Introduction to EFA

FACTOR ANALYSIS IN R

Jennifer Brussow
Psychometrician

Psycho + metrics

psycho = "of the mind"

metrics = "related to measurement"

Learning objectives

- Run a unidimensional exploratory factor analysis (EFA)
- View and interpret items' factor loadings
- Interpret individuals' factor scores

Factor Analysis' relationship to other analyses

1. Classical Test Theory: Scores are the unweighted sum of item scores.
2. **Factor Analysis**: Scores are an empirically weighted sum of item scores, where weights are determined by the items' correlations to each other.
3. Structural Equation Modeling: Extends factor analyses to allow the relationships between latent variables to be modeled.

Types of Factor Analysis

Exploratory Factor Analysis (EFA):

- Used during development
- Explore factor structure
- Evaluate items

Confirmatory Factor Analysis (CFA):

- Validate a measure
- Used after development

Package

Package: The `psych` package

- Developed by William Revelle.
- More info at **The Personality Project**.

```
library(psych)
```

Dataset

The `gcbs` dataset: Generic Conspiracist Beliefs Survey

- Take the assessment at Open Source Psychometrics Project
- Full test is 75 items measuring five conspiracist facets

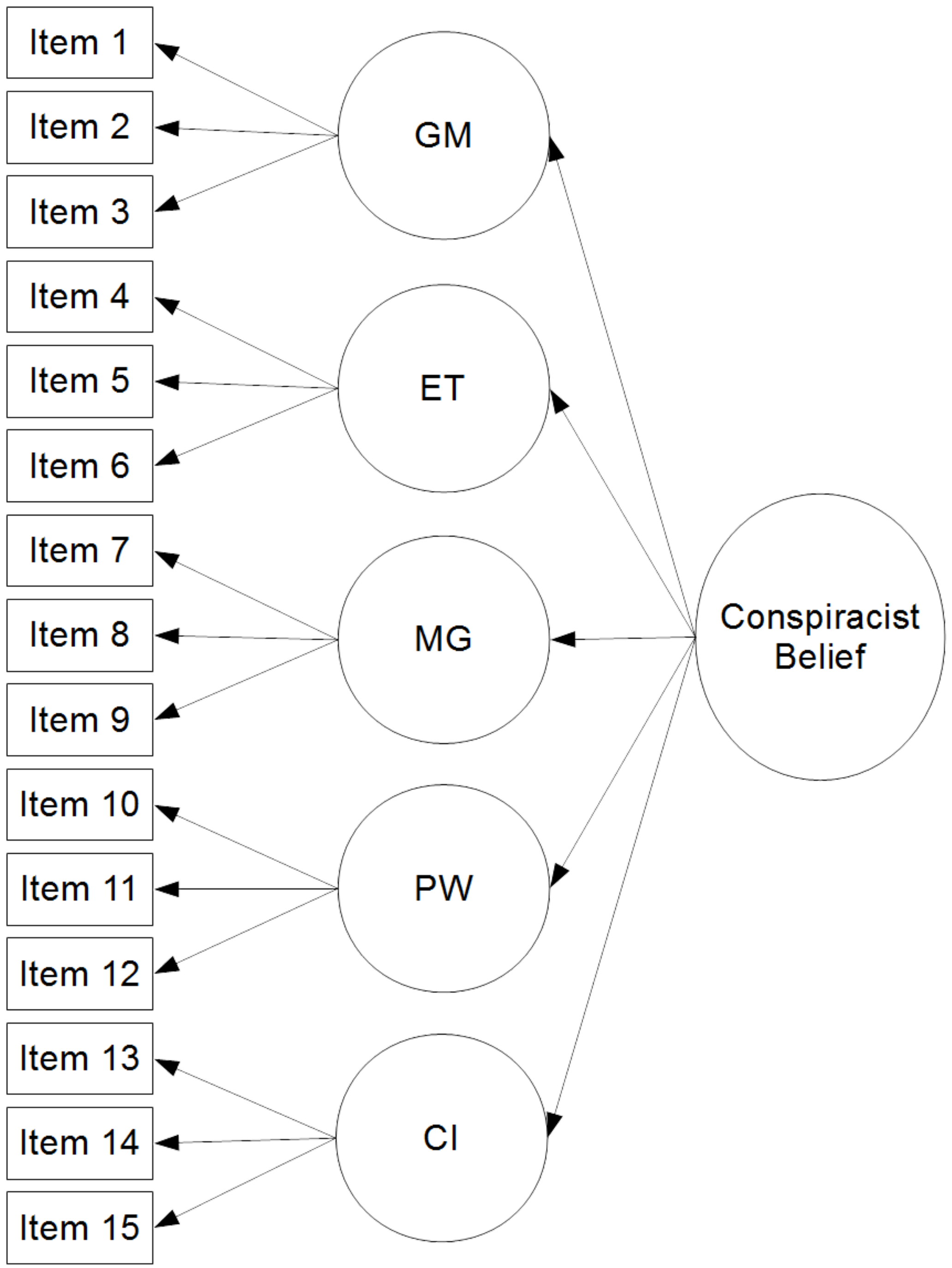
```
str(gcbs)
```

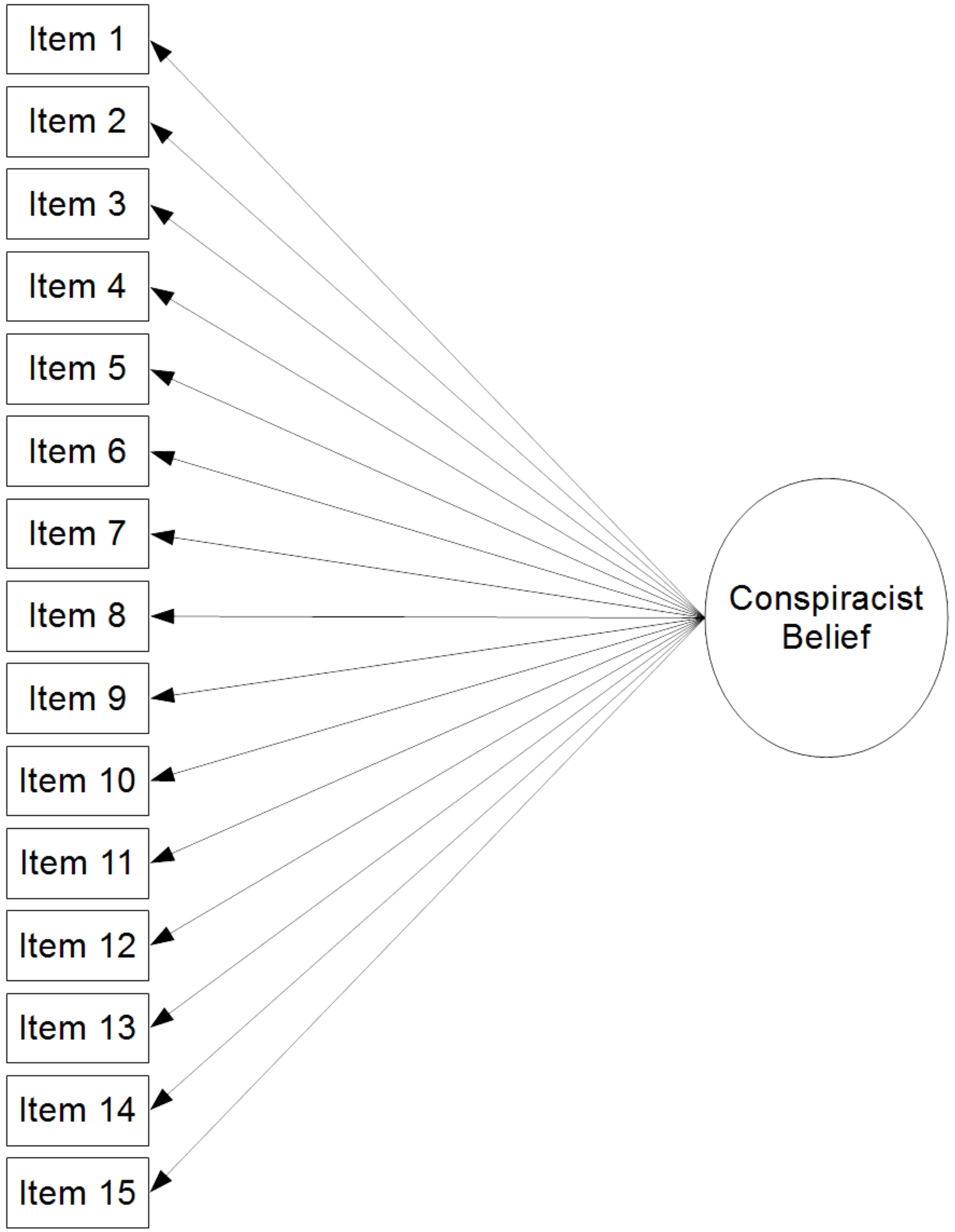
```
'data.frame':    2495 obs. of  15 variables:
 $ Q1 : int  5 5 2 5 5 1 4 5 1 1 ...
 $ Q2 : int  5 5 4 4 4 1 3 4 1 2 ...
 $ Q3 : int  3 5 1 1 1 1 3 3 1 1 ...
 $ Q4 : int  5 5 2 2 4 1 3 3 1 1 ...
 $ Q5 : int  5 5 2 4 4 1 4 4 1 1 ...
 ...
```

