*PSY 613, Spring 2021*

**Lab 4 - Principal Components Analysis (PCA) and Confirmatory Factor Analysis (CFA)**

**…and a little SEM ☺**

***Disambiguation of PCA and CFA***

PCA (Principal Components Analysis):

* Components are merely a way to summarize the relationships (correlations) between items; it’s data reduction, not causal explanation.
* You’re attempting to explain *all* *of the variance* in items.
* Descriptive/exploratory (usually not appropriate for theory testing). For example, it isn’t necessary to specify the number of factors beforehand – you can play around with it.

CFA (Confirmatory Factor Analysis):

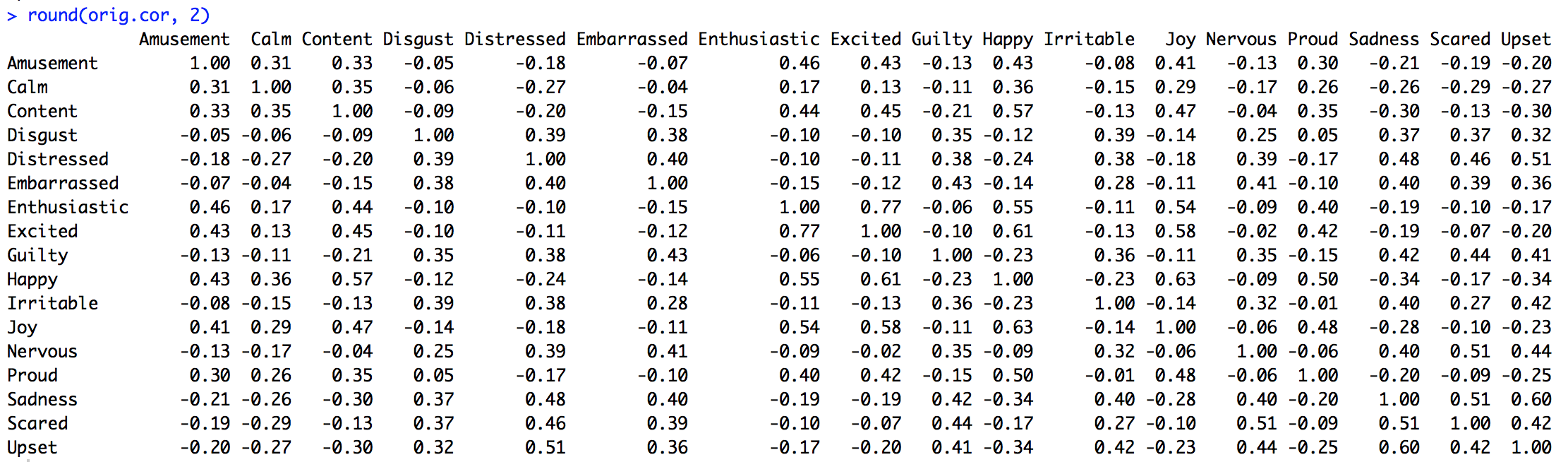
* Factors represent some underlying construct which *causes* the observable items (indicators).
* You’re attempting to explain *only the shared* *variance* between items; any variance unique to a particular item is considered to be its error.
* R has a great packing for SEM (lavaan), but it can also be done with a software package like Mplus.
* Theory-driven, tests *a priori* hypotheses (you must specify in advance how many factors there are, how they covary, and which items load onto which factors). Typically, you want to compare an entire model against another competing model to see which one fits the data better.

