POLSCI 9592

Lecture 11: Measurement

Dave Armstrong

Goals for This Session

1. What are measurement models about?
2. Summated Ratings Scales
3. Exploratory Factor Analysis

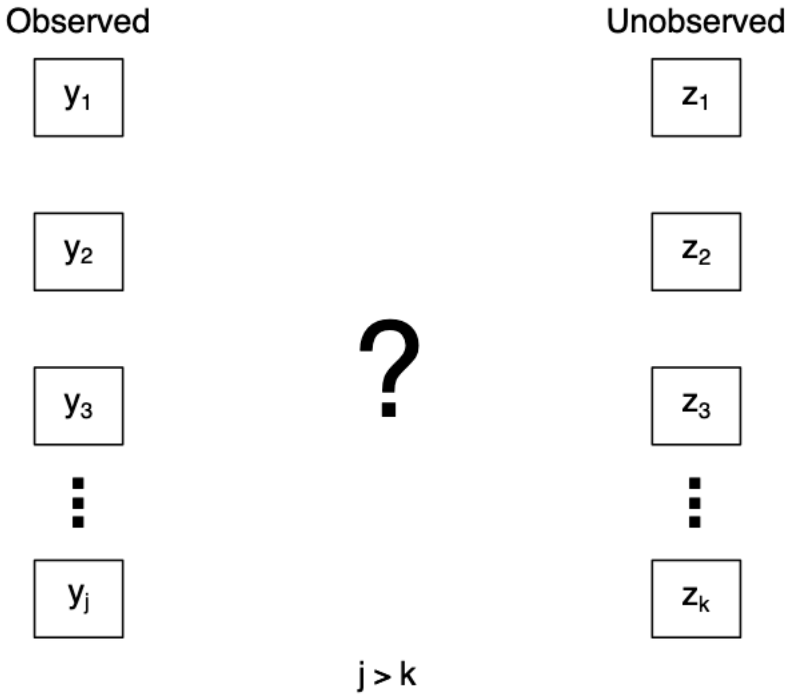
What Exactly Are We Doing?

- Statistically: Trying to use existing measures of the same underlying concept to generate "better" measures.
- Theoretically: Trying to obtain more precise, often more nuanced, and generally less error-laden measures of concepts of interest.

Why do this?

- "Better" measure of some underlying concept, where "better" means:
- Less polluted by measurement error.
- Higher level of measurement.
- Better able to distinguish between observations.
- Dimension Reduction.
- Reduce the effects of Multi-collinearity.

Motivation



Observed

y_1

y_2

y_3

⋮

y_j

?

Unobserved

z_1

z_2

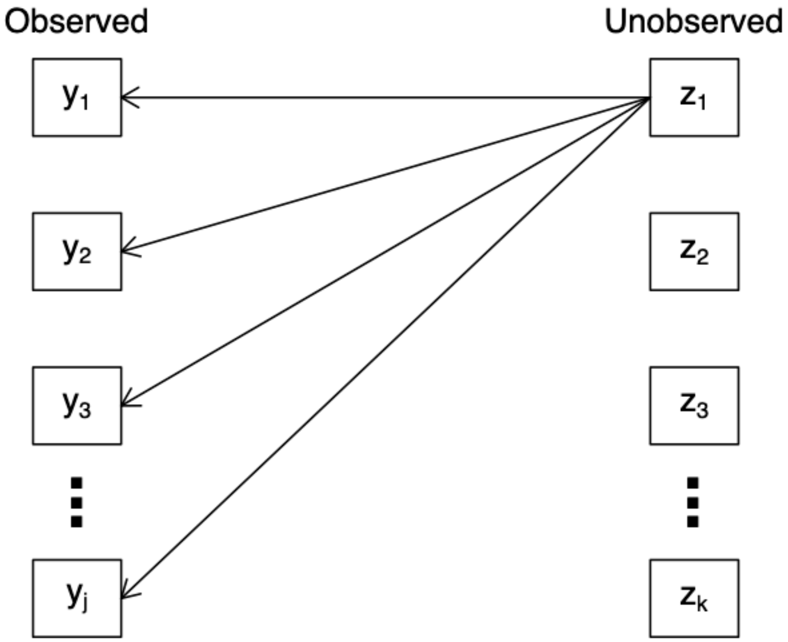
z_3

⋮

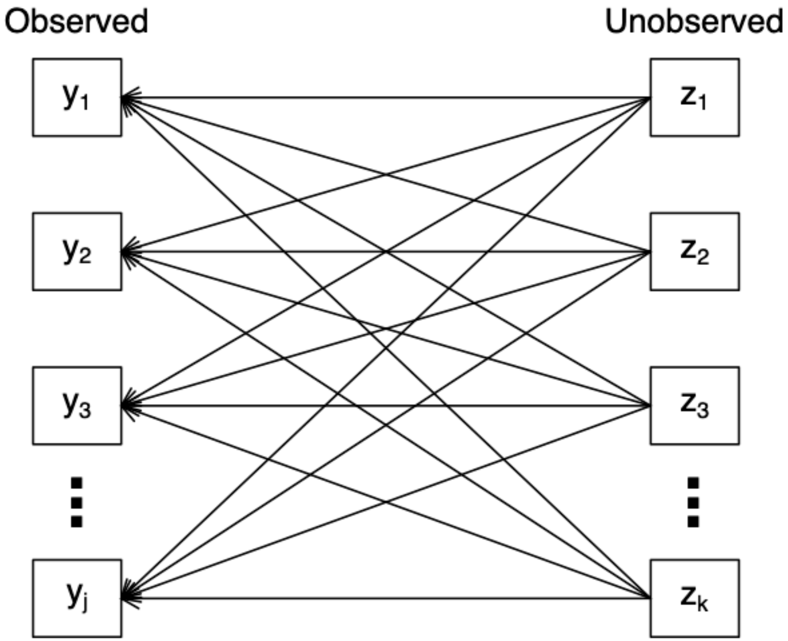
z_k

$j > k$

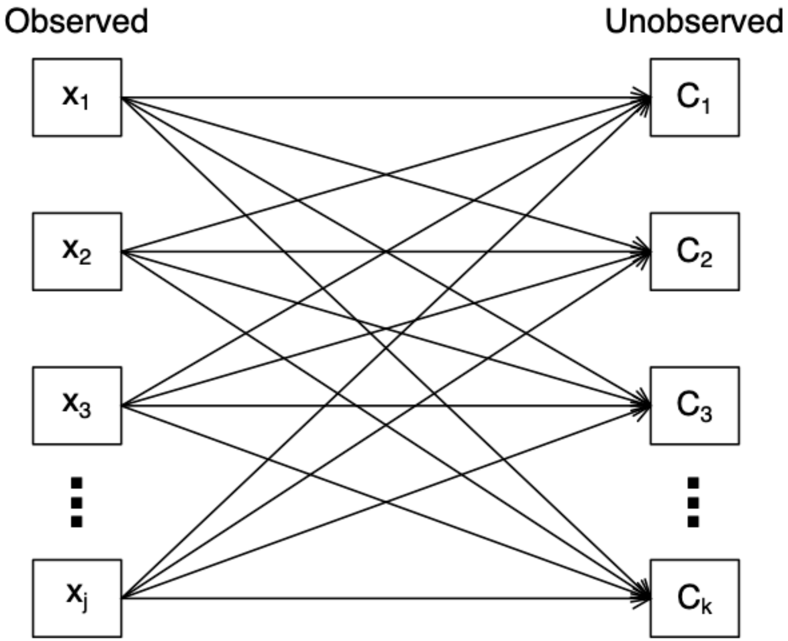
Simple



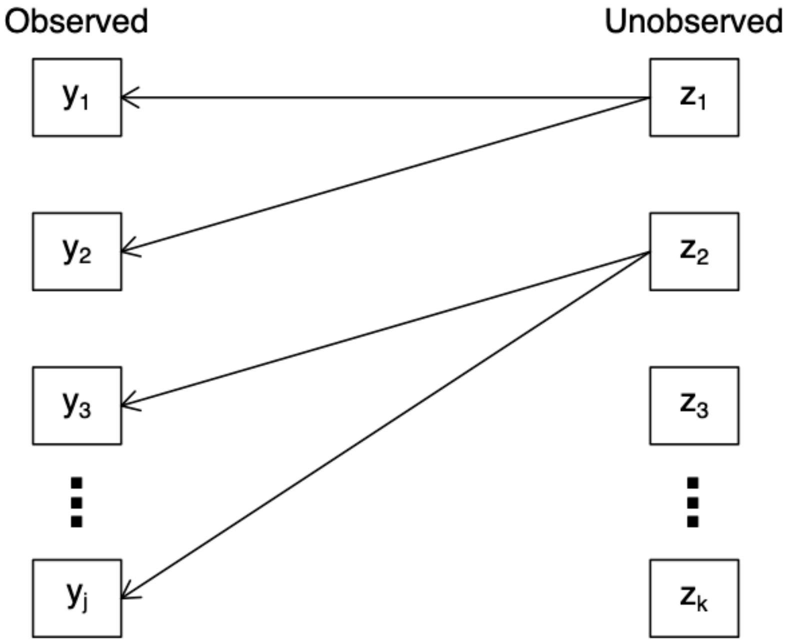
Complex



Complex (2)



