Introduction to Exploratory and Confirmatory Factor Analysis (Using R)

Princeton University

Jason Geller, PH.D.

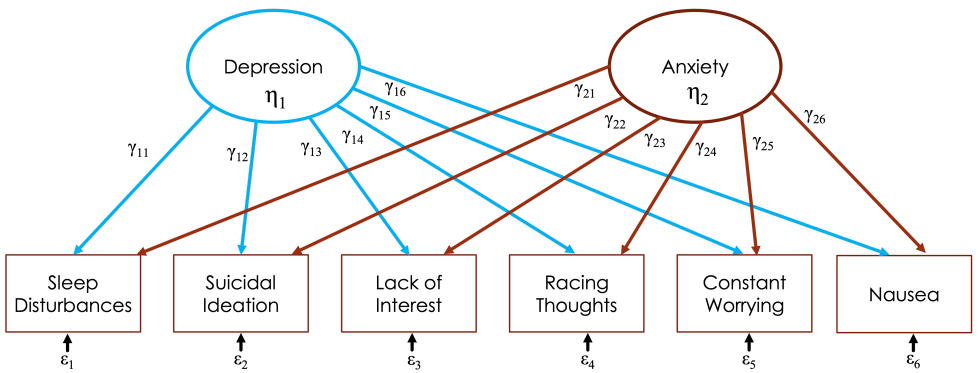
4/8/23

How does it work?

- **Let's say we have 6 items in a scale:**
  - Sleep disturbances (insomnia/hypersomnia)
  - Suicidal ideation
  - Lack of interest in normally engaging activities
  - Racing thoughts
  - Constant worrying
  - Nausea
- FA "looks" at the relationships between these items and finds that some of them seem to hang together

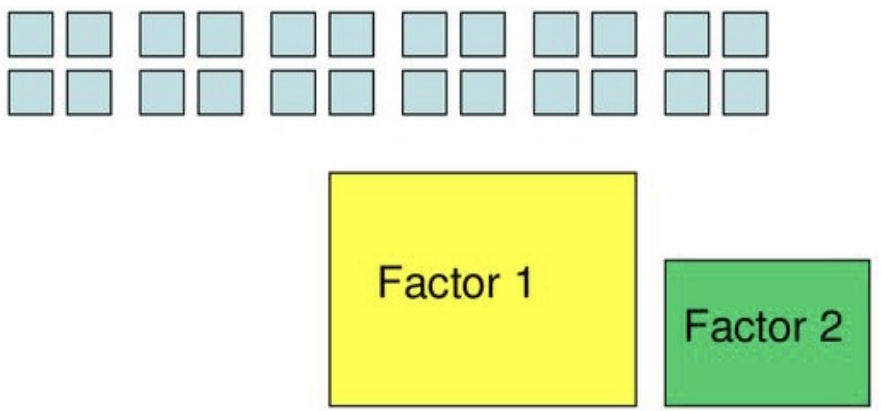
How does it work?

- Let's say we have 6 items in a scale:
  - **Sleep disturbances (insomnia/hypersomnia)**
  - **Suicidal ideation**
  - **Lack of interest in normally engaging activities**
  - Racing thoughts
  - Constant worrying
  - Nausea
    - Some of these could cross-load
    - FA considers this and items load on all factors



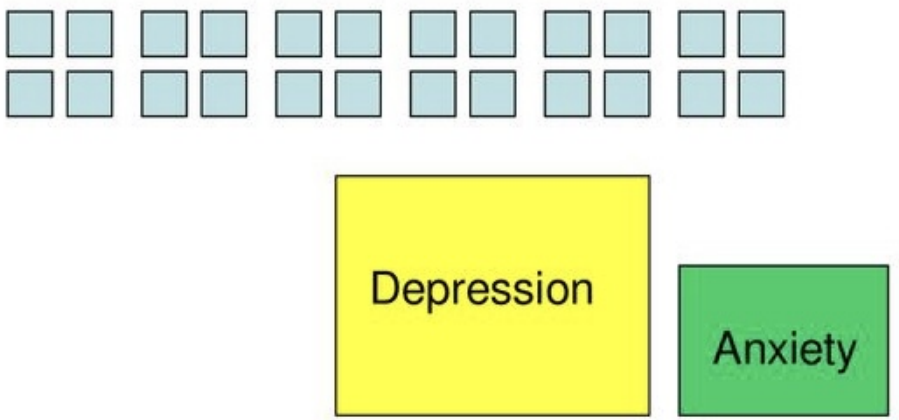
Uses

- Simplify data
  - 6 variables to 2 variables



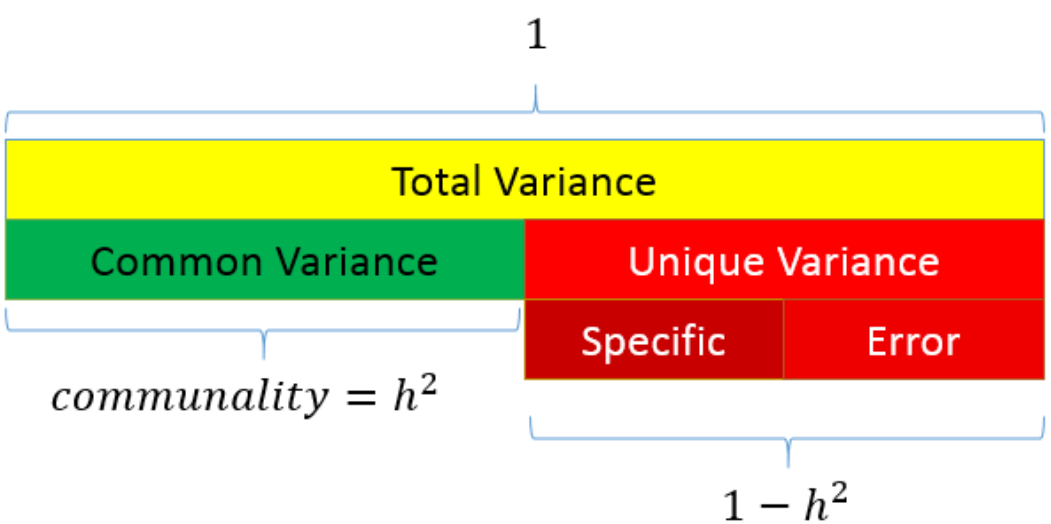
Uses

- Identify underlying constructs
  - Depression and anxiety



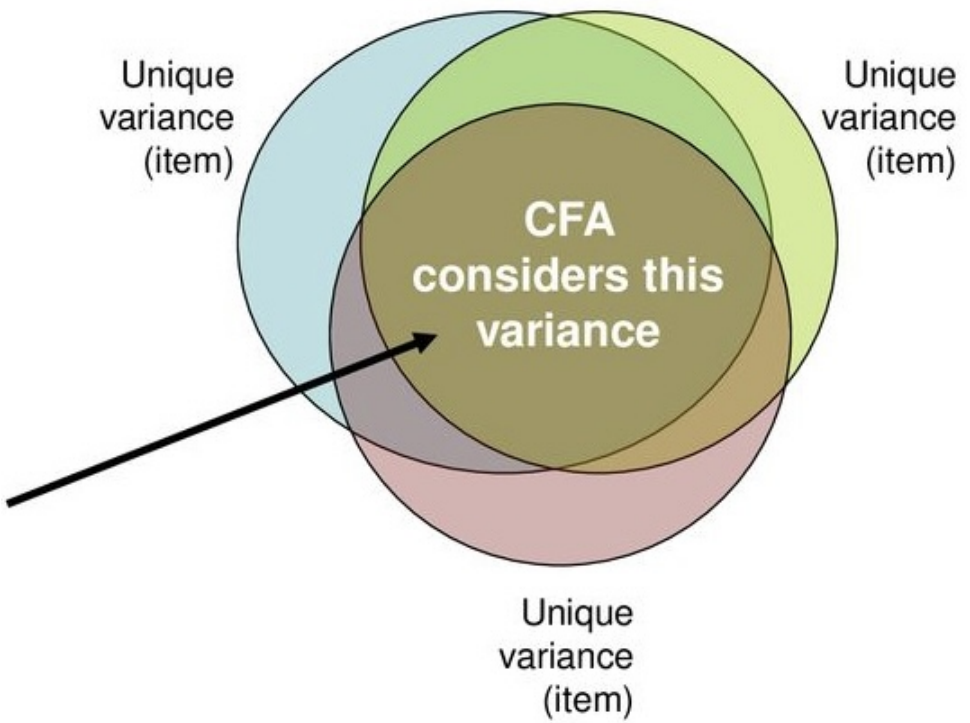
Partitioning Variance

1. Variance common to other variables
    - Communality h^2: **proportion of each variable's/item's variance that can be explained by the factors**
        - How much an item is related to other items in the analysis
2. Variance specific to that variable
3. Random measurement error



Common factor analysis

- Common factor analysis
  - Attempts to achieve parsimony (data reduction) by:
    - Explaining the *maximum amount of common variance* in a correlation matrix
      - Using the *smallest* number of explanatory constructs (factors)



Common factor analysis

> Partitions variance that is in common with other variables. How?

