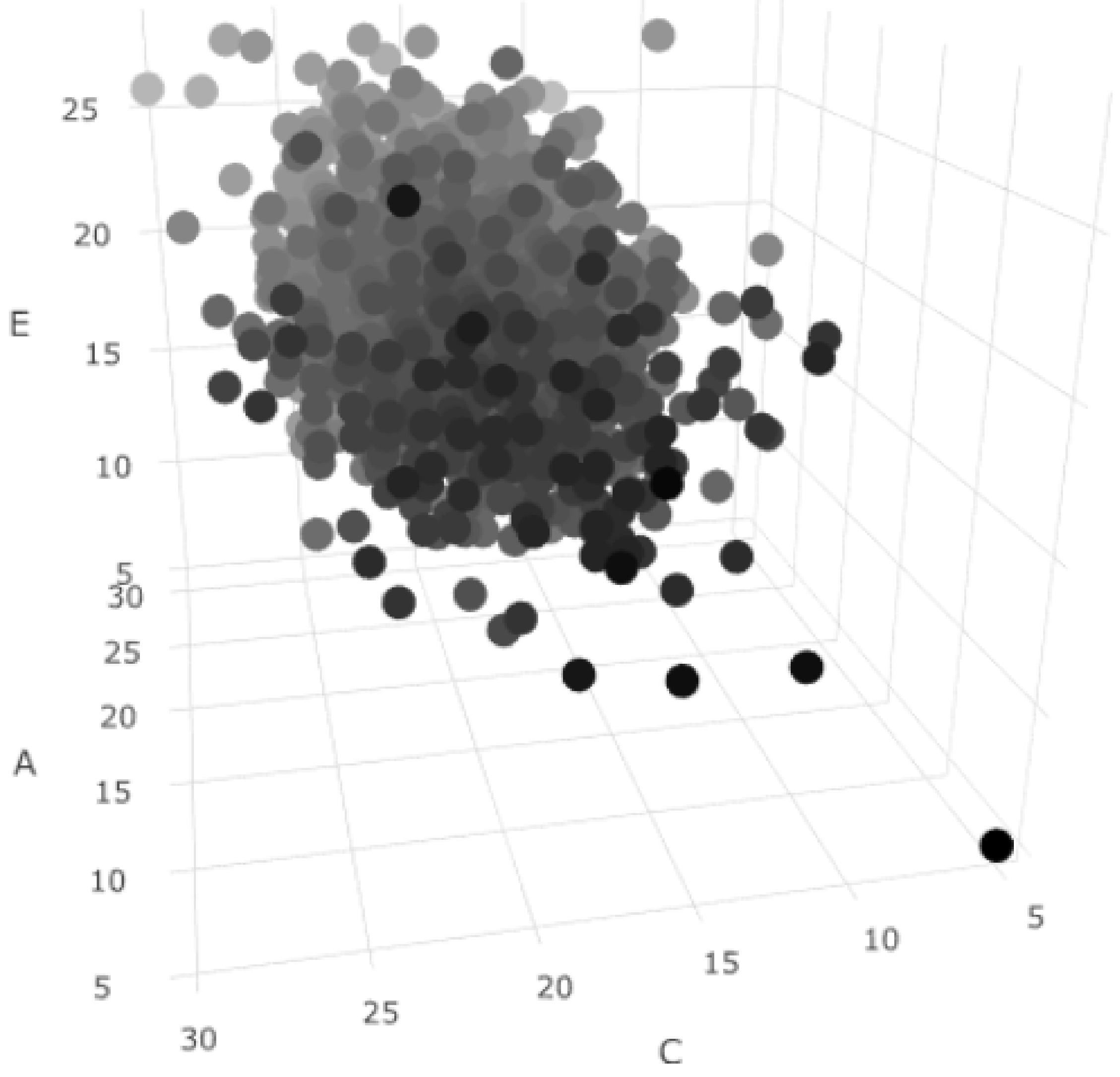
Determining dimensionality

FACTOR ANALYSIS IN R

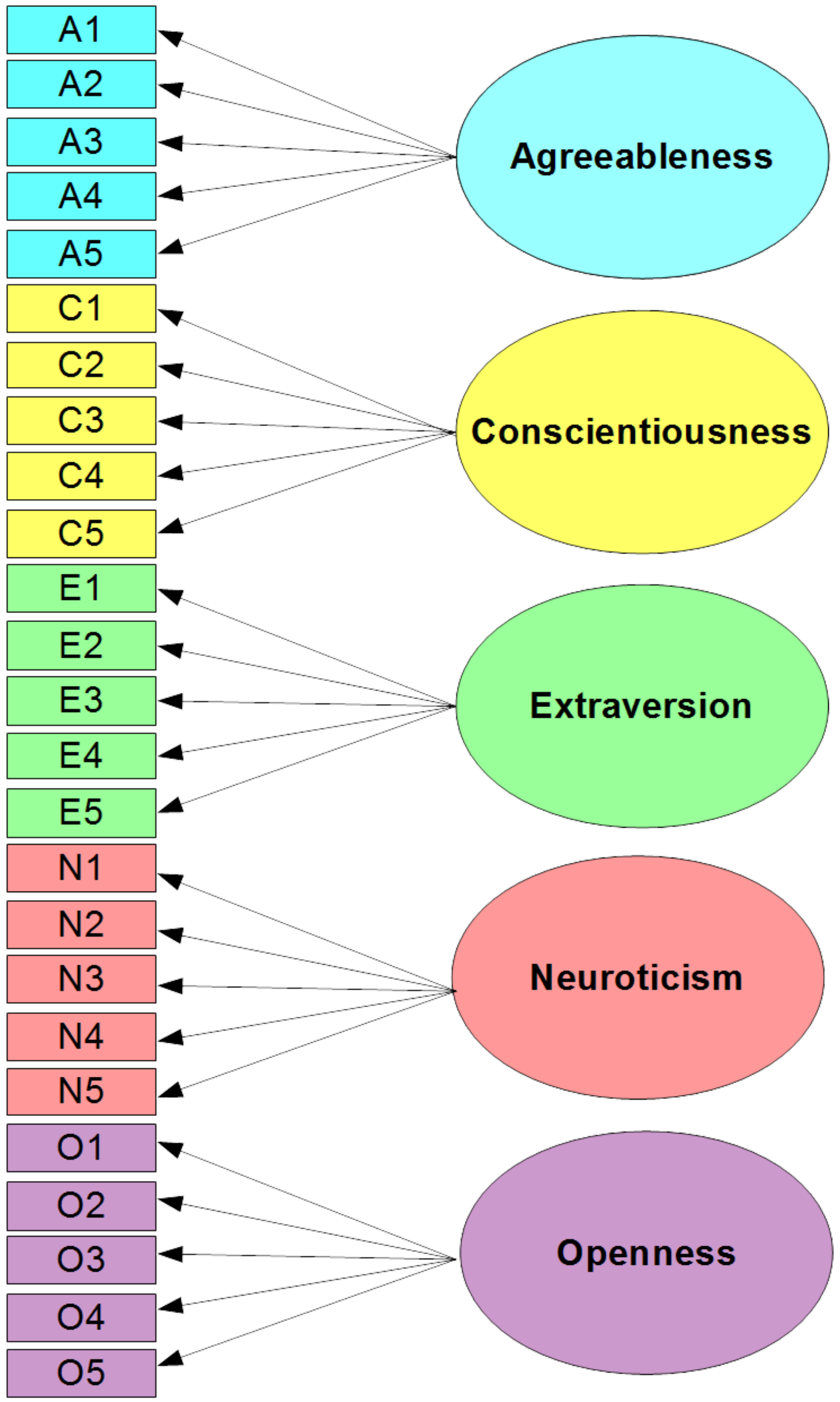
Jennifer Brussow
Psychometrician

How many dimensions does your data have?



The bfi dataset

- Big Five Inventory
- 2,800 subjects
- 25 questions
- Data collected from the Synthetic Aperture Personality Assessment (SAPA)


A1
A2
A3
A4
A5
C1
C2
C3
C4
C5
E1
E2
E3
E4
E5
N1
N2
N3
N4
N5
O1
O2
O3
O4
O5
Agreeableness
Conscientiousness
Extraversion
Neuroticism
Openness

1 = Very Inaccurate ... 6 = Very Accurate

```
head(bfi)
```

```
      A1 A2 A3 A4 A5 C1 C2 C3 C4 C5 E1 E2 E3 E4 E5 N1 N2 N3 N4 N5 O1  ...
61617  2  4  3  4  4  2  3  3  4  4  3  3  3  4  4  3  4  2  2  3  3  ...
61618  2  4  5  2  5  5  4  4  3  4  1  1  6  4  3  3  3  3  5  5  4  ...
61620  5  4  5  4  4  4  5  4  2  5  2  4  4  4  5  4  5  4  2  3  4  ...
61621  4  4  6  5  5  4  4  3  5  5  5  3  4  4  4  2  5  2  4  1  3  ...
61622  2  3  3  4  5  4  4  5  3  2  2  2  5  4  5  2  3  4  4  3  3  ...
61623  6  6  5  6  5  6  6  6  1  3  2  1  6  5  6  3  5  2  2  3  4  ...
```

```
names(bfi)
```