***Principal Components Analysis (in R)***

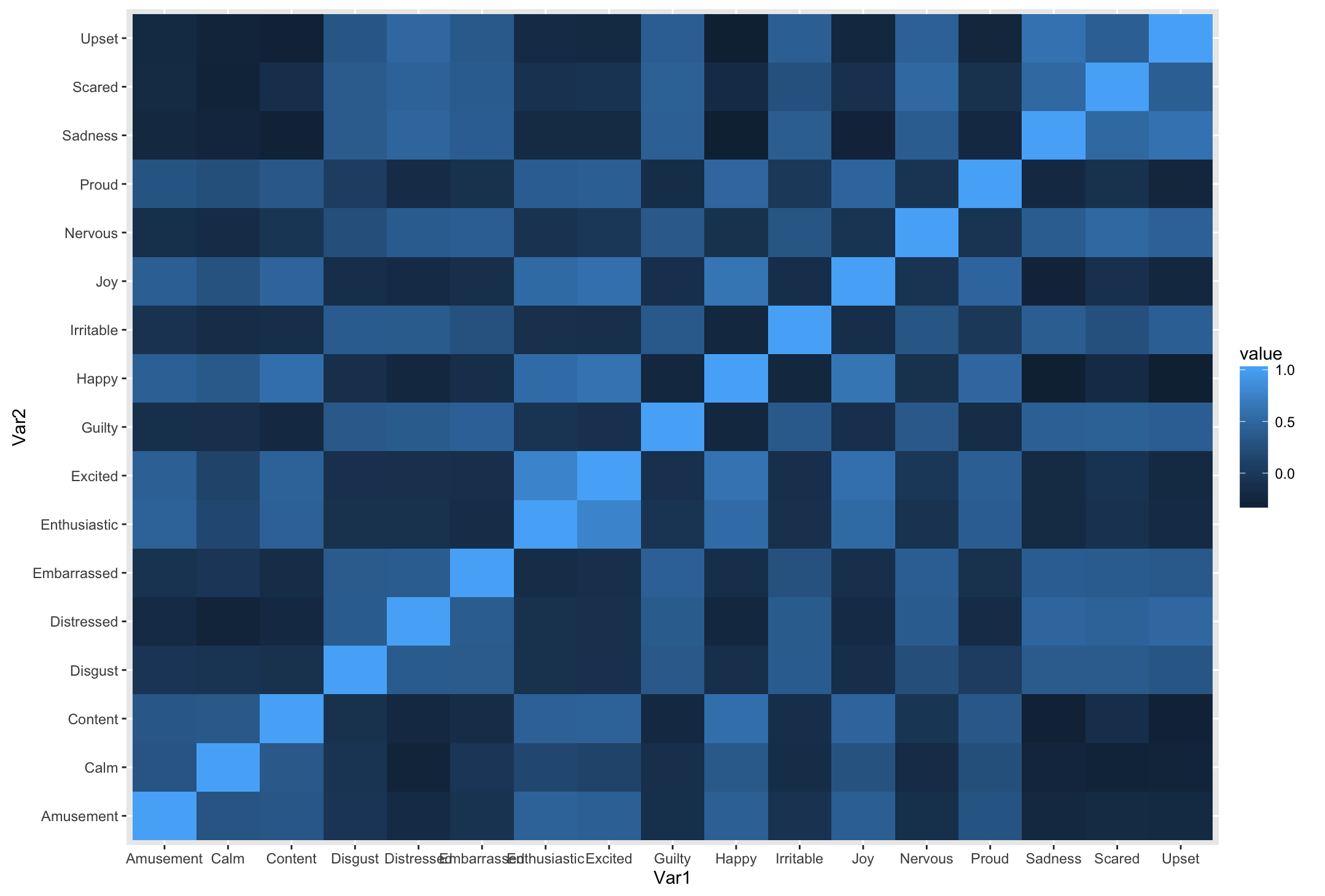
*The story:* A researcher has collected test data from 313 subjects on 17 different emotions. She wants to reduce the scale from 17 items to a fewer number of items. Run a principal components analysis on the data and interpret the results.

*Run a PCA without rotation (helpful for determining the number of components to extract):*

**Part 1: Get a Correlation Matrix**



*Heatmap of Correlations:*



Since the goal of principal components analysis is to explain the correlations among the items, each item should have moderate-large correlations with *at least* one of the other items.