- Use multiple regression
  - Each item as an outcome in MR
  - Use all other items as predictors
  - Finds the communality among all of the variables, relative to one another

Common factor analysis

Predictors:

Item 2

Item 3

Item 4

Item 5

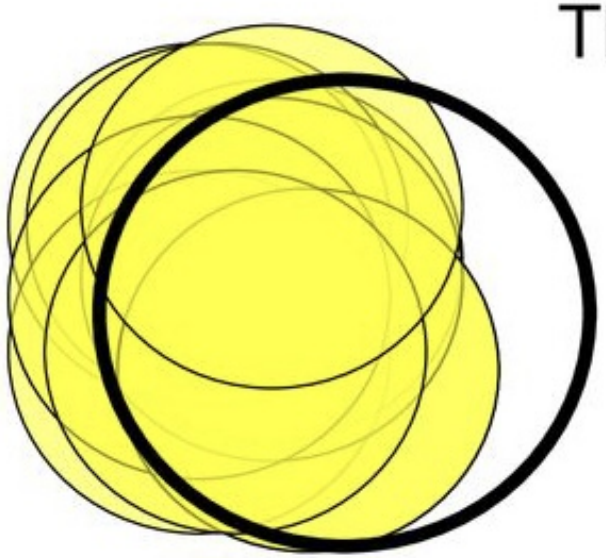
Item 6

Item 7

Item 8

Item 9

Item 10



Outcome:

Item 1

The R square is the average shared variance for that item with the other items

Item 1

Common factor analysis

Predictors:

Item 1

Item 3

Item 4

Item 5

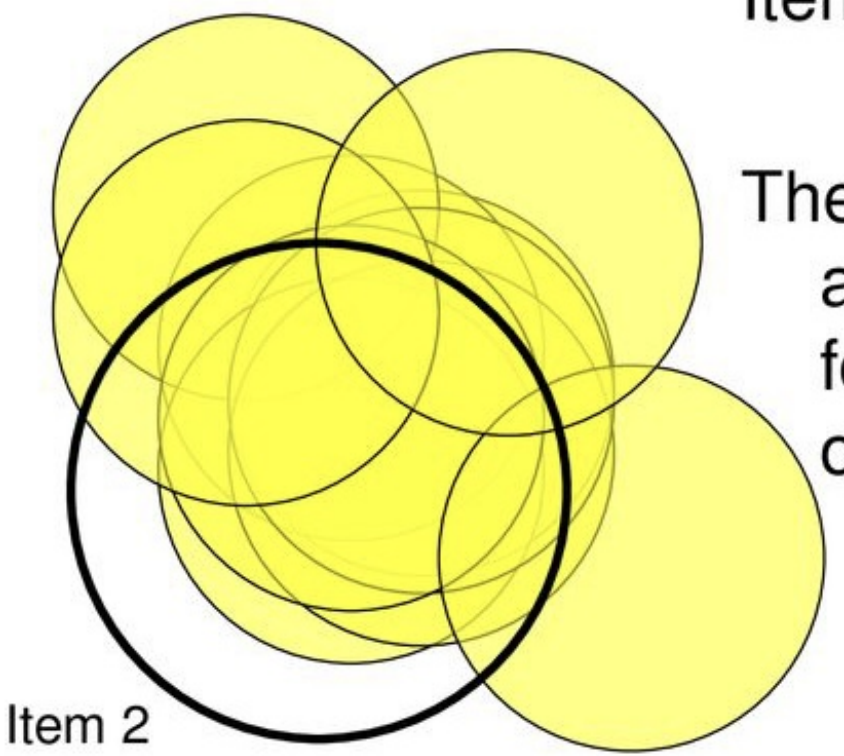
Item 6

Item 7

Item 8

Item 9

Item 10



Outcome:

Item 2

The average R square is the average shared variance for that item with the other items

Common factor analysis

Predictors:

