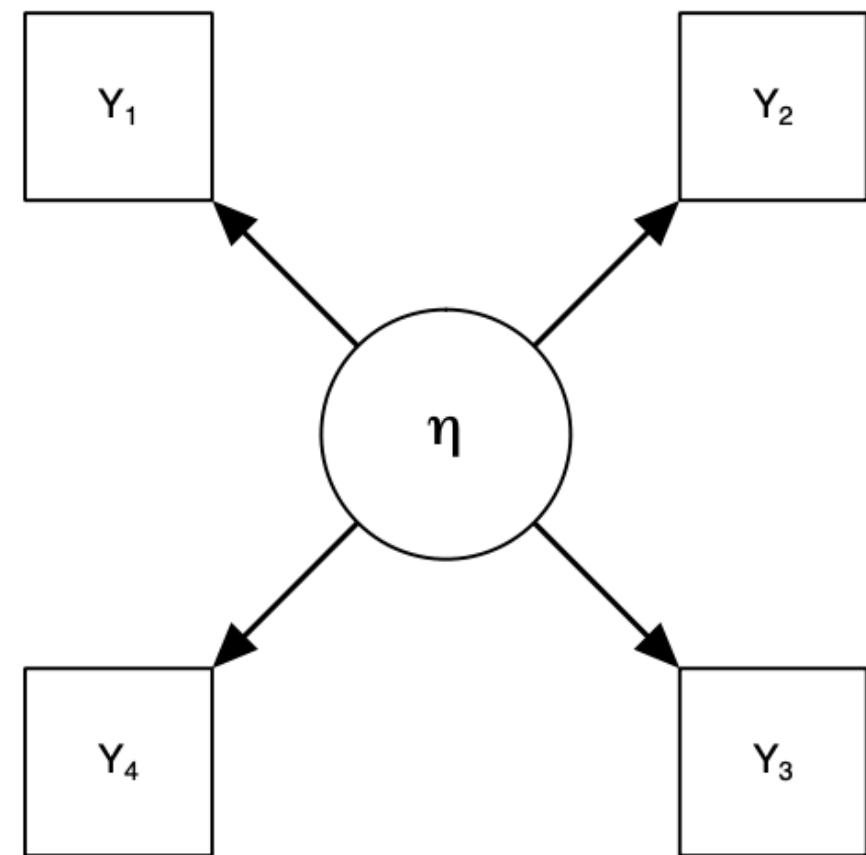


- If we could condition on η

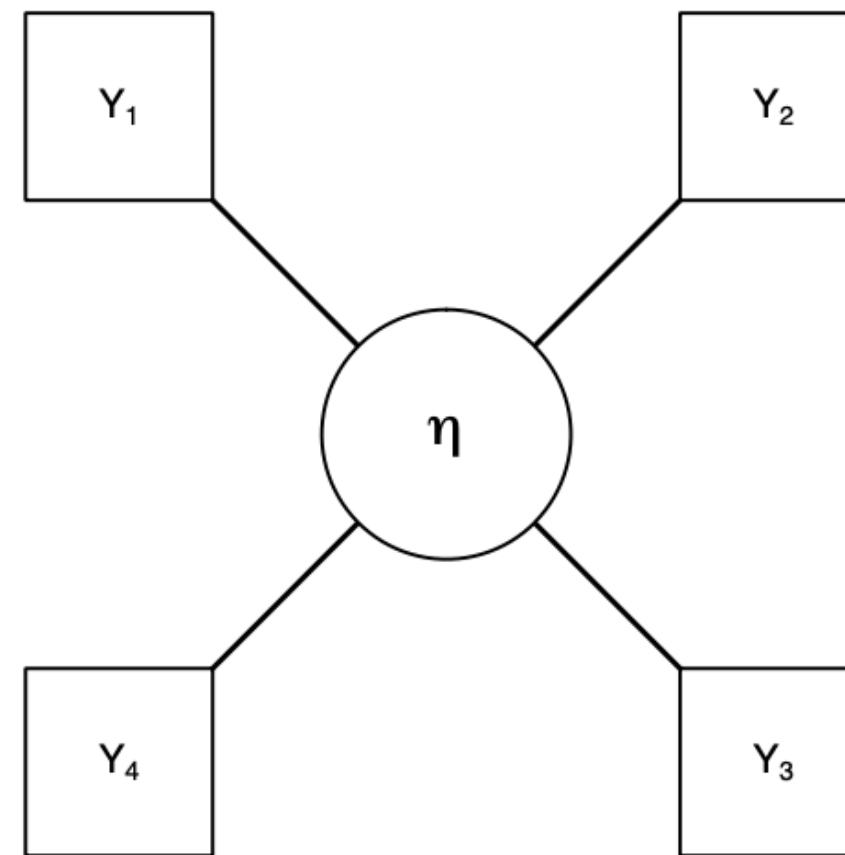
Pairwise Markov Random Fields

5. Exploratory tool in factor analysis

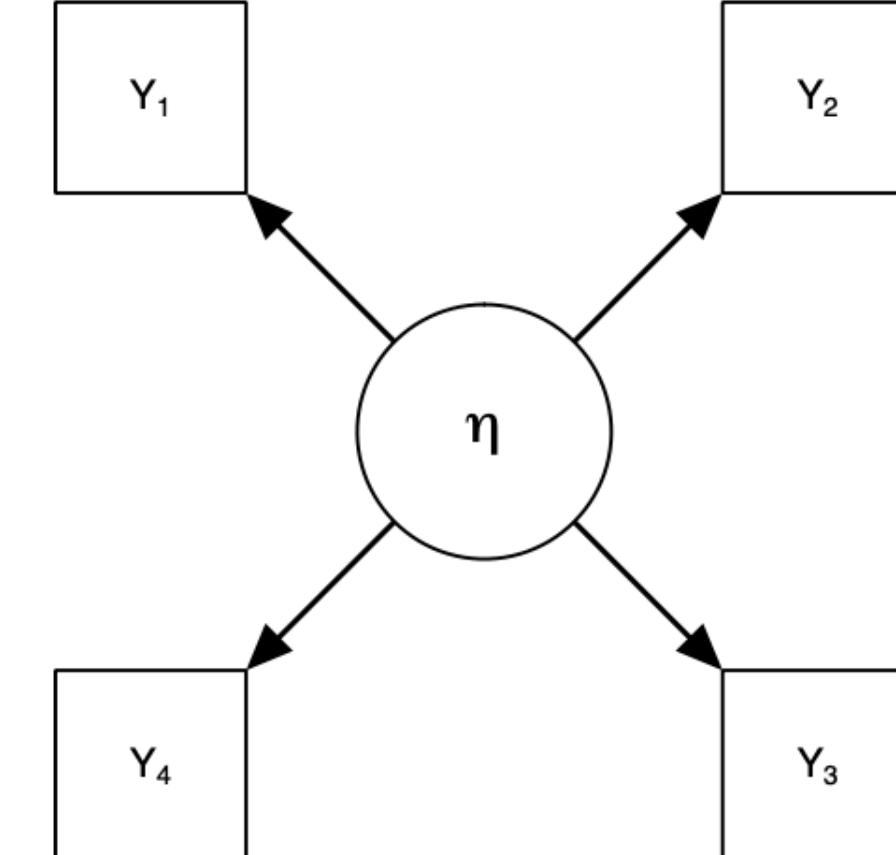
Data generating structure



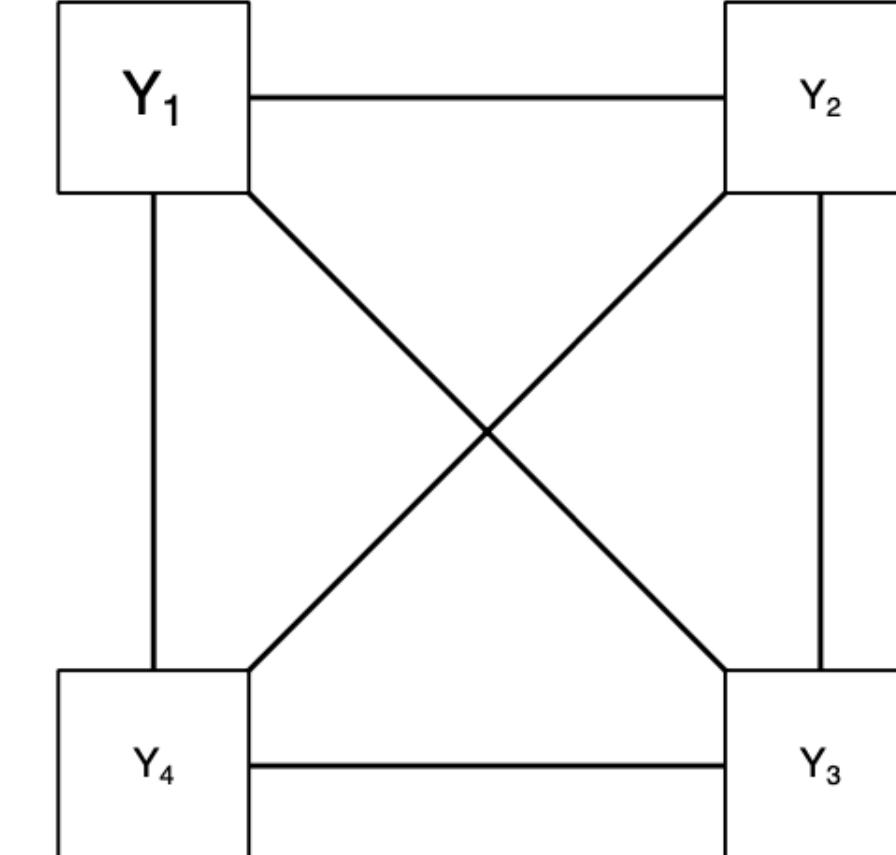
Markov Random Field



Data generating structure



Markov Random Field



- If we could condition on η

- If we could *not* condition on η

Pairwise Markov Random Fields

5. Exploratory tool in factor analysis

Data generating structure

PLOS ONE

RESEARCH ARTICLE

Exploratory graph analysis: A new approach for estimating the number of dimensions in psychological research

Hudson F. Golino^{1,2*}, Sacha Epskamp³

¹ Department of Psychology, University of Virginia, Charlottesville, VA, United States of America,
² Graduate School of Psychology, Universidade Salgado de Oliveira, Rio de Janeiro, Brasil, ³ University of Amsterdam, Amsterdam, Netherlands

arkov Random Field

- If we could condition on η , we would have a pairwise MRF

Pairwise Markov Random Fields: recap

- A PMRF encodes the *conditional* association between two variables after conditioning on all variables in the network
- This network *structure* is inferred from the statistical distribution of the nodes in the network



Pairwise Markov Random Fields

- A PMRF encodes the *conditional* association between two variables after conditioning on all variables in the network
- This network *structure* is inferred from the statistical distribution of the nodes in the network
- State of the nodes can be either continuous (**gaussian graphical model**), binary (**Ising model**), or mixed (**mixed graphical model**)

MULTIVARIATE BEHAVIORAL RESEARCH, 2018
VOL. 53, NO. 4, 453–480
<https://doi.org/10.1080/00273171.2018.1454823>

The Gaussian Graphical Model in Cross-Sectional and Time-Series Data

Sacha Epskamp^a, Lourens J. Waldorp^a, René Möttus^b, and Denny Borsboom^a

^aDepartment of Psychological Methods, University of Amsterdam; ^bDepartment of Psychology, University of Edinburgh



OPEN A new method for constructing networks from binary data

Claudia D. van Borkulo^{1,2}, Denny Borsboom², Sacha Epskamp², Tessa F. Blanken², Lynn Boschloo¹, Robert A. Schoevers¹ & Lourens J. Waldorp²

¹Interdisciplinary Center Psychopathology and Emotion regulation, University Medical Center Groningen, University of Groningen,
²Department of Psychology, Psychological Methods, University of Amsterdam.



Journal of Statistical Software

April 2020, Volume 93, Issue 8.

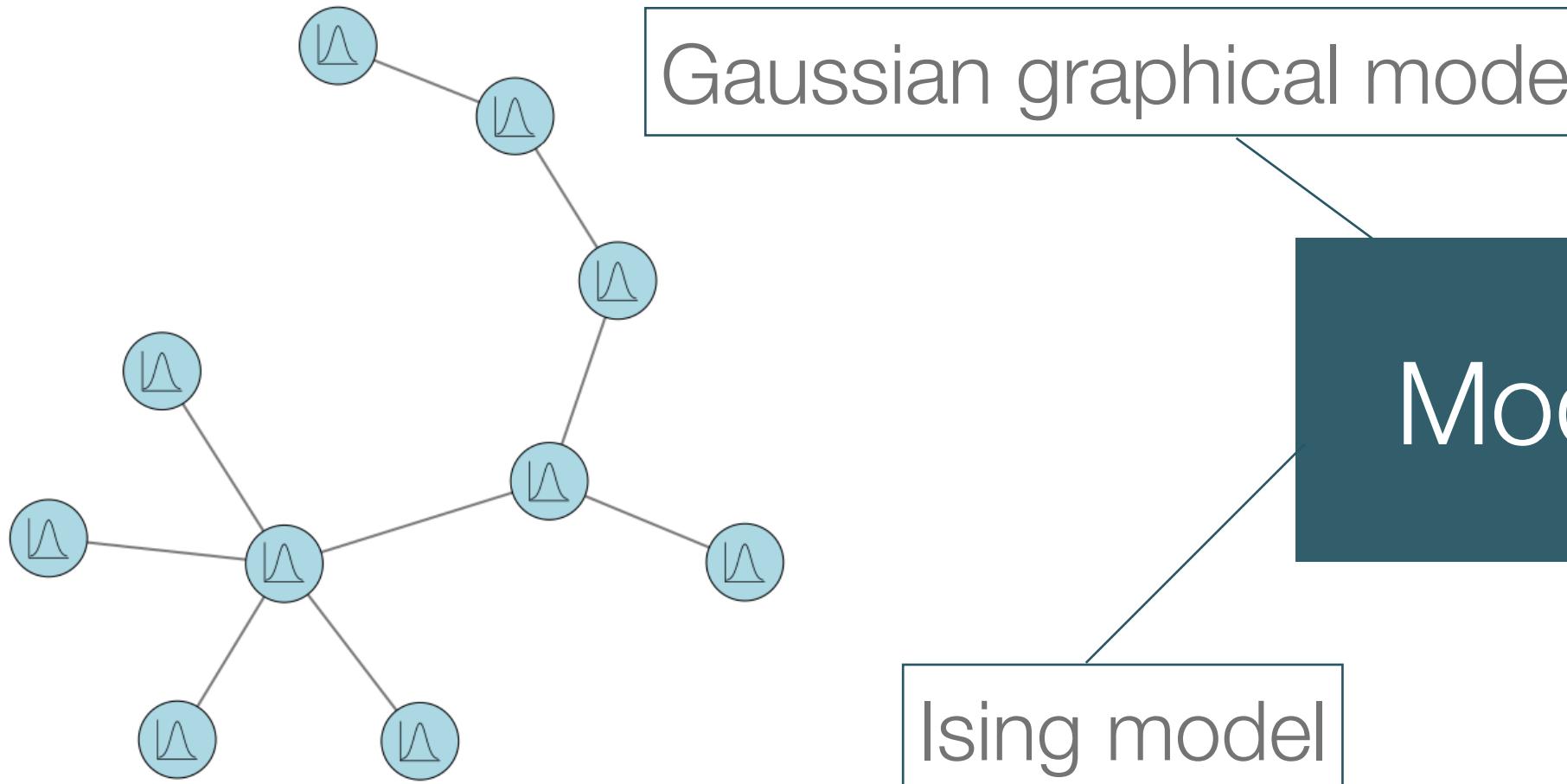
[doi: 10.18637/jss.v093.i08](https://doi.org/10.18637/jss.v093.i08)

mgm: Estimating Time-Varying Mixed Graphical Models in High-Dimensional Data

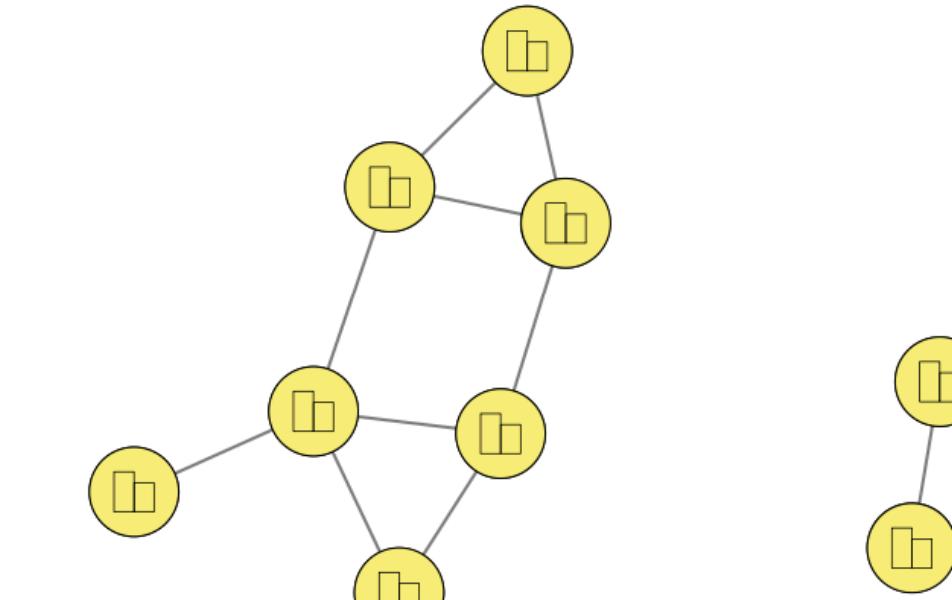
Jonas M. B. Haslbeck
University of Amsterdam

Lourens J. Waldorp
University of Amsterdam

Models



Ising model



Mixed graphical model

Estimation

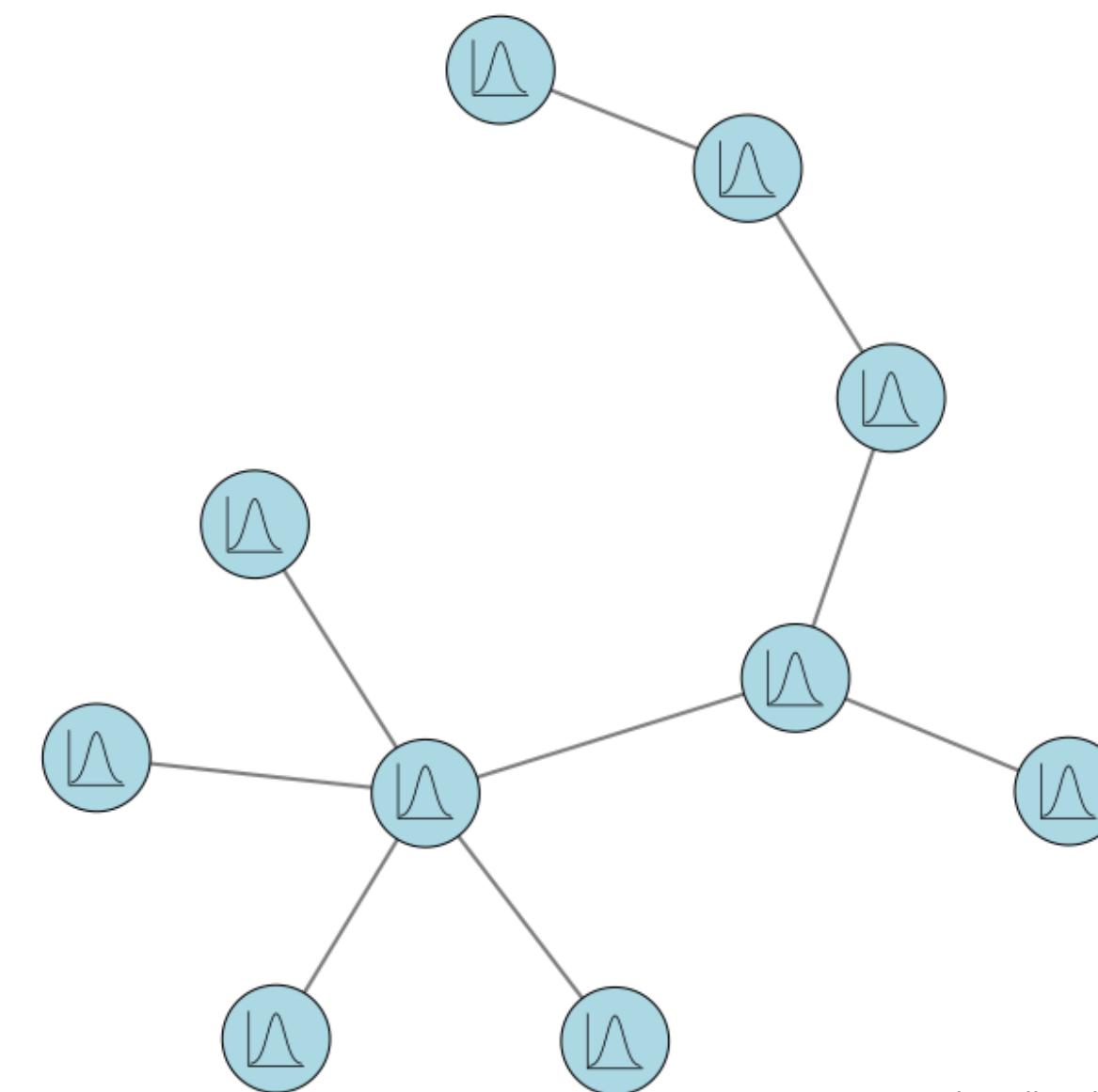
Multivariate

Univariate (nodewise)

Selection

Gaussian graphical model

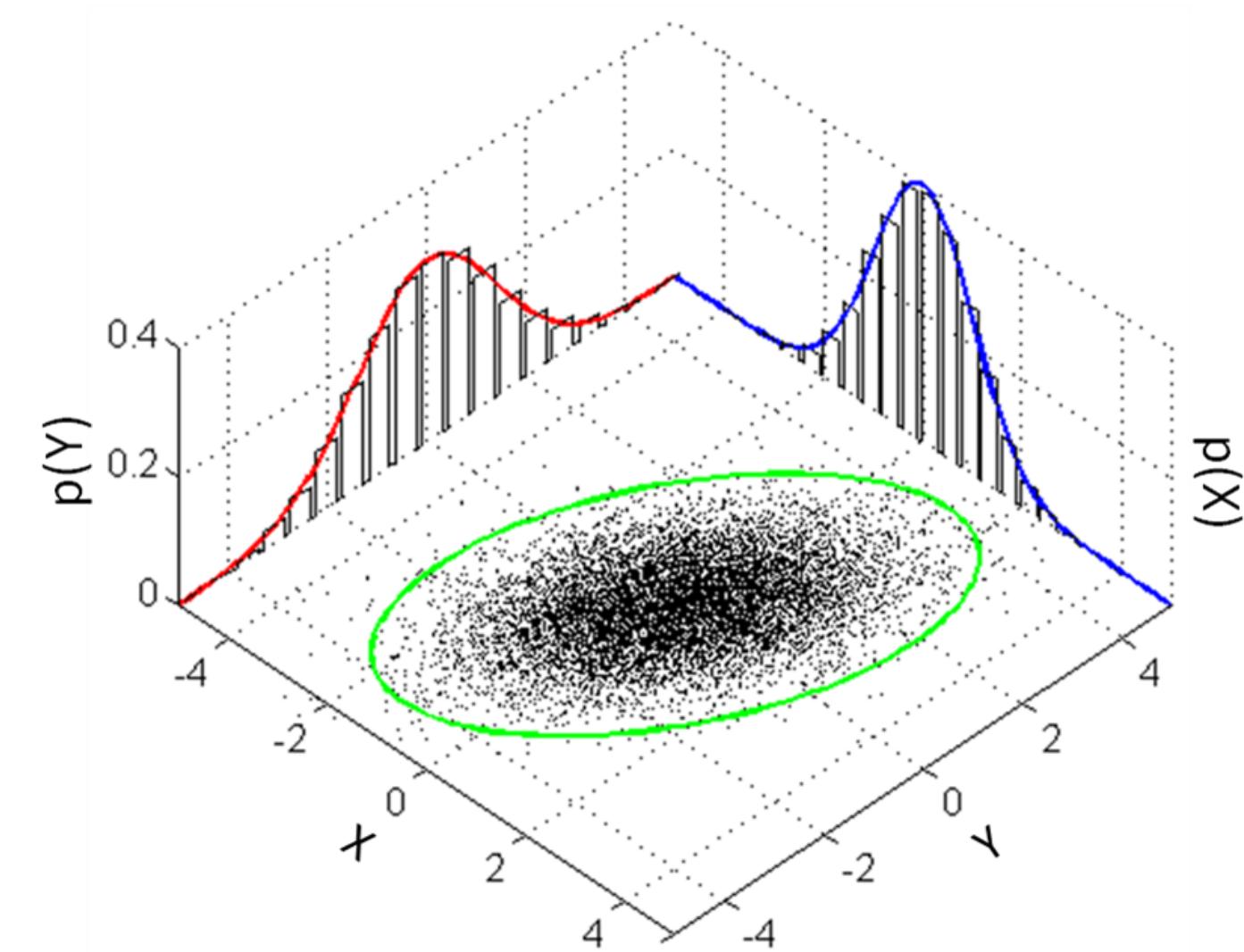
When all variables are gaussian



visualisation by
Jonas Haslbeck

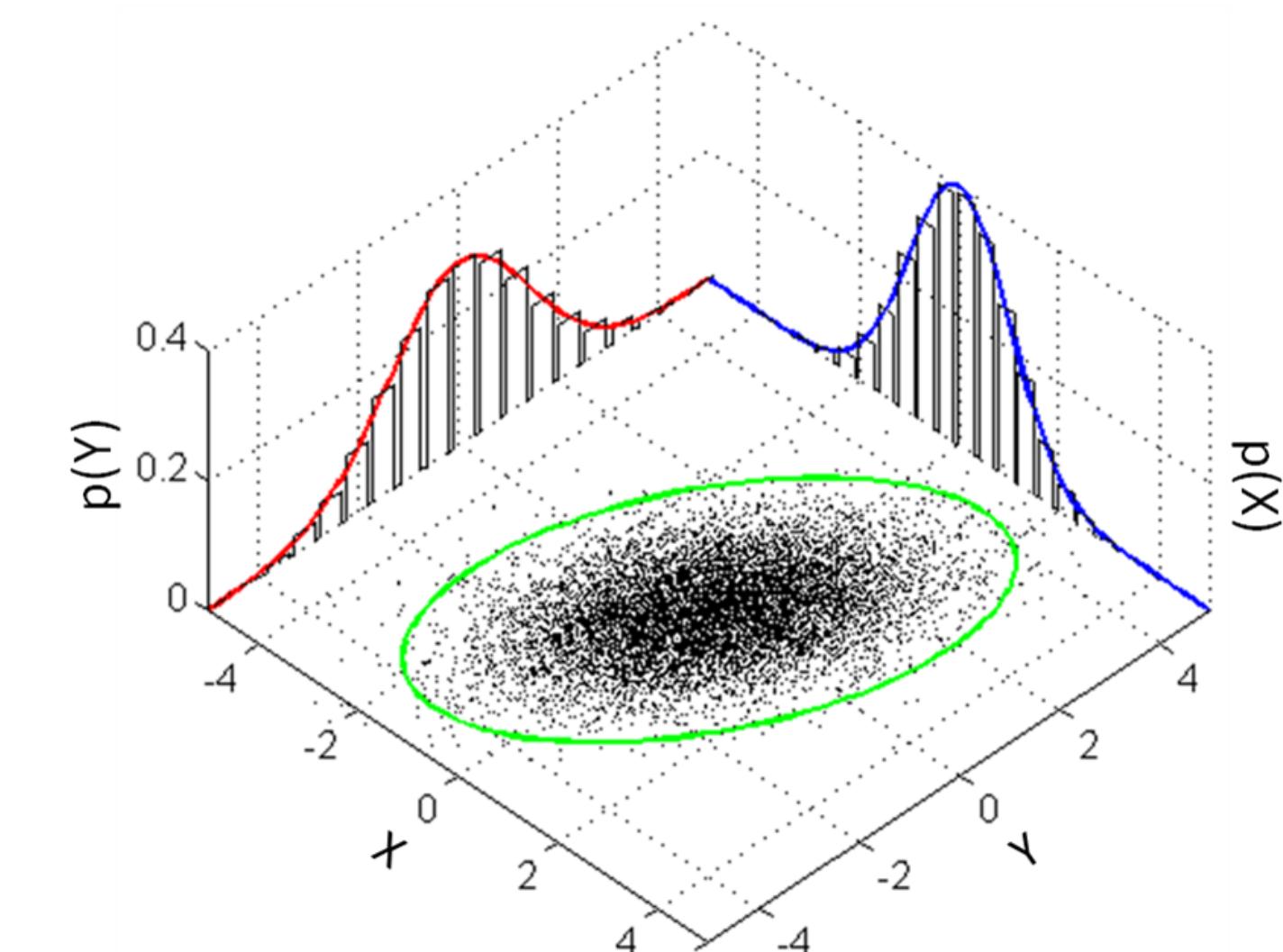
The Gaussian Graphical Model (GGM)

- A GGM can be estimated in two ways:
 1. Inverting the sample covariance matrix
(i.e., *multivariate estimation*)
 2. Performing multiple regressions
(i.e., *univariate estimation*)



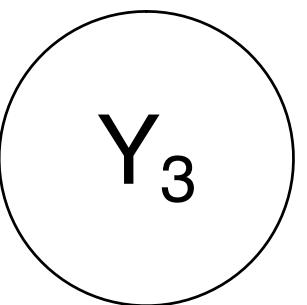
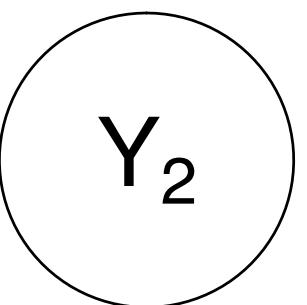
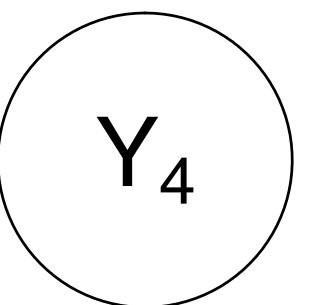
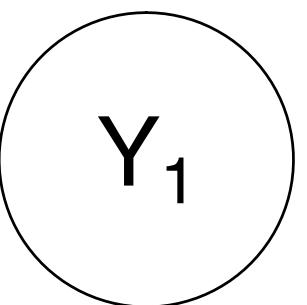
The Gaussian Graphical Model (GGM)

- A GGM can be estimated in two ways:
 1. Inverting the sample covariance matrix (i.e., *multivariate*)
 2. Performing multiple regressions (i.e., *univariate*)
- We must assume multivariate normality because in this case, the variance-covariance matrix Σ encodes all variation on how variables relate to one-another
- The inverse of the variance-covariance matrix Σ is called the precision matrix \mathbf{K} , which encodes *all* conditional relationships
 - $K = \Sigma^{-1}$

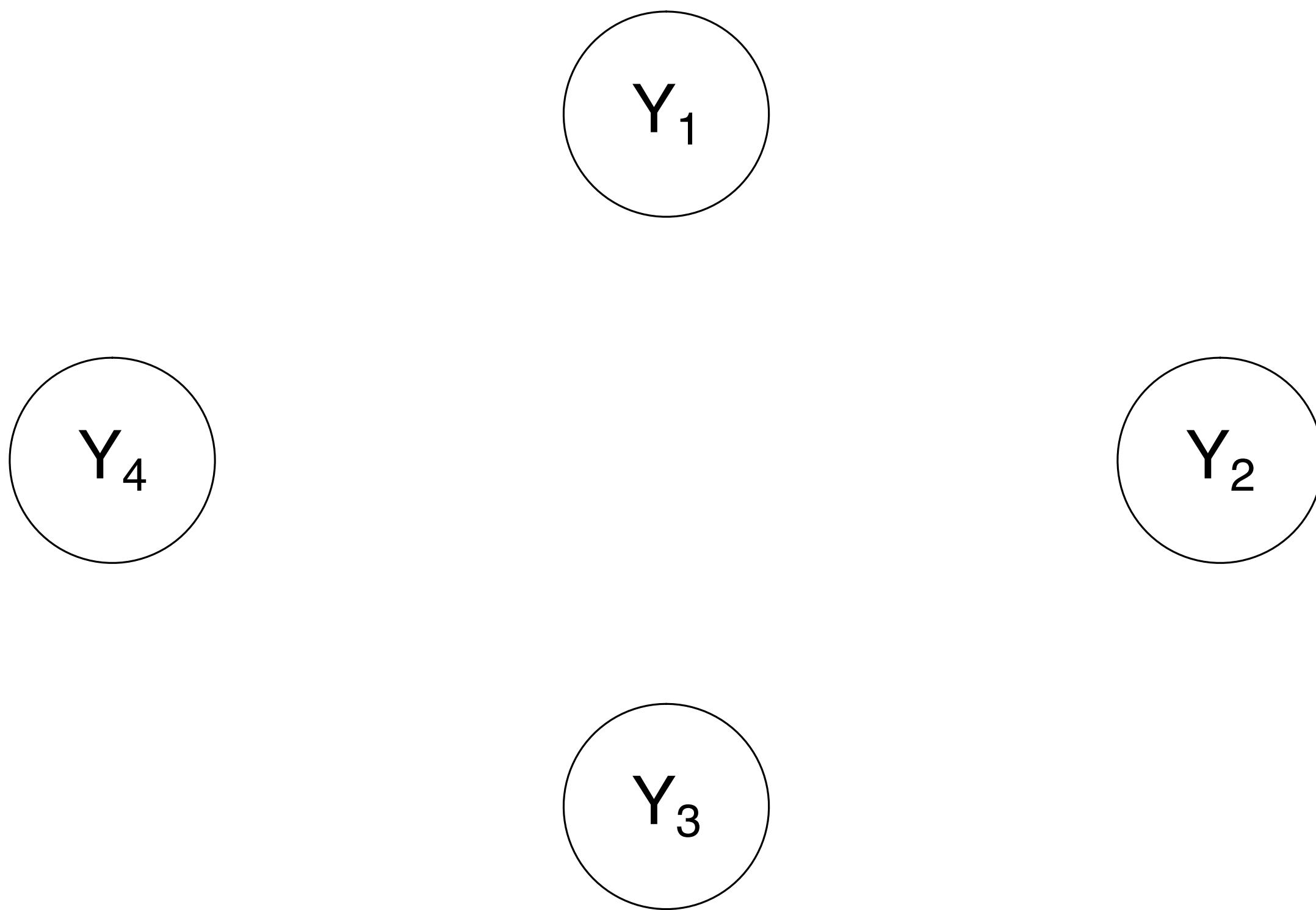


The Gaussian Graphical Model (GGM)

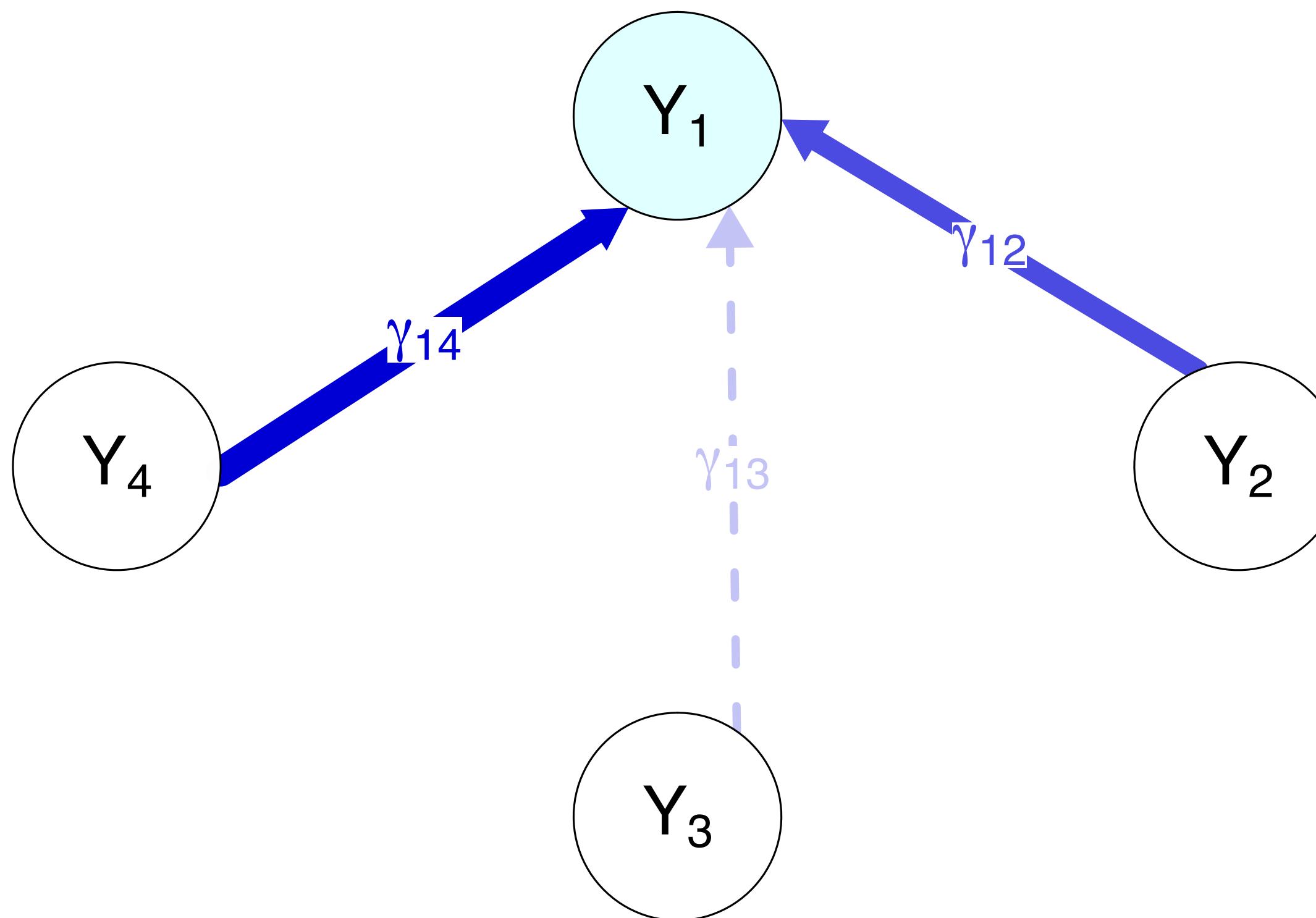
- A GGM can be estimated in two ways:
 1. Inverting the sample covariance matrix
(i.e., multivariate)
 2. Performing multiple regressions
(i.e., *univariate*)



GGM and multiple regressions

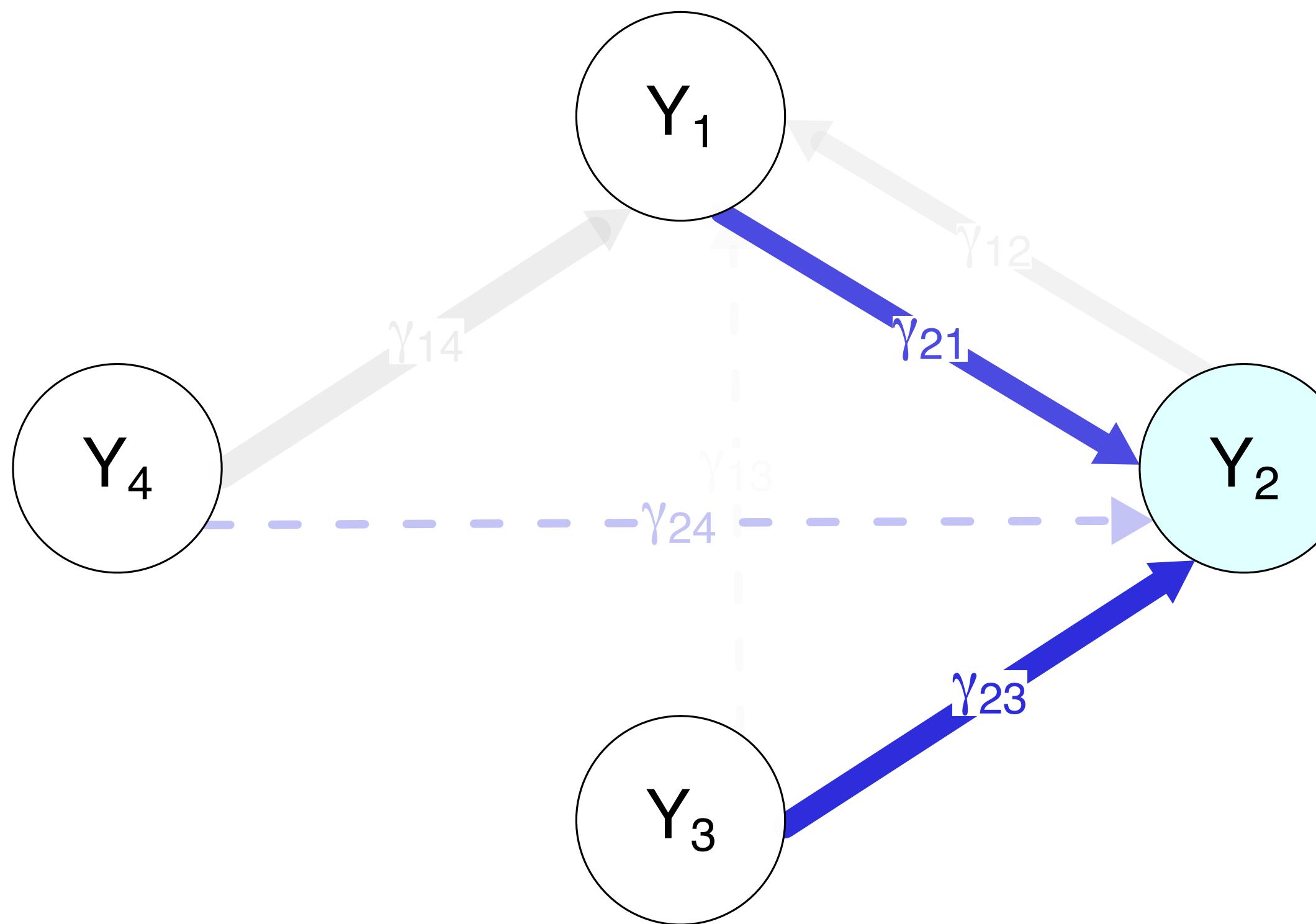


GGM and multiple regressions



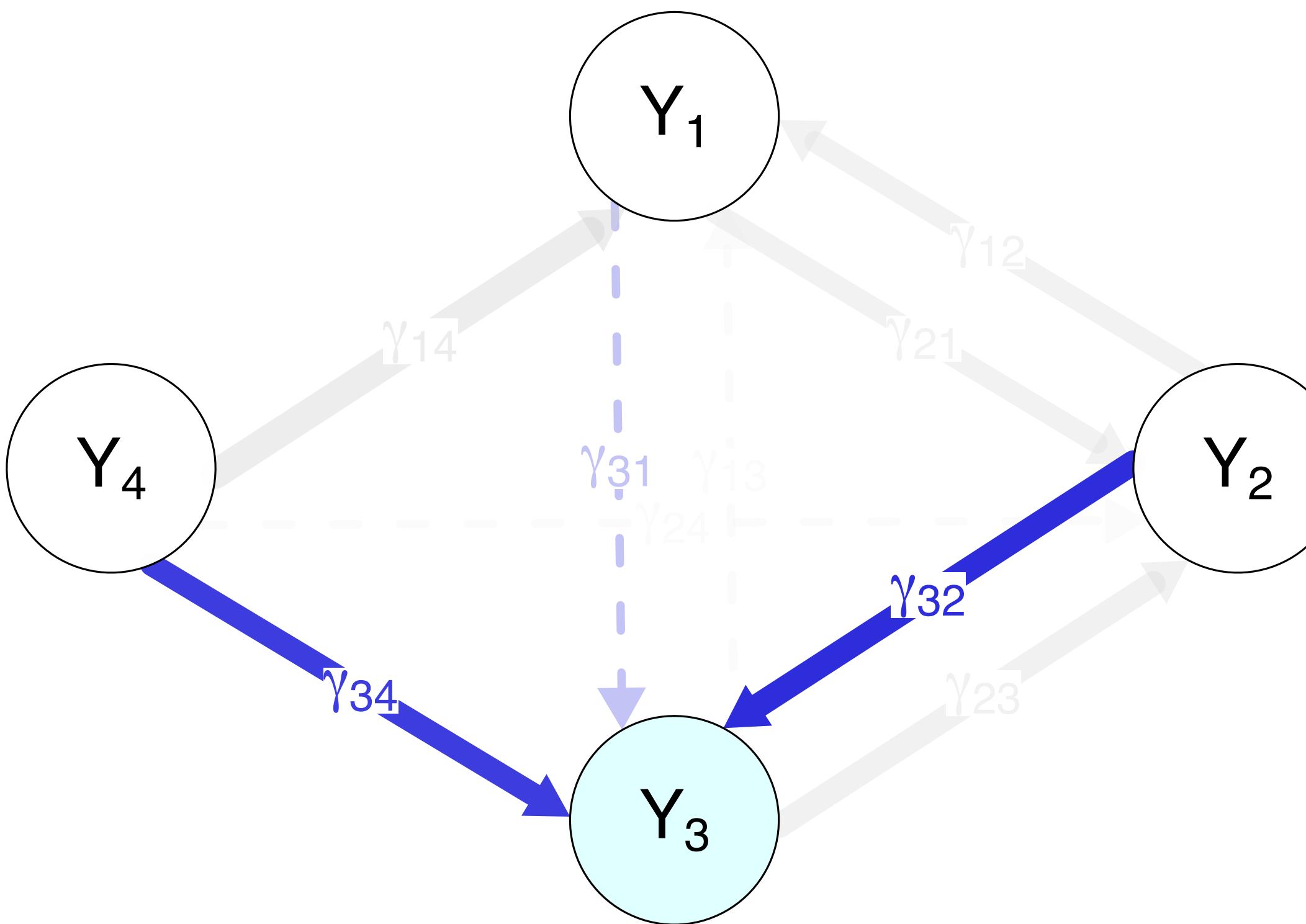
$$y_1 = \tau_1 + \gamma_{12}y_2 + \gamma_{13}y_3 + \gamma_{14}y_4 + \epsilon_1$$

GGM and multiple regressions



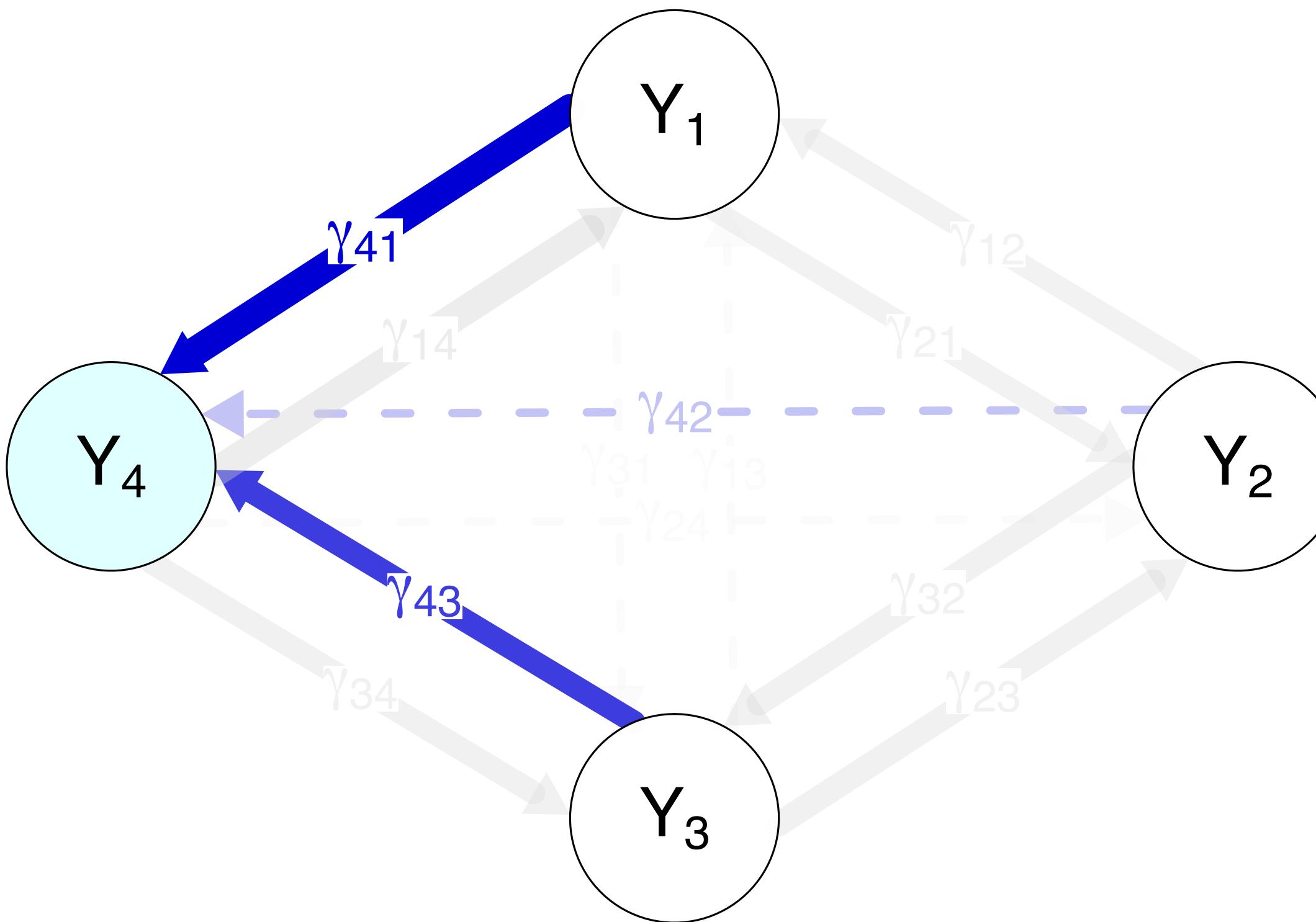
$$y_2 = \tau_2 + \gamma_{21}y_1 + \gamma_{23}y_3 + \gamma_{24}y_4 + \epsilon_2$$

GGM and multiple regressions



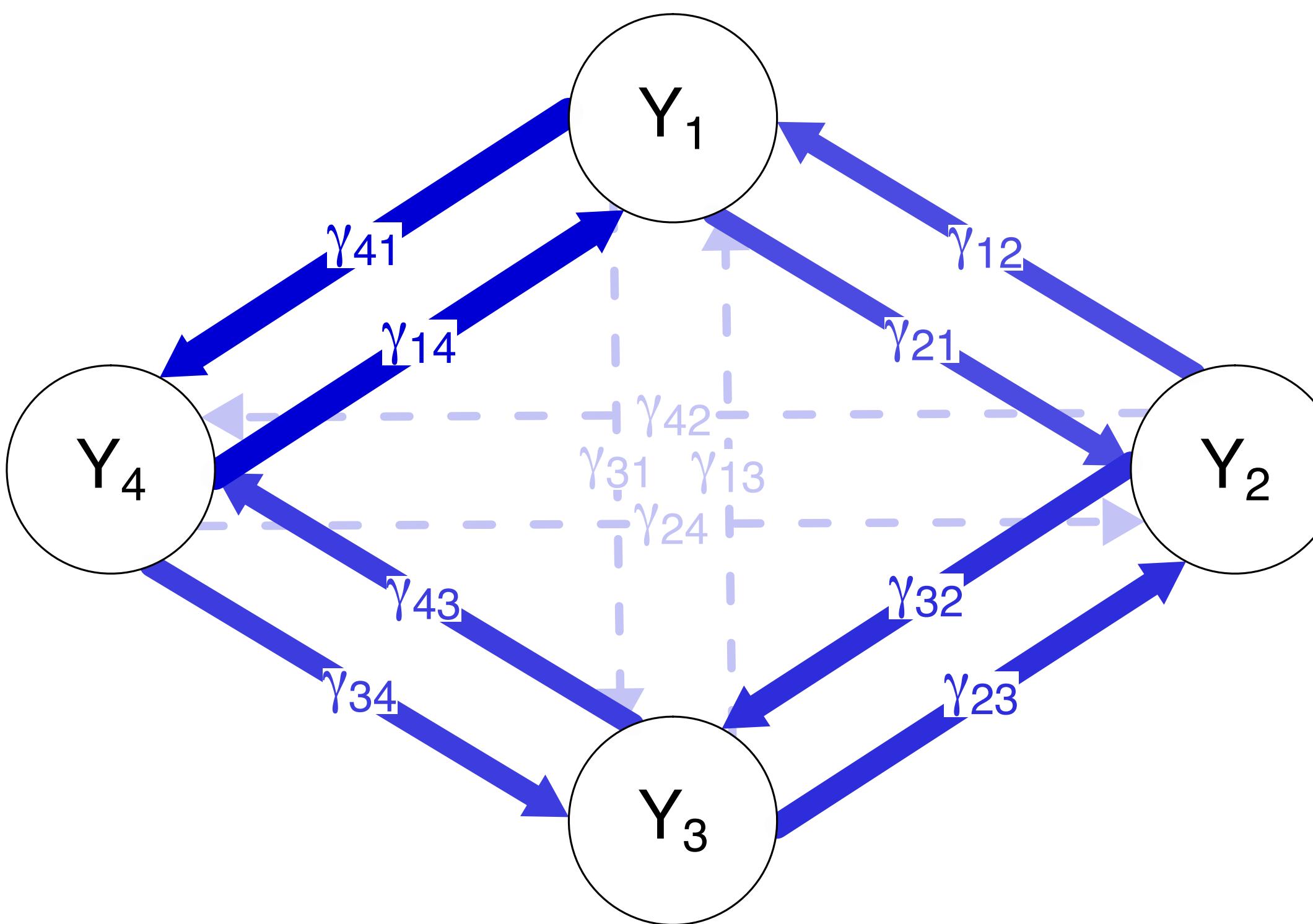
$$y_3 = \tau_3 + \gamma_{31}y_1 + \gamma_{32}y_2 + \gamma_{34}y_4 + \epsilon_3$$

GGM and multiple regressions

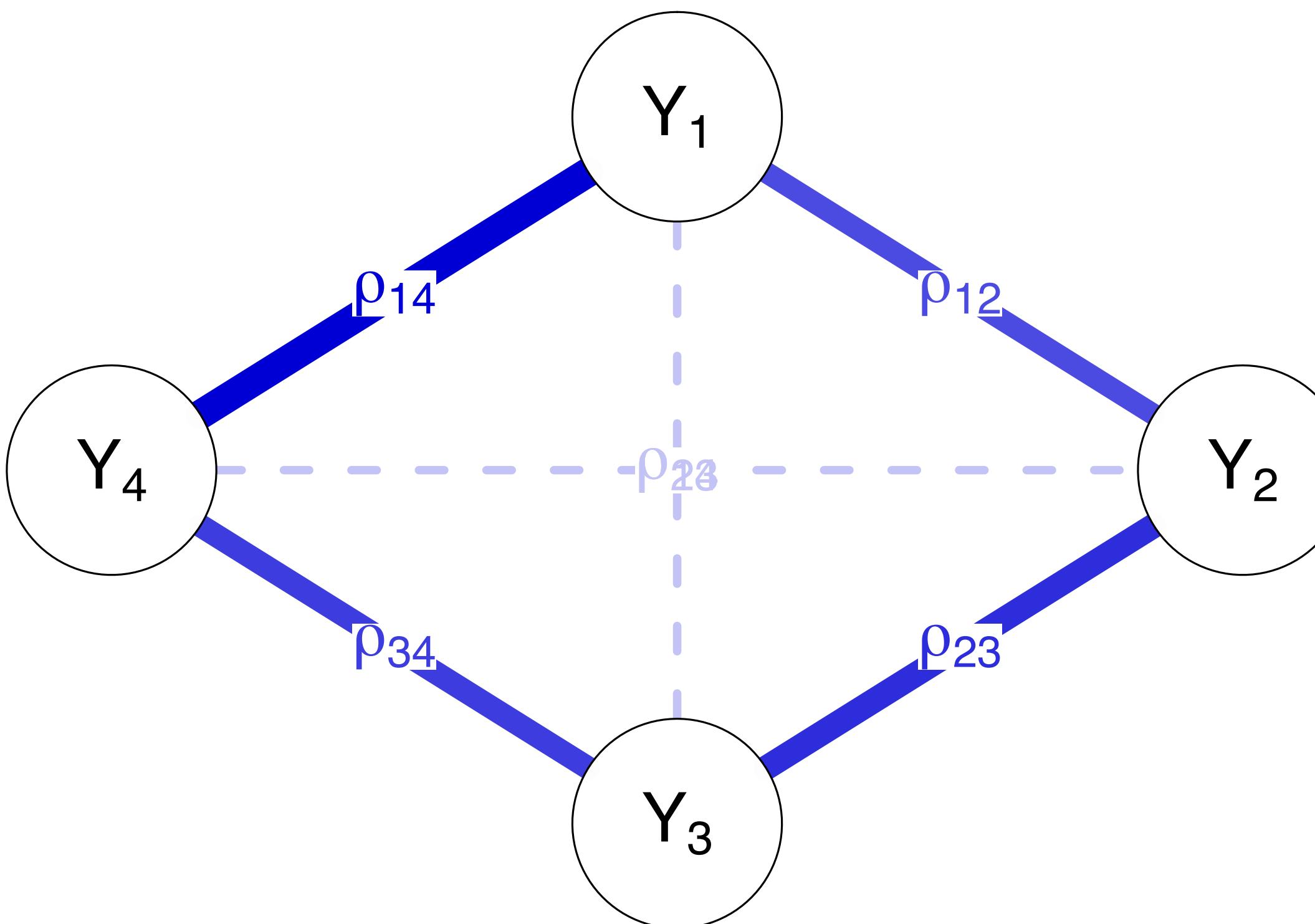


$$y_4 = \tau_4 + \gamma_{41}y_1 + \gamma_{42}y_2 + \gamma_{43}y_3 + \epsilon_4$$

GGM and multiple regressions



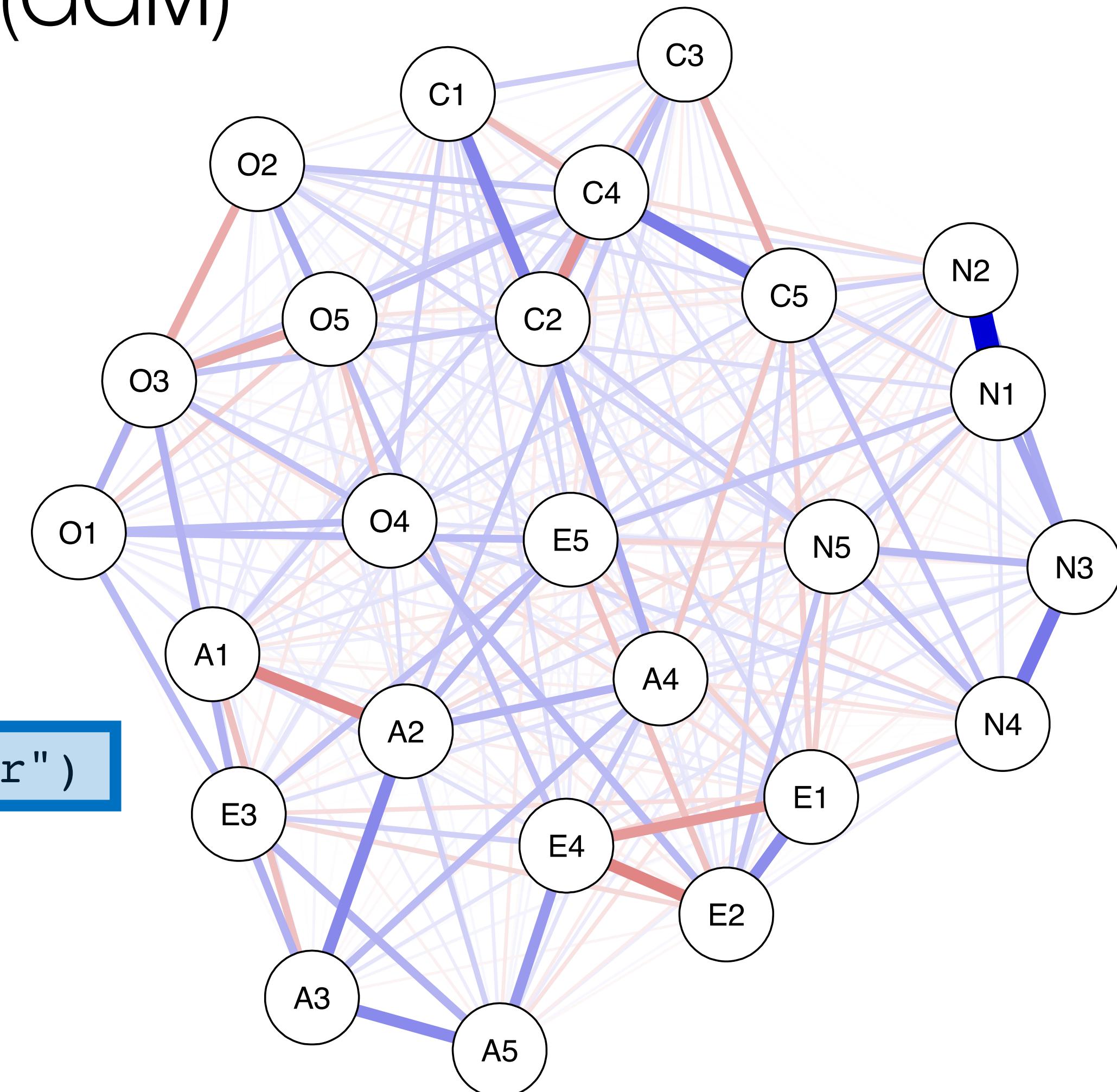
GGM and multiple regressions



$$\rho_{ij} = \gamma_{ij} \times \frac{SD(\epsilon_j)}{SD(\epsilon_i)} = \gamma_{ji} \times \frac{SD(\epsilon_i)}{SD(\epsilon_j)} = - \frac{\kappa_{ij}}{\sqrt{(\kappa_{ii})}\sqrt{(\kappa_{jj})}}$$

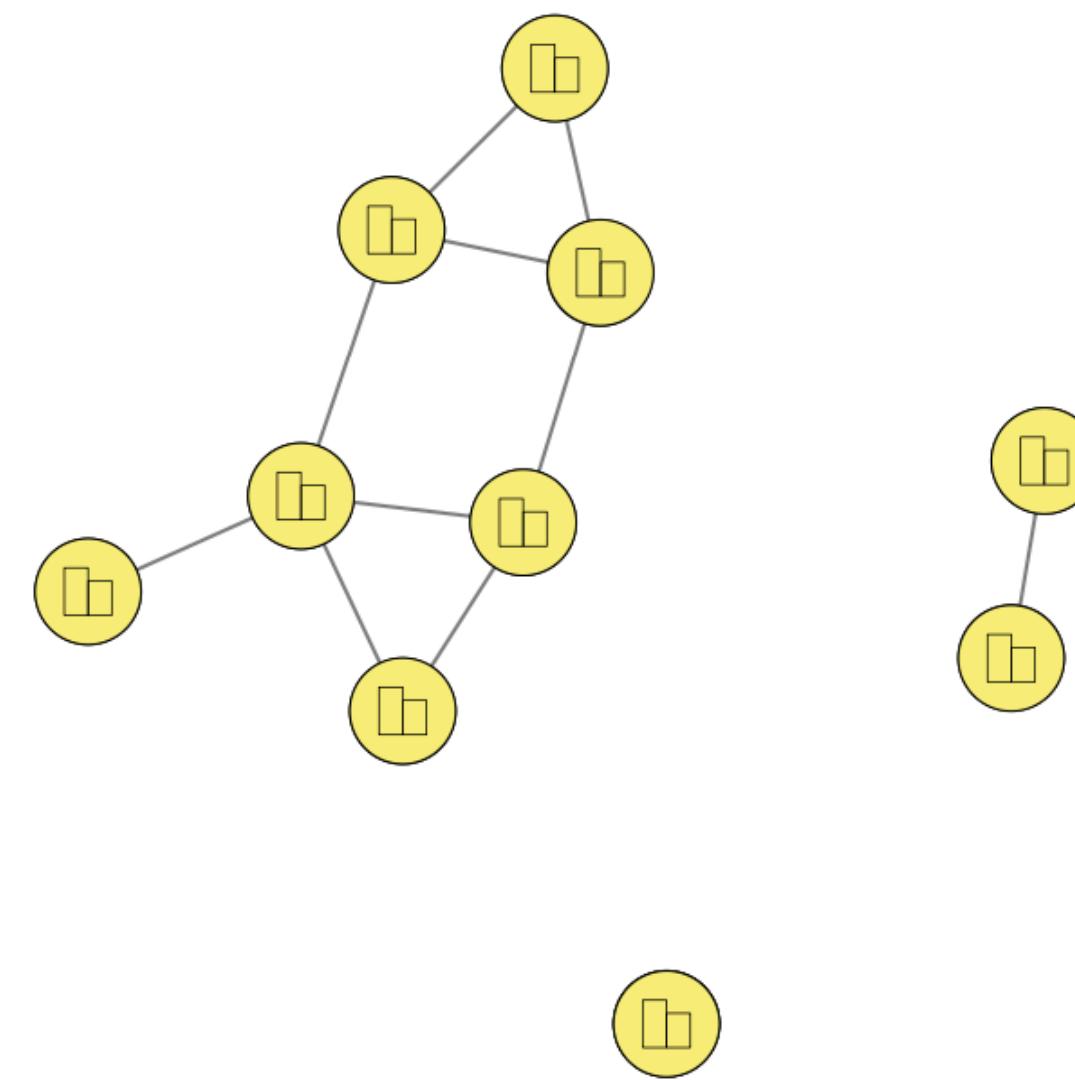
Estimating a partial correlation network (GGM)

```
# Load packages:  
library("bootnet")  
library("psychTools")  
  
# Load data:  
data("bfi")  
  
# First 25 items:  
bfiData <- bfi[,1:25]  
  
# Estimate partial correlation network:  
net_pcor <- estimateNetwork(bfiData, default = "pcor")  
  
# Plot network (saving to PDF):  
plot(net_pcor, filetype = "pdf")
```



