Estimating and Interpreting Psychological Networks in R

Jens Lange and Janis Zickfeld

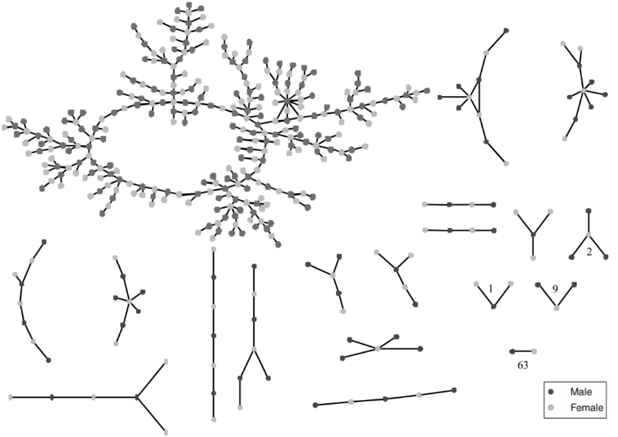
Sep 11, 2022

Introduction to Complex Networks

Complexity Science

- *complex systems* entail parts that interact with each other to produce system behavior

Complex Networks - Examples

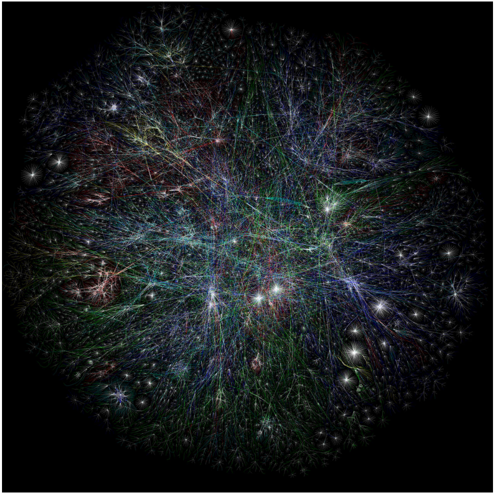


Bearman et al. (2004, AJS)

Complex Networks - Examples

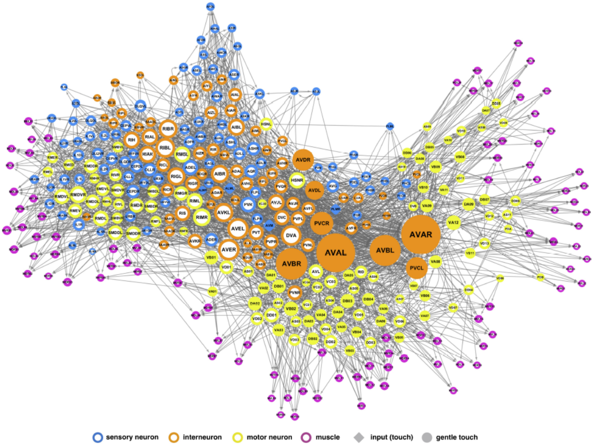
Xiaoqian et al. (2017, CJA)

Complex Networks - Examples



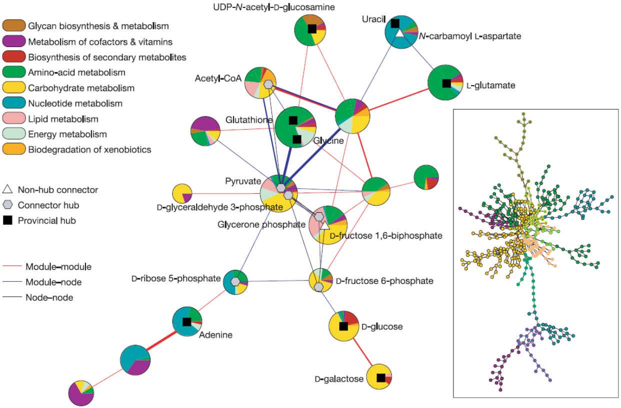
Albert et al. (1999, Nature)

Complex Networks - Examples



Yan et al. (2017, Nature)

Complex Networks - Examples

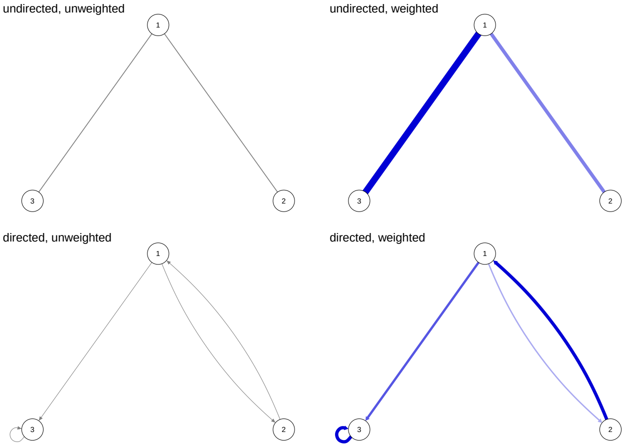


Guimerà & Amaral (2005, Nature)

Complex Networks - Definitions

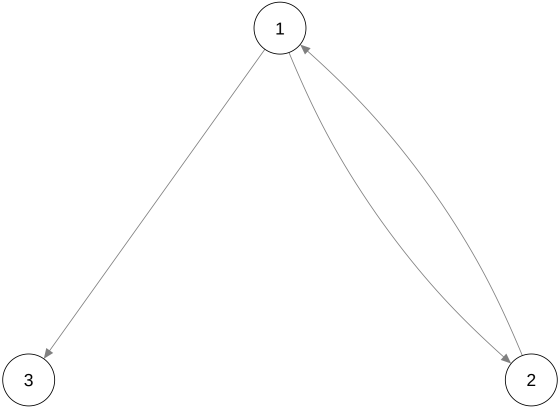
- network: *nodes* that are connected via *edges*
- nodes can be...
  - people
  - neurons
  - psychological variables
  - ...
- edges can be...
  - friendships
  - synapses
  - causal connections
  - ...

Complex Networks - Definitions



Complex Networks - Mathematical Notation

- a Graph G entails sets of nodes N and edges E
  - $G = \{N, E\}$
  - $N = \{1, 2, 3\}$
  - $E = \{(1, 2), (1, 3), (2, 1)\}$

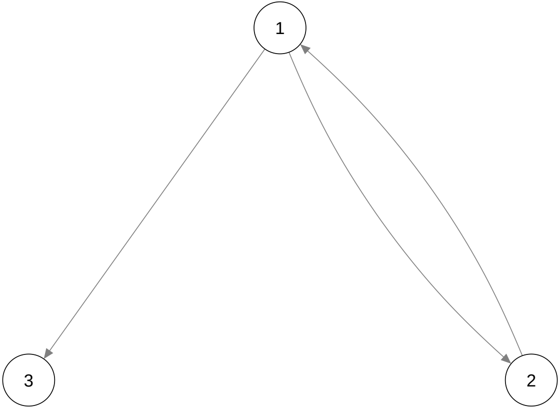


Complex Networks - Adjacency/Weights Matrix

- *adjacency matrix*: $N \times N$ matrix, with elements 0 or 1
  - 1 in row i and column j: edge from node i to j
  - undirected network has symmetric adjacency matrix
- *weights matrix* has the same form but with weights

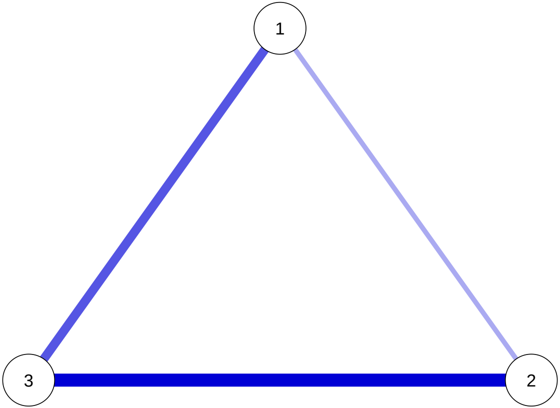
Complex Networks - Adjacency/Weights Matrix