Something in Between



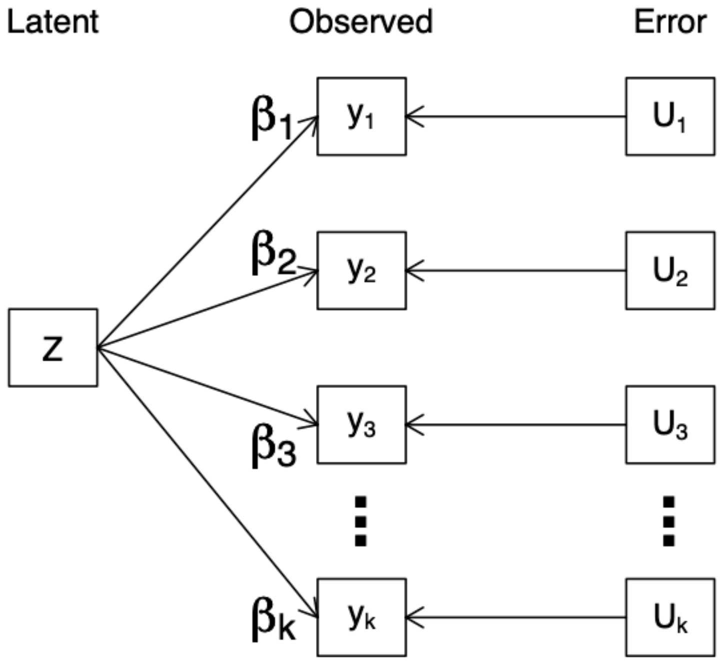
What we need

- **Theoretical Model** A theoretical understanding and/or hypothesis and/or assumption about how observed and unobserved variables are related.
- **Operational Model** A method of getting estimates of the parameters in the theoretical model.

Glossary

- **Measurement Error** Whatever makes an observation's value on a variable different from the "true" value.
- **Parameter** Anything that requires estimation to obtain a numerical value.
- **Latent Variable** (a.k.a. Unobserved Variable, Underlying Variable/Concept) Some variable that is unobserved and/or inherently unobservable.
- **Vector** A series of numbers in a particular order. (e.g., a vector of coefficients, a variable vector). Can be a row or column vector.
- **Matrix** A concatenation of a set of vectors with the same length, Indexed by [r,c].

Simple model



Simple model: Equation form

$$
\begin{aligned}
y_{i1} &= \beta_1 z_i + \varepsilon_{i1} \\
y_{i2} &= \beta_2 z_i + \varepsilon_{i2} \\
y_{i3} &= \beta_3 z_i + \varepsilon_{i3} \\
\vdots &= \quad \vdots \\
y_{ij} &= \beta_j z_i + \varepsilon_{ij}
\end{aligned}
$$

- How many data points do we have?
- How many parameters are there in this model?
- Note, from here on out, we are assuming that the observed variables have been standardized to have 0 mean and variance=1.

Simplifying Assumptions

- $\beta_1 = \beta_2 = \beta_3 = \ldots = \beta_j = 1$, which gives us the following:

$$
\begin{aligned}
y_{i1} &= z_i + \varepsilon_{i1} \\
y_{i2} &= z_i + \varepsilon_{i2} \\
y_{i3} &= z_i + \varepsilon_{i3} \\
\vdots &= \quad \vdots \\
y_{ij} &= z_i + \varepsilon_{ij}
\end{aligned}
$$

- There is some error that keeps the observed variable from being a perfect reflection of the "true" score on z.

One Observation

- Now, let's look what happens to one observation i. For simplicity of notation here, we remove the i subscripts:

$$\begin{aligned} Y &= z + U \\ E(Y) &= E(z + U) \\ &= z + E(U) \end{aligned}$$

- Remember, $E(U) = \bar{\varepsilon}$. If we can assume that the errors cancel out, that is that the sum of the errors is 0 (making the mean of the errors also 0), then we have the following result:

$$E(Y) = z$$

Is this model appropriate?

- Have to assume that the "signal" is the same across all of the variables in this model.
- Have to assume that there is no systematic measurement error. If every observation is off in the same direction, the estimate of the "true" dimension will also be biased, even in the limit.
- Also have to make a couple of other assumptions, but we'll get into those later.

Reliability

- Let's look at one observed variable: $Y_j = z + U_j$. One thing that we're often interested in is how can we account for the variance in a variable:

$$\begin{aligned} var(Y_j) &= var(z + U_j) \\ &= var(z) + var(U_j) + 2cov(z, U_j) \end{aligned}$$

- If we now assume that $cov(z, U_j) = 0$, that is, the errors are independent of z and we'll also assume $cov(e_i e_j) = 0 \quad \forall \quad i \neq j$ ($\forall$ just means "for every").

$$var(Y_j) = var(z) + var(U_j)$$

Reliability II

- Now, let's do a bit of rearranging:

$$1 = \frac{var(z)}{var(Y_j)} + \frac{var(U_j)}{var(Y_j)}$$

- The first piece of the first equation on this page, is then like an R^2 between the unobserved "true" score and the observed variable Y_j. We call this "reliability" - the proportion of variance in the observed variable that is due to the true, but unobservable dimension.

- In general, reliability refers to the repeatability and consistency of the measurement instrument. Thus, a measure is reliable if, when repeated, it produces similar results.

What does Reliability mean?

- The higher a variable's reliability, the better a measure it is of the estimated underlying dimension.

- If a variable has very low reliability, it is unlikely that it is measuring the same thing as the other variables **(no matter how much it's name or operationalization would suggest that it is)**.

Assumptions thus far (and one new one)

- $e_{ij} \sim iid$ (i.e., no systematic measurement error).
- z_i is the same for all y_{ij} (uni-dimensionality).

Now, I'm going to introduce one more assumption now: Monotone Homogeneity.

- Monotone Homogeneity means that the relationship between the observed variables and the true underlying dimension is monotonically increasing.
- By monotonically increasing, I mean that as the true dimension increases, the observed variable cannot decrease (though it could stay the same).
- This suggests that the input variables don't need to be interval-level, they only need to be ordinal.

Theoretical Conclusions

What does all of this mean?

- It means, that if the assumptions of this model are reasonable, then you can get an estimate of the underlying dimension by adding up, or taking the mean, of the observed variables for each observation.
- Remember, we found that:

$$E(Y) = z + \underbrace{E(U)}_{0}$$

- Now, we have to estimate reliability.

Estimating Reliability

- We can never know a variable's true reliability since it depends, in part, on the variance of the true score.
- We can get an estimate of a scale's reliability. In R, we do this with Cronbach's α [`alpha()` in the `psych` package].
- The formula for α is as follows:

$$\alpha = \frac{k\bar{r}}{1 + \bar{r}(k - 1)}$$

where $\bar{r}$ is the average of the elements below the diagnoal of the correlation matrix for the observed variables and k is the number of observed variables.

More definitions.

Assume that we have observed variables $y_{i1}, y_{i2}, \ldots, y_{ik}$

- The "test score" is simply $\sum_{j=1}^{k} y_{ij}$.
- The "rest score" for variable y_{i1} is simply $\sum_{j=2}^{k} y_{ij}$. This is sum of the *rest* of the variables. The rest score for y_{i5} would be $\sum_{j=1}^{4} y_{ij} + \sum_{j=6}^{k} y_{ij}$.

Reliability the easy way

```r
library(psych)
dat <- rio::import('data/srm.dta')
alpha(dat)
```

```
##
## Reliability analysis
## Call: alpha(x = dat)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase  mean   sd median_r
##       0.94      0.94    0.93       0.8  16 0.003 0.023 0.94      0.8
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.94  0.94  0.95
## Duhachek  0.94  0.94  0.95
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r S/N alpha se   var.r med.r
## X1      0.93      0.93    0.89      0.81  13    0.0040 2.4e-04  0.81
## X2      0.93      0.93    0.90      0.81  13    0.0039 1.4e-04  0.81
## X3      0.92      0.92    0.89      0.80  12    0.0042 7.0e-05  0.80
## X4      0.92      0.92    0.89      0.80  12    0.0043 7.1e-05  0.79
##
##  Item statistics
##       n raw.r std.r r.cor r.drop   mean sd
## X1 1000  0.92  0.92  0.88   0.86 0.0293  1
## X2 1000  0.92  0.92  0.88   0.85 0.0302  1
## X3 1000  0.93  0.93  0.90   0.87 0.0071  1
## X4 1000  0.93  0.93  0.90   0.87 0.0254  1
```

We know that $0 \leq$ Reliability ≤ 1 and the closer to 1, the more reliable the scale is. This scale is quite reliable.