Ising model

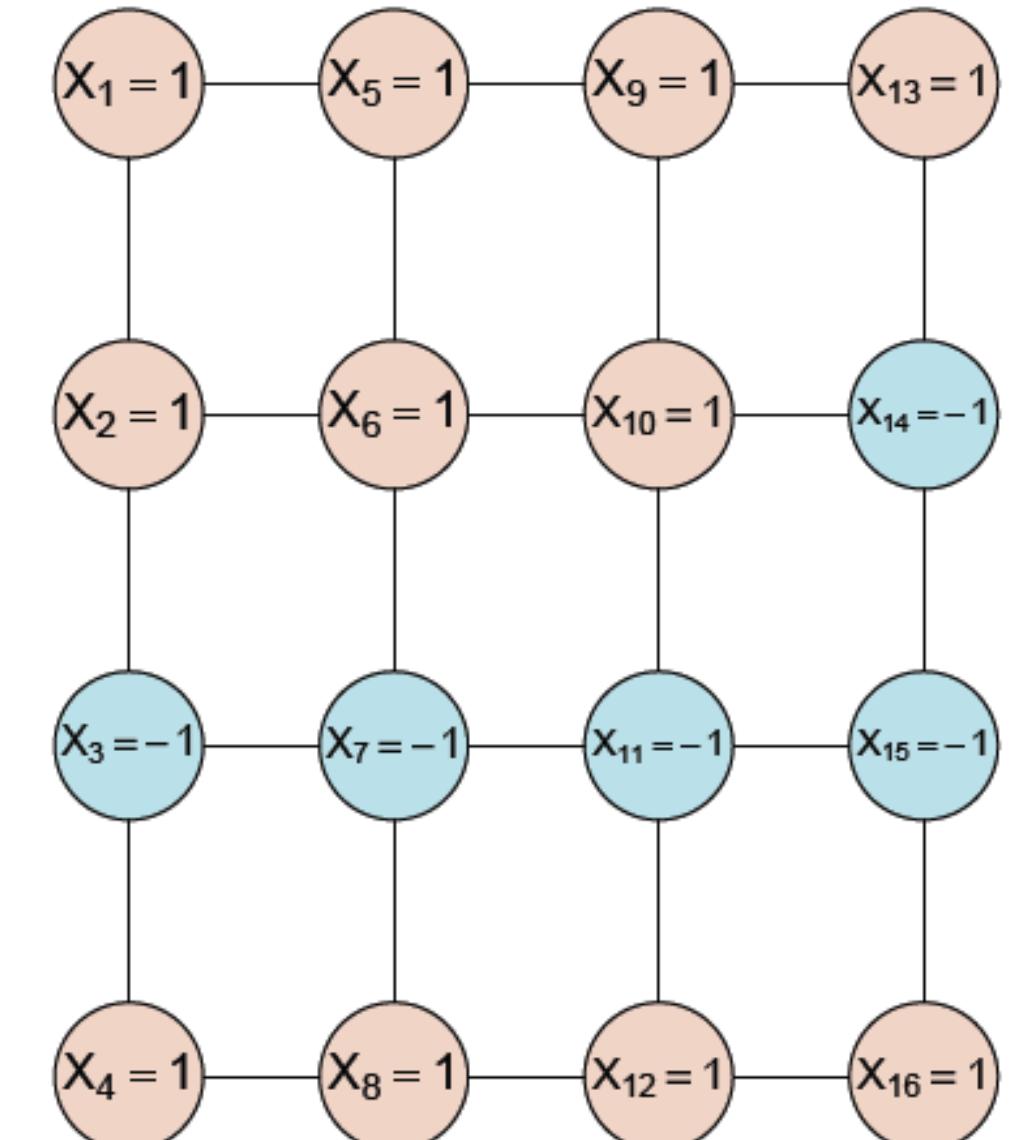
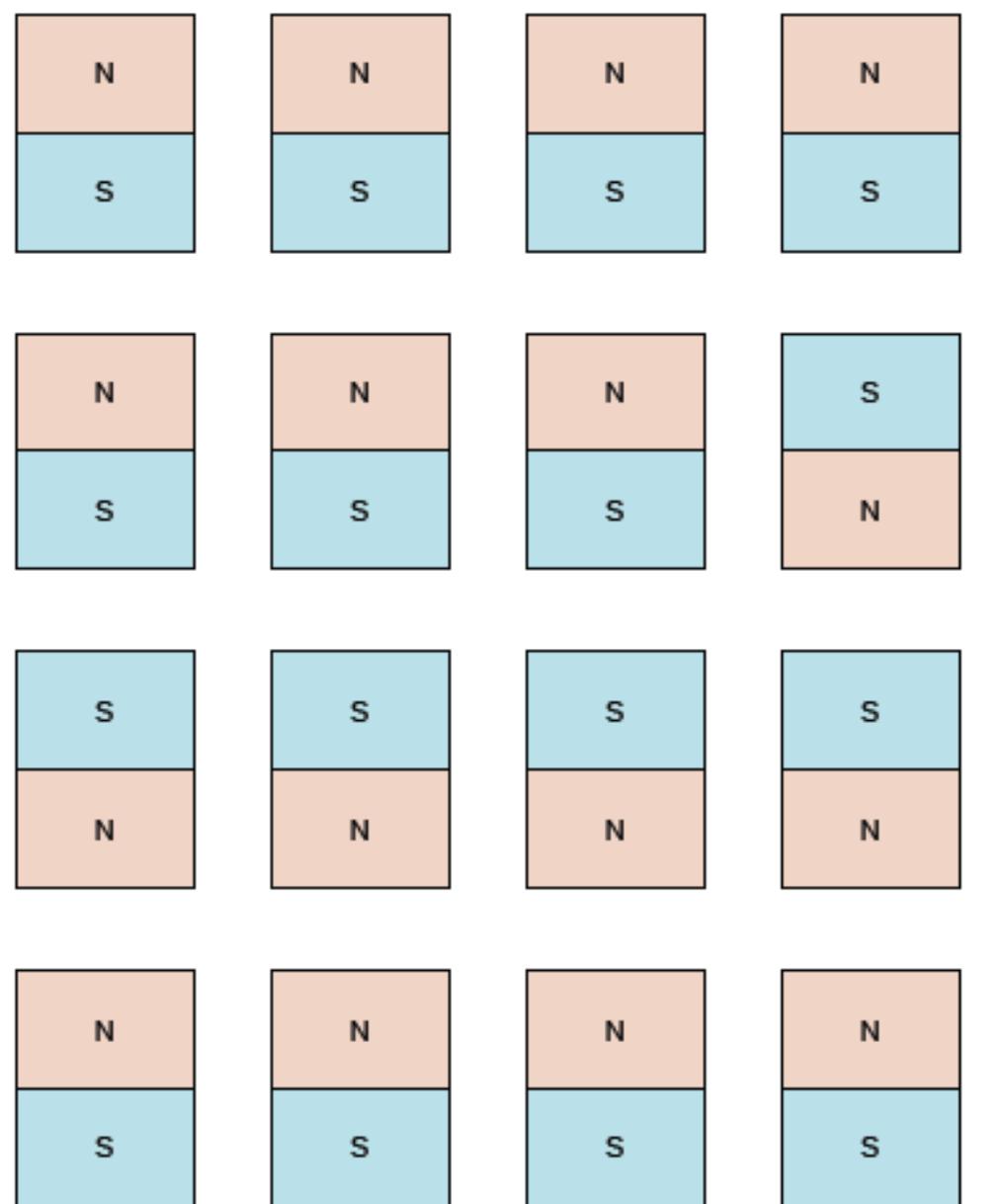
When all variables are binary



visualisation by
Jonas Haslbeck

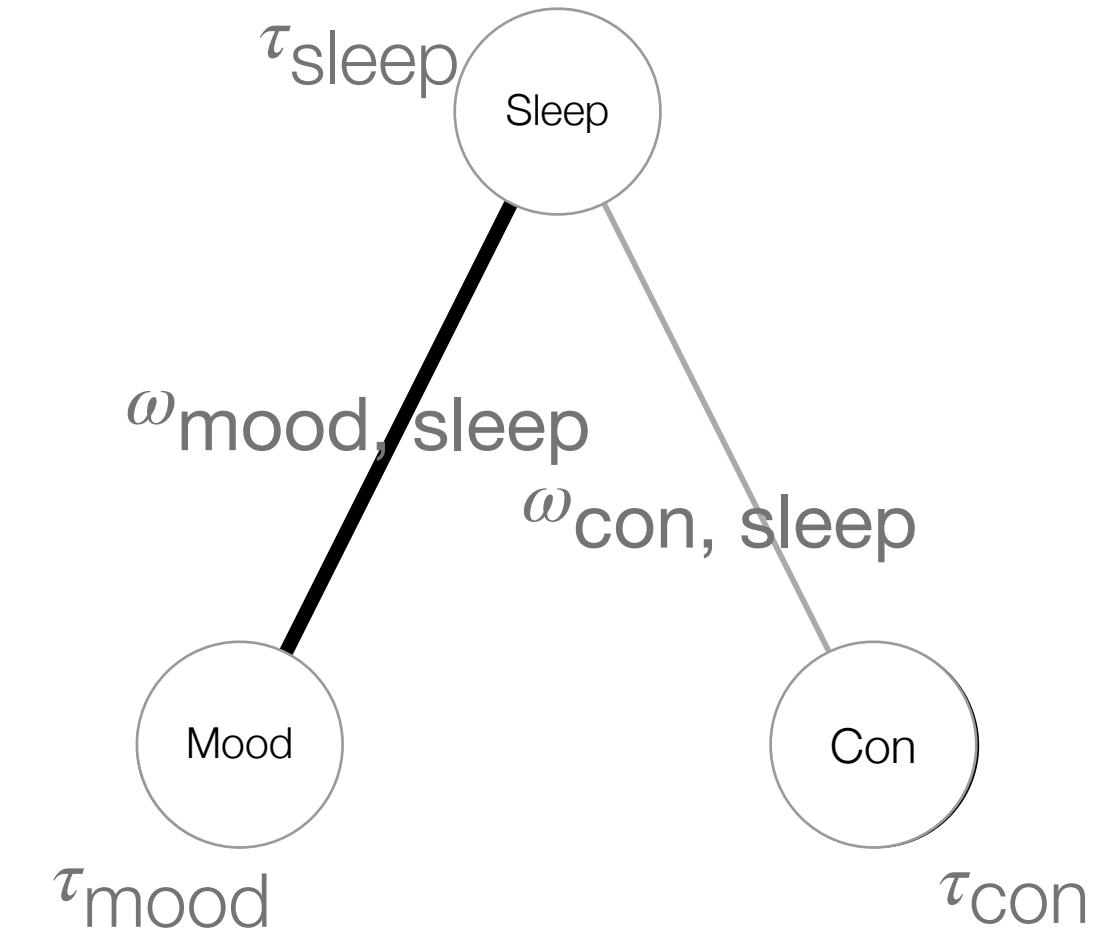
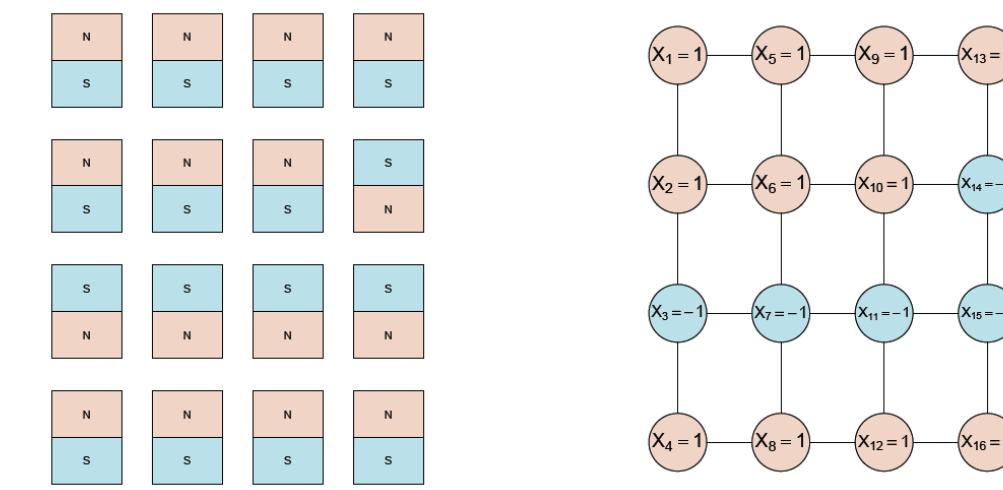
Ising model

- Originally a ferromagnetic model from physics
- Binary nodes that can be in state $\{-1, +1\}$ or $\{0, +1\}$



Ising model

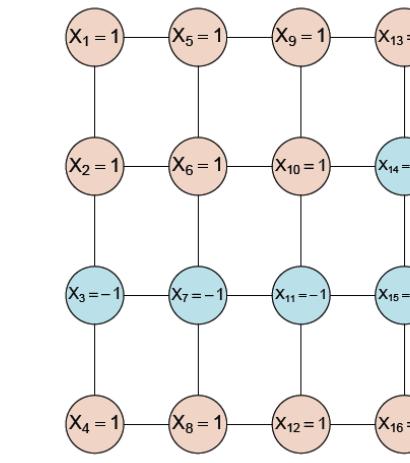
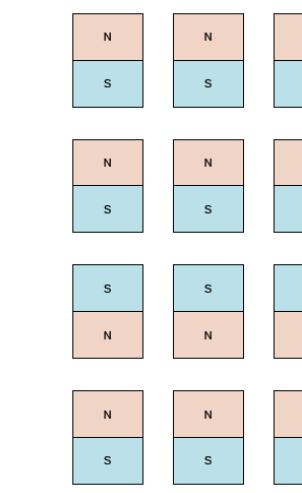
- Originally a ferromagnetic model from physics
- Binary nodes that can be in state $\{-1, +1\}$ or $\{0, +1\}$
- State of a node depends on:
 - Its **threshold** τ_i : the tendency of a node to be in one of two states
 - The state of its direct neighbours, scaled by the **network parameter** ω_{ij}



$$Pr(\text{sleep} = 1) \propto \exp \left(\tau_{\text{sleep}} + \omega_{\text{mood}, \text{sleep}} \times \text{mood} + \omega_{\text{con}, \text{sleep}} \times \text{con} \right)$$

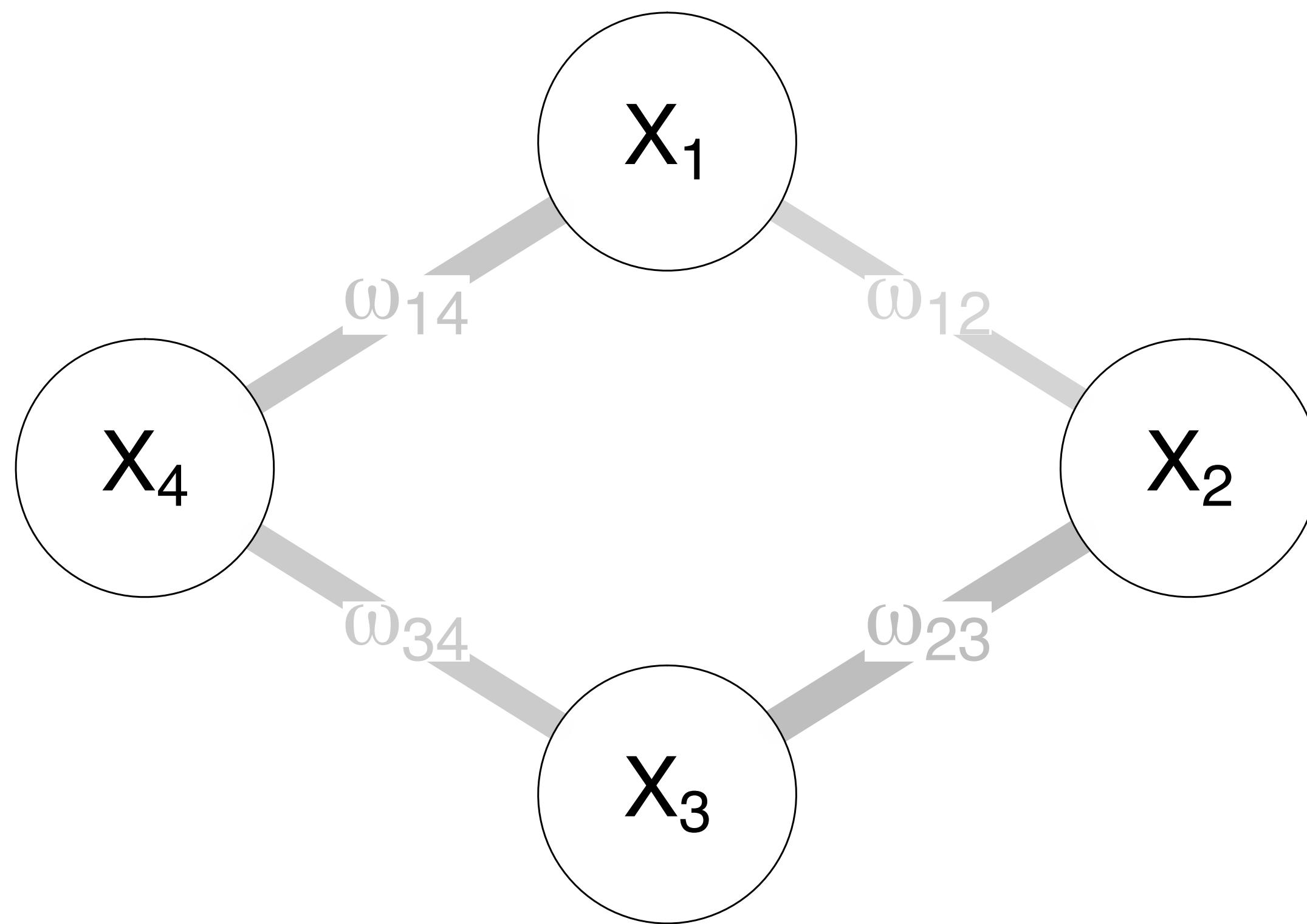
Ising model

- Originally a ferromagnetic model from physics
- Binary nodes that can be in state $\{-1, +1\}$ or $\{0, +1\}$
- State of a node depends on:
 - Its **threshold** τ_i : the tendency of a node to be in one of two states
 - The state of its direct neighbours, scaled by the **network parameter** ω_{ij}
- Joint estimation in large networks (> 20 nodes) is difficult because of the partition function, but model can be estimated through series of logistic regressions.

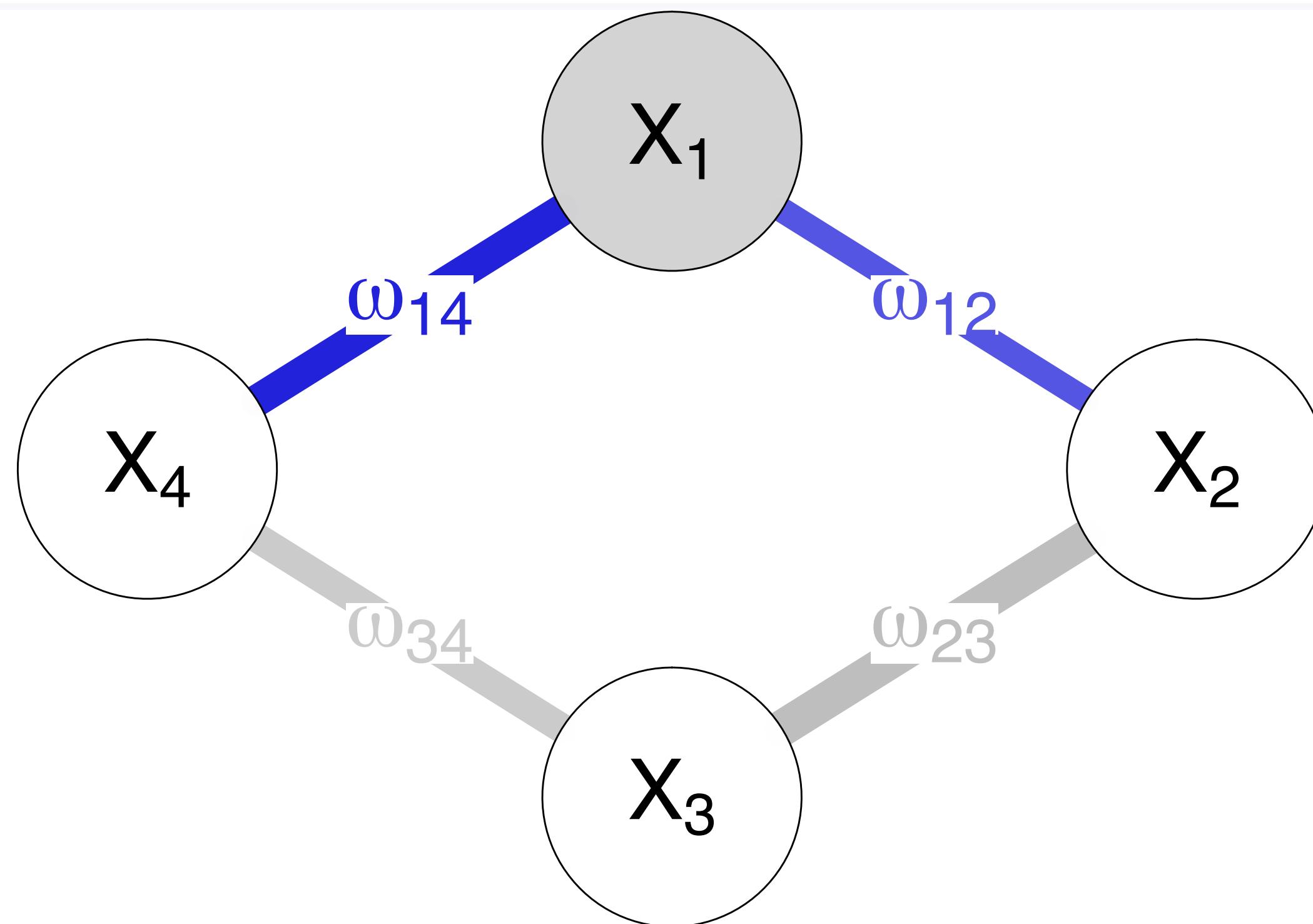


- $Pr(Y = y) = \frac{1}{Z} \exp \left(\sum_i \tau_i y_i + \sum_{\langle ij \rangle} \omega_{ij} y_i y_j \right)$
 - τ_i : **threshold** parameter
 - ω_{ij} : **network** parameter indicating the preference of node i and j to be in the same state (edge weights)
 - Z : **normalising constant** (partition function) and takes the sum over all possible configurations of Y
- $Z = \sum_y \exp \left(\sum_i \tau_i y_i + \sum_{\langle ij \rangle} \omega_{ij} y_i y_j \right)$

Ising model

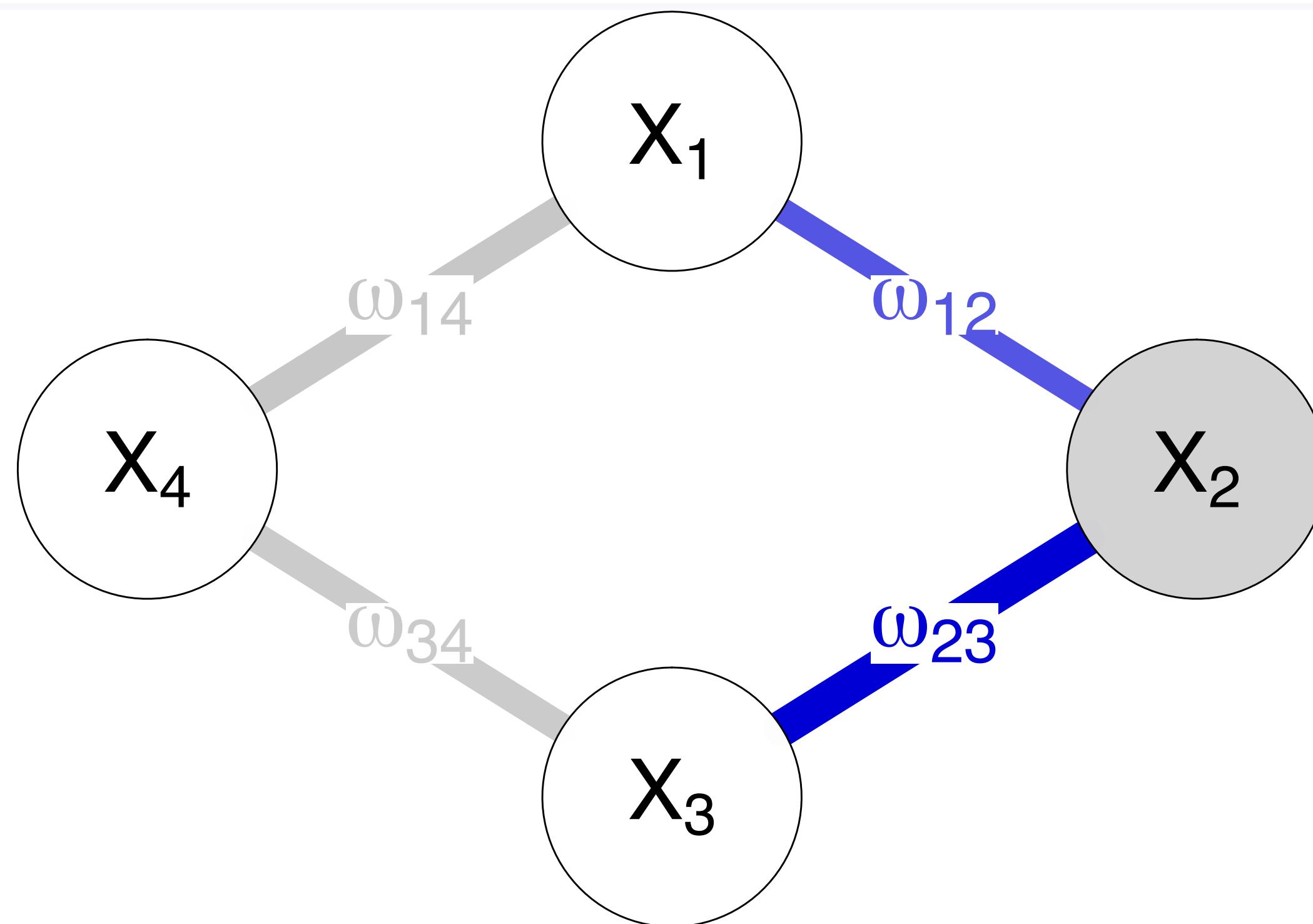


Ising model



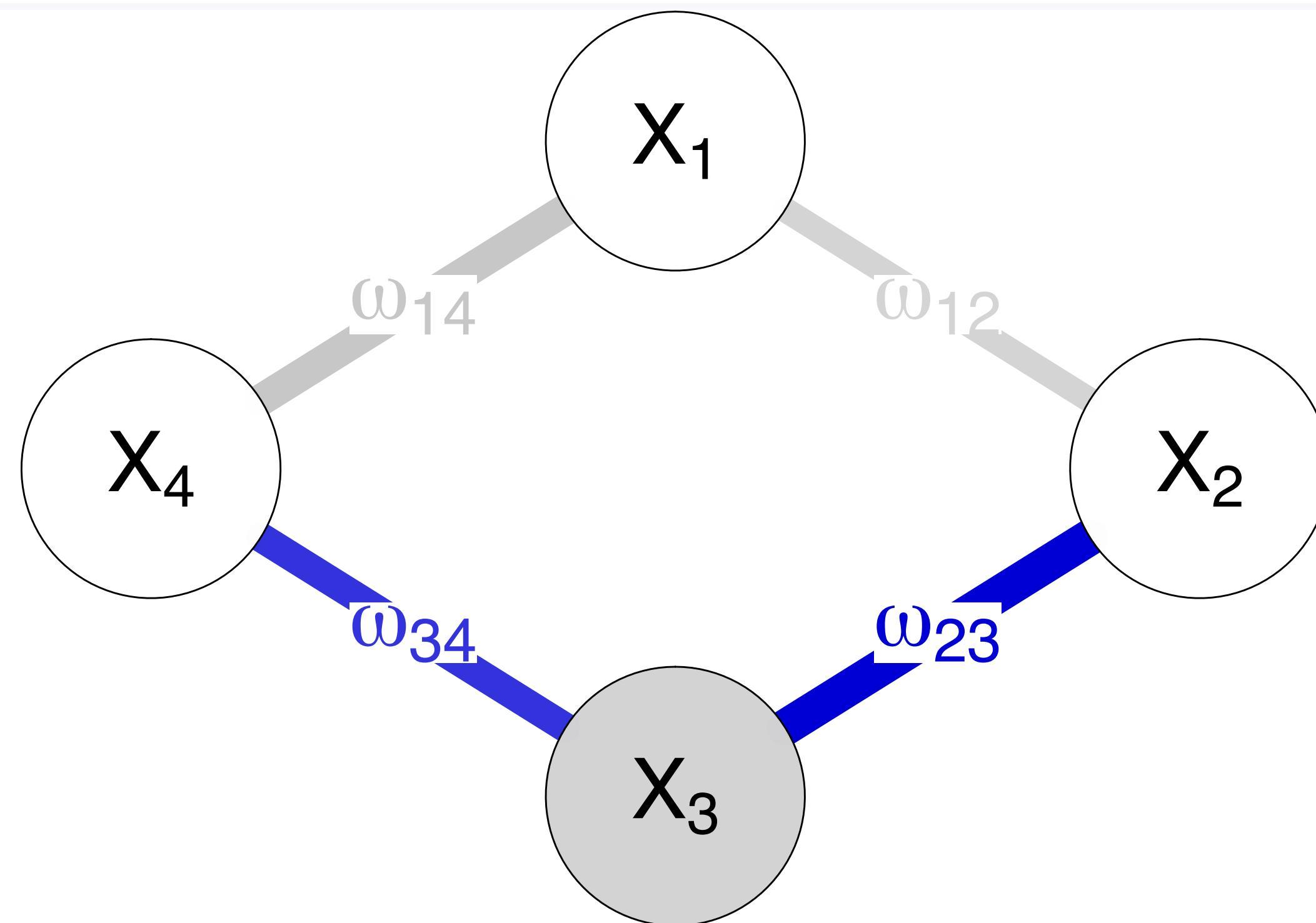
$$\Pr(X_1 = 1) \propto \exp(\tau_1 + \omega_{12}x_2 + \omega_{14}x_4)$$

Ising model



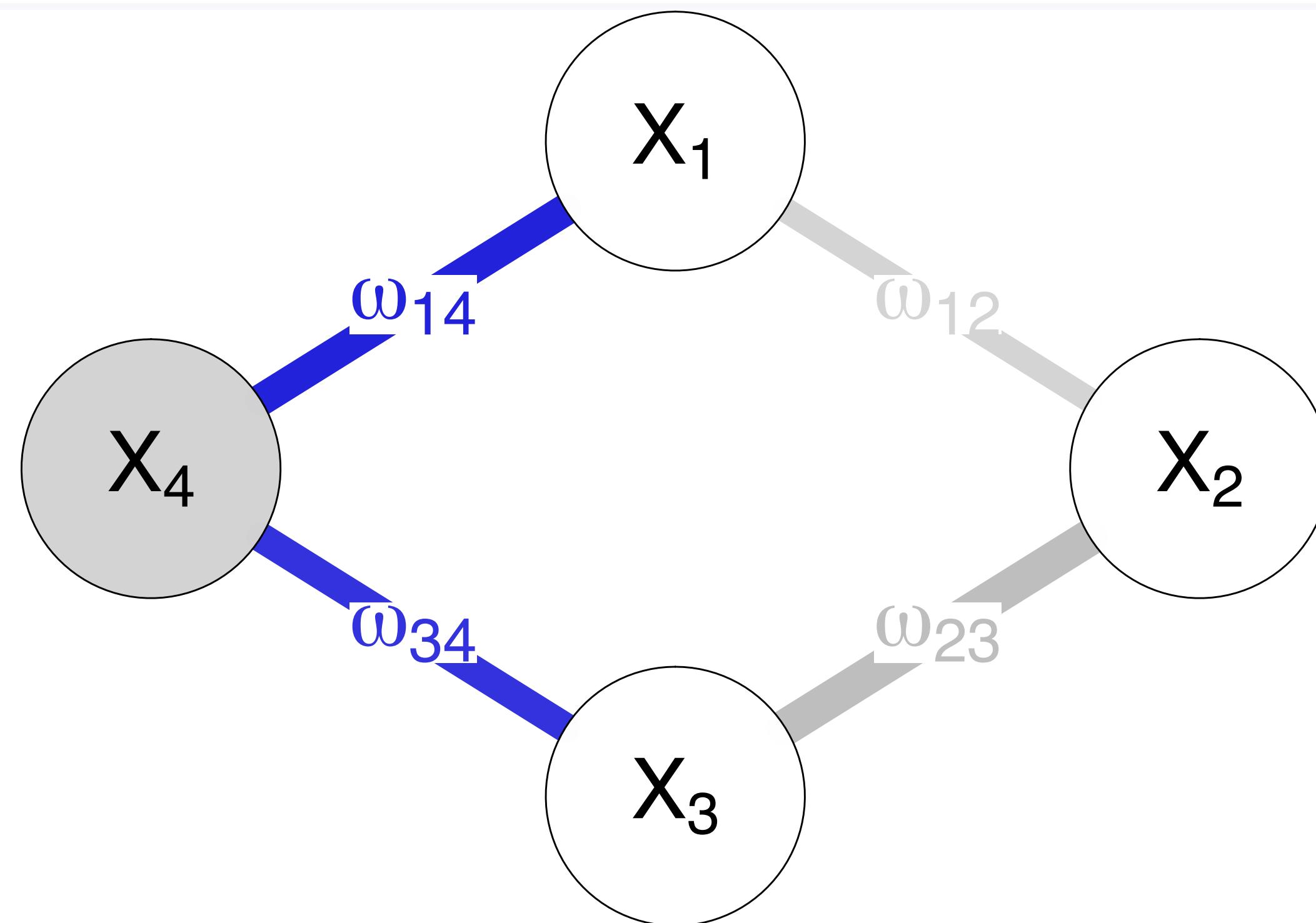
$$\Pr(X_2 = 1) \propto \exp(\tau_2 + \omega_{12}x_1 + \omega_{23}x_3)$$

Ising model



$$\Pr(X_3 = 1) \propto \exp(\tau_3 + \omega_{23}x_2 + \omega_{34}x_4)$$

Ising model

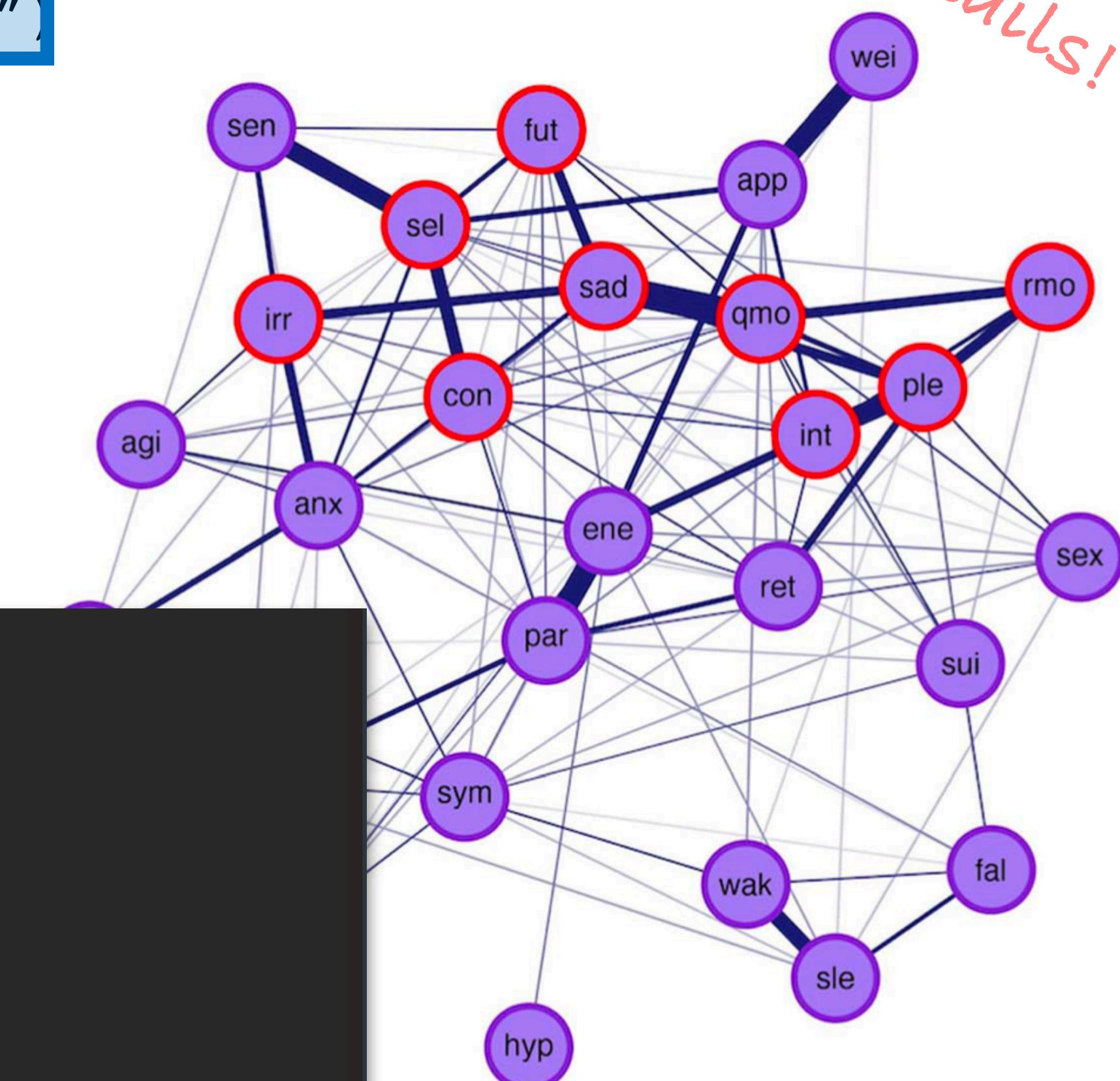


$$\Pr(X_4 = 1) \propto \exp(\tau_4 + \omega_{14}x_1 + \omega_{34}x_3)$$

Estimating an Ising model

```
# Estimate Ising model:  
net_Ising <- estimateNetwork(data, default = "IsingFit")  
  
# Plot network (saving to PDF):  
plot(net_Ising, filetype = "pdf")  
  
bootnet_IsingFit(data, tuning = 0.25, missing = c("listwise", "stop"),  
    verbose = TRUE, rule = c("AND", "OR"), split =  
    "median", principalDirection = FALSE, unlock = FALSE)  
  
> net_Ising  
  
==== Estimated network ====  
Number of nodes: 25  
Number of non-zero edges: 136 / 300  
Mean weight: 0.06740769  
Network stored in x$graph  
  
Default set used: IsingFit  
  
Use plot(x) to plot estimated network  
Use bootnet(x) to bootstrap edge weights and centrality indices  
  
Relevant references:  
  
    van Borkulo, C. D., Borsboom, D., Epskamp, S., Blanken, T. F., Boschloo, L., Schoevers, R. A., & Waldorp, L. J. (2014). A new method for constructing networks from binary data. Scientific reports, 4 (5918), 1-10.  
    Epskamp, S., Borsboom, D., & Fried, E. I. (2016). Estimating psychological networks and their accuracy: a tutorial paper. arXiv preprint, arXiv:1604.08462.
```

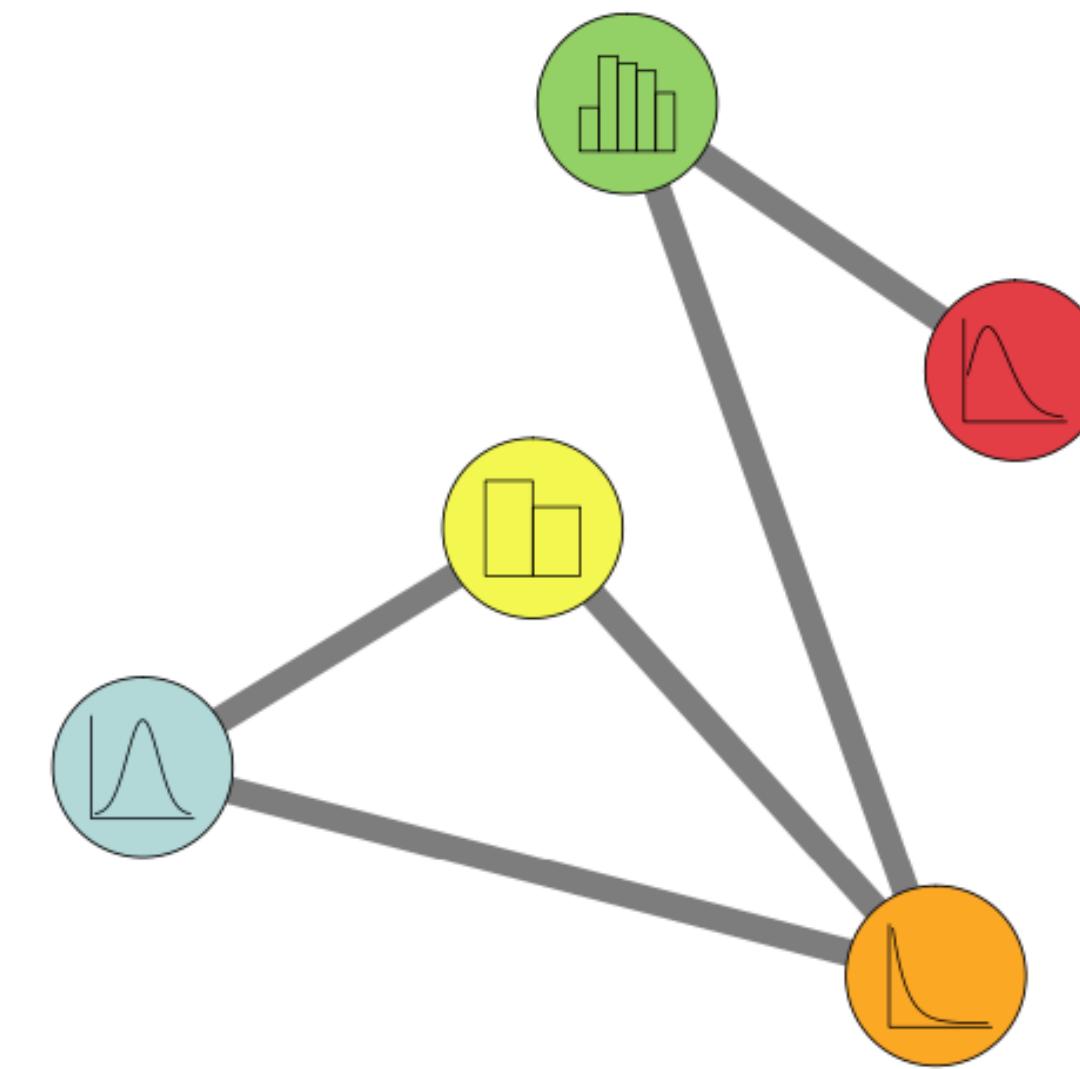
Check for arguments
and see original paper
for more details!



Borkulo et al. 2014

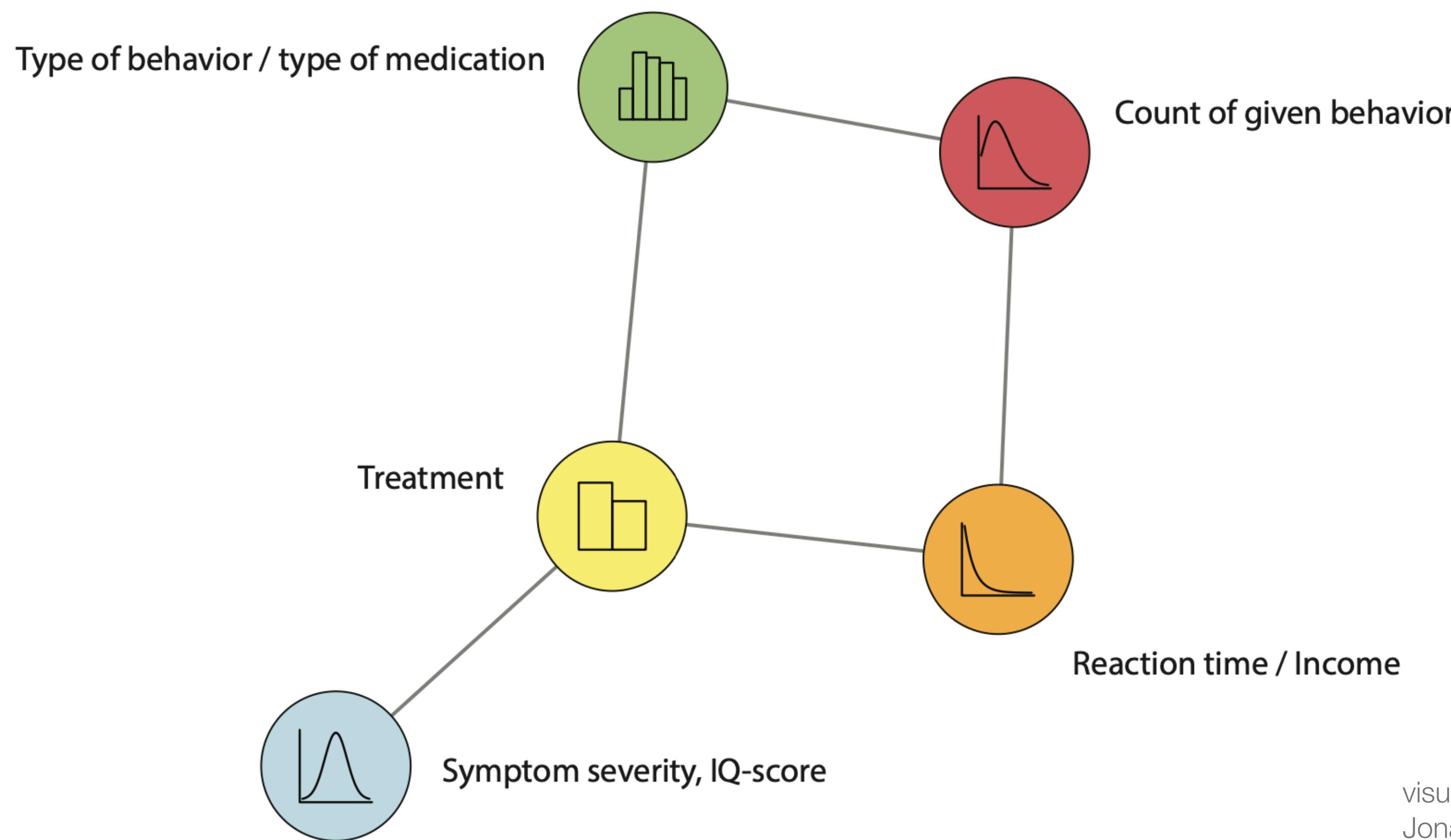
Mixed graphical models

When the variables are of different types



visualisation by
Jonas Haslbeck

Mixed Graphical Model (MGM)

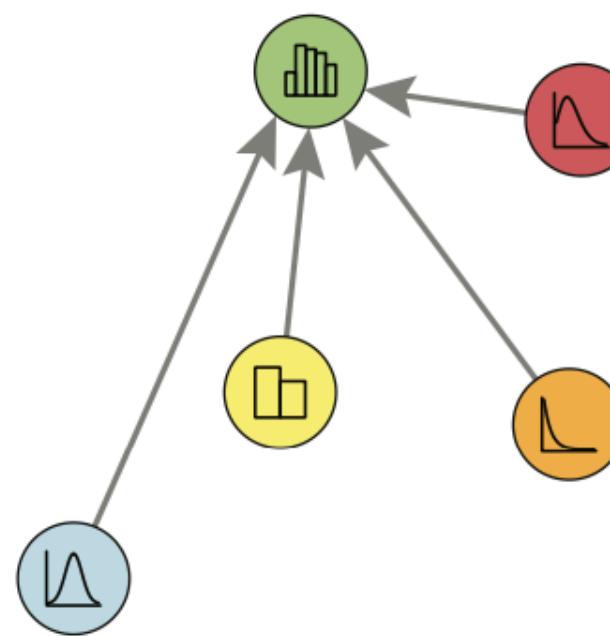


visualisation by
Jonas Haslbeck

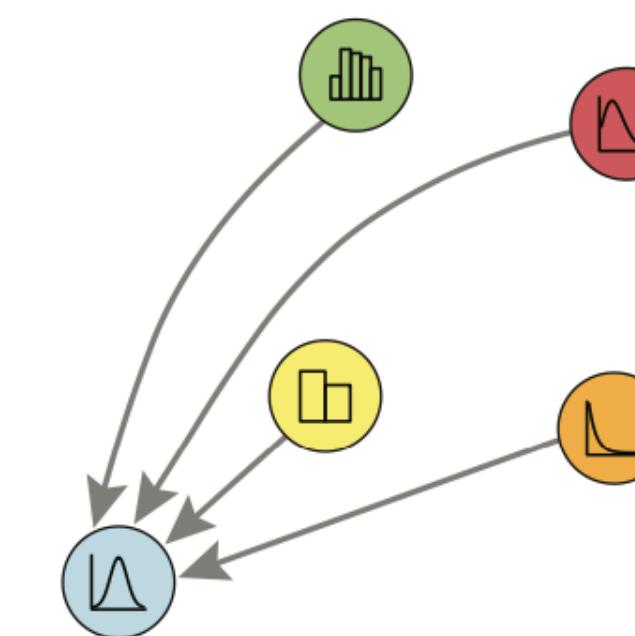
MGM: estimation

Also see: <https://jmbh.github.io/>

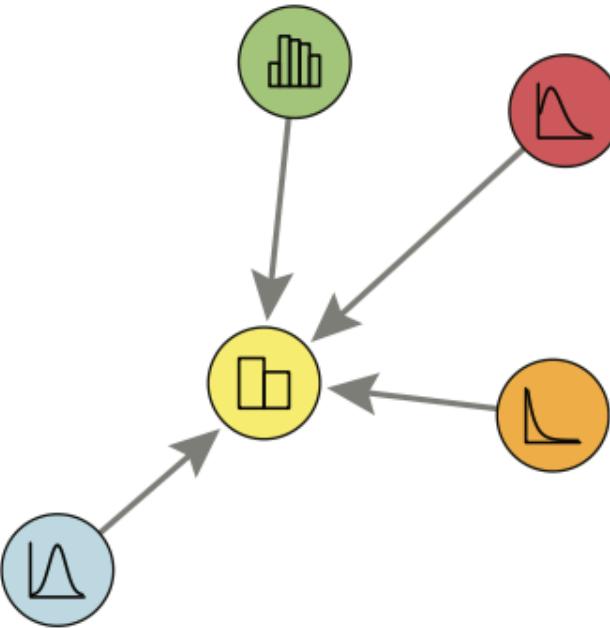
Step 1



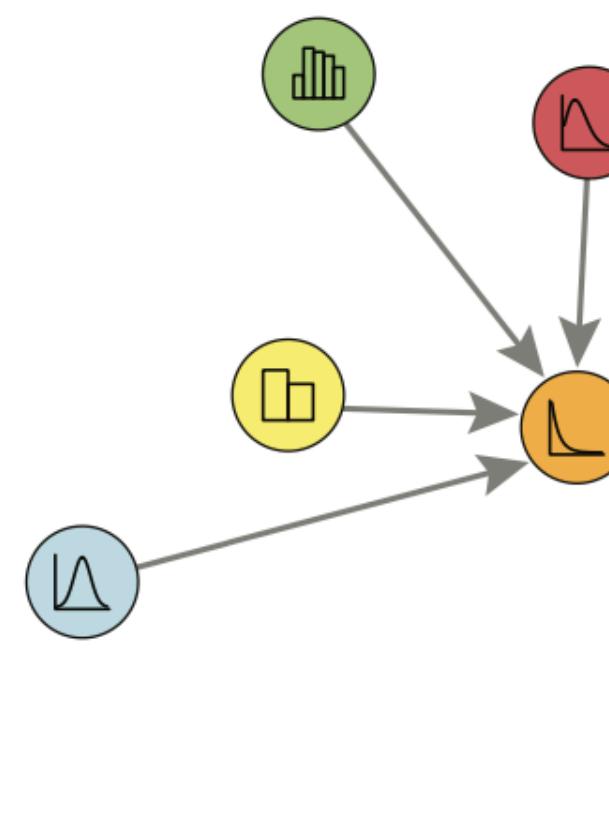
Step 2



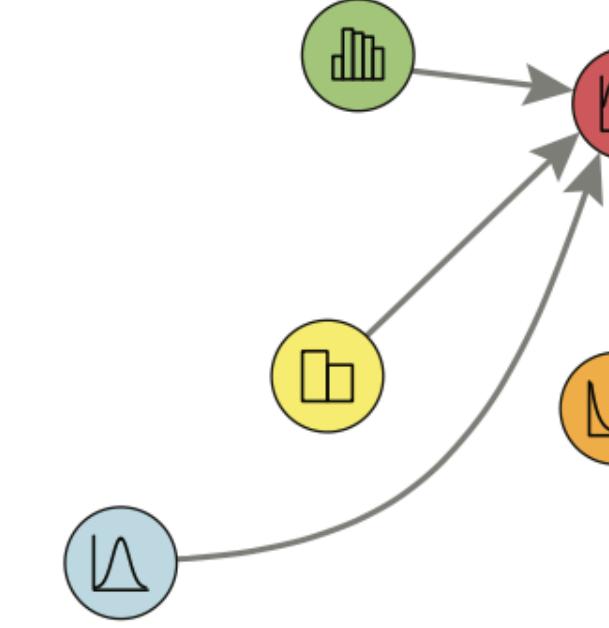
Step 3



Step 4



Step 5



visualisation by
Jonas Haslbeck

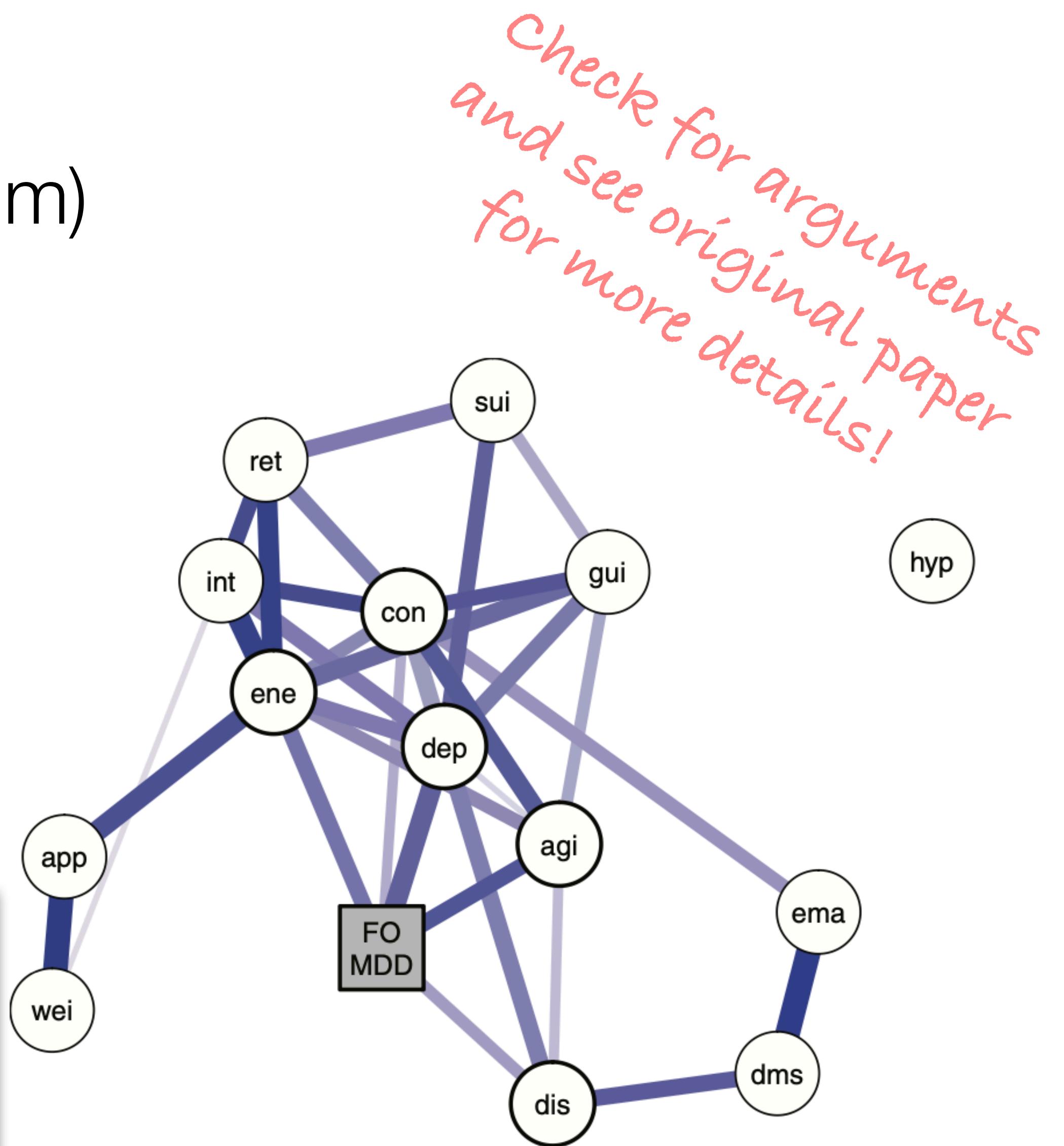
Estimating an mixed graphical model (mgm)

```
# Estimate Ising model:  
net_mgm <- estimateNetwork(data, default = "mgm")
```

```
# Plot network (saving to PDF):  
plot(net_mgm, filetype = "pdf")
```

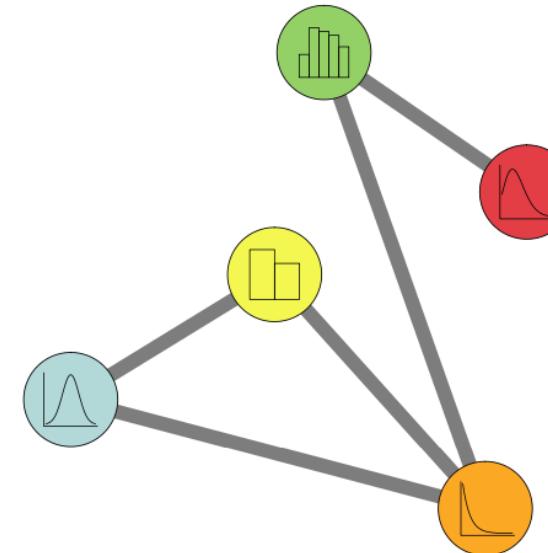
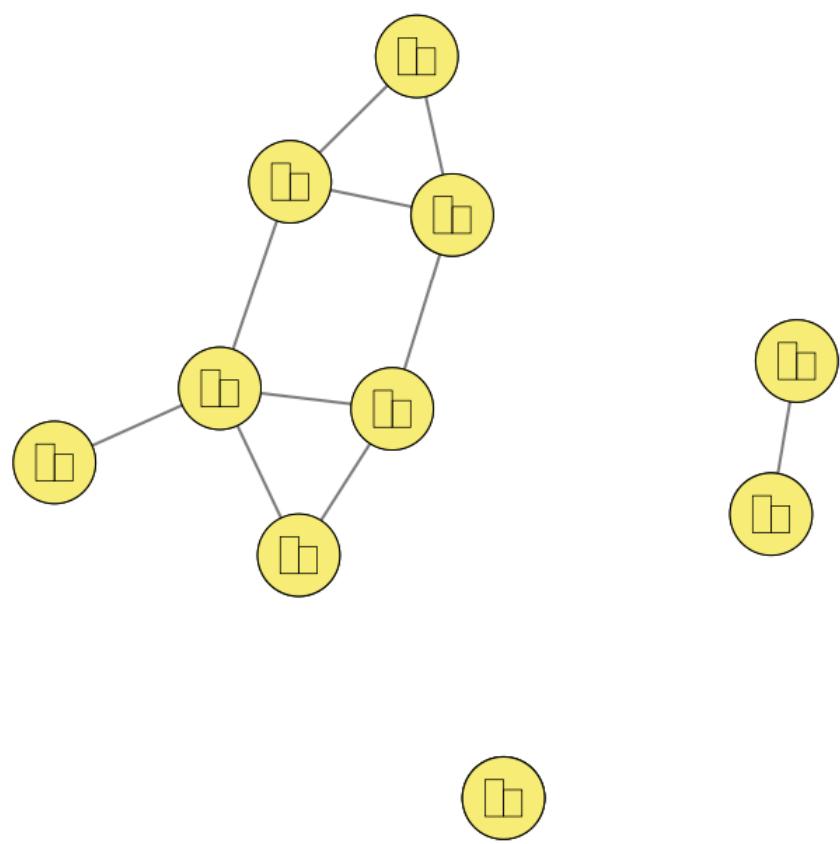
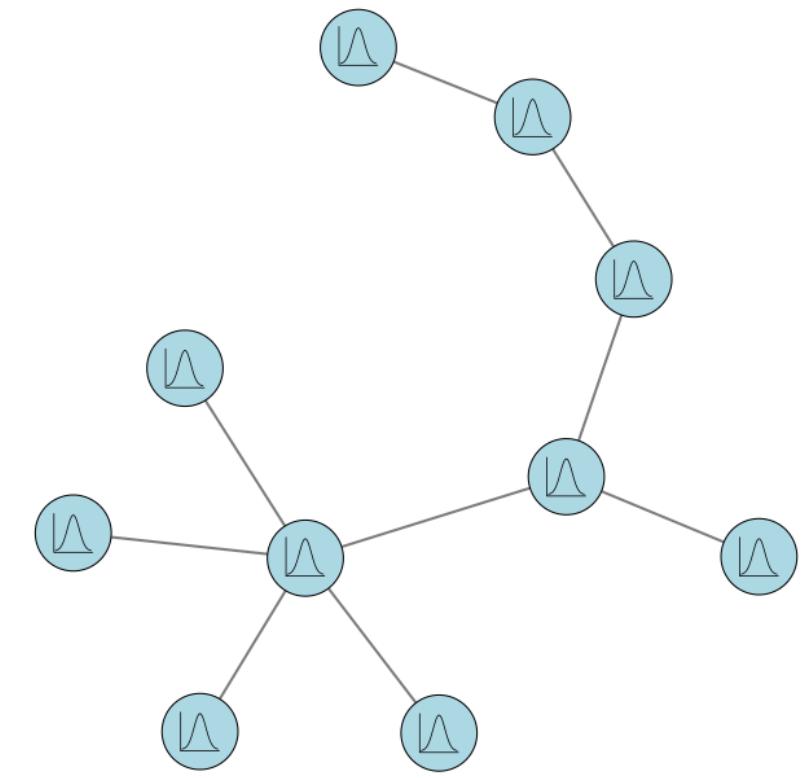
```
bootnet_mgm(data, type, level, tuning = 0.5, missing =  
  c("listwise", "stop"), verbose = TRUE, criterion =  
  c("EBIC", "CV"), nFolds = 10, order = 2, rule =  
  c("AND", "OR"), binarySign, unlock = FALSE, transform  
  = c("none", "rank", "quantile"), ...)
```

```
> net_mgm  
--- Estimated network ---  
Number of nodes: 9  
Number of non-zero edges: 10 / 36  
Mean weight: 0.01130081  
Network stored in x$graph  
  
Default set used: mgm  
  
Use plot(x) to plot estimated network  
Use bootnet(x) to bootstrap edge weights and centrality indices  
  
Relevant references:  
  
  Jonas M. B. Haslbeck, Lourens J. Waldorp (2016). mgm: Structure Estimation for Time-Varying Mixed Graphic  
al Models in high-dimensional Data arXiv preprint:1510.06871v2 URL http://arxiv.org/abs/1510.06871v2.  
  Epskamp, S., Borsboom, D., & Fried, E. I. (2016). Estimating psychological networks and their accuracy: a  
tutorial paper. arXiv preprint, arXiv:1604.08462.
```



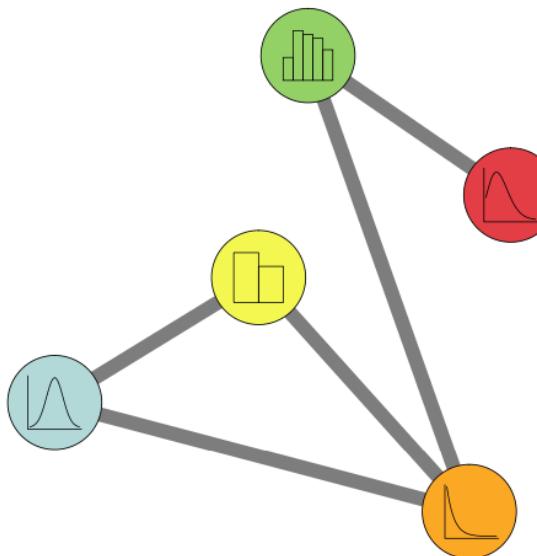
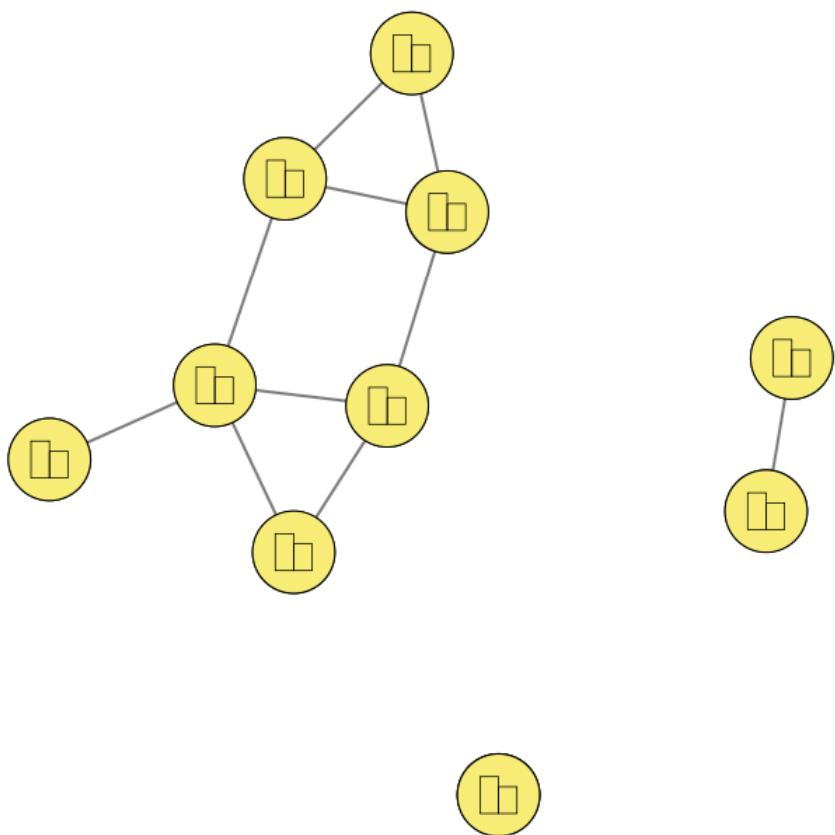
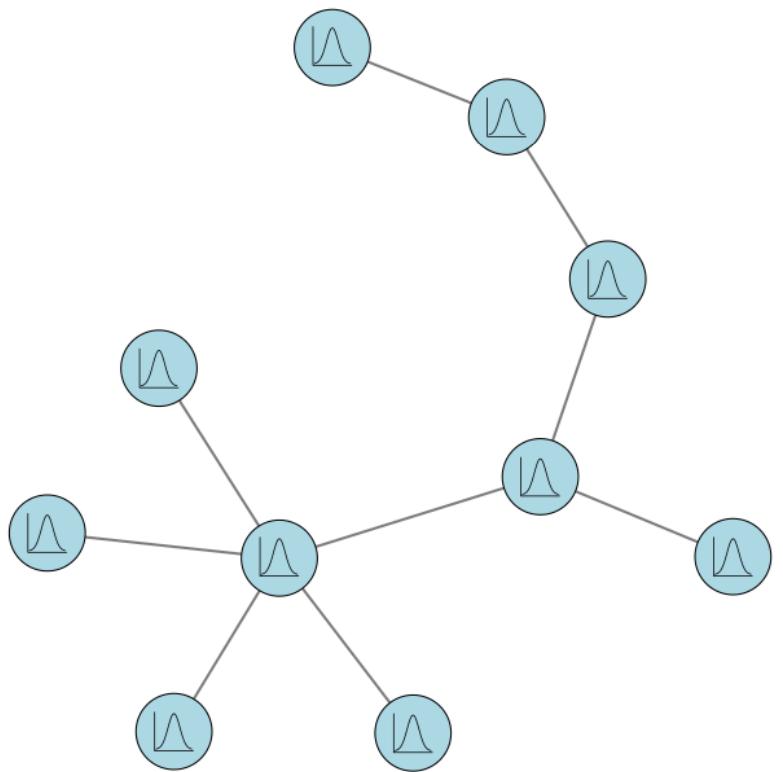
Interim summary

- Pairwise Markov random fields are undirected networks, of which the networks can be interpreted in several ways
 1. A representation of conditional independence relationships
 2. Pairwise interactions
 3. Highlight potential causal pathways
 4. Highlight latent variables as clusters
 5. Predictive effects