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$



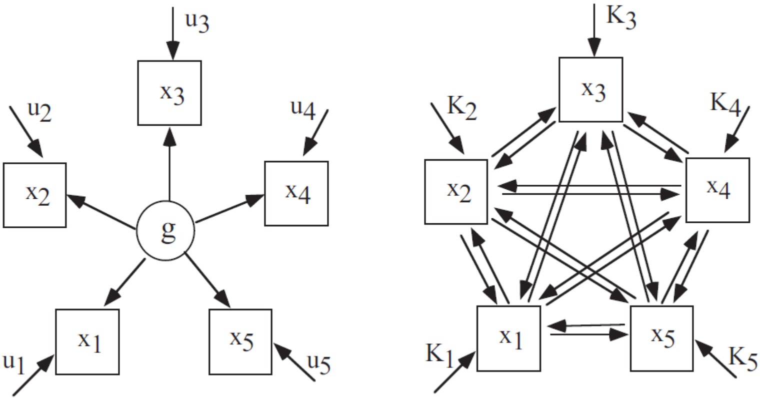
Complex Networks - Adjacency/Weights Matrix

$$\mathbf{W} = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \end{bmatrix}$$



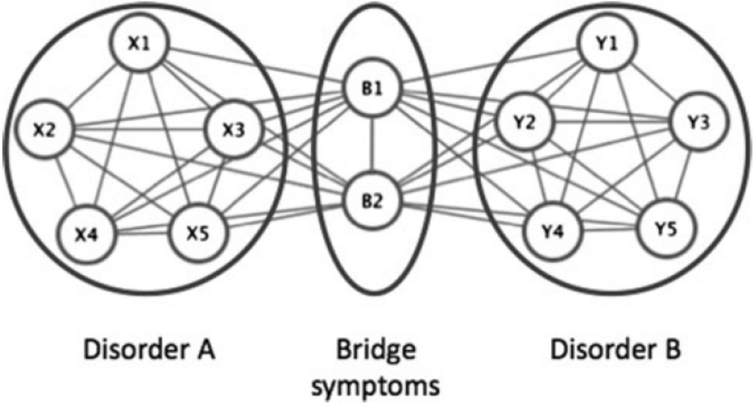
Psychological Networks

Psychological Networks - Intelligence



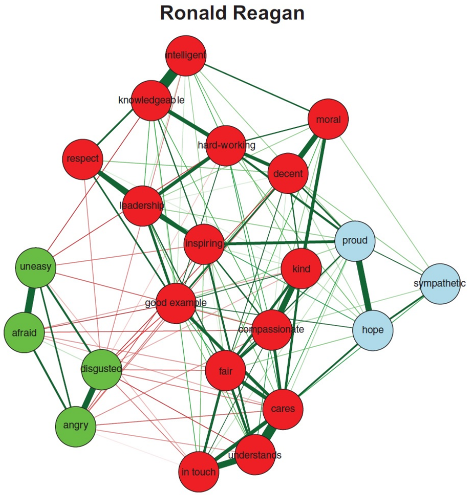
Van der Maas et al. (2006, Psych Review)

Psychological Networks - Psychological Disorders



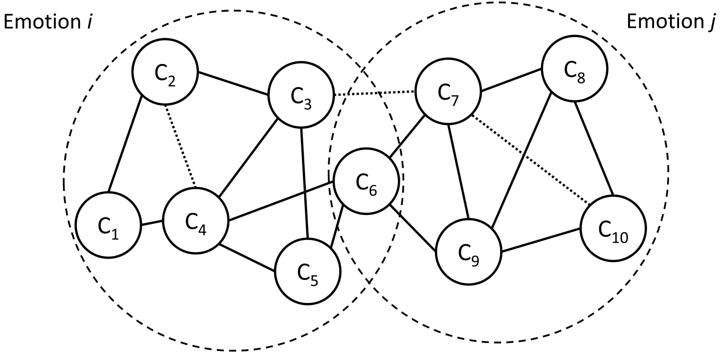
Cramer et al. (2010, BBS)

Psychological Networks - Attitudes



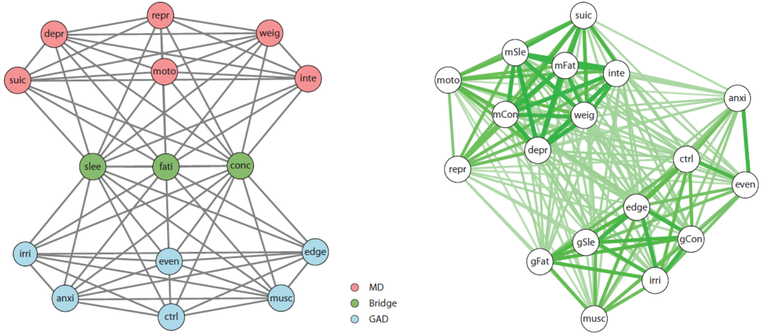
Dalege et al. (2016, PsychRev)

Psychological Networks - Emotions



Lange et al. (2020, PPS); Lange & Zickfeld (2021, ER)

Psychological Networks - Network Analysis



Borsboom et al. (2021, NRMP)

Summary for Introduction

Summary

- networks are versatile approach to complex systems
- networks entail sets of nodes and edges
- in psychological networks...
  - nodes: indicators of a psychological construct
  - edges: causal connections between indicators
- network analysis as methodological tool to estimate causal networks

Estimating Cross-sectional Psychological Networks

Data Structure

- Person X Node-Rating Matrix

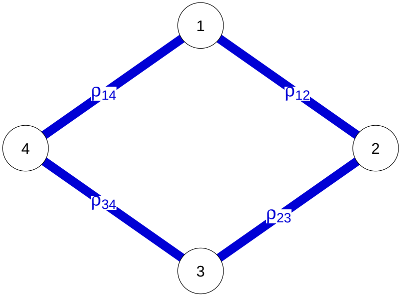
```
#>   Subject Node1 Node2 Node3 Node4
#> 1       1     3     2     5     1
#> 2       2     2     1     6     2
#> 3       3     1     1     4     2
#> 4     ...   ...   ...   ...   ...
#> 5       n     3     3     5     4
```

pairwise Markov Random Field Model (pMRF)

- unique relationships of two nodes

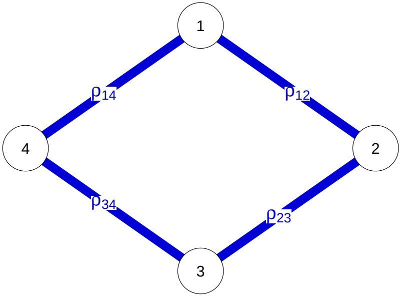
pairwise Markov Random Field Model (pMRF)

- unique relationships of two nodes
- estimation of *conditional (in)dependencies*



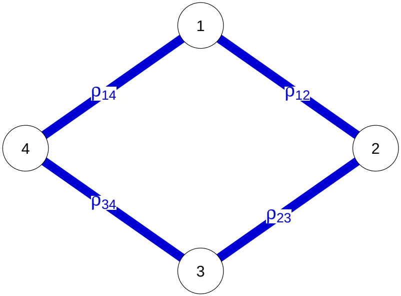
pairwise Markov Random Field Model (pMRF)

- unique relationships of two nodes
- estimation of *conditional (in)dependencies*
  - edge: e.g. $Node1 \not\perp\!\!\!\perp Node2 \mid Node3, Node4$