Item 1

Item 2

Item 4

Item 5

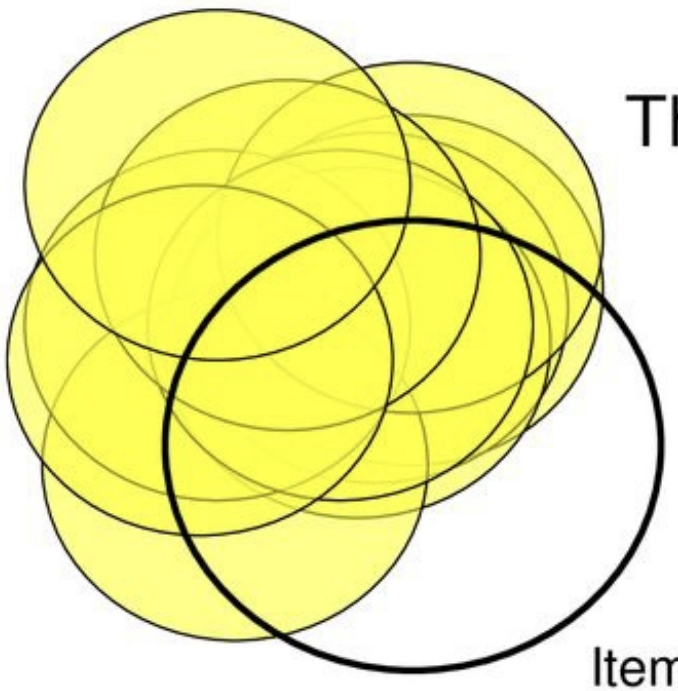
Item 6

Item 7

Item 8

Item 9

Item 10

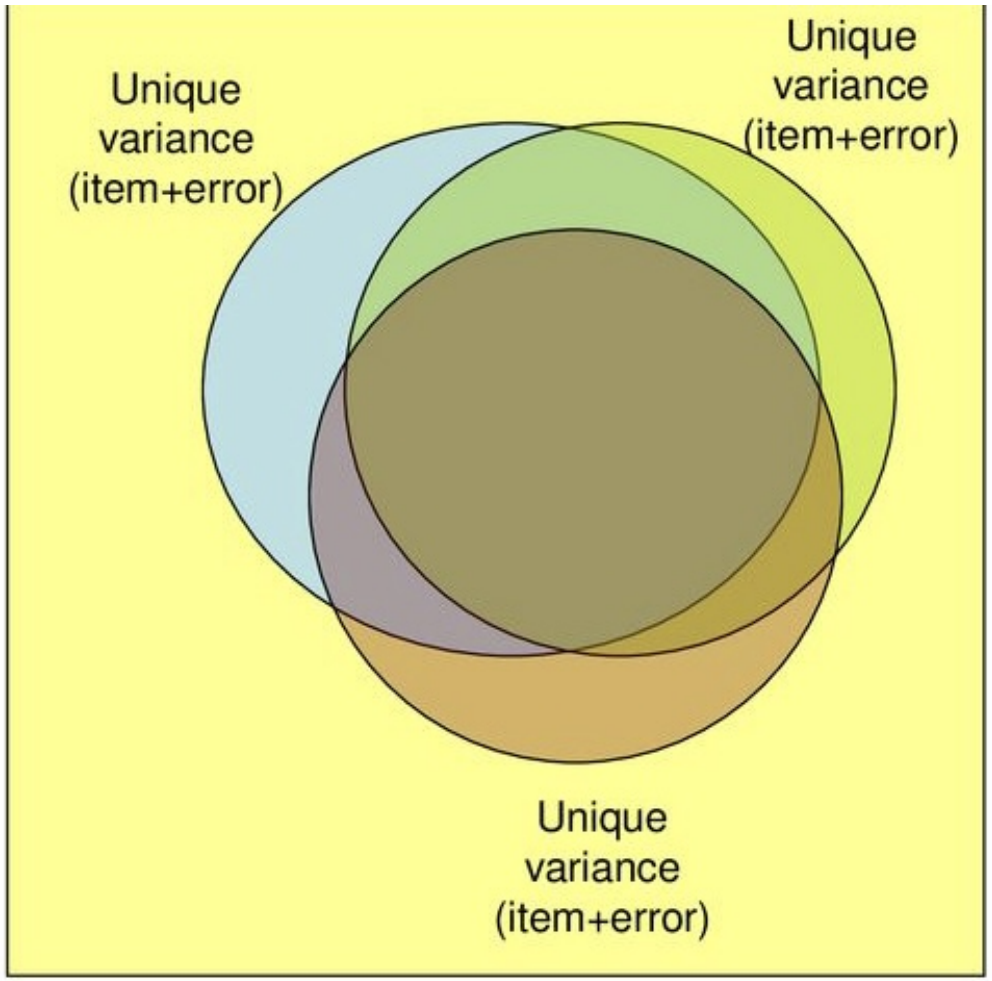


Outcome:

Item 3

The average R square is the average shared variance for that item with the other items

PCA


Unique
variance
(item+error)
Unique
variance
(item+error)
Unique
variance
(item+error)

PCA vs. FA

- Run *factor analysis* if you assume or wish to test a theoretical model of *latent factors* causing observed variables.
- Run *principal component analysis* If you want to simply *reduce* your correlated observed variables to a smaller set of important independent composite variables.

Eigenvalues and Eigenvectors

- **Eigenvalues** represent the total amount of variance that can be explained by a given factor
  - Sum of squared component loadings down all items for each factor
- **Eigenvectors** represent a weight for each eigenvalue
  - Eigenvector times the square root of the eigenvalue gives the **factor loadings**
    - Correlation between item and factor

Factor analysis steps

1. Checking the suitability of data
2. Decide # of factors
3. Extraction
4. Rotation
5. Interpret
6. *Optional*: Compute factor scores

Data

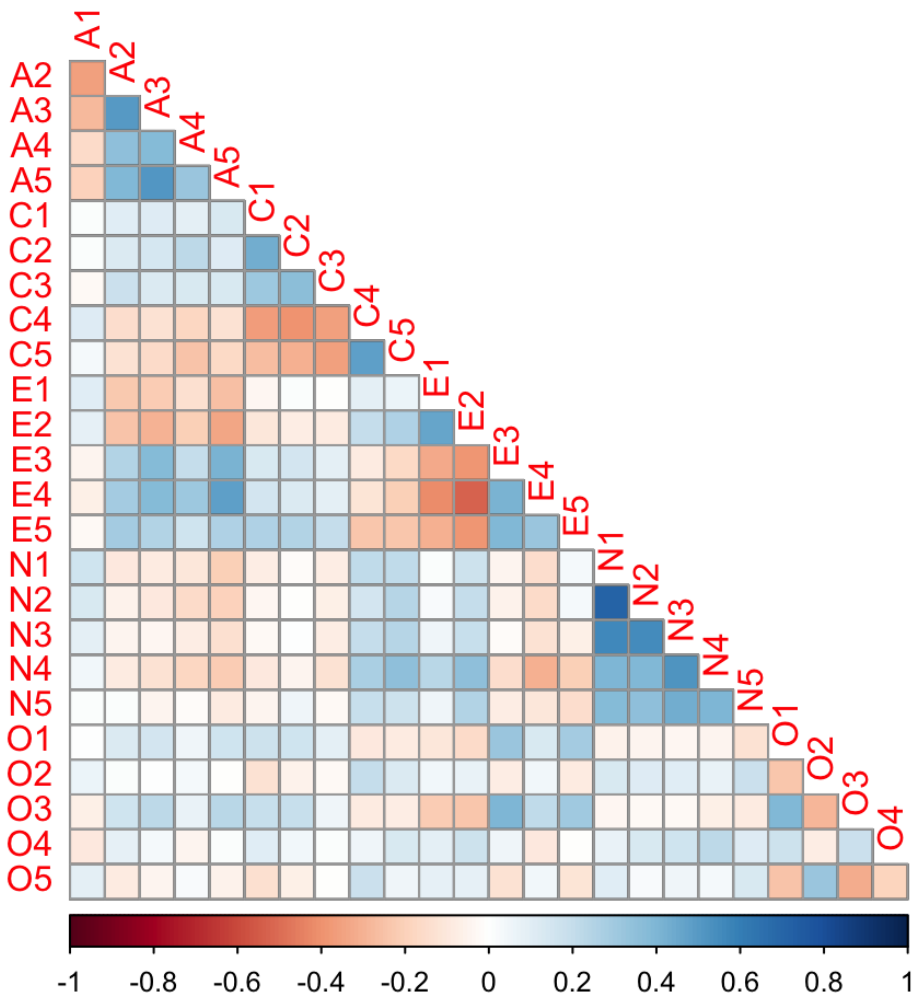
```
p_load(psych, tidyverse, corrplot,pa

# Load the data
data <- psych::bfi[, 1:25] # Select
data <- na.omit(data)
```

- 2800 participants
- 25 self-report items
  - The personality items are split into 5 categories.

Big 5

Big Five Personality Traits

Data visualization



Is factor analysis warranted?

- Bartlett's test
  - Correlation matrix significantly different from identity matrix (0s)?

1	0	0
0	1	0
0	0	1

Is factor analysis warranted?

- Kaiser-Meyer-Olkin (KMO)

$$KMO = \frac{\Sigma(r)^2}{\Sigma(r)^2 + \Sigma(r_p)^2}$$

- Two variables share a common factor they will have small partial correlation (most of variance is explained by common factor so not much left)

KMO Criterion	Adequacy Interpretation
0.00-0.49	Unacceptable
0.50-0.59	Poor
0.60-0.69	Fair
0.70-0.79	Good
0.80-0.89	Very Good
0.90-1.00	Excellent

Is factor analysis warranted?

```r
#easystats
performance::check_factorstructure(data)
```

```
# Is the data suitable for Factor Analysis?

  - KMO: The Kaiser, Meyer, Olkin (KMO) measure of sampling adequacy suggests that
  - Sphericity: Bartlett's test of sphericity suggests that there is sufficient sig
```

```r
#get MSA for each var
MSA <- check_kmo(data)
# delete items < .5
MSA$MSA_variable
```

```
       A1        A2        A3        A4        A5        C1        C2        C3
0.7540716 0.8364320 0.8702024 0.8780416 0.9035590 0.8433626 0.7958161 0.8519722
       C4        C5        E1        E2        E3        E4        E5        N1
0.8265898 0.8641133 0.8381302 0.8838897 0.8970459 0.8774011 0.8933998 0.7794802
       N2        N3        N4        N5        O1        O2        O3        O4
0.7803909 0.8623967 0.8852681 0.8602403 0.8586864 0.7803388 0.8444575 0.7701770
       O5
0.7615938
```

Assumptions

- No outliers
- Large sample
  - >100
- Normality
- No missingness
- no multicolinerarity

Assumptions: Outliers

```r
# check outliers (uses Mahal)
performance::check_outliers(data)
```

```
84 outliers detected: cases 31, 42, 149, 154, 170, 236, 285, 323, 357,
  371, 397, 398, 416, 486, 488, 499, 579, 658, 693, 699, 724, 726, 753,
  771, 773, 776, 840, 879, 880, 923, 992, 1002, 1012, 1056, 1078, 1111,
  1116, 1127, 1131, 1155, 1243, 1255, 1277, 1309, 1310, 1313, 1316, 1359,
  1363, 1364, 1368, 1369, 1370, 1371, 1435, 1500, 1537, 1541, 1558, 1677,
  1685, 1738, 1775, 1786, 1797, 1815, 1816, 1865, 1906, 1933, 1936, 2018,
  2186, 2194, 2257, 2259, 2263, 2272, 2314, 2315, 2346, 2393, 2398, 2413.
- Based on the following method and threshold: mahalanobis (52.62).
- For variables: 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.
```

```r
outliers_list<- check_outliers(data)
data <- data[!outliers_list, ] # remove outliers
```