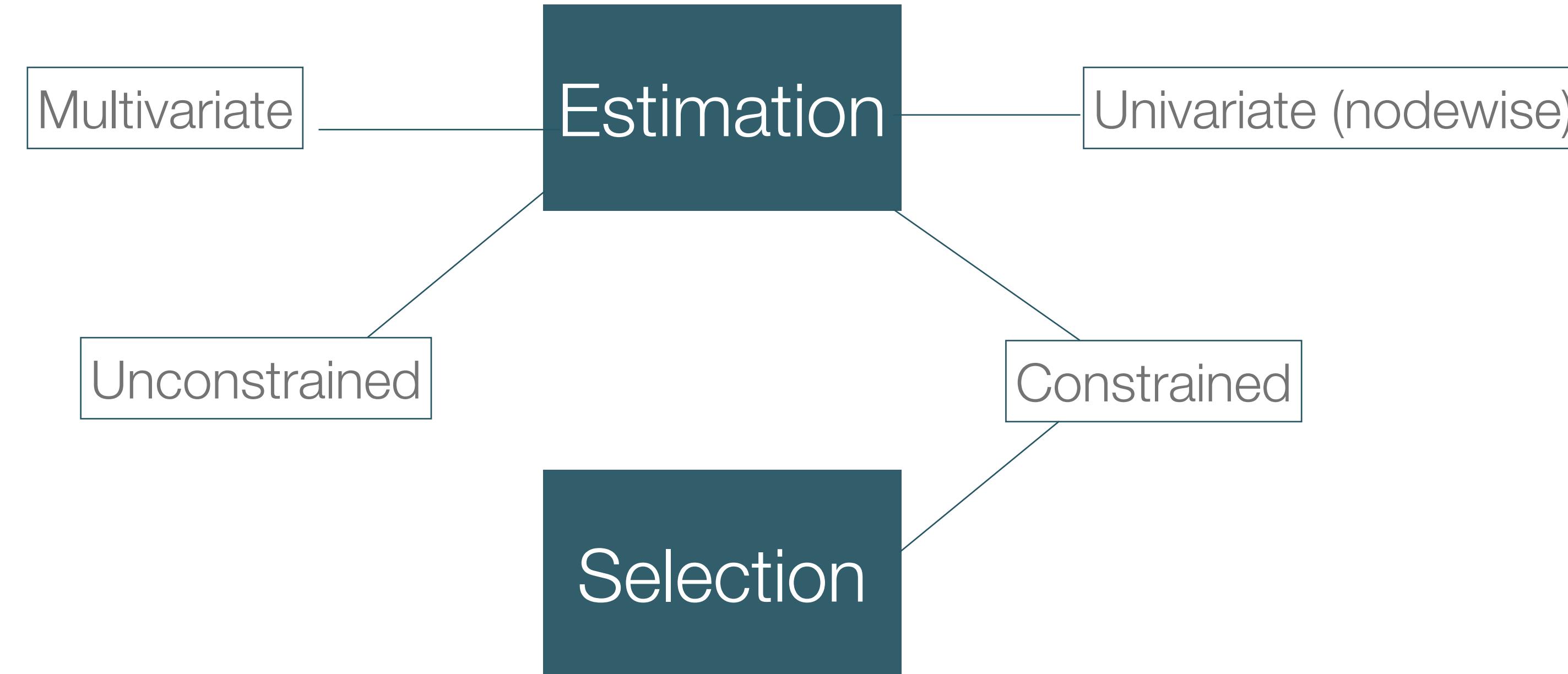
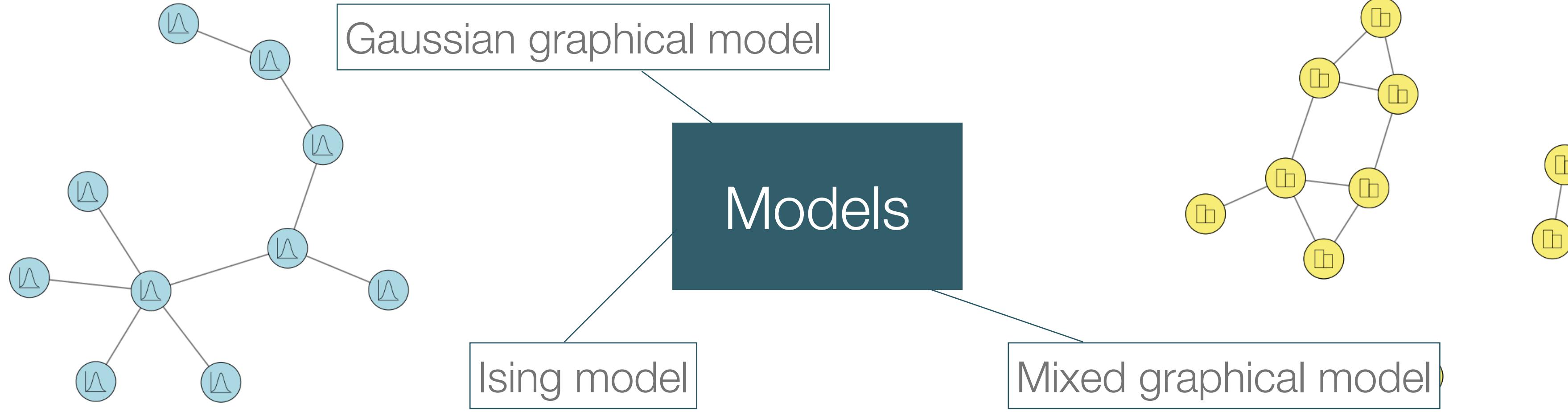
Interim summary

- We discussed three main types of PMRFs
 - Gaussian graphical model
 - Continuous (normal) variables
 - Edge-weights equal to partial correlation coefficients
 - Obtained from:
 - Inverting and standardising the variance-covariance matrix
 - Multiple regression models for each variable
 - Ising model
 - Binary variables
 - Edge-weights indicate conditional association
 - Obtained from multiple logistic regression models for each variable
 - Mixed graphical model
 - Different (mixed) types of variables
 - Edge-weights indicate conditional association
 - Obtained from multiple regression models for each variable



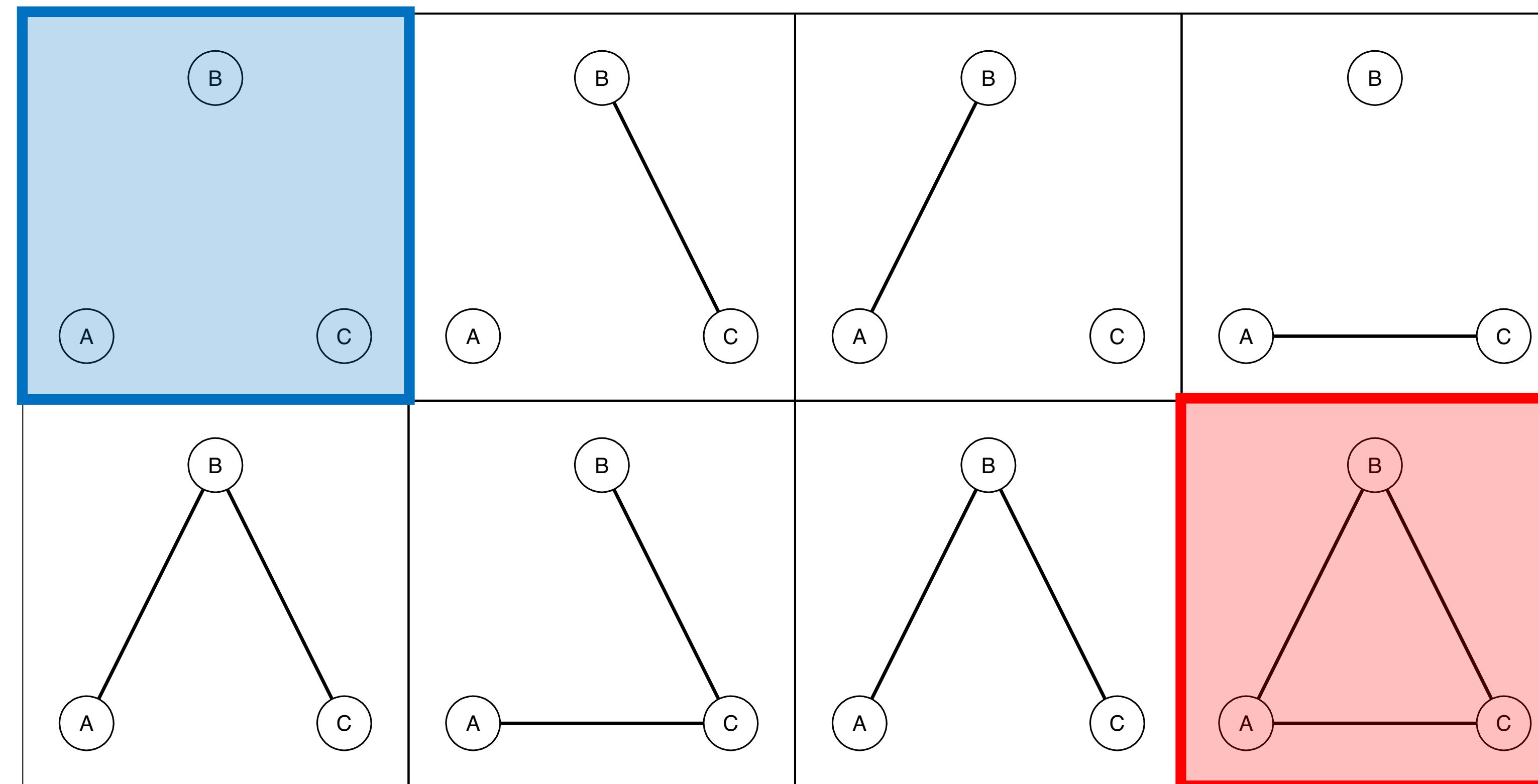


Model selection



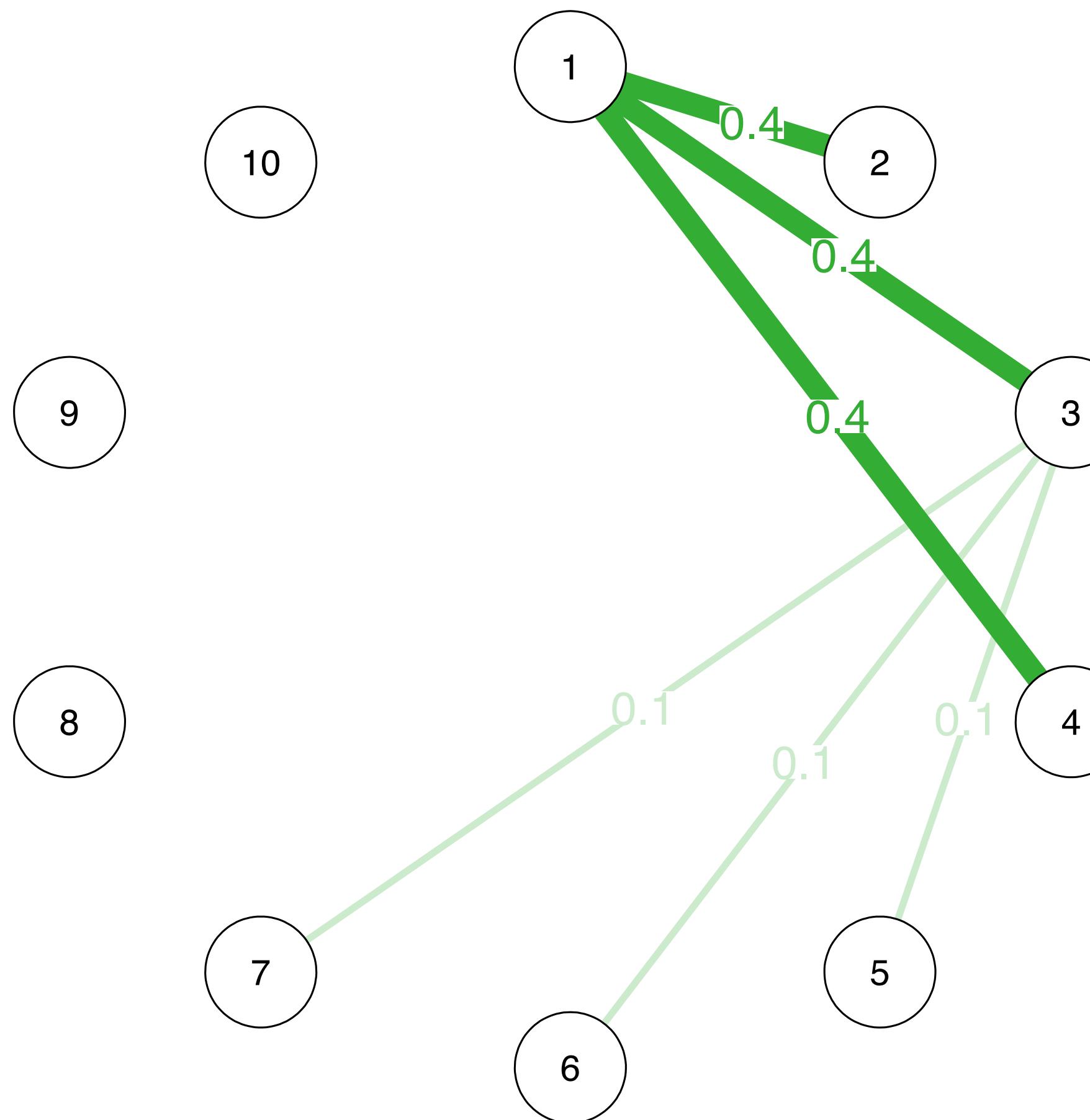
The problem of model selection

A simple (independence) model

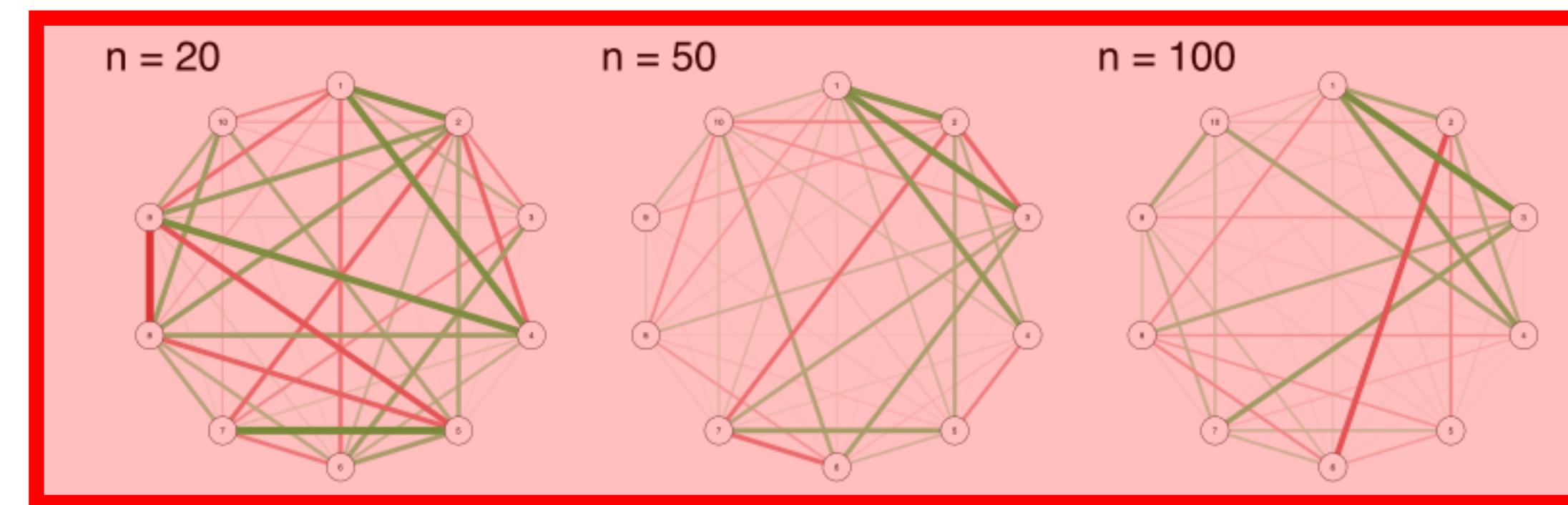


A complicated (saturated) model

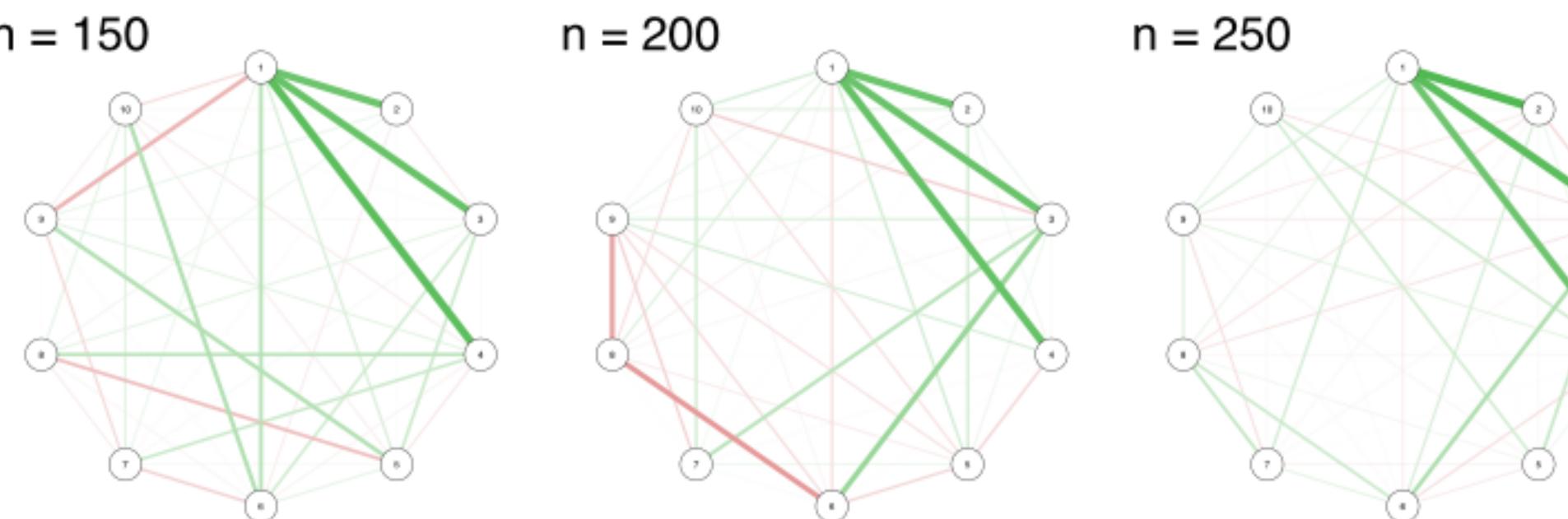
Let's sample data from a multivariate Gaussian with the following partial correlations:



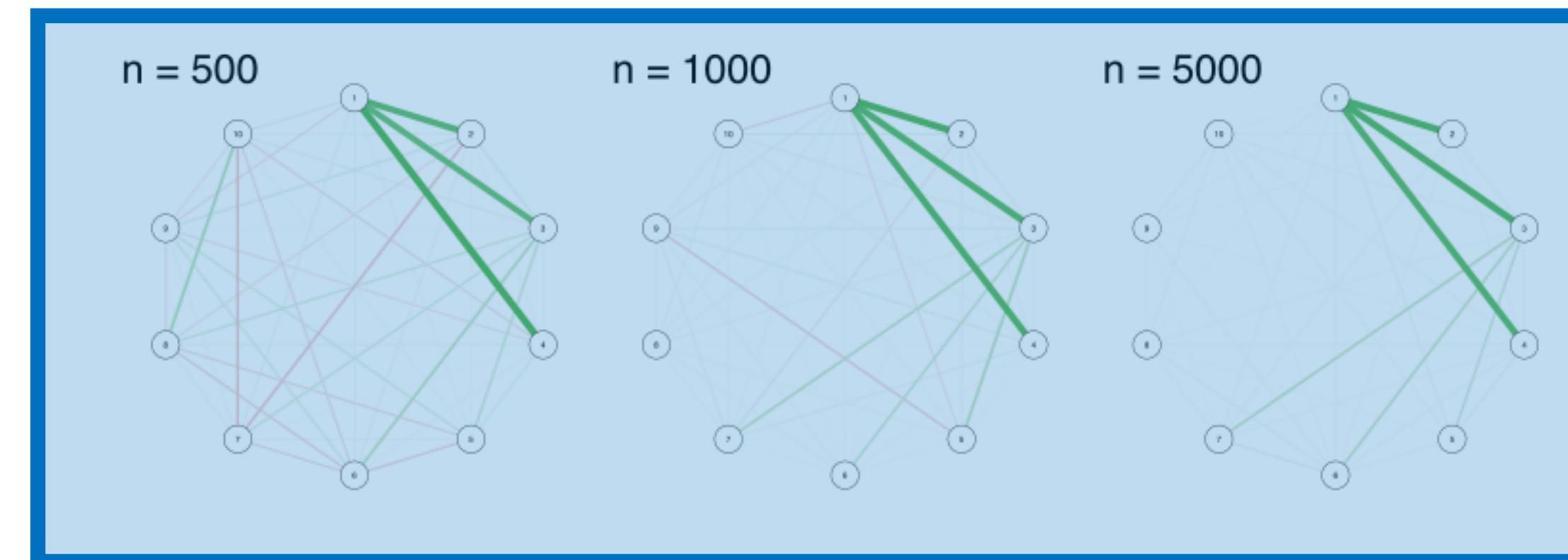
And estimate the model with different numbers of observations n :



At low n :
terrible
inference!



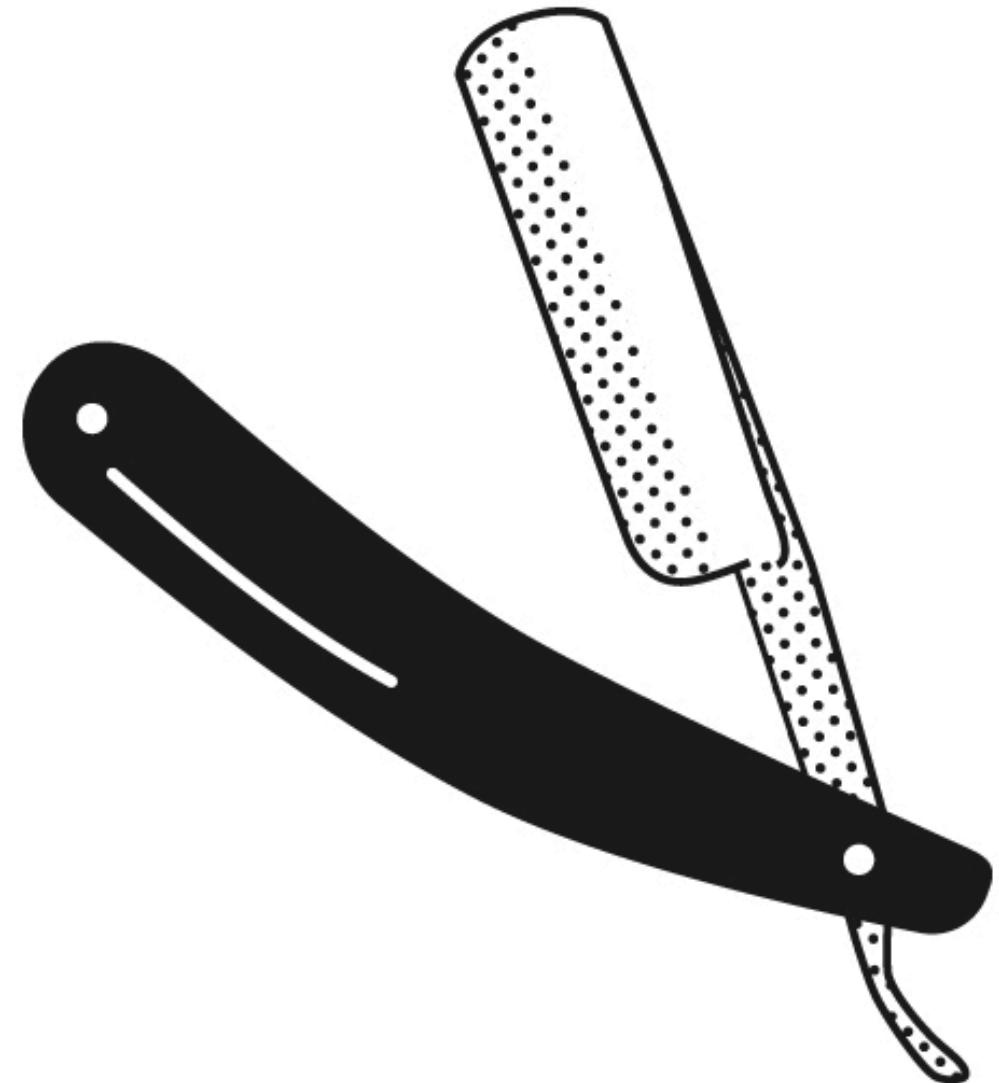
At higher n ,
more accurate
pictures, but
still many false
(vague) edges



No model selection! All models are equally complex (all edges are estimated)!

Model selection

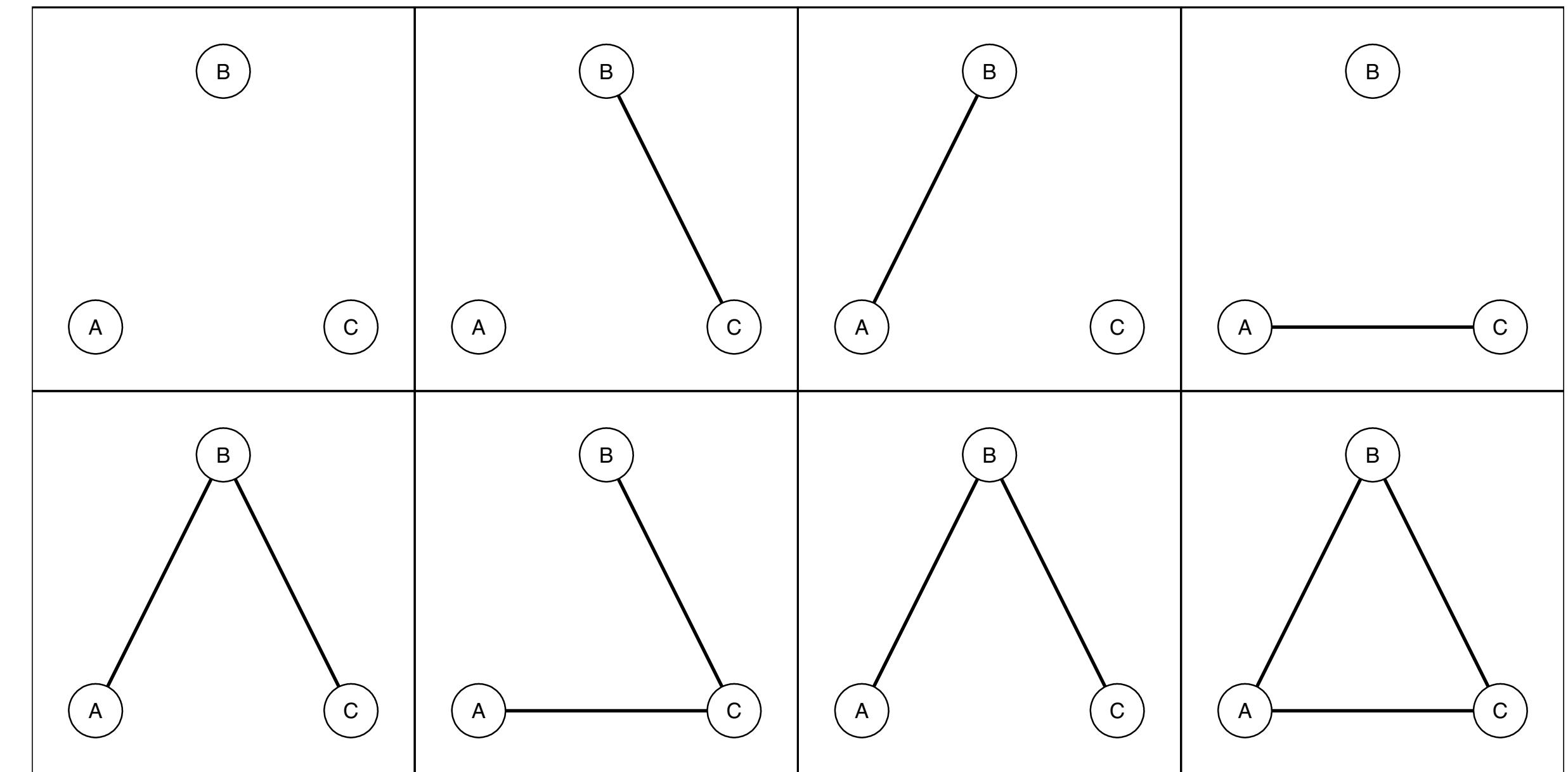
- Occam's Razor: when comparing equally performing models, we prefer the simplest one!
- In networks: we want to remove edges (simplify the model) whenever possible
- Because of this: edges that **are** included in the network are **substantial**
 - At least, so we hope
- Many model selection algorithms have been proposed, not clear yet which one is the best!



How to select the best model?

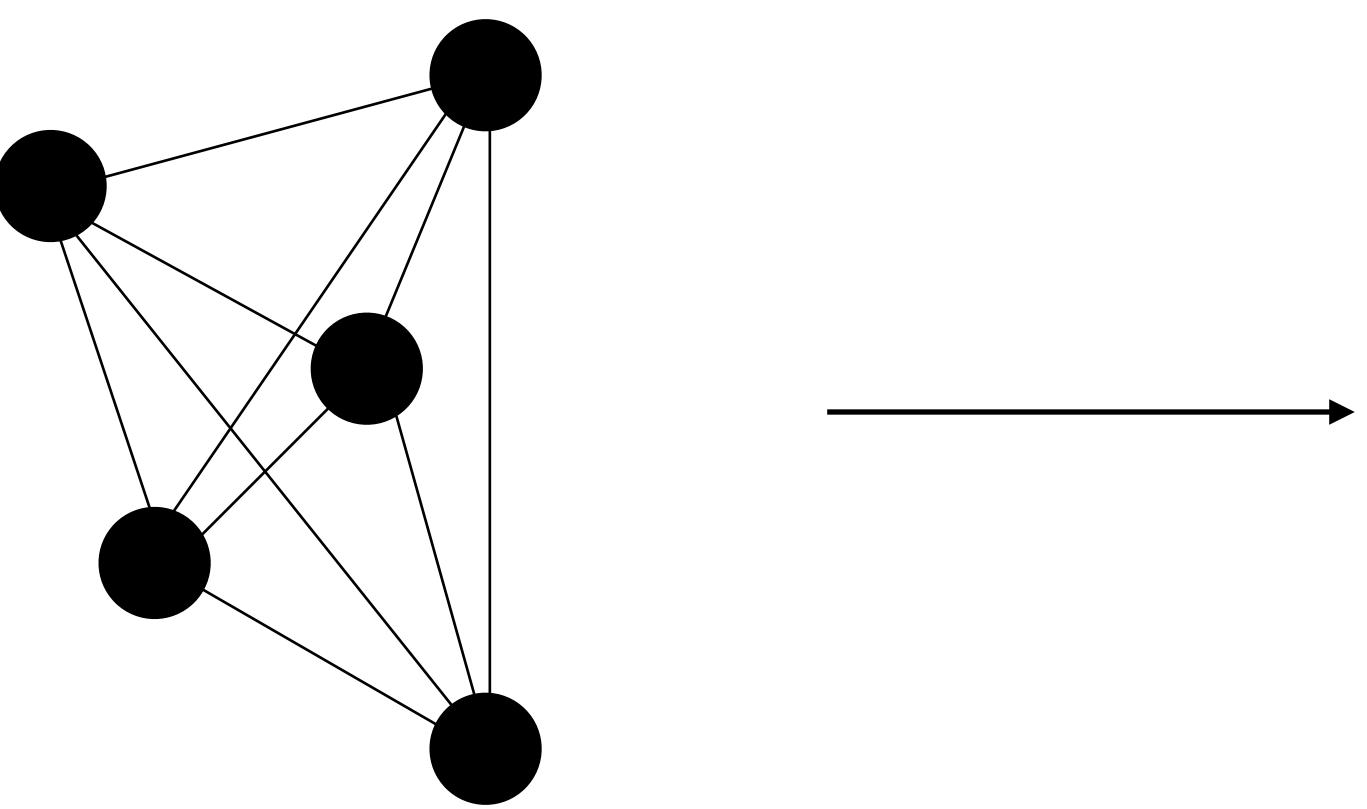
Different estimation strategies

1. Thresholding and pruning
2. Regularisation
3. Model search

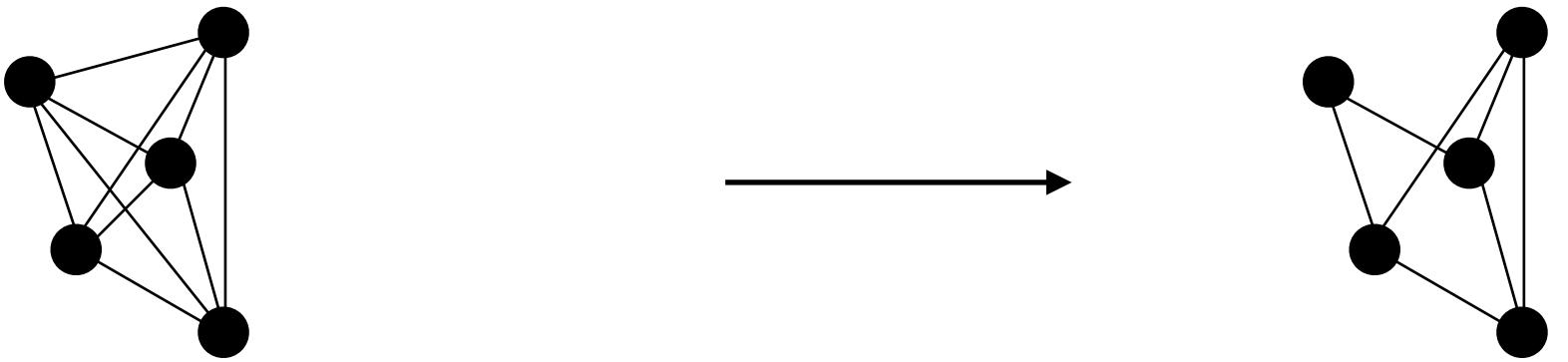


Thresholding and pruning

- Estimate saturated model, then remove edges based on some criterion
 - e.g., significance, Bayes Factor, false discovery rate
- Thresholding: *hide* edges
- Pruning: *remove* edges and re-estimate model with those edges fixed to zero
- In node-wise estimation: two values per edge
 - “AND” rule: select edge if *both* $A \rightarrow B$ and $A \leftarrow B$ meet criterion
 - “OR” rule: select edge if *either* $A \rightarrow B$ or $A \leftarrow B$ meets criterion

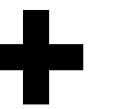
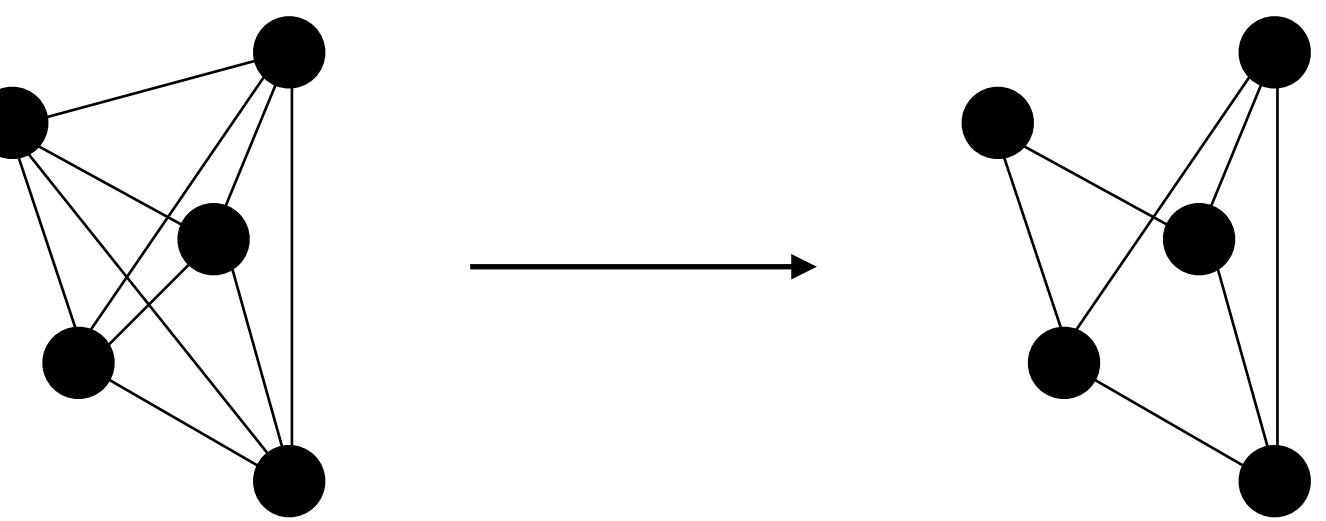


Thresholding and pruning



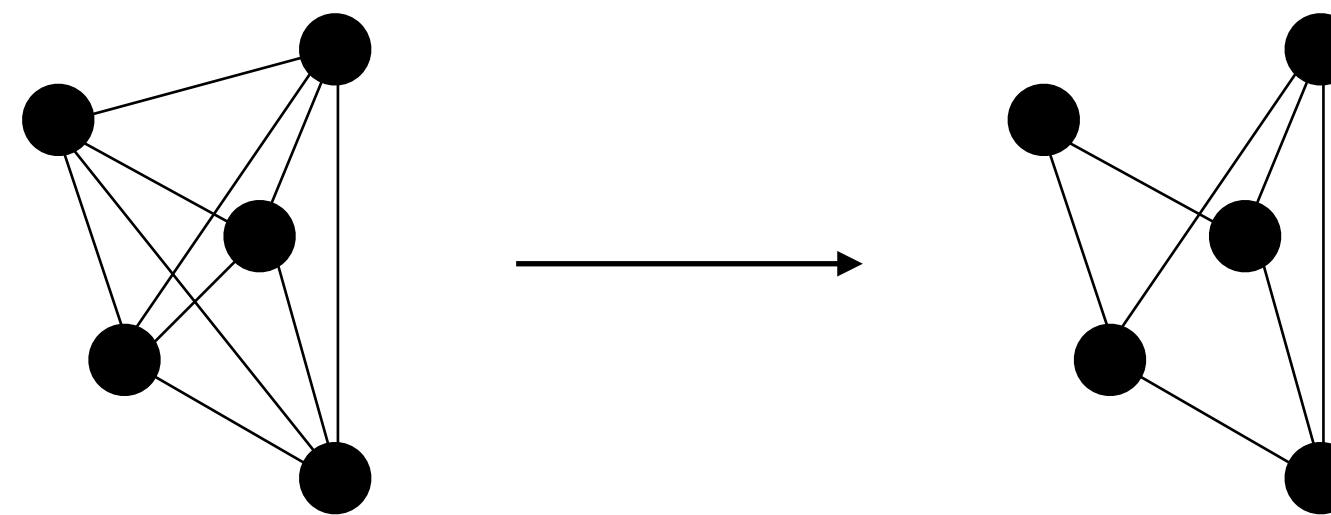
- Estimate saturated model, then remove edges based on some criterion
 - e.g., significance, Bayes Factor, false discovery rate
- Thresholding: *hide* edges
 - `estimateNetwork(data = bfiData,
default = "pcor",
threshold = "sig", alpha = 0.01)`
- Pruning: *remove* edges and re-estimate model with those edges fixed to zero
 - implemented in psychonetrics package
 - `mod <- ggm(bfiData) %>% runmodel %>% prune(alpha = 0.01)
net <- getmatrix(mod, "omega")`
 - Can be done recursively, in that case pruning is a form of model selection

Pruning



- Very fast
- Fixed false positive rate per edge
 - e.g., $\alpha = 0.05$ then 5% chance of including false positive, 5% of false positive edges in your network
- Non-biased estimates (as compared to regularised estimates, see later)

Pruning



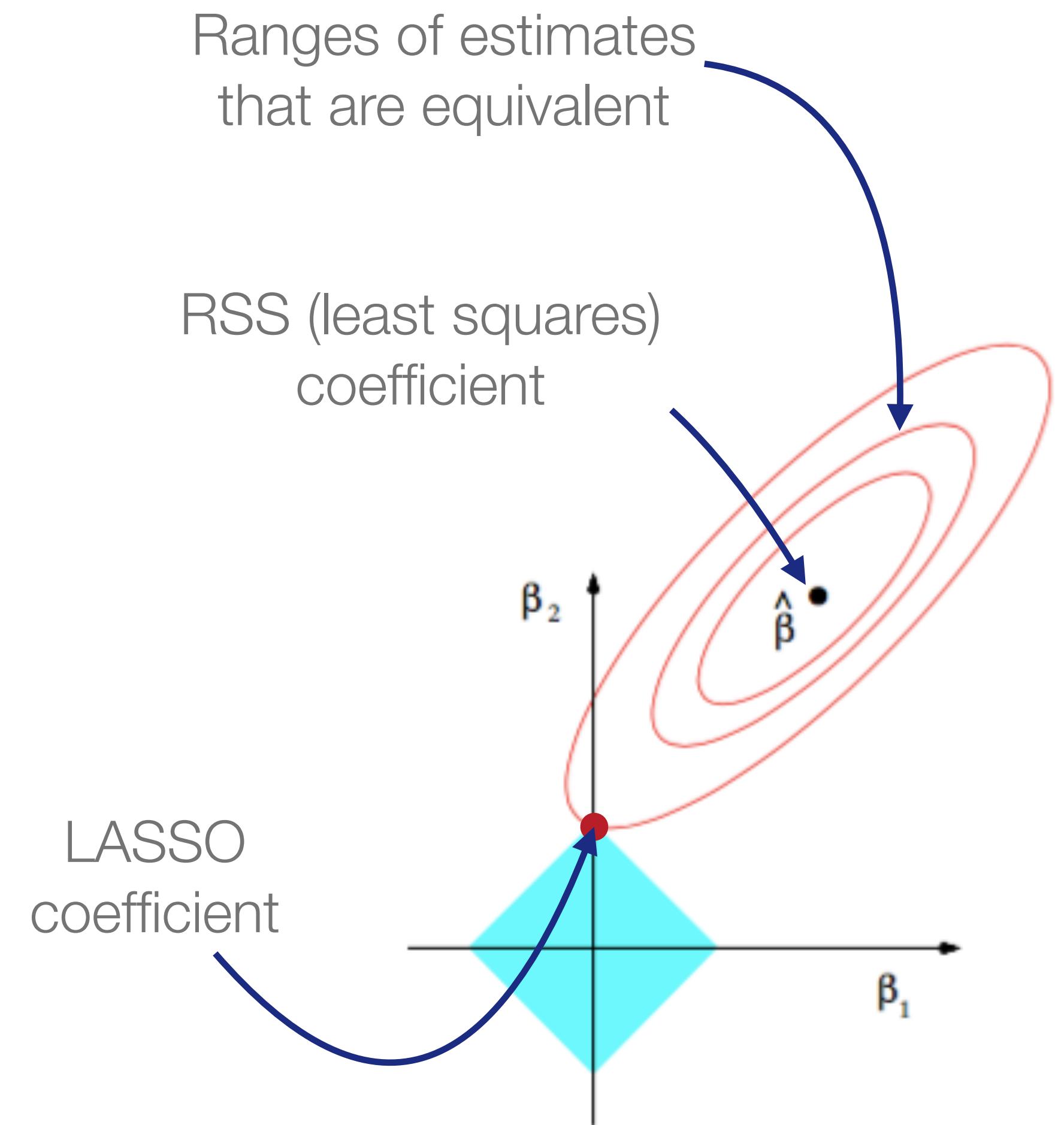
+

-

- Very fast
- Fixed false positive rate per edge
 - e.g., $\alpha = 0.05$ then 5% chance of including false positive, 5% of false positive edges in your network
- Non-biased estimates (as compared to regularised estimates, see later)
- No model selection (except when using recursive pruning)
- Conservative: can lack sensitivity (power)
- Possibly more sampling variation at low sample sizes, leading to potentially prominent false positive edges
 - especially in Ising models (because of the range of the parameters)
- Not always possible (if sampling distribution of edges is not known)

Regularisation

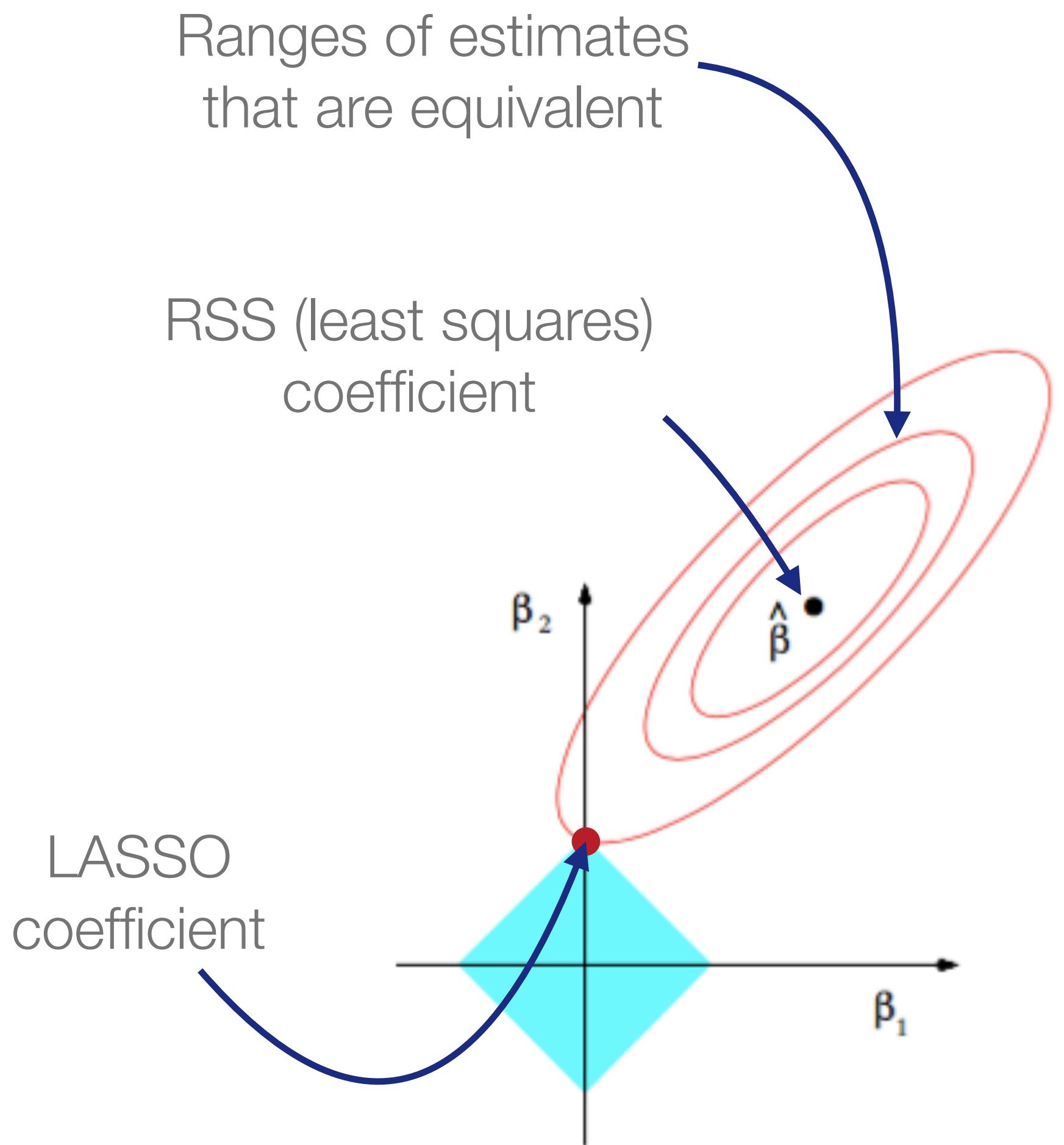
- Technique for joint model selection and parameter estimation from machine learning
 - Lasso regularisation:
least absolute shrinkage and selection operator



Friedman, Hastie, Tibshirani. (2001).
The elements of statistical learning.

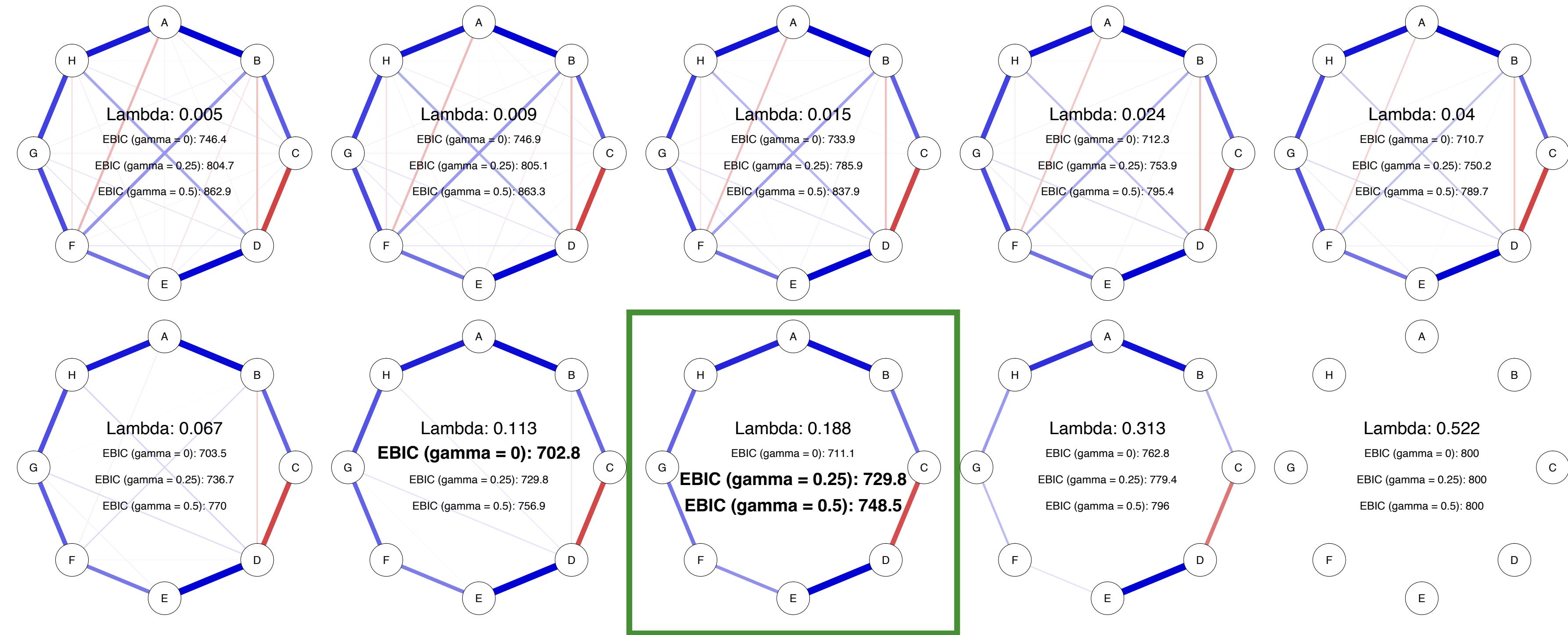
Regularisation

- Technique for joint model selection and parameter estimation from machine learning
 - Lasso regularisation:
least absolute shrinkage and selection operator
- Size of the box controls the amount of shrinkage and selection of parameters:
 - Larger the box: less regularisation
- Size of the box is controlled by the tuning parameter λ
 - $\lambda = 0$: box is infinitely big and you will end up at the maximum likelihood estimate (no regularisation)
 - Higher lambda \rightarrow smaller the box \rightarrow more regularisation



Friedman, Hastie, Tibshirani. (2001).
The elements of statistical learning.

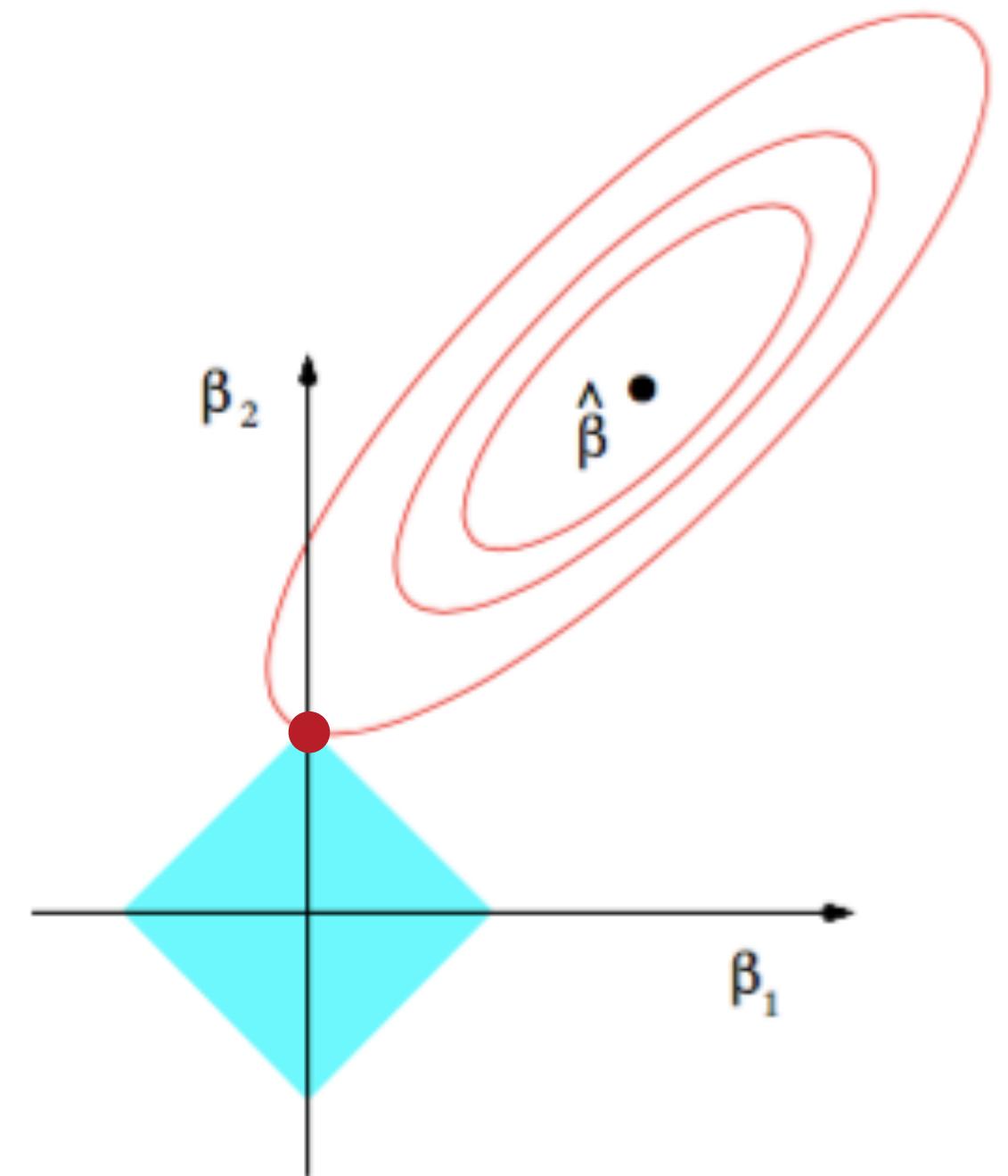
Regularisation: determine λ



- Vary λ many times, and for each λ you get a different network: from very dense network (top left) to unconnected very sparse network (bottom right)
- Use fit criterion to select the best model and to “determine the size of the box”
- Multiple criteria possible (e.g., cross-validation), in psychological networks we often minimise the Extended Bayesian Information Criteria (EBIC)
 - EBIC requires a hypertuning parameter γ that needs to be set manually: higher values \rightarrow more penalisation for complexity \rightarrow more conservative

Regularisation

- Network structure and parameters are estimated at the same time
- For GGM can be applied directly to the precision matrix (EBICglasso; i.e., multivariate estimation)
- When used in node-wise regression (i.e., univariate estimation):
 - “AND” rule: select edge if *both* $A \rightarrow B$ and $A \leftarrow B$ are included
 - “OR” rule: select edge if *either* $A \rightarrow B$ or $A \leftarrow B$ is included

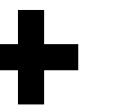


Regularisation

- Technique for joint model selection and parameter estimation from machine learning
 - Lasso regularisation:
least absolute shrinkage and selection operator
 - EBICglasso for GGM
 - IsingFit for Ising model
- `estimateNetwork(data = bfiData,
 default = "EBICglasso")`
- `estimateNetwork(data = binaryData,
 default = "IsingFit")`

When gamma is not specified (called 'tuning' in `estimateNetwork()`), the default is used. Default values differ across packages, so be aware!

Regularisation



- Very fast and sensitive to detecting edges
- Retrieves a structure on low sample sizes
- Shrinkage can improve visual representation (clearer picture)

Regularisation



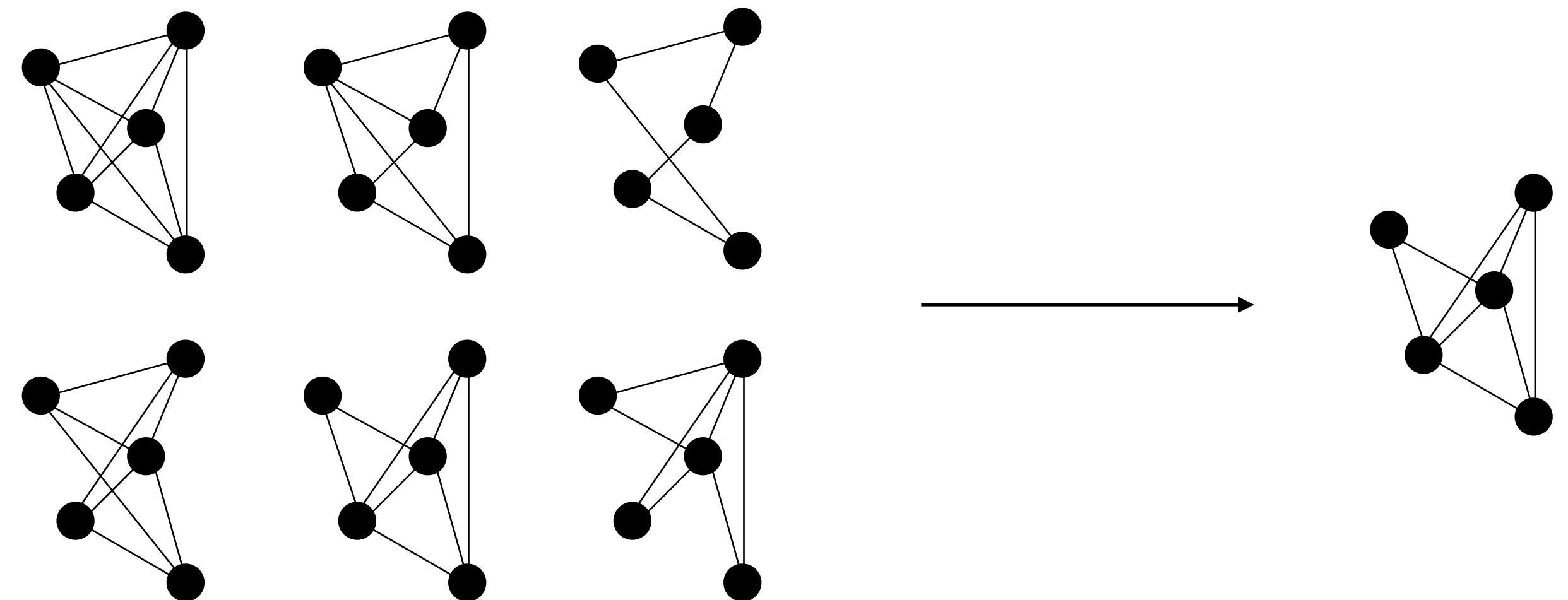
- Very fast and sensitive to detecting edges
- Retrieves a structure on low sample sizes
- Shrinkage can improve visual representation (clearer picture)



- Always biased estimates
- Poor performance in high sample sizes: low specificity
- Strongly reliant on assumption of sparsity: poorer performance in dense networks
- No fixed false positive rate

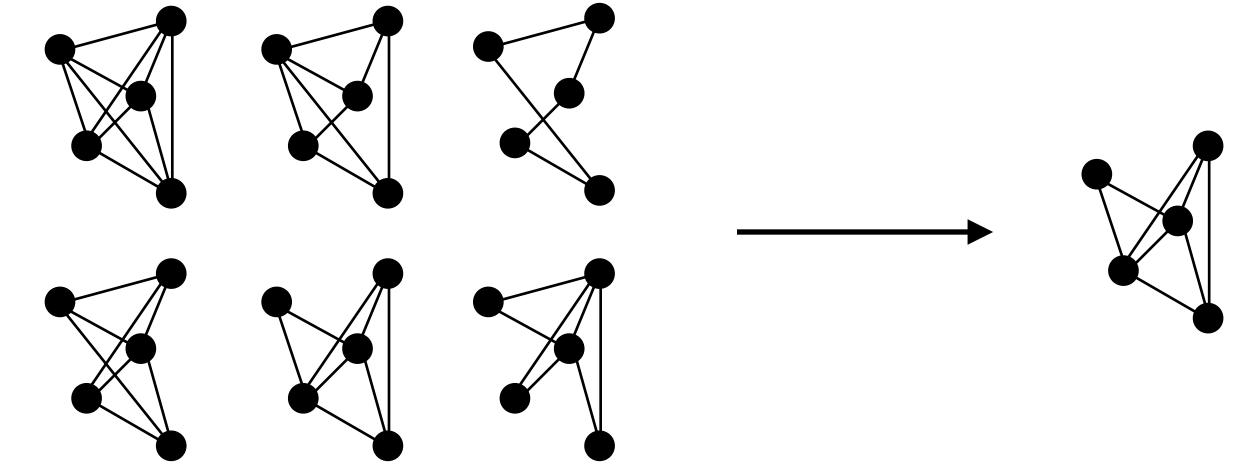
Model search

- Search for an optimal fitting model by adding and removing edges to optimise some criterion
 - e.g., BIC, AIC or CV prediction error
- Extensive model search strategies can be used to find the best model
- For each edge it is evaluated whether it can be *added* or removed by optimising some fit criterion
- For the resulting network you are certain that **no** edges can be added or removed to optimise the fit



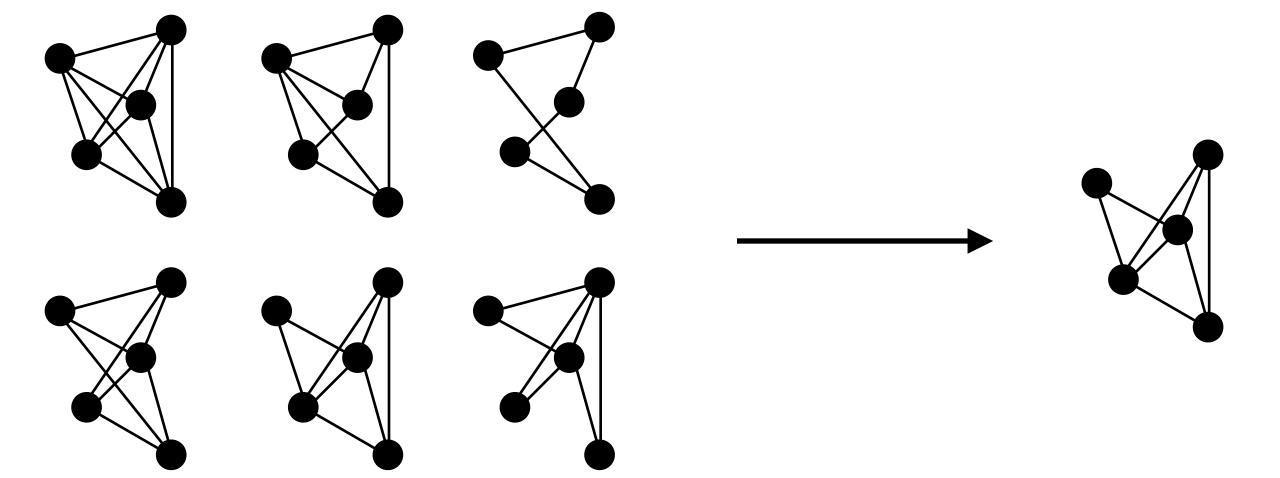
Model search

- Search for an optimal fitting model by adding and removing edges to optimise some criterion
 - e.g., BIC, AIC or CV prediction error
- ```
estimateNetwork(data = bfiData,
 default = "ggmModSelect",
 stepwise = FALSE)
```
- ggmModSelect uses regularisation:
  1. Starts with glasso to estimate 100 regularised networks (from sparse to dense)
  2. Re-fit all models without regularisation
    - This way we use glasso to fit the structure, but not for the parameters itself (this way we don't have biased parameter estimates)



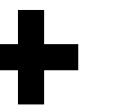
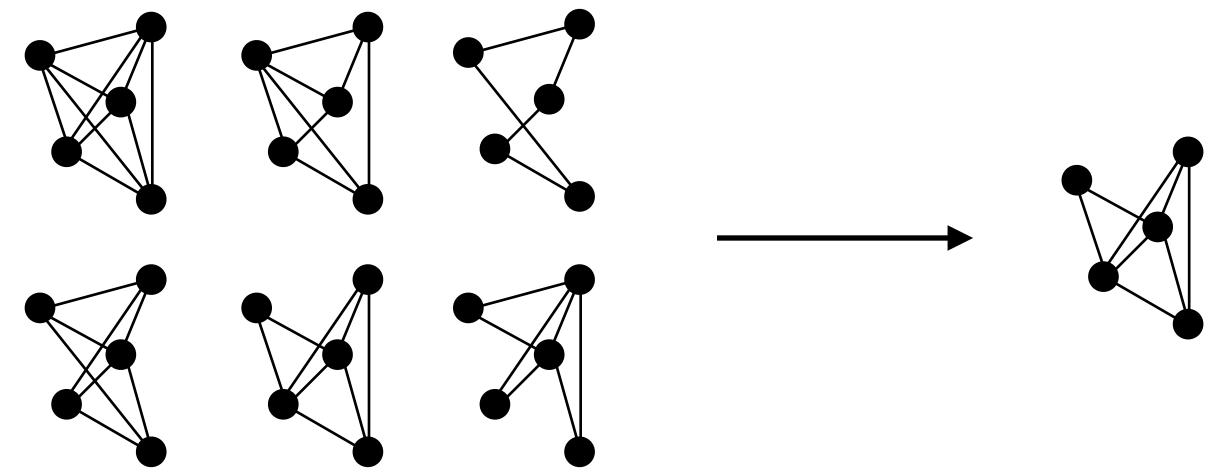
# Model search

- Search for an optimal fitting model by adding and removing edges to optimise some criterion
  - e.g., BIC, AIC or CV prediction error
- ```
estimateNetwork(data = bfiData,
                  default = "gqmModSelect",
                  stepwise = TRUE)
```
- ggmModSelect uses regularisation:
 1. Starts with glasso to estimate 100 regularised networks (from sparse to dense)
 2. Re-fit all models without regularisation
 - This way we use glasso to fit the structure, but not for the parameters itself (this way we don't have biased parameter estimates)
 3. Iteratively add and remove edges until BIC is optimised



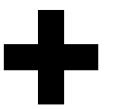
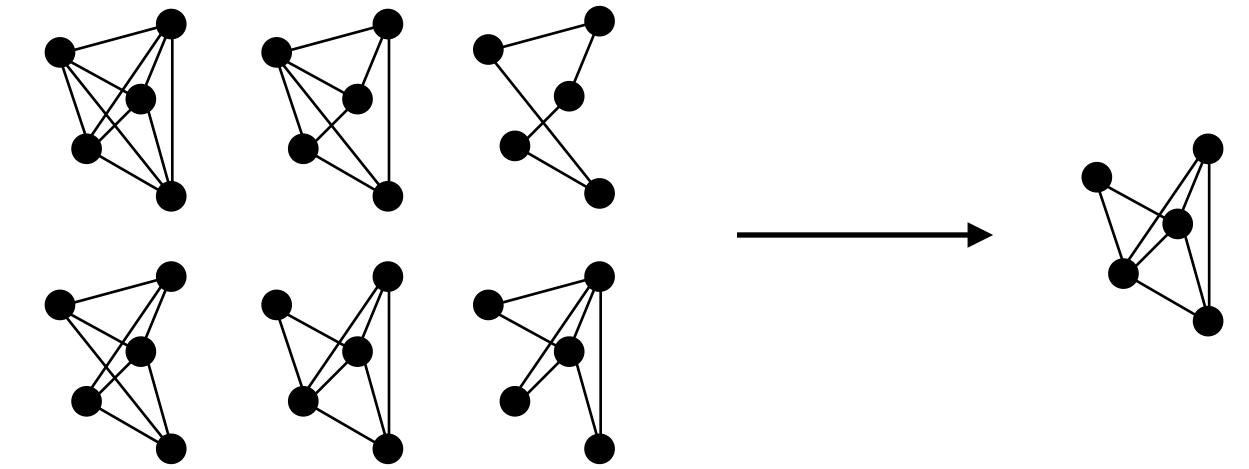
Can be slow when for
large networks

Model search



- Good performance in sensitivity and specificity
 - Conservative but powerful
- Can converge to the true model
- Non-biased estimates
- Local optimum: no edge can be added or removed to improve the fit

Model search



- Good performance in sensitivity and specificity
 - Conservative but powerful
- Can converge to the true model
- Non-biased estimates
- Local optimum: no edge can be added or removed to improve the fit



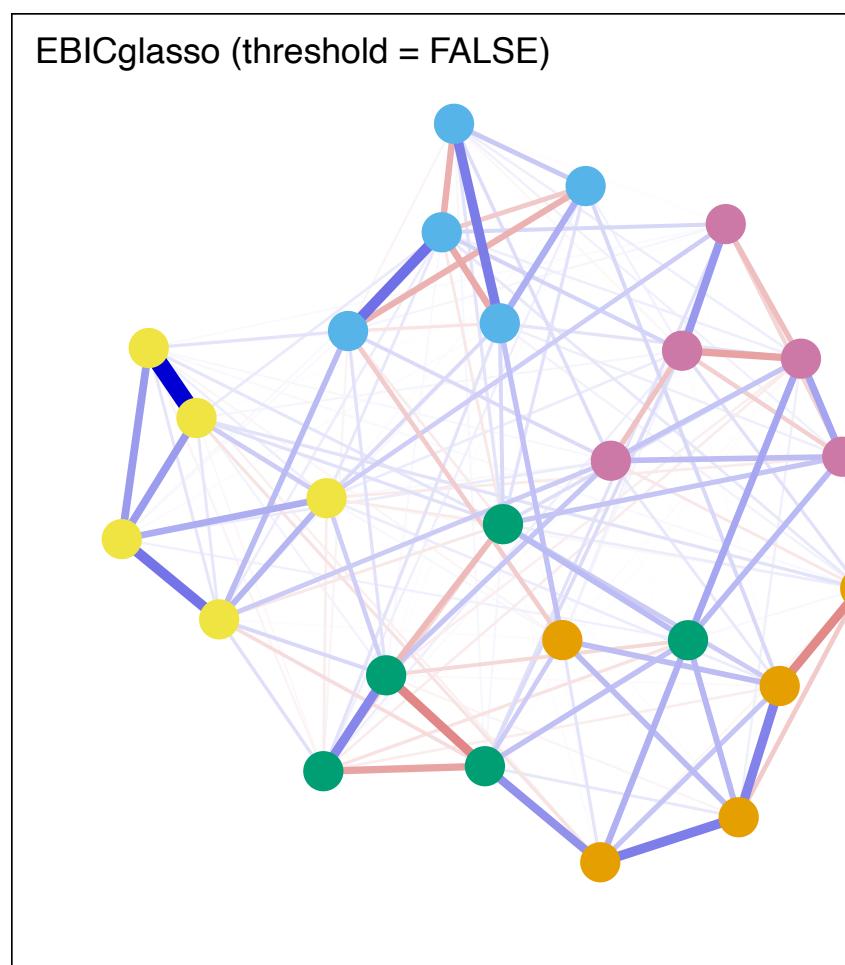
- Very slow with stepwise estimation
 - Especially for large networks (>25 nodes)
- No regularisation
 - Potentially prominent false edges

Summary

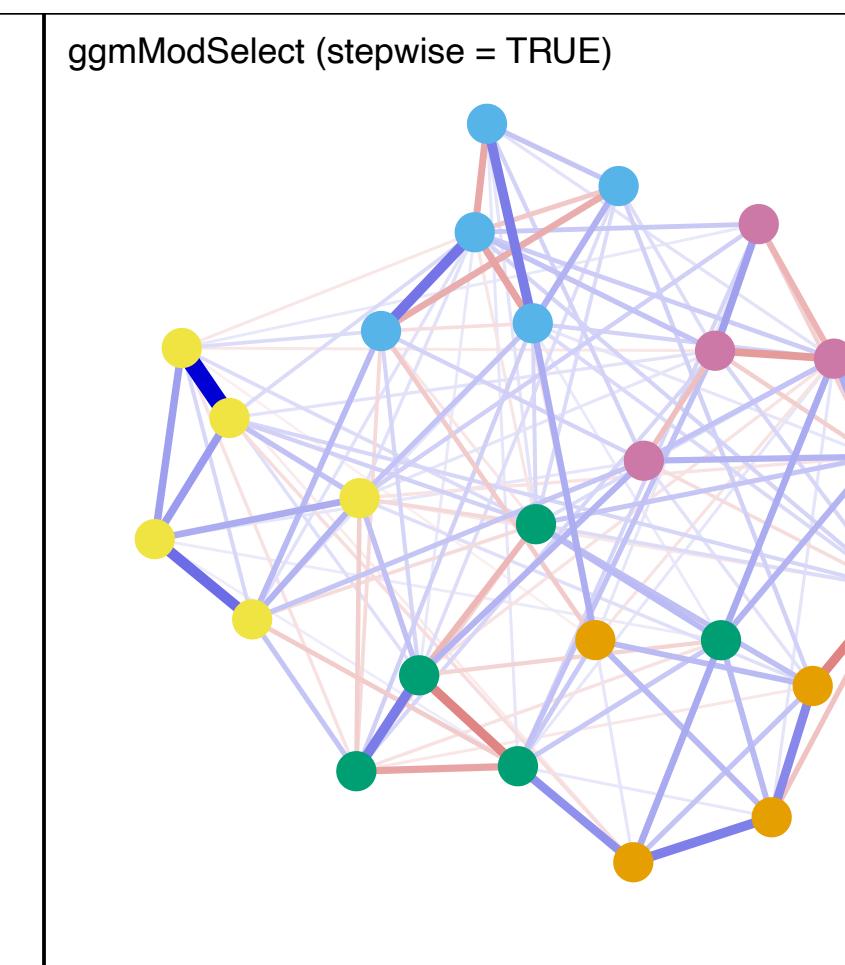
- 1. Thresholding and pruning: estimate saturated model, then remove edges based on some criterion (e.g., significance, Bayes Factor, false discovery rate)
 - Pros: very fast, non-biased estimates, can feature fixed false positive rate per edge
 - Cons: no model selection, not always possible, might not converge to true model, possibly more sampling variation at low sample sizes (especially in Ising model)
- 2. Regularisation: technique for joint model selection and parameter estimation from machine learning
 - Pros: fast, can be used when normal methods fail (e.g., low sample size or violated assumptions), shrinkage to zero can improve visual representation
 - Cons: always biased estimates, poorer performance in high sample sizes
- 3. Model search: Search an optimal fitting model by adding and removing edges to optimise some criterion (e.g., stepwise search)
 - Pros: model selection, non-biased estimates, can converge to true model
 - Cons: can be slow (especially when performing multivariate model search), possibly more sampling variation at low sample sizes

What method to use?