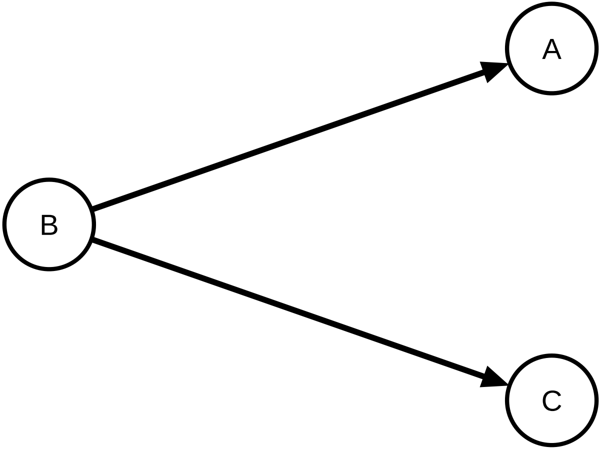
pairwise Markov Random Field Model (pMRF)

- unique relationships of two nodes
- estimation of *conditional (in)dependencies*
  - edge: e.g. $Node1 \not\perp\!\!\!\perp Node2 \mid Node3, Node4$
  - no edge: e.g. $Node1 \perp\!\!\!\perp Node3 \mid Node2, Node4$



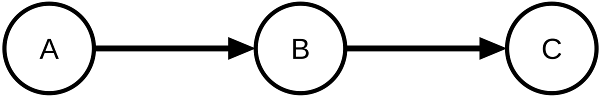
pairwise Markov Random Field Model (pMRF)

- conditional (in)dependencies imply causal structure
- three cases $\rightarrow$ *common cause*
  - $r(A, C) > 0$ and $A \perp\!\!\!\perp C \mid B$



pairwise Markov Random Field Model (pMRF)

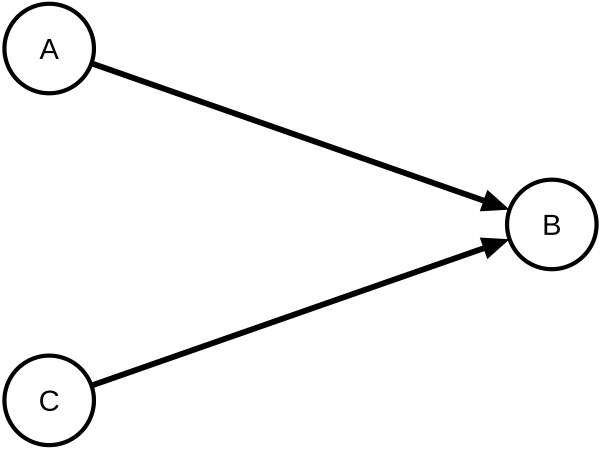
- conditional (in)dependencies imply causal structure
- three cases $\rightarrow$ *chain*
  - $r(A, C) > 0$ and $A \perp\!\!\!\perp C \mid B$



pairwise Markov Random Field Model (pMRF)

- conditional (in)dependencies imply causal structure

- three cases $\rightarrow$ *collider*
  - $r(A, C) = 0$ and $A \not\perp\!\!\!\perp C \mid B$



Gaussian Graphical Model

- pMRF for continuous data

Gaussian Graphical Model

- pMRF for continuous data
- Option 1: conditional (in)dependence relationships estimated via *partial correlations*

Gaussian Graphical Model

- pMRF for continuous data
- Option 1: conditional (in)dependence relationships estimated via *partial correlations*
  1. take variance-covariance matrix (unstandardized correlation matrix) of all nodes

Gaussian Graphical Model

- pMRF for continuous data
- Option 1: conditional (in)dependence relationships estimated via *partial correlations*
    1. take variance-covariance matrix (unstandardized correlation matrix) of all nodes
    2. inverse of the matrix is called *Gaussian Graphical Model*

Gaussian Graphical Model

- pMRF for continuous data
- Option 1: conditional (in)dependence relationships estimated via *partial correlations*
  1. take variance-covariance matrix (unstandardized correlation matrix) of all nodes
  2. inverse of the matrix is called *Gaussian Graphical Model*
  3. standardized inverted matrix: partial correlation matrix

Gaussian Graphical Model

- pMRF for continuous data
- Option 1: conditional (in)dependence relationships estimated via *partial correlations*
  1. take variance-covariance matrix (unstandardized correlation matrix) of all nodes
  2. inverse of the matrix is called *Gaussian Graphical Model*
  3. standardized inverted matrix: partial correlation matrix
- Option 2: conditional (in)dependence relationships estimated via *node-wise regression*

Gaussian Graphical Model

- pMRF for continuous data
- Option 1: conditional (in)dependence relationships estimated via *partial correlations*
  1. take variance-covariance matrix (unstandardized correlation matrix) of all nodes
  2. inverse of the matrix is called *Gaussian Graphical Model*
  3. standardized inverted matrix: partial correlation matrix
- Option 2: conditional (in)dependence relationships estimated via *node-wise regression*
  1. regress each node on all other nodes

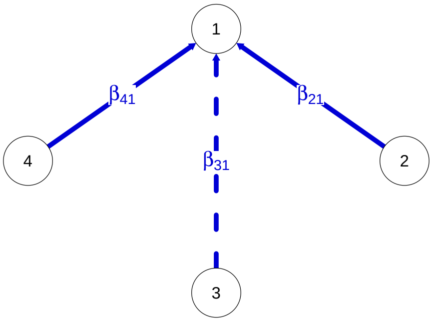
Gaussian Graphical Model

- pMRF for continuous data
- Option 1: conditional (in)dependence relationships estimated via *partial correlations*
    1. take variance-covariance matrix (unstandardized correlation matrix) of all nodes
    2. inverse of the matrix is called *Gaussian Graphical Model*
    3. standardized inverted matrix: partial correlation matrix
- Option 2: conditional (in)dependence relationships estimated via *node-wise regression*
    1. regress each node on all other nodes
    2. save regression weights

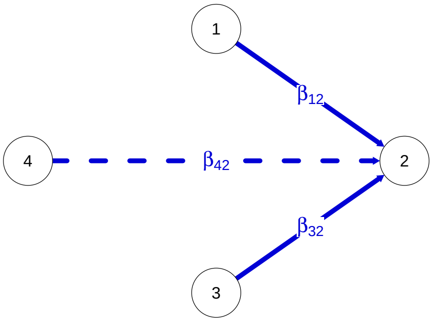
Gaussian Graphical Model

- pMRF for continuous data
- Option 1: conditional (in)dependence relationships estimated via *partial correlations*
    1. take variance-covariance matrix (unstandardized correlation matrix) of all nodes
    2. inverse of the matrix is called *Gaussian Graphical Model*
    3. standardized inverted matrix: partial correlation matrix
- Option 2: conditional (in)dependence relationships estimated via *node-wise regression*
    1. regress each node on all other nodes
    2. save regression weights
    3. apply AND or OR rule to determine edge weights

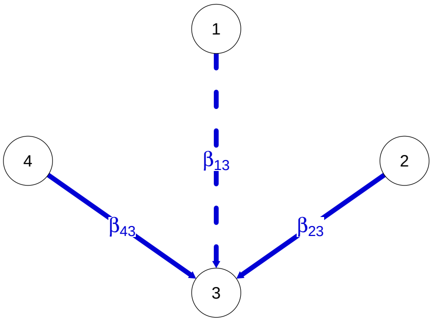
Gaussian Graphical Model – Node-wise Regression



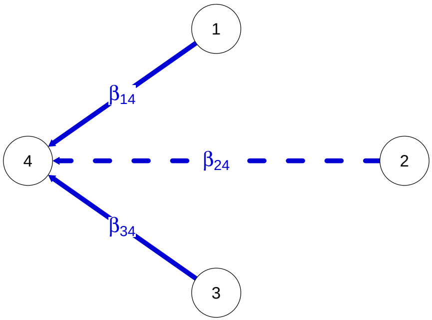
Gaussian Graphical Model – Node-wise Regression