Assumptions: Multicollinearity

- We do not want variables that are too highly correlated
- Determinant of correlation matrix
  - Smaller < .00001 (close to 0) suggests a problem with multicollinearity

```
1 cormatrix <- cor(data)
2 det(cormatrix)
```

```
[1] 0.0003920442
```

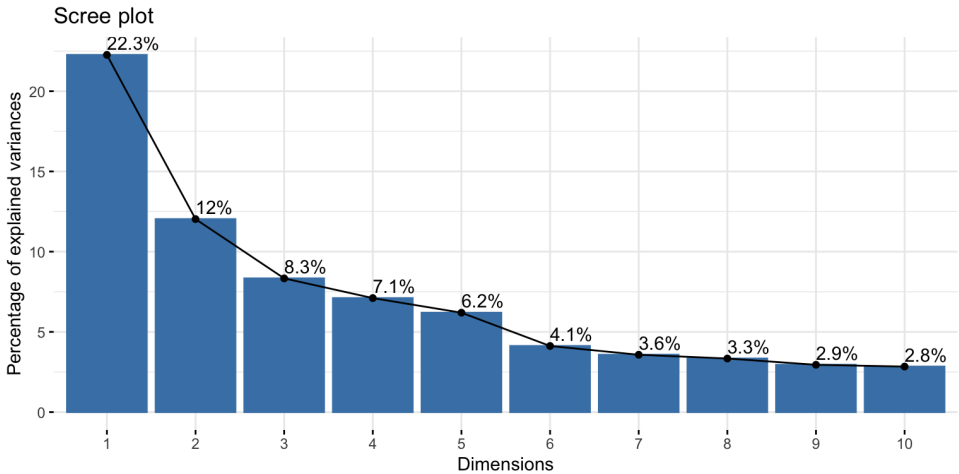
Fitting Factor Model: # of factors

Several different ways:

Fitting Factor Model: # of factors

Scree plot

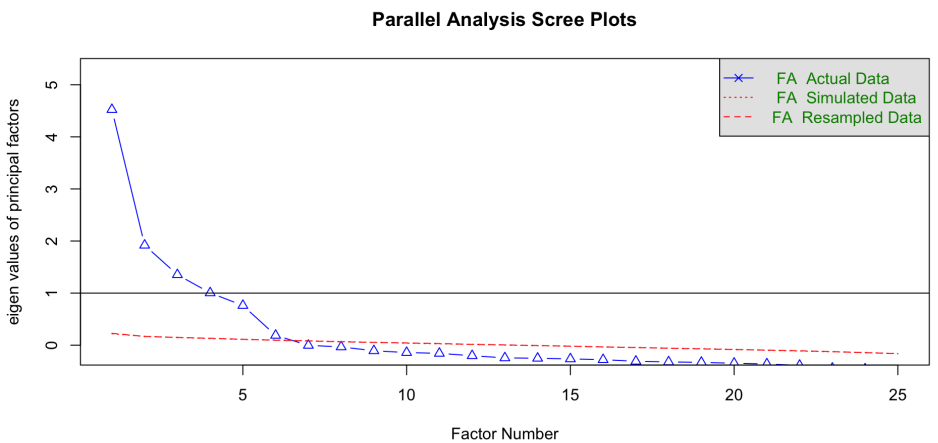
- A plot of the Eigenvalues in order from largest to smallest
- Look for the elbow (shared variability starting to level off)
    - Above the elbow is how many components you want


Scree plot
Percentage of explained variances
Dimensions
22.3%
12%
8.3%
7.1%
6.2%
4.1%
3.6%
3.3%
2.9%
2.8%
0
5
10
15
20
1
2
3
4
5
6
7
8
9
10

Fitting Factor Model: # of factors

- Parallel analysis
  - Run simulations pulling eigenvalues from randomly generated datasets
  - If eigenvalues > eigenvalues from random datasets more likely to represent meaningful patterns in the data
  - More objective and reliable

```
1  fa.parallel(data, fa="fa")
```

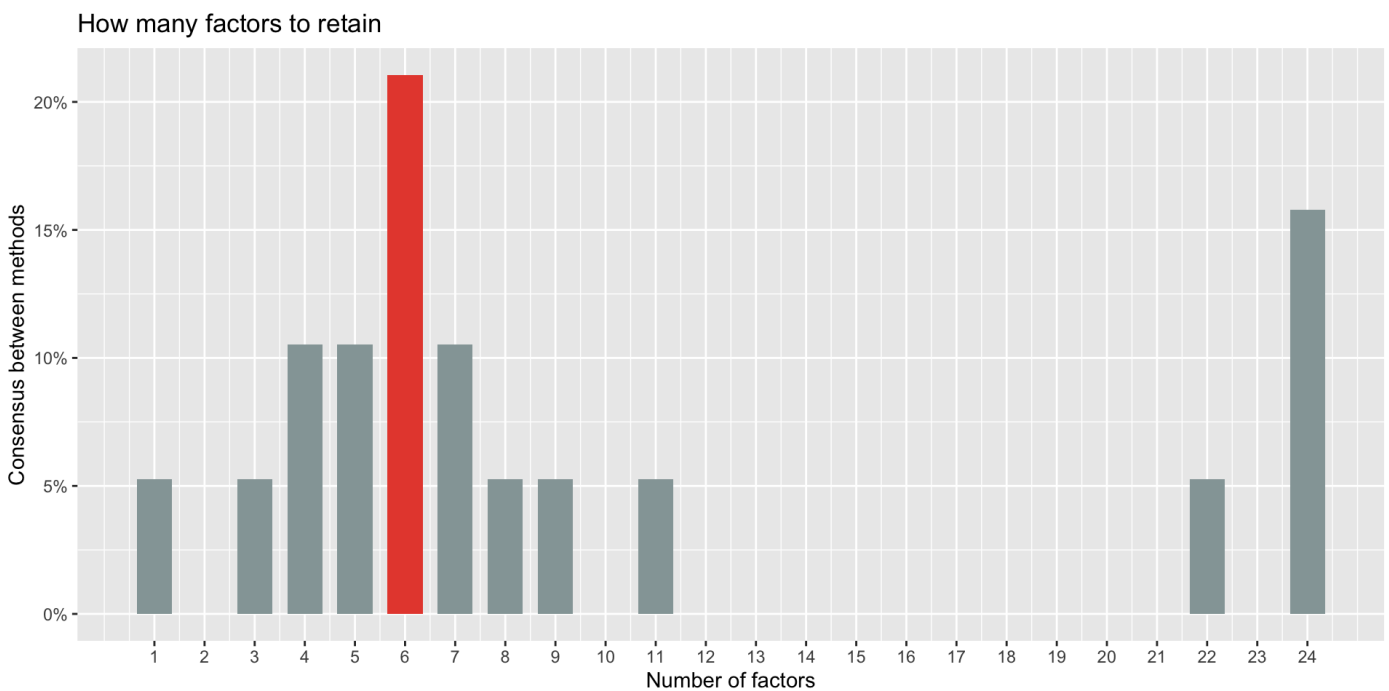



```
Parallel analysis suggests that the numb
```

Method agreement procedure

- Uses many methods to determine how many factor you should get
  - This is the approach I would use

```
#| fig.align: "center"
#|
library(parameters)
n_factors(data) %>% plot()
```



Extracting factor loadings

- Once the number of factors are decided the researcher runs another factor analysis to get the loading for each of the factors
  - Principal axis factoring (PAF)
    - Get initial estimates of communalities
      - Squared multiple correlations (highest absolute correlation)
    - Take correlation matrix and replace diagonal elements with communalities

```
library(parameters)
# nfactor number of factors from par
# rotate rotation method
# fm is principle axis
efa <- psych::fa(data, nfactors = 5,
   model_parameters(sort = TRUE)
#use pca instead
efa_pca <- psych::fa(data, nfactors
   model_parameters(sort = TRUE)
```