Reading the R Output

- The `r.cor` column is the correlation between the individual variable and the test score (the scale created by all of the variables).

- This may not be the best measure because the test score contains the variable of interest, so those two things are related by definition.

- The `r.drop` column is the correlation between the individual variable and that variable's rest score (the scale made from summing all of the rest of the variables).

- The `raw_alpha` column shows what Cronbach's α would be if we omitted that variable from the scale. Ideally, you want the overall α to be bigger than it would be if you deleted any of the individual items.

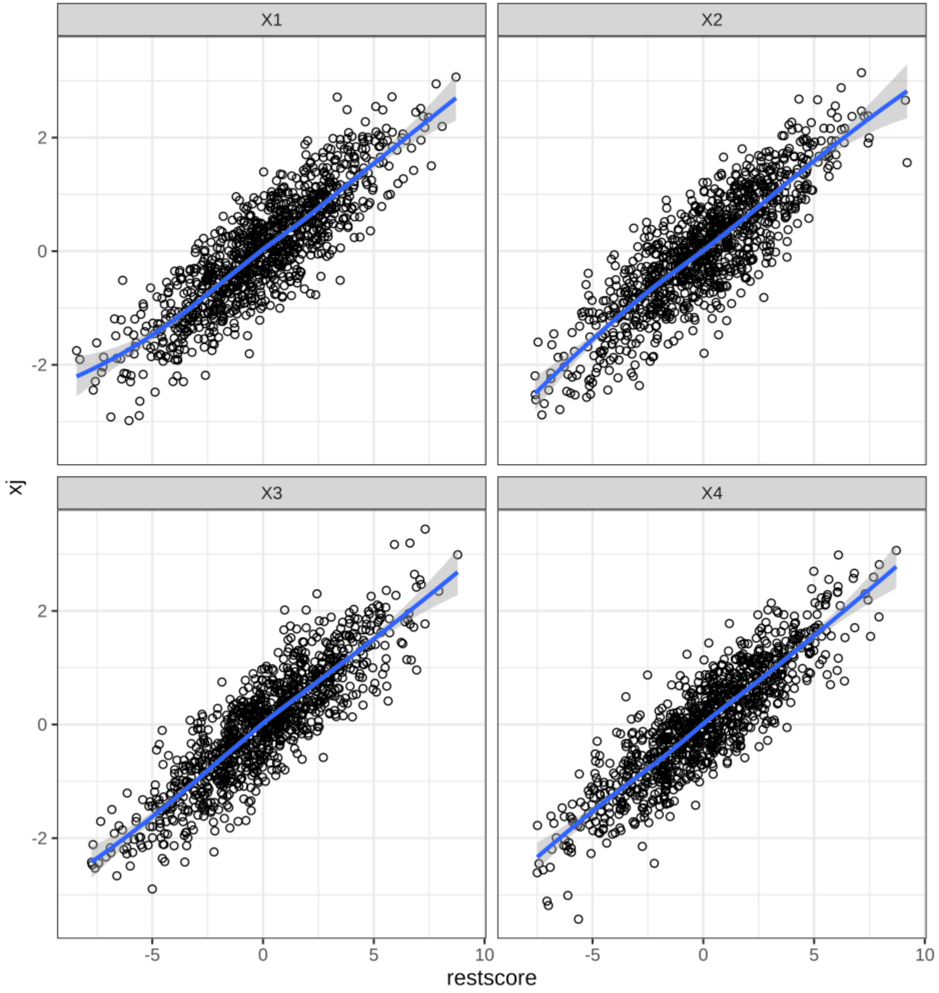
Evaluating Assumptions

- Uni-dimensionality: To evaluate this assumption, it is probably best to look at the correlation matrix. What you want to see is relatively similar (hopefully high-ish) correlations across the matrix. To the extent the blocks of high and low correlations exist, that is a problem.

- Monotone homogeneity can be evaluated by plotting each variable against its rest score. I've written an R program that will do this called "restplot". The program just takes a variable list and then plots each variable against the row-mean of the remaining variables.

Restplot in R

```r
library(uwo9592)
restplot(dat,
         span =    .7,
         family = "symmetric",
         degree = 2)
```

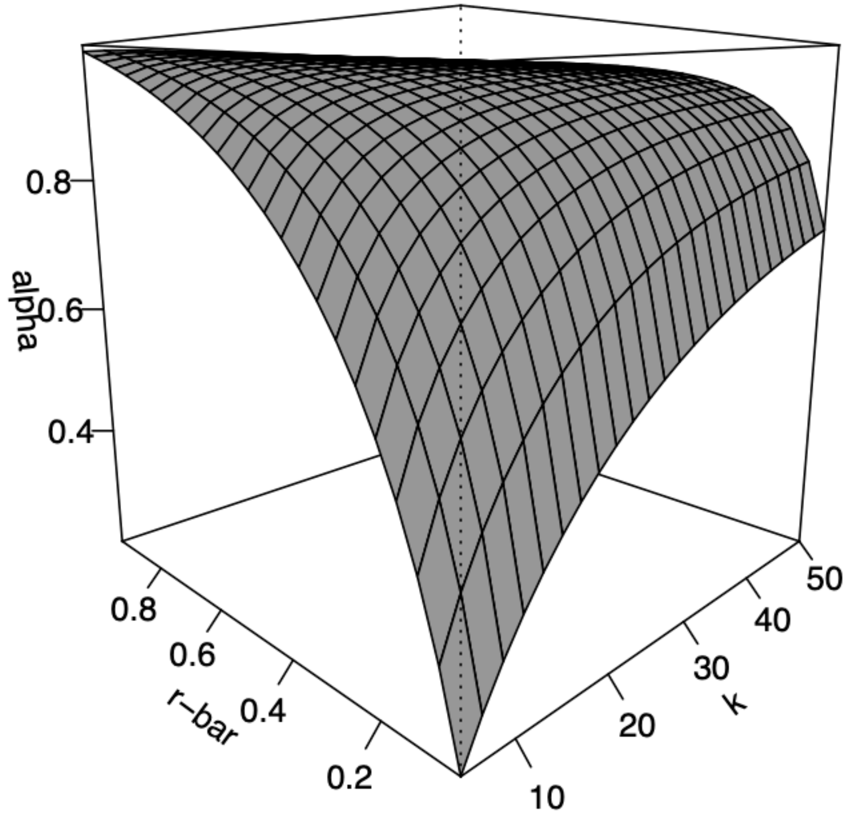


Caveats and Cautions

- Cronbach's α is not a good *test* of dimensionality. It is likely to give an underestimate of the dimensionality of your data. Testing dimensionality is a task better suited to factor analysis, which we will talk about later.

- As you include more variables, it is possible to see a relatively high α value without very high inter-correlations.

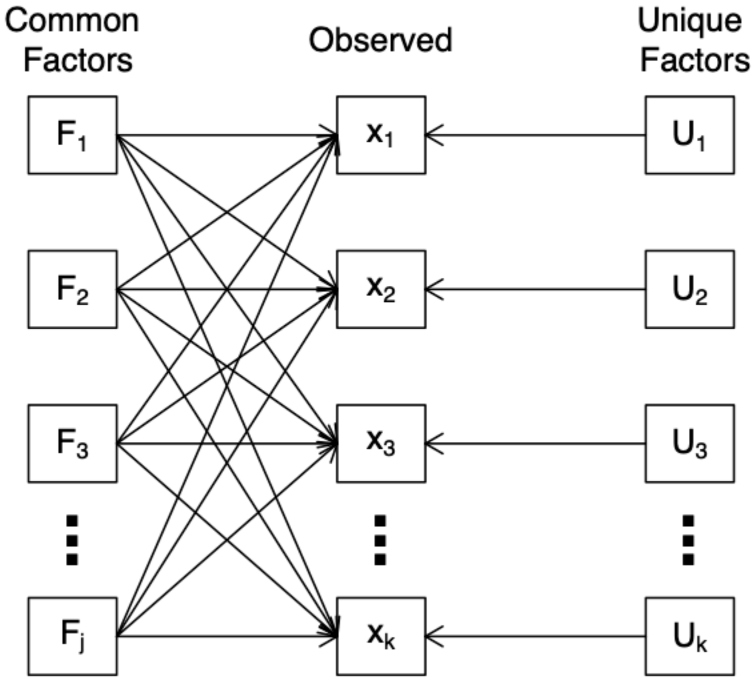
α as a function of k and $\bar{r}$



SRM Conclusion

- At the end, you have a new variable (the sum of the observed variables for each observation) that is an estimate of the latent dimension.
- If the assumptions of this model hold, the estimate is "better" than any of the individual variables because the idiosyncrasies and measurement error have canceled each other out.
- The resulting variable is (roughly) an interval level variable after starting with ordinal level input data.