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

Setup: split your dataset

```
# Establish two sets of indices to split the dataset
N <- nrow(bfi)
indices <- seq(1, N)
indices_EFA <- sample(indices, floor((.5*N)))
indices_CFA <- indices[!(indices %in% indices_EFA)]
# Use those indices to split the dataset into halves for your EFA and CFA
bfi_EFA <- bfi[indices_EFA, ]
bfi_CFA <- bfi[indices_CFA, ]
```

Setup: split your dataset

```
head(bfi_EFA, 2)
```

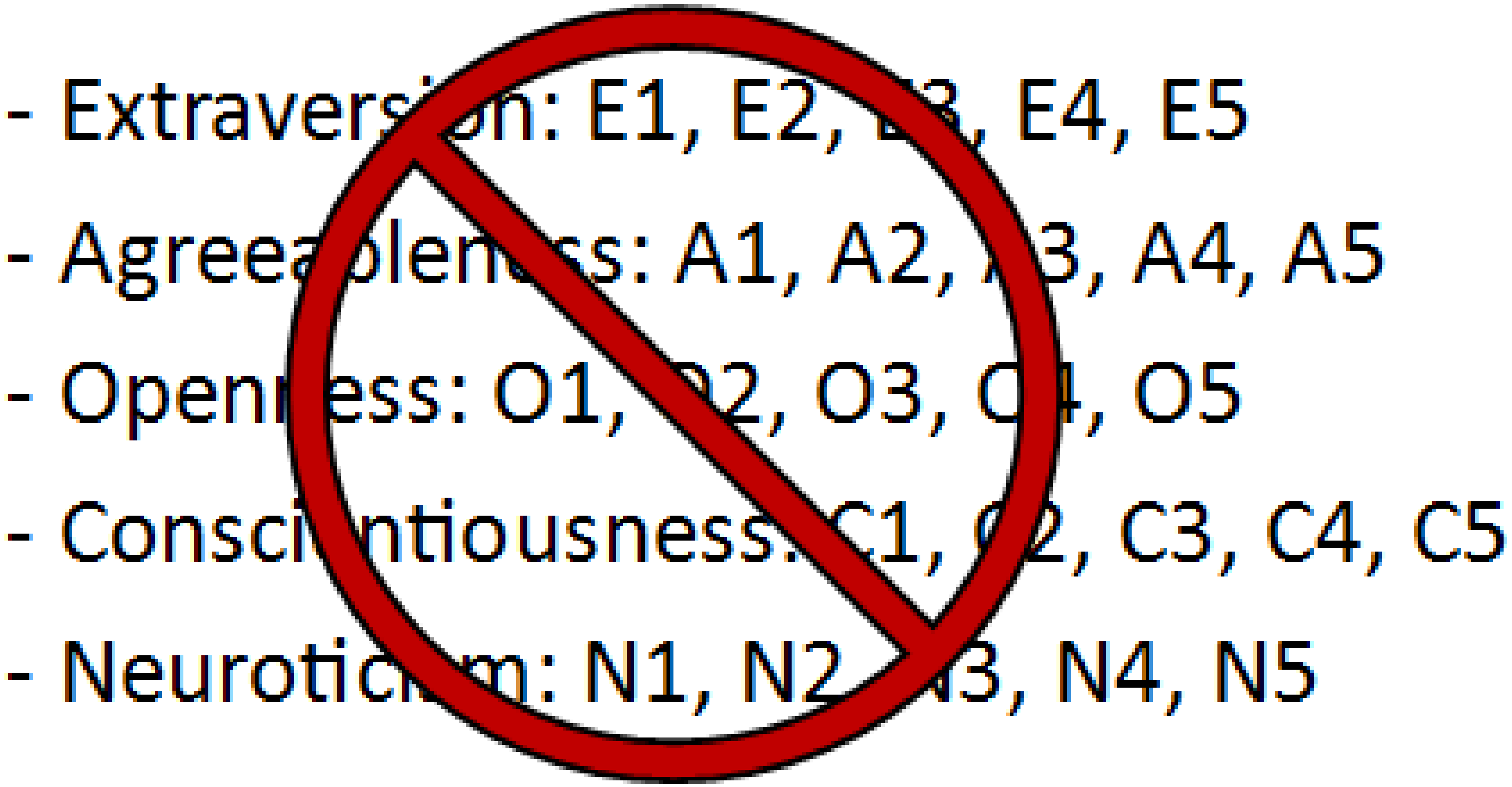
```
      A1 A2 A3 A4 A5 C1 C2 C3 C4 C5 E1 E2 E3 E4 E5 N1 N2 N3 N4 N5 O1  ...
65237  3  4  4  4  4  4  4  5  2  3  3  4 NA  4  4  4  3  1  3  2  4  ...
61825  3  1  2  2  2  2  1  2  6  6  6  6  1  1  1  3  5  4  4  4  5  ...
```

```
head(bfi_CFA, 2)
```

```
      A1 A2 A3 A4 A5 C1 C2 C3 C4 C5 E1 E2 E3 E4 E5 N1 N2 N3 N4 N5 O1  ...
61617  2  4  3  4  4  2  3  3  4  4  3  3  3  4  4  3  4  2  2  3  3  ...
61621  4  4  6  5  5  4  4  3  5  5  5  3  4  4  4  2  5  2  4  1  3  ...
...
```

An empirical approach to dimensionality

Imagine we have no theory...