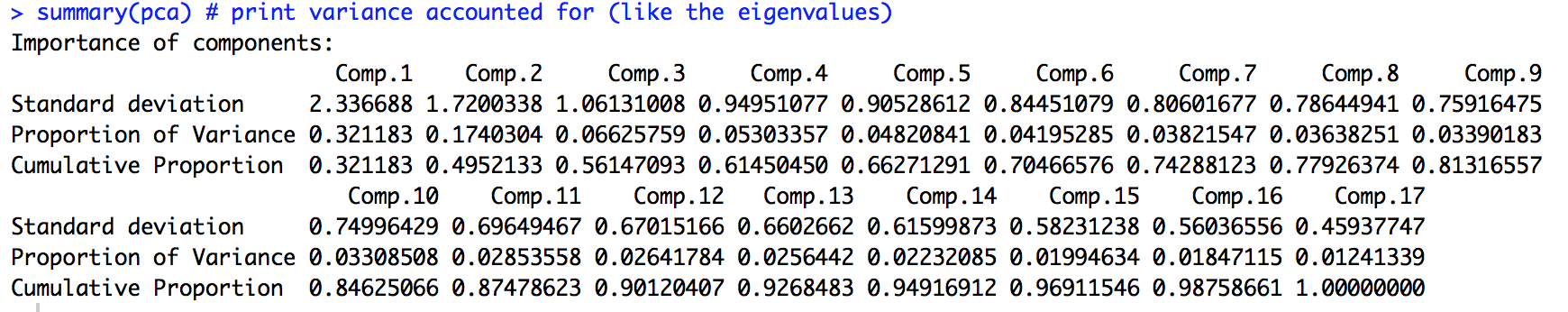
***—R demo: What does it mean to transform data?—***

There are a lot of ways to transform data. PCA uses the correlation matrix to find the optimal way to transform your data, if you want to reduce the number of variables/items you have.

**Part 2: Run PCA (unrotated) & explore components**

Explore variance explained by each component of the transformed data

*Note: this is in standard deviation units, so it’s the square root of the eigenvalue*