	Item 1	Item 2	Item 3	Item 4	Item 5	Item
Item 1	.76					
Item 2	.60	.56				
Item 3	.43	.76	.87			
Item 4	.34	.45	.64	.56		
Item 5	.33	.32	.34	.65	.52	
Item 6	.82	.81	.45	.57	.33	.41

Squared multiple correlations (R square) are on the diagonal of the correlation matrix

Factor loadings

- Correlation between item and factor
- Naming: PA1-PA2...
  - Reflects fitting method
- Factors ordered by variance explained

Variable	PA1	PA2	PA3	PA4	PA5	Complexity	Uniqu
E2	-0.638	-0.0434	-0.212	0.0484	0.322	1.76	
E4	0.606	0.159	0.352	0.0879	-0.195	2.07	
A5	0.588	0.161	0.292	0.0251	0.164	1.83	
A3	0.54	0.288	0.278	0.0678	0.3	2.82	
N4	-0.535	0.411	-0.058	-0.0539	0.211	2.28	
E3	0.535	0.311	0.122	-0.182	-0.112	2.12	

E5	0.524	0.293	-0.0975	-0.00785	-0.246	2.14
C5	-0.504	0.122	0.277	-0.292	0.0838	2.46
A2	0.481	0.275	0.205	0.0898	0.35	3.02
C4	-0.47	0.0689	0.453	-0.239	0.00192	2.53
A4	0.416	0.111	0.141	0.249	0.185	2.56
E1	-0.413	-0.178	-0.247	0.103	0.251	2.99
N1	-0.447	0.643	-0.0177	0.111	-0.247	2.2
N2	-0.436	0.629	-0.0673	0.064	-0.202	2.08
N3	-0.414	0.618	-0.00413	0.0656	0.00409	1.77
N5	-0.354	0.415	0.064	0.203	0.146	2.79
C2	0.343	0.22	-0.449	0.299	0.0544	3.26
C1	0.353	0.149	-0.436	0.152	0.0135	2.46
O3	0.405	0.303	-0.166	-0.462	-0.0109	3.02

O5	-0.212	-0.0623	0.304	0.409	-0.0993	2.62
O1	0.333	0.21	-0.214	-0.366	-0.0298	3.28
O2	-0.191	0.0556	0.335	0.355	-0.043	2.62
C3	0.329	0.0818	-0.331	0.331	0.0425	3.15
A1	-0.237	-0.00852	-0.132	0.0442	-0.399	1.91
O4	-0.0777	0.257	-0.173	-0.255	0.302	3.74

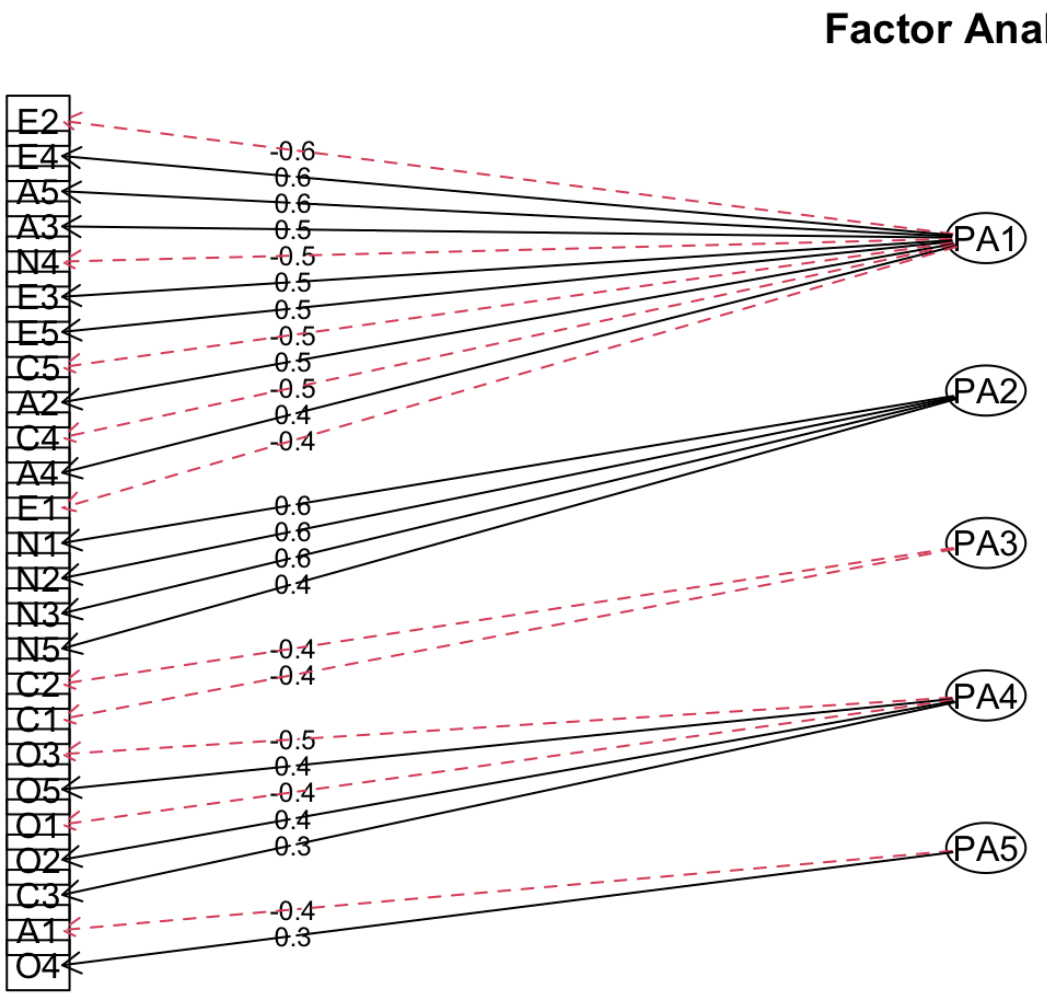
Variance accounted for

```
1 summary(efa)
```

Parameter	PA1	PA2	PA3	PA4	PA5
Eigenvalues	4.76	2.28	1.6	1.27	0.994
Variance	0.19	0.091	0.064	0.0506	0.0398
Variance_Cumulative	0.19	0.282	0.346	0.396	0.436
Variance_Proportion	0.437	0.209	0.147	0.116	0.0912

Path diagram

```
efa <- psych::fa(data, nfactors = 5, rotate="none", fm="pa")
fa.diagram(efa)
```