Graphical Representation of Common Factor Model



Common Factor Model in Equation Form

$$\begin{aligned}
x_{i1} &= a_{11}F_{i1} + a_{12}F_{i2} + \cdots + a_{1m}F_{im} + U_{i1} \\
x_{i2} &= a_{21}F_{i1} + a_{22}F_{i2} + \cdots + a_{2m}F_{im} + U_{i2} \\
&\vdots \\
x_{ik} &= a_{k1}F_{i1} + a_{k2}F_{i2} + \cdots + a_{km}F_{im} + U_{ik}
\end{aligned}$$

For each observation $i = 1, 2, \ldots, n$ on each observed variable in $1, 2, \ldots, k$ and each factor $1, 2, \ldots, m$, where always $m \leq k$ and usually $m \ll k$.

Factor Analysis Glossary (1)

- **Factor Loading** Coefficient relating the unobserved variable to the observed variable a_{km}.
- **Factor Pattern Matrix** Matrix of factor loadings, usually referred to as $\mathbf{A}$

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,m} \\ \vdots & \vdots & \vdots & \vdots \\ a_{k,1} & a_{k,2} & \cdots & a_{k,m} \end{bmatrix}$$

- **Communality** Amount of an observed variable's variance shared with the other variables; usually denoted as $h_k^2 = \sum_{j=1}^m a_{kj}$.
- **Uniqueness** Amount of an observed variable's variance *not* shared with the other variables; usually denoted U_k.

Factor Analysis Glossary (2)

- **Eigenvalue** In an un-rotated factor solution, the amount of variance explained by each factor.
- **Rotation** Factor solutions are only identified up to a rotation, meaning there are infinitely many solutions that are equally "good" in terms of variance explained (ability to reproduce the correlation matrix). Rotating means moving the factors around in space often so they explain the same amount of variance, but so they also have other desirable properties.
- **Factor Structure Matrix** Asymmetric matrix of correlations between the observed variables and the factors. This is the same as the Factor Pattern Matrix for orthogonal factors, but these are not the same when we allow the factors to be correlated.

Factor Pattern Matrix

					$\sum_{j=1}^{k} a_{jm}^2$
	$a_{1,1}$	$a_{1,2}$	$\cdots$	$a_{1,m}$	h_1^2
	$a_{2,1}$	$a_{2,2}$	$\cdots$	$a_{2,m}$	h_2^2
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
	$a_{k,1}$	$a_{k,2}$	$\cdots$	$a_{k,m}$	h_k^2
$\sum_{l=1}^{m} a_{kl}^2$	λ_1^2	λ_2^2	$\cdots$	λ_m^2	

Things to note

- This model proposes that there is potentially less variance than the total combined variance to explain. In math, we would say: $\sum_{l=1}^{m} h_l^2 < k$.

- This will be a source of uncertainty for us as we will have to get an estimate of the communality for each variable.

- This is a model of *linear* structure. Factor Analysis models the underlying linear association among the variables with a smaller set of factors.

What we Want

- $\mathbf{A}$ (Factor Pattern Matrix), the coefficients relating the unobserved factors to the observed variables.
- h_j^2 (Communality), we need to supply initial estimates and then we can either iteratively improve them or not.
- λ_m^2 (Eigenvalue), we will also want to know the amount of total variance explained by each of the factors.
- $\hat{F}_{im}$, estimates of the unobserved variables for each observation.

The Fundamental Theorem of Factor Analysis

$$\begin{aligned}
\mathbf{X} &= \mathbf{AF} + \mathbf{U} \\
\mathbf{XX}' &= (\mathbf{AF} + \mathbf{U})(\mathbf{AF} + \mathbf{U})' \\
&= (\mathbf{AF} + \mathbf{U})(\mathbf{F}'\mathbf{A}' + \mathbf{U}') \\
&= \mathbf{AFF}'\mathbf{A}' + \mathbf{AFU}' + \mathbf{UF}'\mathbf{A}' + \mathbf{UU}' \\
E(\mathbf{XX}') &= \mathbf{A}E(\mathbf{FF}')\mathbf{A}' + \mathbf{A}E(\mathbf{FU}') + E(\mathbf{UF}')\mathbf{A}' + E(\mathbf{UU}') \\
\mathbf{R}_{XX} &= \mathbf{AR}_{FF}\mathbf{A}' + \mathbf{A}\underbrace{\mathbf{R}_{FU}}_{0} + \underbrace{\mathbf{R}_{UF}}_{0}\mathbf{A}' + \mathbf{R}_{UU} \\
&= \mathbf{AR}_{FF}\mathbf{A}' + \mathbf{R}_{UU}
\end{aligned}$$

Here, $\mathbf{R}_{FF}$ is the correlation between the factors, which we'll assume for the moment is $\mathbf{I}$, a matrix with 1 on the diagonal and 0 elsewhere and $\mathbf{R}_{UU}$ we'll call the variance-covariance matrix of the uniquenesses, with unique variance on the diagonal and 0 elsewhere.

The Fundamental Theorem of Factor Analysis (2)

$$\begin{aligned} \mathbf{R}_{XX} &= \mathbf{AIA}' + \mathbf{R}_{UU} \\ \mathbf{R}_{XX} - \mathbf{R}_{UU} &= \mathbf{AIA}' \\ \tilde{\mathbf{R}}_{XX} &= \mathbf{AA}' \end{aligned}$$

What we're saying here is that we can break the adjusted correlation matrix $\tilde{\mathbf{R}}_{XX}$ into the product of something $\mathbf{A}$ and itself. This is where we get the term *Factor* analysis, because we're essentially *factoring* a matrix.

- We can "decompose" a correlation matrix into the product of a matrix $(\mathbf{A})$ and its transpose.

How do we figure out $\mathbf{R}_{UU}$?

Communality

- Communality is the amount (proportion) of a variable's variance that it shares with the other variables. The common variance we are trying to explain.
- Let's think of a very simple factor model:

$$y = b_1 x_1 + \varepsilon$$

- Here, we are not saying that all of the variance in y can be explained by x. Rather, we're saying that there is some part of y that varies systematically with x and some part that is unique to y and unrelated to x.
- Now, transform the above equation into the notation for our factor model:

$$x_1 = a_{11} F_1 + U_1$$

Here, we're saying the same thing - that part of the variance of x_1 is unique to x_1 and has nothing to do with F_1.

Communality Options

- $h_j^2 = 1$: This is called the "principal components factor model". This is almost certainly an *overestimate* of a variable's communality.
- $h_j^2 = R^2_{-j}$: This is the "squared multiple correlation" - the R^2 of a regression with x_j as the dependent variable and all of the rest of the observed variables as the independent variables. The SMC is (in a properly specified factor model) the lower bound of the true, but unknown communality.
- $h_j^2 = \text{Reliability}$: If we know a variable's reliability, we could use this as the communality. This is theoretically the upper bound of the true, but unknown communality, but we usually don't know it. .

Improving Communality Estimates

- Once we decide on a strategy, we can then decide if we want to iteratively make our estimates better or not, this applies to the $h_j^2 = R_{-j}^2$ strategy.

- The "Uniqueness" just equals $1 - h_j^2$. Remember, the Uniqueness and Communality sum to 1. All of a variable's variance is a function of 1) the part we can explain, plus 2) the part we can't explain.

Singular Value Decomposition

- We employ a mathematical tool called the Singular Value Decomposition.
- This uncovers the "basic structure" of a matrix.
- It has three components: $\underbrace{\mathbf{U}}_{n\times k}$, $\underbrace{\mathbf{D}}_{k\times k}$ and $\underbrace{\mathbf{V}'}_{k\times k}$ where:
- n is the number of rows in the original data, and
- k is the number of columns in the original data
- $\mathbf{U}$ gives information about the rows of the original matrix,
- $\mathbf{V}$ gives information about the columns of the original matrix, and
- $\mathbf{D}$ gives the variance accounted for by the rows of $\mathbf{U}$ and $\mathbf{V}$.

The $\mathbf{d}$'s are what we call the "eigenvalues" when the original matrix is square and symmetric.