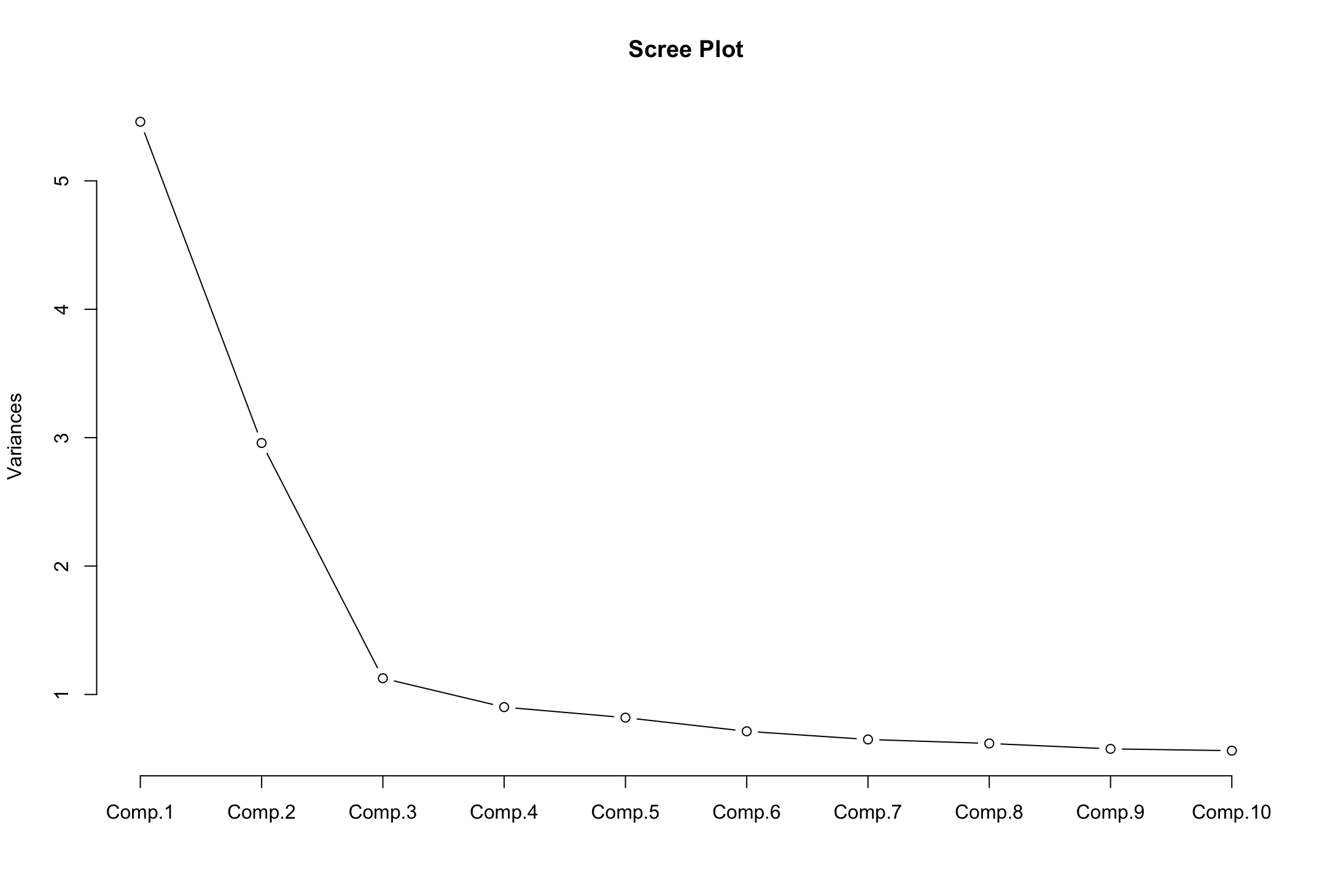


* First principal component: the linear combination of items that accounts for the largest amount of variance in the sample.
* Second principal component: accounts for the next largest amount of variance
  + If *varimax rotation* is selected, 2nd component is uncorrelated with the 1st
  + If an *oblique rotation (i.e., promax)* is selected, components are allowed to correlate.
* Successive components: explain smaller amounts of variance

***Fill in the blanks:*** Using the criteria of eigenvalue greater than 1 (i.e., the amount of total variance explained by a component should be greater than 1), the principle component analysis conducted by R extracted \_\_\_\_\_ components that account for \_\_\_\_\_ of the total variance.

*Question:* Why do we use the criteria of eigenvalue greater than 1?