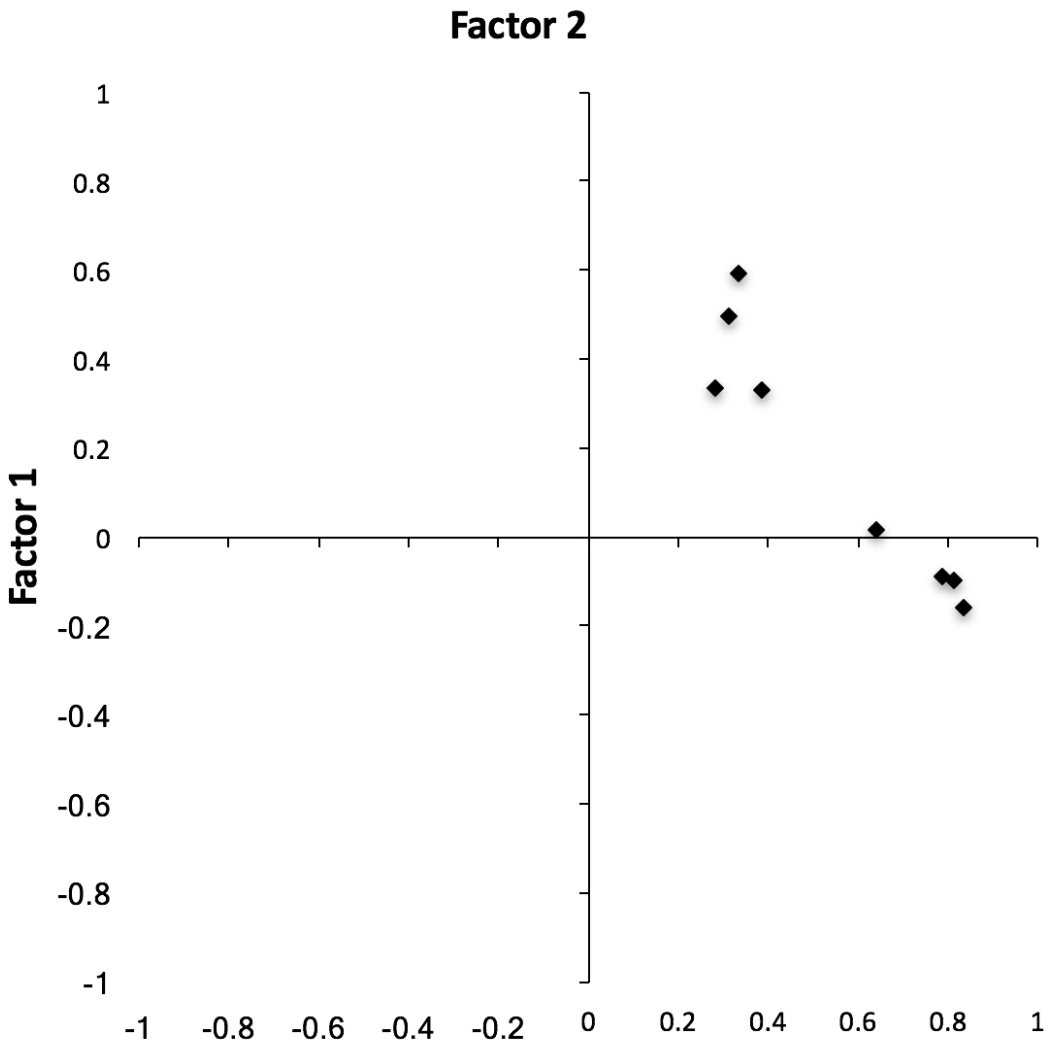
Rotation

- Simple structure
  - Made more interpretable (understandable) without actually changing the relationships among the variables
    - High factor loadings for each item on one factor
    - Low factor loadings for all other factors

Rotation

- Different types of rotation:
  - Orthogonal Rotation (Varimax)
    - This method of rotation prevents the factors from being correlated with each other
    - Useful if you have factors that should theoretically be unrelated
  - Oblique rotation (Direct Oblimin)
    - Allows factors to correlate (more common)
    - Good idea to always use this

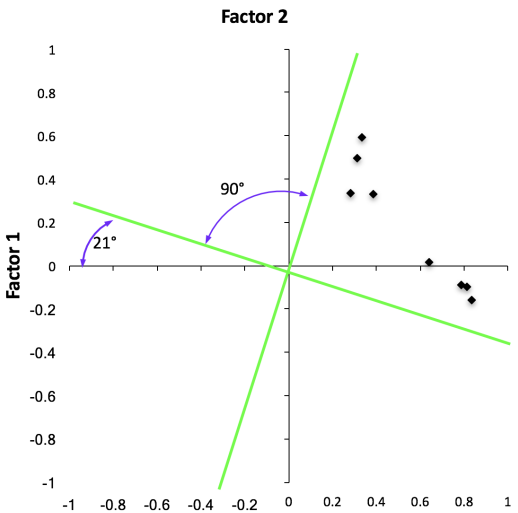
Rotation



	Factor 1	Factor 2
Y1	.834	-.160
Y2	.813	-.099
Y3	.788	-.088
Y4	.642	.015
Y5	.386	.329
Y6	.333	.593
Y7	.313	.497
Y8	.284	.336

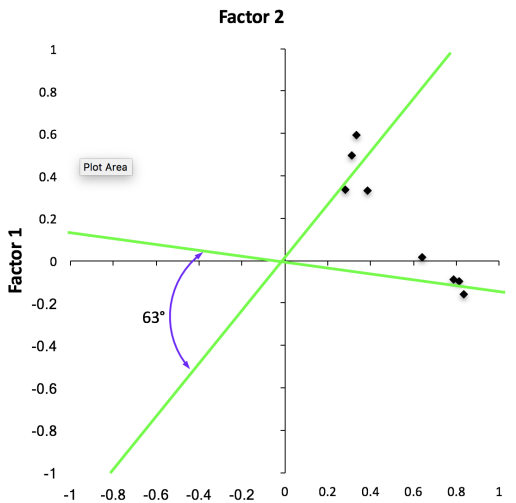
Rotation

- Orthogonal



	Factor 1	Factor 2
Y1	.836	.150
Y2	.794	.199
Y3	.767	.201
Y4	.594	.244
Y5	.242	.445
Y6	.098	.673
Y7	.114	.576
Y8	.145	.416

- Oblique



	Factor 1	Factor 2
Y1	.875	-.062
Y2	.817	.003
Y3	.788	.012
Y4	.588	.106
Y5	.154	.418
Y6	-.059	.704
Y7	-.018	.595
Y8	.055	.413

Rotation

```r
#change rotate arg to desired rotation
#orthogonal rotation
efa_or <- psych::fa(data, nfactors = 5, rotate="none", fm="varimax") %>%
   model_parameters(sort = TRUE, threshold = "max")

# correlated rotation
efa_obs <- psych::fa(data, nfactors = 5, rotate="oblimin", fm="pa") %>%
   model_parameters(sort = TRUE, threshold = "max")
```

Rotation

```
1 efa_obs
```

Variable	PA2	PA1	PA3	PA5	PA4	Complexity	Uniqueness
N1	0.833					1.07	0.313
N2	0.795					1.04	0.365
N3	0.714					1.09	0.442
N5	0.495					2.02	0.636
N4	0.485					2.26	0.494
E2		-0.666				1.11	0.441
E4		0.611				1.46	0.438
E1		-0.539				1.24	0.663
E5		0.442				2.52	0.57

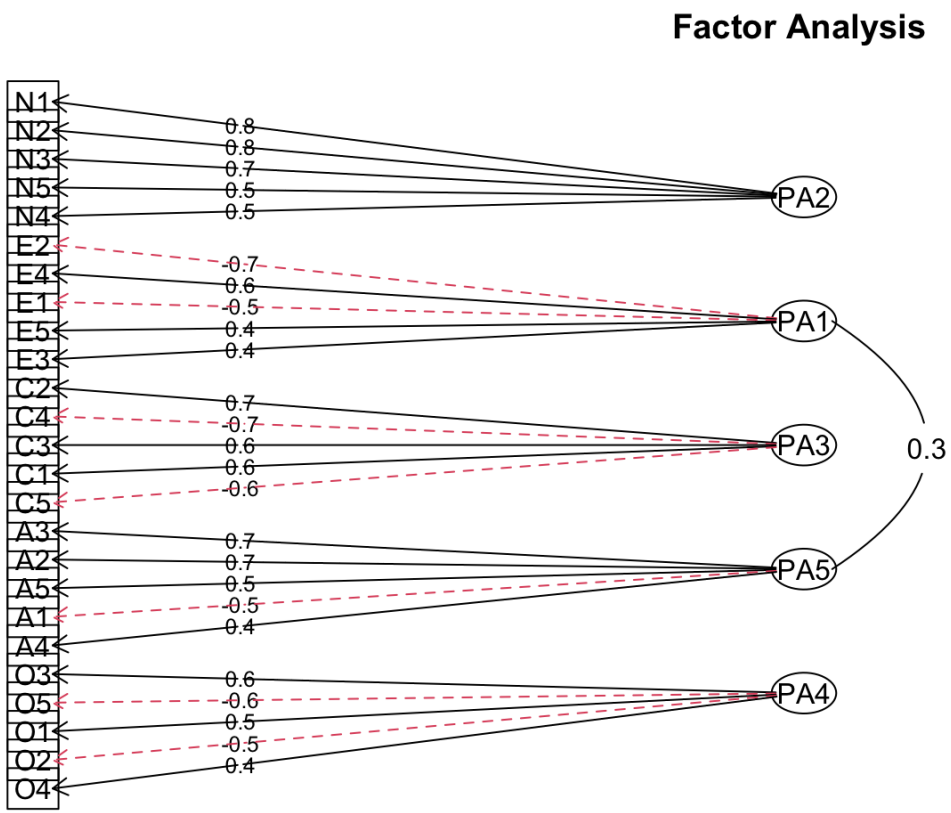
E3	0.431		2.45	0.556
C2		0.676	1.16	0.54
C4		-0.656	1.12	0.512
C3		0.586	1.09	0.665
C1		0.569	1.16	0.64
C5		-0.562	1.41	0.563
A3		0.673	1.08	0.454
A2		0.658	1.04	0.52
A5		0.534	1.56	0.516
A1		-0.46	1.89	0.765
A4		0.442	1.79	0.698
O3			0.638 1.17	0.503
O5			-0.554 1.21	0.682

O1	0.531	1.12	0.665
O2	-0.48	1.69	0.72
O4	0.368	2.74	0.742

Rotation

- After rotation

```
1  efa_obs <- psych::fa(data, nfactors = 5, rotate="oblimin", fm="pa")
2
3  fa.diagram(efa_obs)
```



What makes a good factor?

