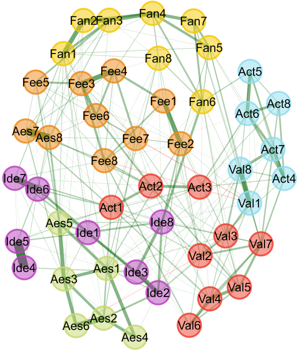
EGA Framework

PSY-GS 8875 Behavioral Data Science



Overview

Overview: Week 13

Readings (Optional)

- Christensen and Golino - 2021 - bootEGA
- Christensen et al. - 2023 - UVA
- Christensen and Golino - 2021 - loadings
- Jamison et al. - 2022
- Jimenez et al. - 2023
- Samo et al. - 2023

Overview | Week 13

- Stability of communities and items
- Local dependence detection
- Network loadings
- Metric invariance
- Hierarchical dimensions

Stability of Communities and Items

Stability of Communities and Items

Stability of Communities and Items

reliability: are your measurements consistent (i.e., can they be repeated)?

- **internal consistency**: whether your items are interrelated – that is, moderate ($r \geq 0.30$ to strongly correlated $r \geq 0.50$)
- **test-retest**: true "reliability" – whether your items can be repeated and are consistent each time you measure them

Stability of Communities and Items

Internal Consistency

$$\text{Cronbach's } \alpha = \frac{k}{k-1}\left(\frac{\sum_{i=1}^{k}\sigma^2_{x_i}}{\sigma_x}\right),$$

where

- k = number of items
- $\sigma^2_{x_i}$ = variance of item i
- σ_x = variance associated with sum total of items $x = \sum_{i=1}^{k} x_i$

For more internal consistency measures, see McNeish (2018)

Stability of Communities and Items

Homogeneity

- Whether a set of items reflect a single underlying construct
- Often implicitly assumed and not usually tested (e.g., unidimensionality)

Stability of Communities and Items

What seems stronger to be a **stronger** statement?

A. internal consistency: items are interrelated

B. homogeneity: items reflect a single underlying construct

Stability of Communities and Items

What seems stronger to be a **stronger** statement?

A. internal consistency: items are interrelated

B. homogeneity: items reflect a single underlying construct

Both psychometric characteristics are important for measurement but are usually tested in a "silo"

Stability of Communities and Items

Stability of Communities and Items

The question we usually want to answer is:

> Do the items hang together in their representative dimensions *taking into account* the other items and dimensions?

Stability of Communities and Items

The question we usually want to answer is:

> Do the items hang together in their representative dimensions *taking into account* the other items and dimensions?

That is, we want to know that our items are internally consistency *and* homogeneous in a multivariate, multidimensional context

Stability of Communities and Items

The question we usually want to answer is:

> Do the items hang together in their representative dimensions *taking into account* the other items and dimensions?

That is, we want to know that our items are internally consistency *and* homogeneous in a multivariate, multidimensional context

⚠ Traditional psychometric approaches do not consider multidimensionality

Stability of Communities and Items

We also want to know whether our dimensions and the items placed in those dimensions are likely to **generalize**

Stability of Communities and Items

We also want to know whether our dimensions and the items placed in those dimensions are likely to **generalize**

Bootstrap Exploratory Graph Analysis

- Bootstrap using resampling with replacement (non-parametric) or multivariate normal data based on correlation matrix (parametric)

- Apply EGA to the replicate bootstrap sample

The community detection algorithm places items into dimensions *automatically*

Stability of Communities and Items

From the bootstraps, we can...

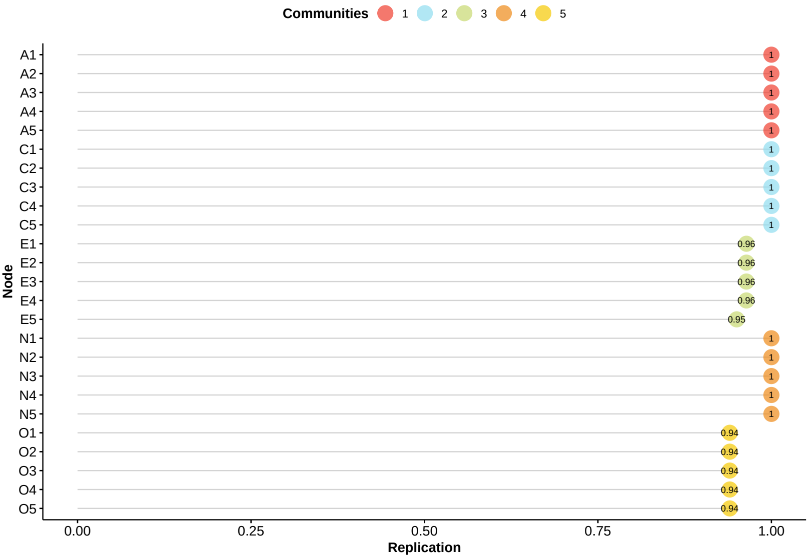
- Determine how frequent the empirical number of dimensions appear across the bootstraps
- Determine how often items are placed into their empirical (or other) dimension
- Determine how often a dimension replicates *exactly* across bootstraps

Stability of Communities and Items

Implementation

```
# Load packages
library(EGAnet); library(psychTools)

# Load data
data <- bfi[,1:25]

# Implement bootstrap EGA (empirical automatically computed)
bfi_boot <- bootEGA(data, seed = 42, ncores = 2)
# Seeds are set independent of R

# Print summary
summary(bfi_boot)

# Print dimension stability summary
summary(bfi_boot$stability)
```