*Answer:*

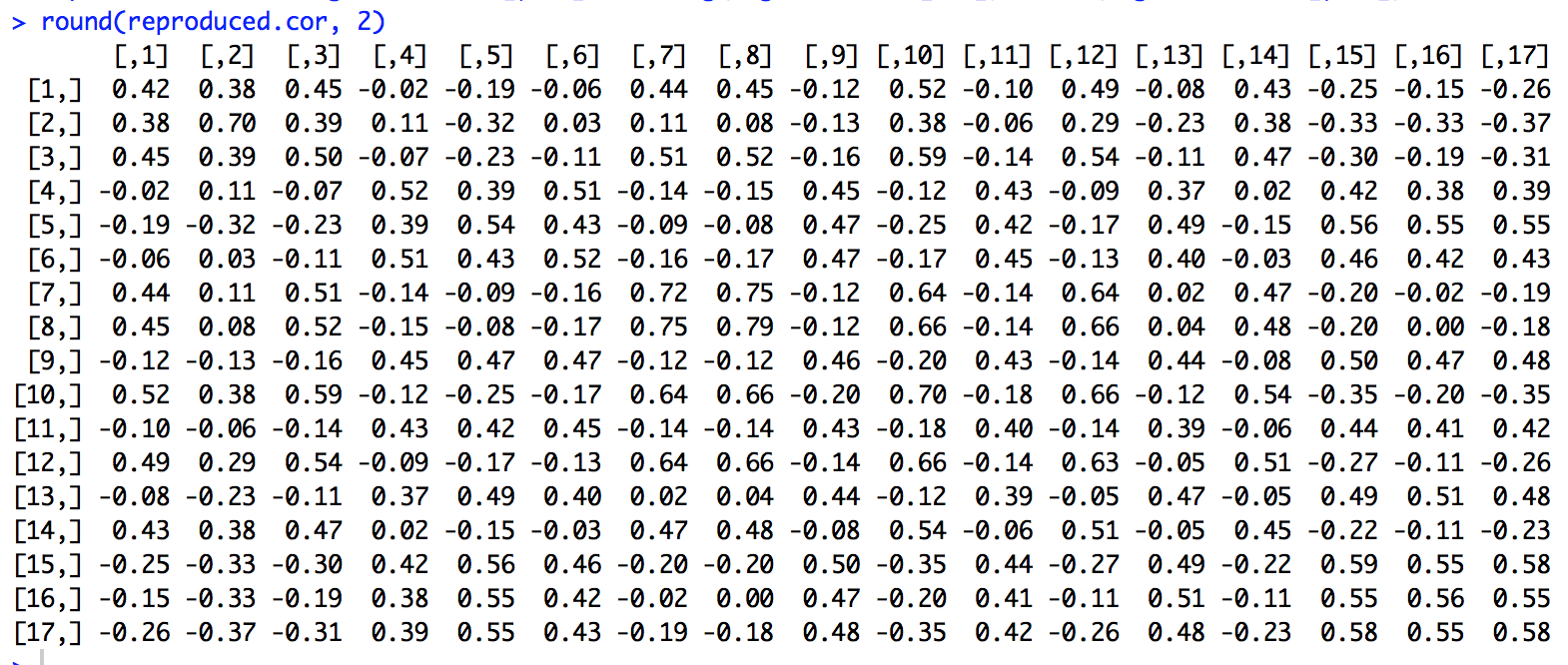


We can also look at the *Scree Plot* to help us determine the number of components to extract.

|  |
| --- |
| **Part 3: Explore loadings to understand your components**   * The loadings tell you how your original variables were combined to create each component. * *Example:* Component #1 = -.219\*Amusement – .196\*Calm – .253\*Content + .191\*Disgust +…. * In this case, it’s not clear what these components represent, but we’ll get to that soon     The total variance of a sample is equal to the sum of the variance of each item. Since each item is standardized to have a standard deviation (and hence variance) of 1, and there are 17 items in our sample, the total variance of our sample is 17.    These are factor (component) loadings between each item and the three components extracted by R **before** rotation. When the components are orthogonal (uncorrelated with each other), you can think of these component loadings as correlations between the items and components.  These component loadings are difficult to interpret for several reasons. First, an item may be moderately to highly correlated with more than one component. Second, since the first component explains the greatest proportion of the variance, it is expected that most of the items will exhibit the highest correlations with this component. |