- Makes sense
- Loadings on the same factor do not appear to measure completely different things
- Easy to interpret
- Simple structure
  - Contains either high or low loadings with few moderately sized loadings
  - Lacks cross-loadings
    - You don’t have items that load equally onto more than 1 factor
      - Keep items > .3 and delete items < .3
- 3 or more indicators per latent factor

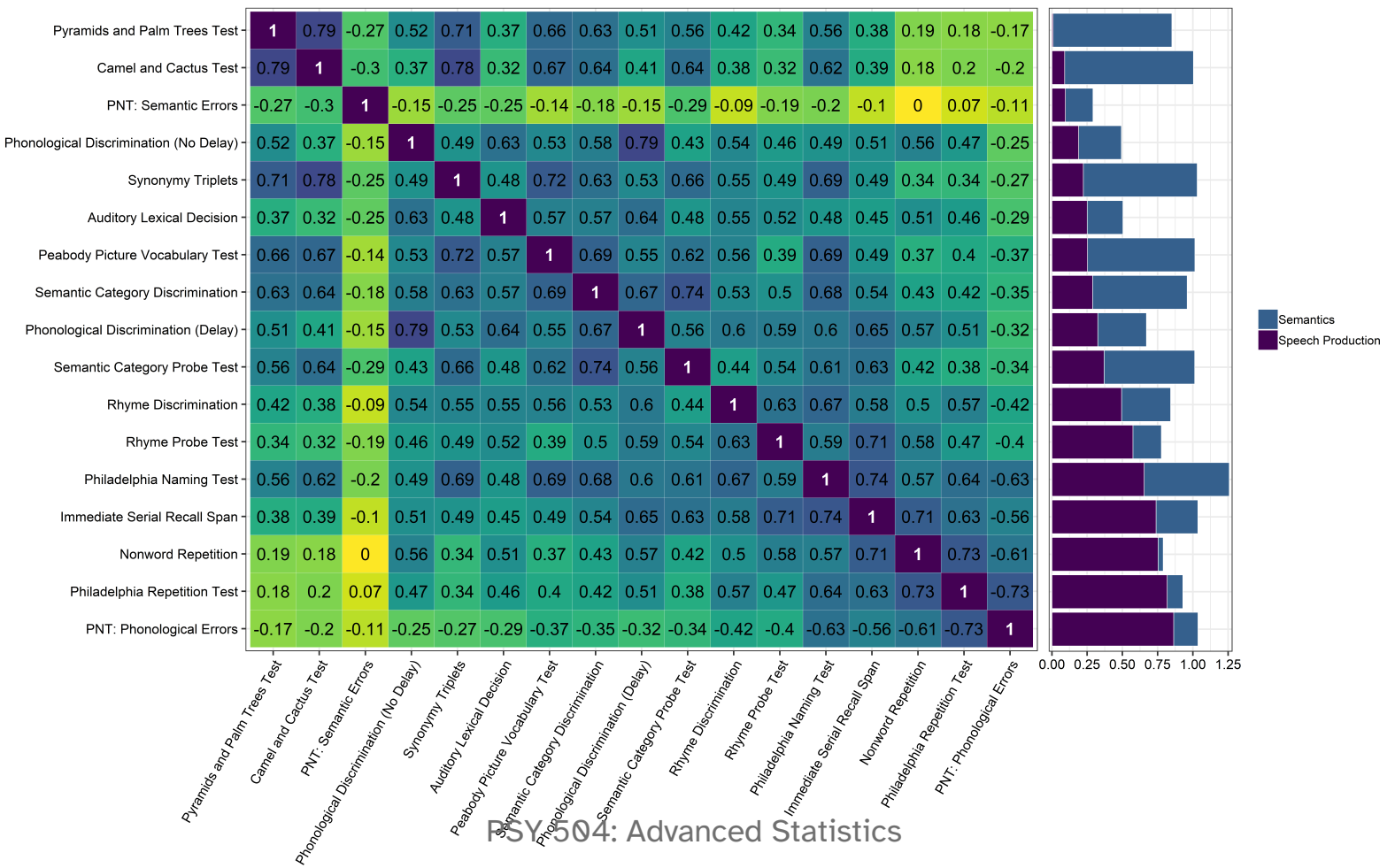
Factor scores

- Estimated scores for each participant on each underlying factor (standing on factor)
  - Standardize the factor loadings by dividing each loading by the square root of the sum of squares of the factor loading for that factor.
  - Multiply scores on each item by the corresponding standardized factor loading and then summing across all items.
- Can use them in multiple regression!

```
1 efa_obs <- psych::fa(data, nfactors = 5, rotate="oblimin", fm="pa", scores="reg
```

Factor scores

Geller, J., Thye, M., & Mirman, D. (2019). Estimating effects of graded white matter damage and binary tract disconnection on post-stroke language impairment. *NeuroImage*, *189*. https://doi.org/10.1016/j.neuroimage.2019.01.020



Plotting FA

```r
# correlated rotation
efa_obs <- psych::fa(data, nfactors
   model_parameters(sort = TRUE, thr

efa_plot <- as.data.frame(efa_obs) %
  pivot_longer(PA2:PA4) %>%
  dplyr::select(-Complexity, -Unique


#For each test, plot the loading as
# note that the length will be the a
# fill color will be the signed valu
efa_fact_plot <- ggplot(efa_plot, ae
  facet_wrap(~ Personality, nrow=1)
  geom_bar(stat="identity") + #make
  coord_flip() + #flip the axes so t
  #define the fill color gradient: b
  scale_fill_gradient2(name = "Loadi
                       high = "blue"
```

```r
efa_fact_plot
```

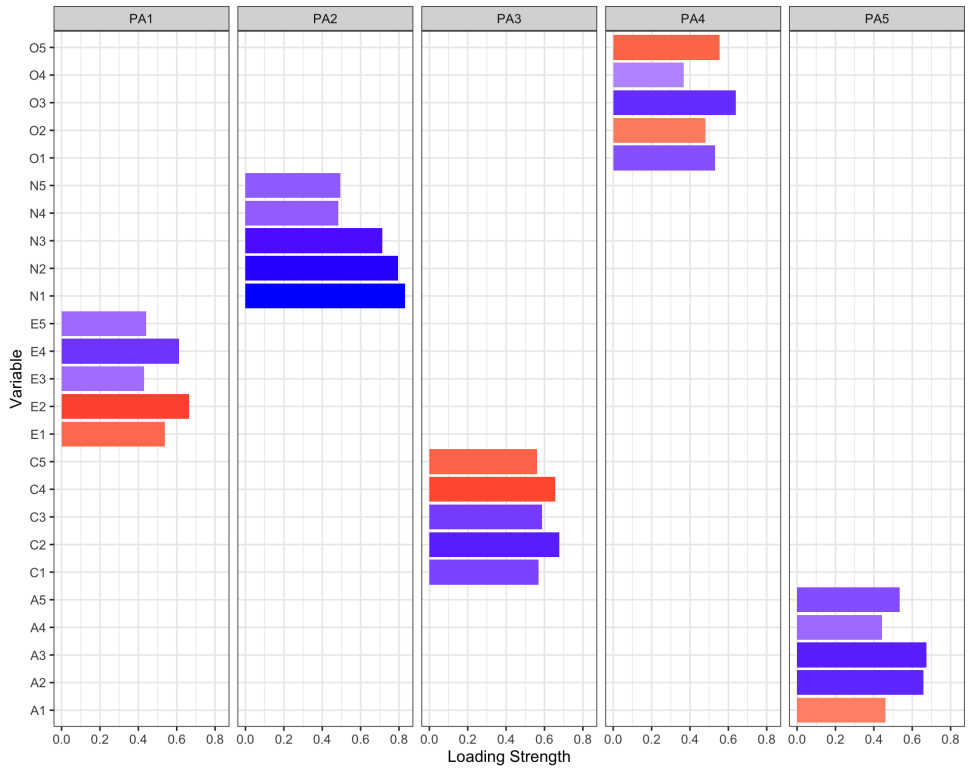


Table FA

```r
source("https://raw.githubusercontent.com/franciscowilhelm/r-collection/master/

efa_obs <- psych::fa(data, nfactors = 5, rotate="oblimin", fm="pa")

table<- fa_table(efa_obs)
```