Stability of Communities and Items



Stability of Communities and Items

```
Model: GLASSO (EBIC)
Correlations: auto
Algorithm:  Walktrap
Unidimensional Method:  Louvain

----

EGA Type: EGA
Bootstrap Samples: 500 (Parametric)

                4     5
Frequency:  0.096 0.904

Median dimensions: 5 [4.42, 5.58] 95% CI
```

Stability of Communities and Items

```
EGA Type: EGA
Bootstrap Samples: 500 (Parametric)

Proportion Replicated in Dimensions:

   A1    A2    A3    A4    A5    C1    C2    C3    C4    C5    E1    E2    E3
1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 0.964 0.964 0.964
   E4    E5    N1    N2    N3    N4    N5    O1    O2    O3    O4    O5
0.964 0.950 1.000 1.000 1.000 1.000 1.000 0.940 0.940 0.940 0.940 0.940

----

Structural Consistency:

   1    2    3    4    5
1.00 1.00 0.95 1.00 0.94
```

Stability of Communities and Items

Guidelines

- Empirical solution frequency should be majority
- Item stability (replication) > 0.75
- Dimension stability > 0.75

Resampling (non-parametric) will tend to produce equal or lower estimates to multivariate normal (parametric)

Stability of Communities and Items

R Script

Stability of Communities and Items

Causes of Instability

- Smaller sample sizes
- Local dependence
  - Items will form "minor factors" where a major factor will split into two or more communities
- Multidimensional
  - Items will replicate relatively evenly across two or more communities

Local (In)dependence

Local (In)dependence

Latent Variable Definition

Variables are unrelated after conditioning on a latent variable

- Shared semantic references (e.g., similar item phrasing)
- Shared substantive causes *not* related to the latent variable (e.g., social desirability)
- Conventional psychometric practices such as maximizing Cronbach's α

Local (In)dependence

Network Psychometrics

- Components of the network are defined as "unique causal systems"
- Components are *unique* such that they are causally autonomous (i.e., distinct causal process)
- **Consequence**: variables in the network should be *unique* and **not** redundant

Local (In)dependence

Take a network with many variables that are fairly unique but you have the two items

1. I like to be the center of attention
2. I don't like attention

These two variables will be **strongly** connected (i.e., large edge weight)

Local (In)dependence

Take a network with many variables that are fairly unique but you have the two items

1. I like to be the center of attention
2. I don't like attention

These two variables will be **strongly** connected (i.e., large edge weight)

When evaluating the *node strength* or the sum of the connections to each node in the network, these two variables will likely have *inflated* values

Node strength quantifies how well connected a node is in the network and many researchers take this meaning as "importance"

Local (In)dependence

Take a network with many variables that are fairly unique but you have the two items

1. I like to be the center of attention
2. I don't like attention

These two variables will be **strongly** connected (i.e., large edge weight)

When evaluating the *node strength* or the sum of the connections to each node in the network, these two variables will likely have *inflated* values

Node strength quantifies how well connected a node is in the network and many researchers take this meaning as "importance"

A question arises: Is the strength of these two nodes because they are indeed important or because they are redundant

Local (In)dependence

Unique Variable Analysis

To assess whether there local dependence, Unique Variable Analysis (UVA) can be applied:

1. Estimate a network (usually EBICglasso)
2. Compute weighted topological overlap (wTO) on the network
3. Apply a cut-off (≥ 0.25) to determine redundant pairs
4. Eliminate pairs based on some heuristics

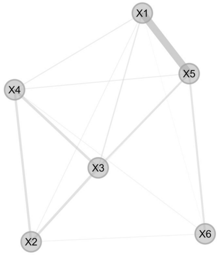
Local (In)dependence

	X_1	X_2	X_3	X_4	X_5	X_6
P_1	3	1	4	1	3	1
P_2	5	5	5	3	5	2
...	...	...	...	...	...	...
P_i	$x_{i,1}$	$x_{i,2}$	$x_{i,3}$	$x_{i,4}$	$x_{i,5}$	$x_{i,6}$

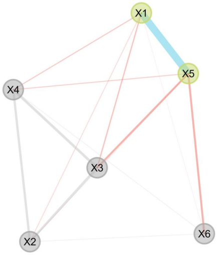
Raw Data

$$\omega_{ij} = \frac{\sum_u a_{iu}a_{uj} + a_{ij}}{\min\{k_i, k_j\} + 1 - a_{ij}}$$

Weighted Topological Overlap (wTO)



Network



Network with wTO

Local (In)dependence

After cut-off, heuristics are used to eliminate redundant variable sets down to a single variable

2 variables: variable with the *lowest* maximum wTO to all *other* variables is retained

3 or more variables: variable with the *highest* mean wTO to all other variables in the *redundant set* is retained

Local (In)dependence

Implementation

```
# Apply UVA
bfi_uva <- UVA(
  data, key = as.character(bfi.dictionary$Item[1:25])
)

# Print summary
summary(bfi_uva)
```

Local (In)dependence

```
Variable pairs with wTO > 0.30 (large-to-very large redundancy)

           node_i                node_j   wto
 Get angry easily. Get irritated easily. 0.431

----

Variable pairs with wTO > 0.25 (moderate-to-large redundancy)

----

Variable pairs with wTO > 0.20 (small-to-moderate redundancy)

                                  node_i                                     node_j   wto
                       Don't talk a lot. Find it difficult to approach others. 0.226
                 Am exacting in my work. Continue until everything is perfect. 0.225
 Am indifferent to the feelings of others.     Inquire about others' well-being. 0.219
            Do things in a half-way manner.                        Waste my time. 0.209
                Know how to comfort others.              Make people feel at ease. 0.207
                       Get angry easily.              Have frequent mood swings. 0.205
                 Have frequent mood swings.                        Often feel blue. 0.204
          Inquire about others' well-being.             Know how to comfort others. 0.203
```