Calculate the correlation matrix

```
# Calculate the correlation matrix first
bfi_EFA_cor <- cor(bfi_EFA, use = "pairwise.complete.obs")
```

```
            A1          A2          A3          A4          A5         C1 ...
A1  1.00000000 -0.31920397 -0.25651343 -0.12441523 -0.20083692  0.058252
A2 -0.31920397  1.00000000  0.46698961  0.30599175  0.36599749  0.075002
A3 -0.25651343  0.46698961  1.00000000  0.32762347  0.47616038  0.089720
A4 -0.12441523  0.30599175  0.32762347  1.00000000  0.27182236  0.083987
A5 -0.20083692  0.36599749  0.47616038  0.27182236  1.00000000  0.116890
C1  0.05825219  0.07500228  0.08972097  0.08398741  0.11689059  1.000000
C2  0.04236764  0.12843266  0.10471200  0.22697628  0.09639765  0.421518
C3 -0.02289831  0.18618382  0.14009601  0.09975850  0.13797236  0.301556
C4  0.09865372 -0.11178917 -0.11576273 -0.15035049 -0.10248897 -0.354081
C5  0.04925038 -0.10820392 -0.15392300 -0.24998065 -0.15667123 -0.269701
...
```

Eigenvalues

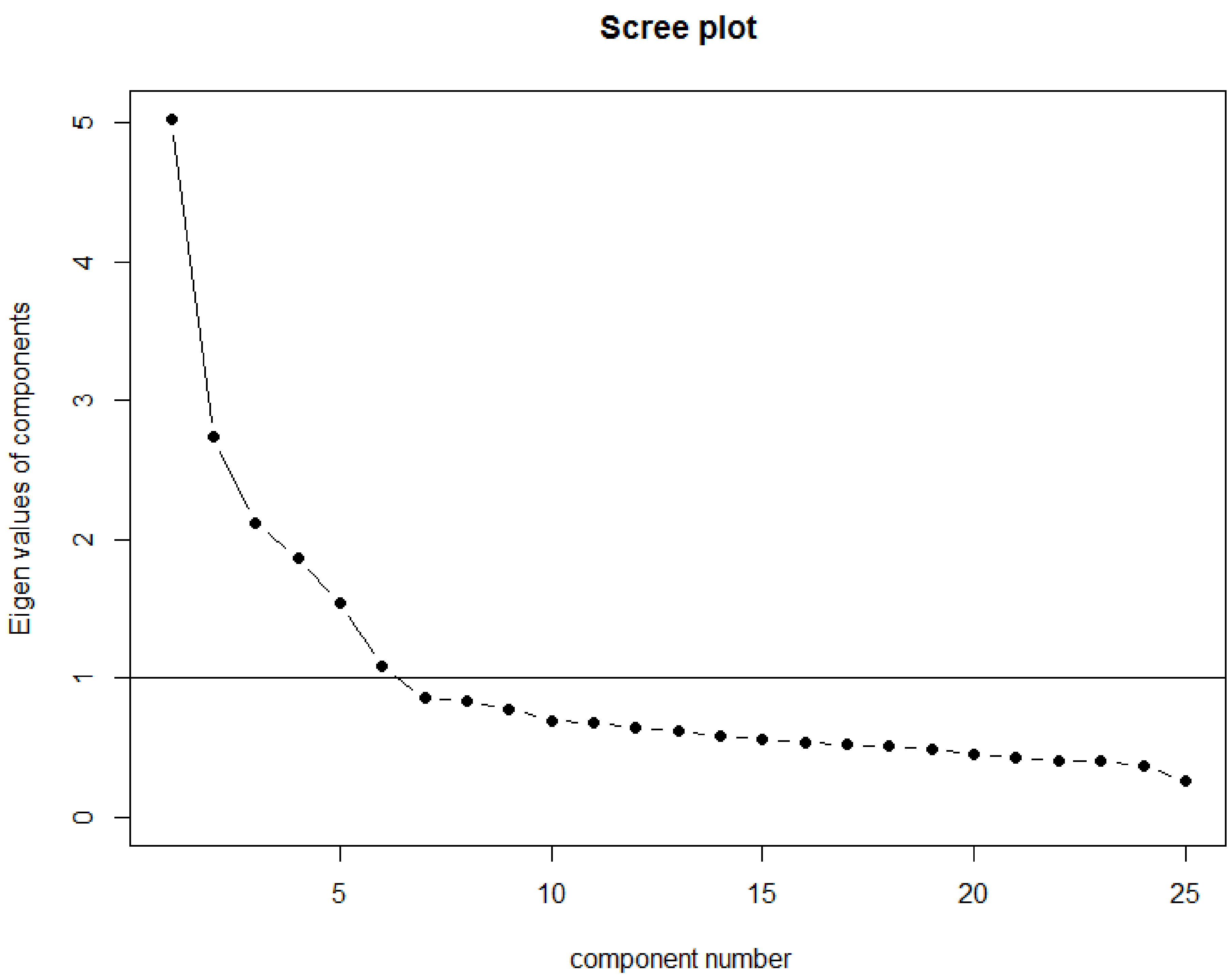
```
# Calculate the correlation matrix first
bfi_EFA_cor <- cor(bfi_EFA, use = "pairwise.complete.obs")

# Then use that correlation matrix to create the scree plot
scree(bfi_EFA_cor, factors = FALSE)
```