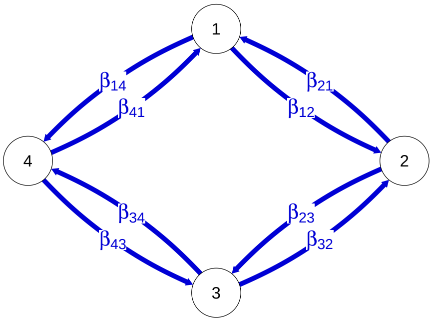
Gaussian Graphical Model – Node-wise Regression



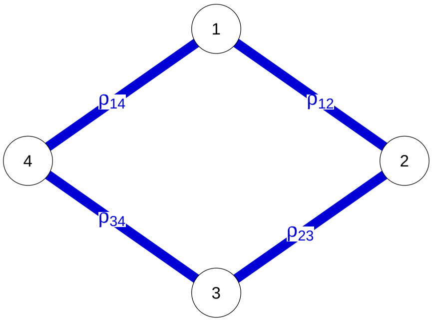
Gaussian Graphical Model – Node-wise Regression



Gaussian Graphical Model – Node-wise Regression



Gaussian Graphical Model – Node-wise Regression



Model Selection

- estimation of many parameters

Model Selection

- estimation of many parameters
  - k nodes: $k * (k-1)/2$ edges

Model Selection

- estimation of many parameters
  - k nodes: $k * (k - 1)/2$ edges
  - preferred model should be sparse and true

Model Selection

- estimation of many parameters
  - k nodes: $k * (k-1)/2$ edges
  - preferred model should be sparse and true
- model selection

Model Selection - Thresholding

- estimate all edges as partial correlations

Model Selection - Thresholding

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)

Model Selection - Thresholding

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- hide all edges not fulfilling threshold

Model Selection - Thresholding

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- hide all edges not fulfilling threshold

- advantages

Model Selection - Thresholding

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- hide all edges not fulfilling threshold

- advantages
  - very fast

Model Selection - Thresholding

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- hide all edges not fulfilling threshold

- advantages
  - very fast
  - fixed false positive rate

Model Selection - Thresholding

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- hide all edges not fulfilling threshold

- advantages
  - very fast
  - fixed false positive rate
  - non-biased estimates

Model Selection - Thresholding

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- hide all edges not fulfilling threshold

- advantages
  - very fast
  - fixed false positive rate
  - non-biased estimates
- disadvantages

Model Selection - Thresholding

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- hide all edges not fulfilling threshold

- advantages
  - very fast
  - fixed false positive rate
  - non-biased estimates
- disadvantages
  - not really model selection

Model Selection - Thresholding

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- hide all edges not fulfilling threshold

- advantages
  - very fast
  - fixed false positive rate
  - non-biased estimates
- disadvantages
  - not really model selection
  - can lack sensitivity

Model Selection - Thresholding

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- hide all edges not fulfilling threshold

- advantages
  - very fast
  - fixed false positive rate
  - non-biased estimates
- disadvantages
  - not really model selection
  - can lack sensitivity
  - potentially prominent false positive edges

Model Selection - Pruning

- estimate all edges as partial correlations

Model Selection - Pruning

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)

Model Selection - Pruning

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- reestimate model with edges lacking characteristic set to zero

Model Selection - Pruning

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- reestimate model with edges lacking characteristic set to zero

- advantages

Model Selection - Pruning

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- reestimate model with edges lacking characteristic set to zero

- advantages
  - as before

Model Selection - Pruning

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- reestimate model with edges lacking characteristic set to zero

- advantages
  - as before
- disadvantages

Model Selection - Pruning

- estimate all edges as partial correlations
- consider characteristic of edges (e.g., significance)
- reestimate model with edges lacking characteristic set to zero

- advantages
  - as before
- disadvantages
  - as before, except that it is model selection

Model Selection - Model Search

- search through entire model space with criterion

Model Selection - Model Search

- search through entire model space with criterion
- e.g., stepup with empty start

Model Selection - Model Search

- search through entire model space with criterion
- e.g., stepup with empty start
  - start wih empty network

Model Selection - Model Search

- search through entire model space with criterion
- e.g., stepup with empty start
  - start wih empty network
  - add edge with highest modification indices

Model Selection - Model Search

- search through entire model space with criterion
- e.g., stepup with empty start
  - start wih empty network
  - add edge with highest modification indices
  - continue until information criterion is optimized

Model Selection - Model Search

- search through entire model space with criterion
- e.g., stepup with empty start
  - start wih empty network
  - add edge with highest modification indices
  - continue until information criterion is optimized
- complex algorithms: ggmModSelect, modelsearch

Model Selection - Model Search

- search through entire model space with criterion
- e.g., stepup with empty start
  - start wih empty network
  - add edge with highest modification indices
  - continue until information criterion is optimized
- complex algorithms: ggmModSelect, modelsearch
- advantages

Model Selection - Model Search

- search through entire model space with criterion
- e.g., stepup with empty start
  - start wih empty network
  - add edge with highest modification indices
  - continue until information criterion is optimized
- complex algorithms: ggmModSelect, modelsearch
- advantages
  - good sensitivity and specificity

Model Selection - Model Search

- search through entire model space with criterion
- e.g., stepup with empty start
  - start wih empty network
  - add edge with highest modification indices
  - continue until information criterion is optimized
- complex algorithms: ggmModSelect, modelsearch
- advantages
  - good sensitivity and specificity
  - converges to true model by overcoming false positive rate

Model Selection - Model Search

- search through entire model space with criterion
- e.g., stepup with empty start
  - start wih empty network
  - add edge with highest modification indices
  - continue until information criterion is optimized
- complex algorithms: ggmModSelect, modelsearch
- advantages
  - good sensitivity and specificity
  - converges to true model by overcoming false positive rate
  - local optimum

Model Selection - Model Search

- search through entire model space with criterion
- e.g., stepup with empty start
  - start wih empty network
  - add edge with highest modification indices
  - continue until information criterion is optimized
- complex algorithms: ggmModSelect, modelsearch

- advantages
  - good sensitivity and specificity
  - converges to true model by overcoming false positive rate
  - local optimum
- disadvantages

Model Selection - Model Search

- search through entire model space with criterion
- e.g., stepup with empty start
  - start wih empty network
  - add edge with highest modification indices
  - continue until information criterion is optimized
- complex algorithms: ggmModSelect, modelsearch
- advantages
  - good sensitivity and specificity
  - converges to true model by overcoming false positive rate
  - local optimum
- disadvantages
  - can be extremely slow

Model Selection - Model Search

- search through entire model space with criterion
- e.g., stepup with empty start
  - start wih empty network
  - add edge with highest modification indices
  - continue until information criterion is optimized
- complex algorithms: ggmModSelect, modelsearch
- advantages
  - good sensitivity and specificity
  - converges to true model by overcoming false positive rate
  - local optimum
- disadvantages
  - can be extremely slow
  - unknown number of false positive edges

Model Selection - Regularization

- method: Least Absolute Shrinkage and Selection Operator (LASSO)

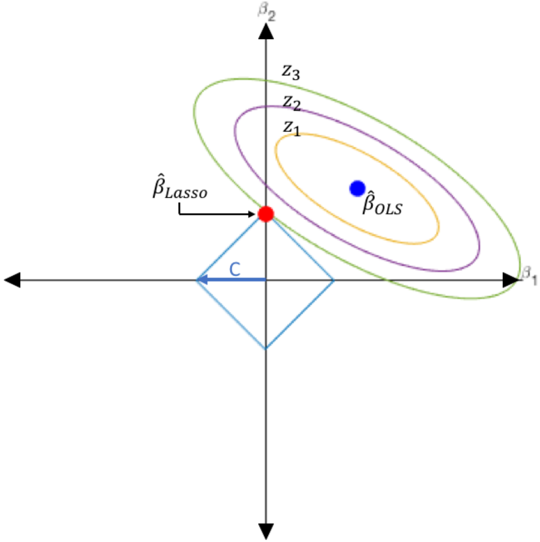
Model Selection - Regularization

- method: Least Absolute Shrinkage and Selection Operator (LASSO)
  - limits the size of all edges