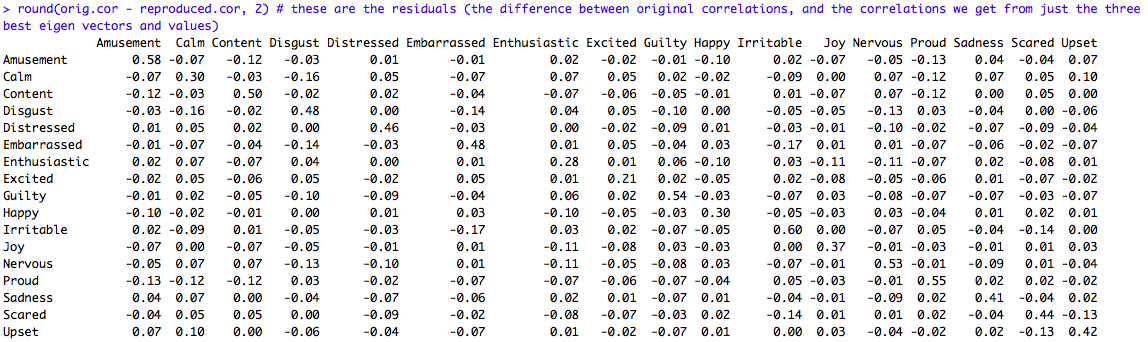
**Part 4: Compare reproduced & original correlation matrices to determine fit**

*Reproduced Correlation Matrix:*

 “Reproduced correlations” are ***estimated correlations between the items*** based on the ***correlations between items & the selected components***

If you haven’t selected enough components, your reproduced correlations won’t look like your original ones (you’ve lost too much info)

*Residual Correlation Matrix:*

**

The **difference between the observed and reproduced** correlation coefficients is the ***residual.***

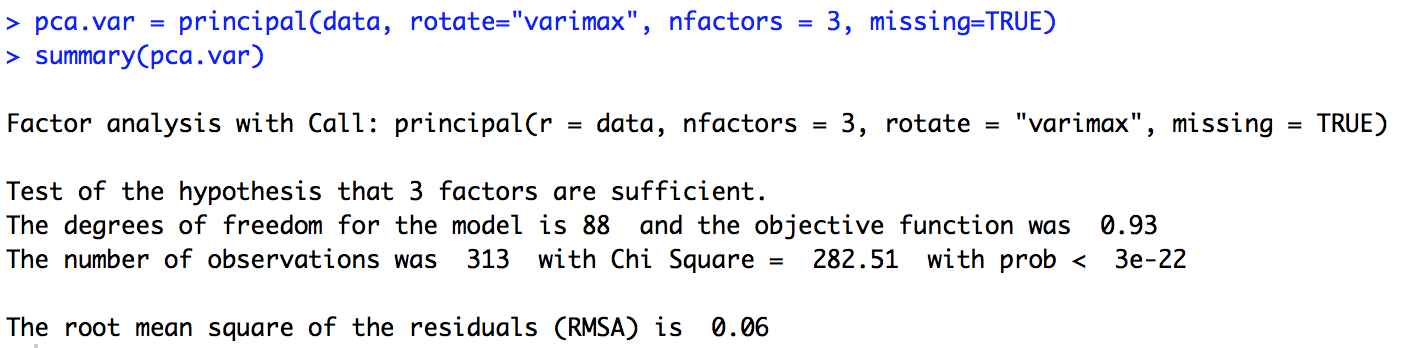
Example: Residual for *Amusement* and *Calm*: Observed (.31) – Reproduced (.38) = -.07

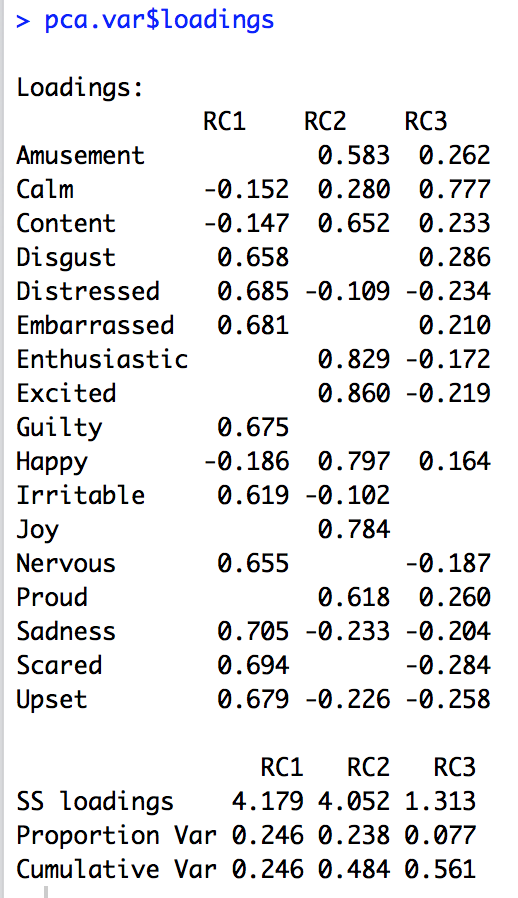
*Question:* What do small residuals (close to 0) indicate?