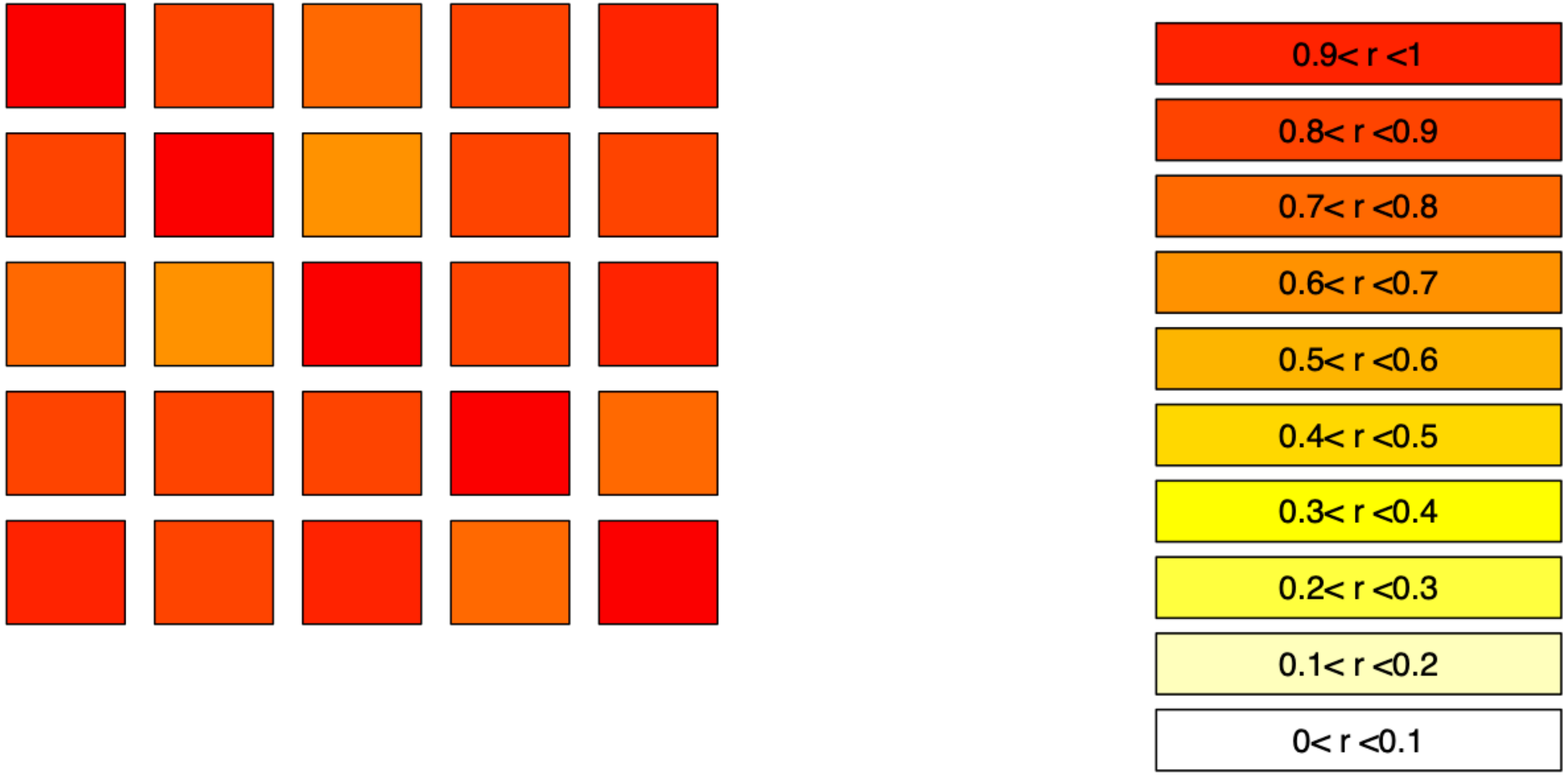
- Here, $\mathbf{X} = \mathbf{UDV}'$

A Small Digression about SVD

- Sometimes, when you're unsure about what a particular "thing" does, it might make sense to see what it does on made-up data, data where you know the properties. So, if SVD *actually* uncovers the basic structure of a matrix, let's see what kind of results we get in different situations.

- I've got two different situations below - one where there is one underlying trait and one where there are two, mostly uncorrelated underlying traits.

One Underlying Trait: Graphical Correlation Matrix

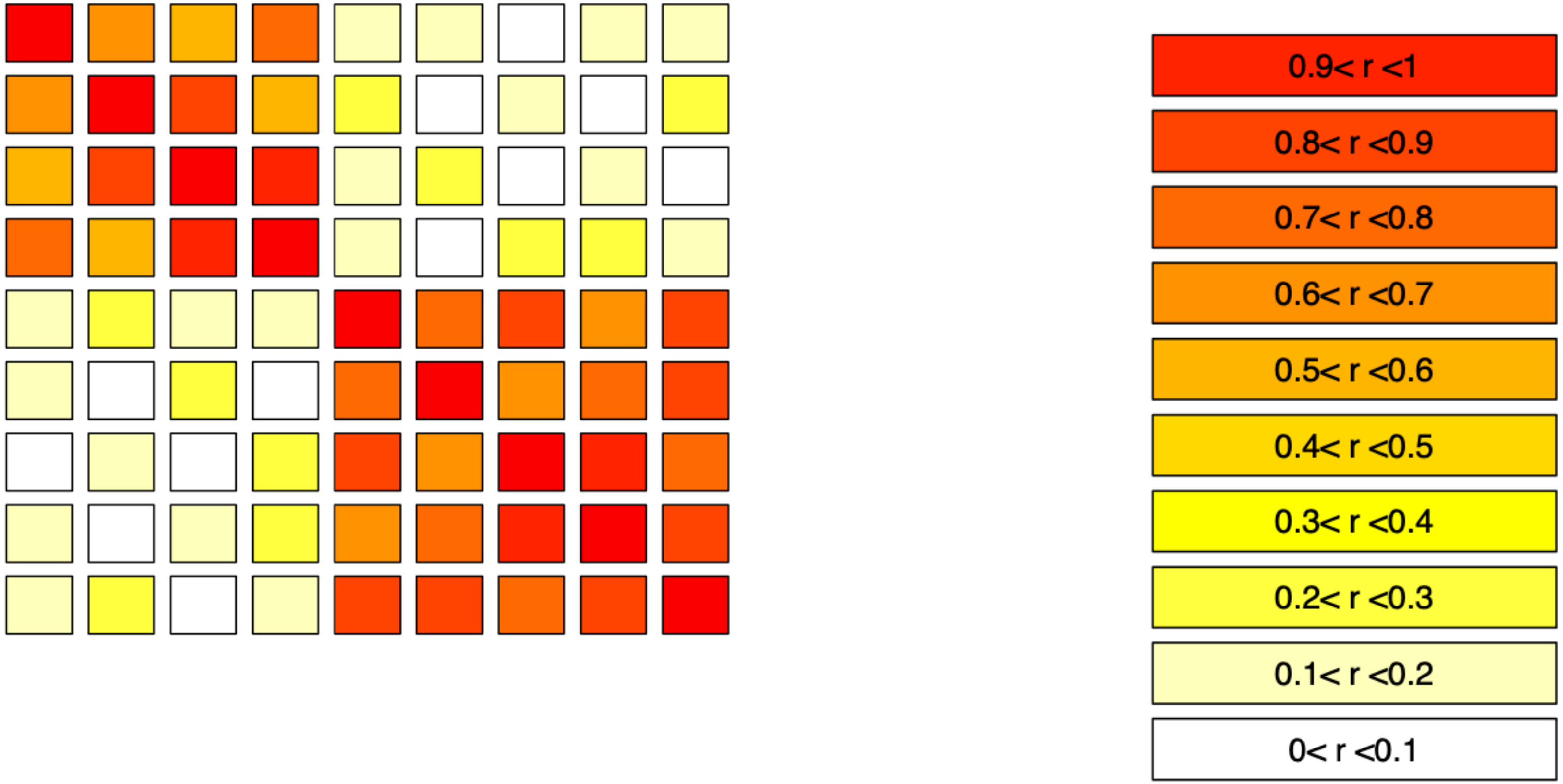


One Underlying Trait: SVD

```
Eigenvalues
4.12 0.43 0.31 0.17 0.04

Eigenvectors
-0.45  0.21 -0.33 -0.72 -0.33
-0.43  0.61 -0.03  0.59 -0.26
-0.43 -0.70  0.18  0.20 -0.48
-0.44  0.12  0.75 -0.22  0.40
-0.46 -0.24 -0.52  0.17  0.64
```