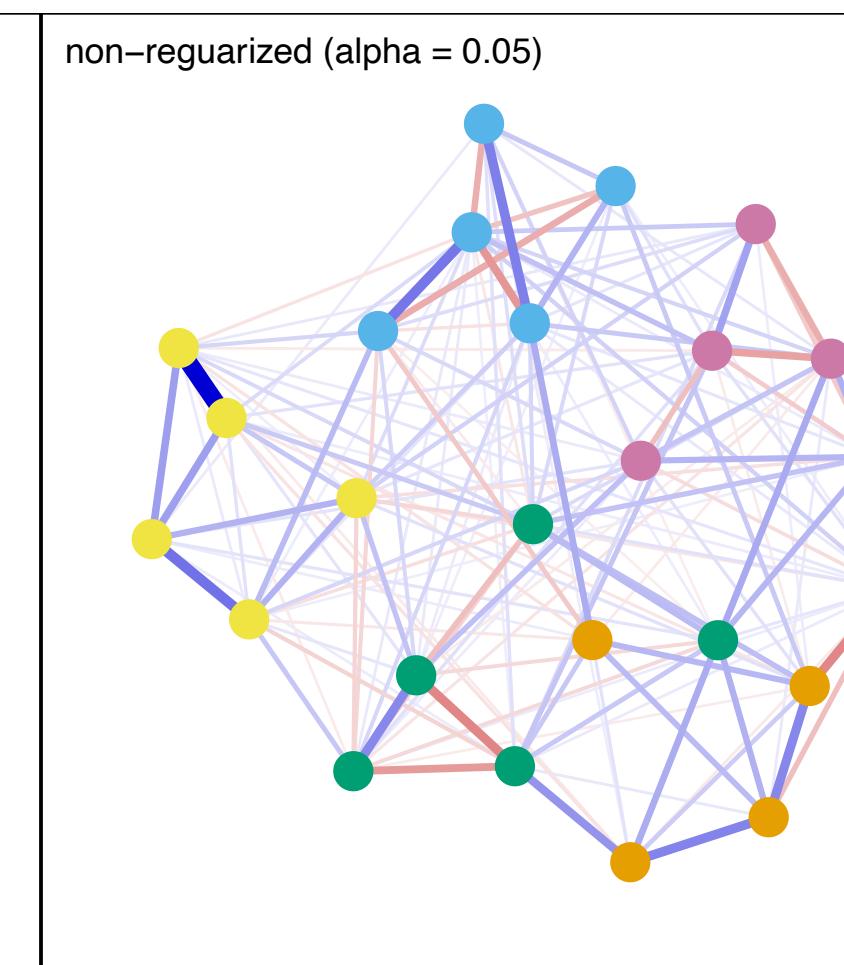
Regularization



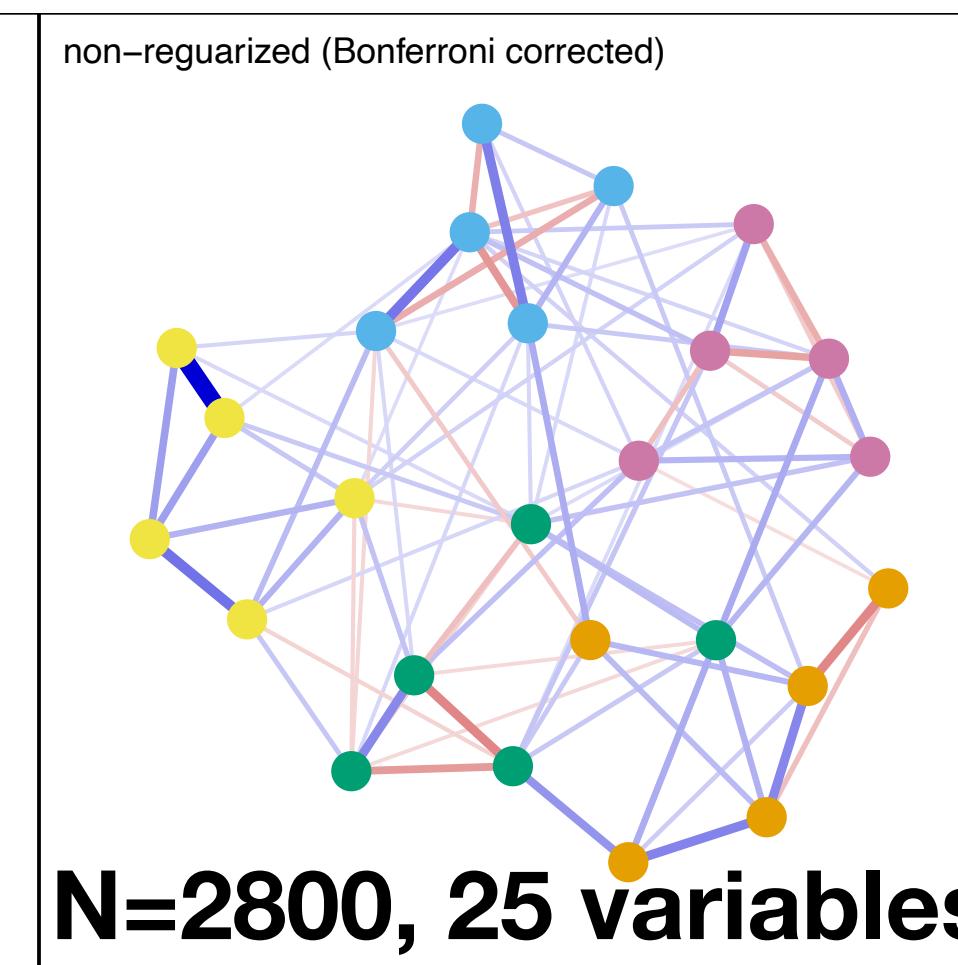
Model selection



Pruning (sig)



Pruning (bonferroni)

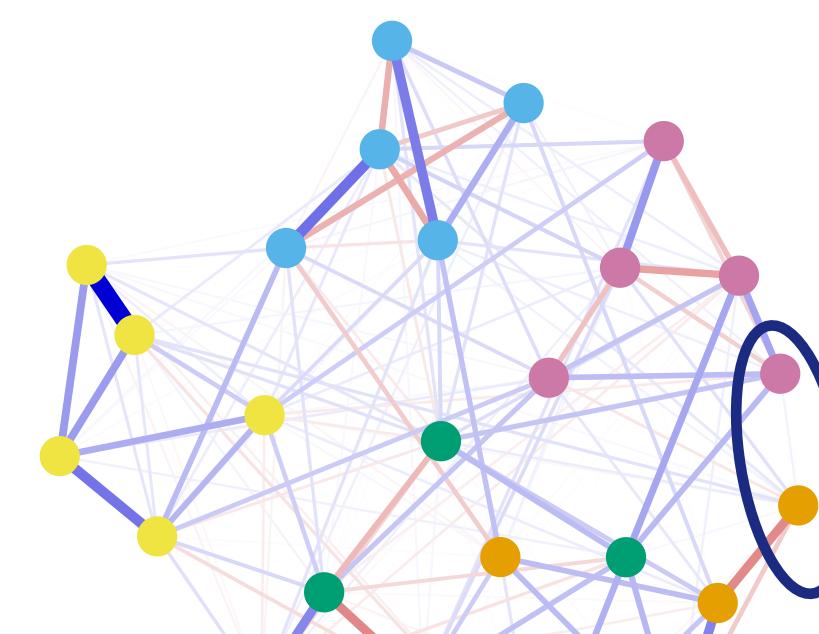


N=2800, 25 variables

What method to use?

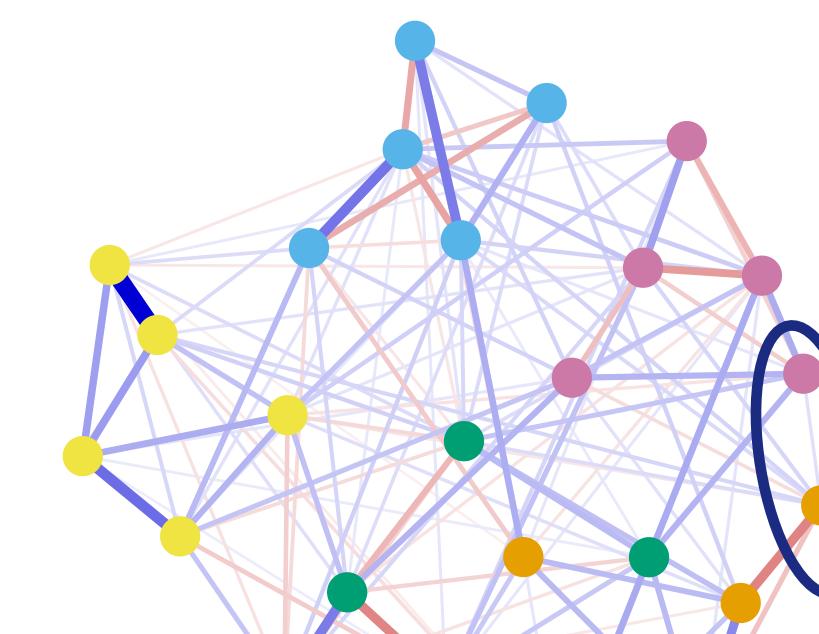
Regularization

EBICglasso (threshold = FALSE)



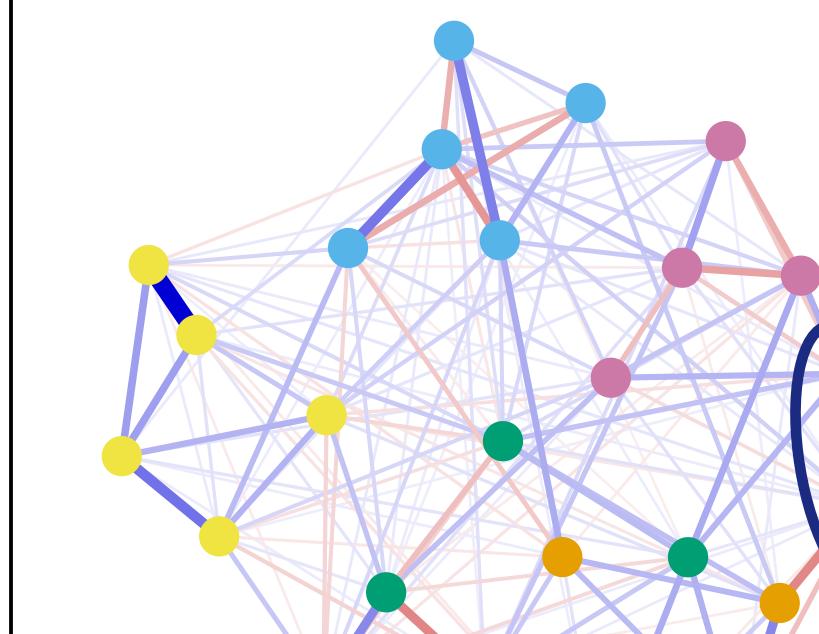
Model selection

gmmModSelect (stepwise = TRUE)



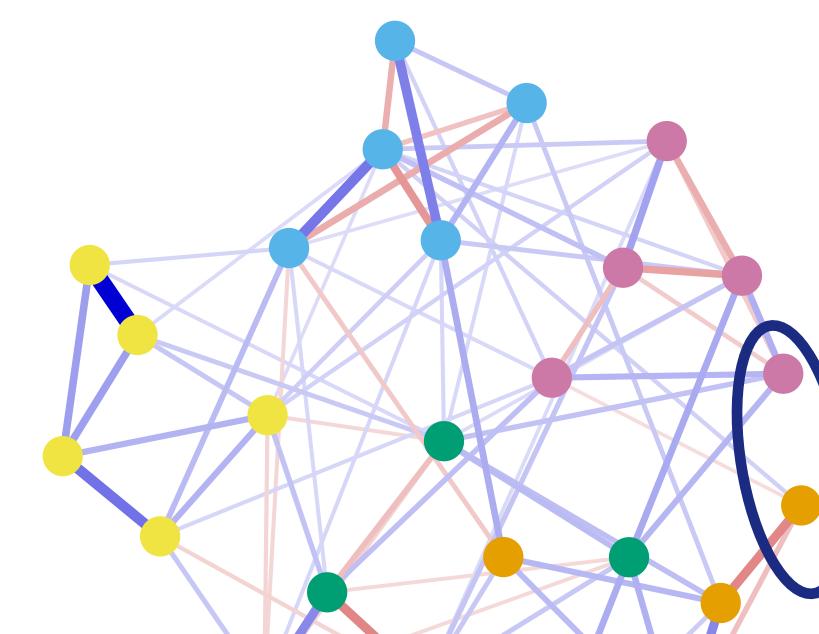
Pruning (sig)

non-regularized ($\alpha = 0.05$)



Pruning (bonferroni)

non-regularized (Bonferroni corrected)



N=2800, 25 variables

Psychological Medicine

cambridge.org/psm

Original Article

Cite this article: Isvoranu A-M, Guloksuz S, Epskamp S, van Os J, Borsboom D, GROUP Investigators (2019). Toward incorporating genetic risk scores into symptom networks of psychosis. *Psychological Medicine* 1–8. <https://doi.org/10.1017/S003329171900045X>

Received: 7 October 2018
Revised: 14 February 2019
Accepted: 18 February 2019

Key words:
GWAS; network analysis; polygenic risk score;
psychosis; schizophrenia

Toward incorporating genetic risk scores into symptom networks of psychosis

Adela-Maria Isvoranu¹, Sinan Guloksuz^{2,3}, Sacha Epskamp¹, Jim van Os⁴,
Denny Borsboom¹ and GROUP Investigators^{†*}

¹Department of Psychology, Psychological Methods, University of Amsterdam, Amsterdam, The Netherlands;

²Department of Psychiatry and Neuropsychology, School of Mental Health and Neuroscience, Maastricht University Medical Center, Maastricht, The Netherlands; ³Department of Psychiatry, Yale School of Medicine, New Haven, CT, USA and ⁴Utrecht University Medical Centre, Utrecht, The Netherlands

Abstract

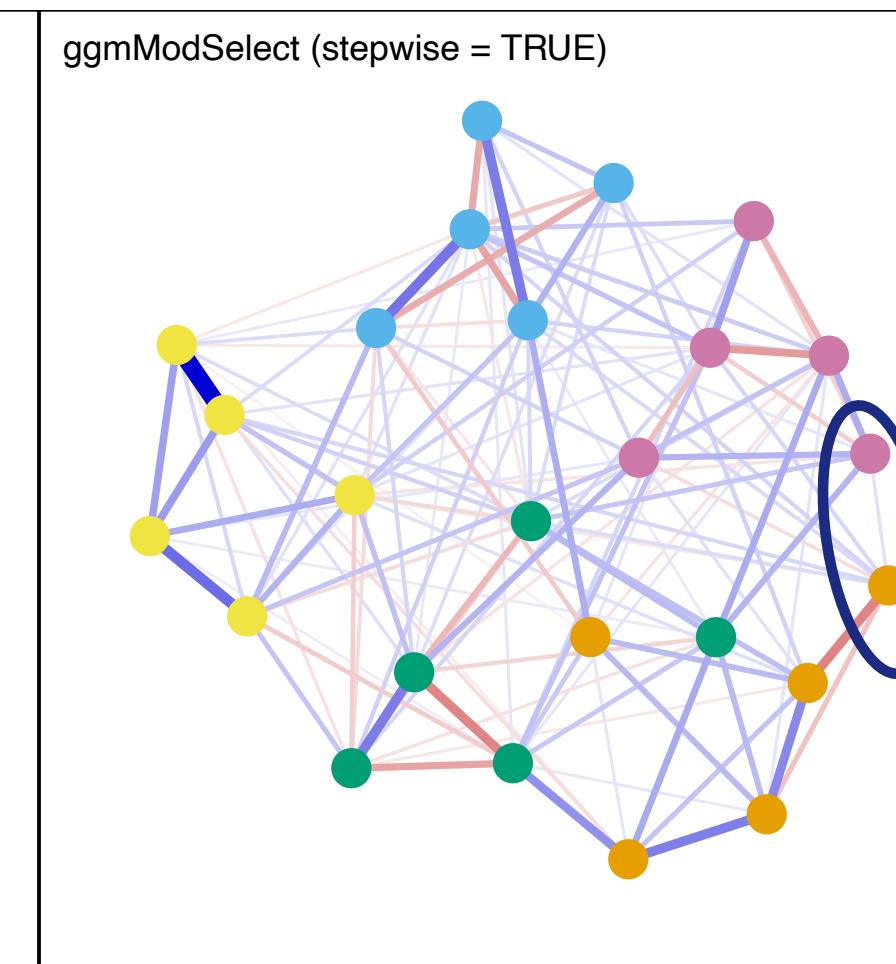
Background. Psychosis spectrum disorder is a heterogeneous, multifactorial clinical phenotype, known to have a high heritability, only a minor portion of which can be explained by molecular measures of genetic variation. This study proposes that the identification of genetic variation underlying psychotic disorder may have suffered due to issues in the psychometric conceptualization of the phenotype. Here we aim to open a new line of research into the genetics of mental disorders by explicitly incorporating genes into symptom networks.

What method to use?

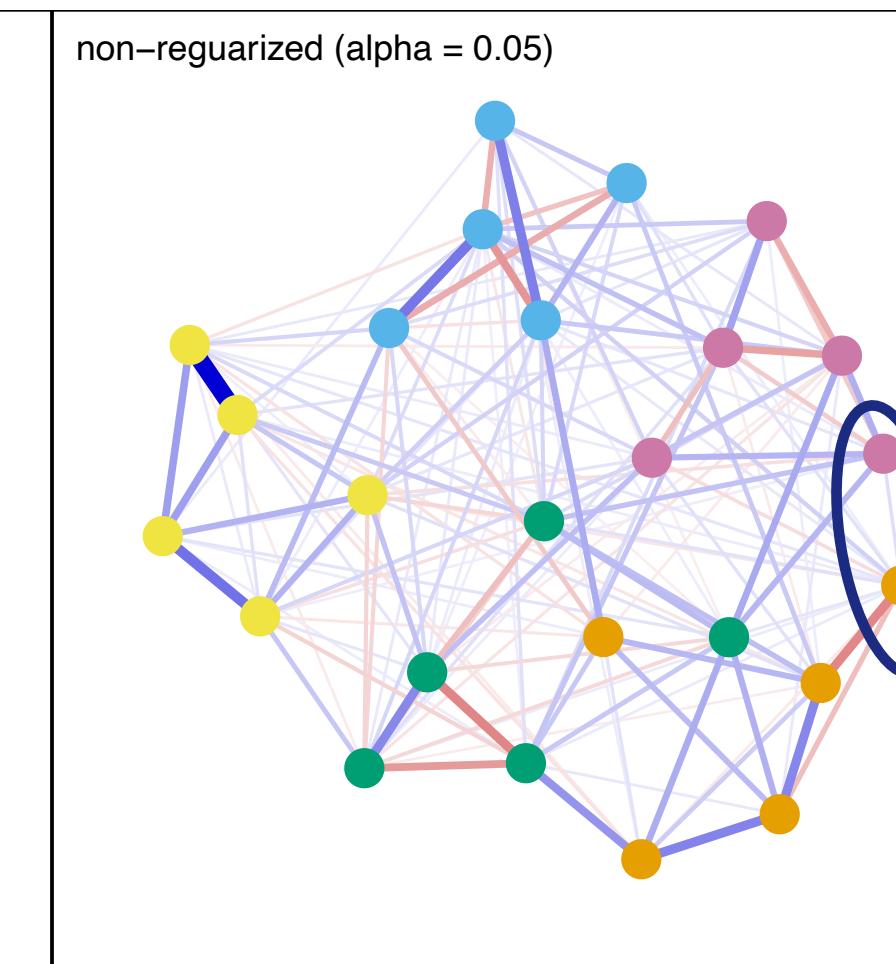
Regularization



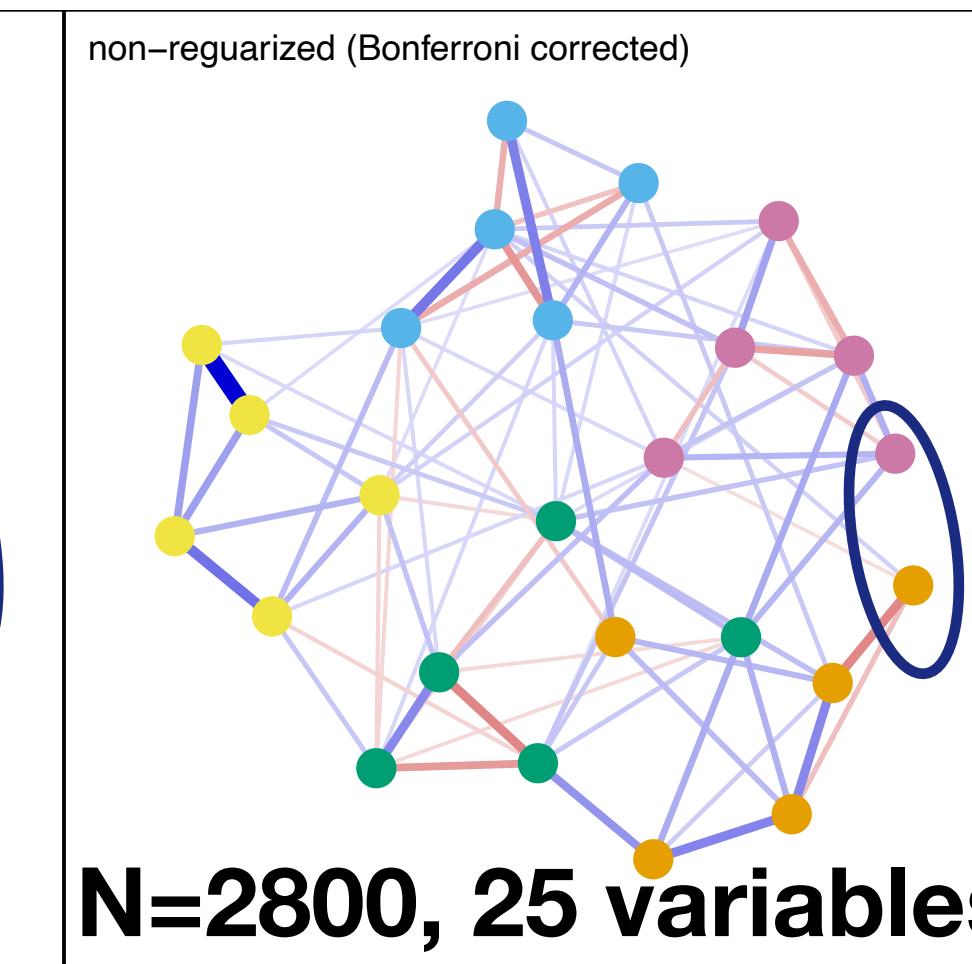
Model selection



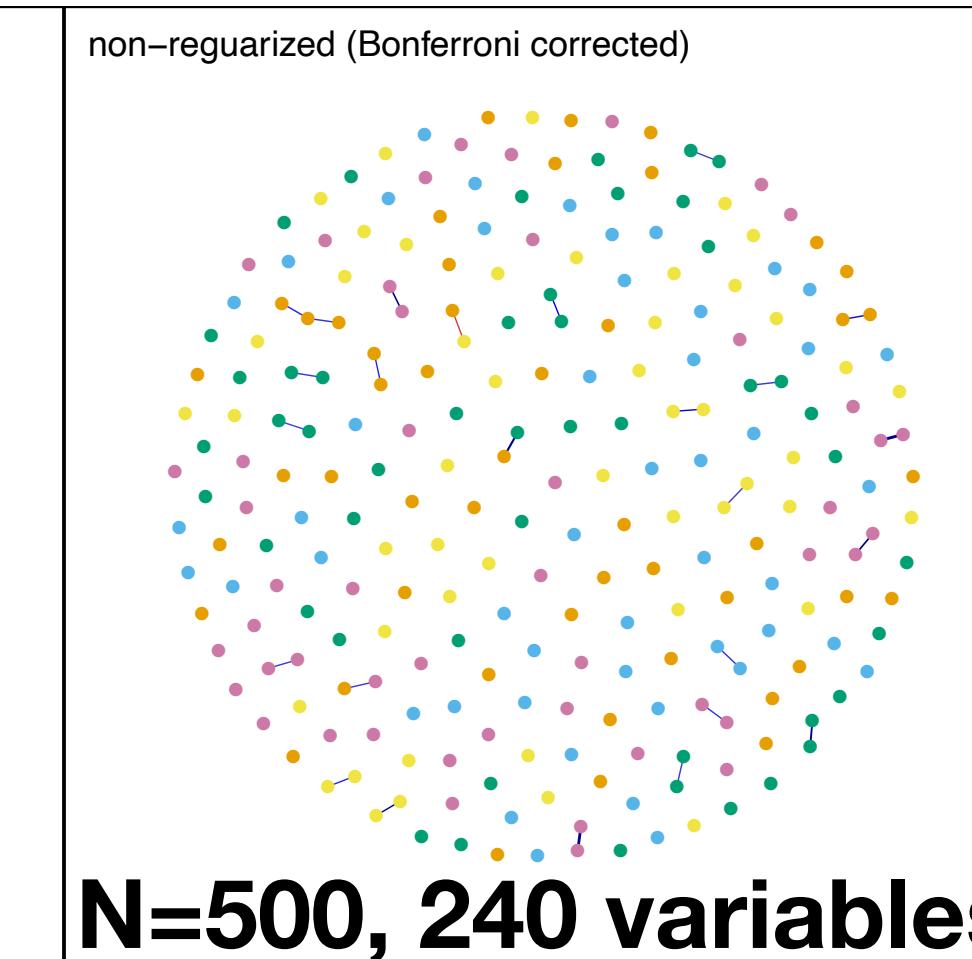
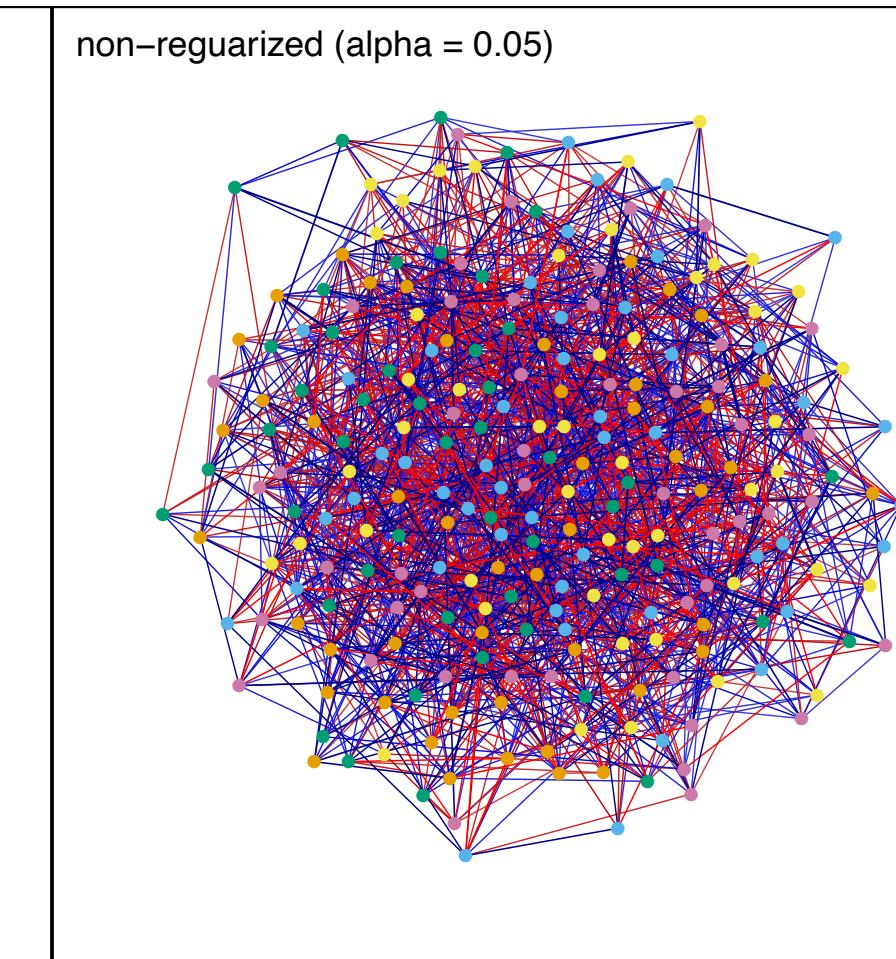
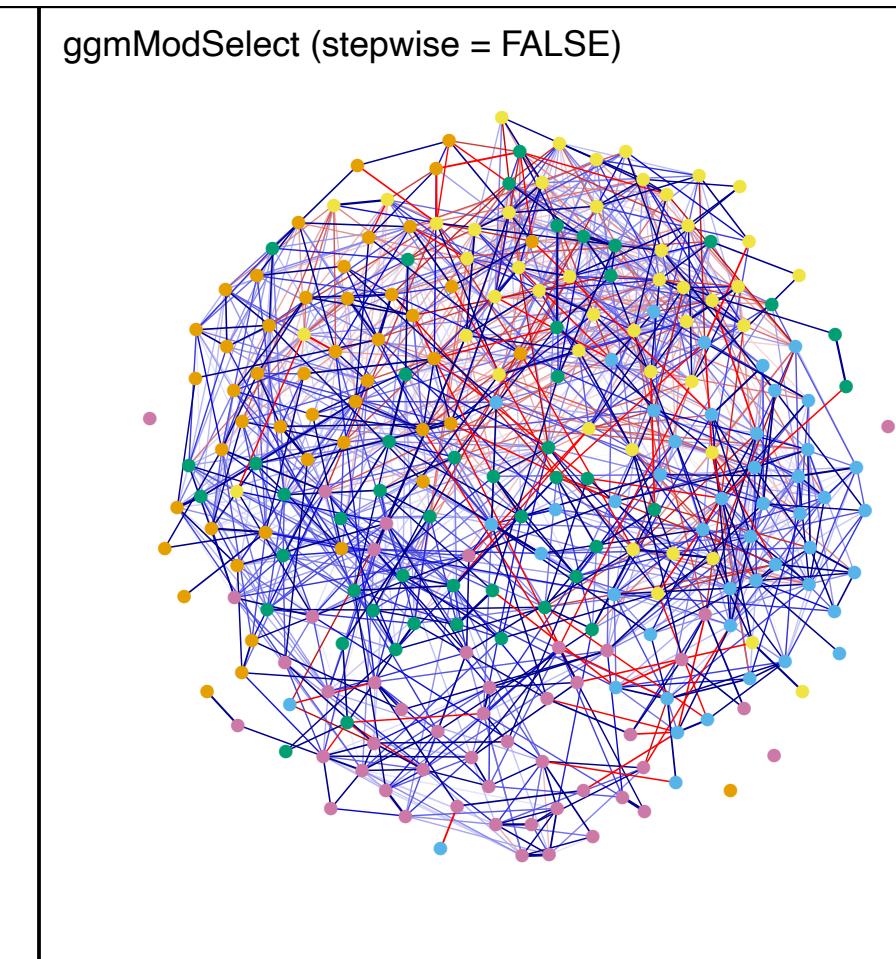
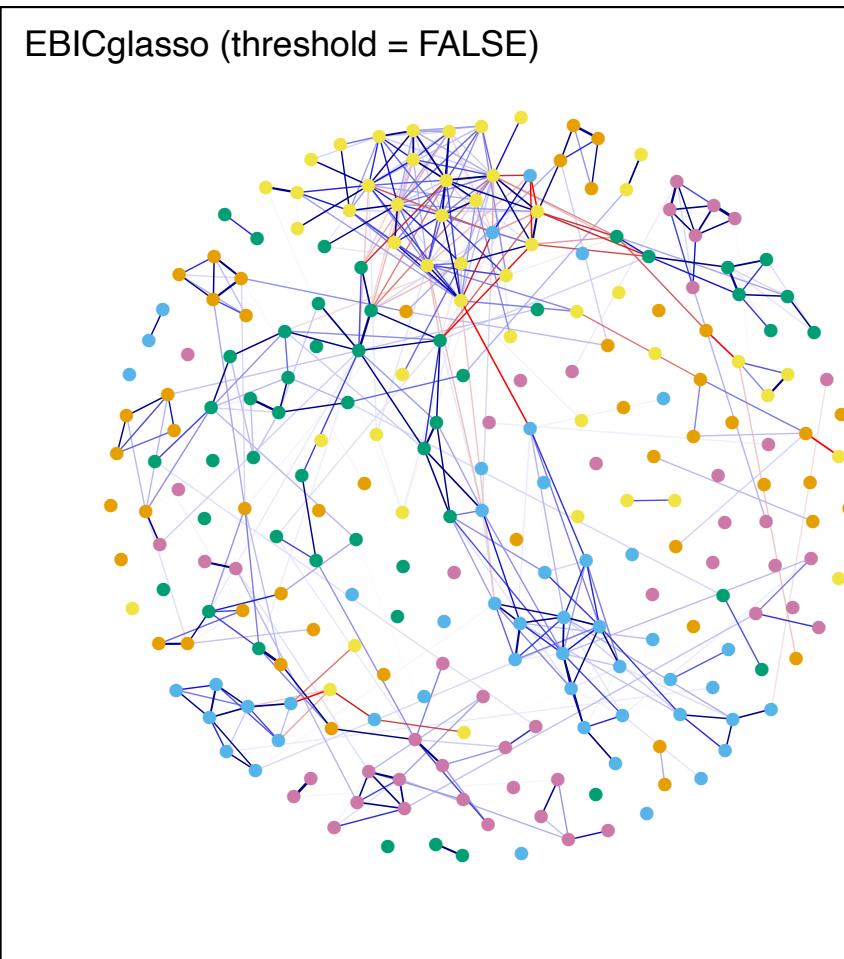
Pruning (sig)



Pruning (bonferroni)



N=2800, 25 variables



N=500, 240 variables

What methods to choose?

```
estimateNetwork(data = data,
                default = "...")
```

binary data

mixed data

Default	Model	Data	Estimation	Penalty	Selection	Package
pcor	GGM	Continuous, ordinal	Inverse	None	None, sig, FDR	qgraph
EBICglasso	GGM	Continuous, ordinal	Inverse	glasso	EBIC	qgraph
ggmModSelect	GGM	Continuous, ordinal	Stepwise model fit	None	BIC	qgraph
IsingFit	Ising	Binary	Nodewise	LASSO	EBIC	IsingFit
IsingSampler	Ising	Binary	Nodewise, MLE	None	None	IsingSampler
adalasso	GGM	Continuous	Nodewise	Adaptive LASSO	CV	parcor
huge	GGM	Continuous	Inverse	glasso	EBIC	huge
mgm	MGM	Continuous, count, binary, categorical	Nodewise	LASSO	EBIC, CV	mgm
relimp	"GGM"	Continuous	Nodewise	None	Any other default	relimpo

CV = Cross validation; Inverse = Inverse covariance matrix; MLE = Maximum (pseudo) likelihood

What methods to choose?

```
estimateNetwork(data = data,
                default = "...")
```

Default	Model	Data	Estimation	Penalty	Selection	Package
pcor	GGM	Continuous, ordinal	Inverse	None	None, sig, FDR	qgraph
EBICglasso	GGM	Continuous, ordinal	Inverse	glasso	EBIC	qgraph
ggmModSelect	GGM	Continuous, ordinal	Stepwise model fit	None	BIC	qgraph
IsingFit	Ising	Binary	Nodewise	LASSO	EBIC	IsingFit
IsingSampler	Ising	Binary	Nodewise, MLE	None	None	IsingSampler
adalasso	GGM	Continuous	Nodewise	Adaptive LASSO	CV	parcor
huge	GGM	Continuous	Inverse	glasso	EBIC	huge
mgm	MGM	Continuous, count, binary, categorical	Nodewise	LASSO	EBIC, CV	mgm
relimp	"GGM"	Continuous	Nodewise	None	Any other default	relimpo

CV = Cross validation; Inverse = Inverse covariance matrix; MLE = Maximum (pseudo) likelihood

Interest in overall structure (not small edges)
small sample size

Interest in small edges
large sample size

What methods to choose?

Default	Model	Data	Estimation	Penalty	Selection	Package
pcor	GGM	Continuous, ordinal	Inverse	None	None, sig, FDR	qgraph
EBICglasso	GGM	Continuous, ordinal	Inverse	glasso	EBIC	qgraph
ggmModSelect	GGM	Continuous, ordinal	Stepwise model fit	N		
IsingFit	Ising	Binary	Nodewise	L		
IsingSampler	Ising	Binary	Nodewise, MLE	N		
adalasso	GGM	Continuous	Nodewise	A		
huge	GGM	Continuous	Inverse	g		
mgm	MGM	Continuous, count, binary, categorical	Nodewise	LASSO	EBIC, CV	mgm
relimp	“GGM”	Continuous	Nodewise	None	Any other default	relimpo



© 2021 American Psychological Association
ISSN: 1082-989X

Psychological Methods

<https://doi.org/10.1037/met0000439>

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

Adela-Maria Isvoranu¹ and Sacha Epskamp^{1, 2}

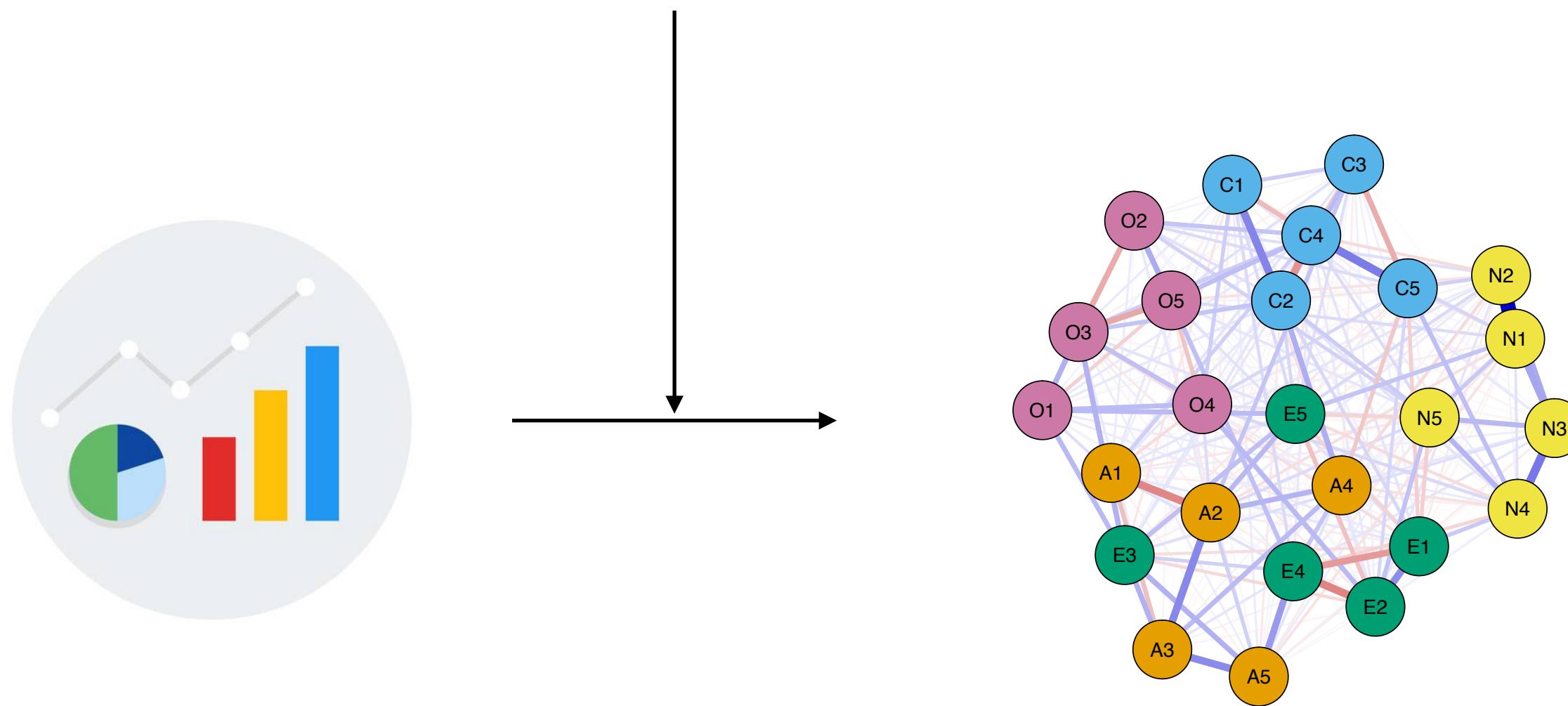
¹ Department of Psychology, Psychological Methods, University of Amsterdam

² Centre for Urban Mental Health, University of Amsterdam

CV = Cross validation; Inverse = Inverse covariance matrix; MLE = Maximum (pseudo) likelihood

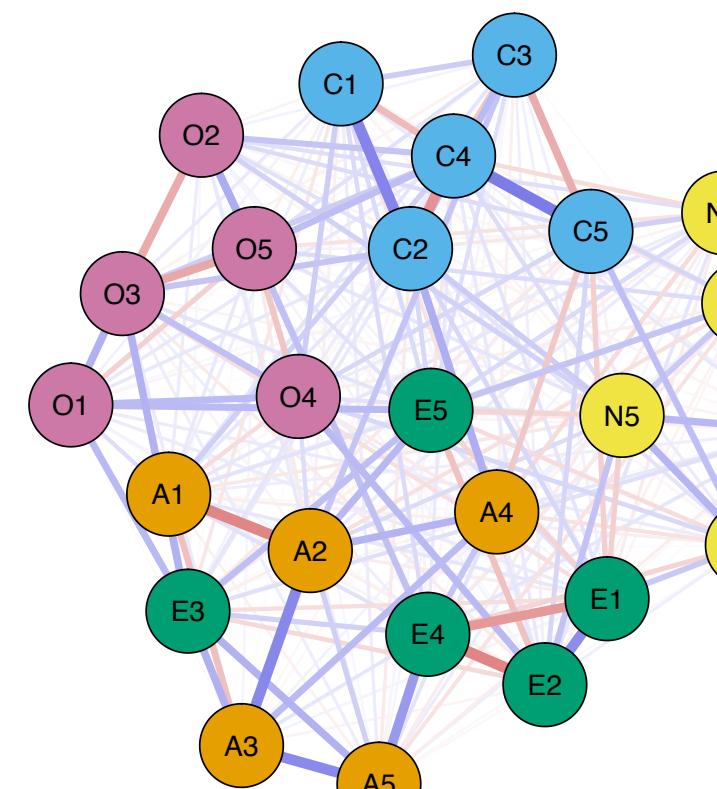
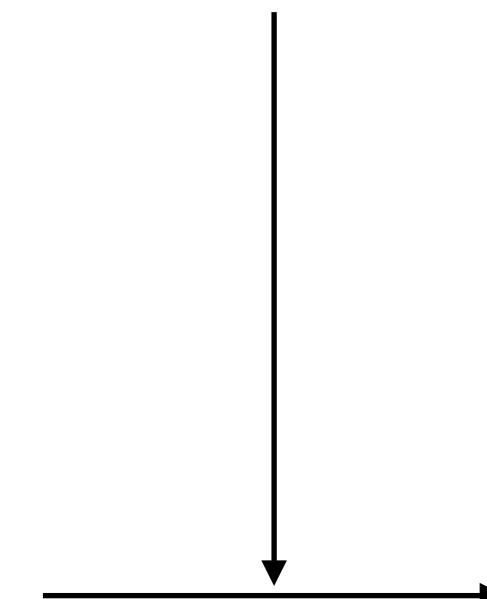
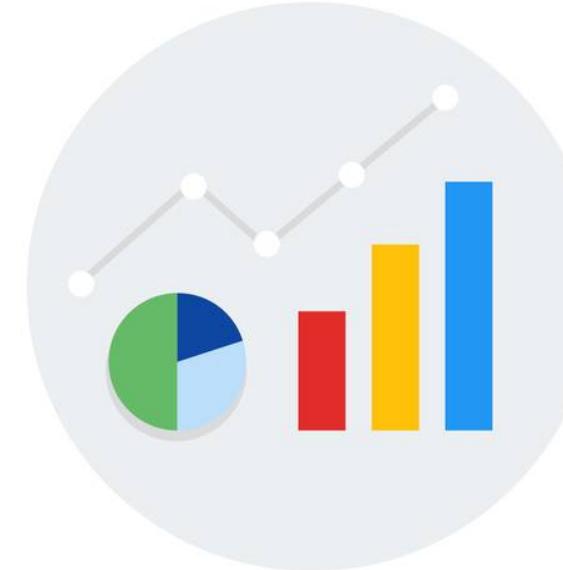
After model estimation

- How many observations?
- How many variables?
- What is the focus of my research question?



After model estimation

- How many observations?
- How many variables?
- What is the focus of my research question?



Toward incorporating genetic risk scores into symptom networks of psychosis

Adela-Maria Isvoranu¹, Sinan Guloksuz^{2,3}, Sacha Epskamp¹, Jim van Os⁴,
Denny Borsboom¹ and GROUP Investigators†*



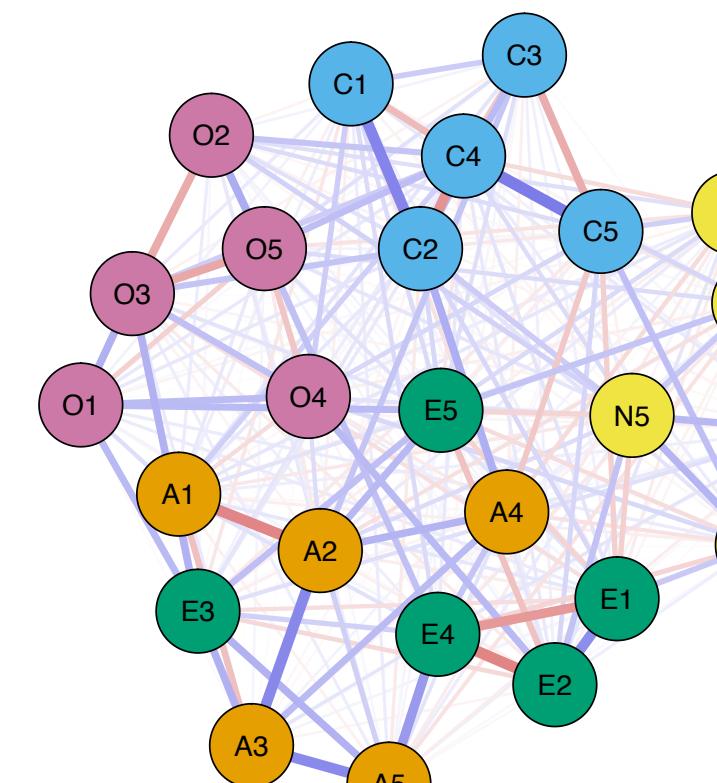
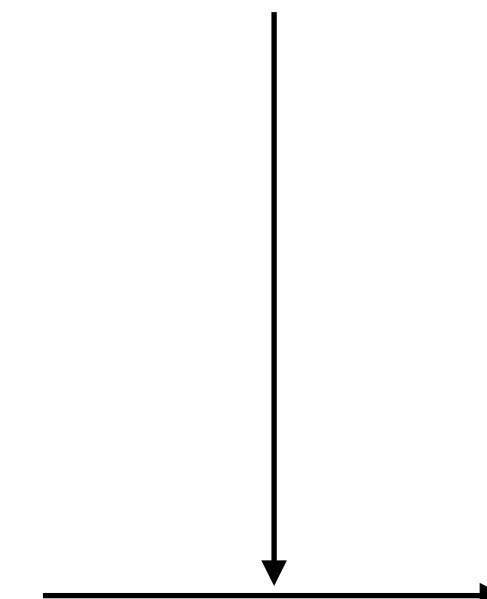
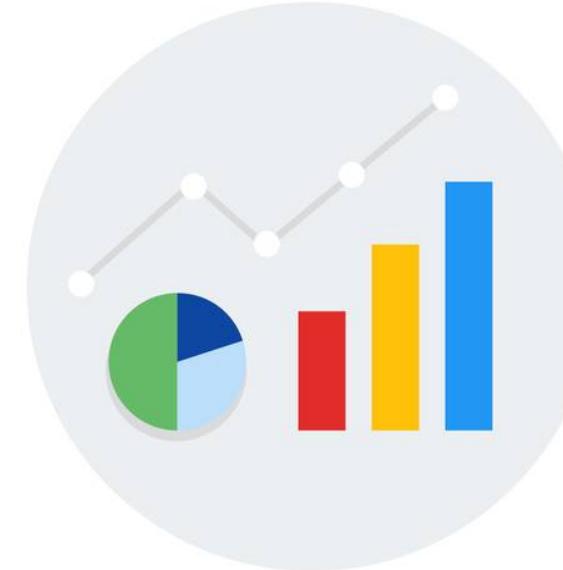
Original Investigation

Association of Symptom Network Structure With the Course of Depression

Claudia van Borkulo, MSc; Lynn Boschloo, PhD; Denny Borsboom, PhD; Brenda W. J. H. Penninx, PhD;
Lourens J. Waldorp, PhD; Robert A. Schoevers, MD, PhD

After model estimation

- How many observations?
- How many variables?
- What is the focus of my research question?



Toward incorporating genetic risk scores into symptom networks of psychosis

Adela-Maria Isvoranu¹, Sinan Guloksuz^{2,3}, Sacha Epskamp¹, Jim van Os⁴,
Denny Borsboom¹ and GROUP Investigators†*





Stability & replicability

Why is it important?

- Edges are estimated from data
- Uncertainty around estimates due to sampling variation

NEWS

Heart attack risk highest on Christmas Eve, study says

A study has singled out quite possibly the most dangerous holiday of the year: Christmas Eve. While the researchers couldn't explain why, they did have a hunch.

Sheila Jacobs
April 25, 2018

Healthy Diet Associated With Fewer Facial Wrinkles Among Women

Lifestyle Health & wellness Wellbeing

Smiling can influence our emotions to make us feel happy, study finds

Home > Exclusive > Mental Health

Eating more fruit and fewer savory snacks predicts better mental health, study finds

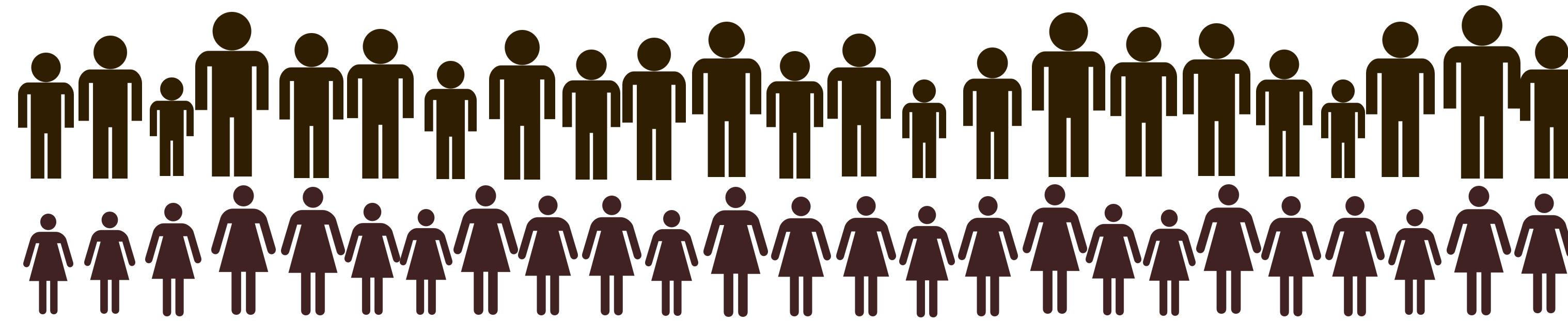
by Beth Ellwood — September 14, 2022 in Mental Health



accuracy & stability

part I: conceptually

Stability & power



- Are men taller than women?

Stability & power



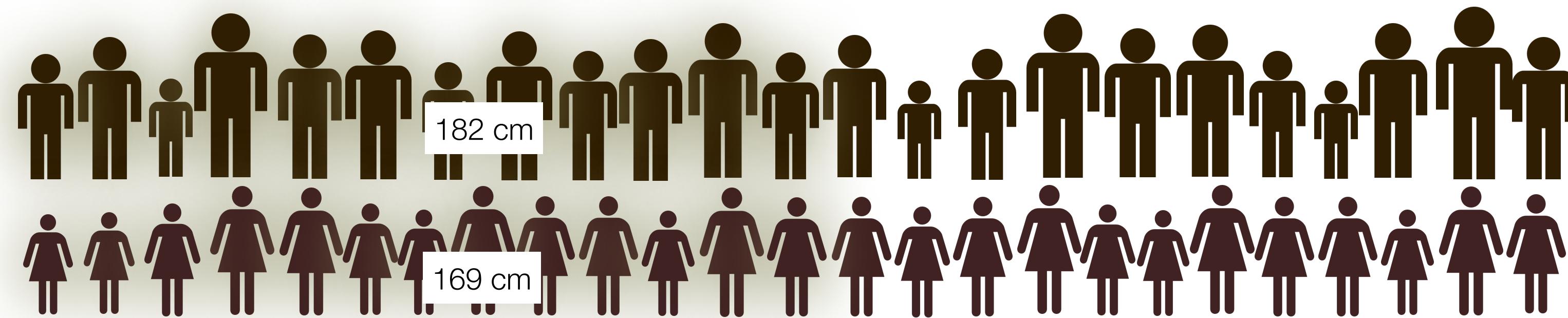
- Are men taller than women?
 - A small sample will result in a lot of *uncertainty* (high sampling variation, large confidence intervals)

Stability & power



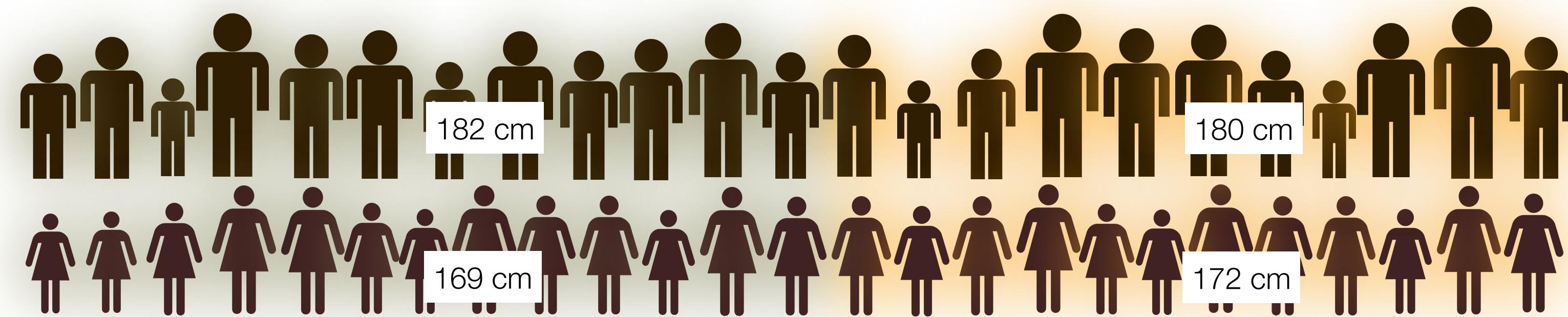
- Are men taller than women?
 - A small sample will result in a lot of *uncertainty* (high sampling variation, large confidence intervals)

Stability & power



- Are men taller than women?
 - A small sample will result in a lot of *uncertainty* (high sampling variation, large confidence intervals)
 - A large sample will reduce the uncertainty (smaller sampling variation, smaller confidence intervals), and will thereby *increase the accuracy* (model stability)

Stability & power

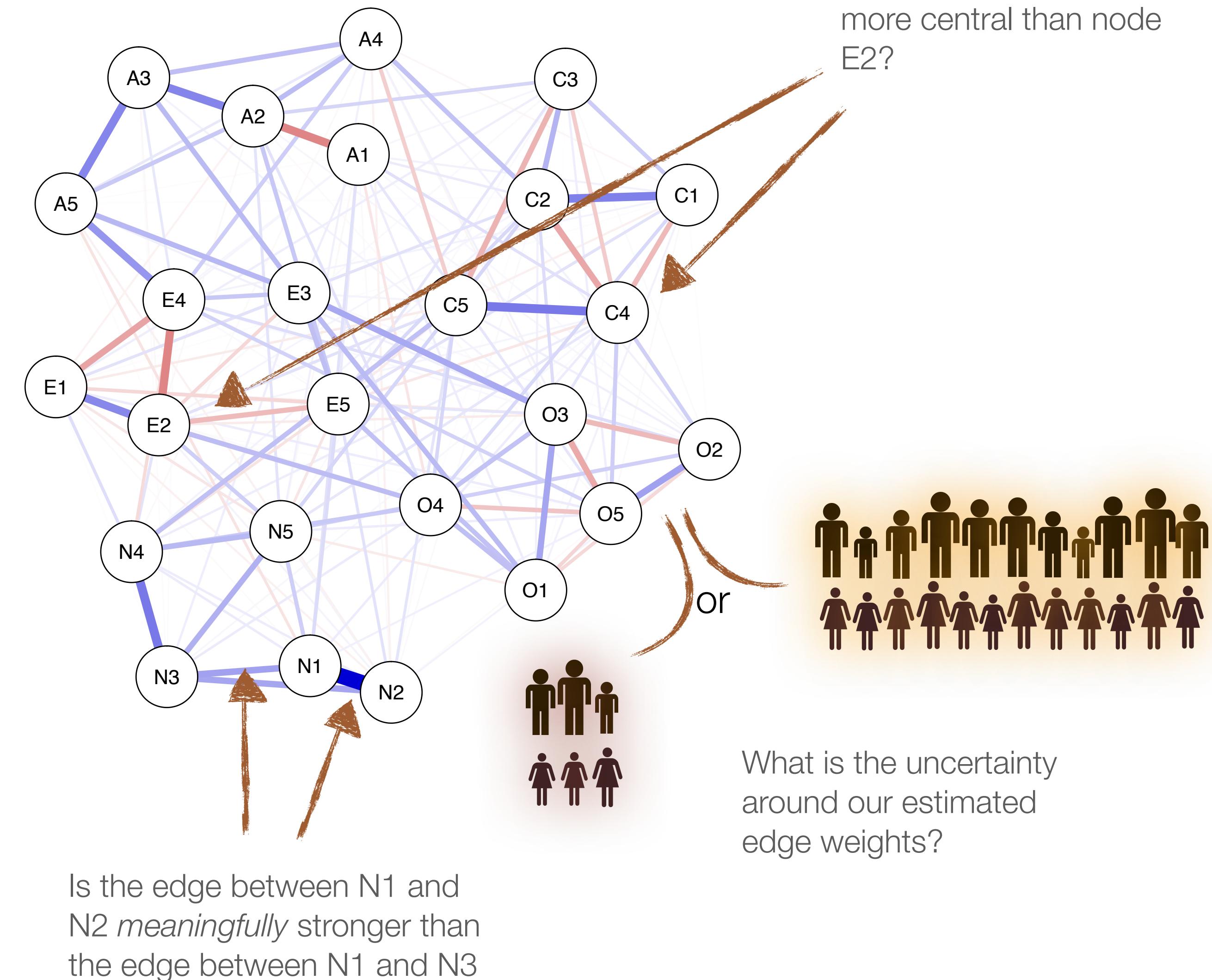


- Are men taller than women?
 - A small sample will result in a lot of *uncertainty* (high sampling variation, large confidence intervals)
 - A large sample will reduce the uncertainty (smaller sampling variation, smaller confidence intervals), and will thereby *increase the accuracy* (model stability)

Stability in networks

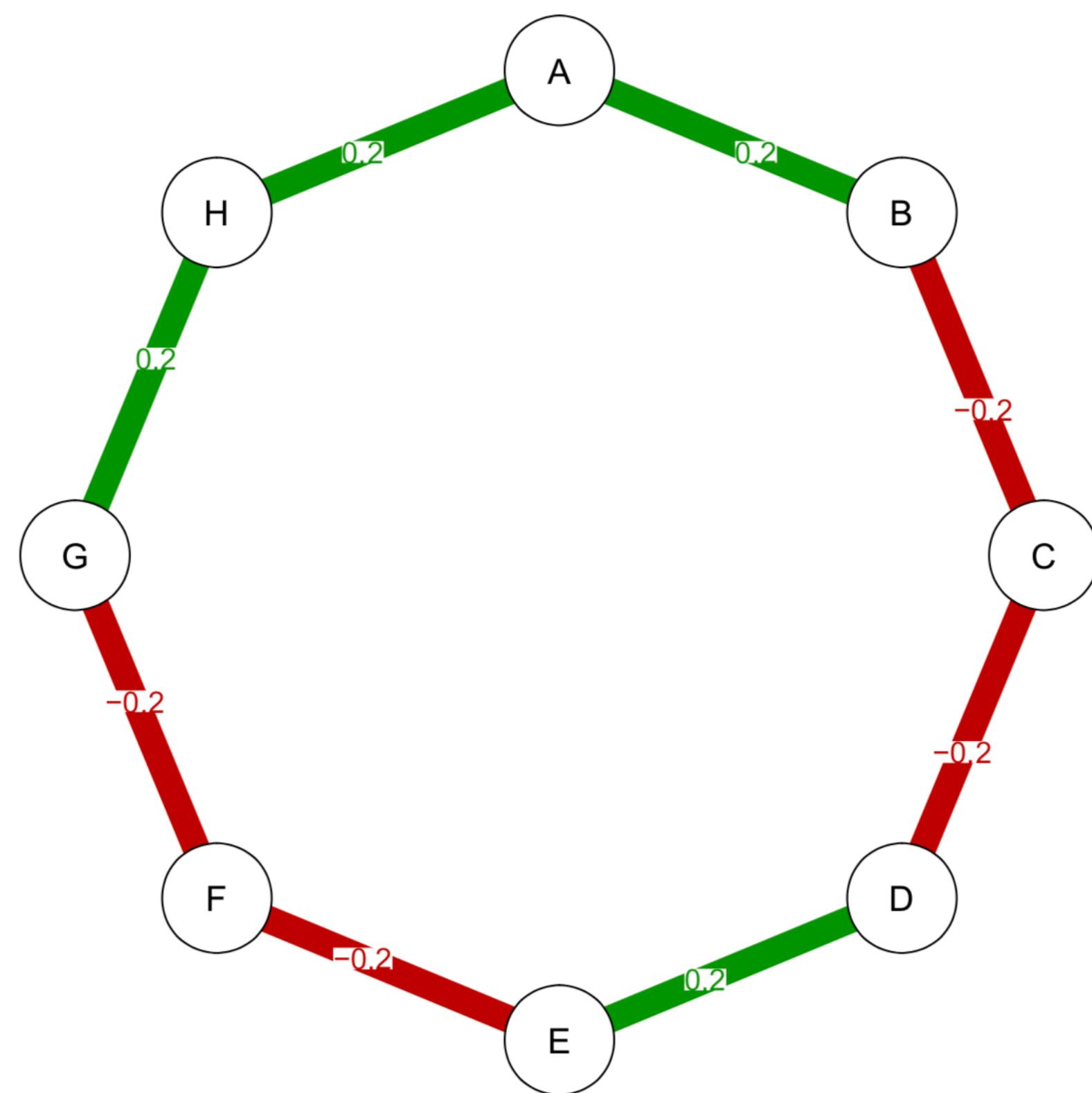
- Estimated network structures are subject to sampling variation
- How accurate and stable are our estimated parameters?