Scree plots

```
# Calculate the correlation matrix first
bfi_EFA_cor <- cor(bfi_EFA, use = "pairwise.complete.obs")

# Then use that correlation matrix to create the scree plot
scree(bfi_EFA_cor, factors = FALSE)
```


Scree plot
Eigen values of components
component number
0
1
2
3
4
5
10
15
20
25

Let's practice!

FACTOR ANALYSIS IN R

Multidimensionality: What does it mean?

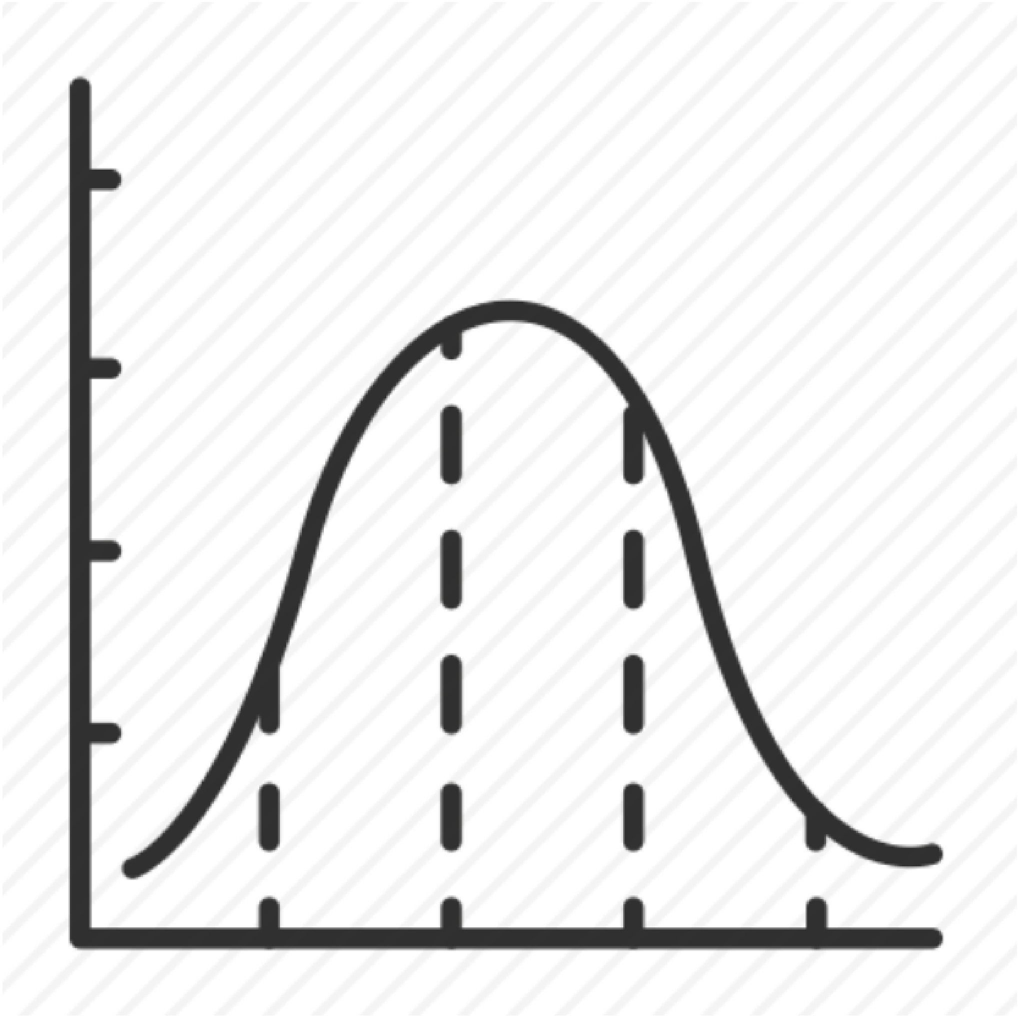
FACTOR ANALYSIS IN R

Jennifer Brussow
Psychometrician

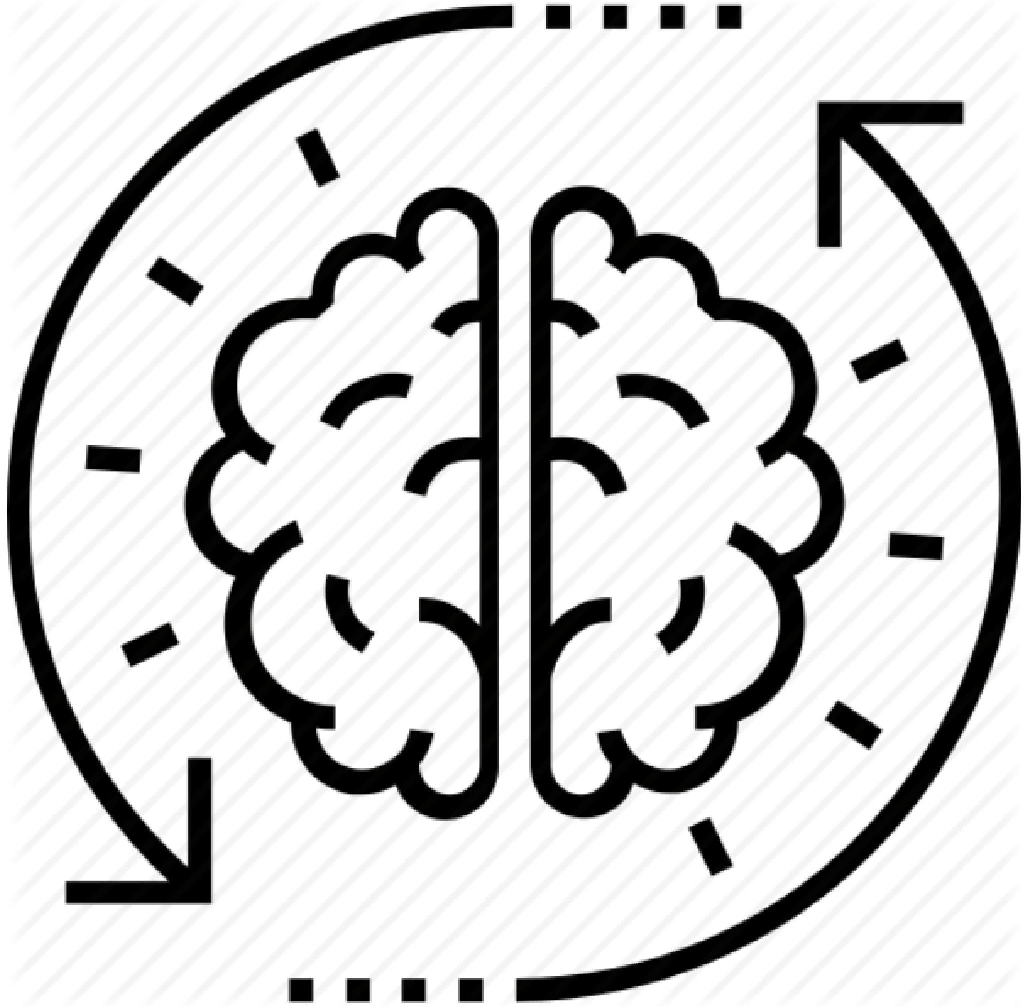
Factors = constructs

- **Construct:** an attribute of interest
  - Can't be directly measured
- Examples:
  - Self-determination
  - Reasoning ability
  - Political affiliation
  - Extraversion