Local (In)dependence

R Script

Effects of Reducing Redundancy

1. More accurate dimension estimation: resolves issues associated with “minor factors” (i.e., smaller dimensions that form because of high shared variance between a smaller set of variables intend to form a dimension in a larger set)

2. More accurate edge weights: associations between variables are due less to redundancy and more to their actual contribution to the network (assuming the network captures all variables of interest)

Local (In)dependence

Effects of Reducing Redundancy

1. More accurate dimension estimation: resolves issues associated with "minor factors" (i.e., smaller dimensions that form because of high shared variance between a smaller set of variables intend to form a dimension in a larger set)

2. More accurate edge weights: associations between variables are due less to redundancy and more to their actual contribution to the network (assuming the network captures all variables of interest)

Is reducing redundancy always necessary?

Network Loadings

Network Loadings

Statistically consistent with factor/component loadings

Loading Definitions

- **factor**: how much one item is related to the factor or how well an item represents and *measures* the latent factor
- **network**: each node's contribution to the *emergence* of a coherent dimension in the network

In most applied circumstances, there is little difference

Network Loadings

In network science, network measures are more common:

- local = a node's position in the network (e.g., centrality)
- meso-scale = sub-strutures such as communities
- global = overall structure of the network (e.g., average shortest path length)

Network Loadings

Centrality (local) measures are still the most commonly applied measures in psychometric networks:

- node strength = absolute sum of a node's connections to other nodes in the network
- expected influence = signed sum of a node's connections to other nodes in the network

💡 There are hundreds of centrality measures but most are problematic with respect to psychometric interpretations (see Bringmann et al., 2019)

Network Loadings

Node Strength

$$S_i = \sum_{j=1}^{n} |w_{ij}|$$

Expected Influence

$$E_i = \sum_{j=1}^{n} w_{ij}$$

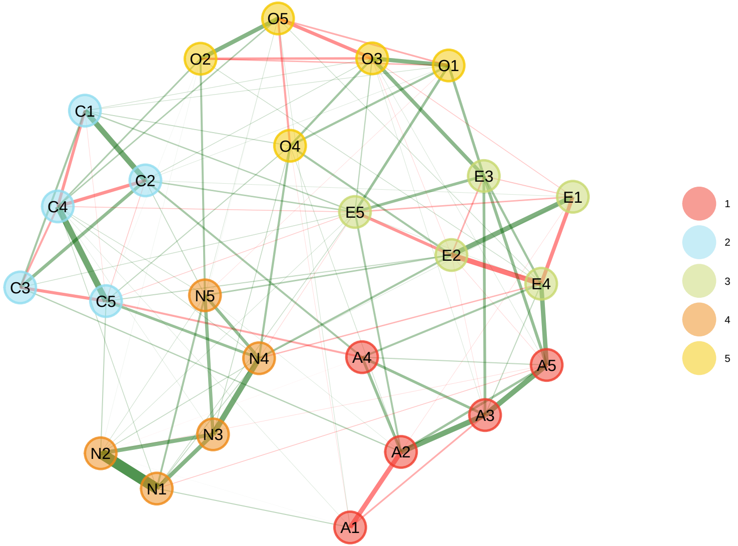
Network Loadings

- Node strength is commonly used as a measure of "influence"
- In psychopathology, many have proposed symptoms highest in node strength as intervention targets
- These interpretations are misleading...
  - Assumes between-person model applies to all individuals in the sample
  - Assumes the network is unidimensional
  - Assumes all variables are unique (i.e., node strength is not due to redundancy)

Network Loadings

```r
# Apply EGA
bfi_ega <- EGA(data)

# Compute node strength
sort(colSums(abs(bfi_ega$network)))
```

Network Loadings



Network Loadings

```
  A1   C3   O2   A4   E1   C1   O4   O5   N5   O1   N3 
0.43 0.63 0.64 0.64 0.66 0.73 0.74 0.75 0.76 0.80 0.92 

  A5   N2   C2   C5   A3   A2   E3   E5   N4   O3   C4 
0.93 0.93 0.94 0.96 1.00 1.01 1.01 1.02 1.03 1.04 1.07 

  N1   E4   E2 
1.07 1.11 1.16 
```

Recall that... *Get angry easily* (`N1`) and *Get irritated easily* (`N2`) were determined to be locally dependent

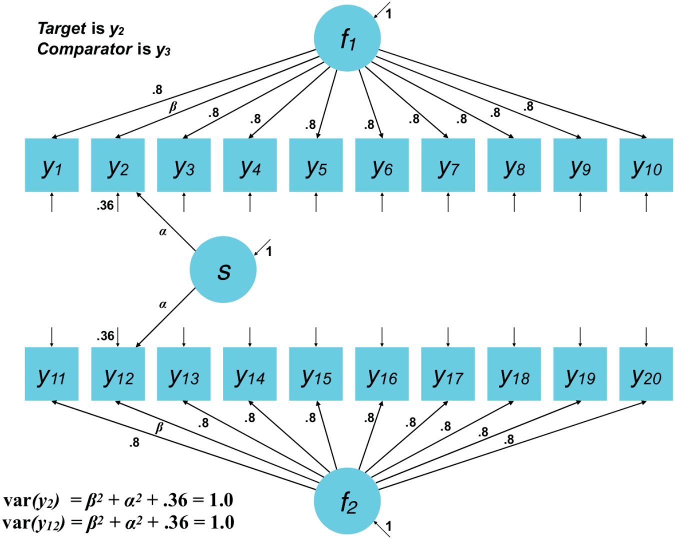
Network Loadings

Connections to Factor Loadings

Hallquist, Wright, and Molenaar (2019)