1. Stability of edge weights
2. Stability of network inferences
 - Centrality indices (e.g., strength)
 - Structure (e.g., density)

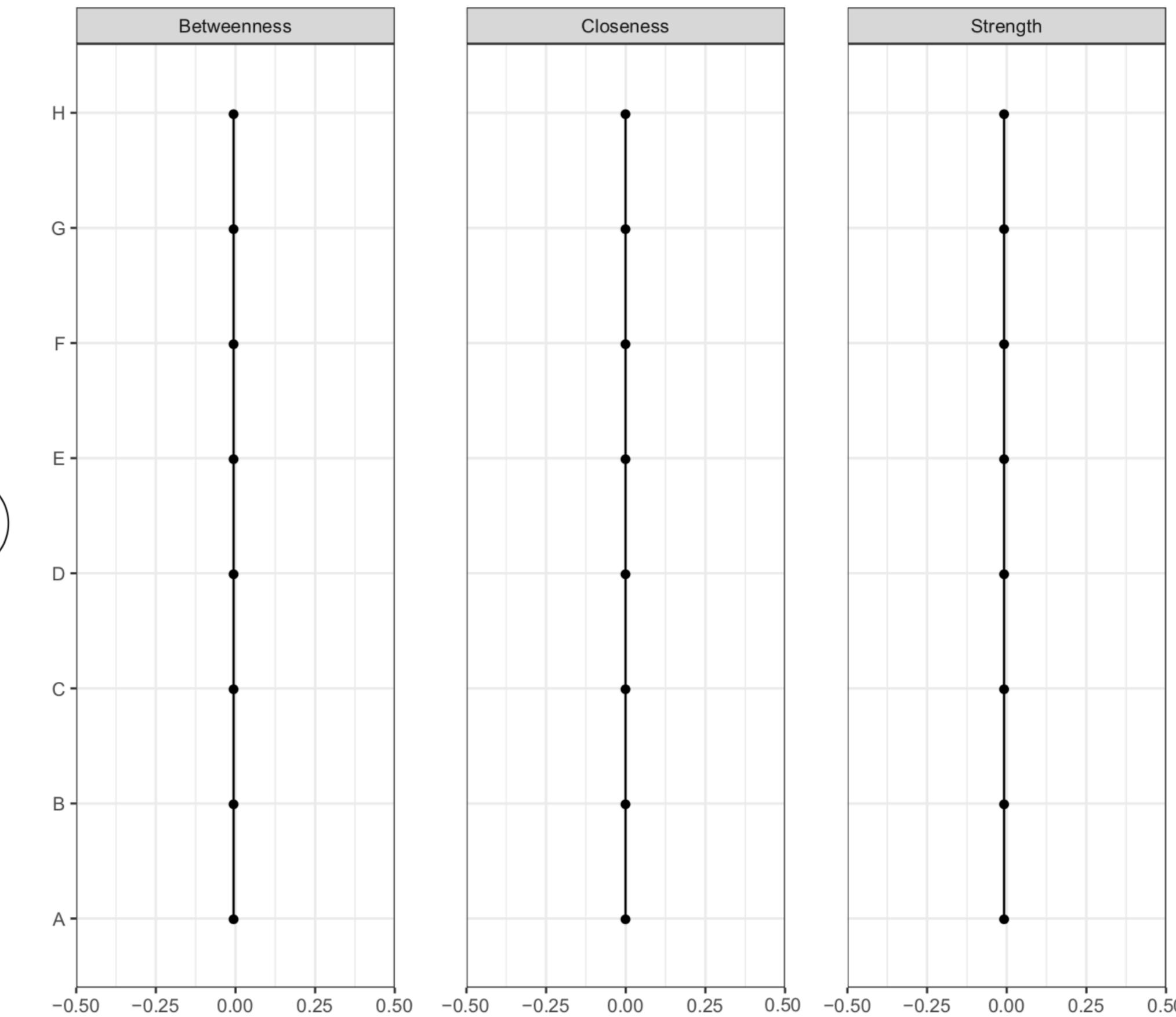


Simulation example: stability in networks

true network

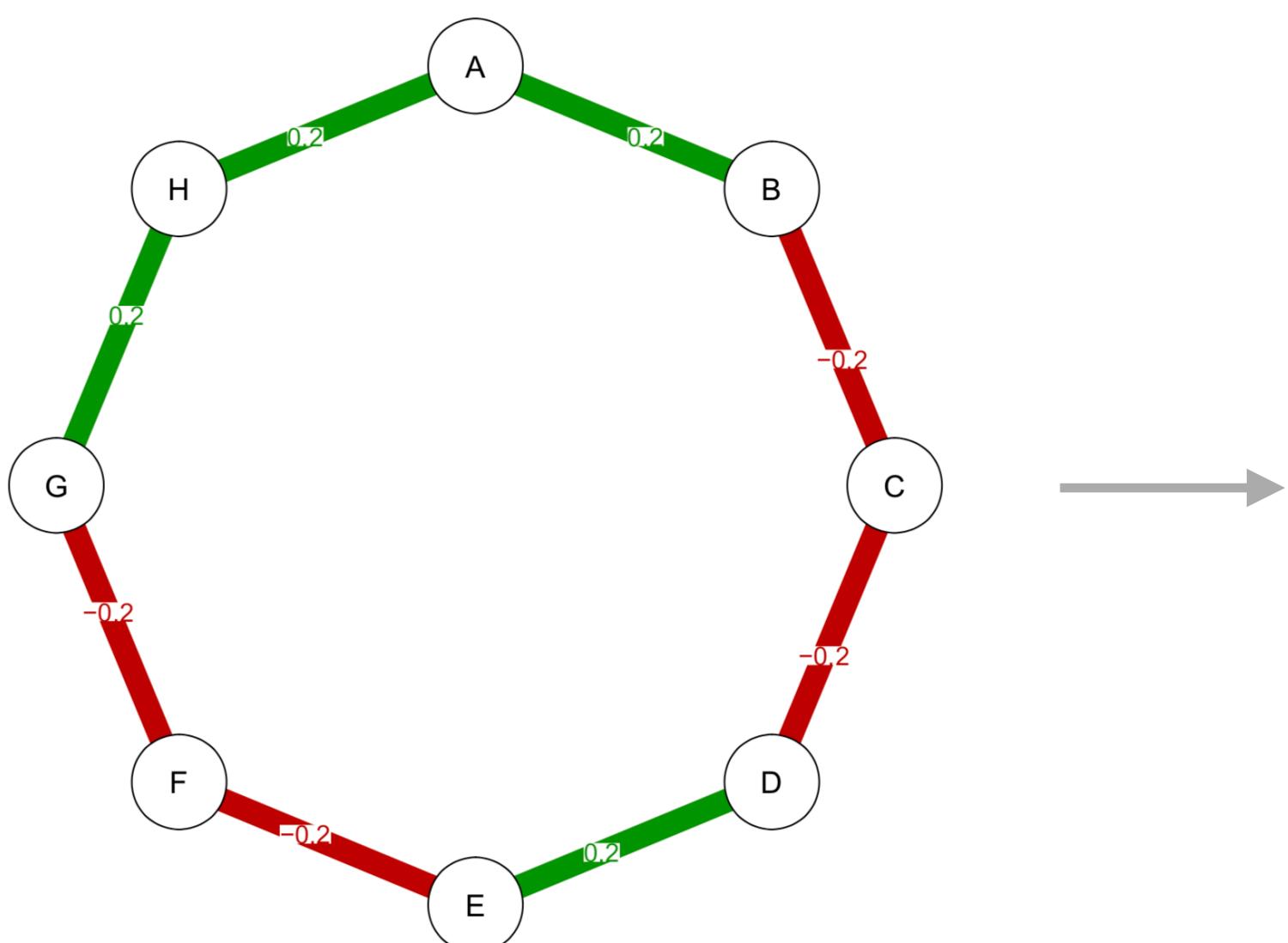


corresponding centrality



Simulation example: stability in networks

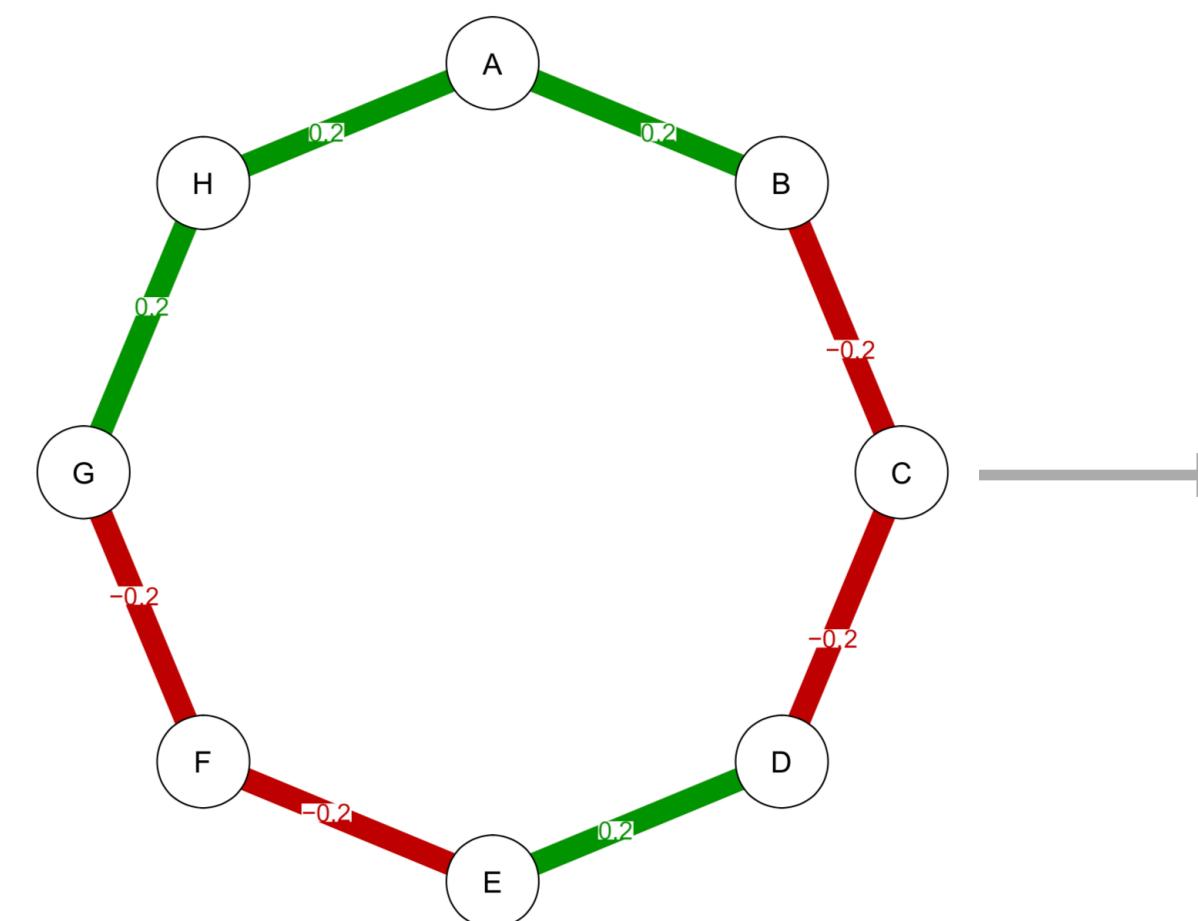
true network



simulated sample

Simulation example: stability in networks

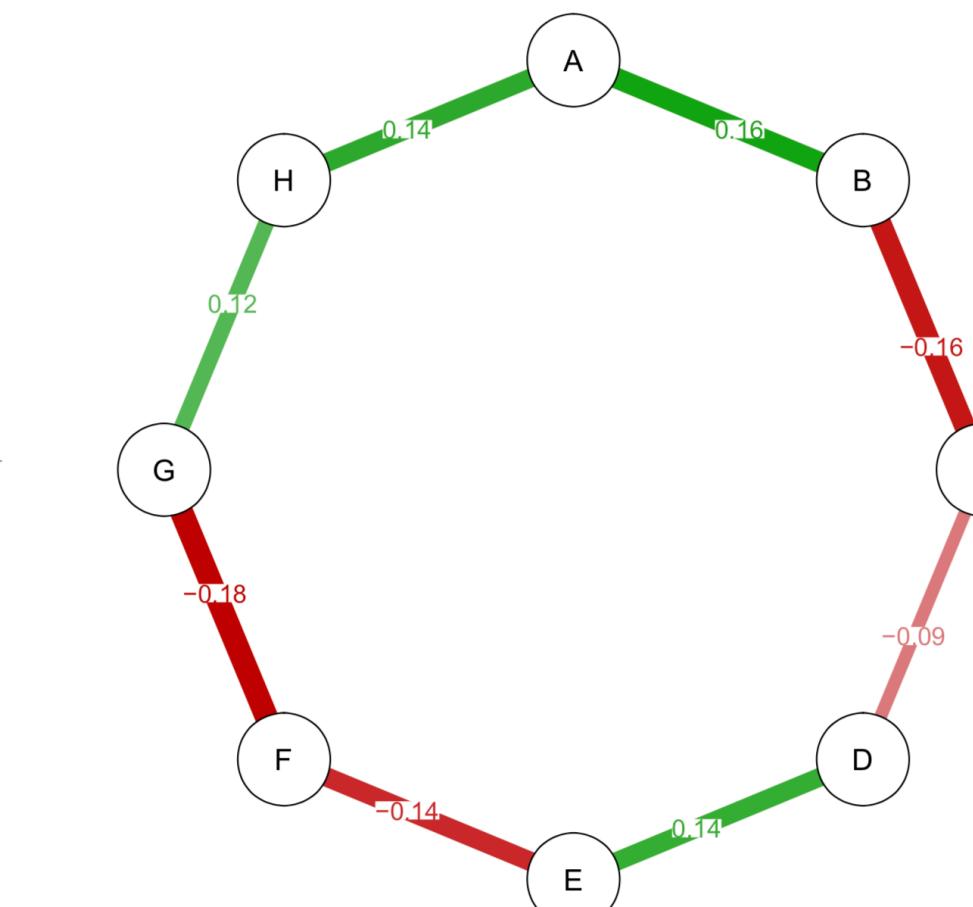
true network



simulated sample

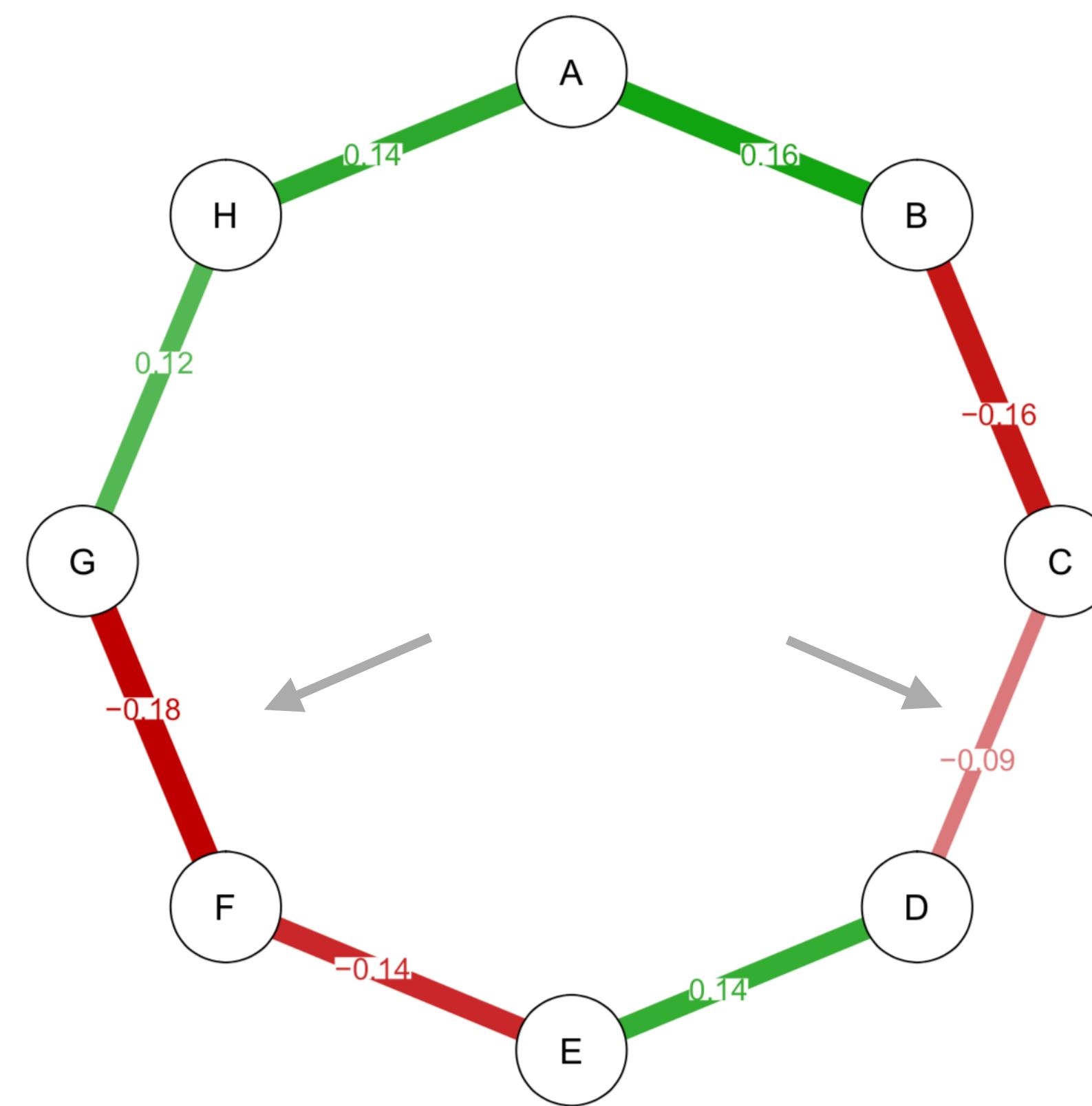
	A	B	C	D	E	F	G	H
1	●							
2	●							
3	●							
4	●							
...								
500								

estimated network

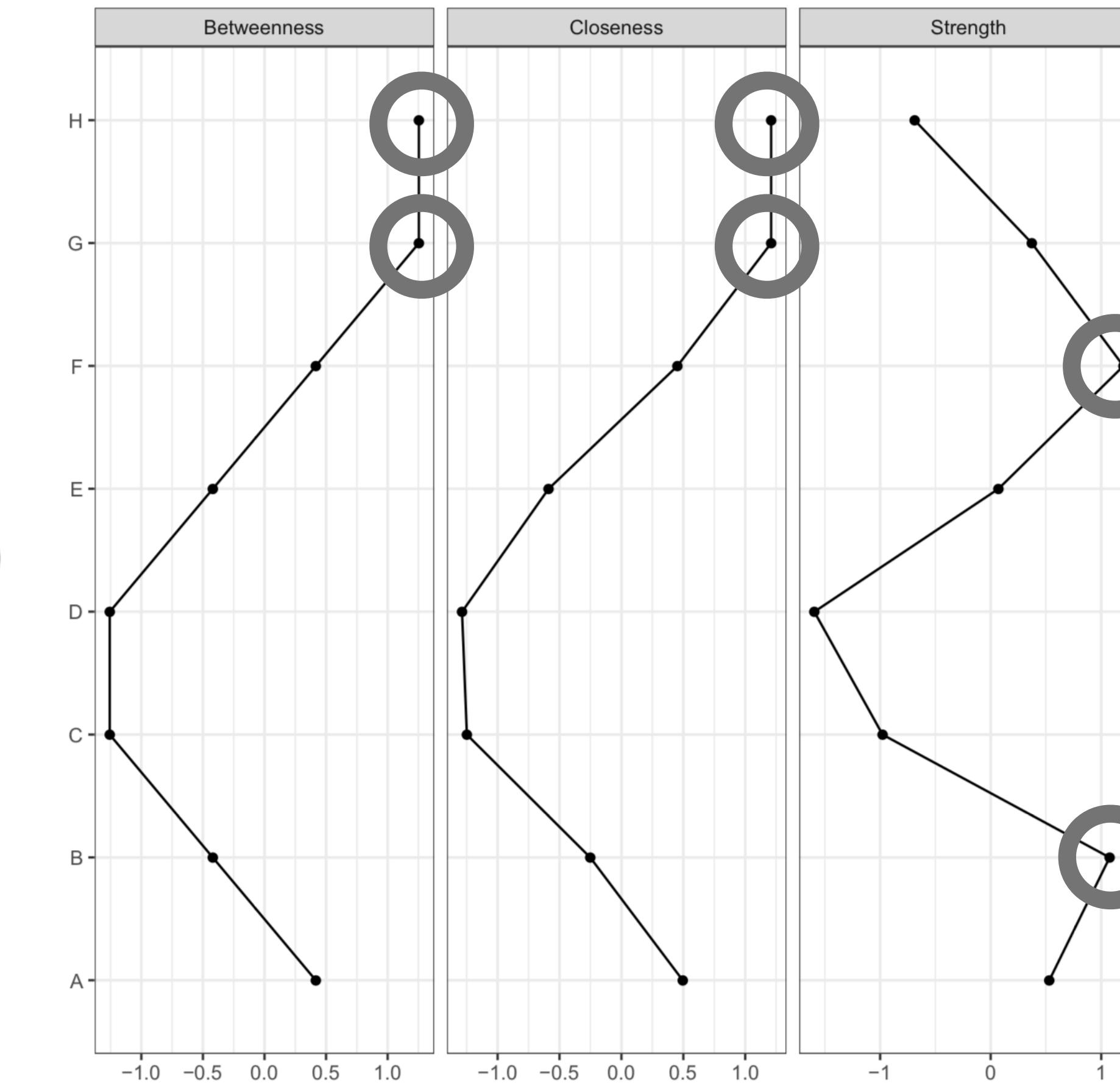


Simulation example: stability in networks

estimated network



corresponding centrality



Why is it important?

Clinical Psychological Science
Volume 8, Issue 1, January 2020, Pages 139-154
© The Author(s) 2019, Article Reuse Guidelines
<https://doi.org/10.1177/2167702619865958>



Empirical Article

Central Symptoms Predict Posttreatment Outcomes and Clinical Impairment in Anorexia Nervosa: A Network Analysis

Haley Elliott ¹, Payton J. Jones¹, and Ulrike Schmidt²



the british
psychological society
promoting excellence in psychology

British Journal of Clinical Psychology (2022), 61, 31–50
© 2021 The British Psychological Society
www.wileyonlinelibrary.com

Emotion network density is a potential clinical marker for anxiety and depression: Comparison of ecological momentary assessment and daily diary

Ki Eun Shin*¹ , Michelle G. Newman² and
Nicholas C. Jacobson³

¹Teachers College, Columbia University, New York, New York, USA
²The Pennsylvania State University, State College, Pennsylvania, USA
³Dartmouth College, Hanover, New Hampshire, USA

Original Investigation

Association of Symptom Network Structure With the Course of Depression

Claudia van Borkulo, MSc; Lynn Boschloo, PhD; Denny Borsboom, PhD; Brenda W. J. H. Penninx, PhD; Lourens J. Waldorp, PhD; Robert A. Schoevers, MD, PhD

Take home message part I

- Accuracy and stability are important...
 - ...if we want to make inferences about the network structure and centrality indices

Take home message part I

- Accuracy and stability are important...
 - ...if we want to make inferences about the network structure and centrality indices
 - ...for the networks to replicate

Take home message part I

- Accuracy and stability are important...
 - ...if we want to make inferences about the network structure and centrality indices
 - ...for the networks to replicate
 - ...in network estimation *just as in any other statistical method.*



accuracy & stability

part II: estimation

How to?

Behav Res (2018) 50:195–212
DOI 10.3758/s13428-017-0862-1



Estimating psychological networks and their accuracy: A tutorial paper

Sacha Epskamp¹ · Denny Borsboom¹ · Eiko I. Fried¹

Methodology to asses the accuracy of psychological networks, including three steps:

1. assess the *accuracy* of estimated network connections
2. investigate the *stability* of centrality indices
3. *test* whether network connections and centrality estimates for different variables differ from each other

How to?

Behav Res (2018) 50:195–212
DOI 10.3758/s13428-017-0862-1



Estimating psychological networks and their accuracy: A tutorial paper

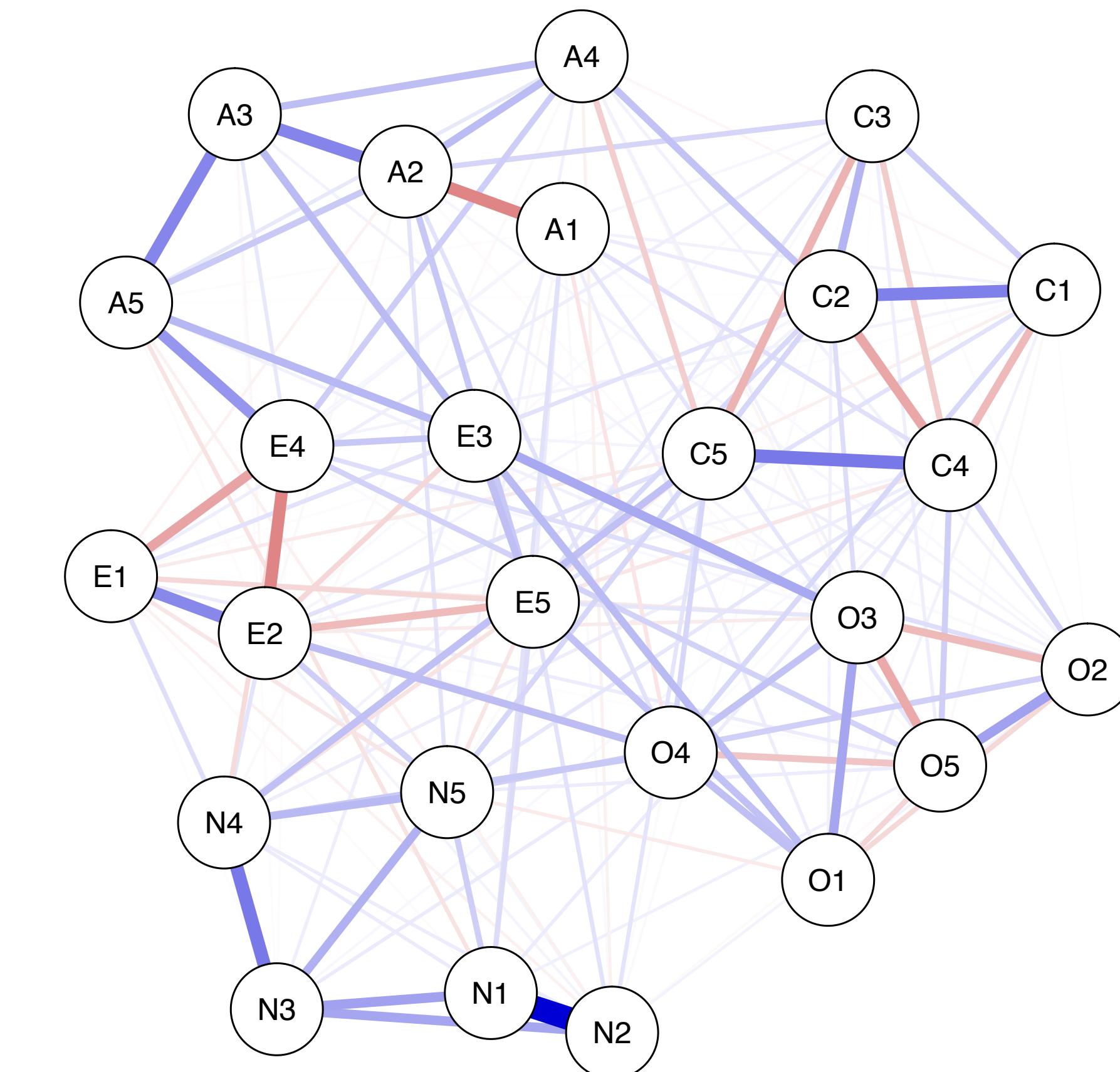
Sacha Epskamp¹ · Denny Borsboom¹ · Eiko I. Fried¹

Methodology to assess the accuracy of psychological networks, including three steps:

1. assess the *accuracy* of estimated network connections
2. investigate the *stability* of centrality indices
3. test whether network connections and centrality estimates for different variables differ from each other

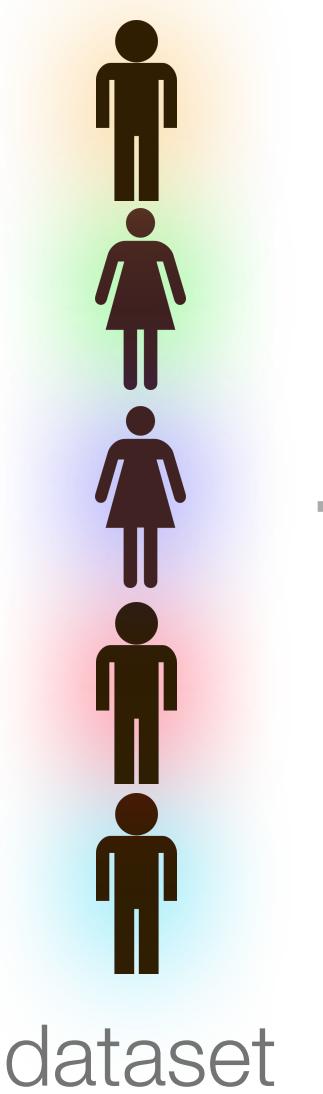
1. Stability of connections

- Obtain confidence interval around estimated edge weight using *non-parametric bootstrapping*



1. Stability of connections: non-parametric bootstrap

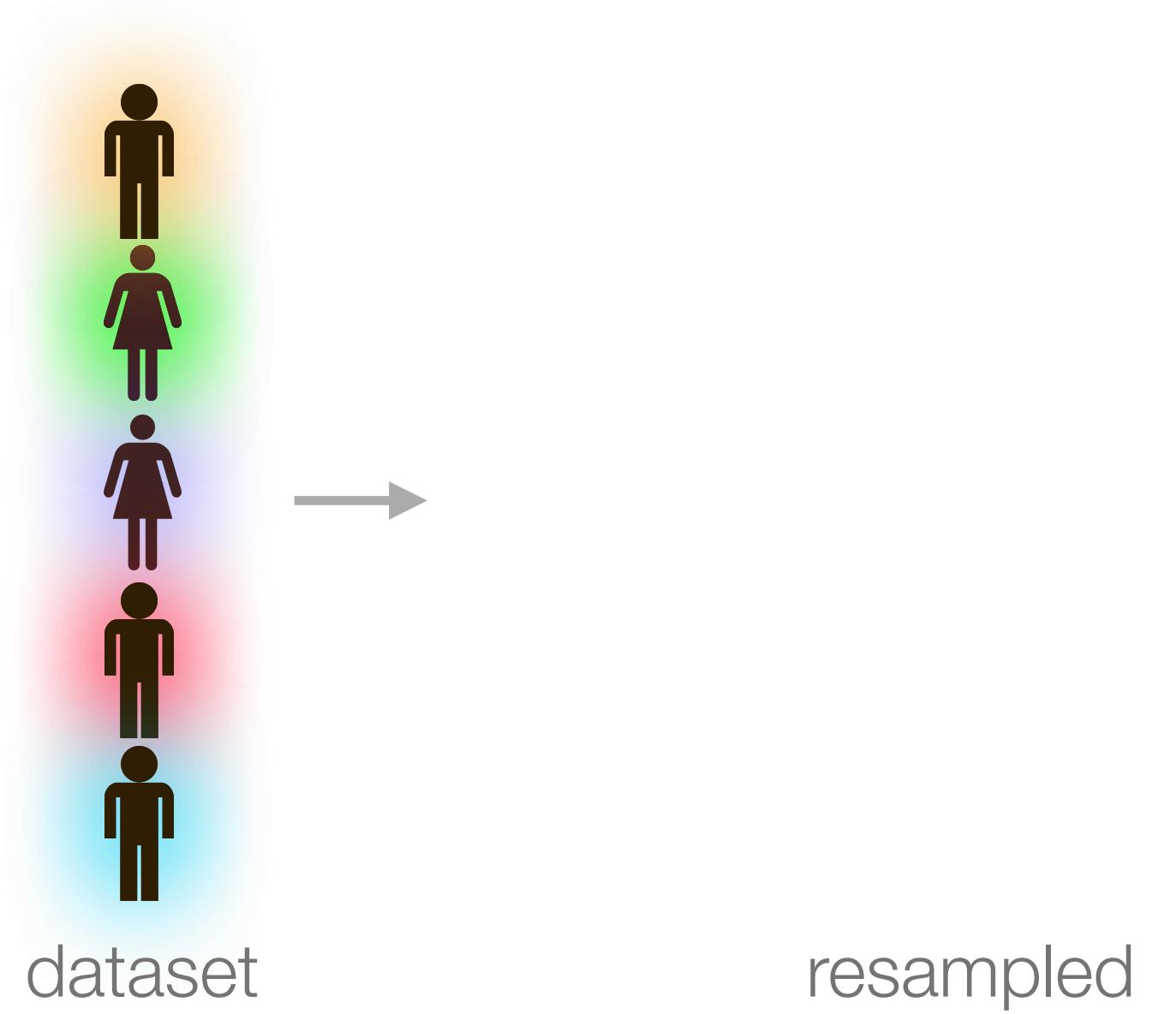
- Non-parametric bootstrap:
 1. compute statistic (e.g., edge weight) in original sample



dataset

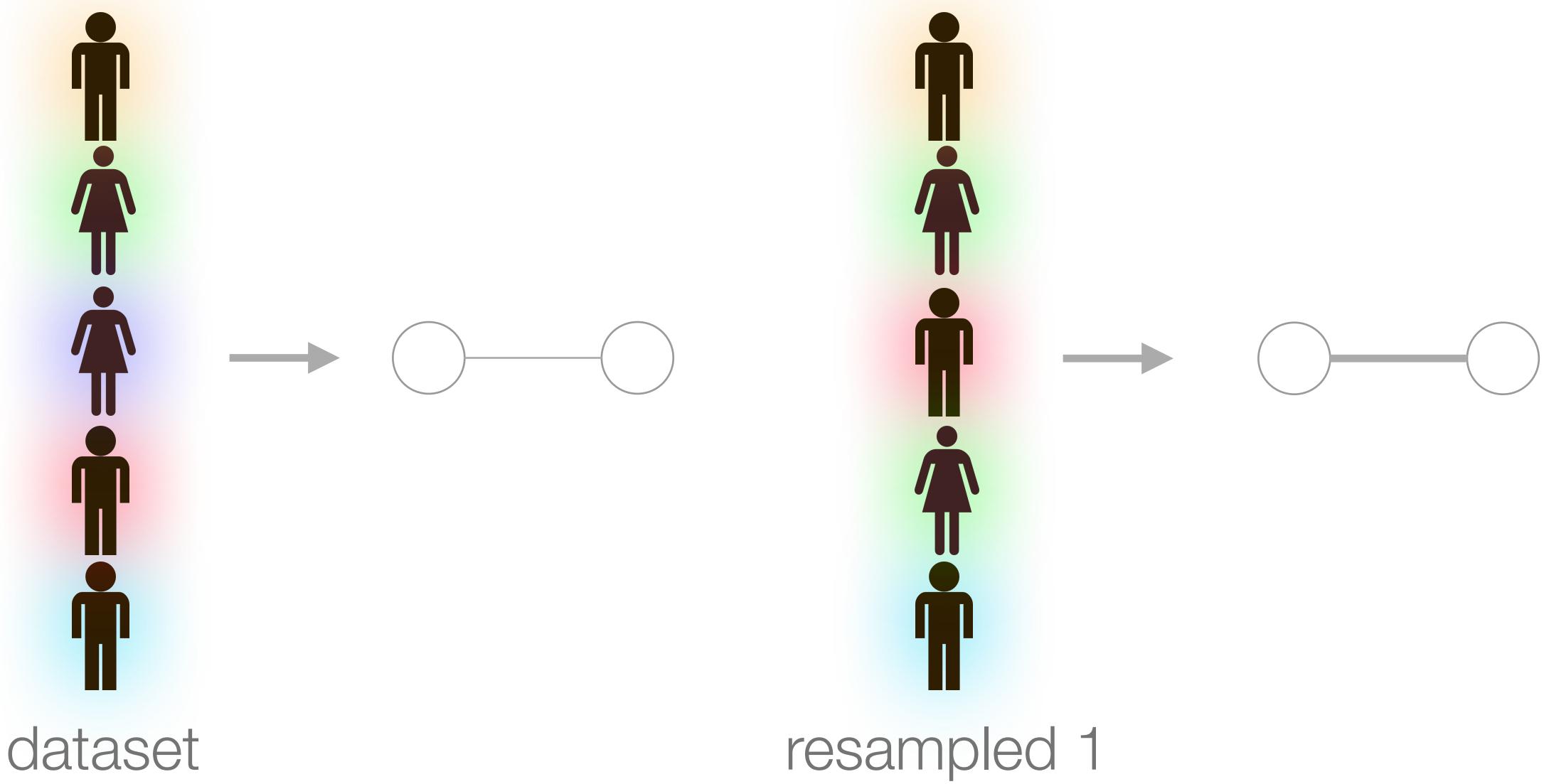
1. Stability of connections: non-parametric bootstrap

- Non-parametric bootstrap:
 1. compute statistic (e.g., edge weight) in original sample
 2. generate a new dataset by sampling from your original data with replacement



1. Stability of connections: non-parametric bootstrap

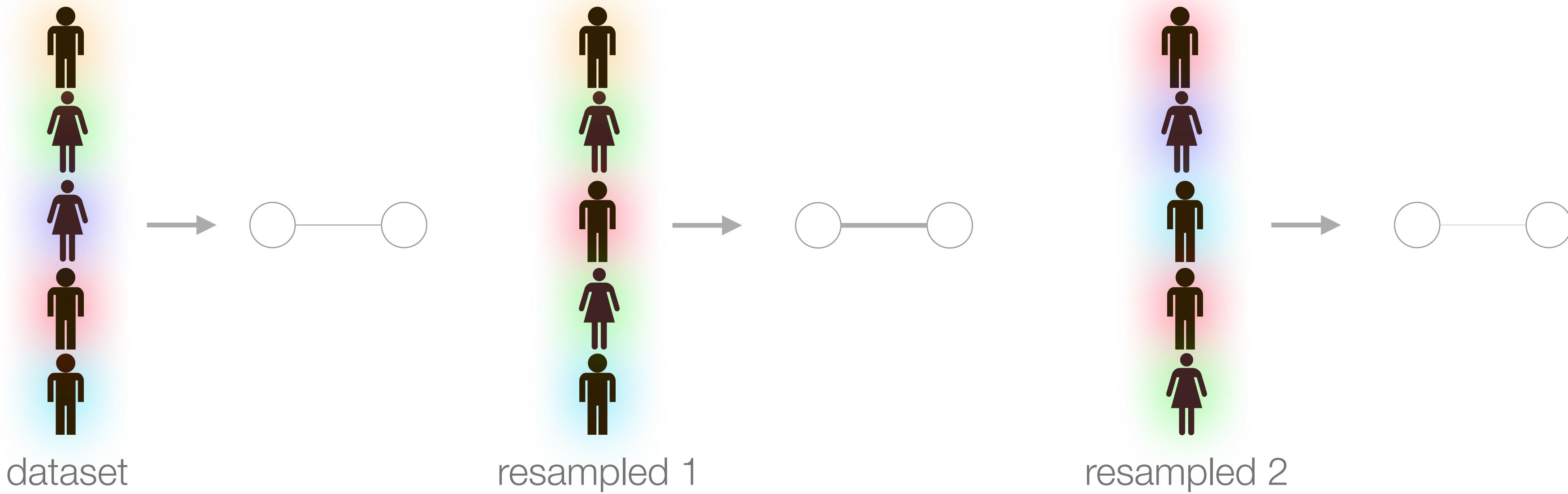
- Non-parametric bootstrap:
 1. compute statistic (e.g., edge weight) in original sample
 2. generate a new dataset by sampling from your original data with replacement
 3. compute statistic (e.g., edge weight) in new dataset



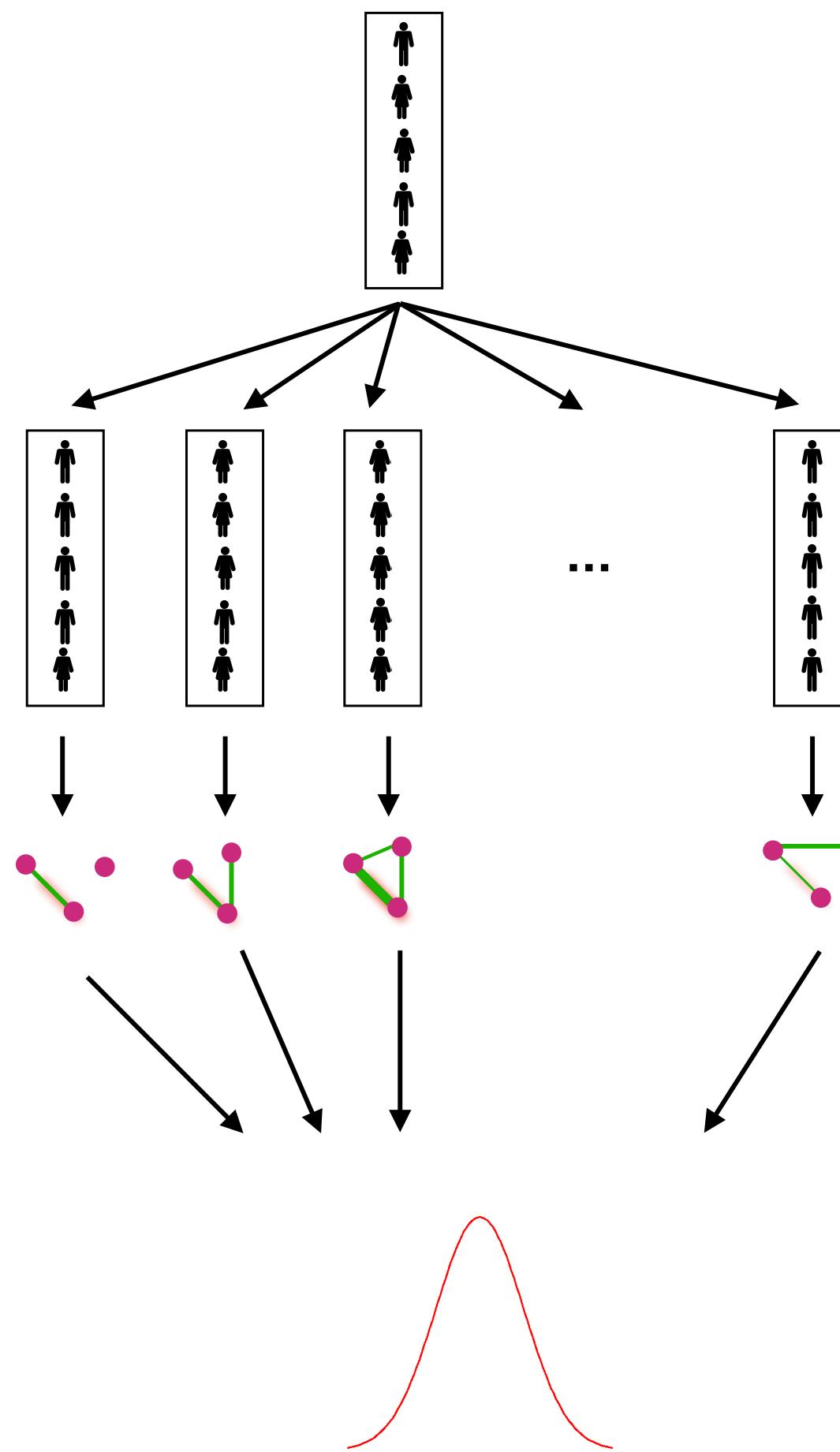
1. Stability of connections: non-parametric bootstrap

- Non-parametric bootstrap:

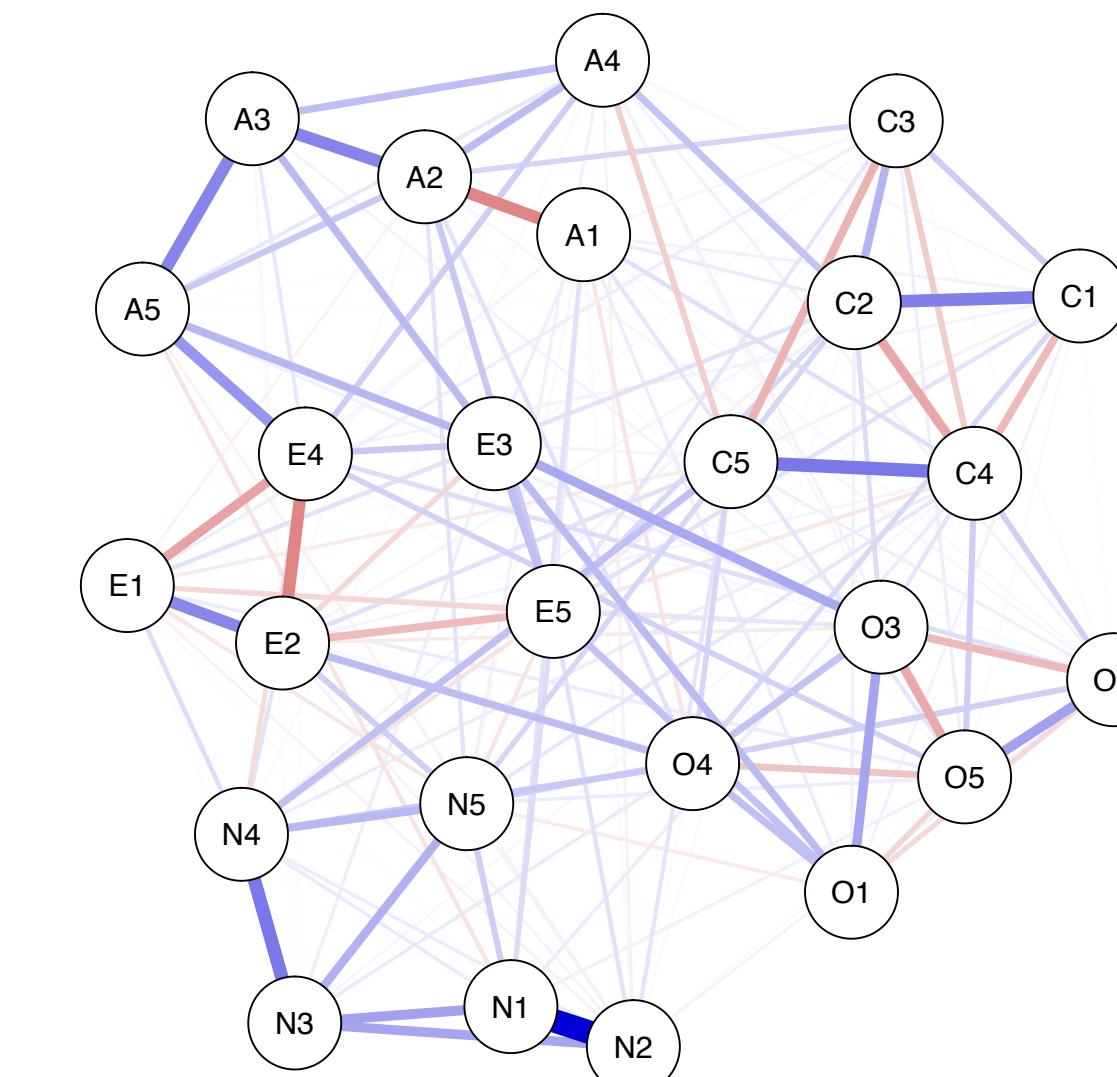
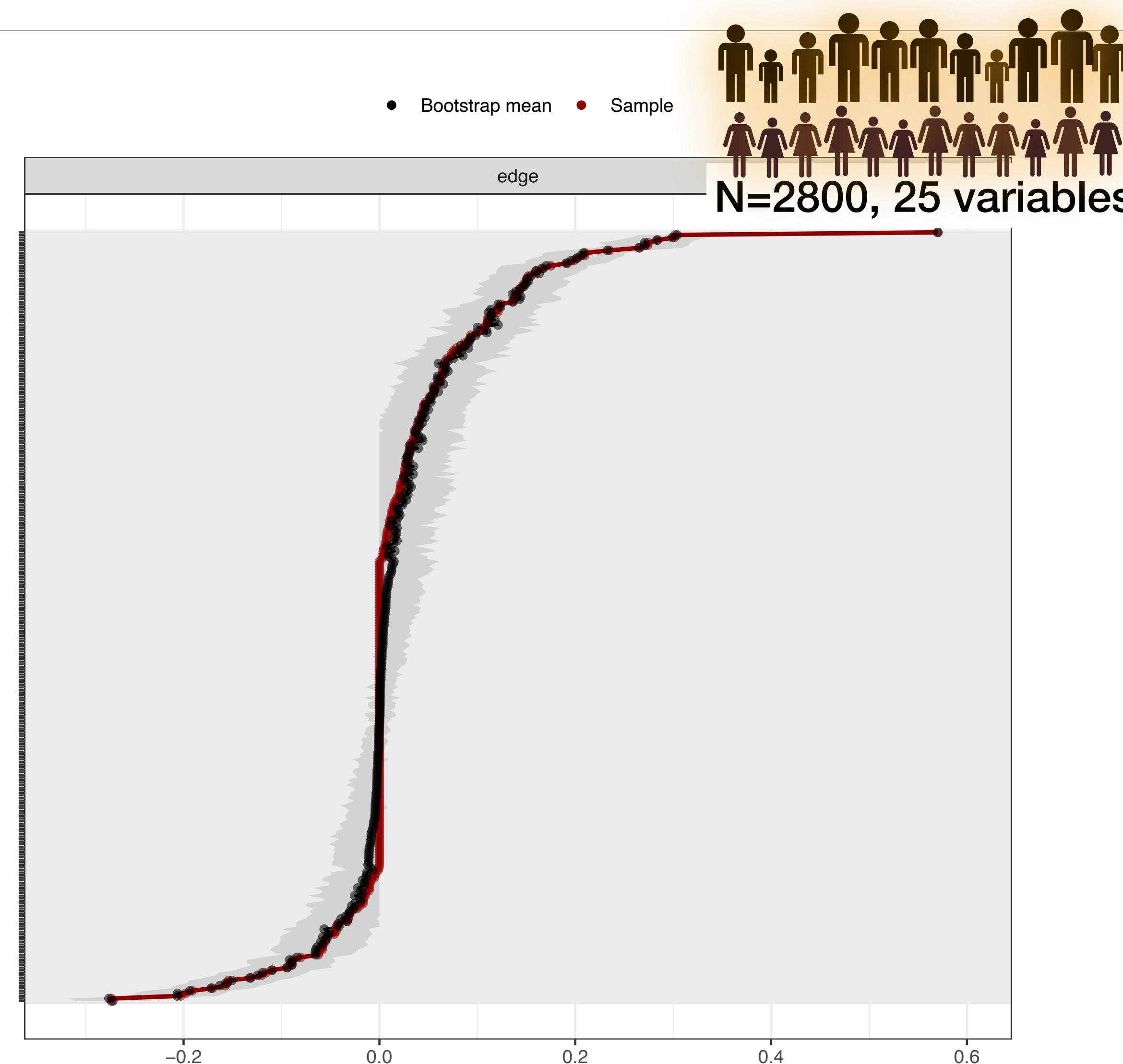
1. compute statistic (e.g., edge weight) in original sample
2. generate a new dataset by sampling from your original data with replacement
3. compute statistic (e.g., edge weight) in new dataset
4. repeat steps 2-3 and use the ranges of compute statistic to draw confidence intervals



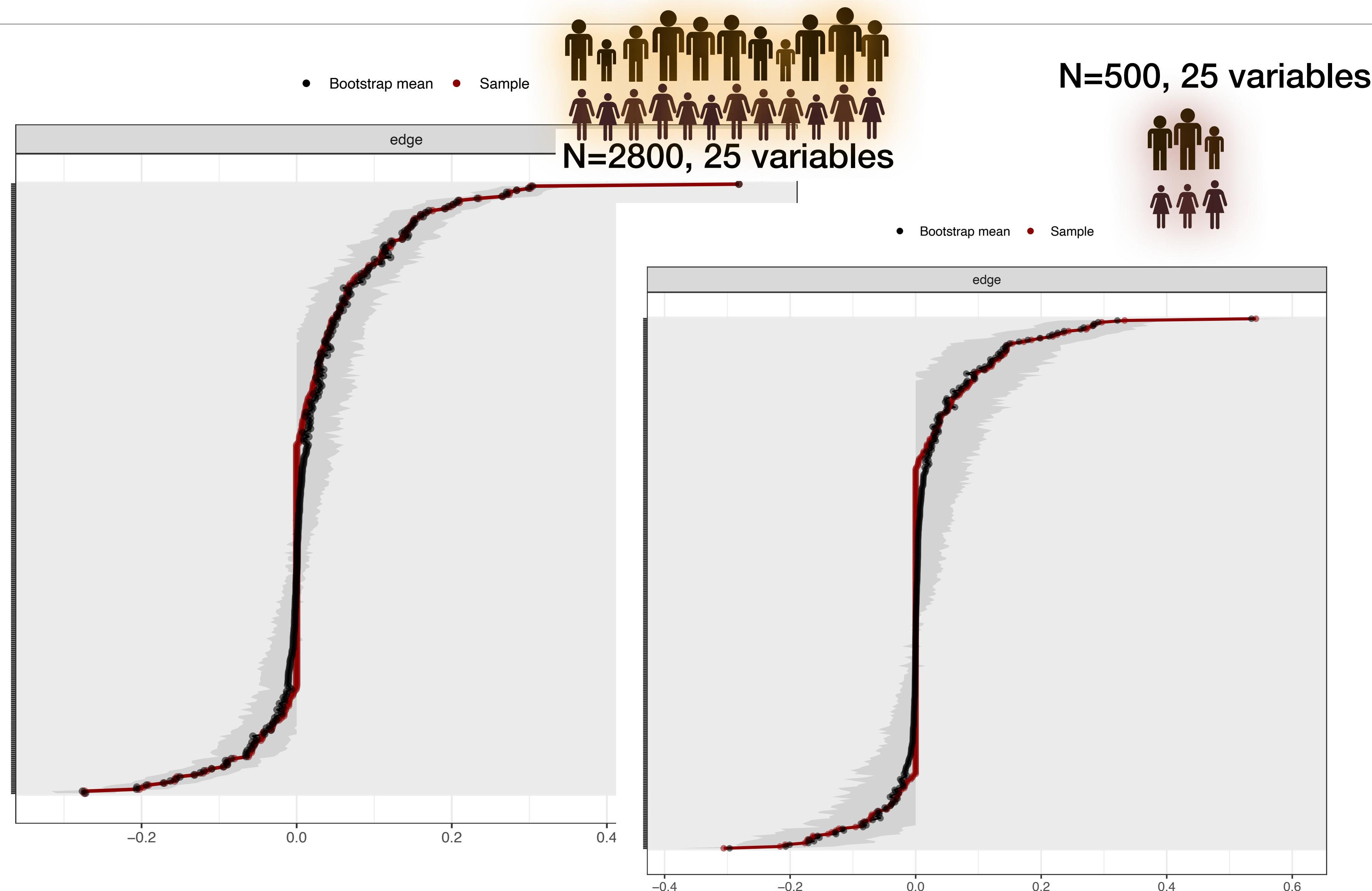
1. Accuracy of connections: non-parametric bootstrap



1. Bootnet output of accuracy estimates

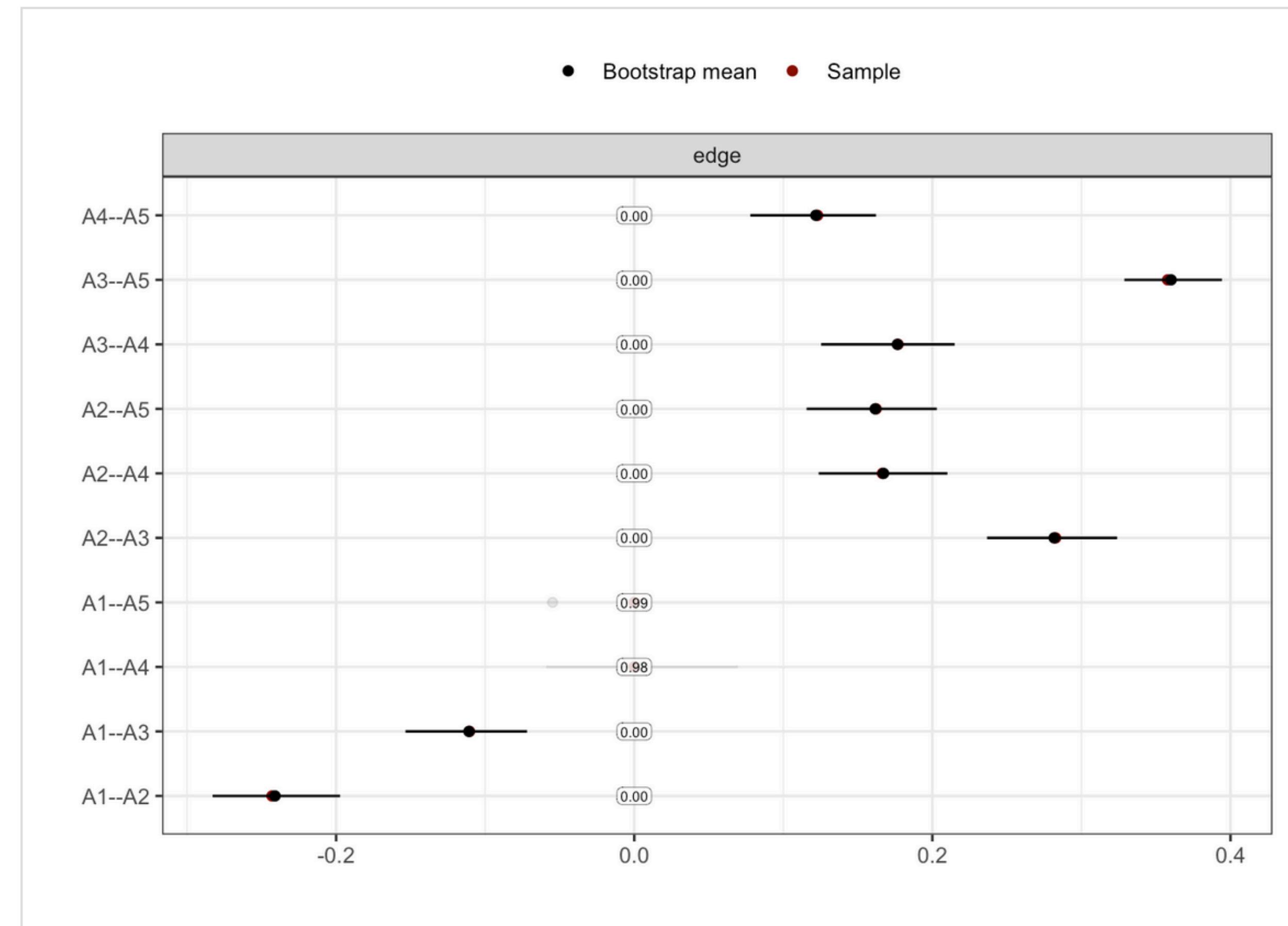


1. Bootnet output of accuracy estimates



1. Bootnet output of accuracy estimates

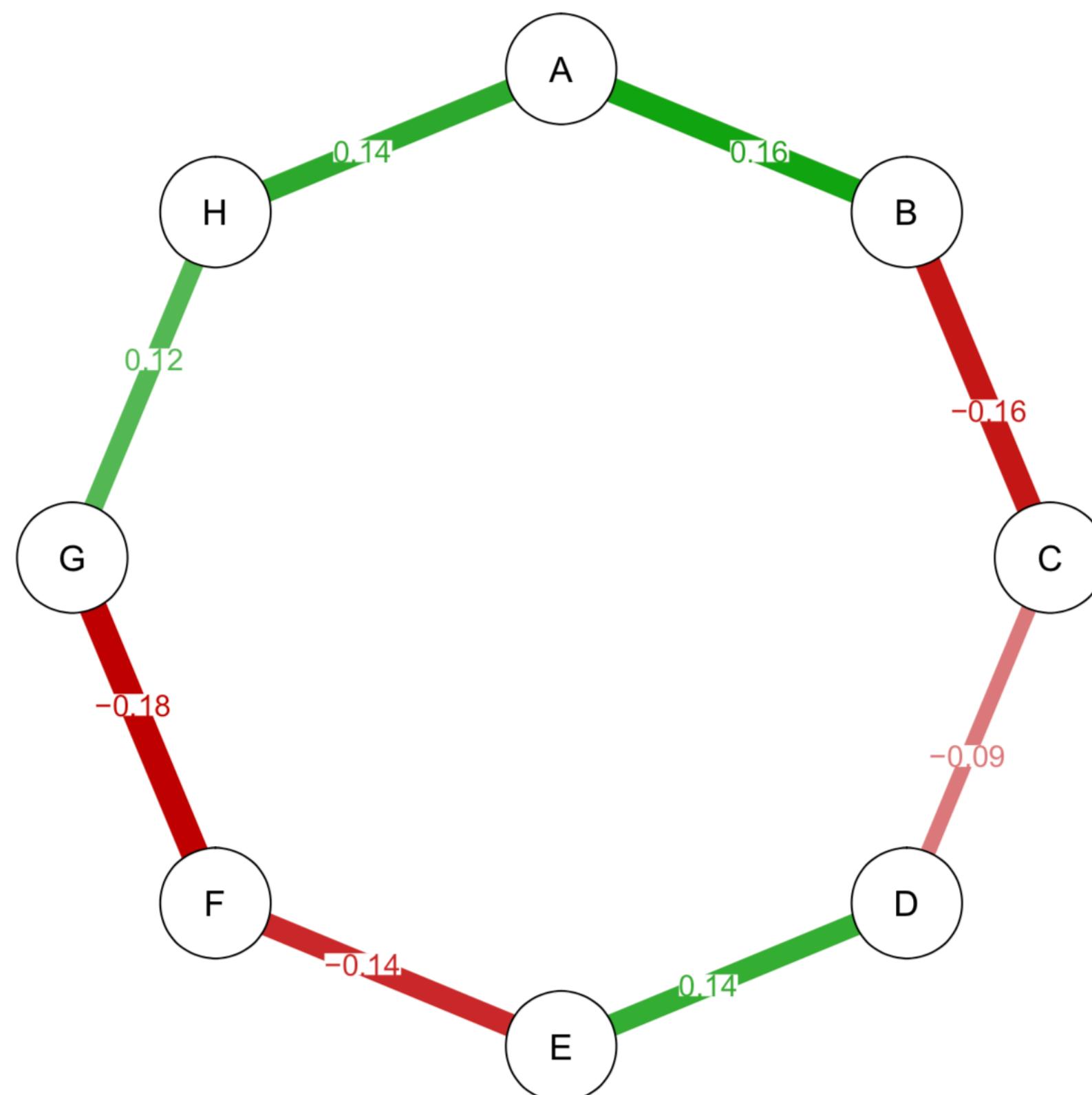
- `plot = "interval"`
- `split0 = TRUE`



Simulation example: accuracy of connections

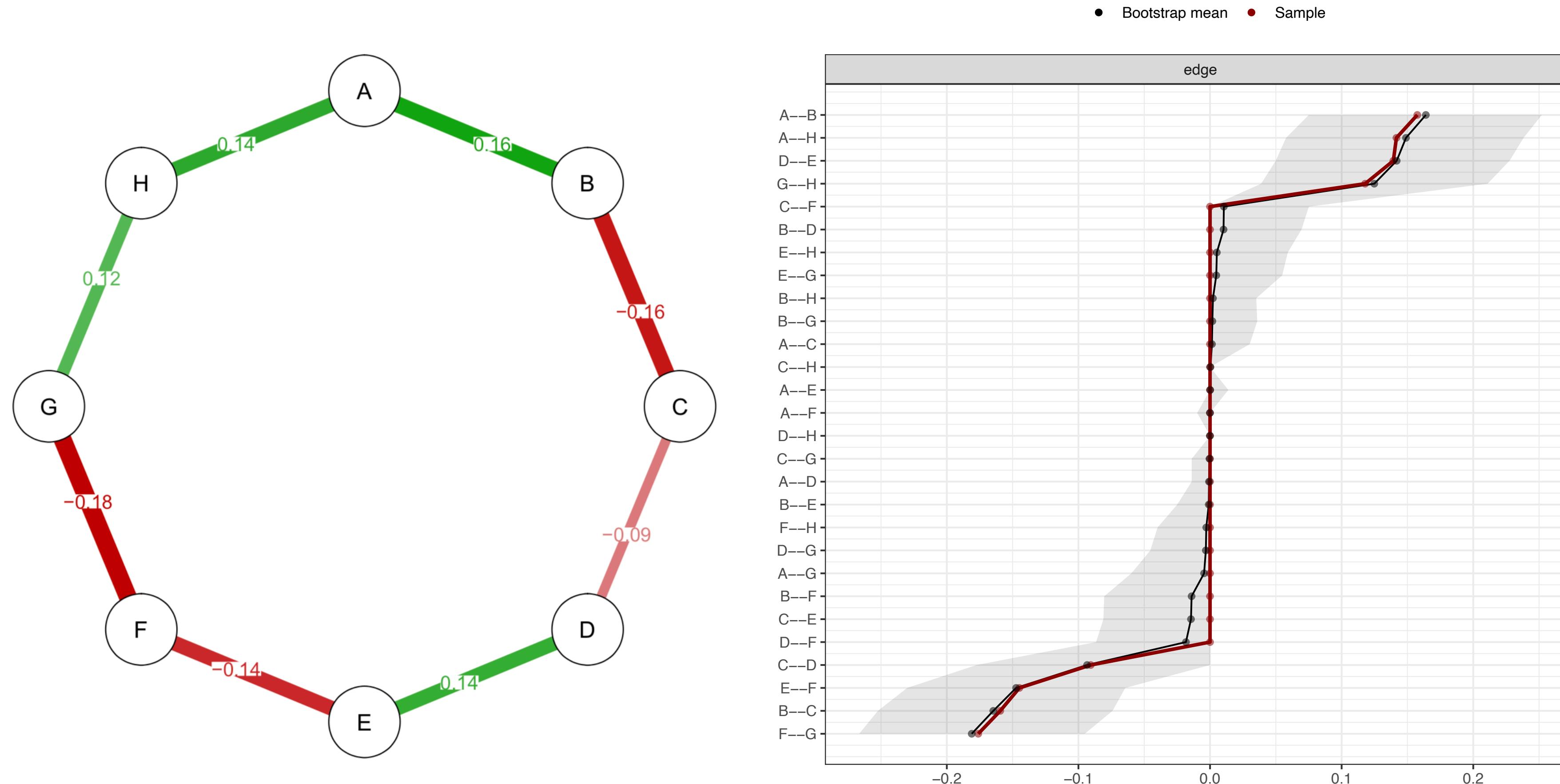
can be slow!

```
boot_accuracy <- bootnet(Network, nBoots = 5000, default = "EBICglasso",  
                           statistics = "edge", nCores = 8)
```



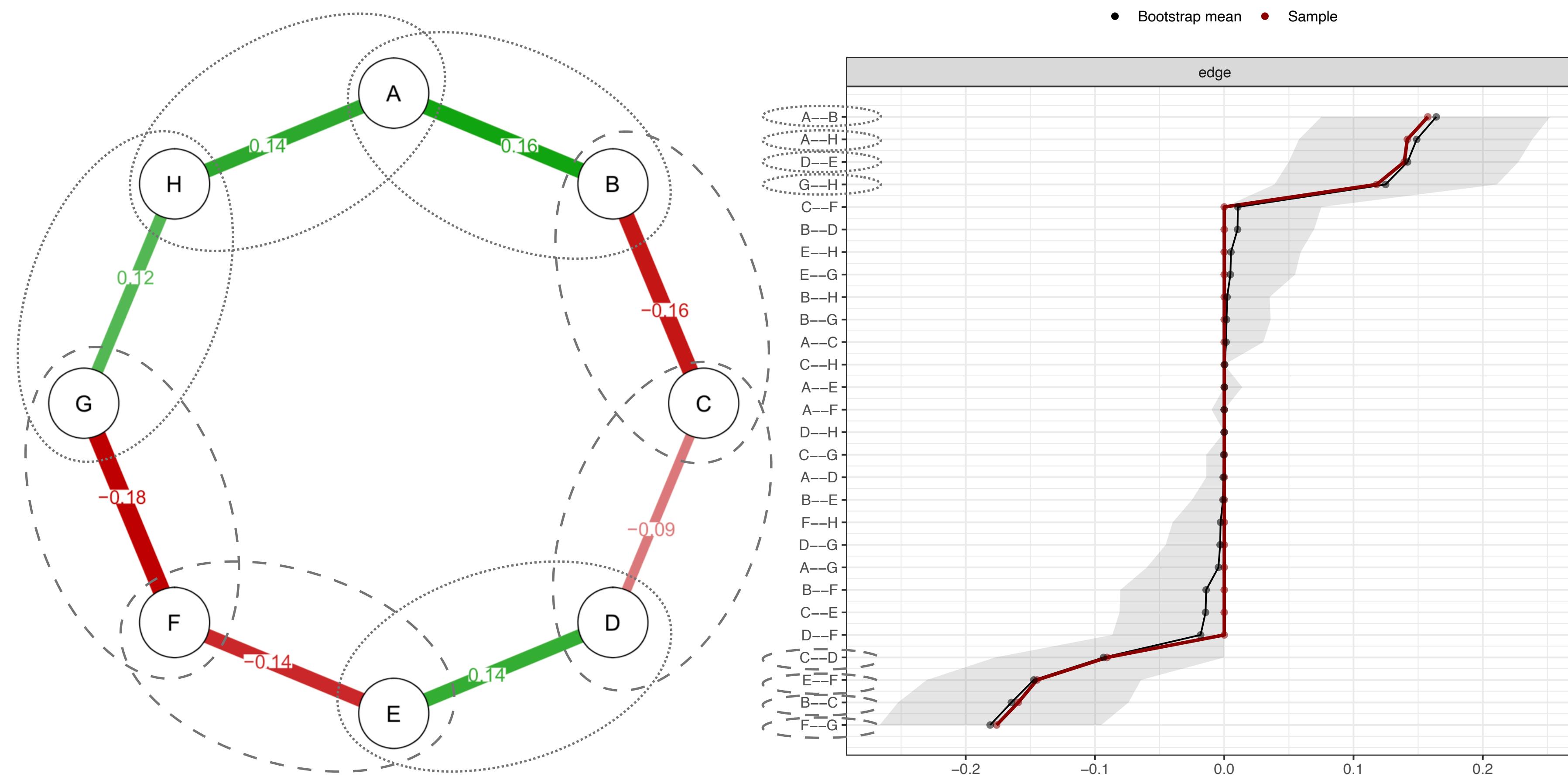
Simulation example: accuracy of connections

```
boot_accuracy <- bootnet(Network, nBoots = 5000, default = "EBICglasso",
                           statistics = "edge", nCores = 8)
plot(boot_accuracy, labels = TRUE, order = "sample")
```



Simulation example: accuracy of connections

```
boot_accuracy <- bootnet(Network, nBoots = 5000, default = "EBICglasso",
                           statistics = "edge", nCores = 8)
plot(boot_accuracy, labels = TRUE, order = "sample")
```



How to?