Item types

- Government malfeasance (GM)
- Extraterrestrial coverup (ET)
- Malevolent global conspiracies (MG)
- Personal wellbeing (PW)
- Control of information (CI)

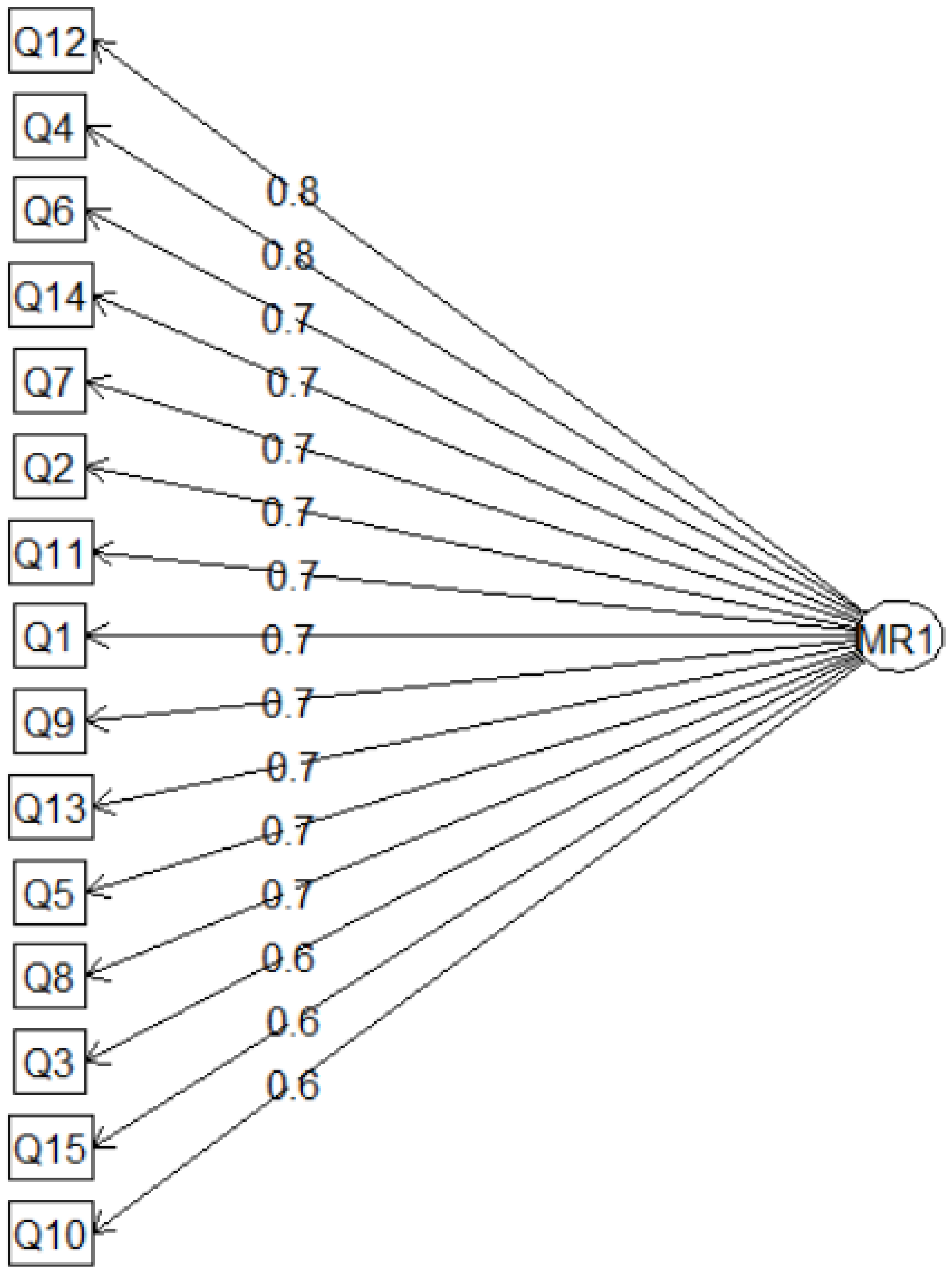
More information in **Brotherton, French, & Pickering (2013)**


Item 1
Item 2
Item 3
Item 4
Item 5
Item 6
Item 7
Item 8
Item 9
Item 10
Item 11
Item 12
Item 13
Item 14
Item 15
GM
ET
MG
PW
CI
Conspiracist Belief


Item 1
Item 2
Item 3
Item 4
Item 5
Item 6
Item 7
Item 8
Item 9
Item 10
Item 11
Item 12
Item 13
Item 14
Item 15
Conspiracist Belief

```
EFA_model <- fa(gcbs)
fa.diagram(EFA_model)
EFA_model$loadings
```

```
Loadings:
    MR1
Q1  0.703
Q2  0.719
Q3  0.638
Q4  0.770
Q5  0.672
Q6  0.746
Q7  0.734
Q8  0.654
Q9  0.695
Q10 0.565
...
```



Let's practice!

FACTOR ANALYSIS IN R

Overview of the measure development process

FACTOR ANALYSIS IN R

Jennifer Brussow
Psychometrician

Development process

1. *Develop items for your measure*
2. *Collect pilot data from a representative sample*
3. Check out what that dataset looks like
4. *Consider whether you want to use EFA, CFA, or both*
5. If both, split your sample into random halves
6. Compare the two samples to make sure they are similar

Development process

1. Develop items for your measure
2. Collect pilot data from a representative sample
3. **Check out what that dataset looks like**