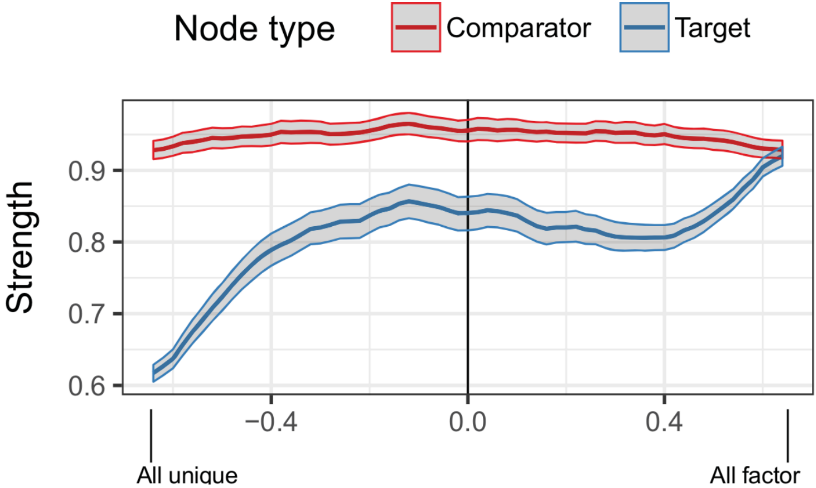
CFA Model	Strength	Closeness	Betweenness
One-factor	0.98	0.94	0.74
Orthogonal Two-factor	0.98	0.42	0.37
Correlated Two-factor	0.97	0.51	0.44
Orthogonal Three-factor	0.98	0.42	0.31
Correlated Three-factor	0.97	0.55	0.41

Hallquist, Wright, and Molenaar (2019)



Network Loadings

Hallquist, Wright, and Molenaar (2019)



Network Loadings

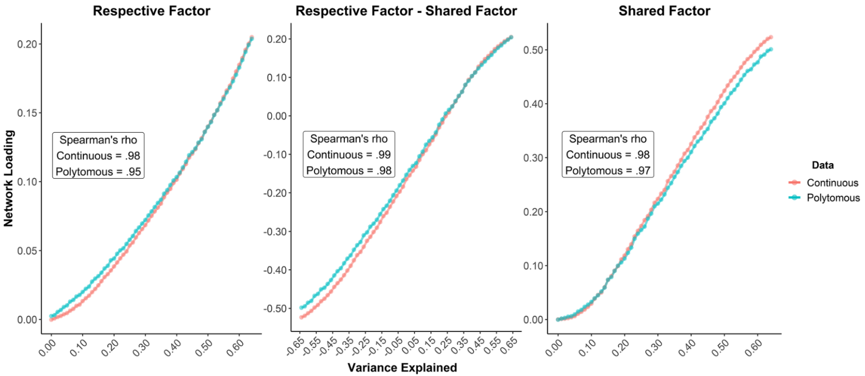
Solution: split node strength by community, c

$$L_{ic} = \sum_{j \in c}^{C} |w_{ij}|$$

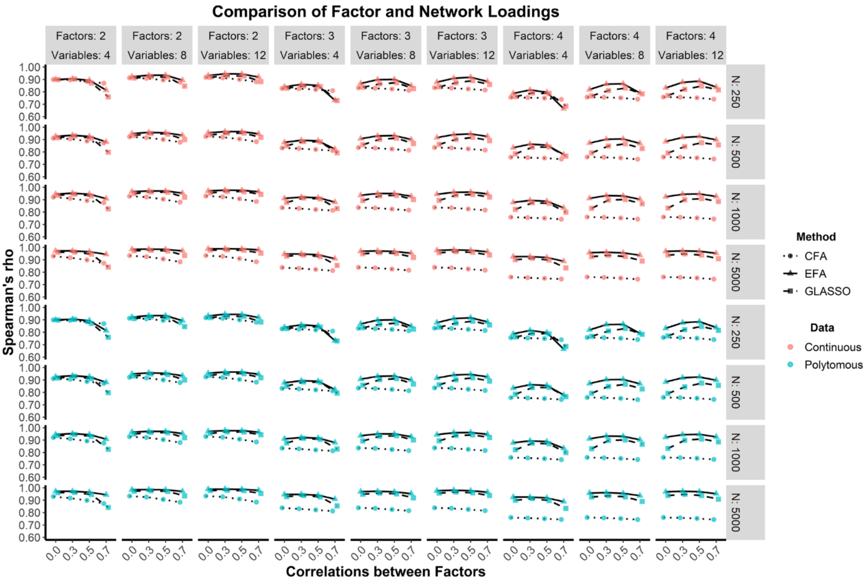
$$\aleph_{ic} = \frac{L_{ic}}{\sqrt{\sum L_{.c}}}$$