Behav Res (2018) 50:195–212
DOI 10.3758/s13428-017-0862-1



Estimating psychological networks and their accuracy: A tutorial paper

Sacha Epskamp¹ · Denny Borsboom¹ · Eiko I. Fried¹

Methodology to asses the accuracy of psychological networks, including three steps:

1. assess the *accuracy* of estimated network connections
2. investigate the *stability* of centrality indices
3. *test* whether network connections and centrality estimates for different variables differ from each other

2. Stability of centrality indices

- Assess *stability* of the centrality index by **case-dropping subset bootstrap**
 - bootstrapping confidence intervals around centrality estimates is not possible

 ELSEVIER

Social Networks
Volume 25, Issue 4, October 2003, Pages 283-307



The stability of centrality measures when networks are sampled

Elizabeth Costenbader ^a✉, Thomas W Valente ^{b, 1}✉

Show more

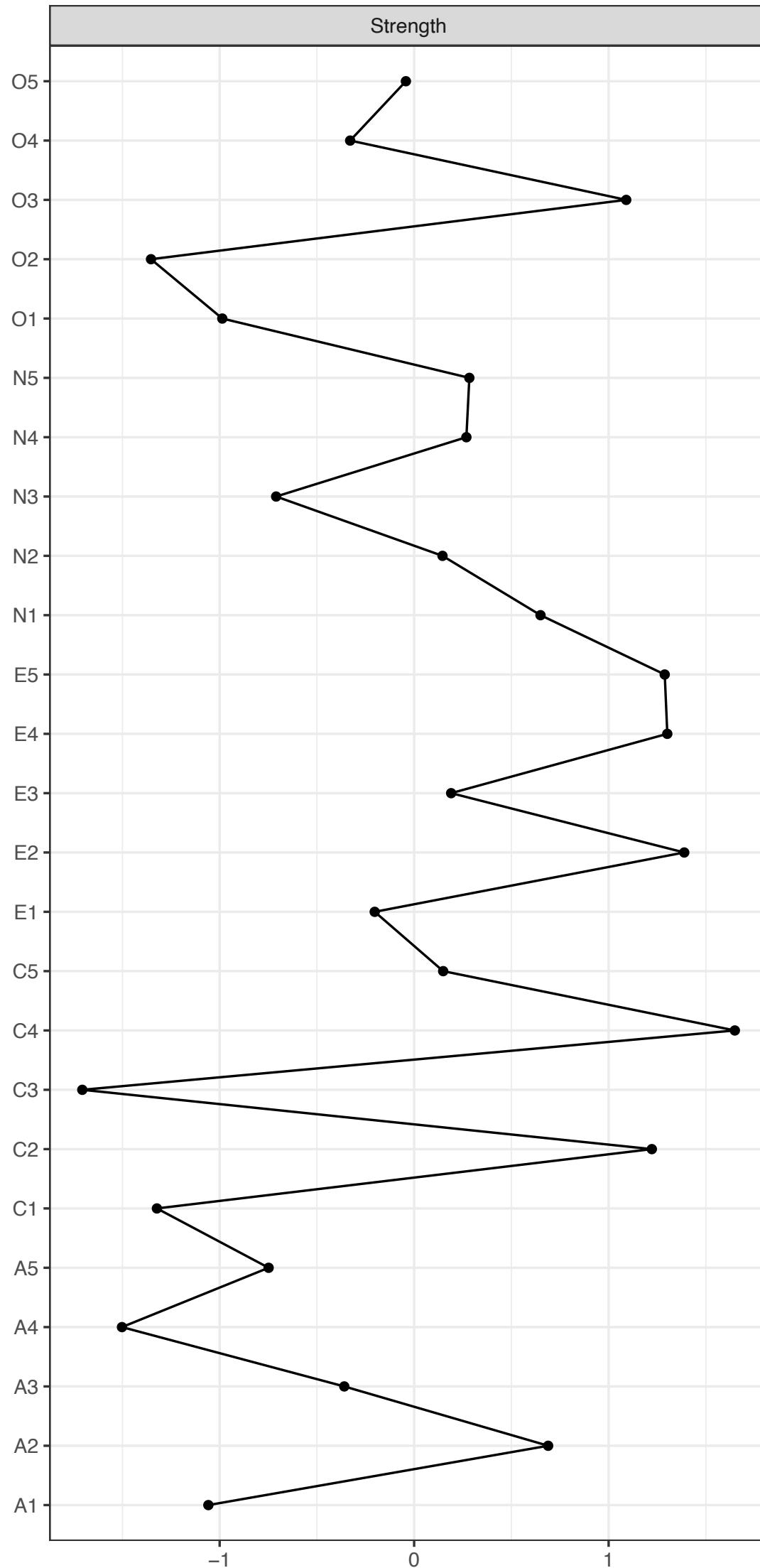
[https://doi.org/10.1016/S0378-8733\(03\)00012-1](https://doi.org/10.1016/S0378-8733(03)00012-1)

Get rights and content

Referred to by Elizabeth Costenbader, Thomas W. Valente

Corrigendum to “The stability of centrality measures when networks are sampled” [Soci...
Social Networks, Volume 26, Issue 4, October 2004, Pages 351

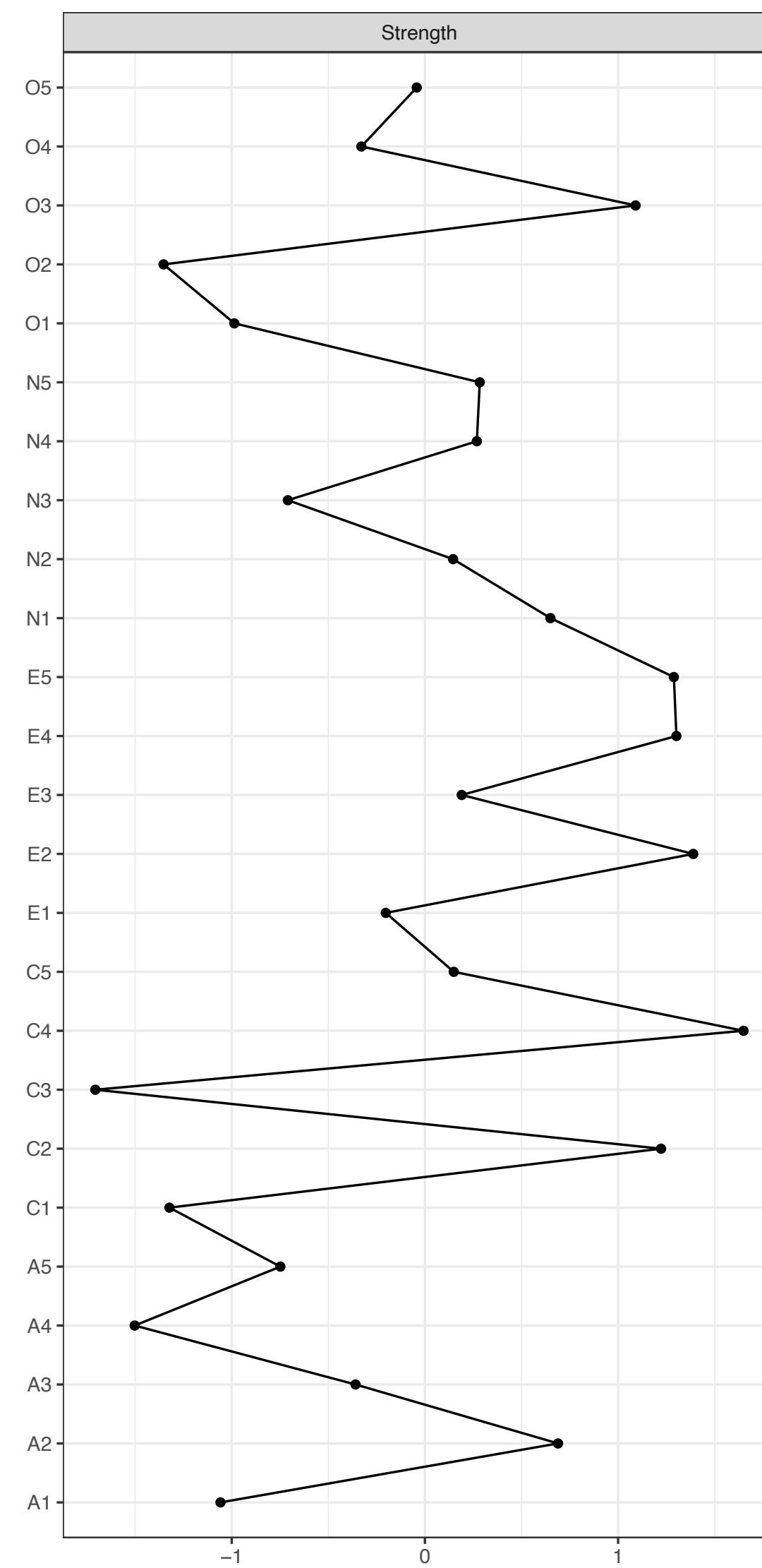
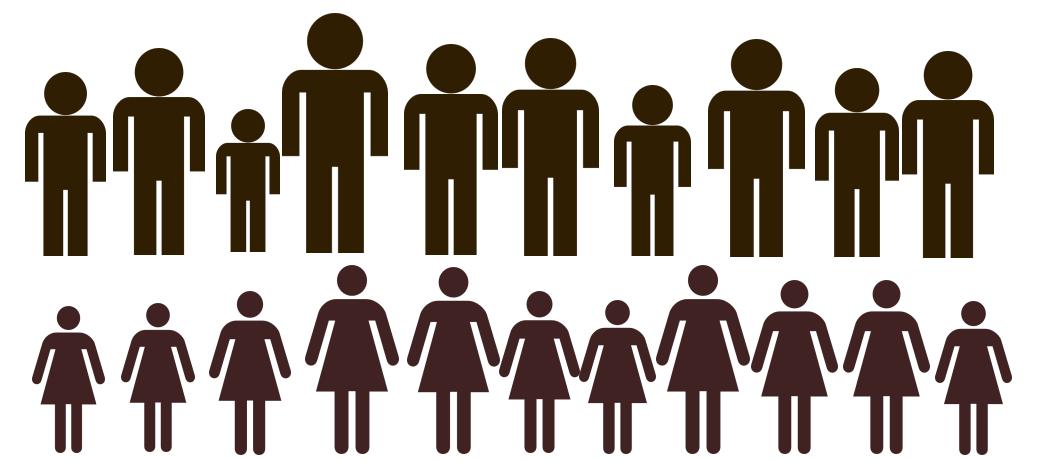
 Download PDF



2. Stability of centrality estimates: case-dropping bootstrap

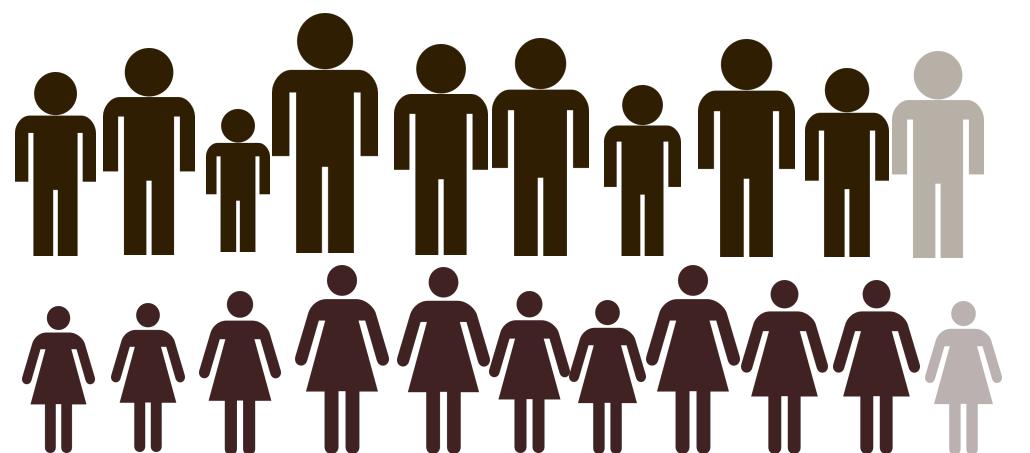
- Case-dropping bootstrap:

1. obtain centrality for dataset ($C_4 > E_2 > E_4 > \dots > C_3$)



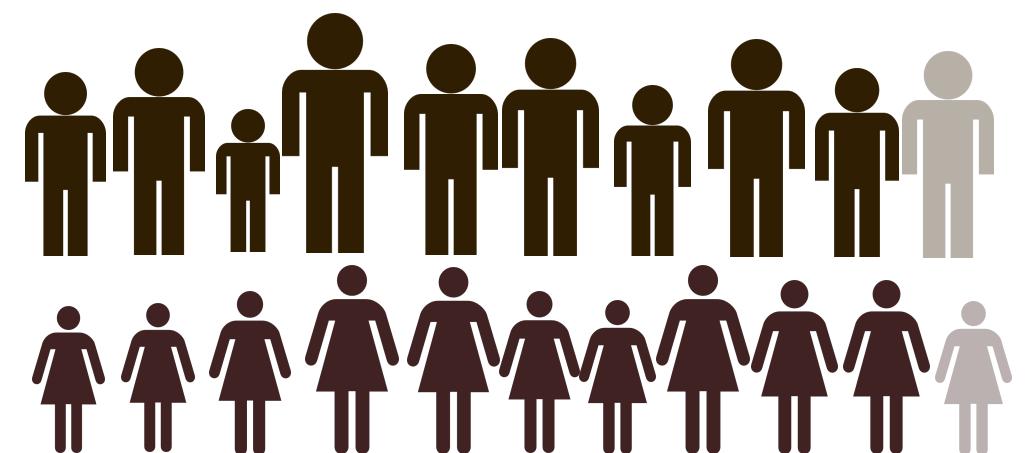
2. Stability of centrality estimates: case-dropping bootstrap

- Case-dropping bootstrap:
 1. obtain centrality for dataset ($C_4 > E_2 > E_4 > \dots > C_3$)
 2. subset data by dropping 10% of the people



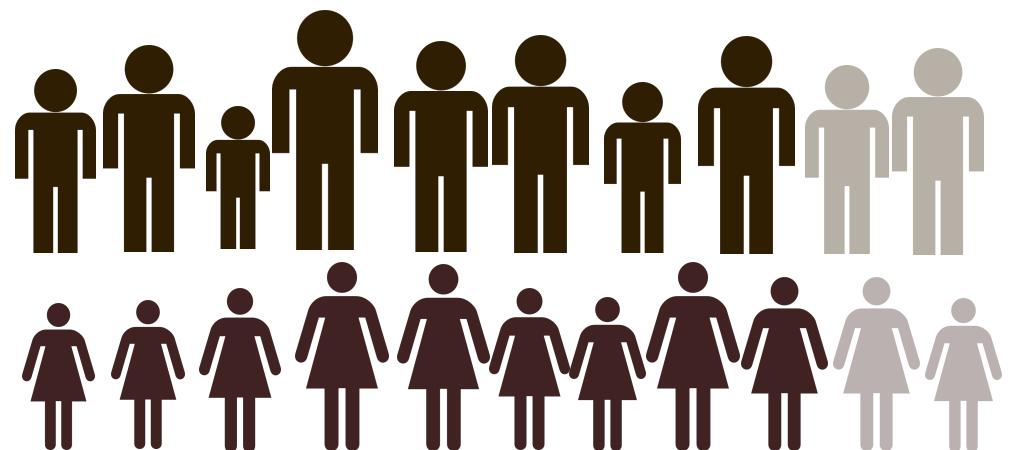
2. Stability of centrality estimates: case-dropping bootstrap

- Case-dropping bootstrap:
 1. obtain centrality for dataset ($C_4 > E_2 > E_4 > \dots > C_3$)
 2. subset data by dropping 10% of the people
 3. obtain centrality for -10% subset of the people ($E_2 > C_4 > E_4 > \dots > C_3$)



2. Stability of centrality estimates: case-dropping bootstrap

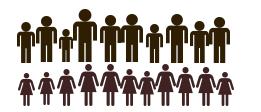
- Case-dropping bootstrap:
 1. obtain centrality for dataset ($C_4 > E_2 > E_4 > \dots > C_3$)
 2. subset data by dropping 10% of the people
 3. obtain centrality for -10% subset of the people ($E_2 > C_4 > E_4 > \dots > C_3$)
 4. repeat steps 2-3 for -20%, -30%, ..., -90%



2. Stability of centrality estimates: case-dropping bootstrap

- Case-dropping bootstrap:

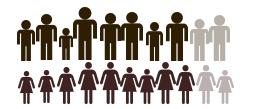
5. obtain centrality estimates for different subsets:



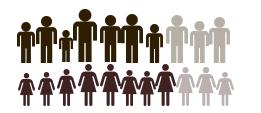
- Full data (11 > 13 > 3 > ... > 8)



- N-10% (13 > 11 > 3 > ... > 8)



- N-20% (13 > 11 > 3 > ... > 6)



- N-30% (13 > 11 > 3 > ... > 8)



- N-40% (13 > 11 > 16 > ... > 8)



- N-50% (11 > 13 > 16 > ... > 8)



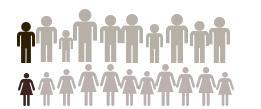
- N-60% (11 > 13 > 16 > ... > 9)



- N-70% (11 > 13 > 3 > ... > 14)



- N-80% (13 > 11 > 7 > ... > 9)



- N-90% (16 > 5 > 3 > ... > 6)

2. Stability of centrality estimates: case-dropping bootstrap

- Case-dropping bootstrap:

5. obtain centrality estimates for different subsets:



- Full data (11 > 13 > 3 > ... > 8)



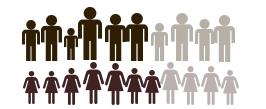
- N-10% (13 > 11 > 3 > ... > 8)



- N-20% (13 > 11 > 3 > ... > 6)



- N-30% (13 > 11 > 3 > ... > 8)



- N-40% (13 > 11 > 16 > ... > 8)



- N-50% (11 > 13 > 16 > ... > 8)



- N-60% (11 > 13 > 16 > ... > 9)



- N-70% (11 > 13 > 3 > ... > 14)



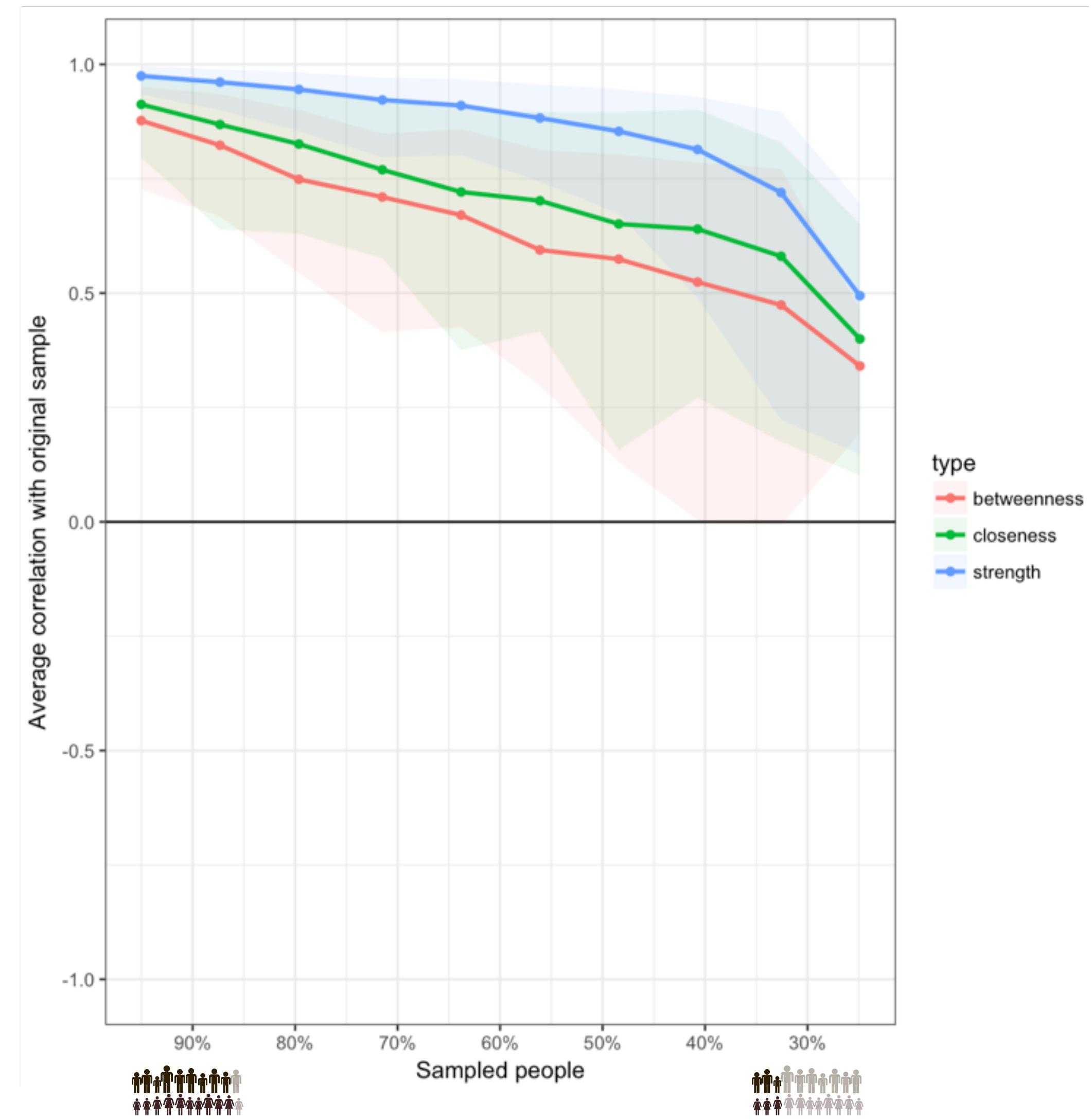
- N-80% (13 > 11 > 7 > ... > 9)



- N-90% (16 > 5 > 3 > ... > 6)

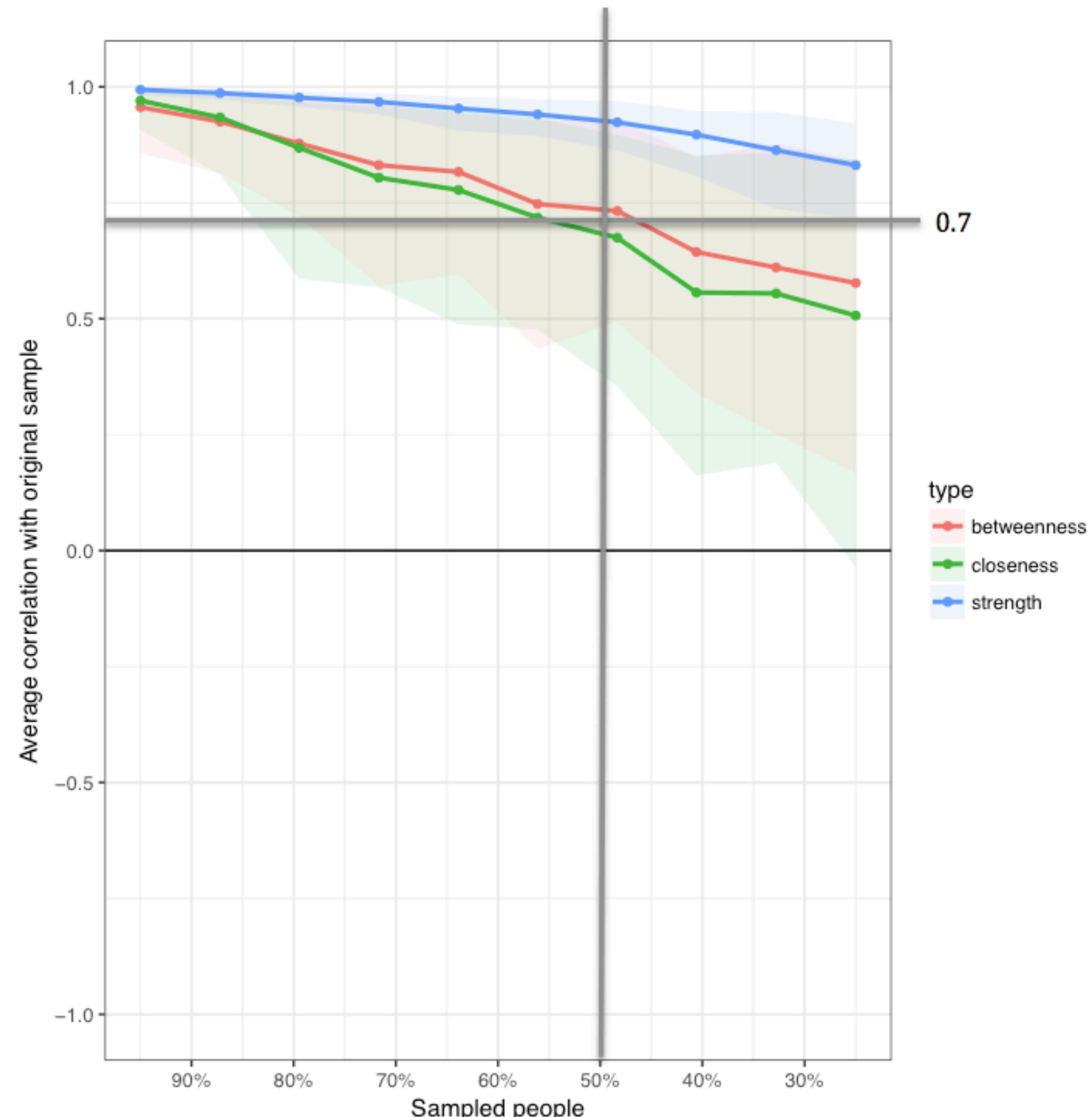
6. correlate centrality indices for each subset with original centrality indices (on full data)

2. Bootnet output of stability estimates



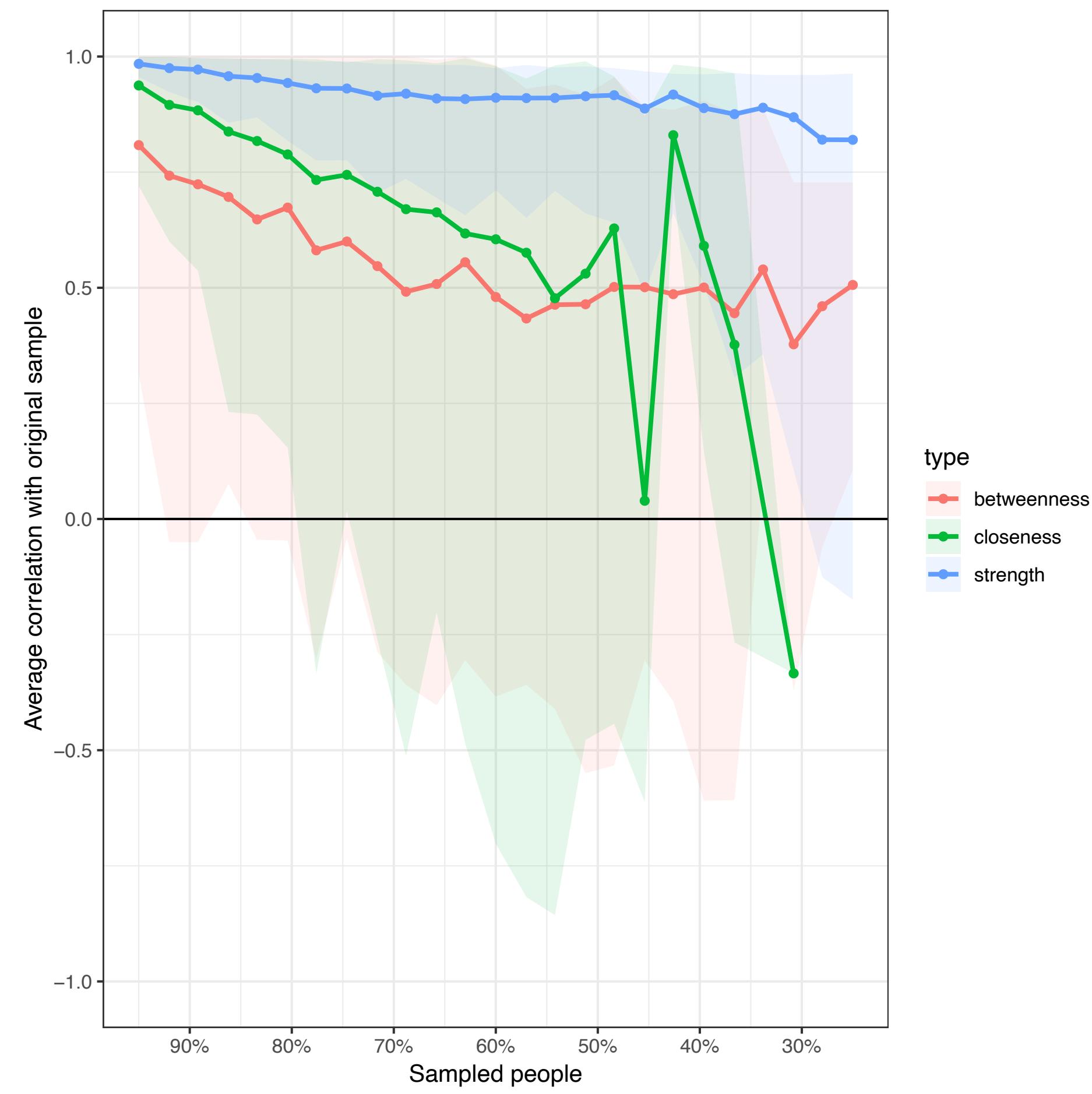
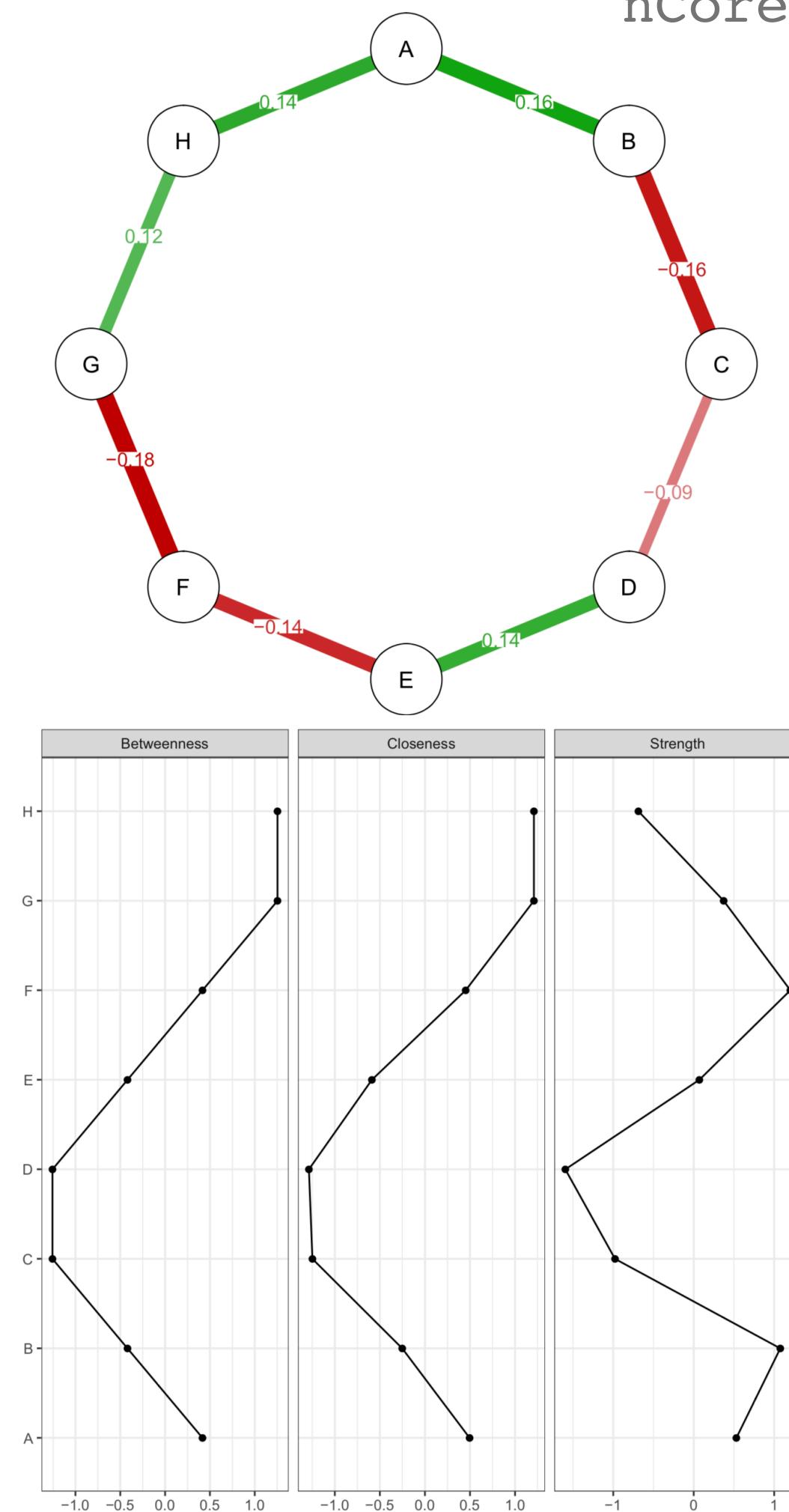
2. Bootnet output of stability estimates: CS-coefficient

- Correlation stability (CS) coefficient:
 - represents the maximum proportion of cases that can be dropped, such that with 95% probability the correlation between the original centrality indices and the centrality indices based on the subset is 0.7 or higher
 - retain correlation of 0.7 in at least 95% of the samples
- recommendation:
 - preferably above 0.50 and should not be below 0.25



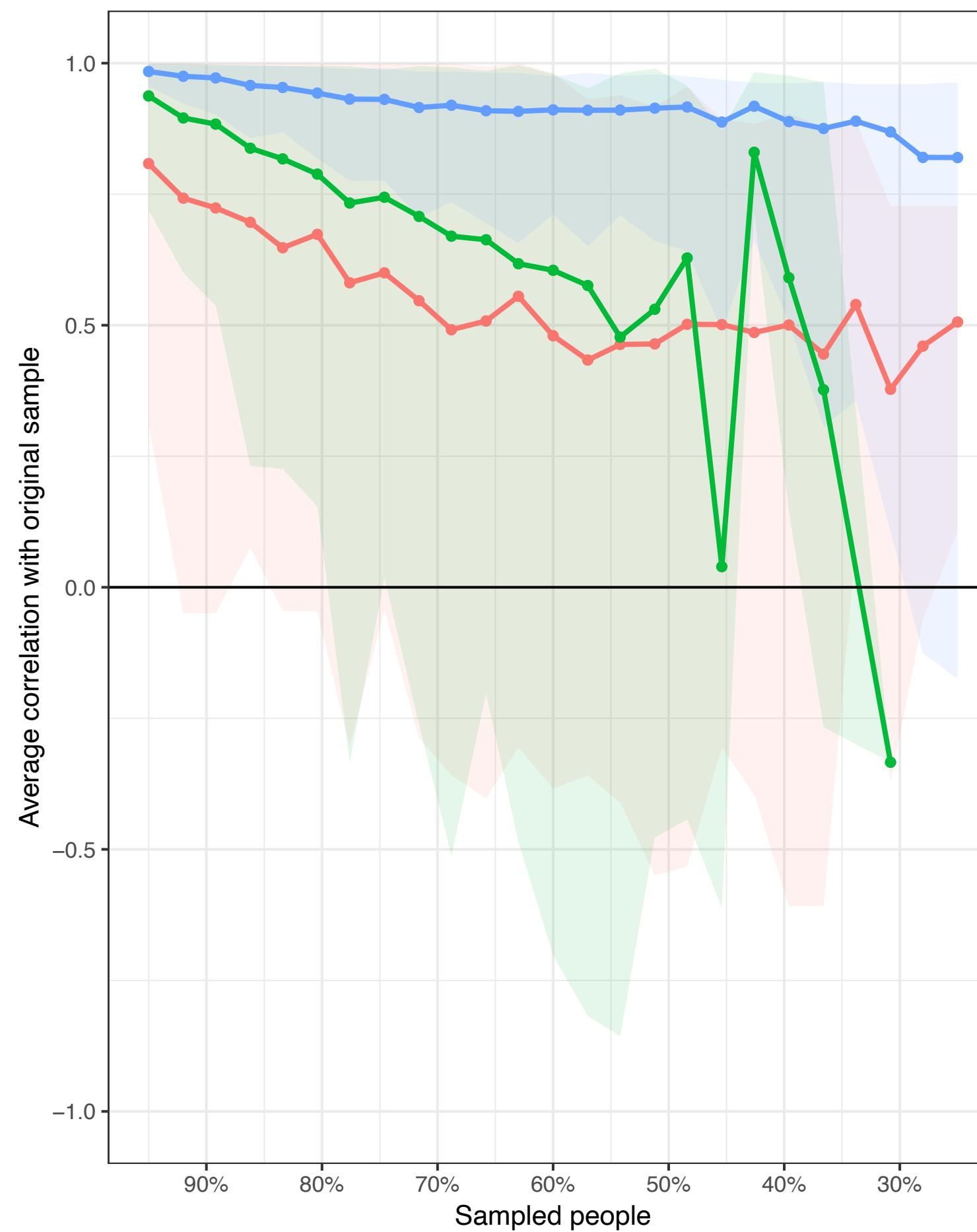
Simulation example: stability of centrality indices

```
boot_stability <- bootnet(Network, nBoots = 5000, default = "EBICglasso", type = "case"  
nCores = 8, statistics = c('betweenness', 'closeness', 'strength'))
```



Simulation example: stability of centrality indices

```
boot_stability <- bootnet(Network, nBoots = 5000, default = "EBICglasso", type = "case"  
                           nCores = 8, statistics = c('betweenness', 'closeness', 'strength'))
```



`corStability(boot_stability)`

Maximum drop proportions to retain correlation of 0.7 in at least 95% of the samples:

betweenness: 0

- For more accuracy, run `bootnet(..., caseMin = 0, caseMax = 0.05)`

closeness: 0.05

- For more accuracy, run `bootnet(..., caseMin = 0, caseMax = 0.08)`

strength: 0.284

- For more accuracy, run `bootnet(..., caseMin = 0.254, caseMax = 0.312)`

Accuracy can also be increased by increasing both 'nBoots' and 'caseN'.

How to?

Behav Res (2018) 50:195–212
DOI 10.3758/s13428-017-0862-1



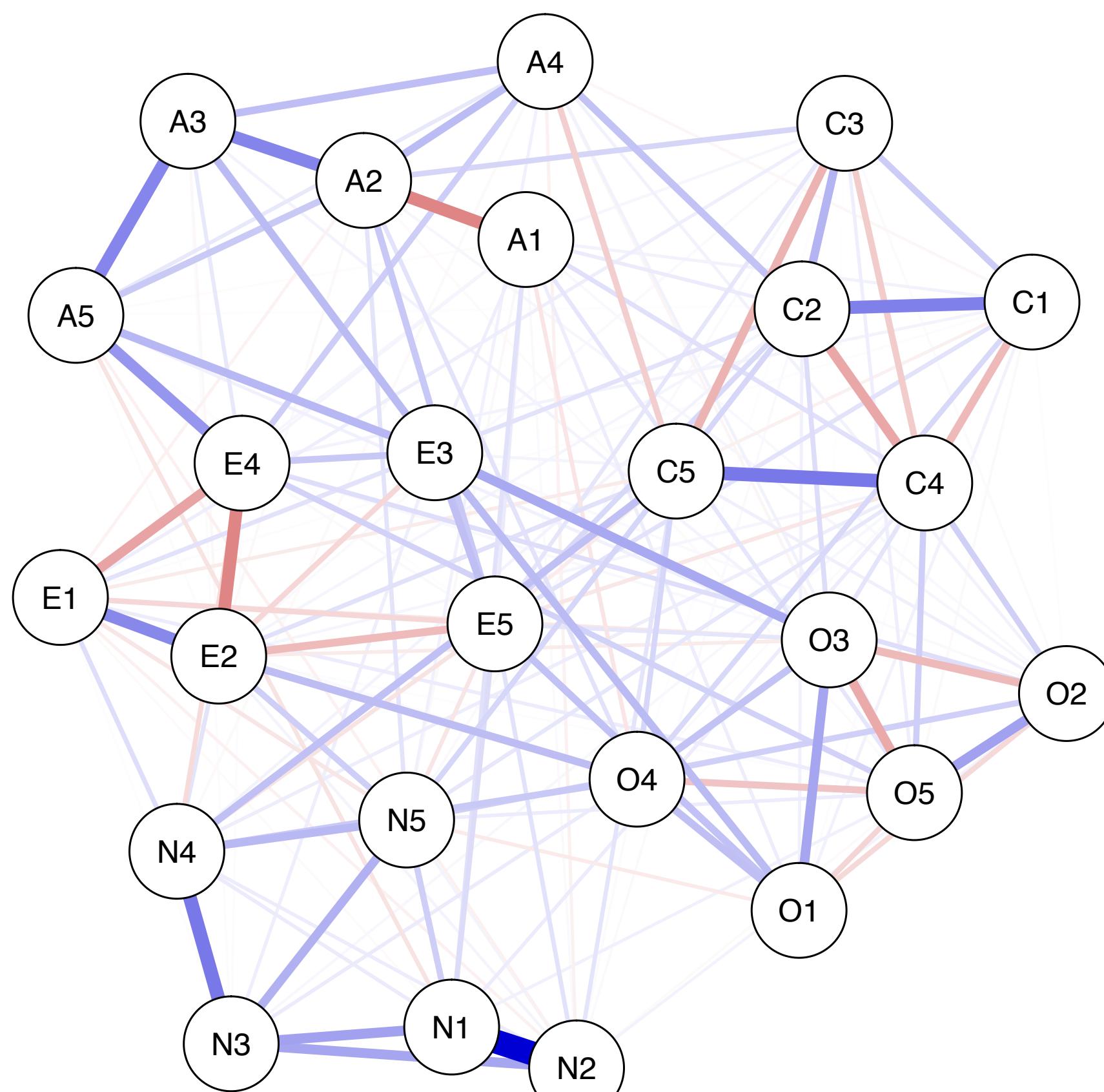
Estimating psychological networks and their accuracy: A tutorial paper

Sacha Epskamp¹ · Denny Borsboom¹ · Eiko I. Fried¹

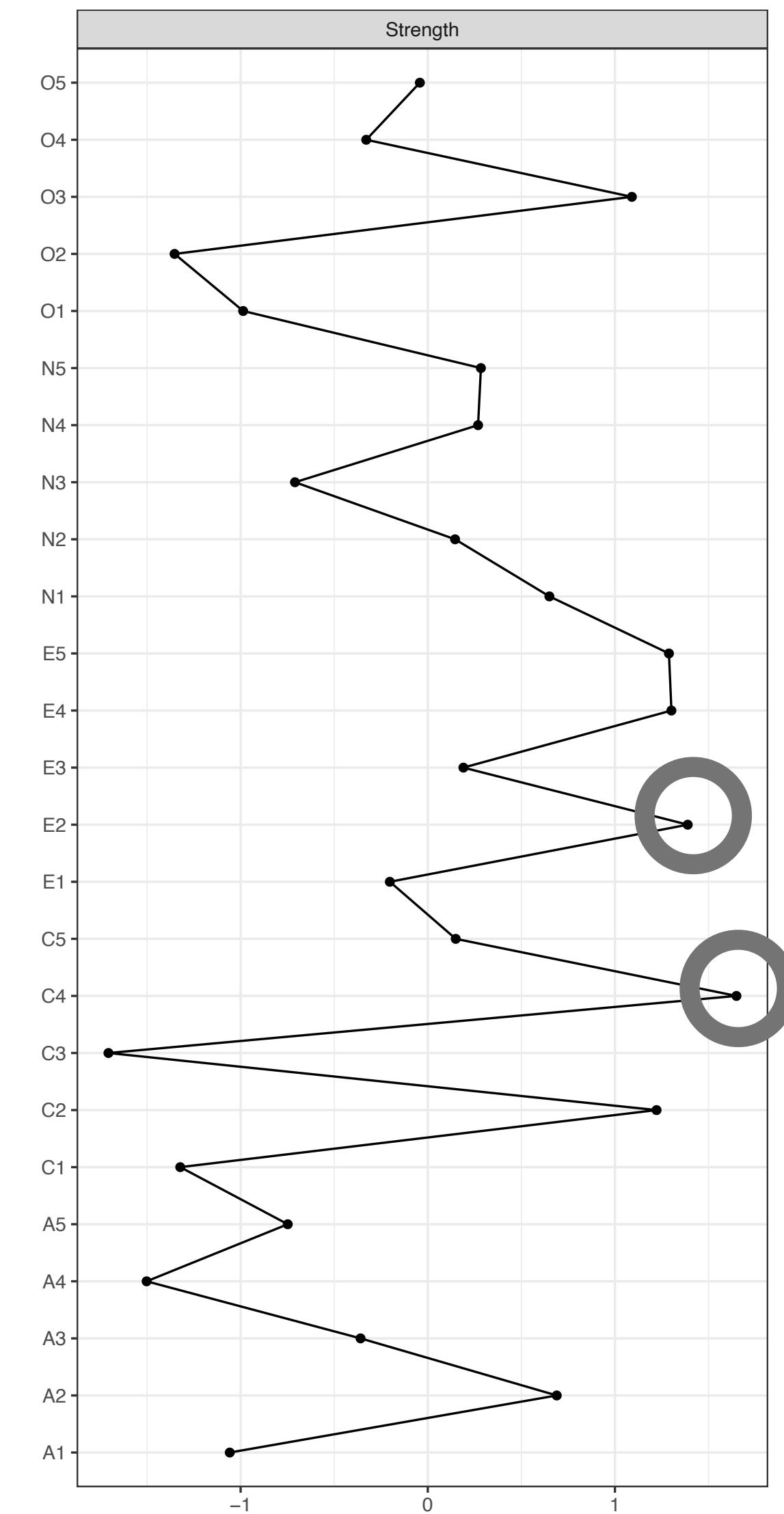
Methodology to asses the accuracy of psychological networks, including three steps:

1. assess the *accuracy* of estimated network connections
2. investigate the *stability* of centrality indices
3. *test whether network connections and centrality estimates for different variables differ from each other*

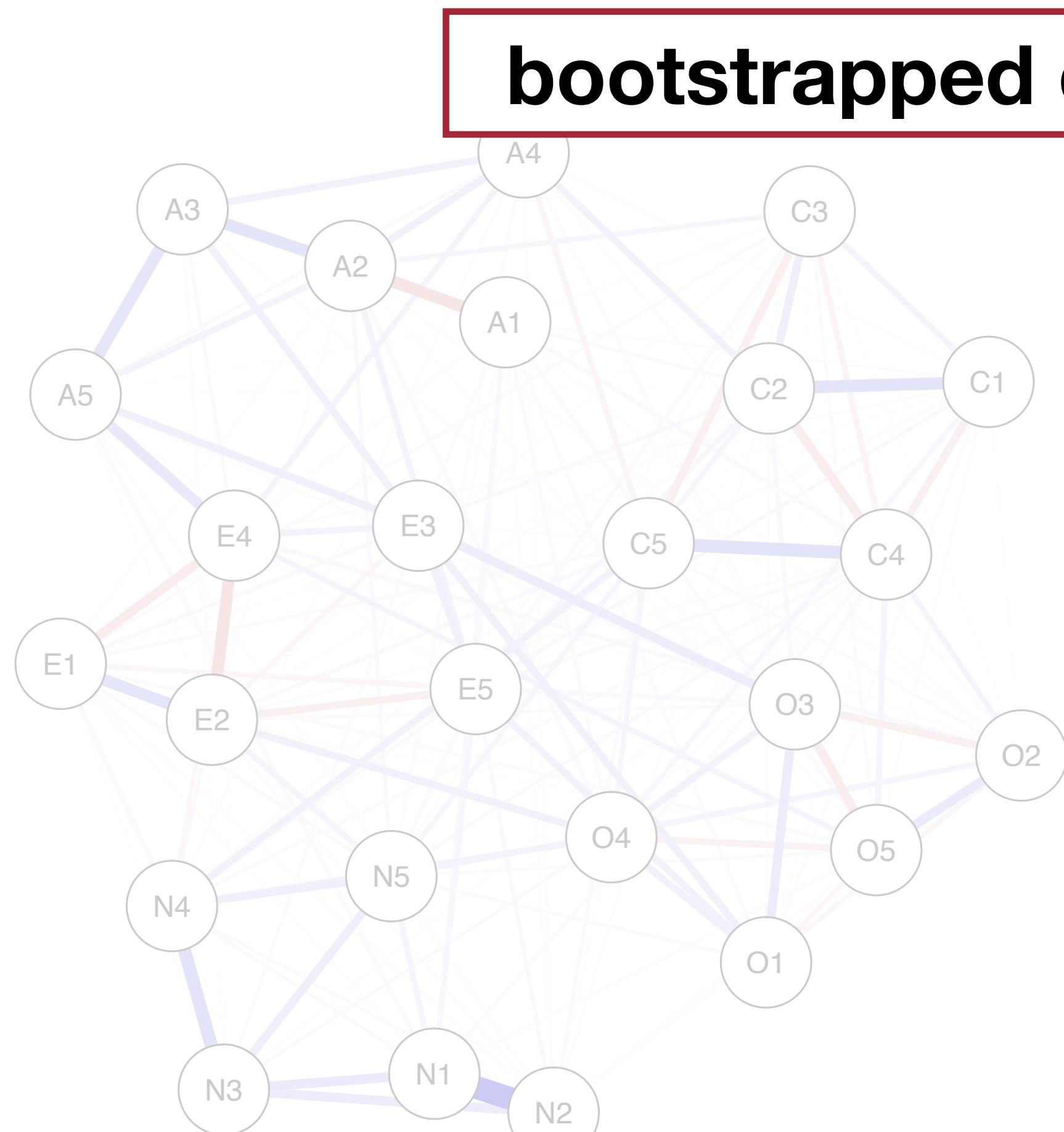
Accuracy and stability in networks



- Is edge N1-N2 *meaningfully* larger than edge N1-N3?
- Is node C4 substantially more central than node E2?

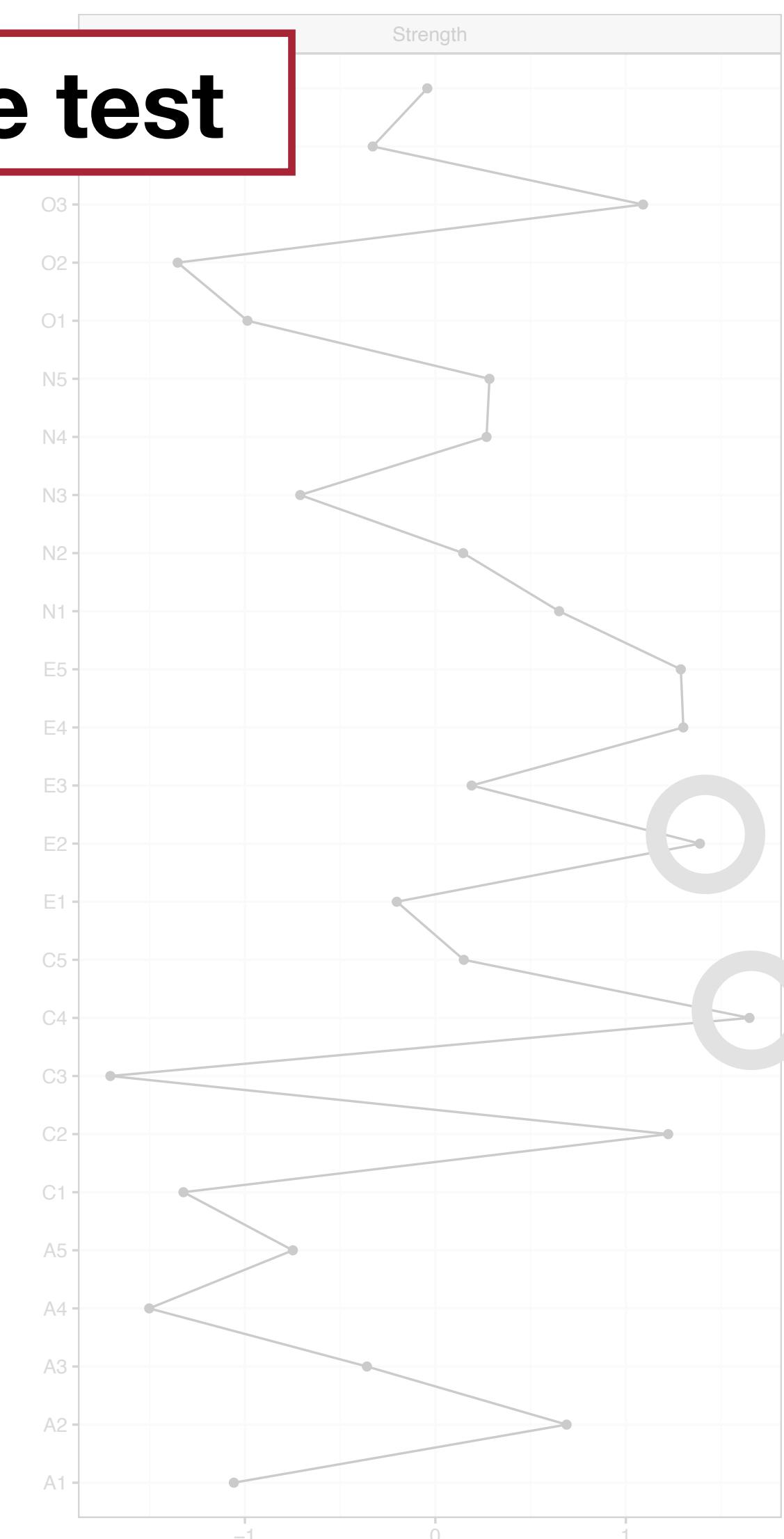


Accuracy and stability in networks

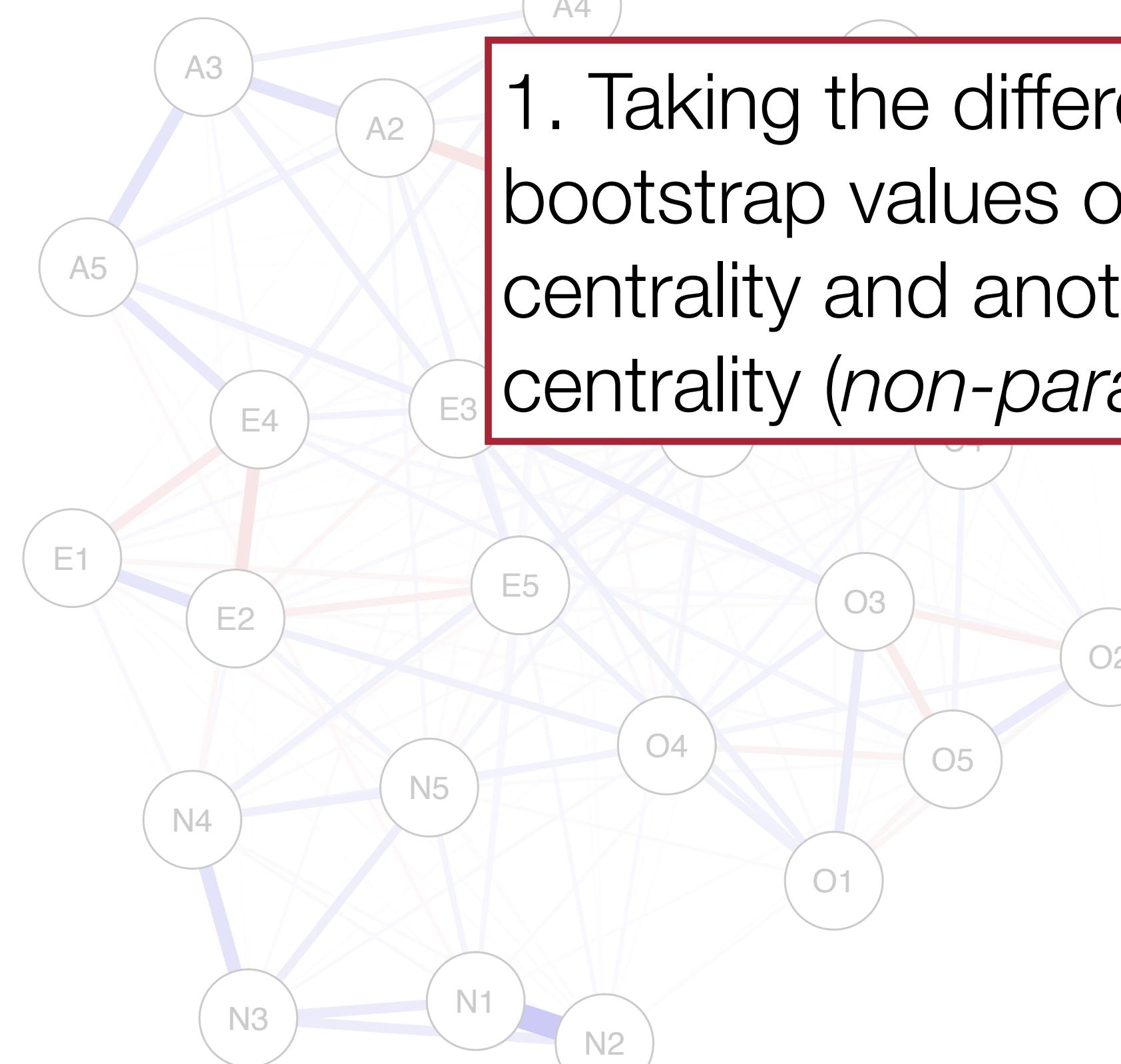


bootstrapped difference test

- Is edge N1-N2 *meaningfully* larger than edge N1-N3?
- Is node C4 substantially more central than node E2?