Inspecting your dataset

```r
library(psych)
describe(gcbs)
```

```
    vars    n mean   sd median trimmed  mad min max range  skew ...
Q1     1 2495 3.47 1.46      4    3.59 1.48   0   5     5 -0.55 ...
Q2     2 2495 2.96 1.49      3    2.96 1.48   0   5     5 -0.01 ...
Q3     3 2495 2.05 1.39      1    1.82 0.00   0   5     5  0.98 ...
Q4     4 2495 2.64 1.45      2    2.55 1.48   0   5     5  0.26 ...
Q5     5 2495 3.25 1.47      4    3.32 1.48   0   5     5 -0.35 ...
...
Q11   11 2495 3.27 1.40      4    3.34 1.48   0   5     5 -0.35 ...
Q12   12 2495 2.64 1.50      2    2.56 1.48   0   5     5  0.29 ...
Q13   13 2495 2.10 1.38      1    1.89 0.00   0   5     5  0.89 ...
Q14   14 2495 2.96 1.49      3    2.95 1.48   0   5     5 -0.02 ...
Q15   15 2495 4.23 1.10      5    4.47 0.00   0   5     5 -1.56 ...
```

Development process

1. Develop items for your measure
2. Collect pilot data from a representative sample
3. Check out what that dataset looks like
4. Consider whether you want to use an exploratory analysis (EFA), a confirmatory analysis (CFA), or both
5. **If both, split your sample into random halves**

Splitting the dataset

```
N <- nrow(gcbs)
indices <- seq(1, N)
indices_EFA <- sample(indices, floor((0.5 * N)))
indices_CFA <- indices[!(indices %in% indices_EFA)]
```

```
gcbs_EFA <- gcbs[indices_EFA, ]
gcbs_CFA <- gcbs[indices_CFA, ]
```

Development process

1. Develop items for your measure
2. Collect pilot data from a representative sample
3. Check out what that dataset looks like
4. Consider whether you want to use EFA, CFA, or both
5. If both, split your sample into random halves
6. **Compare the two samples to make sure they are similar**

Inspecting the halves

```
group_var <- vector("numeric", nrow(gcbs))
group_var[indices_EFA] <- 1
group_var[indices_CFA] <- 2
group_var
```

```
  [1] 2 1 2 2 1 2 1 1 2 2 2 1 2 2 1 1 2 1 1 1 1 2 1 1 2 1 1 1 2 2
 [31] 2 2 2 1 2 2 2 1 2 2 2 1 1 1 2 2 2 2 1 2 2 1 1 2 2 2 2 2 2 2
 [61] 2 1 2 1 2 2 1 2 1 2 2 2 1 2 1 2 1 1 2 2 1 2 1 2 1 1 1 2 2 2
 [91] 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 1 2 2 2 2 1 1 1 2 2 1 1 2 2
[121] 2 1 2 2 1 2 2 1 2 2 2 2 1 2 1 1 1 2 2 1 1 1 2 1 1 1 1 2 2 2
[151] 1 1 1 1 2 2 2 2 2 1 2 1 1 2 1 1 2 1 2 1 2 1 1 1 2 1 1 1 1 2
[181] 2 1 1 2 2 2 1 1 1 1 2 2 2 2 2 1 1 1 1 2 2 1 1 1 2 1 2 1 2 2
```

Inspecting the halves

```
gcbs_grouped <- cbind(gcbs, group_var)
```

```
describeBy(gcbs_grouped, group = group_var)
statsBy(gcbs_grouped, group = "group_var")
```

Let's practice!

FACTOR ANALYSIS IN R

Measure features: correlations and reliability

FACTOR ANALYSIS IN R

Jennifer Brussow
Psychometrician

Correlations

```
lowerCor(gcbs)
```

```
lowerCor(gcbs)
    Q1   Q2   Q3   Q4   Q5   Q6   Q7   Q8   Q9   Q10  ...
Q1  1.00
Q2  0.53 1.00
Q3  0.36 0.40 1.00
Q4  0.52 0.53 0.50 1.00
Q5  0.48 0.46 0.40 0.57 1.00
Q6  0.63 0.55 0.40 0.61 0.50 1.00
Q7  0.47 0.67 0.42 0.57 0.45 0.54 1.00
Q8  0.39 0.38 0.78 0.49 0.41 0.41 0.41 1.00
Q9  0.42 0.49 0.49 0.56 0.46 0.48 0.53 0.48 1.00
Q10 0.44 0.38 0.32 0.40 0.43 0.41 0.39 0.36 0.37 1.00
...
```