with signs added afterward

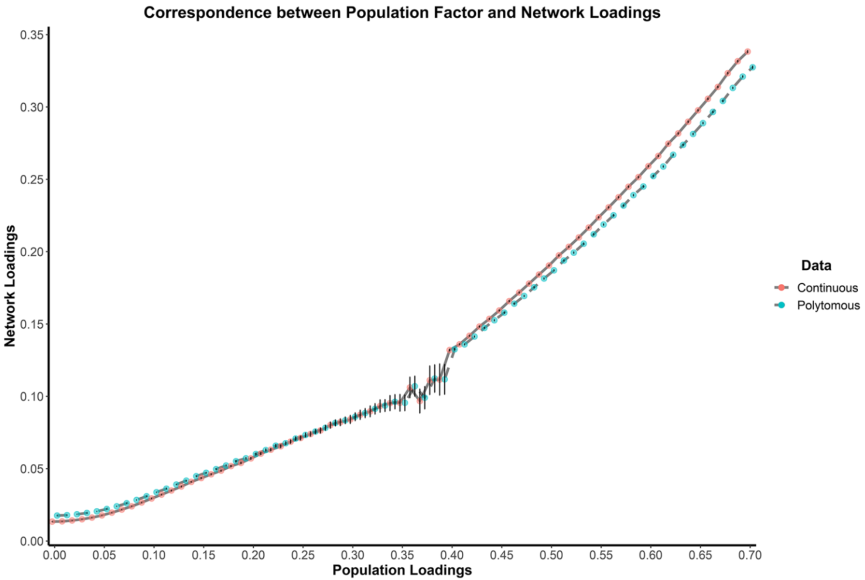
Network Loadings



Network Loadings



Network Loadings



Network Loadings

There were some lingering issues though...

1. Negative signs were added post-hoc and in a way that didn't always align

2. Community-assigned loadings were sometimes *smaller* than their cross-loadings (impossible with factor analysis)

3. Magnitudes are significantly affected by number of variables per community

Revised Loadings

$$\text{within } \ell_{i,c} = n_c \left(\frac{\sum_{j=1}^{n_c} t_{i,j}}{n_c - 1} \right),$$

and

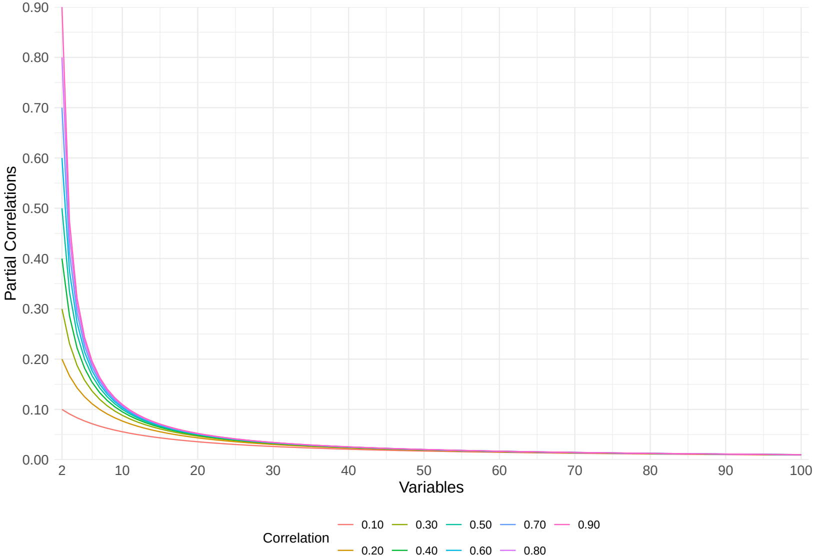
$$\text{between } \ell_{i,k} = \sum_{i=1}^{n_c} \sum_{j=1}^{n_k} w_{i \in c, j \in k},$$

where

- $t_{i,j}$ = target community sub-matrix with node i and j in community c
- n_c = number of nodes in the assigned community, c
- n_k = number of nodes in a community, k, that is not c

Network Loadings

Guttman (1953): as $n_c \to \infty$, then $r_{xy|z} \to 0$



Network Loadings

Revised Loadings

$$\aleph_{i,c} = \frac{\ell_{i,c}}{\sqrt[\log(\zeta n_c)]{\sum_{j=1}^{n_c} |\text{within } \ell_{j,c}|}},$$

where

- $\log(n_c)$ = natural logarithm of the number of variables in community c
- ζ = scaling factor for loading size (defaults to 2)

Network Loadings

There were some lingering issues though...

1. Negative signs were added post-hoc and in a way that didn't always align ☑
2. Community-assigned loadings were sometimes *smaller* than their cross-loadings (impossible with factor analysis) ☑
3. Magnitudes are significantly affected by number of variables per community ⚙

Network Loadings

Okay... but why network loadings at all?

1. Need for community-aligned loadings

$\rightarrow$ Number of communities in factor analysis does not guarantee alignment with variable assignments

2. Network loadings are *unrotated*

3. Networks have fewer assumptions than factor models

$\rightarrow$ Psychometric reference for when factor models don't work

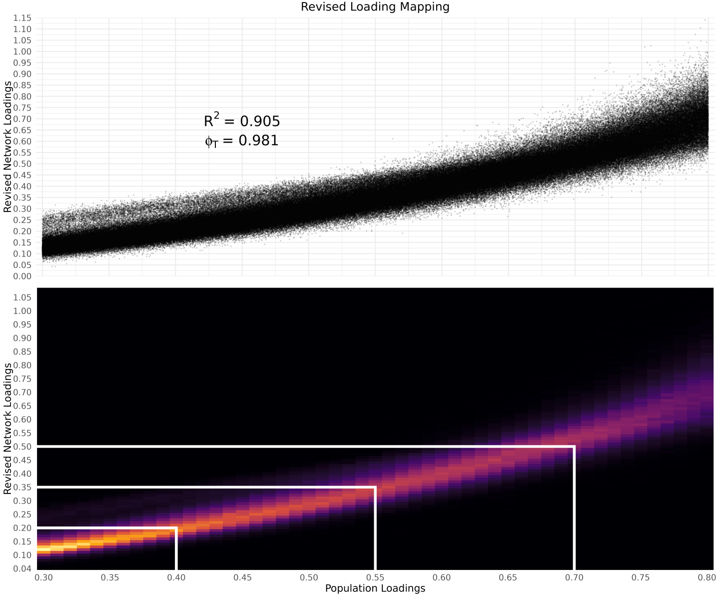
Network Loadings

```r
# Compute network loadings
bfi_loadings <- net.loads(
  bfi_ega, loading.method = "experimental"
)$std[colnames(bfi_ega$network),] # standardized
```