Model Selection - Regularization

- method: Least Absolute Shrinkage and Selection Operator (LASSO)
  - limits the size of all edges
  - sets small edges to exactly zero

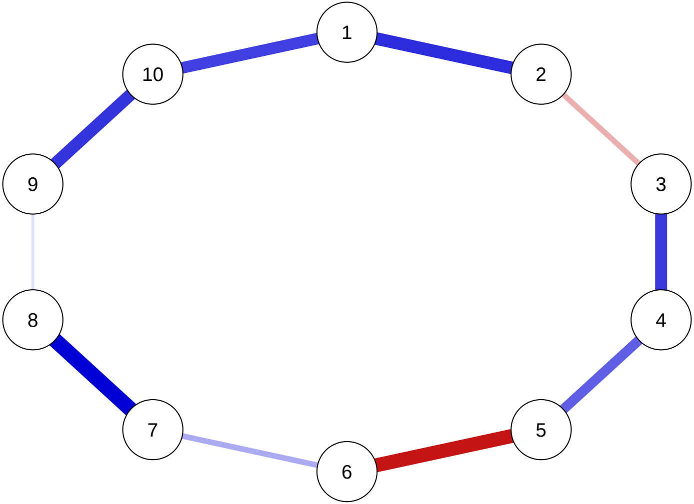
Model Selection - Regularization



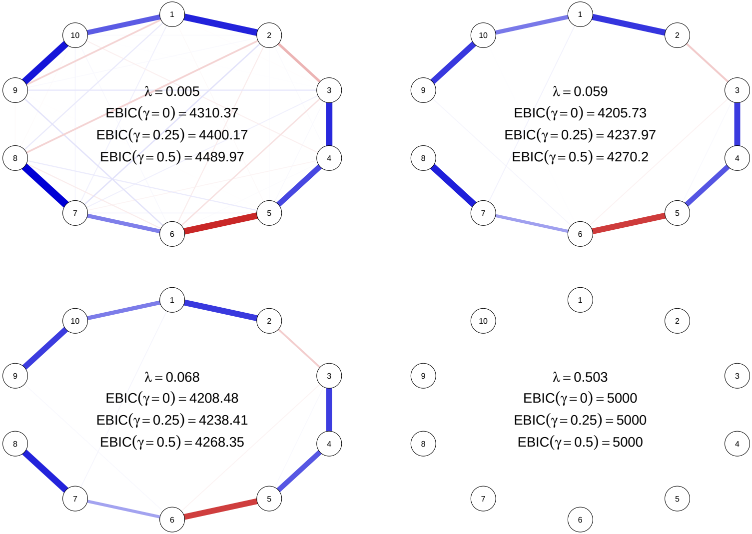
Model Selection - Regularization

- method: Least Absolute Shrinkage and Selection Operator (LASSO)
  - limits the size of all edges
  - sets small edges to exactly zero
  - varying tuning parameter λ changes sparsity (100 networks)
  - minimize Extended Bayesian Information Criterion (EBIC)
  - hyperparamter γ controls EBIC
    - 0: err on the side of discovery
    - 0.5: err on the side of caution
    - typically set 0.25 or 0.5
  - LASSO for partial correlations (gLASSO) and node-wise regression (eLASSO)

Model Selection - Regularization (Example)



Model Selection - Regularization (Example)



Model Selection - Regularization

- advantages

Model Selection - Regularization

- advantages
  - very fast

Model Selection - Regularization

- advantages
  - very fast
  - high sensitivity

Model Selection - Regularization

- advantages
  - very fast
  - high sensitivity
  - receives network structure also with low sample size

Model Selection - Regularization

- advantages
  - very fast
  - high sensitivity
  - receives network structure also with low sample size
  - clearer pictures by setting edges to zero

Model Selection - Regularization

- advantages
  - very fast
  - high sensitivity
  - receives network structure also with low sample size
  - clearer pictures by setting edges to zero
- disadvantages

Model Selection - Regularization

- advantages
  - very fast
  - high sensitivity
  - receives network structure also with low sample size
  - clearer pictures by setting edges to zero
- disadvantages
  - poor performance in large sample sizes

Model Selection - Regularization

- advantages
  - very fast
  - high sensitivity
  - receives network structure also with low sample size
  - clearer pictures by setting edges to zero
- disadvantages
  - poor performance in large sample sizes
  - relies on assumption of sparsity

Model Selection - Regularization

- advantages
  - very fast
  - high sensitivity
  - receives network structure also with low sample size
  - clearer pictures by setting edges to zero
- disadvantages
  - poor performance in large sample sizes
  - relies on assumption of sparsity
  - leads to biased estimates

Model Selection - What to Choose?

- no idea
- check Isvoranu & Epskamp (in press, PM)
- consider, what you want to do
- regularization (EBICglasso) is most common

Ising Model

- pMRF for binary data

Ising Model

- pMRF for binary data
- estimation via node-wise regression

Ising Model

- pMRF for binary data
- estimation via node-wise regression
  - logistic regression

Ising Model

- pMRF for binary data
- estimation via node-wise regression
  - logistic regression
  - regularization via eLASSO

Network Analysis in R

OSF Files

https://osf.io/8akru/

Get Started

- set working directory

```
setwd("...")
```

- load packages

```
#R version 4.2.1
###packages
library(qgraph)             #plotting networks; version 1.9.2
library(bootnet)            #estimating networks; version 1.5
library(psych)              #psych statistics; version 2.2.5
library(CliquePercolation) #network structure and data set; version 0.3.0
library(psychonetrics)      #(confirmatory) network modeling; version 0.10
library(dplyr)              #helpful function; version 1.0.10
```

Load Data

```
#load data - ratings of Obama
data(Obama)
data <- Obama

###Items
##Beliefs - scale from 1 to 5
#Mor - "Is moral"
#Led - "Would provide strong leadership"
#Car - "Really cares about people like you"
#Kno - "Is knowledgeable"
#Int - "Is intelligent"
#Hns - "Is honest"
##Feelings - scale from 1 to 5
#Ang - "Angry"
#Hop - "Hopeful"
#Afr - "Afraid of him"
#Prd - "Proud"
```

Descriptive Statistics

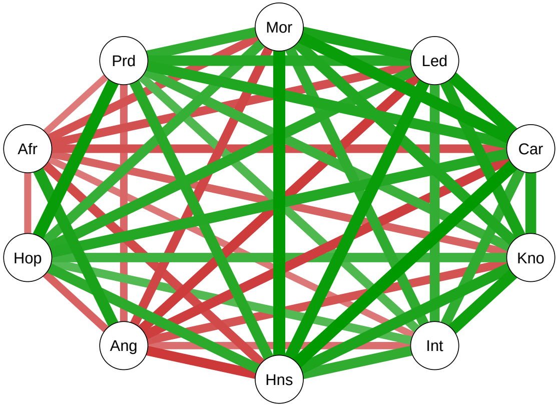
```
describe(data)
#-> several missing values
#-> no obvious skewness or kurtosis
```

Zero-order correlations

```
#compute correlations
correlations <- cor(data, use = "pairwise.complete.obs")
correlations
```

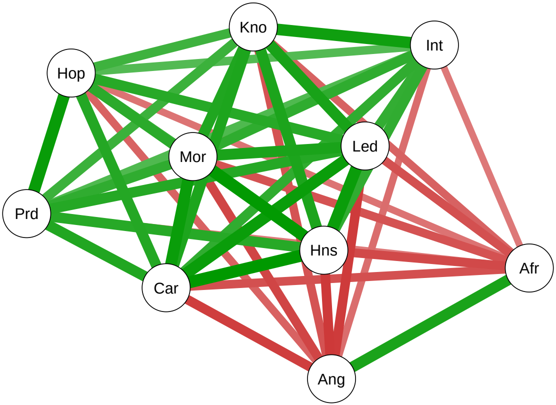
Plotting Correlations with qgraph

```
#plotting correlations with qgraph
cor_graph <- qgraph(correlations)
```