FA table

```
1 table$ind_table
```

Factor analysis results

	Factor_1	Factor_2	Factor_3	Factor_4	Factor_5	Communality	Uni
N1	0.833	0.100	0.002	-0.109	-0.041	0.69	0.3
N2	0.795	0.041	0.012	-0.106	0.022	0.64	0.3
N3	0.714	-0.108	-0.034	0.094	0.013	0.56	0.4
N5	0.495	-0.209	-0.001	0.213	-0.166	0.36	0.6
N4	0.485	-0.394	-0.135	0.099	0.080	0.51	0.4
E2	0.110	-0.666	-0.033	-0.066	-0.076	0.56	0.4
E4	-0.009	0.611	0.010	0.290	-0.071	0.56	0.4
E1	-0.055	-0.539	0.094	-0.101	-0.105	0.34	0.6
E5	0.157	0.442	0.282	0.039	0.206	0.43	0.5

Factor analysis results

E3	0.080	0.431	0.004	0.230	0.292	0.44	0.5
C2	0.150	-0.080	0.676	0.080	0.036	0.46	0.5
C4	0.150	0.011	-0.656	0.038	-0.041	0.49	0.5
C3	0.036	-0.042	0.586	0.078	-0.077	0.34	0.6
C1	0.062	-0.039	0.569	0.005	0.144	0.36	0.6
C5	0.182	-0.147	-0.562	0.019	0.088	0.44	0.5
A3	-0.018	0.122	0.033	0.673	0.031	0.55	0.4
A2	-0.015	0.017	0.081	0.658	0.029	0.48	0.5
A5	-0.119	0.256	0.002	0.534	0.032	0.48	0.5
A1	0.210	0.195	0.058	-0.460	-0.066	0.23	0.7
A4	-0.053	0.084	0.198	0.442	-0.155	0.30	0.7
O3	0.041	0.163	-0.001	0.079	0.638	0.50	0.5
O5	0.122	0.113	-0.048	0.041	-0.554	0.32	0.6

Factor analysis results

O1	*0.012*	*0.113*	*0.065*	*0.000*	0.531	0.34	0.6
O2	*0.190*	*0.102*	*-0.090*	*0.141*	-0.480	0.28	0.7
O4	*0.122*	*-0.342*	*-0.037*	*0.183*	0.368	0.26	0.7

Confirmatory factor analysis

- Do not do a confirmatory analysis with the same data you performed your exploratory analysis!
  - Machine learning approach
- Partition data training and test data

```
# to have reproducible result, we will also set seed here so that similar
# portions of the data are used each time we run the following code
partitions <- datawizard::data_partition(data, training_proportion = 0.7, seed
training <- partitions$p_0.7
test <- partitions$test
```

CFA in Lavaan

Let's compare the big6 to the big5

```r
structure_big5 <- psych::fa(training, nfactors = 5, rotate = "oblimin") %>%
  efa_to_cfa()

# Investigate how the models look
structure_big5
```

```
# Latent variables
MR2 =~ N1 + N2 + N3 + N4 + N5
MR1 =~ E1 + E2 + E3 + E4 + E5
MR3 =~ C1 + C2 + C3 + C4 + C5
MR5 =~ A1 + A2 + A3 + A4 + A5 + .row_id
MR4 =~ O1 + O2 + O3 + O4 + O5
```

CFA in Lavaan

```r
structure_big6 <- psych::fa(training, nfactors = 6, rotate = "oblimin") %>%
  efa_to_cfa()

structure_big6
```

```
# Latent variables
MR2 =~ N1 + N2 + N3 + N5
MR1 =~ E1 + E2 + E3 + E4 + E5 + N4
MR3 =~ C1 + C2 + C3 + C4 + C5
MR5 =~ A1 + A2 + A3 + A4 + A5 + .row_id
MR4 =~ O1 + O2 + O3 + O4 + O5
```

Fit and compare models

```
big5 <- suppressWarnings(lavaan::cfa(structure_big5, data = test))
big6 <- suppressWarnings(lavaan::cfa(structure_big6, data = test))

performance::compare_performance(big5, big6, verbose = FALSE)
```

Name	Model	Chi2	Chi2_df	p_Chi2	Baseline	Baseline_df	p_Baseline	
big5	lavaan	1.42e+03	289	0	5.57e+03	325	0	0
big6	lavaan	1.56e+03	289	0	5.57e+03	325	0	0

Sample Write-up

Table 1. Information to Include in an EFA Report.

- Justification of the measured variables included in the EFA
- Justification of type and number of participants included in the EFA
- Data characteristics (including descriptive statistics, normality, missing data, etc.)
- Appropriateness of data for EFA (Bartlett and KMO statistics)
- Computer program and version
- Correlation matrix analyzed (Pearson, polychoric, etc.)
- Factor model (principal components analysis vs. common factor analysis)
- Estimation method (iterated principal axis, maximum likelihood, etc.)
- Method of estimating communalities
- How number of factors to retain was determined
- Factor rotation method
- Strategy for interpreting factors (including how salience was defined)
- Percentage of variance accounted for by factors (specify before or after rotation)
- Complete pattern coefficient matrix (do not omit low coefficients)
- Interfactor correlations (for oblique rotations)
- Reliability estimates for the identified factors
- Complete structure coefficient matrix (when substantially different from pattern matrix)
- Eigenvalues for all factors if space permits
- Correlation matrix if space permits

Note: EFA = exploratory factor analysis; KMO = Kaiser-Meyer-Olkin.

Write-up

The dimensionality of the 25 items from the Gendered Racial Microagressions Scale for Black Women was analyzed using principal axis factoring. First, data were screened to determine the suitability of the data for this analyses. The Kaiser-Meyer- Olkin measure of sampling adequacy (KMO; Kaiser, 1970) represents the ratio of the squared correlation between variables to the squared partial correlation between variables. KMO ranges from 0.00 to 1.00 – values closer to 1.00 indicate that the patterns of correlations are relatively compact and that component analysis should yield distinct and reliable components (Field, 2012). In our dataset, the KMO value was .86, indicating acceptable sampling adequacy. The Barlett's Test of Sphericity examines whether the population correlation matrix resembles an identity matrix (Field, 2012). When the p value for the Bartlett's test is < .05, we are fairly certain we have clusters of correlated variables. In our dataset, $\chi 1(300)=1683.76, p<.001$, indicating the correlations between items are sufficiently large enough for principal components analysis. The determinant of the correlation matrix alerts us to any issues of multicollinearity or singularity and should be larger than

Write-up

- Number of components
  - Scree plot
  - Eigenvalues > 1
  - Parallel analysis
  - Agreement method

Write-up

Several criteria were used to determine the number of components to extract: a priori theory, the scree test, the eigenvalue-greater-than-one criteria, and the interpretability of the solution. Kaiser's eigenvalue-greater-than-one criteria suggested four components, and, in combination explained 49% of the variance. The inflexion (elbow) in the scree plot justified retaining four components. Based on the convergence of these decisions, four components were extracted. We investigated each with orthogonal (varimax) and oblique (oblimin) procedures. Given the non-significant correlations (ranging from -0.03 to 0.03) and the clear component loadings in the orthogonal rotation, we determined that an orthogonal solution was most appropriate.

Write-up

- Rotation and factor loadings

  The rotated solution, as shown in Table 1 and Figure 1, yielded four interpretable components, each listed with the proportion of variance accounted for