Network Loadings

	1	2	3	4	5
A1	**-0.24**	-0.01	0.00	0.03	-0.03
A2	**0.58**	0.04	0.07	0.01	0.02
A3	**0.57**	0.00	0.12	0.00	0.01
A4	**0.25**	0.11	0.06	0.00	0.00
A5	**0.33**	0.00	0.24	-0.03	0.01
C1	0.00	**0.39**	0.04	0.00	0.07
C2	0.06	**0.46**	0.03	0.02	0.03
C3	0.04	**0.37**	0.02	0.00	0.00
C4	-0.01	**-0.52**	-0.02	0.06	-0.10
C5	-0.06	**-0.37**	-0.05	0.10	0.03
E1	-0.01	0.01	**-0.41**	0.01	-0.02
E2	-0.02	-0.03	**-0.56**	0.09	0.05
E3	0.17	0.00	**0.29**	0.00	0.20
E4	0.23	0.00	**0.44**	-0.04	-0.04
E5	0.06	0.14	**0.29**	0.02	0.12
N1	-0.05	-0.02	0.02	**0.59**	0.00
N2	-0.01	-0.03	0.02	**0.54**	0.00
N3	0.00	-0.01	0.00	**0.56**	0.03
N4	0.00	-0.10	-0.13	**0.35**	0.07
N5	0.01	0.01	-0.05	**0.30**	-0.09
O1	0.00	0.03	0.16	-0.01	**0.37**
O2	0.00	-0.05	0.02	0.06	**-0.31**
O3	0.02	0.04	0.18	0.00	**0.48**
O4	0.03	0.00	-0.06	0.08	**0.26**
O5	-0.02	-0.04	0.02	0.01	**-0.45**

Network Loadings



Network Loadings

Network loadings open the door for many different traditional psychometric procedures

- **group comparison (with dimensionality)**
- **network scores (and hierarchical dimensionality)**
- conversion of loadings to IRT parameters (Muraki & Carlson, 1995)

Metric Invariance

Metric Invariance

Metric Invariance

Group comparison is often a goal in the social sciences

Many methods have been developed to make group comparisons in network psychometrics

- Fused GLASSO
- Network Comparison Test
- Group-as-Moderator
- Bayesian Posteriors

All of these methods implicitly treat the network as unidimensional

Metric Invariance

Motivating Example

Using the Big Five data as an example, let's say we want to examine whether there are any personality differences between those with a college degree and those without

```r
# Obtain groups
groups <- ifelse(bfi[,"education"] < 4, "Non-grad", "Grad")

# Filter for missing groups
group_data <- data[!is.na(groups),]
groups <- na.omit(groups)

# Frequencies
table(groups)
```

```
groups
    Grad Non-grad
     812     1765
```

Procedure

A permutation-based procedure can be employed to test for differences between groups (assuming some structure holds for both groups):

1. Estimate networks and network loadings for both groups
2. Compute the difference between the *assigned* loadings (τ)
3. Permutation: shuffle group label and repeat steps 1. and 2. for P times (e.g., 500; tau_{R_p})
4. Compute $\sum_{p=1}^{P} |tau| \geq tau_{R_p}$ to obtain p-values

Significant differences ($p < 0.05$) suggest non-invariance (group differences exist) whereas $p > 0.05$ suggest invariance (group differences **do not** exist)

Metric Invariance

R Script

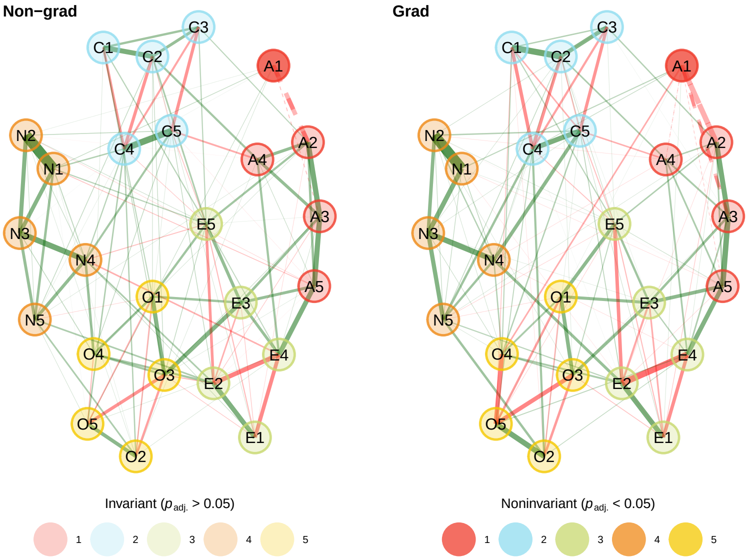
Metric Invariance

```r
# Perform metric invariance
bfi_invariance <- invariance(
  data = group_data, groups = groups,
  structure = rep(1:5, each = 5), # theoretical structure
  loading.method = "experimental", # use latest loadings
  ncores = 8, seed = 42
)

# Summary
summary(bfi_invariance)

# Plot
plot(bfi_invariance, p_type = "p_BH")
```