Two Underlying Traits: Graphical Correlation Matrix



Two Underlying Traits: SVD

```
Eigenvalues

4.11 2.84 0.61 0.55 0.47 0.34 0.20 0.12 0.01

Eigenvectors
-0.16  0.43  0.07 -0.80 -0.15 -0.01  0.22  0.26 -0.02
-0.18  0.45 -0.55  0.17 -0.37  0.40  0.02 -0.37  0.01
-0.19  0.50 -0.06  0.46  0.33 -0.07  0.09  0.60  0.13
-0.20  0.48  0.52  0.08  0.13 -0.27 -0.29 -0.50 -0.17
-0.41 -0.15 -0.25  0.01 -0.31 -0.70 -0.06 -0.04  0.40
-0.40 -0.16 -0.30 -0.13  0.63 -0.06  0.42 -0.29 -0.23
-0.42 -0.18  0.26  0.25 -0.45  0.03  0.29  0.19 -0.58
-0.43 -0.16  0.41  0.02  0.03  0.47  0.15 -0.09  0.61
-0.43 -0.17 -0.17 -0.17  0.13  0.23 -0.76  0.22 -0.18
```

Factor Analysis and SVD

Let's go back to our old friend SVD, we know that we can decompose a data matrix into constituent parts:

$$\mathbf{X} = \mathbf{UDV}'$$

We also know that a correlation matrix is the product of two matrices scaled by $\frac{1}{n-1}$:

$$\mathbf{R}_{XX} = \mathbf{X}'\mathbf{X}(n-1)^{-1}$$

So, now we can substitute the first piece of information into the second:

$$\mathbf{R}_{XX} = (\mathbf{UDV}')'(\mathbf{UDV}')(n-1)^{-1}$$

Solving the Factor Model with SVD

$$
\begin{aligned}
\mathbf{R}_{XX} &= (\mathbf{UDV}')'(\mathbf{UDV}')(n-1)^{-1} \\
&= (\mathbf{VD}'\mathbf{U}'\mathbf{UDV}')(n-1)^{-1} \\
&= (\mathbf{VD}'\mathbf{IDV}')(n-1)^{-1} \\
&= \mathbf{VD}'(n-1)^{-1}\mathbf{DV}' \\
&= \left(\mathbf{VD}'(n-1)^{-\frac{1}{2}}\right)\left((n-1)^{-\frac{1}{2}}\mathbf{DV}'\right)
\end{aligned}
$$

Now, if we rename $\mathbf{D}'(n-1)^{-\frac{1}{2}} = \mathbf{\Lambda}$, then we have:

$$
\mathbf{R}_{XX} = \underbrace{\mathbf{V\Lambda}}_{\mathbf{A}}\underbrace{\mathbf{\Lambda V}'}_{\mathbf{A}'}
$$

Here, $\mathbf{\Lambda}^2$ is the diagonal matrix of eigenvalues, so if we want to create $\mathbf{\Lambda}$, we need to create a diagonal matrix of the square root of the eigenvalues.

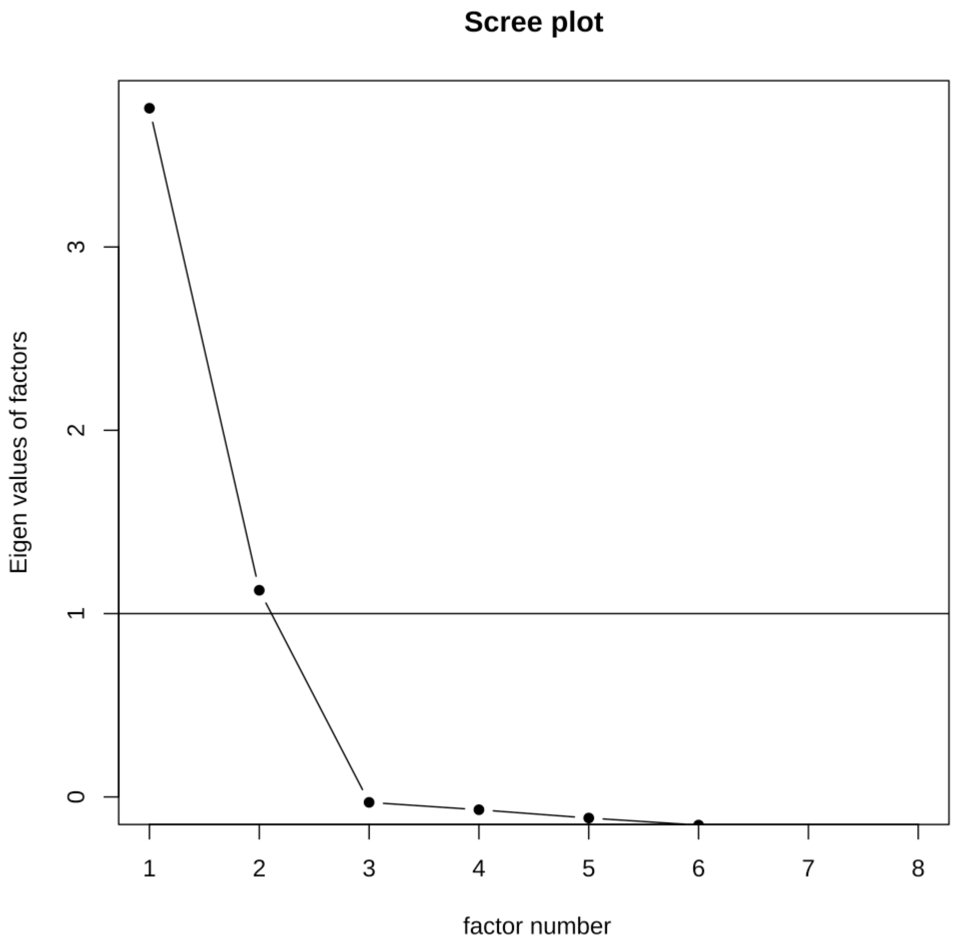
Extraction Methods

When we estimate a factor analysis, we have to choose an "extraction method" which basically relates to what we want to do with the communalities and how we want to estimate the the factor pattern matrix. The possibilities of note are:

1. Iterated Principal Factor ("pa"), starts the same as "pf", but iteratively improves communality estimates. Remember, if the model is appropriate/properly specified, then SMC's are the lower bound (and as such, almost certainly an underestimate) of the true communality.
2. Minimum Residual ("minres") minimizes the sum of squared residuals in an OLS fashion.
3. Maximum Likelihood ("ml") does not employ the SVD, rather it maximizes $\Pr(\tilde{\mathbf{R}}_{XX}|\mathbf{A})$.

Data: Democracy and Repression

```r
library(rio)
library(psych)
dat <- import("data/dem_rep.dta")
X <- na.omit(dat[,-c(1:3)])
R <- cor(X)
scree(R, pc=FALSE)
```