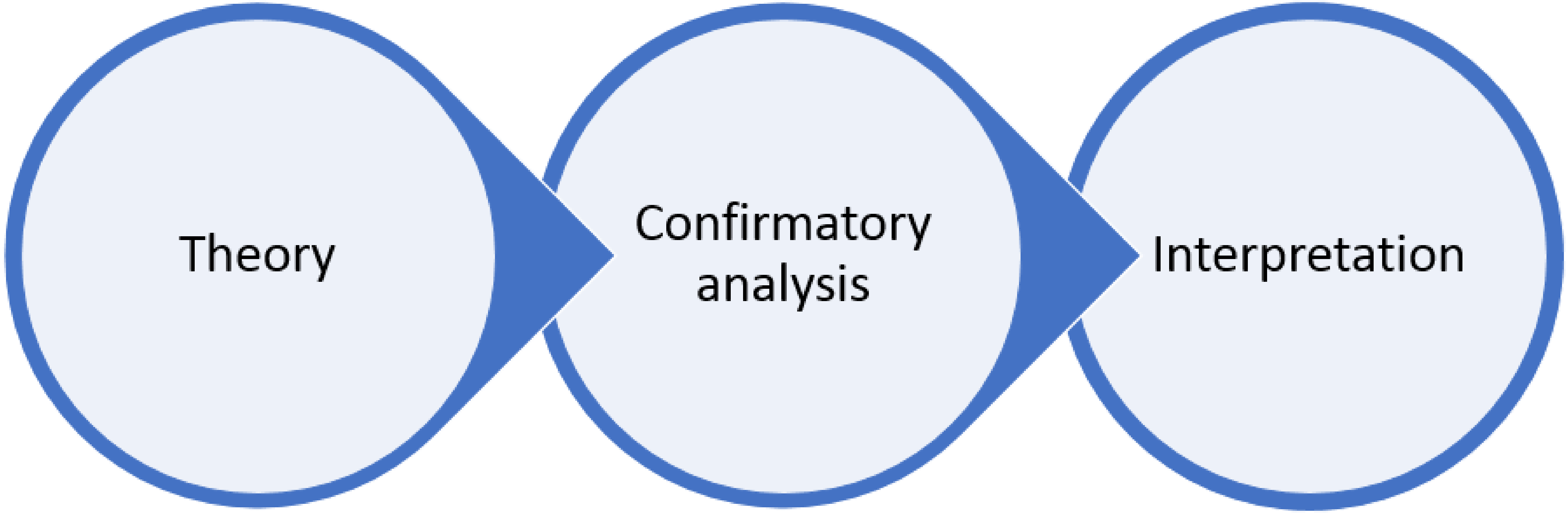
Factors = constructs



=

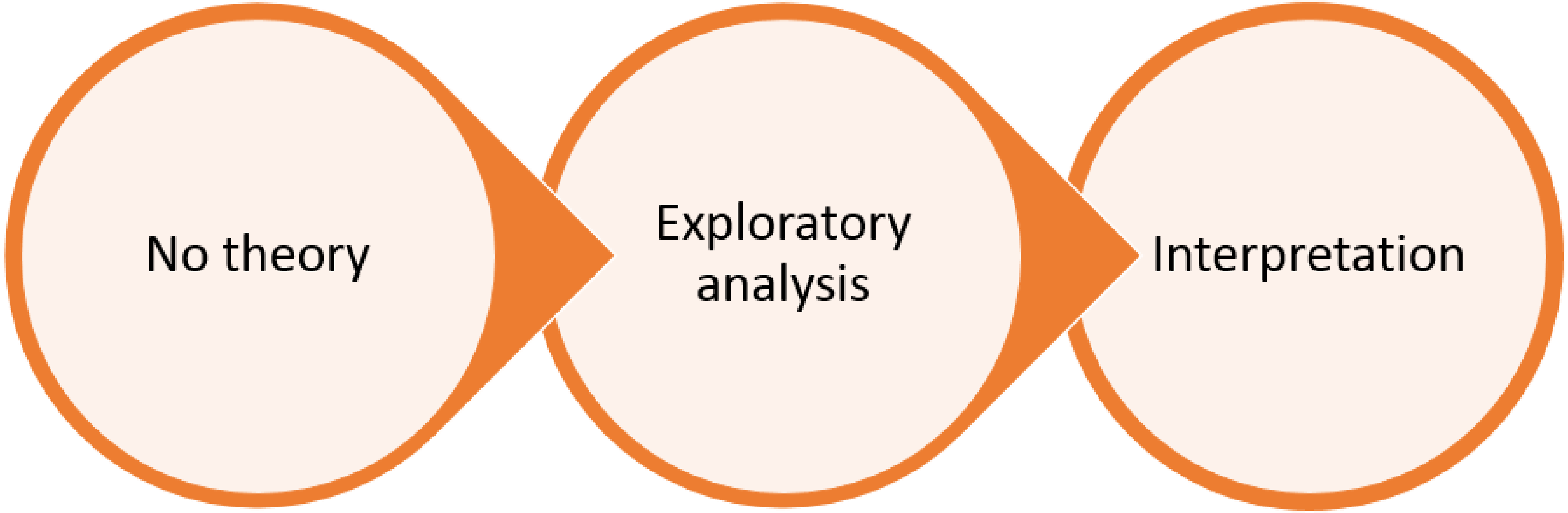


Interpreting confirmatory analyses



- Model fit: How well the hypothesized model fits the data
- Factor loadings: How well the items measure their corresponding constructs