Metric Invariance

```
   Membership Difference     p  p_BH sig       Direction
A1          1      0.181 0.002 0.050  ** Non-grad > Grad
A2          1      0.094 0.070 0.438   .                
A3          1     -0.032 0.570 0.679                    
A4          1      0.042 0.286 0.596                    
A5          1     -0.036 0.376 0.609                    
C1          2     -0.076 0.136 0.486                    
C2          2     -0.084 0.096 0.480   .                
C3          2      0.003 0.940 0.940                    
C4          2     -0.036 0.504 0.630                    
C5          2      0.004 0.936 0.940                    
E1          3      0.033 0.414 0.609                    
E2          3      0.122 0.018 0.150   * Non-grad > Grad
E3          3      0.038 0.410 0.609                    
E4          3      0.044 0.318 0.609                    
E5          3      0.048 0.264 0.596                    
N1          4      0.042 0.278 0.596                    
N2          4     -0.004 0.914 0.940                    
N3          4     -0.052 0.128 0.486                    
N4          4      0.010 0.796 0.905                    
N5          4      0.039 0.234 0.596                    
O1          5      0.038 0.470 0.630                    
O2          5      0.042 0.360 0.609                    
O3          5      0.035 0.484 0.630                    
O4          5     -0.064 0.158 0.494                    
O5          5      0.222 0.008 0.100  ** Non-grad > Grad
----
Signif. code: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 'n.s.' 1
```

Metric Invariance



Metric Invariance

Non-invariant Items

Item	Description	Significance	Direction
A1	Am indifferent to the feelings of others.	p_BH	Non-grad $>$ Grad
E2	Find it difficult to approach others.	p	Non-grad $>$ Grad
O5	Will not probe deeply into a subject.	p	Non-grad $>$ Grad

Metric Invariance

Group differences can be examined with the network psychometric framework *accounting for* the community structure

Tends to show comparable accuracy to traditional methods (e.g., SEM) with some advantage for disparate sample sizes (see Jamison et al., 2022)

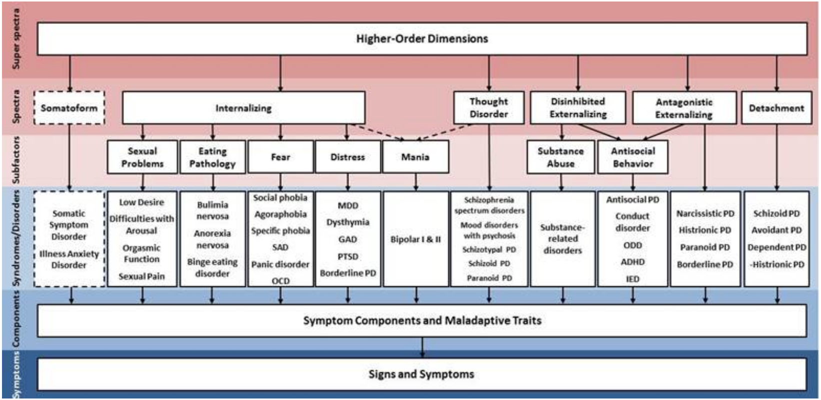
Scores can be computed based on the network loadings using $\mathbf{X}$ (available using `net.scores()`)

Hierarchical Dimensions

Hierarchical Dimensions

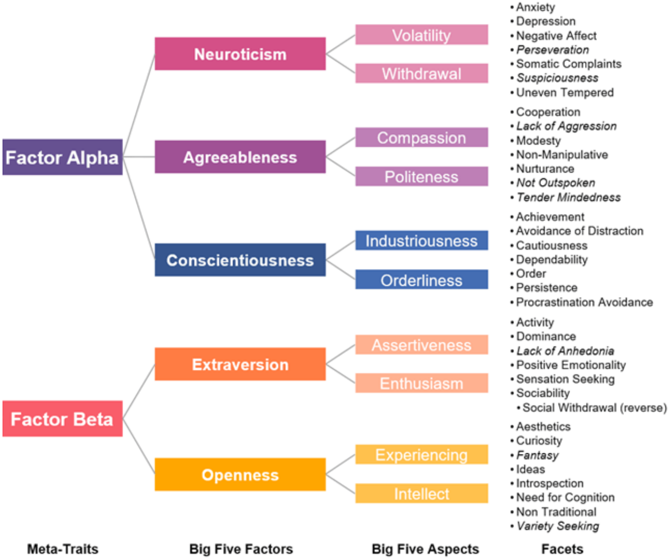
Hierarchical Dimensions

Many psychological phenotypes are theorized to be hierarchically structured



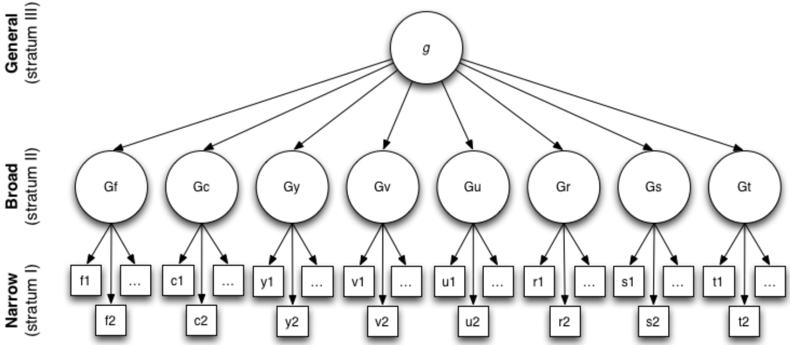
Hierarchical Dimensions

Many psychological phenotypes are theorized to be hierarchically structured



Hierarchical Dimensions

Many psychological phenotypes are theorized to be hierarchically structured



Hierarchical Dimensions

Motivating Example: Synthetic Aperture Personality Assessment

The SPI (SAPA Personality Inventory) is a set of 135 items primarily selected from the International Personality Item Pool

Extensive factor analytic and psychometric analyses (Condon, 2017) have arrived at the "Little" 27 lower-order and can be narrowed to a 70-item Big Five (e.g., last week's AHA)

Motivating Question: Do we find the Little 27 and Big Five using hierarchical EGA?