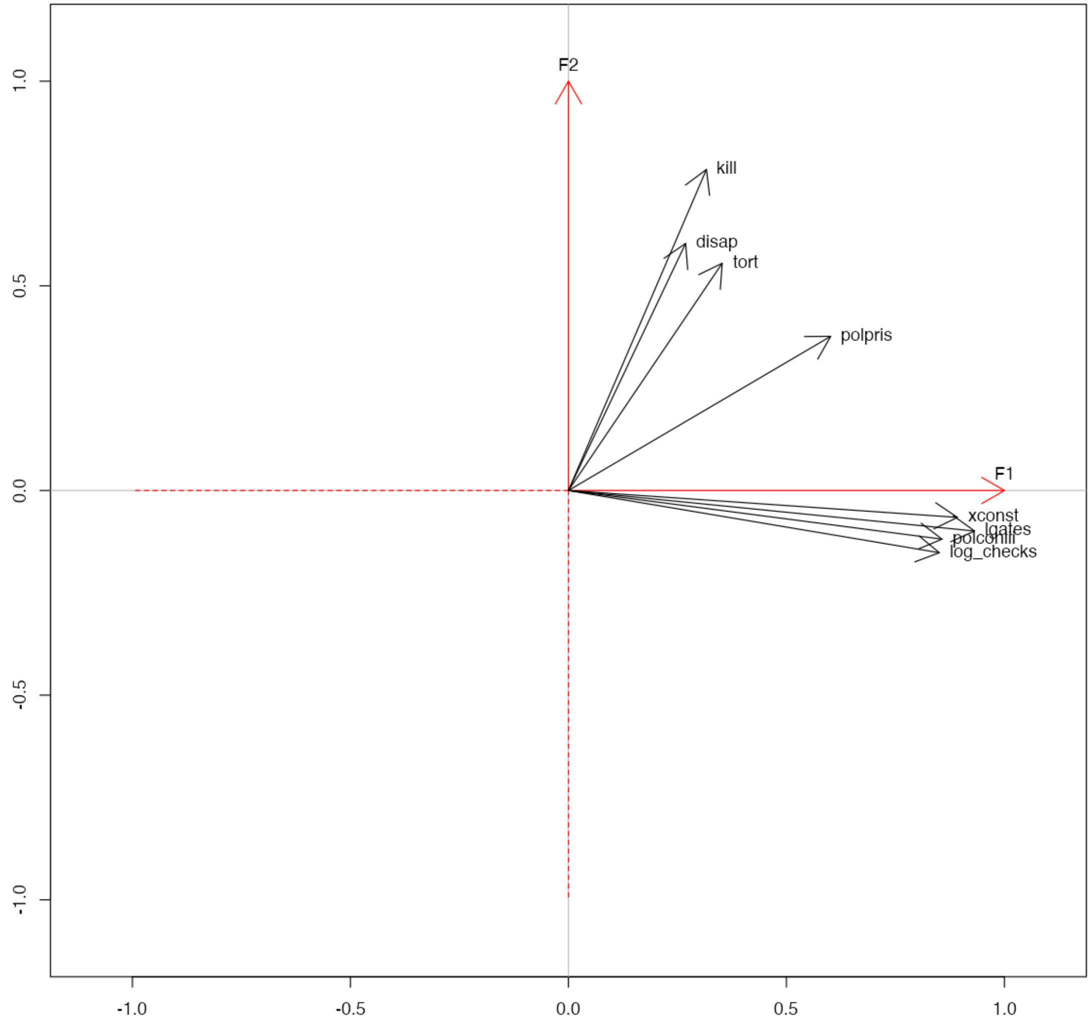


Democracy and Repression: 2 Factors

```
fn <- fa(X, nfactors=2, rotate="none", fm="pa", SMC=TRUE)
fn
```

```
## Factor Analysis using method =  pa
## Call: fa(r = X, nfactors = 2, rotate = "none", SMC = TRUE, fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
##             PA1   PA2   h2   u2 com
## xconst     0.88 -0.23 0.82 0.18 1.1
## polconiii  0.83 -0.30 0.78 0.22 1.3
## lgates     0.88 -0.28 0.85 0.15 1.2
## log_checks 0.81 -0.30 0.76 0.24 1.3
## polpris    0.66  0.30 0.52 0.48 1.4
## disap      0.36  0.56 0.44 0.56 1.7
## tort       0.46  0.48 0.43 0.57 2.0
## kill       0.45  0.72 0.72 0.28 1.7
##
##                        PA1  PA2
## SS loadings           3.87 1.45
## Proportion Var        0.48 0.18
## Cumulative Var        0.48 0.67
## Proportion Explained  0.73 0.27
## Cumulative Proportion 0.73 1.00
##
## Mean item complexity =  1.5
## Test of the hypothesis that 2 factors are sufficient.
##
## df null model =  28  with the objective function =  5.21 with Chi Square =  13214.16
## df of  the model are 13  and the objective function was  0.07
##
## The root mean square of the residuals (RMSR) is  0.02
```

Visualizing the Solution



What is Rotation

- Factor analysis is only *identified* up to a rotation. This means that any orientation of the factors in m dimensional space that preserves the lengths of and angles between all of the variable vectors will give an equally good reproduction of the correlation matrix.

- If we have a factor pattern matrix $\mathbf{\Lambda}$, then we can pick some matrix $\mathbf{T}$ such that $\mathbf{\Lambda}^* = \mathbf{\Lambda}\mathbf{T}$. Where, in the two-factor solution:

$$\mathbf{T} = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix}$$

where α is the number of degrees you want to rotate the solution.

Rotation (2)

If every solution is equally "good" how do we choose discriminate between the solutions? We are looking for something called "simple structure". This mean that :

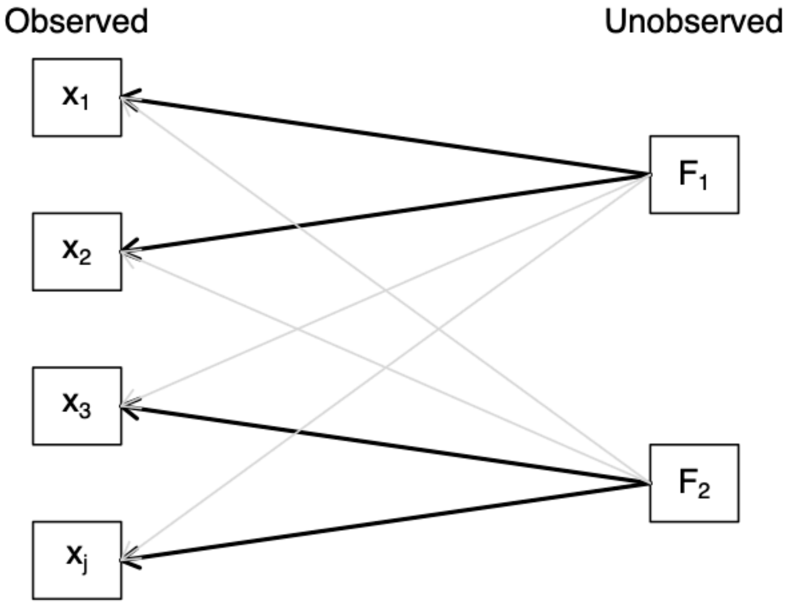
1. As few factors as possible influence any one variable, and
2. Each factor influences as few variables as possible.

This means we want factor pattern coefficients to be as close to zero or one as possible.

- Note, in R, the trigonometric functions require input and produce output in radians.

$$1^\circ = \frac{\pi}{180}\ \text{radians}$$

Simple Structure Solution

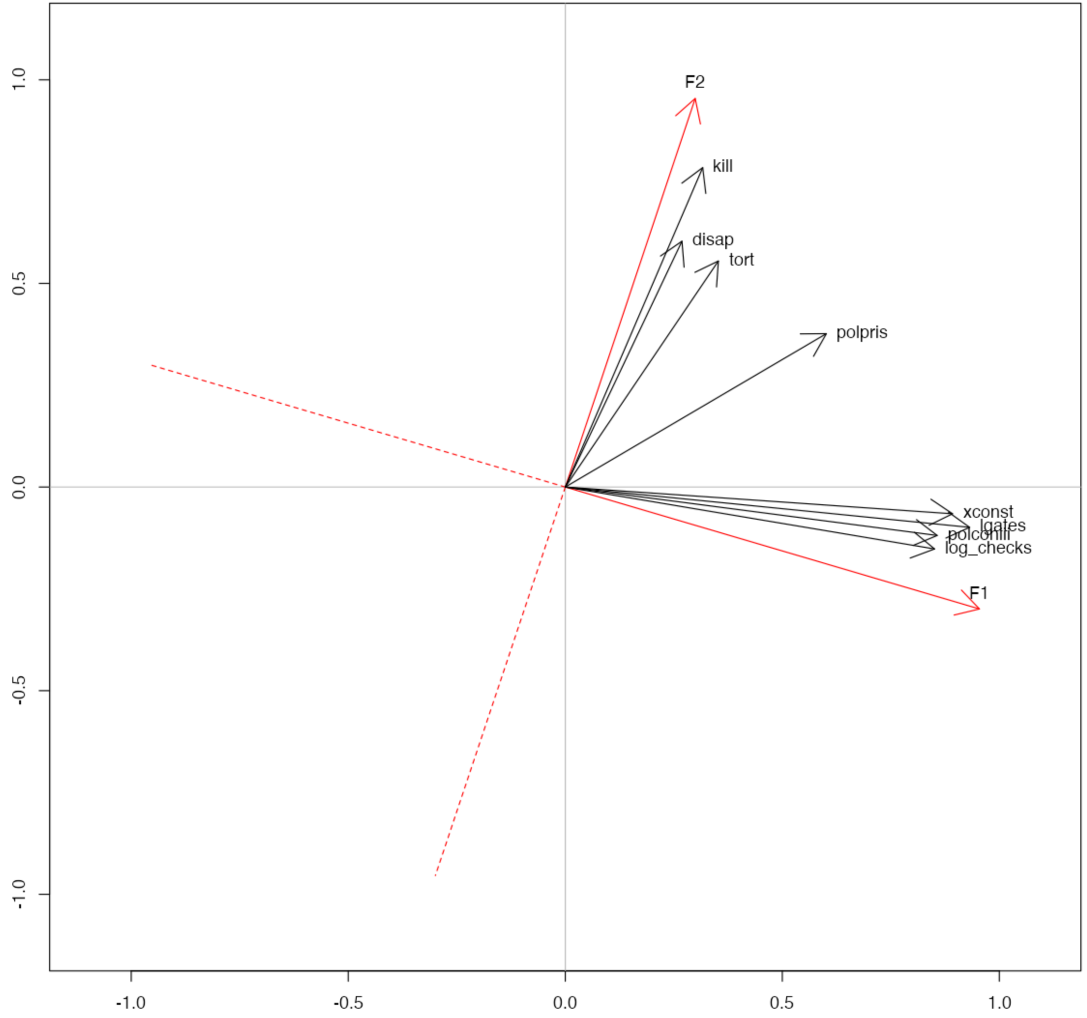


Finding Simple Structure

- A Varimax rotation finds the matrix $\mathbf{T}$ which maximizes the variance within the column. Why do this?