Interpreting exploratory analyses



- Difficult to interpret without theory!
- Factor loadings: Meaning can sometimes be inferred from patterns

```r
# Run the EFA with six factors (as indicated by your scree plot)
EFA_model <- fa(bfi_EFA, nfactors = 6)
# View results from the model object
EFA_model
```

```
Factor Analysis using method =  minres
Call: fa(r = bfi_EFA, nfactors = 6)
Standardized loadings (pattern matrix) based upon correlation matrix
     MR2   MR1   MR3   MR5   MR4   MR6   h2   u2 com
A1  0.10 -0.09  0.07 -0.56  0.11  0.28 0.35 0.65 1.8
A2  0.05 -0.01  0.08  0.69 -0.02  0.01 0.49 0.51 1.0
A3 -0.04 -0.13  0.03  0.57  0.11  0.09 0.47 0.53 1.3
A4 -0.05 -0.08  0.19  0.35 -0.07  0.19 0.25 0.75 2.5
A5 -0.17 -0.20  0.00  0.42  0.20  0.17 0.46 0.54 2.7
C1  0.01  0.07  0.54 -0.07  0.21  0.07 0.35 0.65 1.4
C2  0.09  0.14  0.63  0.01  0.17  0.16 0.46 0.54 1.4
...
```

```
EFA_model$loadings
```

```
Loadings:
   MR2    MR1    MR3    MR5    MR4    MR6   
A1                      -0.559  0.109  0.285
A2                       0.685              
A3        -0.129         0.569  0.113       
A4                0.193  0.348         0.189
A5 -0.172 -0.200         0.421  0.201  0.166
C1                0.542         0.214       
C2         0.138  0.631         0.170  0.157
C3         0.128  0.532  0.110              
C4               -0.683         0.118  0.229
C5  0.103  0.172 -0.599         0.131       
E1 -0.158  0.589  0.133 -0.116         0.106
E2         0.694                            
E3        -0.343         0.104  0.468       
E4        -0.565         0.184         0.255
E5  0.171 -0.408  0.275         0.216       
```

Factor scores

```
head(EFA_model$scores)
```

```
             MR2        MR1        MR3         MR5        MR4         MR6
65237         NA         NA         NA          NA         NA          NA
61825  0.4731267 2.21345215 -2.7650759 -2.72096751 -0.9357389 -1.54036174
67417  0.5217166 0.15834190 -2.1790559  0.47053433  0.4909513 -0.49268634
62051 -1.3333104 -1.32520518  1.0266578 -0.07063958 -0.3670002 -0.07978805
63767 -1.6844911 -1.45769993  1.7776350  1.01101859  0.7490857 -0.35677764
66734 -0.7014448  0.06174358 -0.3530992 -0.05968920 -0.4435187 -0.75311430
```

- WARNING: Do not interpret factor scores until you have a theory!

Let's practice!

FACTOR ANALYSIS IN R

Model fit

FACTOR ANALYSIS IN R

Jennifer Brussow
Psychometrician

Absolute vs. relative model fit

Absolute fit statistics have intrinsic meaning and suggested cutoff values.

- Chi-square test
- Tucker-Lewis Index (TLI)
- Root Mean Square Error of Approximation (RMSEA)

Relative fit statistics only have meaning when comparing models.

- Bayesian Information Criterion (BIC)

Absolute fit statistics

Commonly used cutoff values:

- Chi-square test: Non-significant result
- Tucker Lewis Index (TLI): > 0.90
- Root Mean Square Error of Approximation (RMSEA): < 0.05

Finding the fit statistics

```
# Run the EFA with six factors (as indicated by your scree plot)
EFA_model <- fa(bfi_EFA, nfactors = 6)
# View results from the model object
EFA_model
```

```
The total number of observations was  1400
    with Likelihood Chi Square =  618.43  with prob <  1.2e-53
Tucker Lewis Index of factoring reliability =  0.916
RMSEA index =  0.045  and the 90 % confidence intervals are  0.041 0.048
BIC =  -576.87
```

Relative model fit

```
# Run each theorized EFA on your dataset
bfi_theory <- fa(bfi_EFA, nfactors = 5)
bfi_eigen <- fa(bfi_EFA, nfactors = 6)
```

```
# Compare the BIC values
bfi_theory$BIC
bfi_eigen$BIC
bfi_theory$BIC
bfi_eigen$BIC
```

```
-381.5326
-576.8658
```

In sum: evaluating fit

1. Make sure your model has good absolute fit (chi-square test, TLI, RMSEA)
2. If you are comparing multiple models, use relative fit statistics (BIC)

Let's practice!

FACTOR ANALYSIS IN R