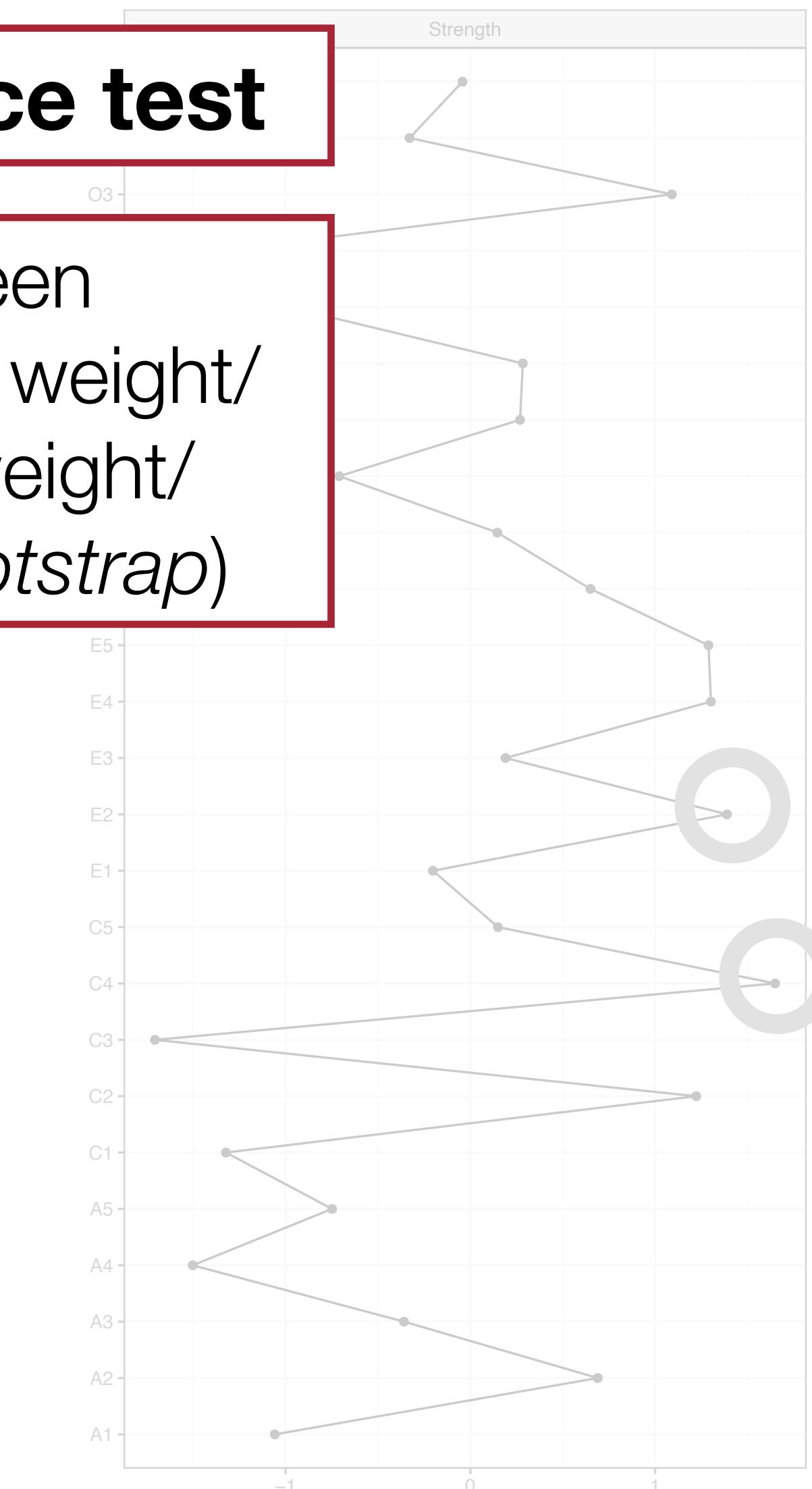
Accuracy and stability in networks



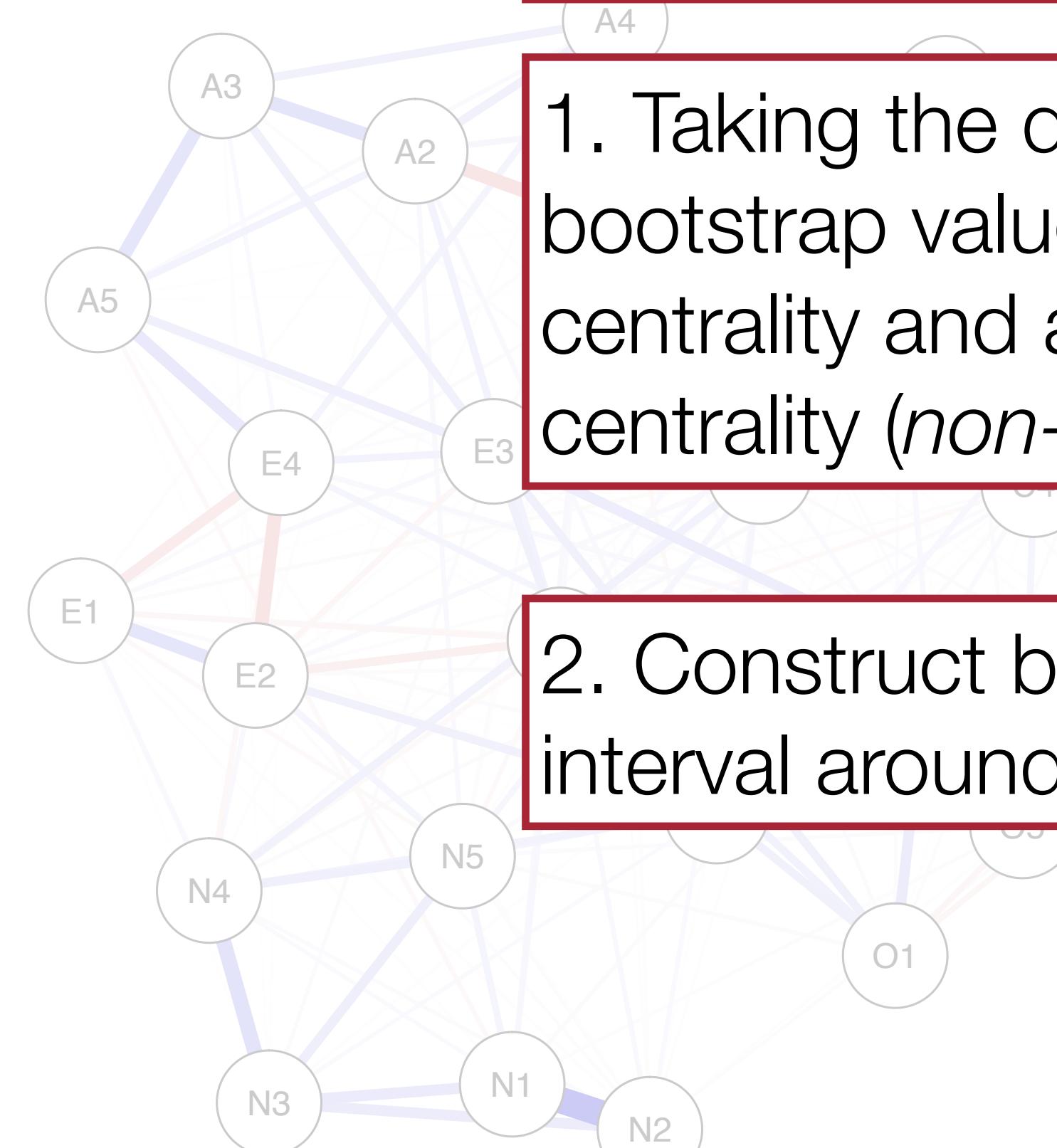
bootstrapped difference test

1. Taking the difference between bootstrap values of one edge weight/centrality and another edge weight/centrality (*non-parametric bootstrap*)

- Is edge N1-N2 *meaningfully* larger than edge N1-N3?
- Is node C4 *substantially* more central than node E2?



Accuracy and stability in networks



bootstrapped difference test

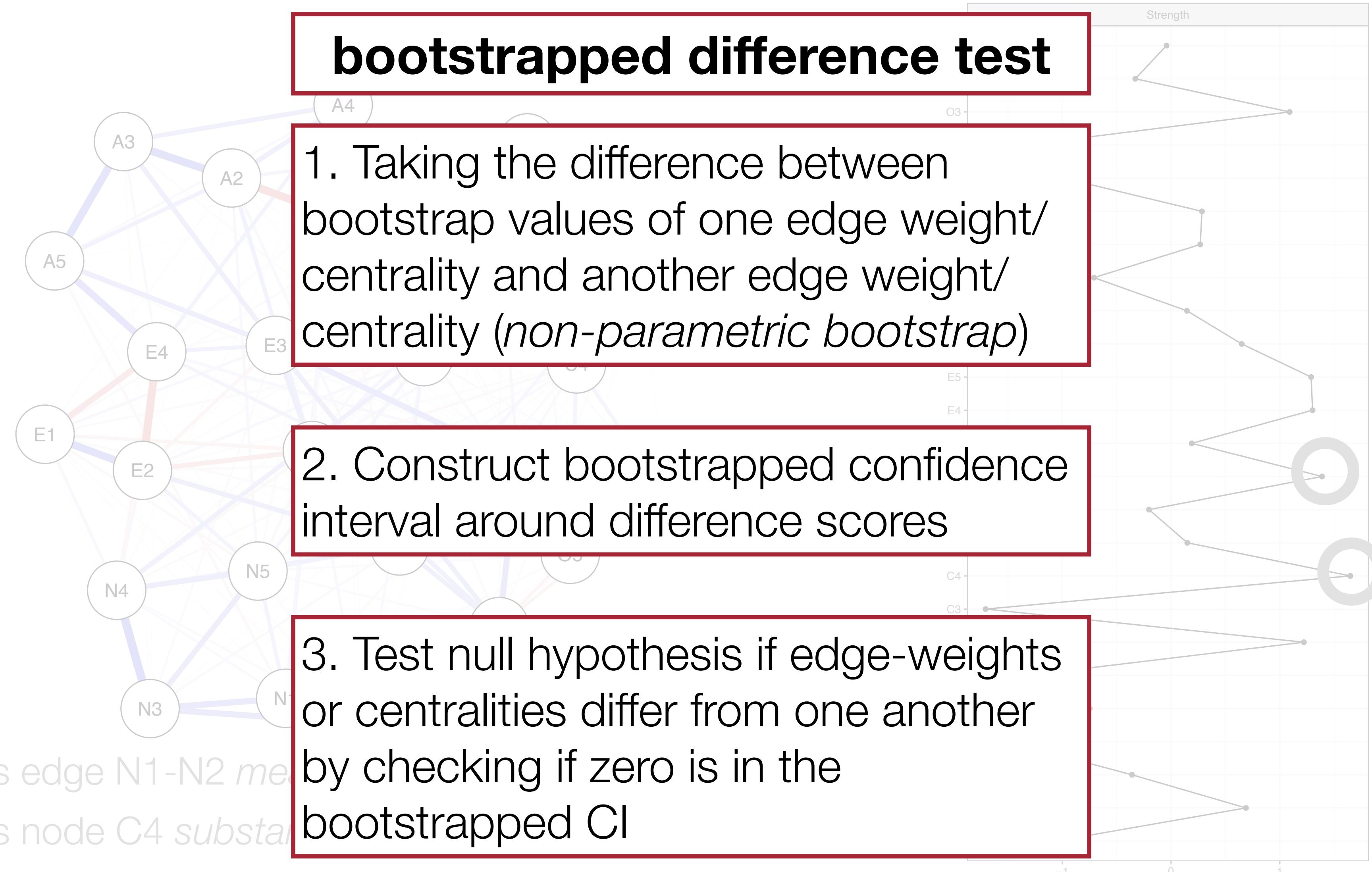
1. Taking the difference between bootstrap values of one edge weight/centrality and another edge weight/centrality (*non-parametric bootstrap*)

2. Construct bootstrapped confidence interval around difference scores

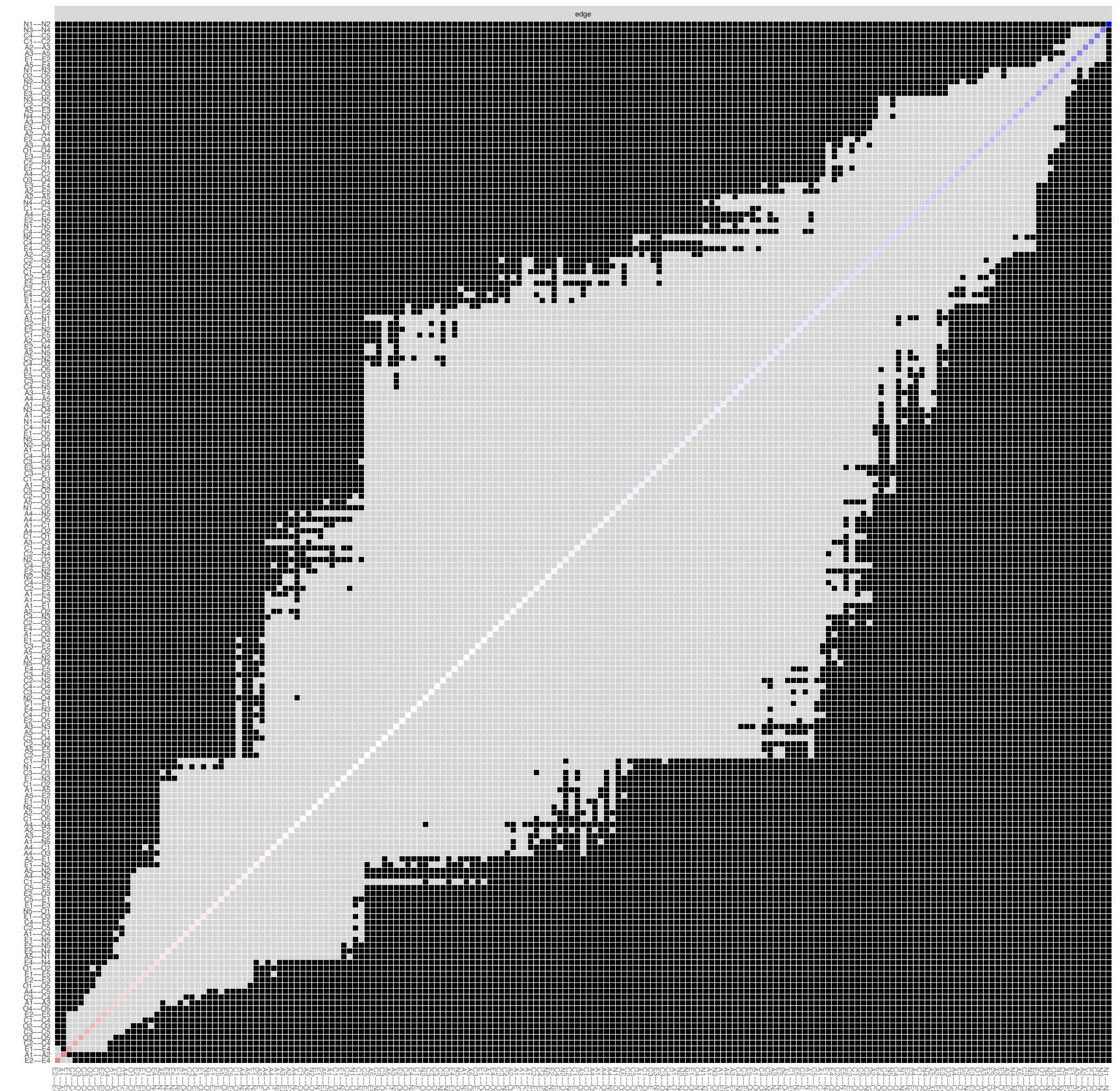
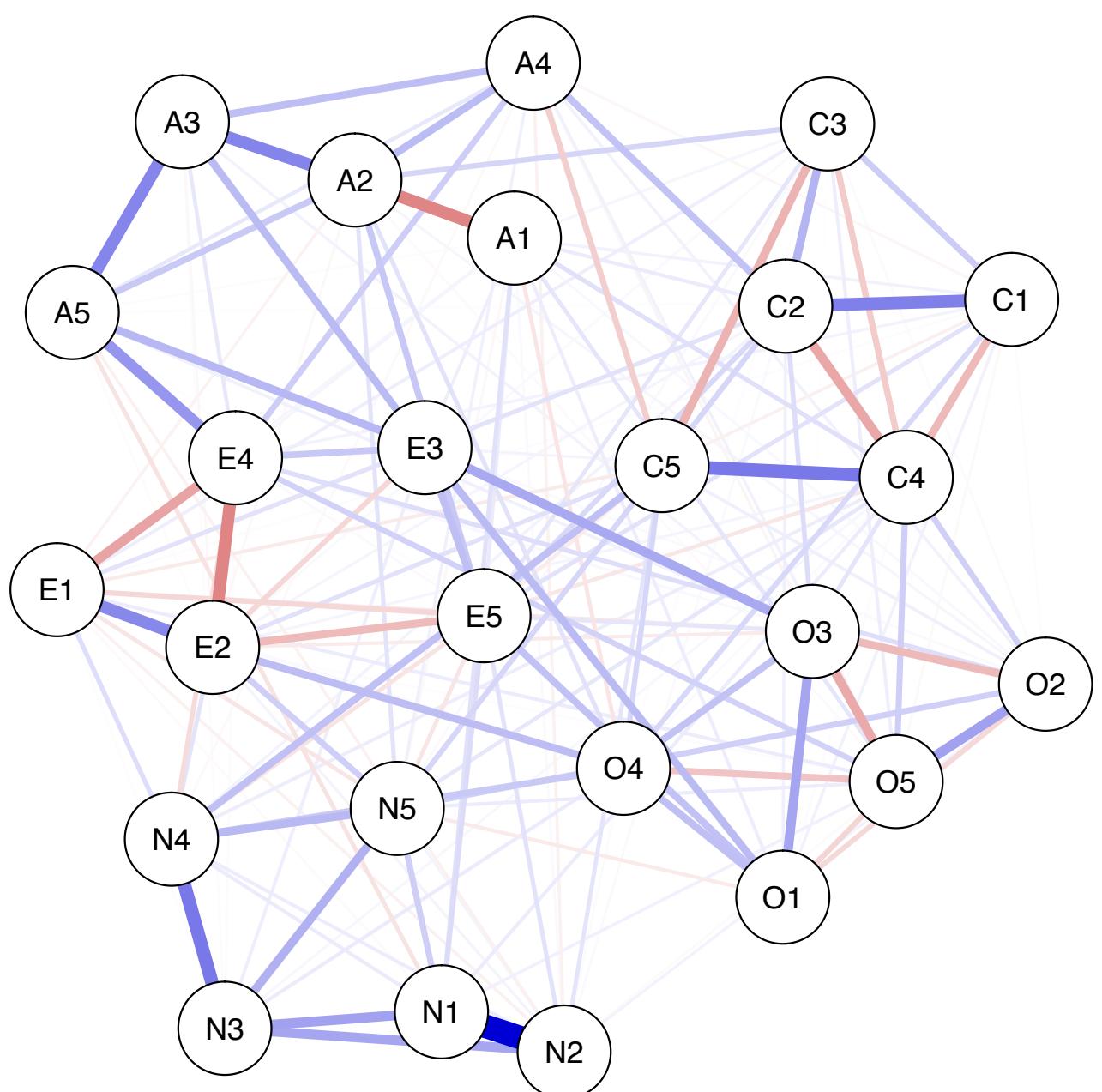
- Is edge N1-N2 *meaningfully* larger than edge N1-N3?
- Is node C4 *substantially* more central than node E2?



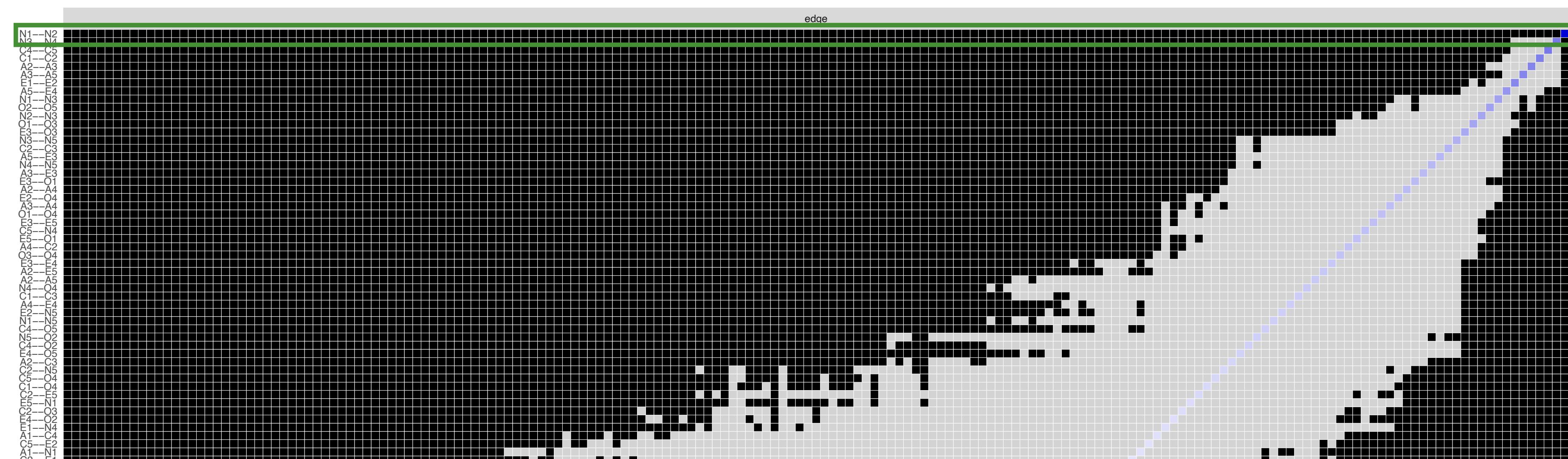
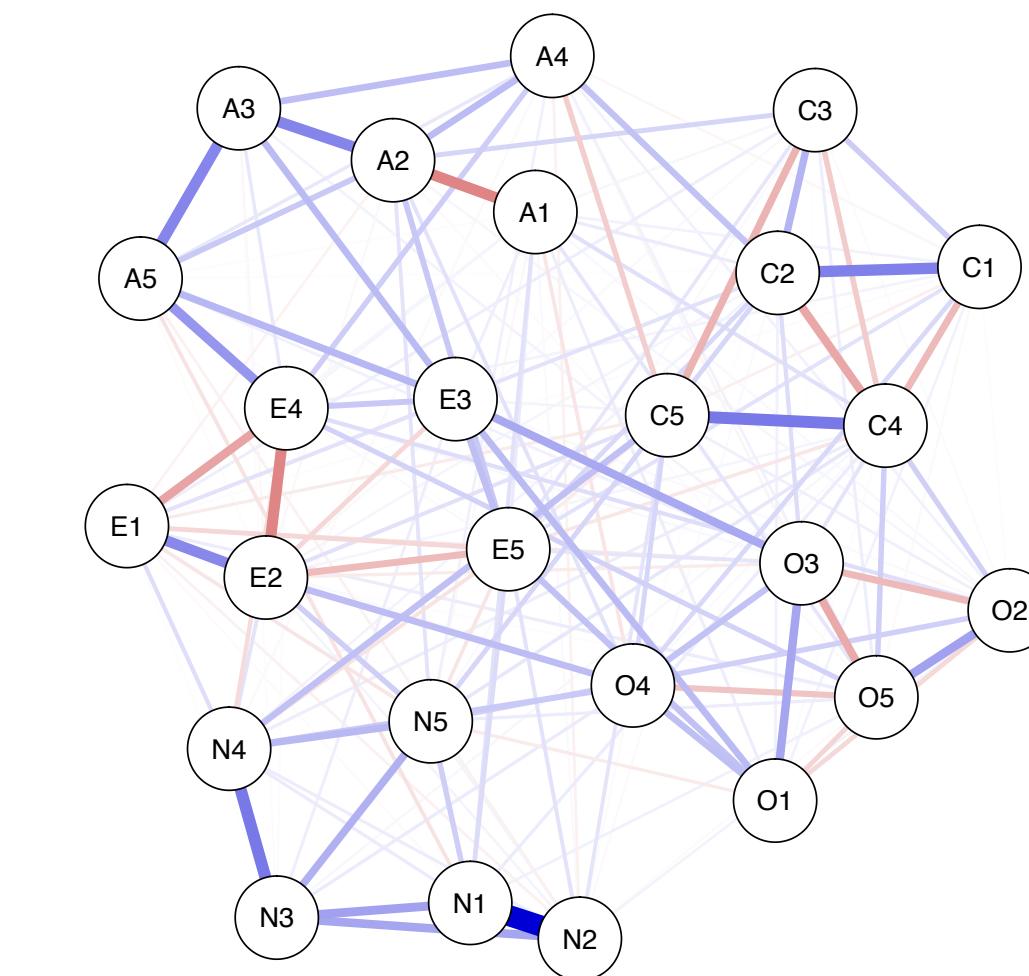
Accuracy and stability in networks



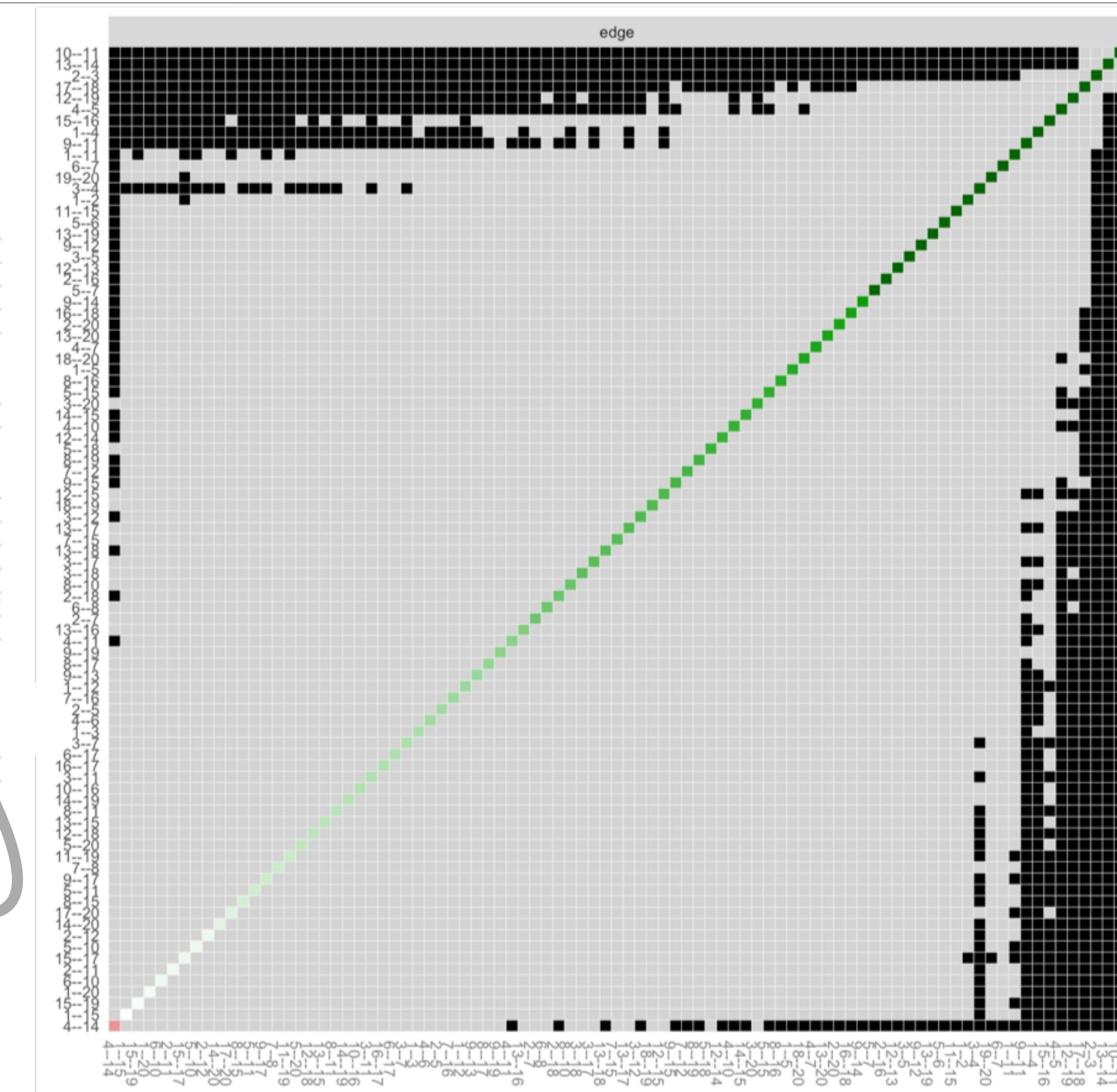
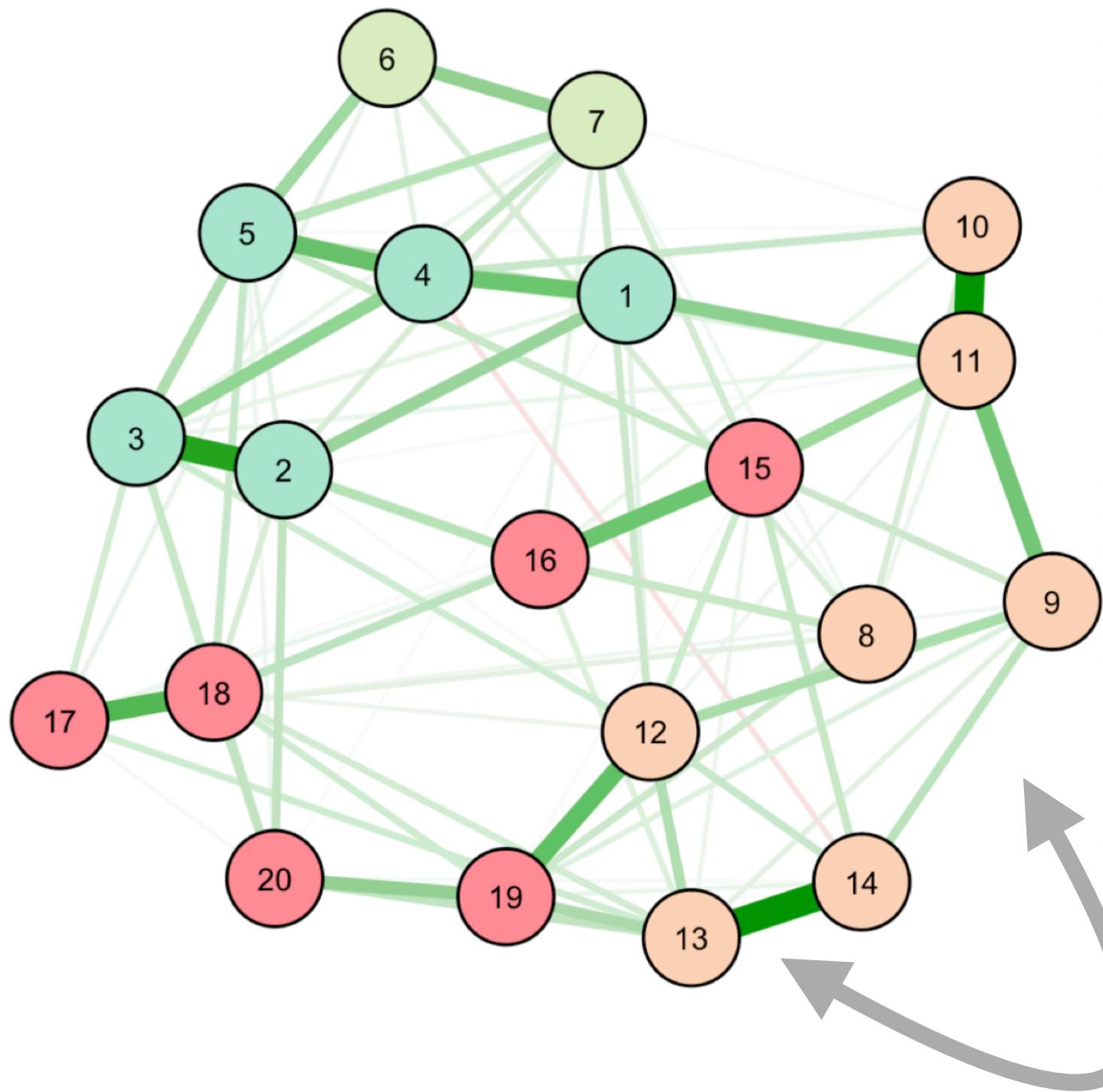
3. Difference test: edge weight difference test



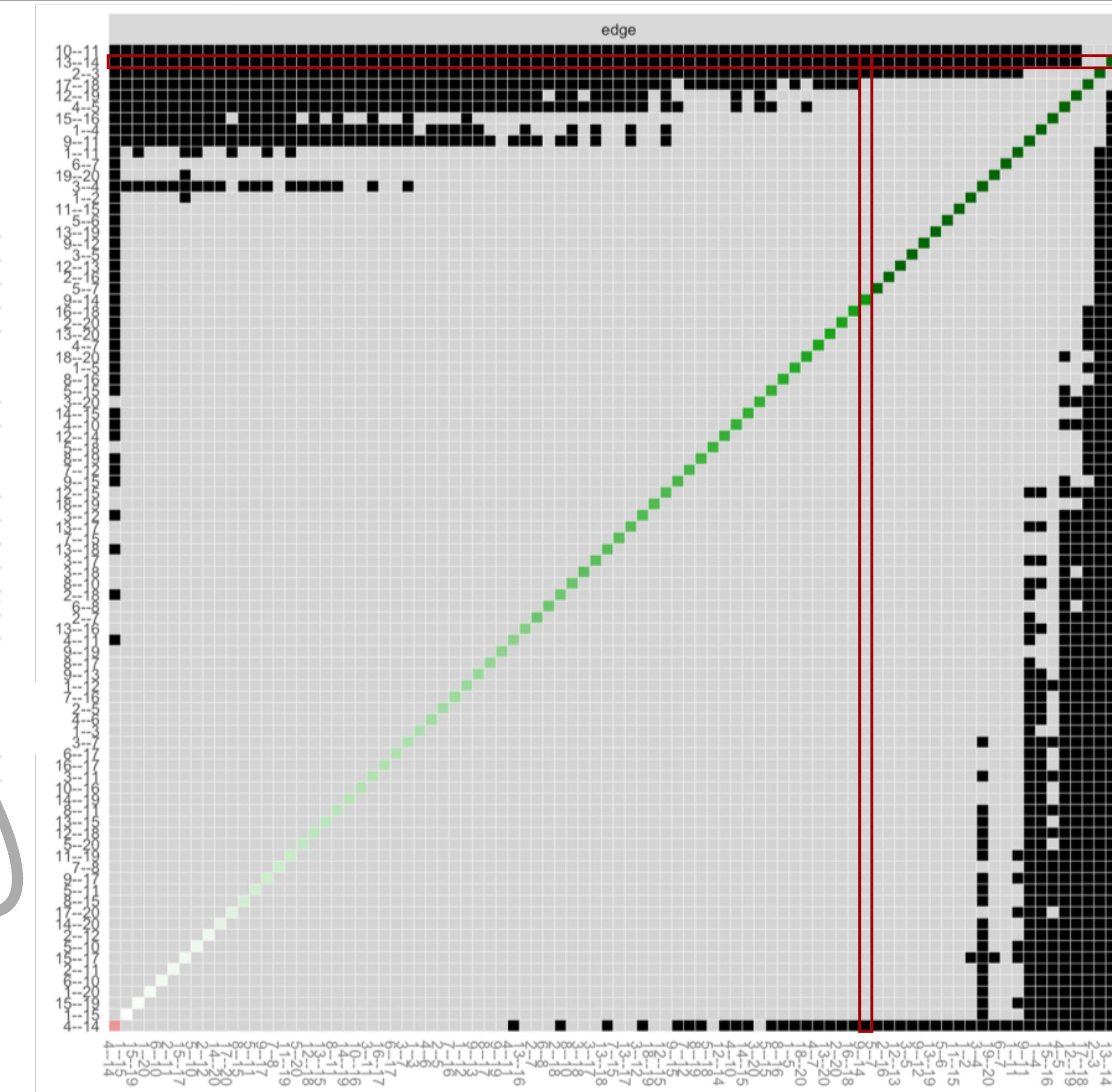
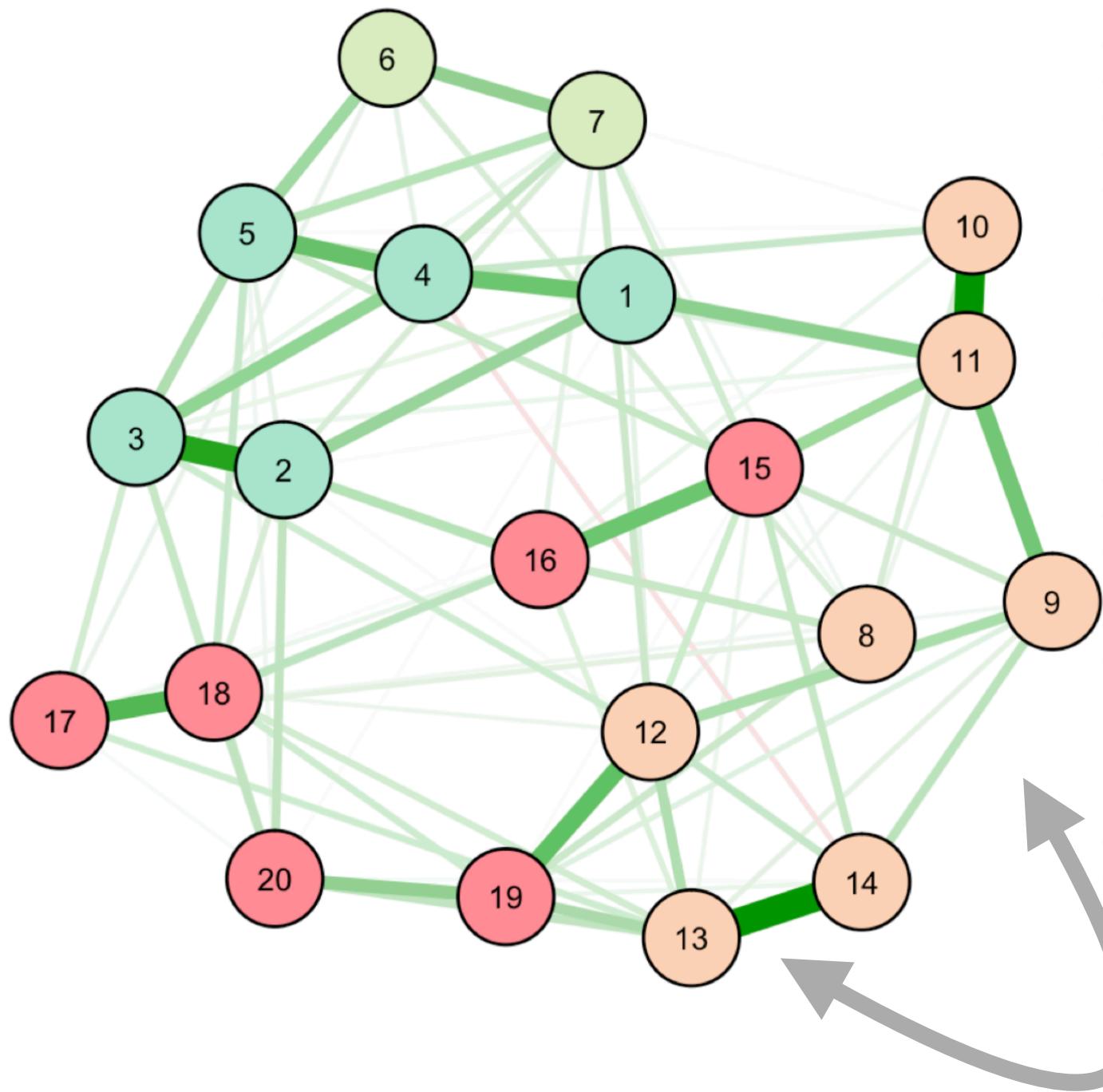
3. Difference test: edge weight difference test



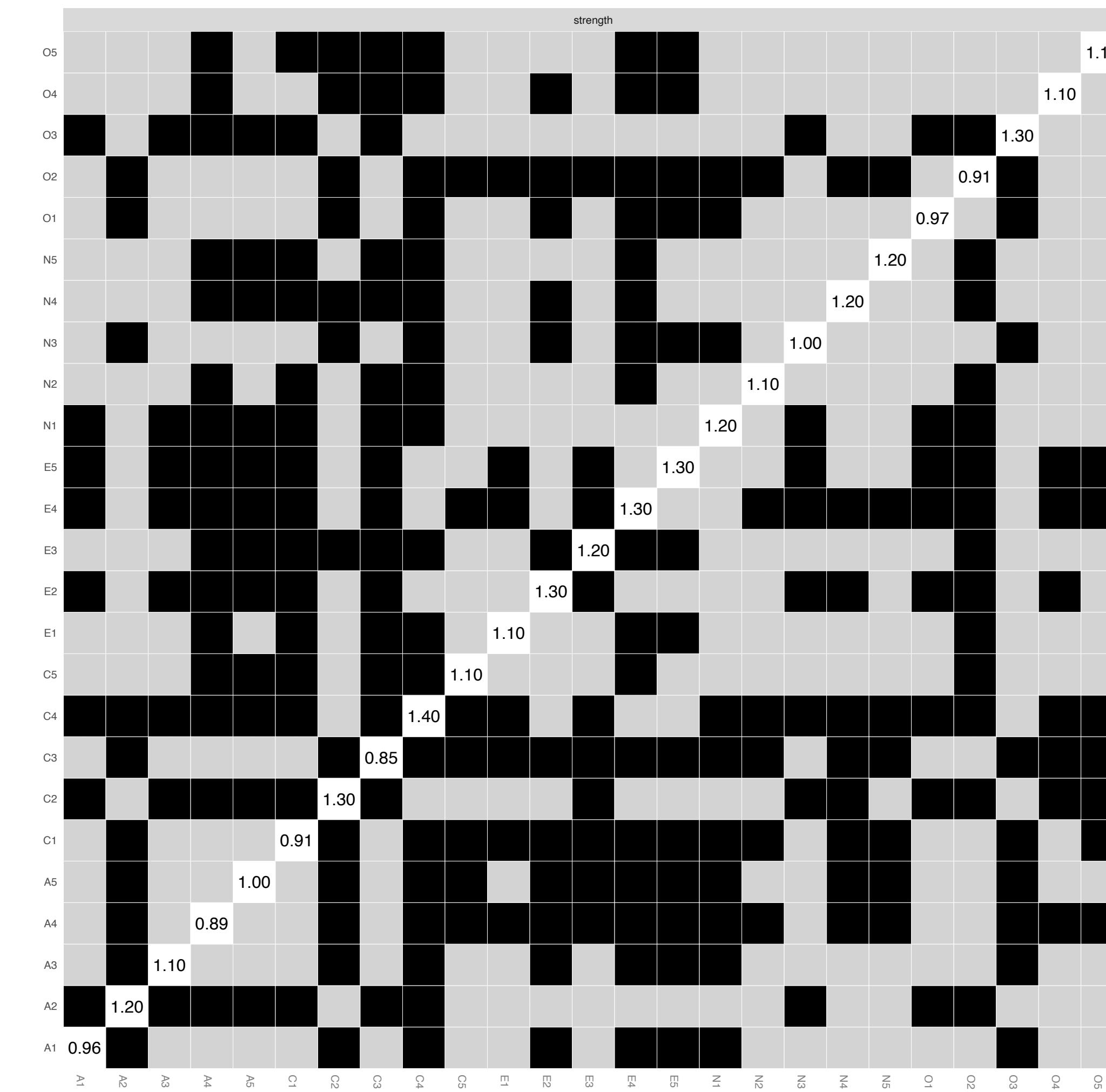
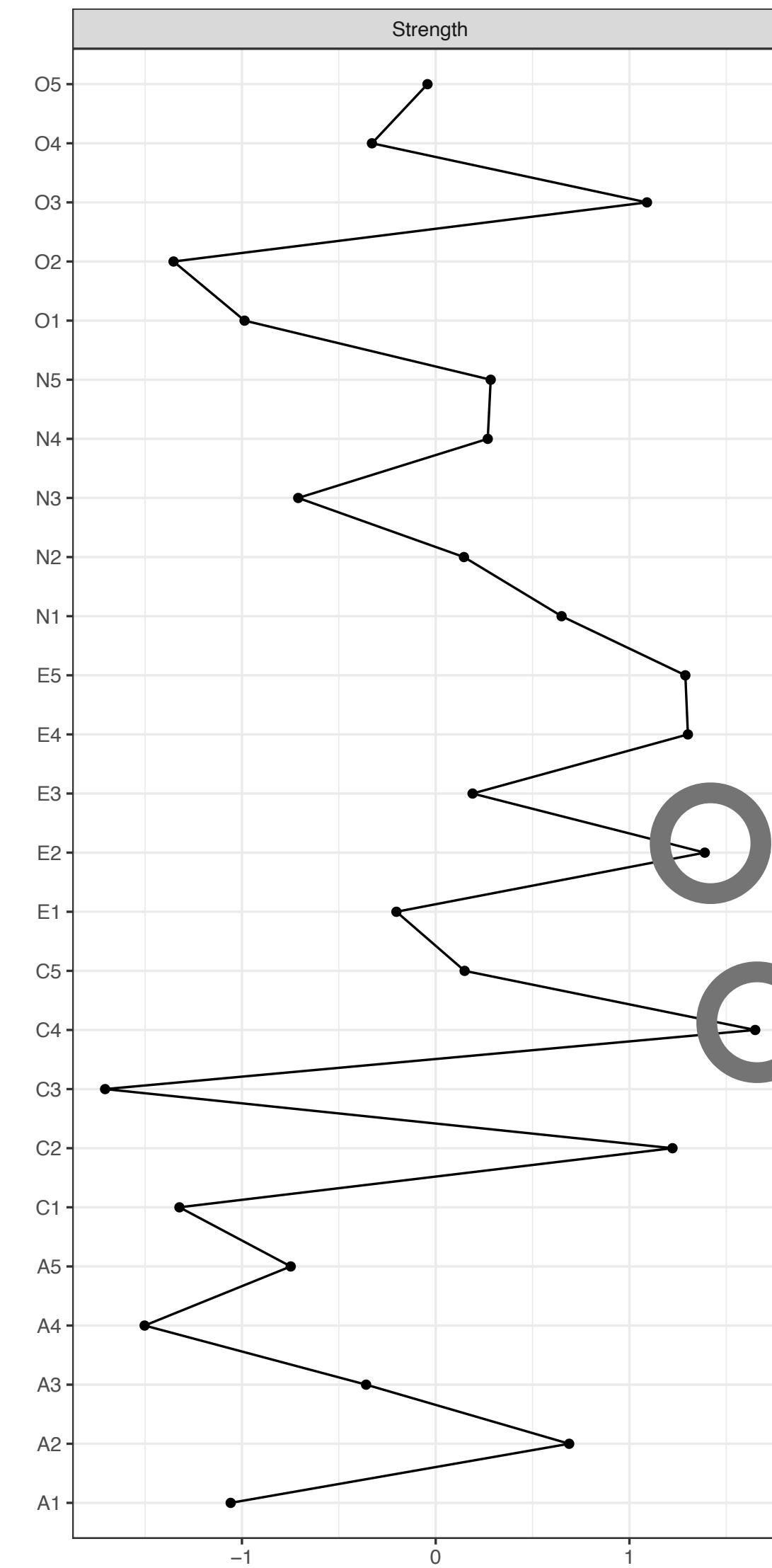
3. Difference test: edge weight difference test



3. Difference test: edge weight difference test

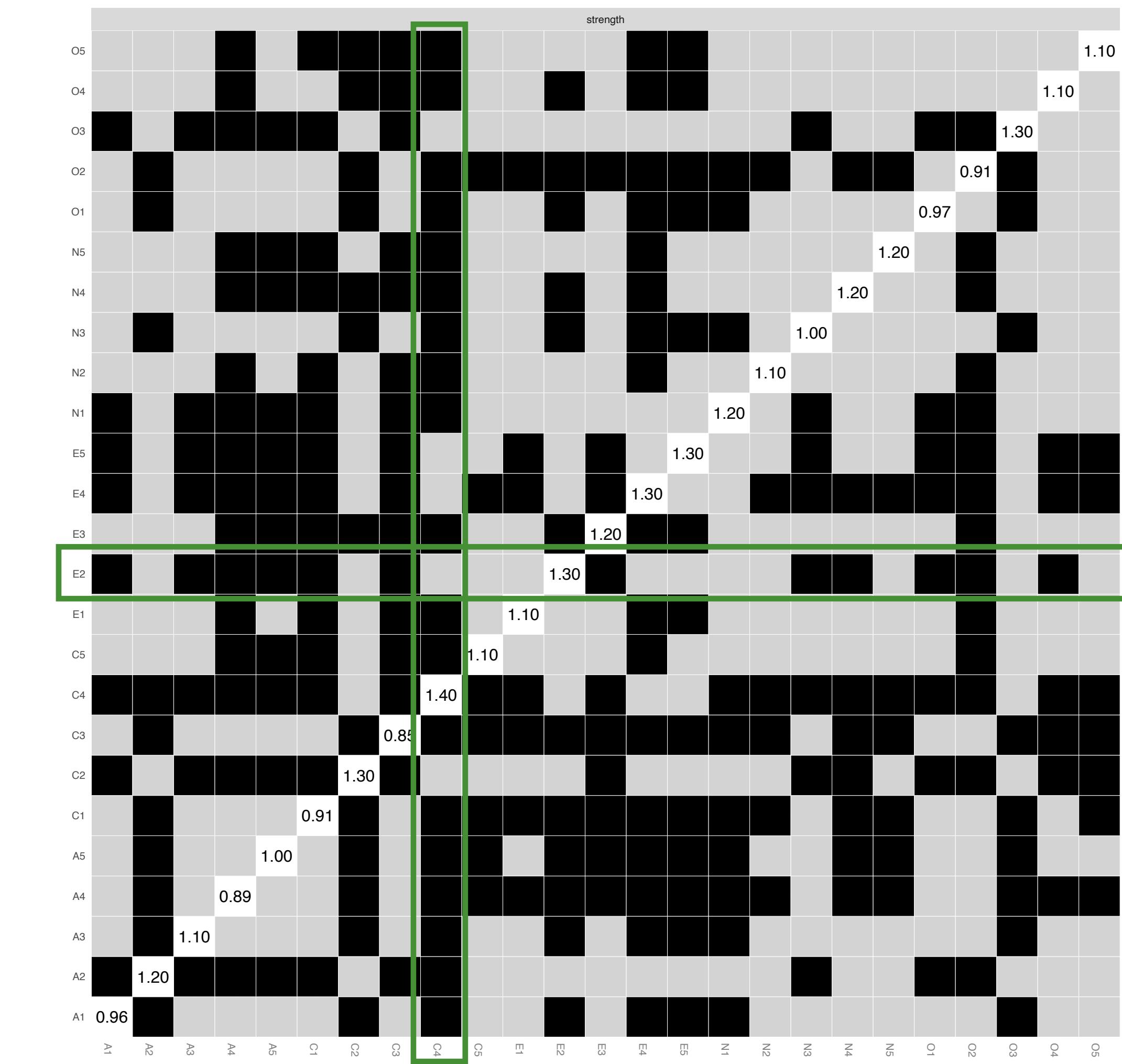
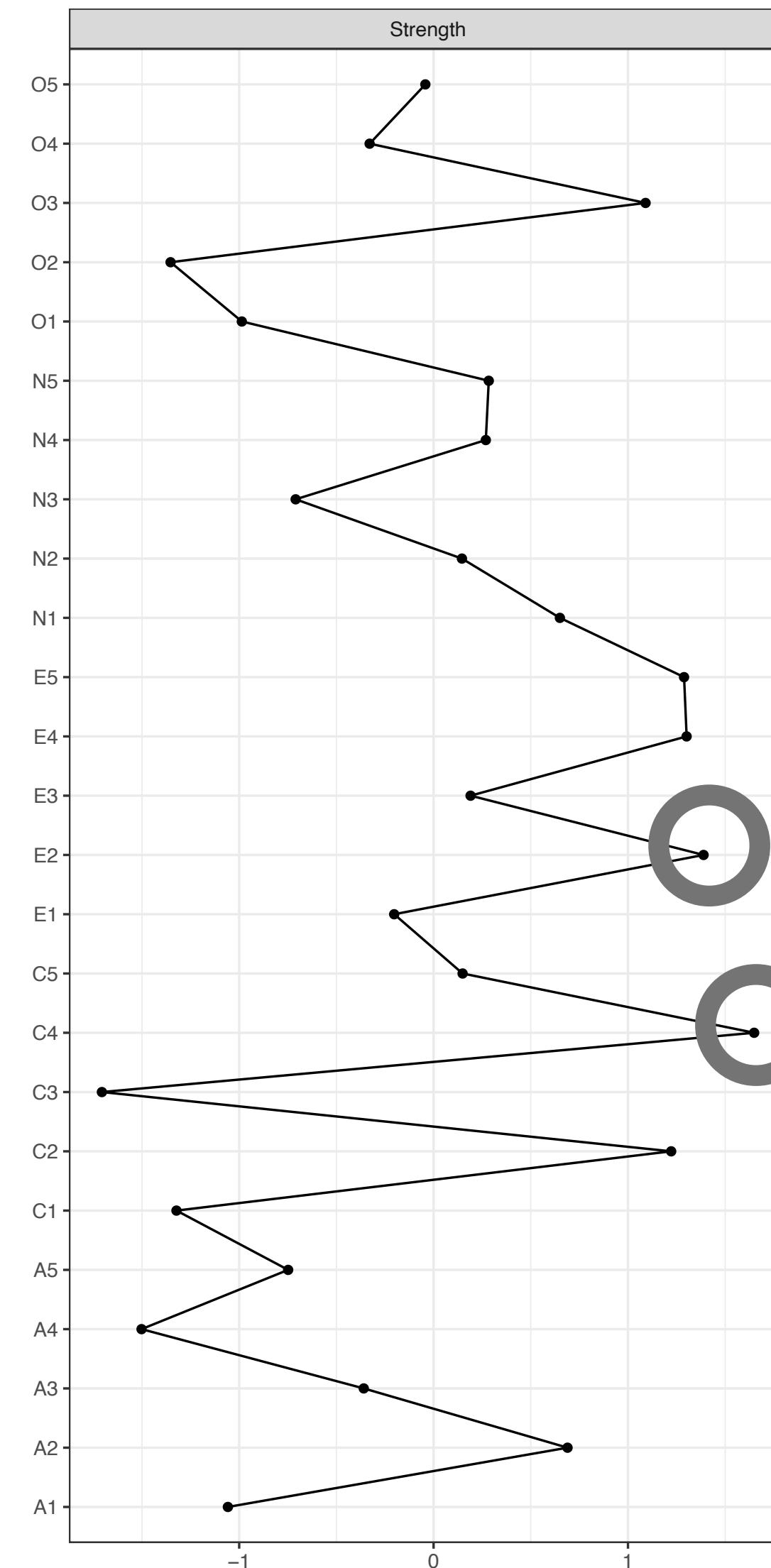


3. Difference test: centrality difference test



*note that bootnet() plots give unstandardized centrality estimates

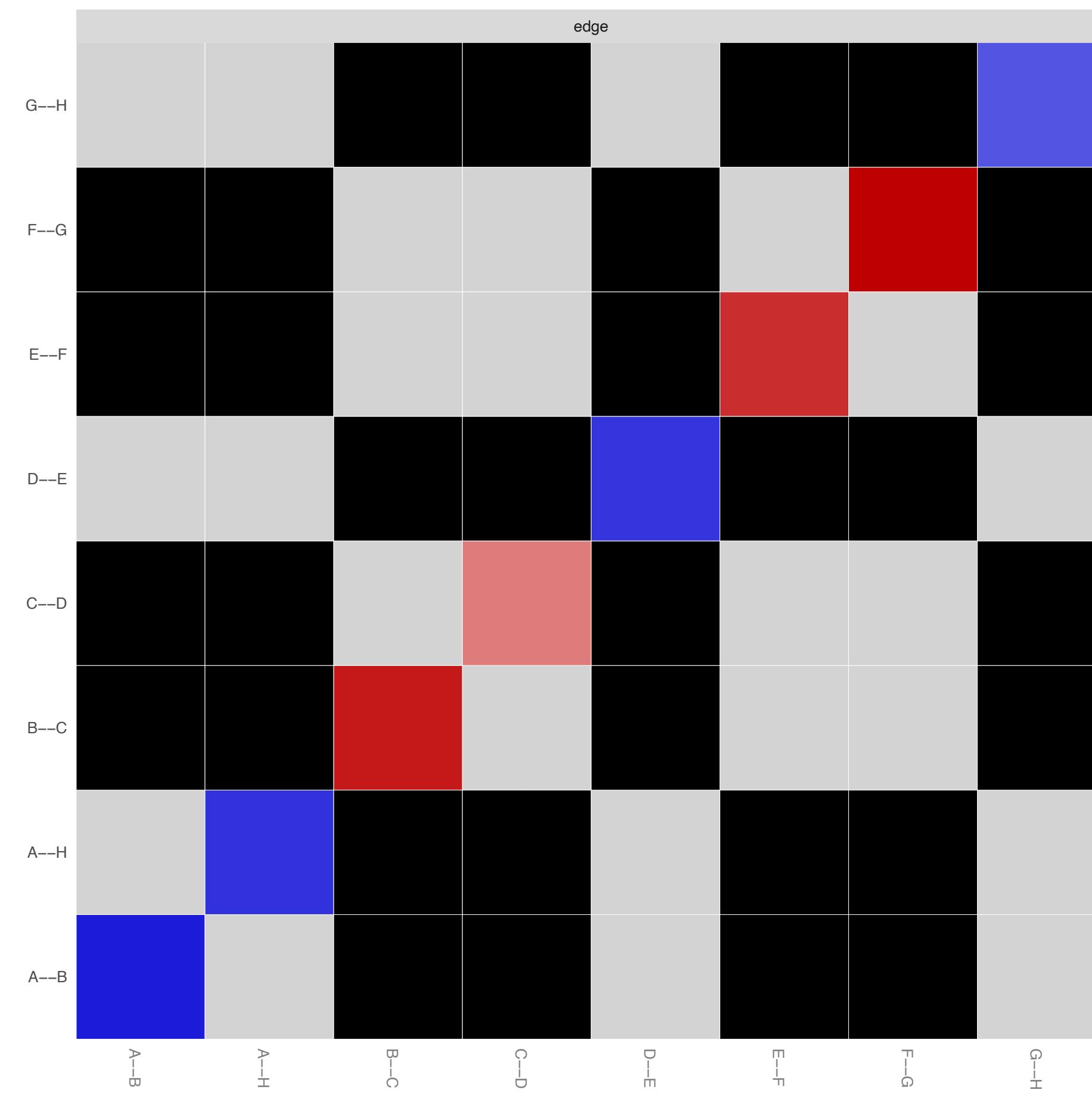
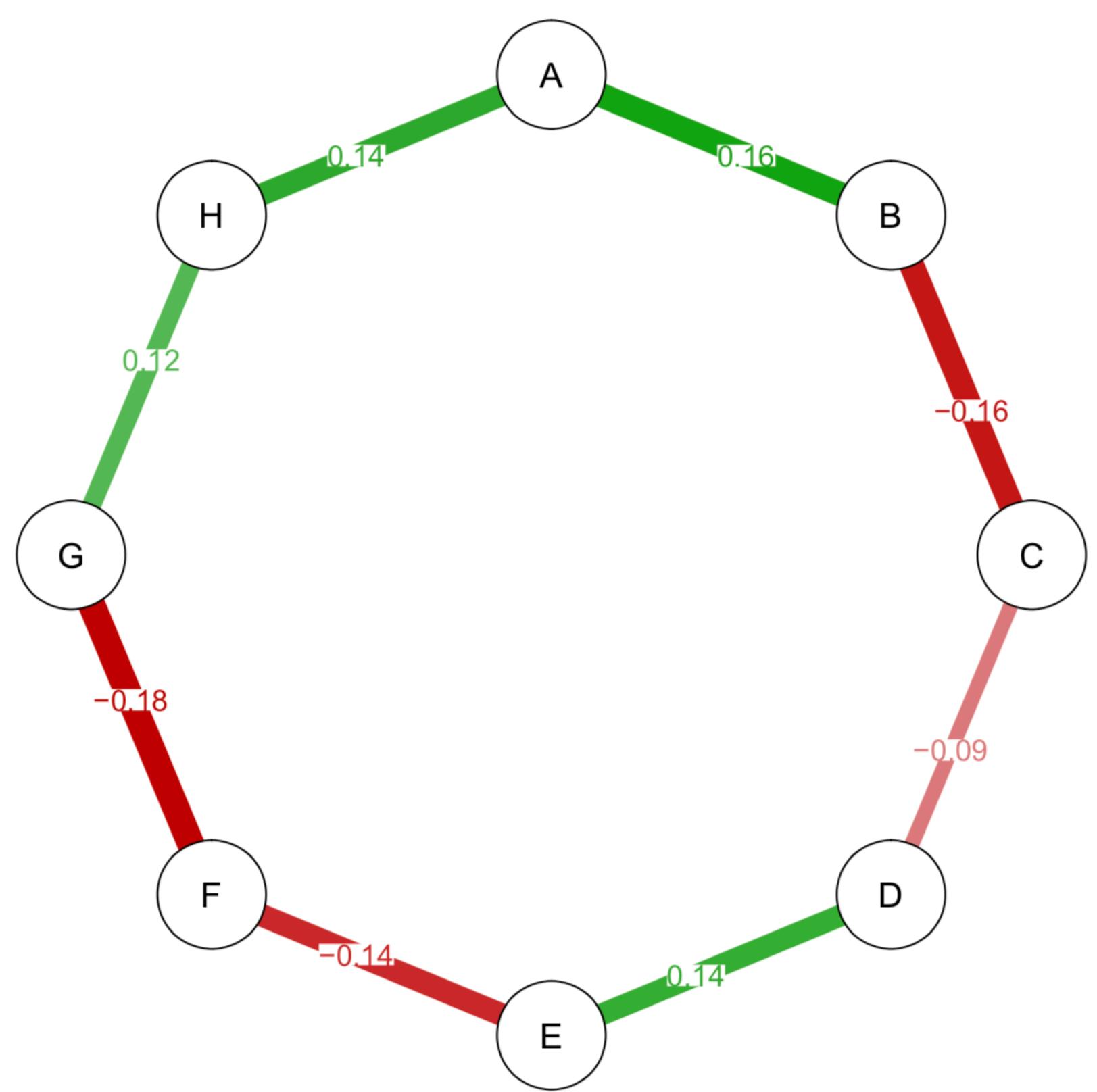
3. Difference test: centrality difference test



*note that bootnet() plots give unstandardized centrality estimates

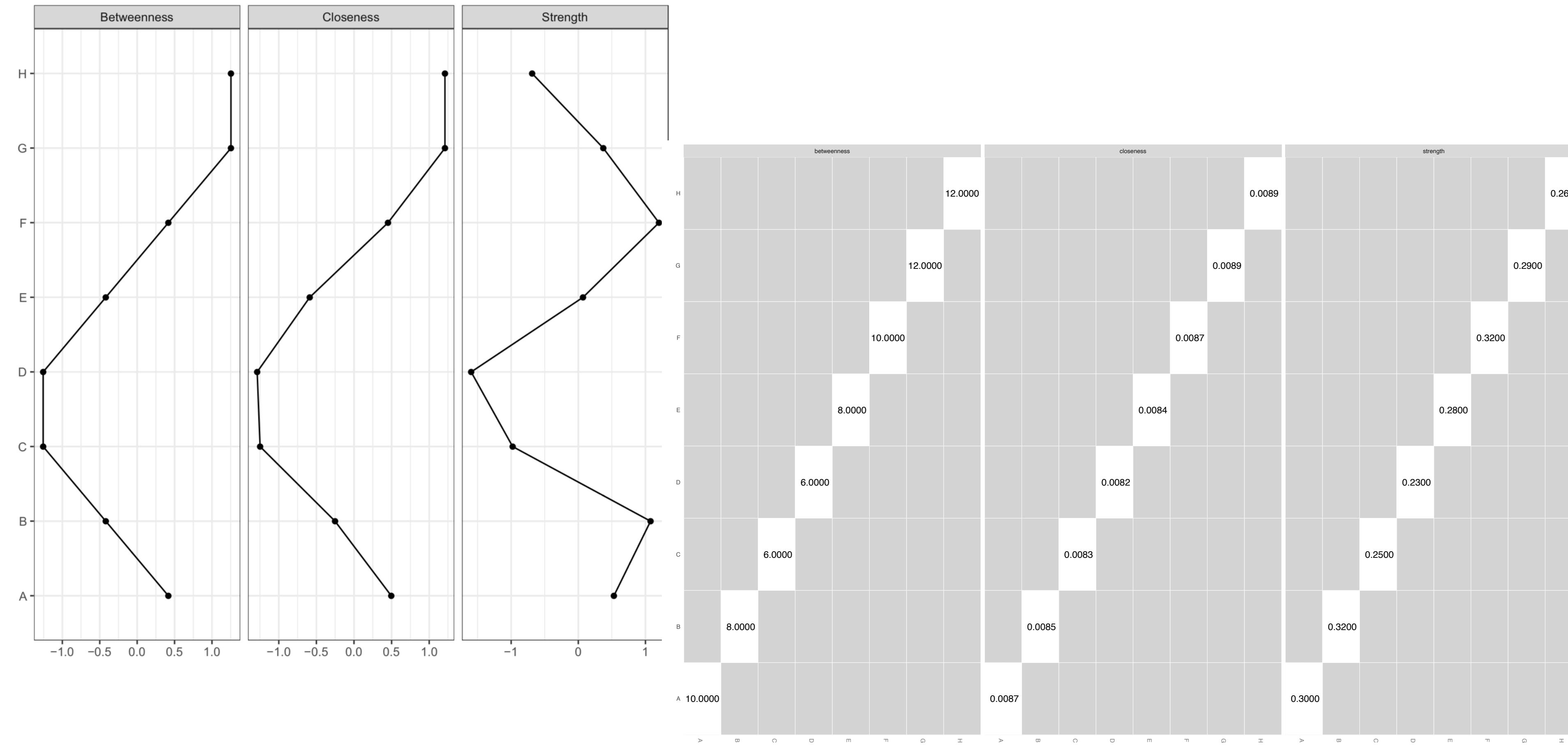
Simulation example: edge weight difference test

```
boot_difference <- bootnet(Network, nBoots = 5000, default = "glasso",
                            nCores = 8)
plot(boot_difference, statistics = "edge", plot = "difference", onlyNonZero = TRUE)
```



Simulation example: centrality difference test

```
boot_difference <- bootnet(Network, nBoots = 5000, default = "glasso",
                            nCores = 8)
plot(boot_difference, statistics = c("betweenness", "closeness", "strength"), plot = "difference")
```



Recap

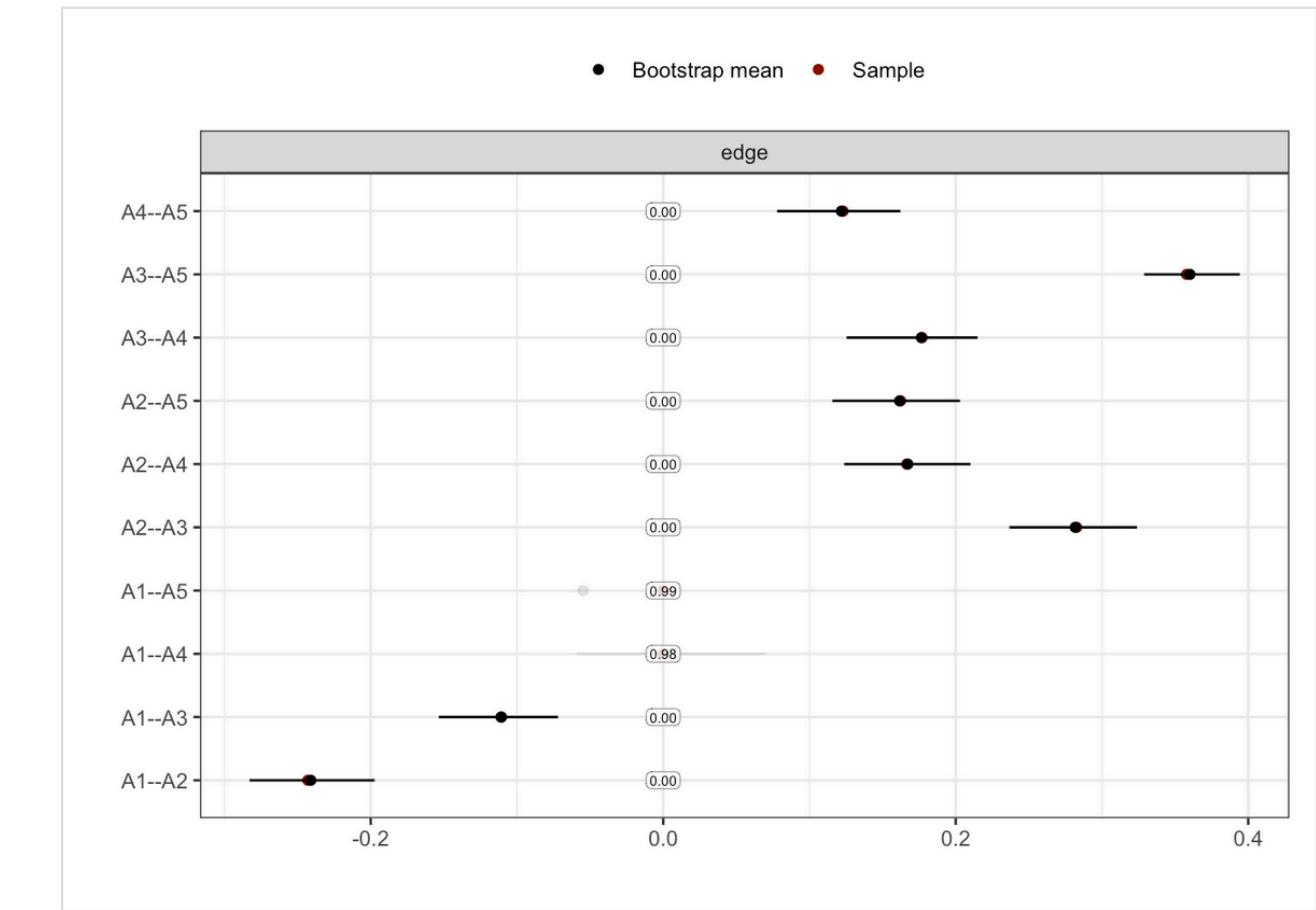
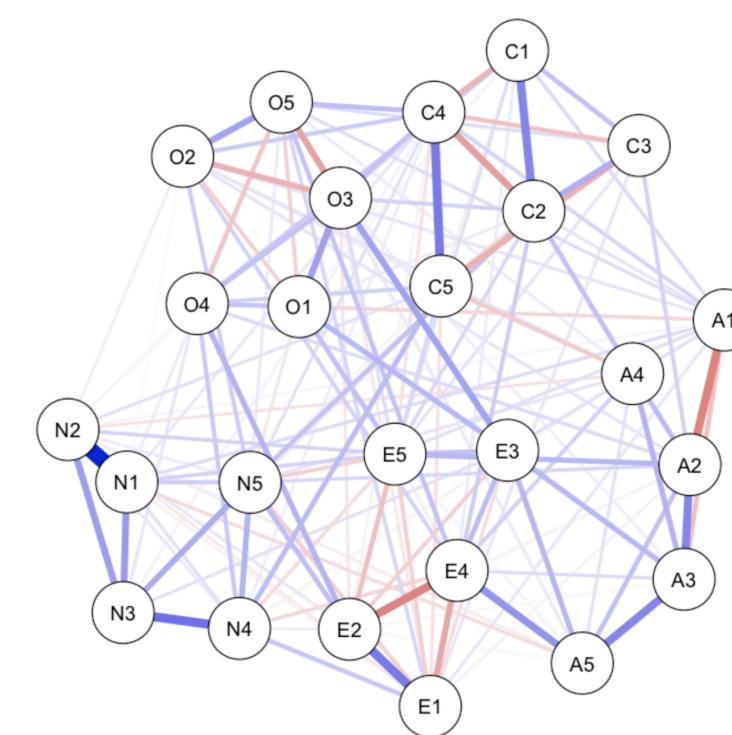
- Psychological networks involve *estimating a statistical model on data* that can be represented as a *weighted network between observed variables*
- For statistical parameters or test statistics (including network parameters!) it is important to understand how precisely they are estimated
- Investigating the accuracy and stability of the estimated network parameters helps us to understand the stability of the network models
- **This is crucial to understanding replicability**



replicability

From accuracy and stability to replicability

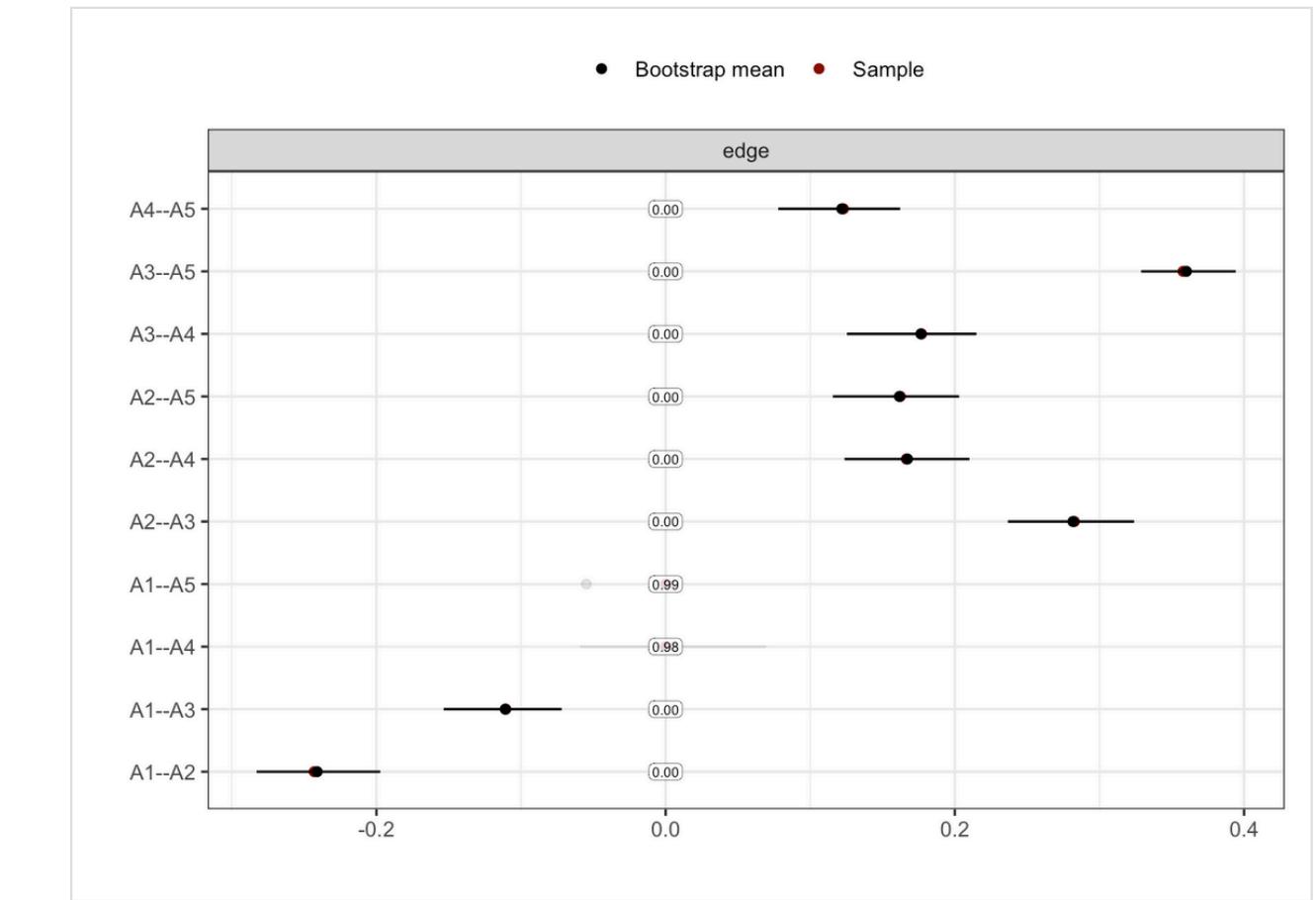
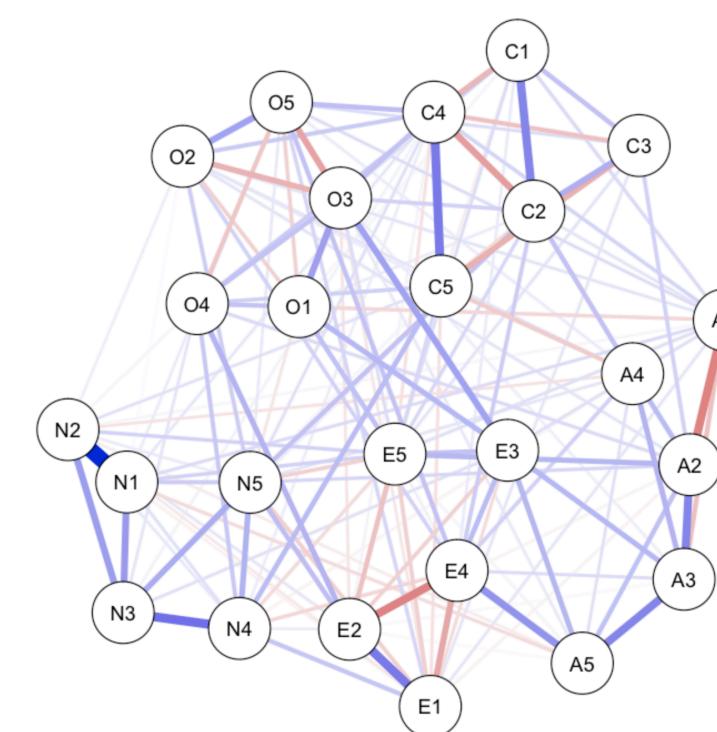
	X1	X2	X3	X4
●				
●				
...				
500				



- The edge weights in my network are estimated *accurately* and the centrality indices are *stable*. Will my network replicate?