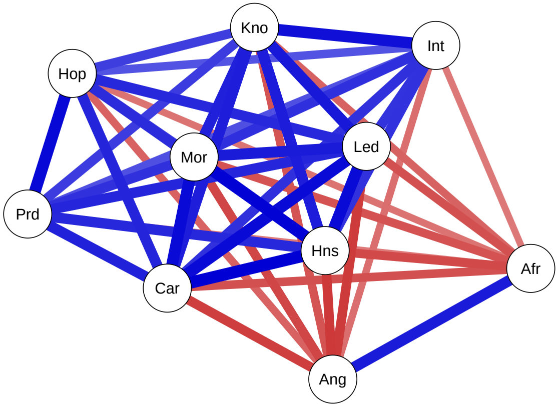
Plotting Correlations with qgraph

```
#+ using Fruchterman-Reingold
cor_graph <- qgraph(correlations, layout = "spring")
```



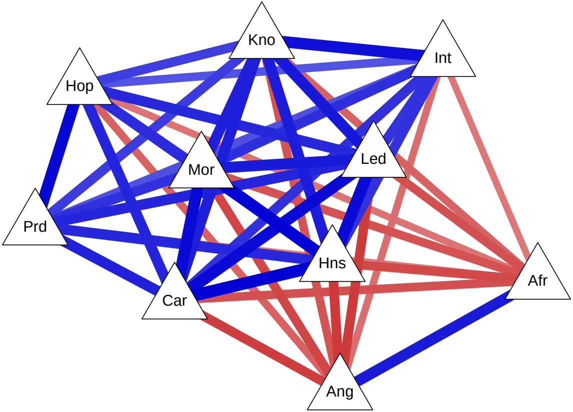
Plotting Correlations with qgraph

```
#+ change for color-blind people
cor_graph <- qgraph(correlations, layout = "spring",
                    theme = "colorblind")
```



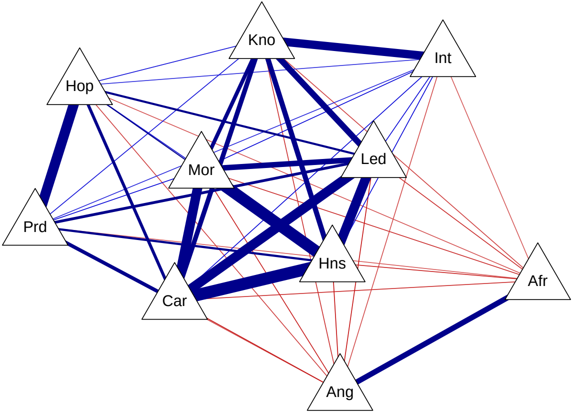
Plotting Correlations with qgraph

```
#+ change shape of nodes
cor_graph <- qgraph(correlations, layout = "spring",
                    theme = "colorblind", shape = "triangle")
```



Plotting Correlations with qgraph

```r
#+ make correlations with absolute value < .70 less important
cor_graph <- qgraph(correlations, layout = "spring", theme = "colorblind",
                    shape = "triangle", cut = .7)
```



Plotting Correlations with qgraph

```
#for list of features see options in qgraph help page
?qgraph
```

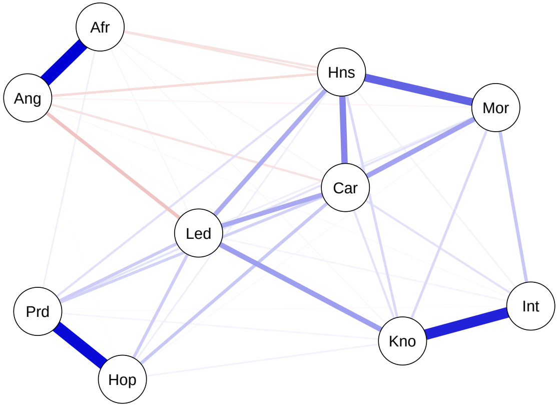
Estimating the Gaussian Graphical Model

```r
#estimate regularized partial correlation network
#EBICglasso (gLASSO with EBIC model selection)
#correlations determined via cor
#pairwise deletion of missing values
GGM_net <- estimateNetwork(data, default = "EBICglasso", corMethod = "cor",
                           missing = "pairwise")
```

```
#> Warning in EBICglassoCore(S = S, n = n, gamma = gamma, penalize.diagonal =
#> penalize.diagonal, : A dense regularized network was selected (lambda < 0.1 *
#> lambda.max). Recent work indicates a possible drop in specificity. Interpret the
#> presence of the smallest edges with care. Setting threshold = TRUE will enforce
#> higher specificity, at the cost of sensitivity.
```

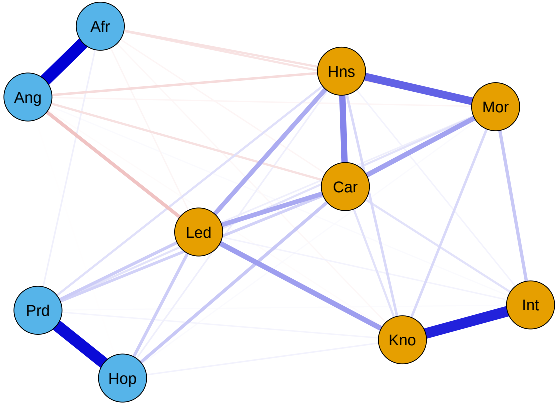
Plotting the Gaussian Graphical Model

```
#plot network with qgraph
GGM_graph <- plot(GGM_net)
```



Plotting the Gaussian Graphical Model

```
###declare beliefs and feelings separate groups
groups <- c(rep("Beliefs",6), rep("Feelings",4))

#plot network with groups; no legend
GGM_graph <- plot(GGM_net, layout = "spring", theme = "colorblind",
                  groups = groups, legend = FALSE)
```



Information about Gaussian Graphical Model

```
#save layout of the graph
layout <- GGM_graph$layout

#get weights matrix
Wmat_GGM <- getWmat(GGM_graph)
Wmat_GGM
```

Node Centrality

- (potentially) information about how important a node is

Node Centrality

- (potentially) information about how important a node is
  - *strength* (*degree*): sum of all edge weights

Node Centrality

- (potentially) information about how important a node is
  - *strength* (*degree*): sum of all edge weights
  - *closeness*: inverse of average shortest path length

Node Centrality

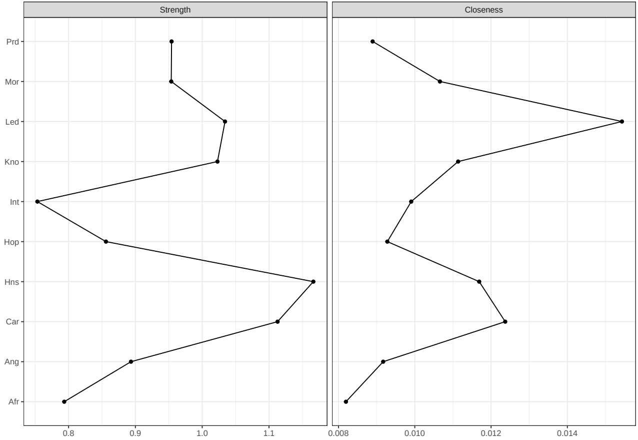
- (potentially) information about how important a node is
  - *strength* (*degree*): sum of all edge weights
  - *closeness*: inverse of average shortest path length
  - . . .