Testing correlations' significance: p-values

```
corr.test(gcbs, use = "pairwise.complete.obs")$p
```

```
    Q1 Q2 Q3 Q4 Q5 Q6 Q7 Q8 Q9 Q10 Q11 Q12 Q13 Q14 Q15
Q1   0  0  0  0  0  0  0  0  0   0   0   0   0   0   0
Q2   0  0  0  0  0  0  0  0  0   0   0   0   0   0   0
Q3   0  0  0  0  0  0  0  0  0   0   0   0   0   0   0
Q4   0  0  0  0  0  0  0  0  0   0   0   0   0   0   0
Q5   0  0  0  0  0  0  0  0  0   0   0   0   0   0   0
...
Q11  0  0  0  0  0  0  0  0  0   0   0   0   0   0   0
Q12  0  0  0  0  0  0  0  0  0   0   0   0   0   0   0
Q13  0  0  0  0  0  0  0  0  0   0   0   0   0   0   0
Q14  0  0  0  0  0  0  0  0  0   0   0   0   0   0   0
Q15  0  0  0  0  0  0  0  0  0   0   0   0   0   0   0
```

Testing correlations' significance: confidence intervals

```r
corr.test(gcbs, use = "pairwise.complete.obs")$ci
```

```
            lower         r     upper p
Q1-Q2   0.4970162 0.5259992 0.5538098 0
Q1-Q3   0.3206223 0.3553928 0.3892067 0
Q1-Q4   0.4953852 0.5244323 0.5523079 0
Q1-Q5   0.4503342 0.4810747 0.5106759 0
...
Q1-Q11  0.6199265 0.6435136 0.6659388 0
Q1-Q12  0.4932727 0.5224025 0.5503620 0
Q1-Q13  0.3464313 0.3805006 0.4135673 0
Q1-Q14  0.5059498 0.5345780 0.5620298 0
Q1-Q15  0.4753633 0.5051815 0.5338405 0
...
```

Coefficient alpha

```
alpha(gcbs)
```

```
Reliability analysis
Call: alpha(x = gcbs)

  raw_alpha std.alpha G6(smc) average_r S/N   ase mean sd
      0.93      0.93    0.94      0.48  14 0.002  2.9  1

 lower alpha upper     95% confidence boundaries
0.93 0.93 0.94
```

Coefficient alpha

```
alpha(gcbs)
```

```
 Reliability if an item is dropped:
    raw_alpha std.alpha G6(smc) average_r S/N alpha se
Q1       0.93      0.93    0.94      0.48  13   0.0021
Q2       0.93      0.93    0.94      0.48  13   0.0021
Q3       0.93      0.93    0.94      0.49  13   0.0020
Q4       0.93      0.93    0.94      0.47  13   0.0022
Q5       0.93      0.93    0.94      0.48  13   0.0021
...
Q11      0.93      0.93    0.94      0.48  13   0.0021
Q12      0.93      0.93    0.94      0.47  13   0.0022
Q13      0.93      0.93    0.94      0.48  13   0.0021
Q14      0.93      0.93    0.94      0.48  13   0.0021
Q15      0.93      0.93    0.94      0.49  14   0.0020
```

Split-Half reliability

```
splitHalf(gcbs)
```

```
Split half reliabilities
Call: splitHalf(r = gcbs)

Maximum split half reliability (lambda 4) =  0.95
Guttman lambda 6                          =  0.94
Average split half reliability            =  0.93
Guttman lambda 3 (alpha)                  =  0.93
Minimum split half reliability  (beta)    =  0.86
```

Let's practice!

FACTOR ANALYSIS IN R