	$\mathbf{A_1}$	$\mathbf{A_2}$	$\mathbf{A_3}$
a_{11}	.5	.7	1
a_{21}	.5	.7	1
a_{31}	.5	.3	0
a_{41}	.5	.3	0
σ_A	0	0.23	0.58

Visualizing Varimax Rotation

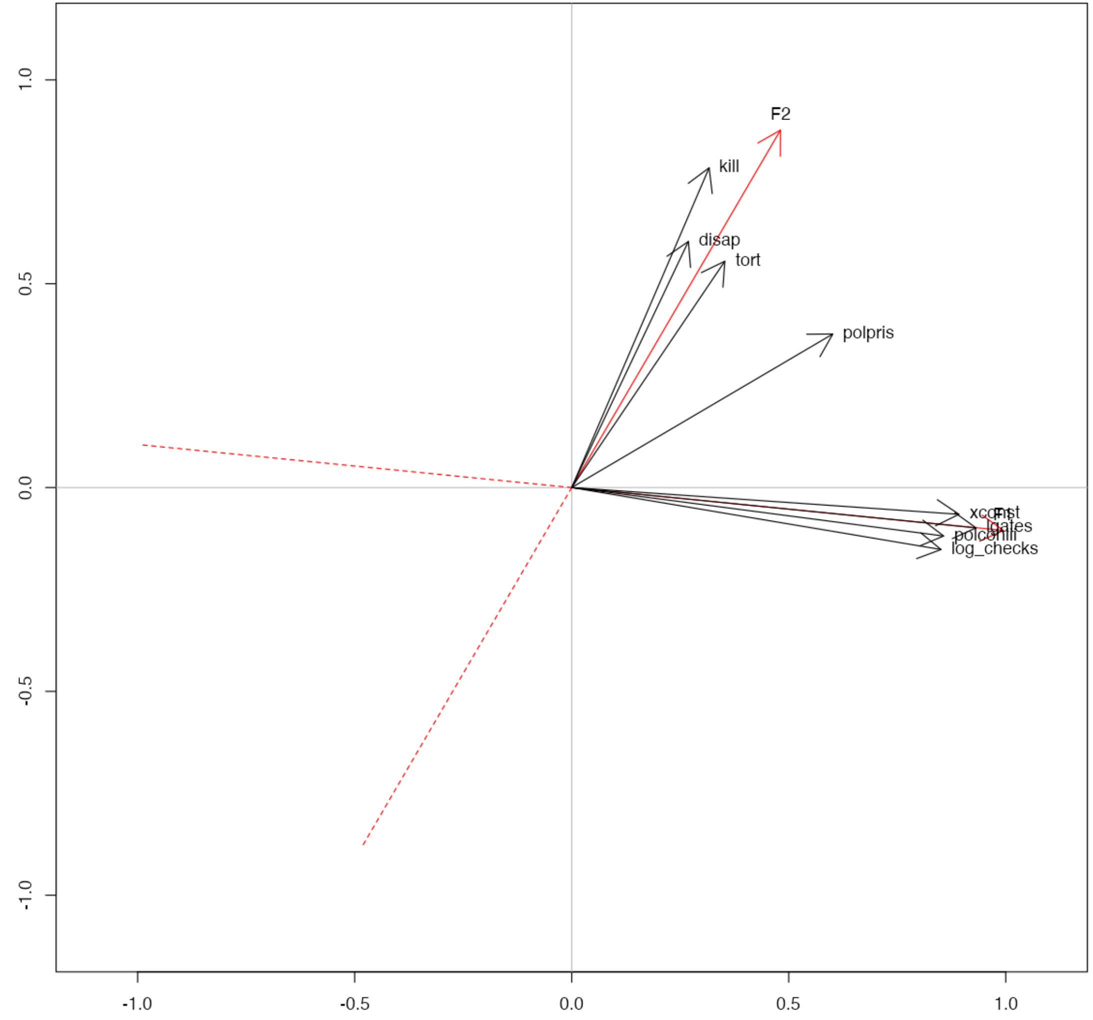


More on Rotation

Both the original solution and the varimax rotation were orthogonal solutions, ones that maintain the zero correlation between all of the factors.

- We might want to relax this assumption.
- It is not the case in the real world that everything we would want to model with latent variables are uncorrelated.
- This idea is operationalized in the *promax* rotation.

Visualizing Promax Rotation



Communality in Obliquely Rotated Factor Solutions

One difference that an oblique rotation induces is that the communality is no longer the sum of the squared factor pattern coefficients. Instead, it is the following:

$$h_j^2 = \sum_{l=1}^{m} a_{jl}^2 + \sum_{l=1}^{m-1} \sum_{k=l+1}^{m} 2a_{jl}a_{jk}r_{F_lF_k}$$

- For example, if we have the following model: $Z_1 = a_{11}F_1 + a_{12}F_2 + U_1$, then

$$h_1^2 = a_{11}^2 + a_{12}^2 + 2a_{11}a_{12}r_{F_1F_2}$$

Factor Scores

Generally, we are doing this not only to understand the structure of the data, but also to get estimates of the m-dimensional representation of our k-dimensional data matrix.

$$X_j = a_{j1}F_1 + a_{j2}F_2 + \cdots + a_{jm}F_m$$

It's nice that we know $\mathbf{A}$, but what we really want is a matrix B with elements b_{jm} such that:

$$\hat{F}_m = X_1 b_{1m} + X_2 b_{2m} + \cdots + X_k b_{km}$$

We need this because the X variables are the only information we have to identify the $\hat{F}$'s.

Factor Scores (2)

This is a pretty complicated problem and I won't bore you with the details, but we can obtain the matrix of scoring coefficients as follows:

$$\mathbf{B} = \mathbf{R}_{XX}^{-1}\mathbf{A}\mathbf{R}_{FF}$$

where $\mathbf{R}_{XX}$ is the correlation matrix of the observed variables, $\mathbf{A}$ is the factor pattern matrix and $\mathbf{R}_{FF}$ is the correlation matrix between the factors.

Factor Scores

```r
demfa2 <- fa(X, 2, fm="pa", rotate="promax", scores=T)
fa.scores <- scale(demfa2$scores)
srm.scores <- scale(cbind(rowMeans(X[,1:4]),
   rowMeans(X[,5:8])))
colnames(fa.scores) <- c("dem", "rep")
colnames(srm.scores) <- c("dem", "rep")
srm.scores <- as.data.frame(srm.scores)
fa.scores <- as.data.frame(fa.scores)
m1 <- lm(rep ~ dem, data=fa.scores)
m2 <- lm(rep ~ dem, data=srm.scores)
```

```
##
## ============================================================
##                                     Dependent variable:
##                                 ----------------------------
##                                             rep
##                                      (1)            (2)
## ------------------------------------------------------------
## dem                                0.440***      0.400***
##                                    (0.018)        (0.018)
##
## Constant                            0.000         -0.000
##                                    (0.018)        (0.018)
##
## ------------------------------------------------------------
## Observations                        2,541          2,541
## R2                                  0.193          0.160
## Adjusted R2                         0.193          0.160
## Residual Std. Error (df = 2539)     0.898          0.917
## F Statistic (df = 1; 2539)        608.786***    484.175***
## ============================================================
## Note:                            *p<0.1; **p<0.05; ***p<0.01
```

Review

1. What are measurement models about?
2. Summated Ratings Scales
3. Exploratory Factor Analysis

Exercise

These data come from the 2019 CES - they relate to different attitudes about women (chauvenism) framed in both positive (benevolent) and negative (hostile) ways:

- `pes19_hostile1`: Most women fail to appreciate all that men do for them.
- `pes19_hostile2`: Women seek to gain power by getting control over men.
- `pes19_hostile3`: Most women interpret innocent remarks or acts as being sexist.
- `pes19_benevolent1`: Women should be cherished and protected by men.
- `pes19_benevolent2`: Many women have a quality of purity that few men possess.
- `pes19_benevolent3`: A good woman ought to be set on a pedestal by her man.

Each is scored on a five-point Likert scale from Strongly disagree to Strongly agree, with a value of 6 corresponding to Don't Know/NA. What is the structure of these data? What do you think is the best way to reduce the dimensionality of these data?

Questions

1. What is Cronbach's Alpha for these data? Do the assumptions we make from the summated rating model make sense?

```r
library(rio)
wom <- import("data/women_dat.dta")
```

1. What does the EFA look like for these data? How many dimensions does the model suggest?