Node Centrality

```
#determine values
cent <- centrality(GGM_net)

#get values
cent$OutDegree
cent$Closeness
```

Node Centrality

```
#plot values
centralityPlot(GGM_net, scale = "raw",
               include = c("Strength","Closeness"))
```



Network Stability and Difference Tests

- edges in regularized networks are significant

Network Stability and Difference Tests

- edges in regularized networks are significant
- nevertheless, construct CIs around edge weights

Network Stability and Difference Tests

- edges in regularized networks are significant
- nevertheless, construct CIs around edge weights
- test whether two edges differ

Network Stability and Difference Tests

- edges in regularized networks are significant
- nevertheless, construct CIs around edge weights
- test whether two edges differ
- test stability of centrality indices

Network Stability and Difference Tests

- edges in regularized networks are significant
- nevertheless, construct CIs around edge weights
- test whether two edges differ
- test stability of centrality indices
- compare centrality indices

Network Stability and Difference Tests

- edges in regularized networks are significant
- nevertheless, construct CIs around edge weights
- test whether two edges differ
- test stability of centrality indices
- compare centrality indices

- bootstrap methods

Network Stability and Difference Tests

- edges in regularized networks are significant
- nevertheless, construct CIs around edge weights
- test whether two edges differ
- test stability of centrality indices
- compare centrality indices

- bootstrap methods
  - non-parametric bootstrap: re-sampling from data with replacement

Network Stability and Difference Tests

- edges in regularized networks are significant
- nevertheless, construct CIs around edge weights
- test whether two edges differ
- test stability of centrality indices
- compare centrality indices

- bootstrap methods
  - non-parametric bootstrap: re-sampling from data with replacement
    - for edges and edges/centrality difference tests

Network Stability and Difference Tests

- edges in regularized networks are significant
- nevertheless, construct CIs around edge weights
- test whether two edges differ
- test stability of centrality indices
- compare centrality indices

- bootstrap methods
  - non-parametric bootstrap: re-sampling from data with replacement
    - for edges and edges/centrality difference tests
  - case-dropping bootstrap: successively drop cases; check parameter consistency

Network Stability and Difference Tests

- edges in regularized networks are significant
- nevertheless, construct CIs around edge weights
- test whether two edges differ
- test stability of centrality indices
- compare centrality indices

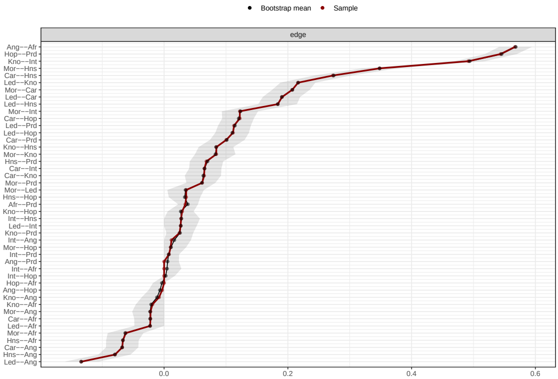
- bootstrap methods
  - non-parametric bootstrap: re-sampling from data with replacement
    - for edges and edges/centrality difference tests
  - case-dropping bootstrap: successively drop cases; check parameter consistency
    - for centrality indices

Network Stability and Difference Tests

```r
#non-parametric bootstrap
#-> for edge stability and edge as well as centrality difference tests
#run bootstrap
set.seed(4186)
boot1 <- bootnet(GGM_net, statistics = c("edge","Strength","Closeness"),
                 nboots = 1000, nCores = 2, type = "nonparametric")
save(boot1, file = "boot_edges.RData") #save results

#case-dropping bootstrap
#-> for centrality stability
#run bootstrap
set.seed(4186)
boot2 <- bootnet(GGM_net, statistics = c("Strength","Closeness"),
                 nboots = 1000, nCores = 2, type = "case")
save(boot2, file = "boot_centrality.RData") #save results
```

Network Stability and Difference Tests

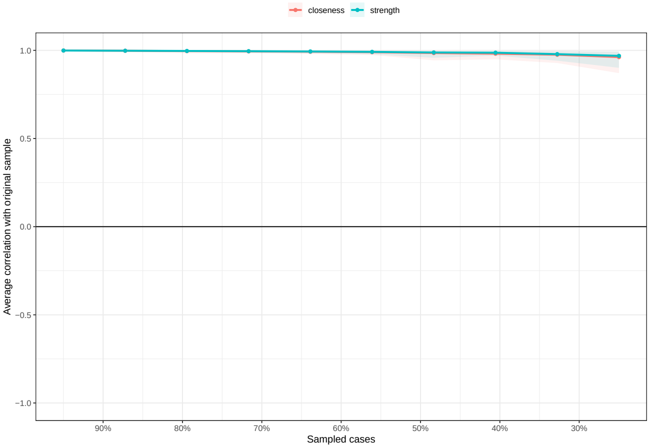
```r
load("boot_edges.RData")
load("boot_centrality.RData")

#plot edge CIs
plot(boot1, statistics = "edge", labels = TRUE, order = "sample")
```




```r
#plot edge CIs + make labels visible by creating large pdf
# pdf("edge_stability.pdf", height = 10)
# plot(boot1, statistics = "edge", labels = TRUE, order = "sample")
# dev.off()
```

Network Stability and Difference Tests

```r
#plot centrality stability
plot(boot2, statistics = c("Strength","Closeness"))
```



Network Stability and Difference Tests

```r
#check stability of centrality indices
corStability(boot2)
```

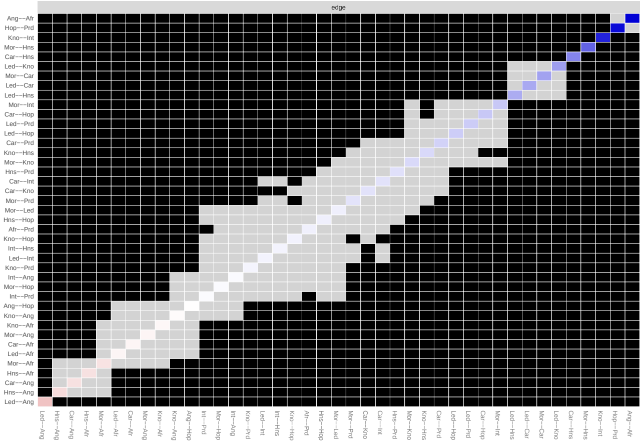
```
#> === Correlation Stability Analysis ===
#>
#> Sampling levels tested:
#>    nPerson Drop%   n
#> 1     1478  75.0 104
#> 2     1938  67.2  95
#> 3     2398  59.5 114
#> 4     2858  51.7 102
#> 5     3318  43.9  86
#> 6     3778  36.1 111
#> 7     4238  28.3  93
#> 8     4698  20.6  96
#> 9     5158  12.8 102
#> 10    5618   5.0  97
#>
#> Maximum drop proportions to retain correlation of 0.7 in at least 95% of the samples:
#>
#> closeness: 0.75 (CS-coefficient is highest level tested)
#>   - For more accuracy, run bootnet(..., caseMin = 0.672, caseMax = 1)
#>
#> strength: 0.75 (CS-coefficient is highest level tested)
#>   - For more accuracy, run bootnet(..., caseMin = 0.672, caseMax = 1)
#>
#> Accuracy can also be increased by increasing both 'nBoots' and 'caseN'.
```

```r
#-> strength and closeness can be interpreted
```

Network Stability and Difference Tests

```r
#plot edge weights difference test
plot(boot1, statistics = "edge", plot = "difference",
     onlyNonZero = TRUE, order = "sample")
```