From accuracy and stability to replicability

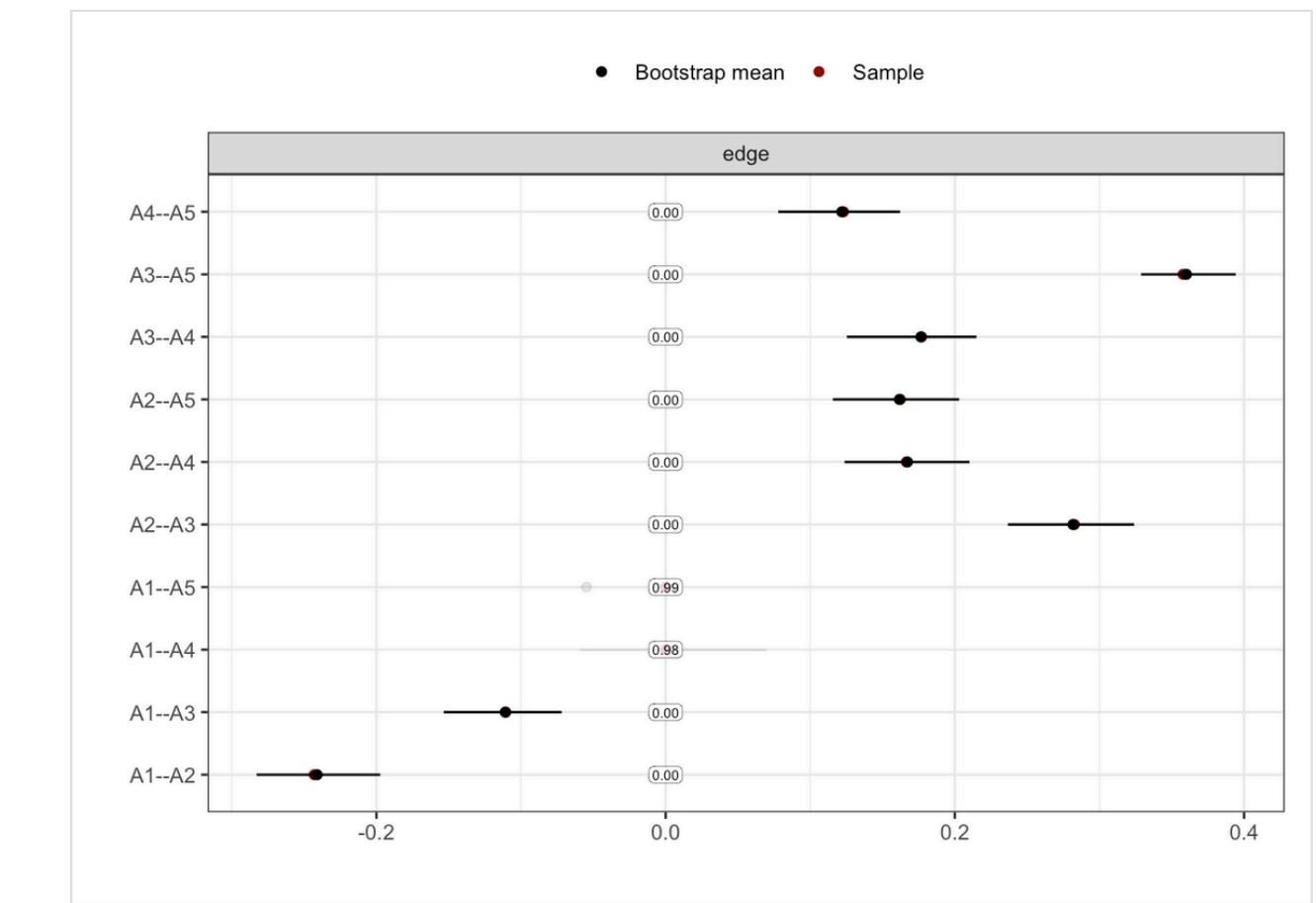
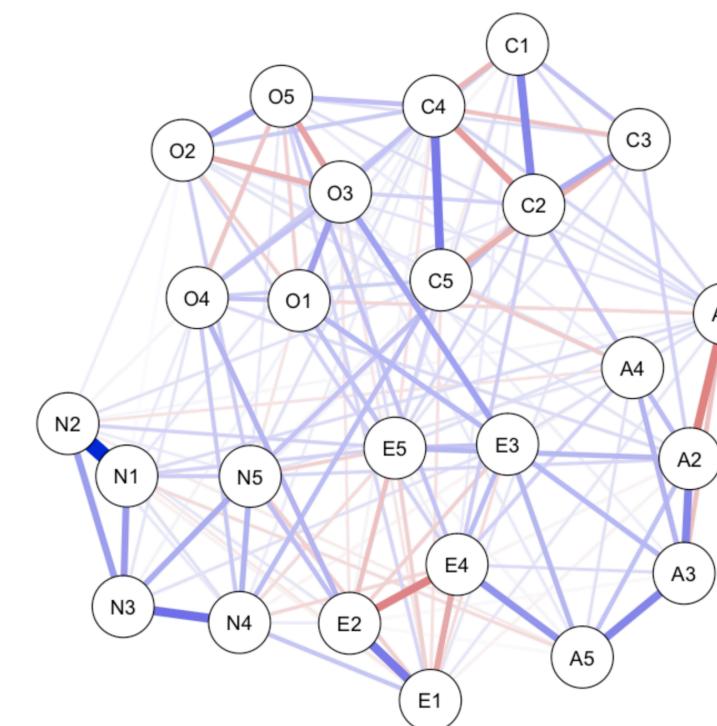
	X1	X2	X3	X4
●				
●				
...				
500				



- The edge weights in my network are estimated *accurately* and the centrality indices are *stable*. Will my network replicate?
 - Collect new data and compare networks
 - Explore replicability using functions ‘`netSimulator()`’ and ‘`replicationSimulator()`’ in `bootnet`

netSimulator()

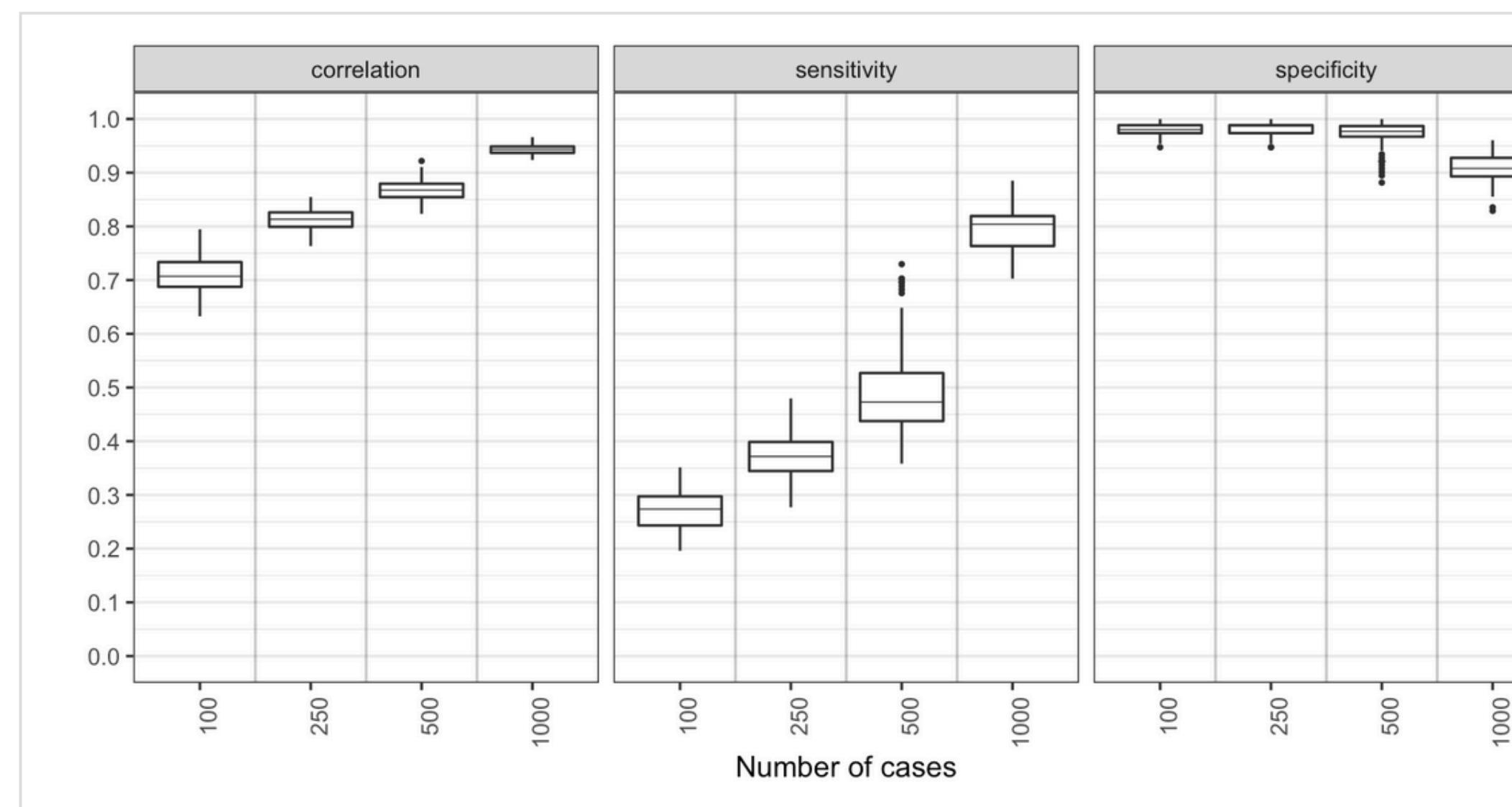
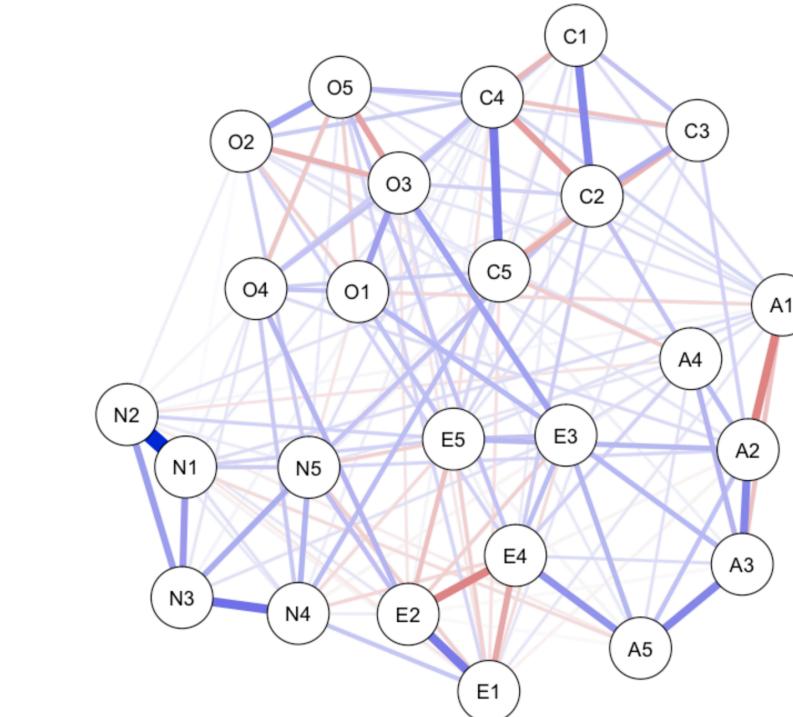
	X1	X2	X3	X4
1				
2				
...				
500				



- `netSimulator()` simulates the performance of an estimation method given an estimated network structure

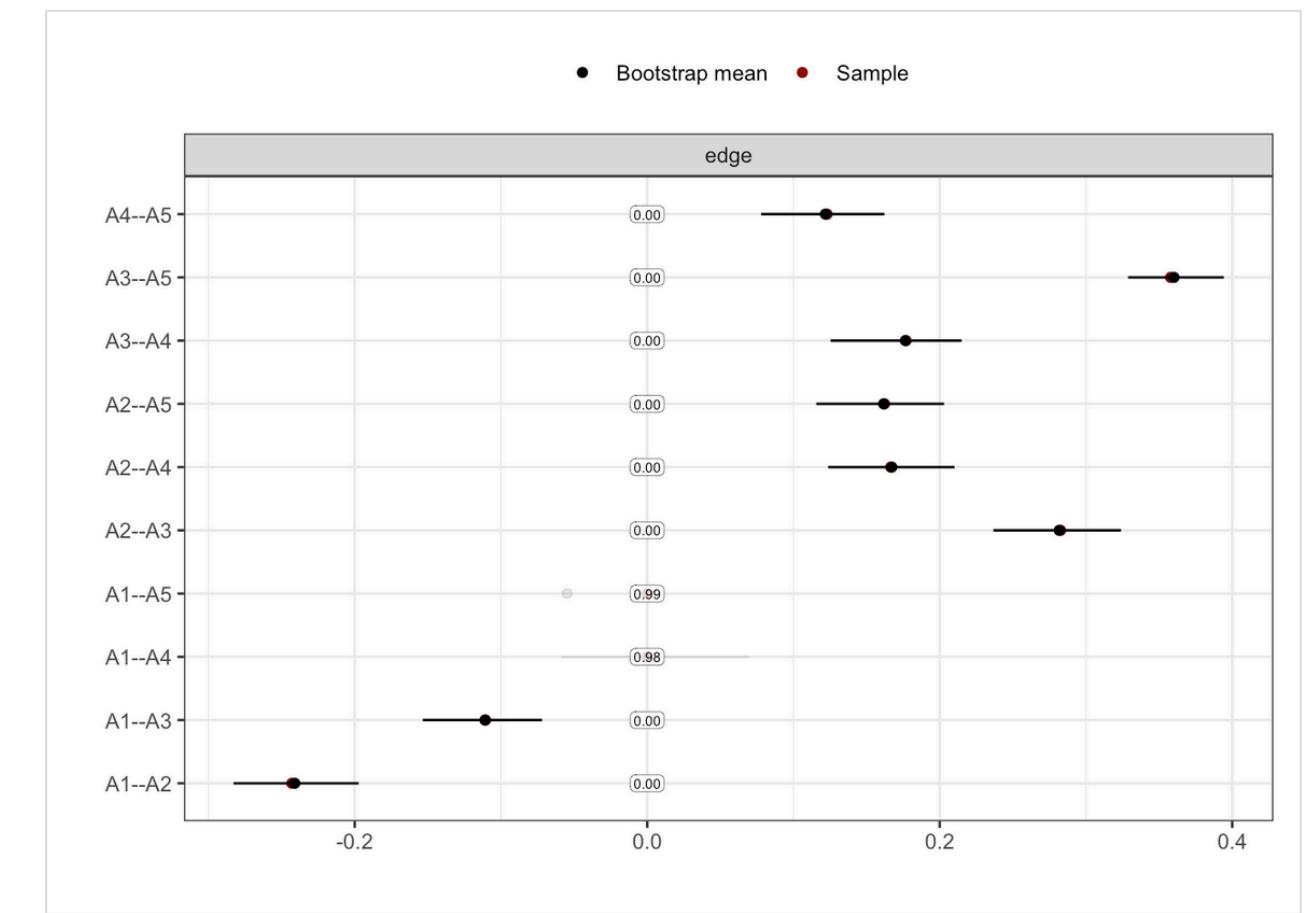
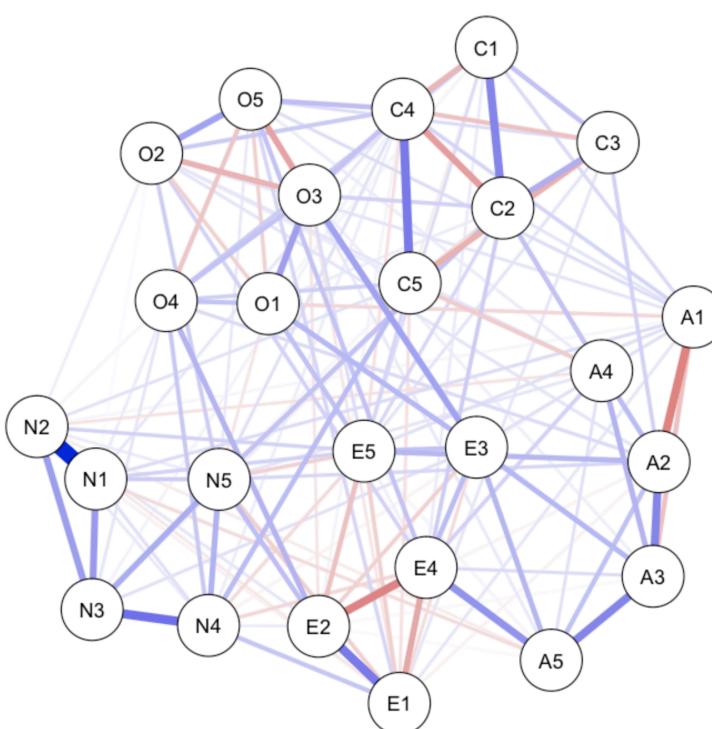
netSimulator()

```
Sim1 <- netSimulator(  
  input = net_modSelect, ← network structure  
  dataGenerator = ggmGenerator(),  
  nCases = c(100,250,500,1000), ← different sample sizes  
  nCores = 8,  
  nReps = 100,  
  default = "ggmModSelect", ← estimation method  
  stepwise = FALSE)  
  
plot(Sim1)
```



replicationSimulator()

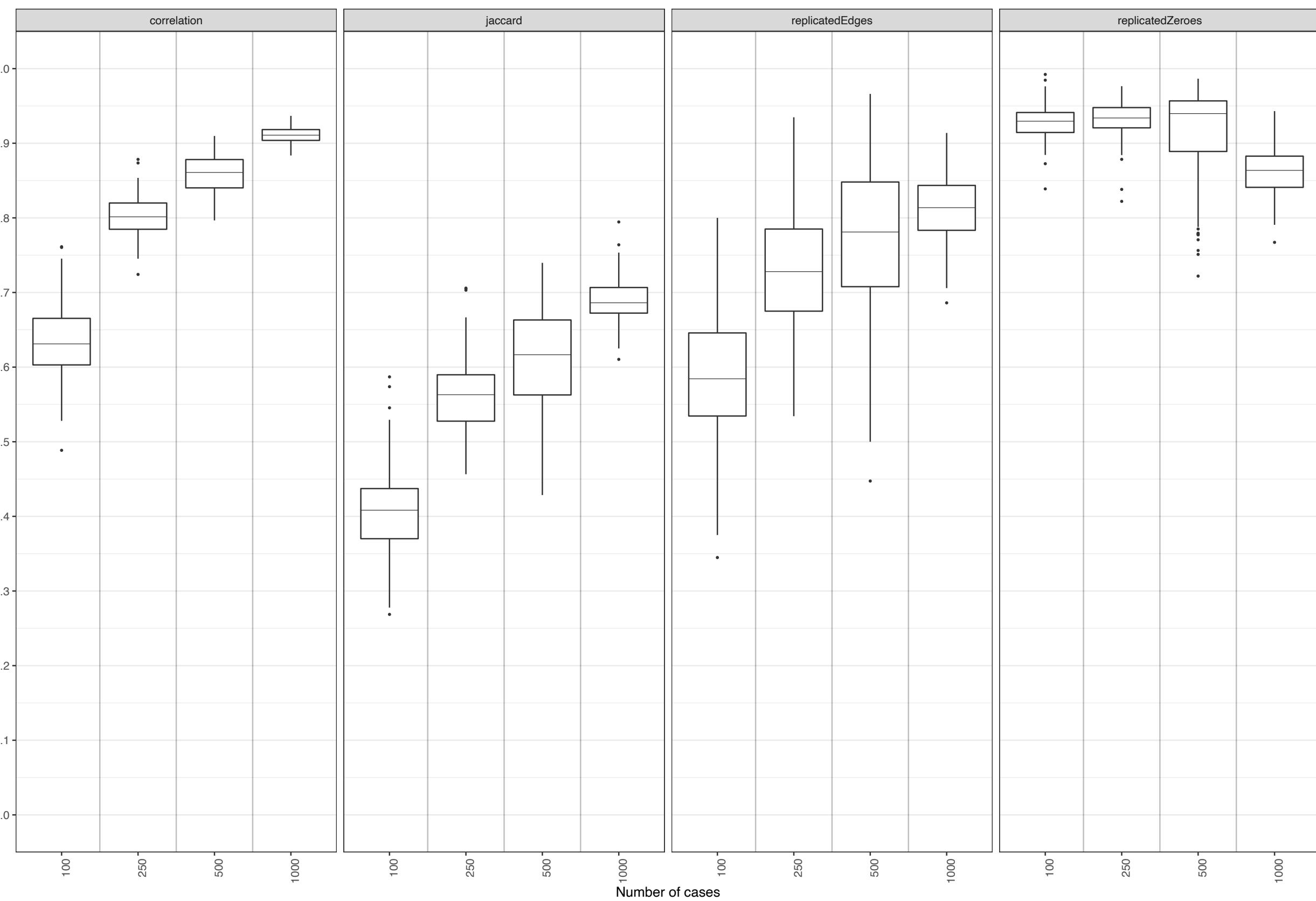
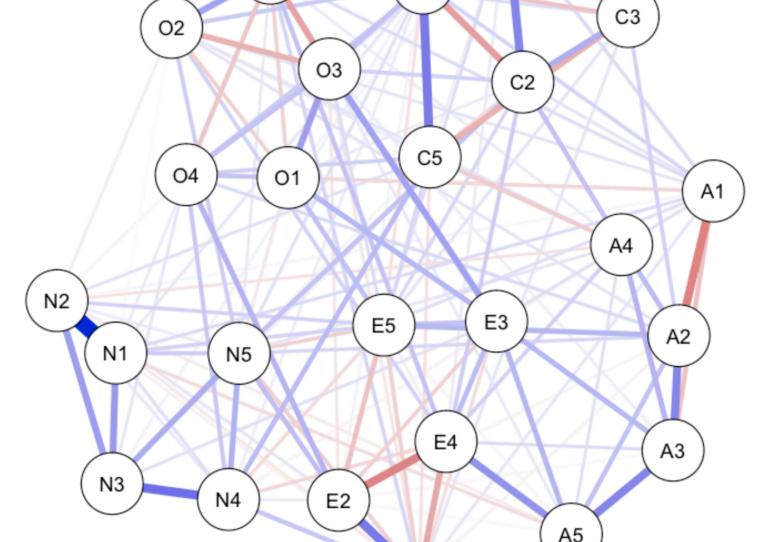
	X1	X2	X3	X4
●				
●				
...				
500				



- Similar to `netSimulator()` but simulates two dataset from the estimated network, treating the second as a replication dataset

replicationSimulator()

```
SimRep <- replicationSimulator(  
  input = net_modSelect,  
  dataGenerator = ggmGenerator(),  
  nCases = c(100,250,500,1000),  
  nCores = 8,  
  nReps = 1000,  
  default = "ggmModSelect",  
  stepwise = FALSE)  
  
plot(SimRep)
```

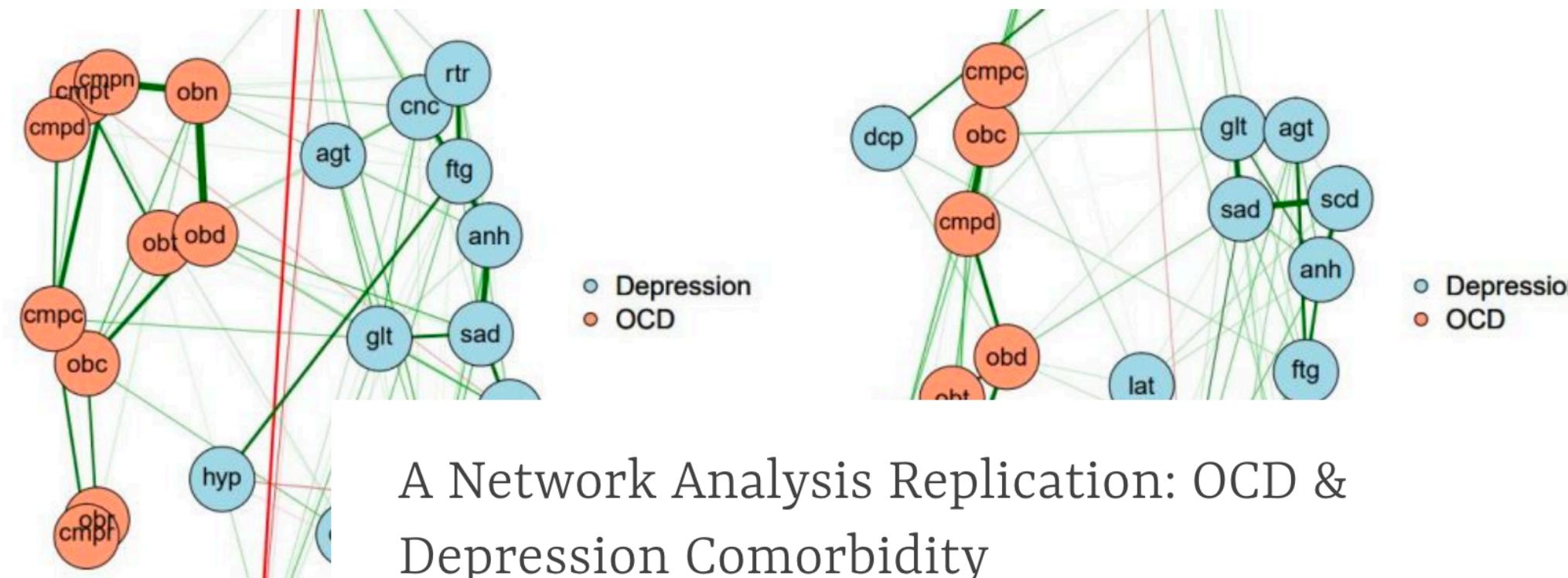


Note on replicability

- Model accuracy and stability are *necessary* for the networks to replicate...
- ... but not sufficient.

	same data	similar data	different data
same analysis	reproducibility	replicability	generalizability

Differences in replication

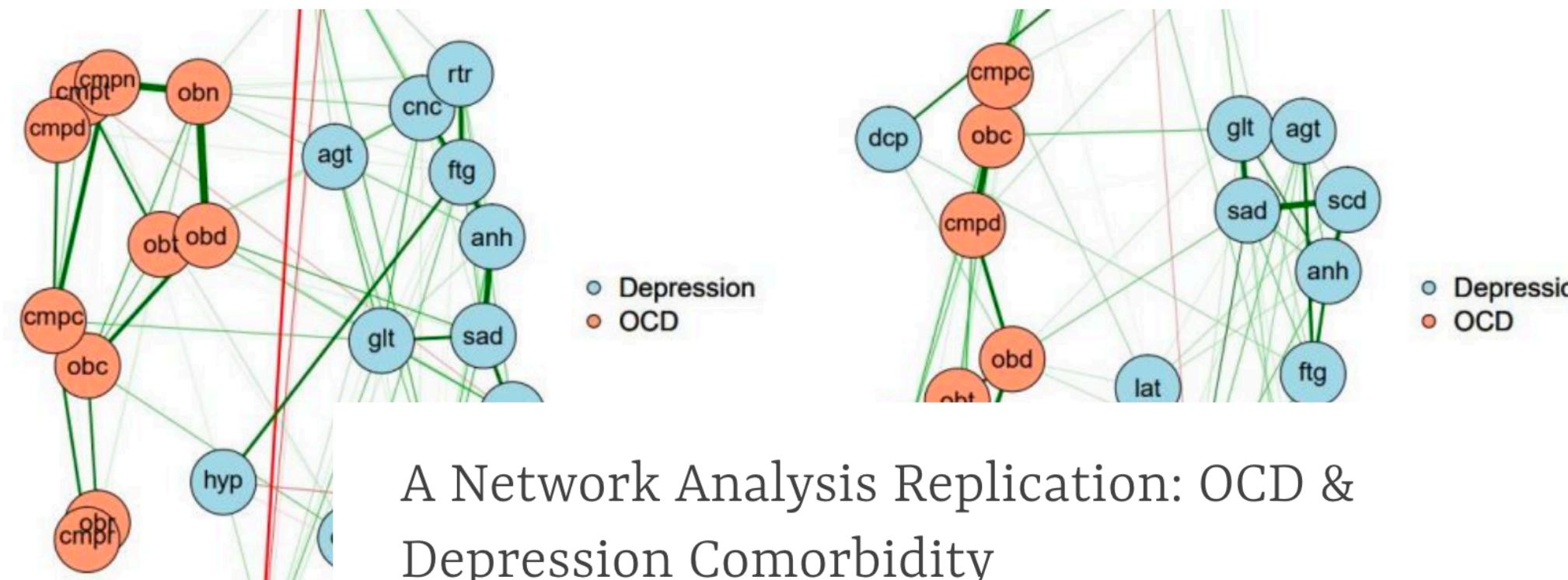


A Network Analysis Replication: OCD & Depression Comorbidity

Blog by Payton Jones (December 15 2018)

This certainly isn't a one-to-one replication — adolescents are not the same as adults! This is particularly true when we are dealing with OCD— several important differences have already been noted in the literature.

Differences in replication



Blog by Payton Jones (December 15 2018)

This certainly isn't a one-to-one replication — adolescents are not the same as adults! This is particularly true when we are dealing with OCD— several important differences have already been noted in the literature.

With this in mind, we were interested to see both the *similarities* between the networks, as well as the *differences*. The similarities (e.g., the parts that “replicate”) are likely to say something universal about how OCD and depression work, and the differences (e.g., the parts that “don’t replicate”) might tell us about what makes adults and adolescents unique (or they might be spurious – we’ll have to be careful).

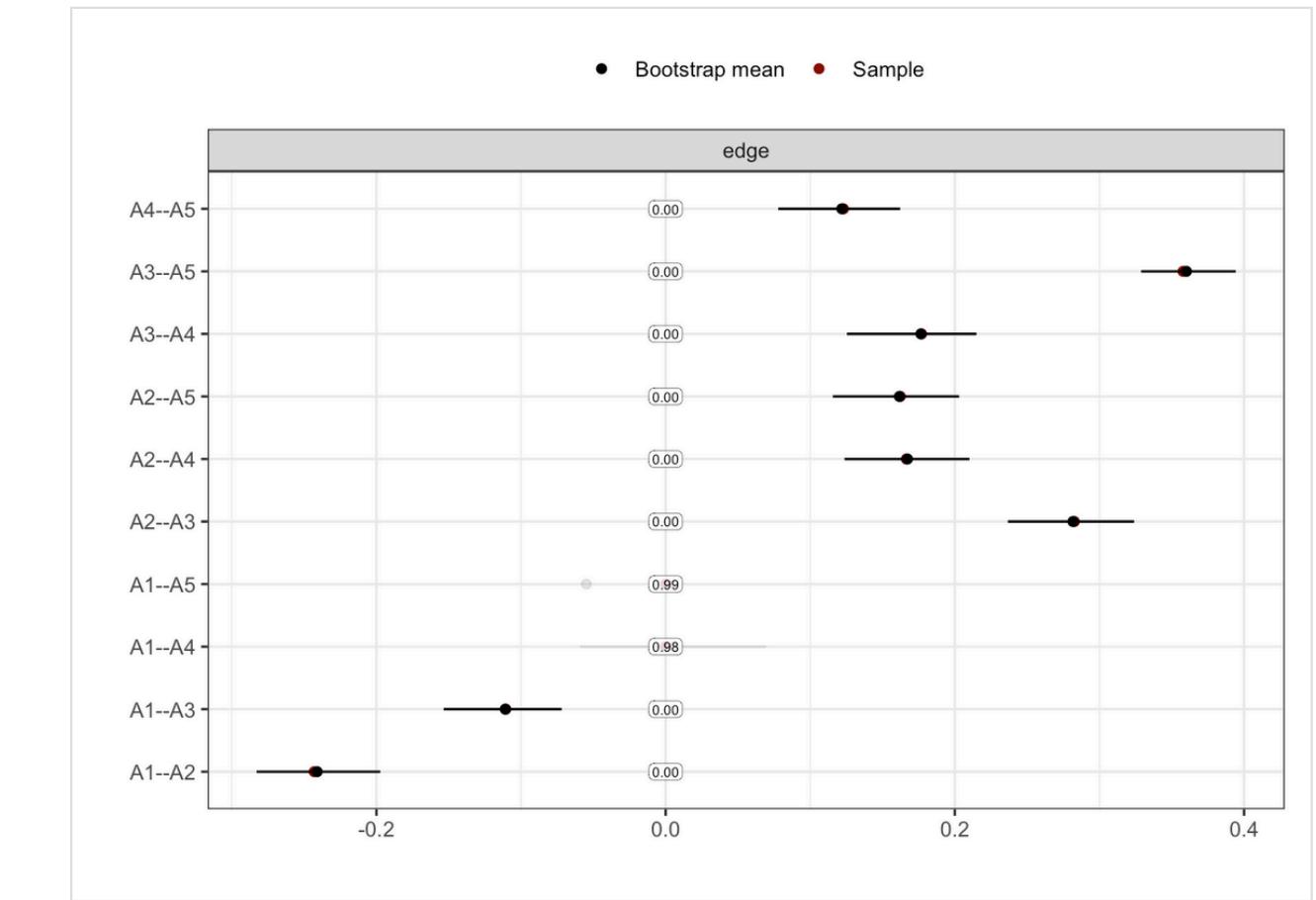
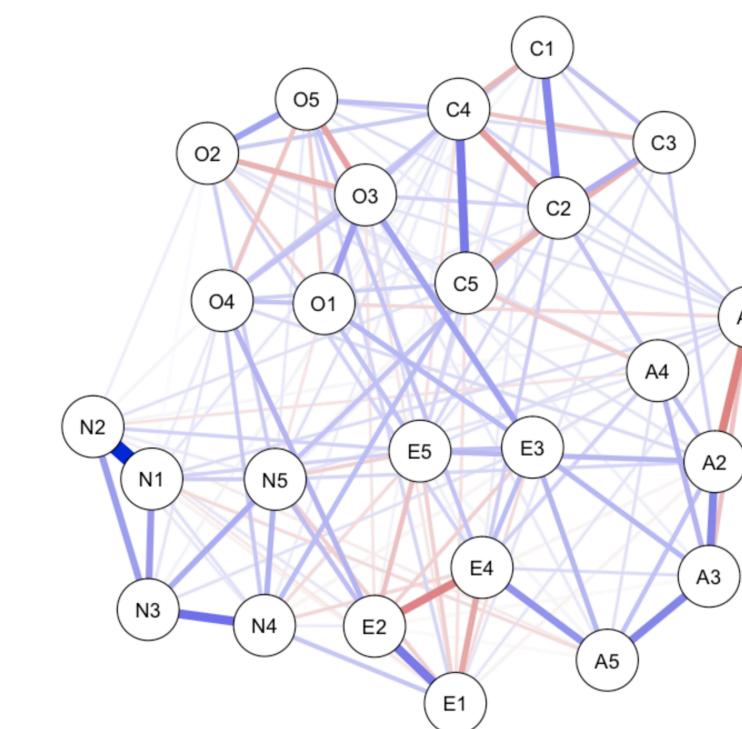
network comparison

network comparison test



Replicability

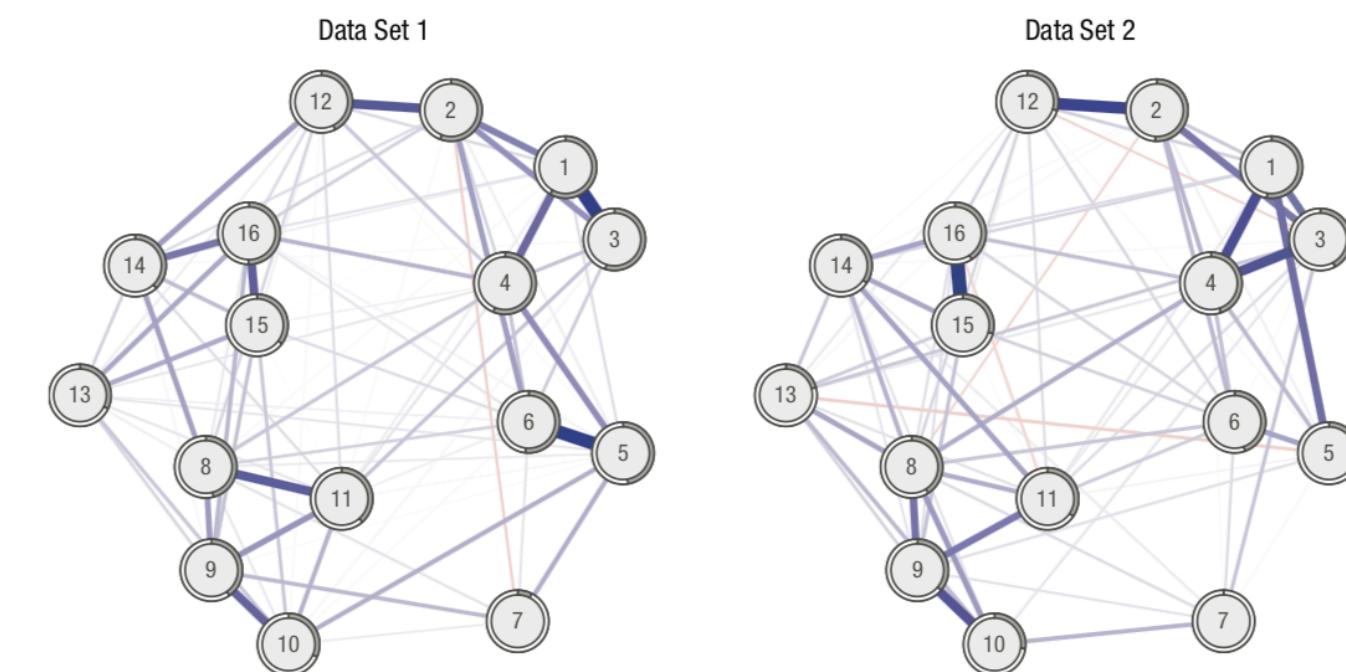
	X1	X2	X3	X4
1				
2				
...				
500				



- The edge weights in my network are estimated *accurately* and the centrality indices are *stable*. Will my network replicate?
 - Collect new data and compare networks
 - Explore replicability using functions ‘`netSimulator()`’ and ‘`replicationSimulator()`’ in `bootnet`

Network comparison

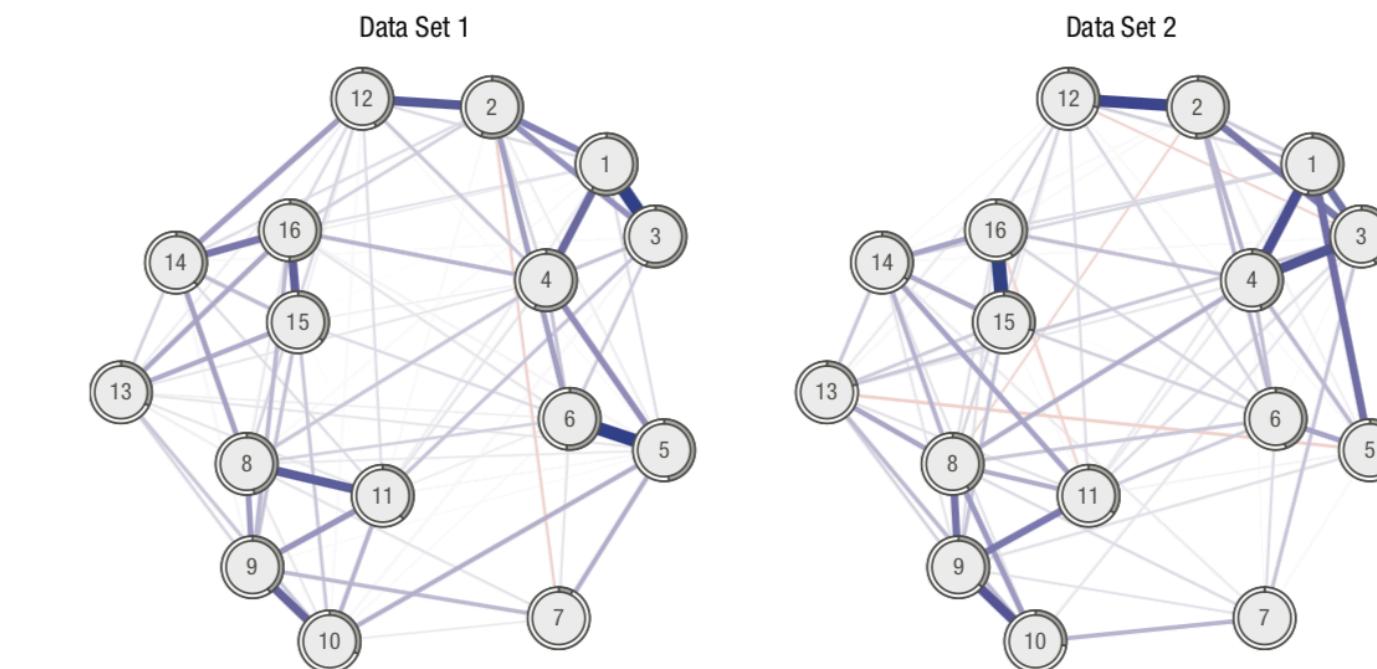
- Assess replicability
 - “*does the posttraumatic stress disorder network replicate across data sets?*”



Fried et al. 2018

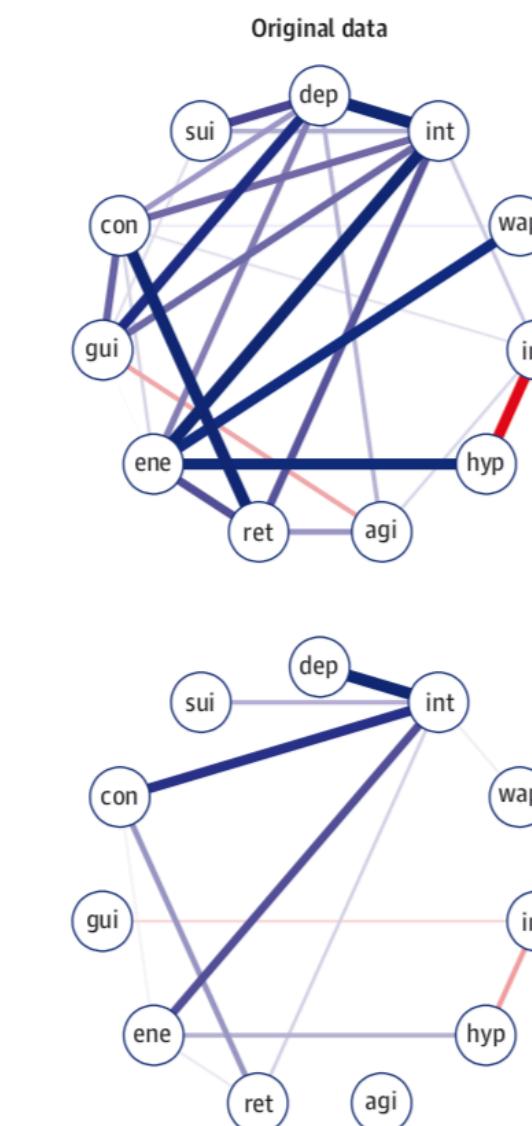
Network comparison

- Assess replicability
 - “does the posttraumatic stress disorder network replicate across data sets?”



Fried et al. 2018

- Assess differences
 - “does the network structure differ for participants with persistent MDD versus patients with remitted MDD?”



Van Borkulo et al. 2015

Network comparison

1. Compare networks visually
 - Make sure layouts are equal (e.g., use ‘averageLayout’ in qgraph)
 - Keep differences in sample size in mind: smaller sample sizes will result in sparser estimated networks

Network comparison

1. Compare networks visually
 - Make sure layouts are equal (e.g., use ‘averageLayout’ in qgraph)
 - Keep differences in sample size in mind: smaller sample sizes will result in sparser estimated networks
2. Correlate the weight matrices

Network comparison

1. Compare networks visually
 - Make sure layouts are equal (e.g., use ‘averageLayout’ in qgraph)
 - Keep differences in sample size in mind: smaller sample sizes will result in sparser estimated networks
2. Correlate the weight matrices
3. Perform test: Network Comparison Test (NCT)
 - Permutation test
 - Van Borkulo et al. (2016). Comparing networks on three aspects: A permutation test.

Network Comparison Test

- A *permutation test*
- Used to investigate:
 1. **Network structure invariance** hypothesis
 - structure is completely identical across subpopulations
 2. **Global strength invariance** hypothesis
 - overall level of connectivity is identical across subpopulations
 3. **Edge strength invariance** hypothesis
 - a specific edge is identical across subpopulations

Network Comparison Test

- A permutation test can be used to test if statistics from two groups differ:



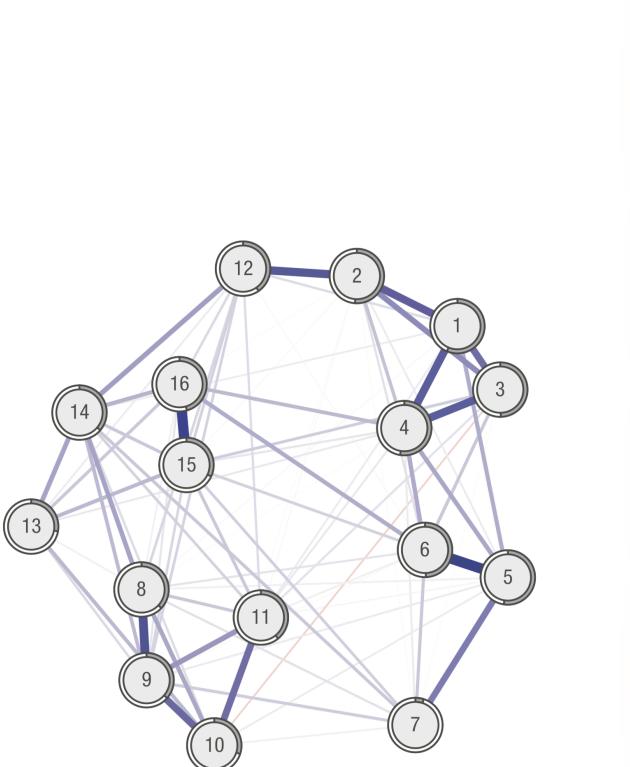
dataset 1



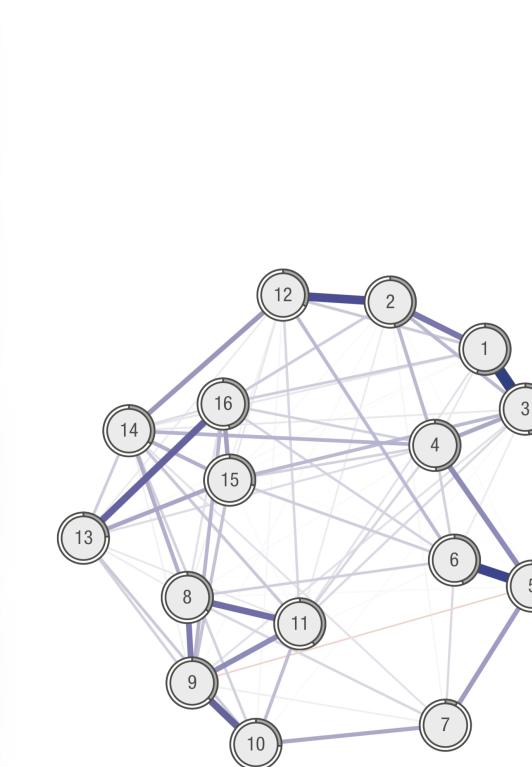
dataset 2

Network Comparison Test

- A permutation test can be used to test if statistics from two groups differ:
 1. compute statistic of interest in both groups (e.g., structure, global strength, edge strength)



dataset 1



dataset 2

Network Comparison Test

- A permutation test can be used to test if statistics from two groups differ:
 1. compute statistic of interest in both groups (e.g., structure, global strength, edge strength)
 2. pool all cases into one large dataset



dataset 1



dataset 2

pooled dataset

Network Comparison Test

- A permutation test can be used to test if statistics from two groups differ:
 1. compute statistic of interest in both groups (e.g., structure, global strength, edge strength)
 2. pool all cases into one large dataset
 3. randomly create new groups by re-distributing the cases



random
dataset 1

pooled dataset

random
dataset 2

Network Comparison Test

- A permutation test can be used to test if statistics from two groups differ:
 1. compute statistic of interest in both groups (e.g., structure, global strength, edge strength)
 2. pool all cases into one large dataset
 3. randomly create new groups by re-distributing the cases



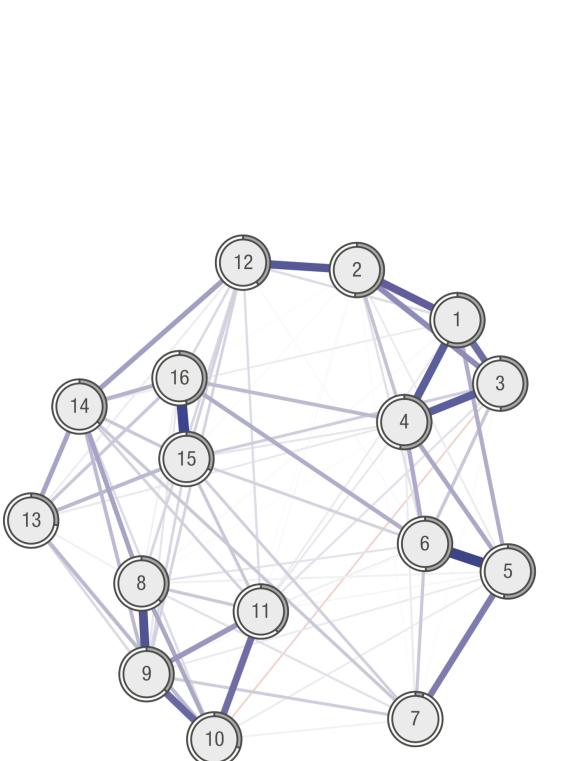
random
dataset 1



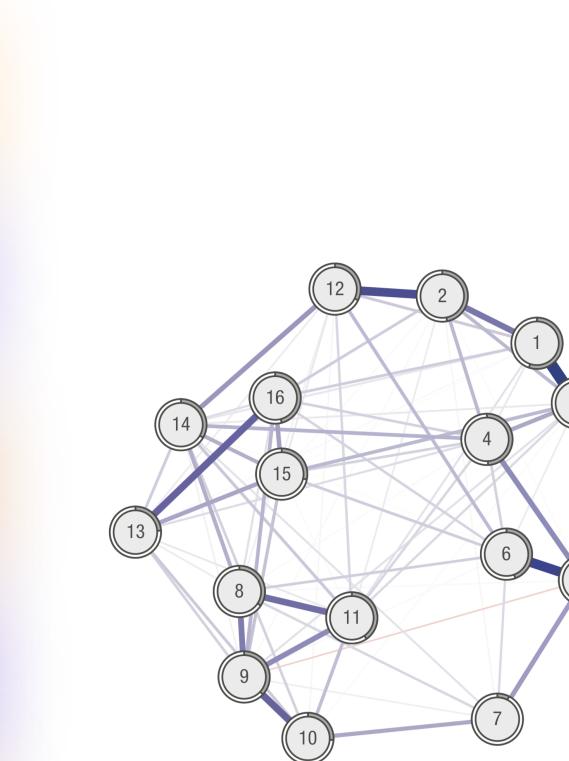
random
dataset 2

Network Comparison Test

- A permutation test can be used to test if statistics from two groups differ:
 1. compute statistic of interest in both groups (e.g., structure, global strength, edge strength)
 2. pool all cases into one large dataset
 3. randomly create new groups by re-distributing the cases
 4. compute the statistic in the new groups and repeat (2-4) to obtain a null-distribution



random
dataset 1

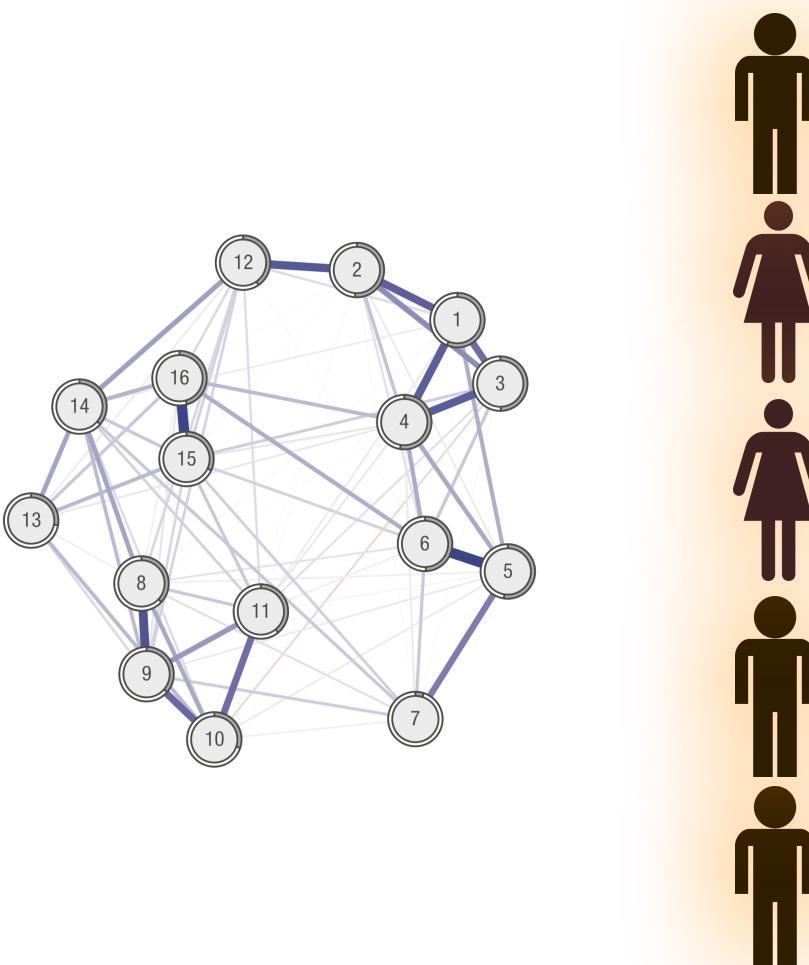


random
dataset 2

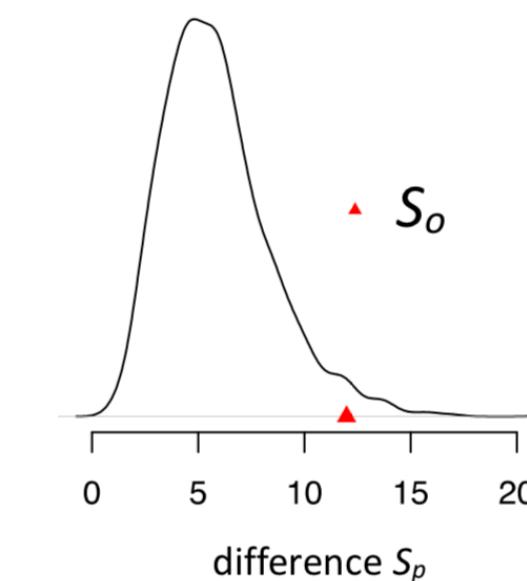


Network Comparison Test

- A permutation test can be used to test if statistics from two groups differ:
 1. compute statistic of interest in both groups (e.g., structure, global strength, edge strength)
 2. pool all cases into one large dataset
 3. randomly create new groups by re-distributing the cases
 4. compute the statistic in the new groups and repeat (2-4) to obtain a null-distribution
 5. test if the observed difference (computed in 1) is in the null-distribution



dataset 1



dataset 2

Network Comparison Test

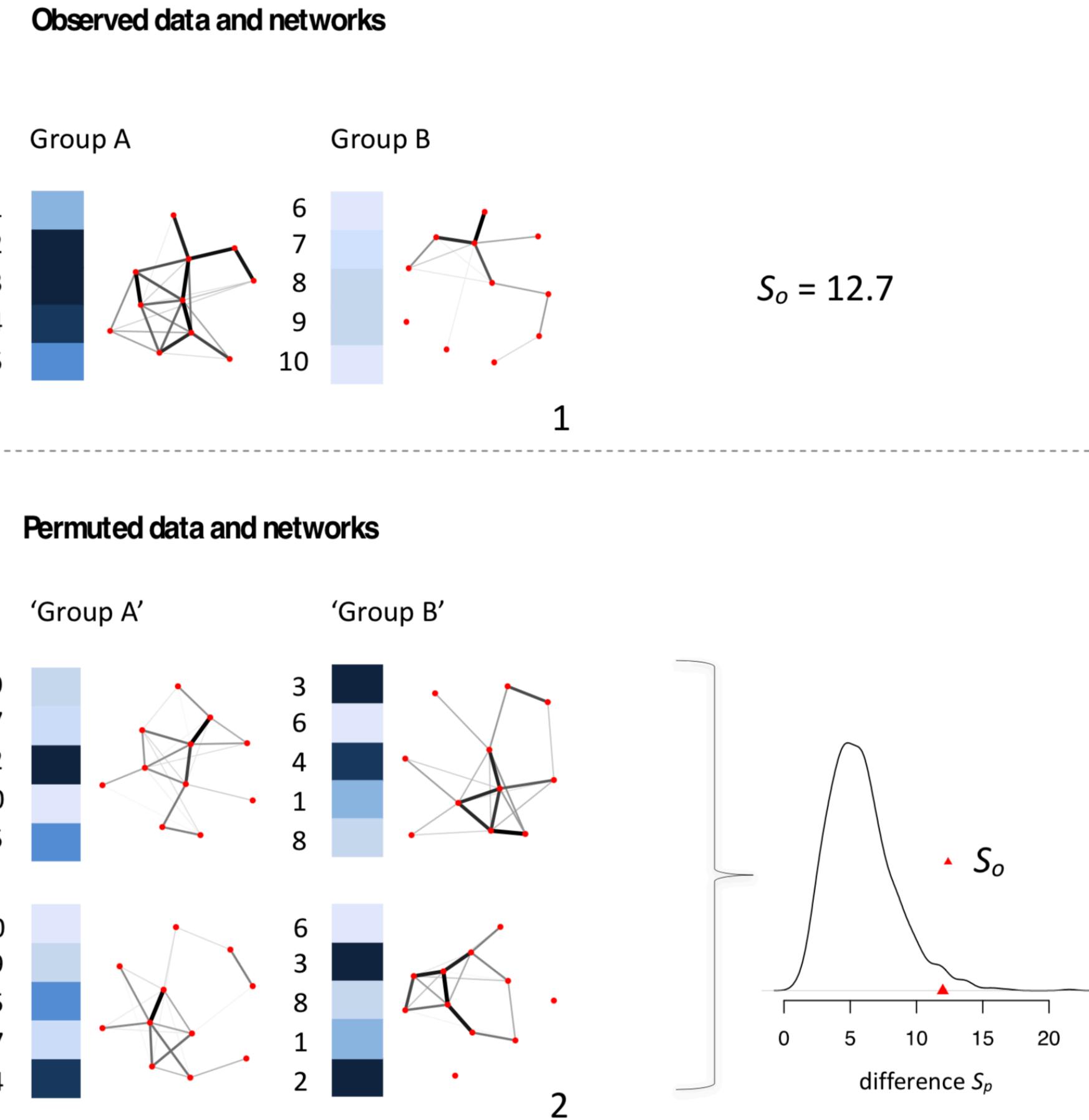


Figure 2. Schematic representation of the three steps involved in NCT.



concluding remarks

considerations

Psychological networks: 3 steps

1. Statistical models estimated from data, from which some parameters can be represented as a weighted network between observed variables
2. These weighted networks can be analysed using measures from graph theory
3. In order to interpret the network and any measures derived from the network, first the accuracy of the network parameters and measures should be assessed

Psychological networks: 3 steps

1. **Statistical models estimated from data**, from which some parameters can be represented as a weighted network between observed variables
 - keep in mind that we estimate many parameters
 - estimation is conservative: low false-positive rate (good specificity), moderate sensitivity
2. These weighted networks can be analysed using measures from graph theory
3. In order to interpret the network and any measures derived from the network, first the accuracy of the network parameters and measures should be assessed

Psychological networks: 3 steps

1. Statistical models estimated from data, from which some parameters can be represented as a weighted network between observed variables
2. These weighted networks can be **analysed using measures from graph theory**
 - keep in mind that step (1) makes psychological networks really different from network structures usually studied using graph theory
3. In order to interpret the network and any measures derived from the network, first the accuracy of the network parameters and measures should be assessed

Psychological networks: 3 steps

1. Statistical models estimated from data, from which some parameters can be represented as a weighted network between observed variables
2. These weighted networks can be analysed using measures from graph theory
3. In order to interpret the network and any measures derived from the network, first the **accuracy of the network parameters and measures** should be assessed
 - investigating the accuracy and stability of the estimated network parameters helps us to understand the stability of the network models
 - this is crucial to **understanding replicability**: can you expect your estimated networks to replicate?