Hierarchical Dimensions

Hierarchical EGA

1. Apply EGA using the *first pass* of the Louvain algorithm to obtain the lower order dimensions
2. Estimate network loadings and compute network scores based on lower order dimensions
3. Apply EGA to the network scores to obtain the higher order dimensions

Caveat

Remember: the Louvain algorithm results can change with node ordering

This stochastic nature of the algorithm is more acute at the lowest level (i.e., first pass)

To mitigate this issue, an approach known as *consensus clustering* can be used

Hierarchical Dimensions

Consensus Clustering (Lancichinetti & Fortunato, 2012)

1. Randomly shuffle node order
2. Apply Louvain algorithm
3. Repeat 1. and 2. for N times (e.g., 1000)
4. Obtain *most common* community structure across N applications

Result: more consistent (and accurate) results

Application

```r
# Obtain SAPA data
sapa <- psychTools::spi[,11:145]

# Apply hierarchical EGA
sapa_hier <- hierEGA(
  data = sapa,
  loading.method = "experimental",
  scores = "network"
)

# Summary
summary(sapa_hier)

# Plot
plot(sapa_hier)
```

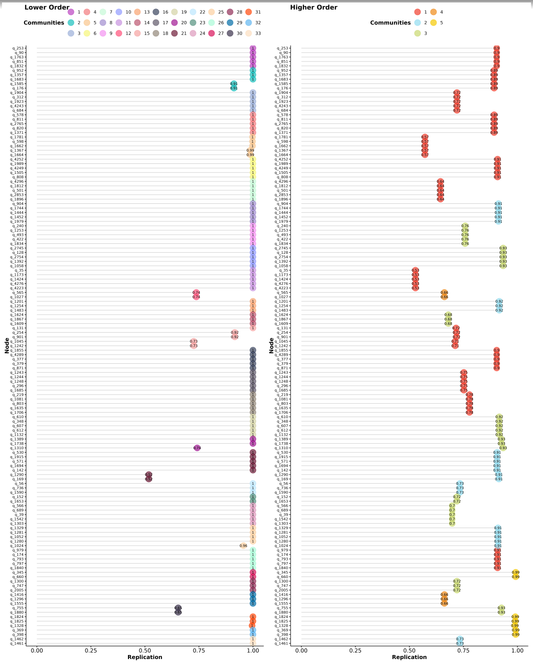
Hierarchical Dimensions

R Script

How Stability are these Dimensions?

```
# Apply bootstrap hierarchical EGA
sapa_hier_boot <- bootEGA(
  data = sapa, EGA.type = "hierEGA",
  loading.method = "experimental",
  scores = "network",
  ncores = 8, seed = 42
)

# Summary
summary(sapa_hier_boot)
```

Hierarchical Dimensions



Summary

Summary

`bootEGA` = determine the dimension and item stability as well as potential for problematic items

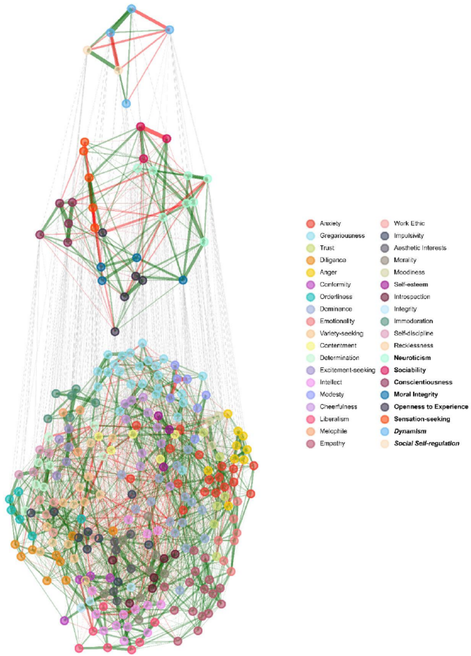
`UVA` = determine redundancies in the network (or local dependence in latent variable modeling)

`net.loads` = estimate network loadings

`invariance` = compute metric invariance based on network psychometrics

`net.scores` = compute network scores based on network loadings

`hierEGA` = estimate hierarchical dimensionality



Anxiety
Gregariousness
Trust
Diligence
Anger
Conformity
Orderliness
Dominance
Emotionality
Variety-seeking
Contentment
Determination
Excitement-seeking
Intellect
Modesty
Cheerfulness
Liberalism
Melophile
Empathy
Work Ethic
Impulsivity
Aesthetic Interests
Morality
Moodiness
Self-esteem
Introspection
Integrity
Immoderation
Self-discipline
Recklessness
Neuroticism
Sociability
Conscientiousness
Moral Integrity
Openness to Experience
Sensation-seeking
Dynamism
Social Self-regulation

Dynamic Readings

Dynamic Readings

Derivatives: Deboeck et al. (2009)

Dynamic EGA: Golino et al. (2022)

Vector autoregression networks: Epskamp et al. (2018)

GIMME: Beltz and Gates (2017)

Heterogeneity in Dynamic Structures

- Golino et al. (2023)
- Santoro and Nicosia (2020)
- De Domenico et al. (2015)