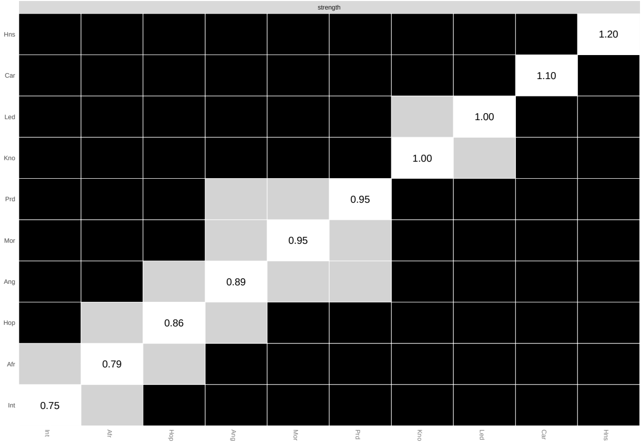
```
#> Expected significance level given number of bootstrap samples is approximately: 0.05
```



Network Stability and Difference Tests

```r
#plot strength difference test
plot(boot1, statistics = "strength", plot = "difference",
     order = "sample")
```

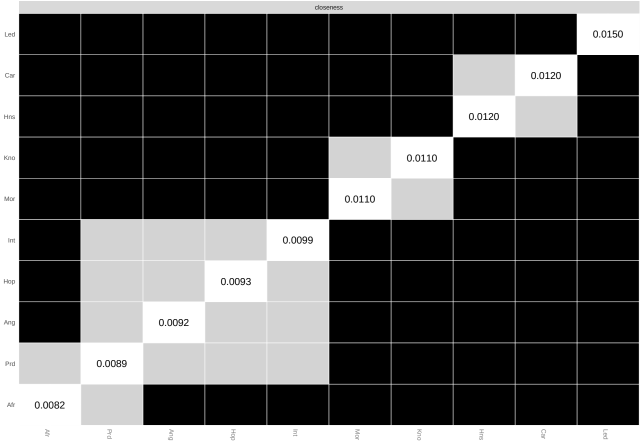
```
#> Expected significance level given number of bootstrap samples is approximately: 0.05
```



Network Stability and Difference Tests

```r
#plot closeness difference test
plot(boot1, statistics = "closeness", plot = "difference",
     order = "sample")
```

```
#> Expected significance level given number of bootstrap samples is approximately: 0.05
```



Estimating the Ising Model

```
#can be estimated via estimateNetwork
#variables automatically binarized at median
#here, feeling variables better binarized at 2 because 1 indicates "no, I never felt this"
split <- describe(data)$median
split[which(groups == "Feelings")] <- 2

#estimate regularized logistic node-wise regression network
#define where to binarize variables
#eLASSO (LASSO with EBIC model selection)
#listwise deletion of missing values (pairwise not possible for regressions)
Ising_net <- estimateNetwork(data, default = "IsingFit", split = split,
                             missing = "listwise", rule = "OR")
```

```
#> Estimating Network. Using package::function:
#>   - IsingFit::IsingFit for network computation
#>     - Using glmnet::glmnet
```

```
#> Warning in bootnet::binarize(data, split = split, verbose = verbose): Splitting
#> data by 3Splitting data by 3Splitting data by 3Splitting data by 4Splitting data
#> by 4Splitting data by 3Splitting data by 2Splitting data by 2Splitting data by
#> 2Splitting data by 2
```

Information about the Ising Model

```r
#extract thresholds
thresholds <- Ising_net$intercepts
thresholds

#extract weights matrix
Wmat_Ising <- getWmat(Ising_net)
Wmat_Ising
```

Comparing the Gaussian Graphical Model and Ising Model

```r
#correlating weights matrices
cor.test(Wmat_GGM[upper.tri(Wmat_GGM)], Wmat_Ising[upper.tri(Wmat_Ising)])
```

```
#>
#>  Pearson's product-moment correlation
#>
#> data:  Wmat_GGM[upper.tri(Wmat_GGM)] and Wmat_Ising[upper.tri(Wmat_Ising)]
#> t = 16.579, df = 43, p-value < 2.2e-16
#> alternative hypothesis: true correlation is not equal to 0
#> 95 percent confidence interval:
#>  0.8752870 0.9610981
#> sample estimates:
#>      cor
#> 0.929904
```

Other model selection

```
###thresholding with significance (bootnet)
GGM_net_thres <- estimateNetwork(data, default = "pcor",
                                 threshold = "sig", alpha = .01)
```

```
#> Estimating Network. Using package::function:
#>   - qgraph::qgraph(..., graph = 'pcor') for network computation
#>   - psych::corr.p for significance thresholding
```

```
plot(GGM_net_thres, layout = layout)
```

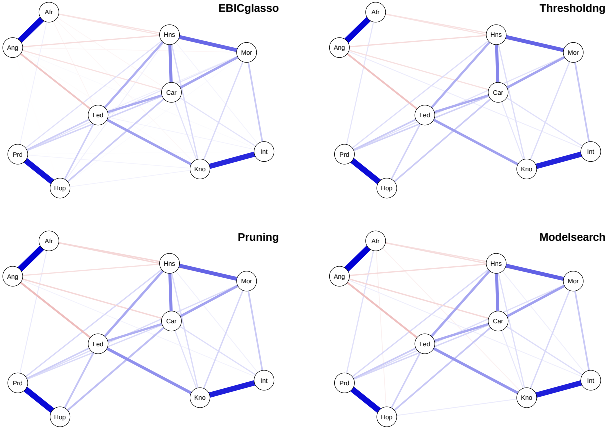
```
###pruning (psychonetrics)
GGM_net_prune <- ggm(data) %>% runmodel %>% prune(alpha = .01)
Wmat_GGM_prune <- getmatrix(GGM_net_prune, "omega")
qgraph(Wmat_GGM_prune, theme = "colorblind",
       layout = layout, labels = names(data))
```

```
###model search with modelsearch (psychonetrics)
GGM_net_modelsearch <- ggm(data) %>% runmodel %>% prune %>% modelsearch
Wmat_GGM_modelsearch <- getmatrix(GGM_net_modelsearch, "omega")
qgraph(Wmat_GGM_modelsearch, theme = "colorblind",
       layout = layout, labels = names(data))
```

Other model selection

```
###compare networks
#-> get maximum edge across all networks
Wmat_GGM_thres <- getWmat(GGM_net_thres)
max_edge <- max(Wmat_GGM, Wmat_GGM_thres,
                Wmat_GGM_prune, Wmat_GGM_modelsearch)
#-> plot networks next to each other
layout(matrix(c(1,2,
                3,4), nrow = 2, ncol = 2, byrow = TRUE))
plot(GGM_net, maximum = max_edge)
title("EBICglasso", adj = 1)
plot(GGM_net_thres, layout = layout, maximum = max_edge)
title("Thresholdng", adj = 1)
qgraph(Wmat_GGM_prune, theme = "colorblind",
       layout = layout, labels = names(data),
       maximum = max_edge)
title("Pruning", adj = 1)
qgraph(Wmat_GGM_modelsearch, theme = "colorblind",
       layout = layout, labels = names(data),
       maximum = max_edge)
title("Modelsearch", adj = 1)
```

Other model selection



Other developments in network analysis in R

See also...

- categorical variables (*mgm*; Haslbeck & Waldorp, 2020, JSS)
- time-series data (idosyncratic and group-level; Epskamp et al., 2018, MBR)
- latent variables in networks (Epskamp et al., 2017, Psychometrika)
- moderated network models (Haslbeck et al., 2021, MBR)
- network comparison (*NCT*; Van Borkulo et al., in press, PM)
- interpretable plots (Jones et al., 2018, Frontiers)
- Bayesian estimation (Williams, 2021, MBR)
- confirmatory network modeling (Kan et al., 2020, JI)
- networks for panel data (Epskamp, 2020, P)
- model selection recommendations (Isvoranu & Epskamp, in press, PM)
- ...

Summary for Network Analysis

Summary

- psychological networks can be estimated via pairwise Markov Random Field Models
  - continuous data: Gaussian Graphical Model
  - binary data: Ising Model
- estimation via model selection
- central nodes can be structurally important
- stability of network parameters should be checked
- *bootnet*, *qgraph*, or *psychonetrics* can estimate and plot networks in R

Thank you very much for your attention!