**Part 5: Run a PCA with varimax (orthogonal) rotation:**

*Question:* How many components should we extract?

*Output:*





These are the component loadings (simple correlations) between each item and the three components extracted by R **after** rotation. The purpose of rotating the component matrix is to achieve simple structure so we can better interpret the components.

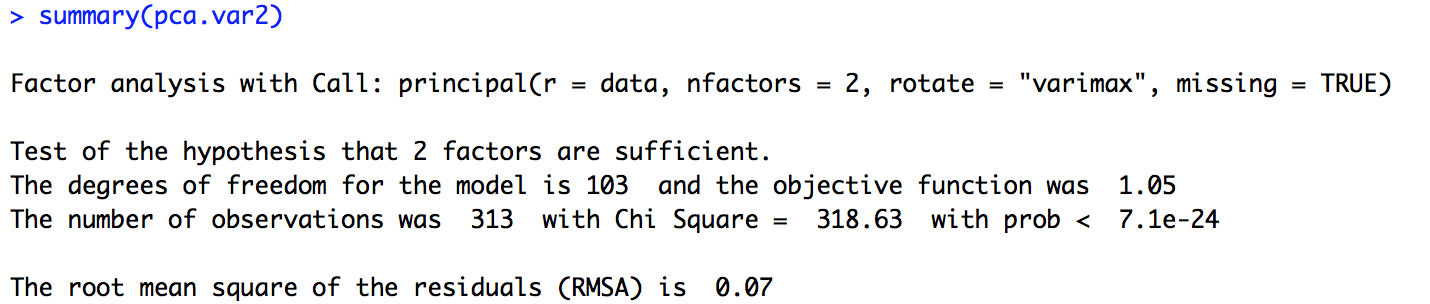
We used **varimax** rotation (an orthogonal rotation technique), which makes high loadings higher and low loadings lower.

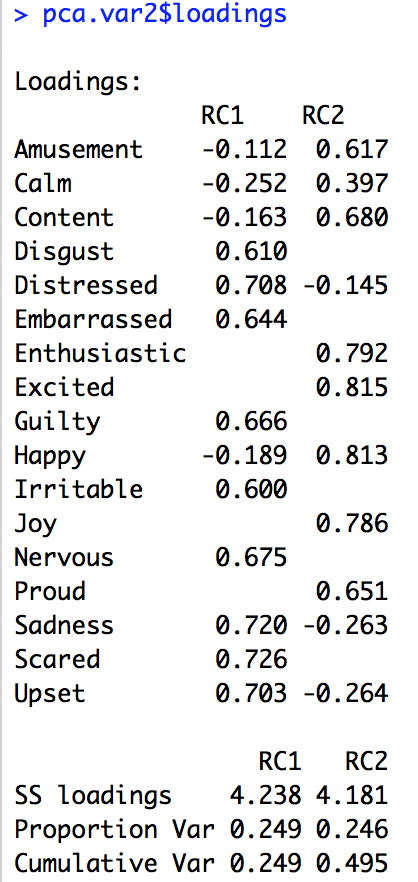
Summing the squared loadings for each item gives us that item’s communality.

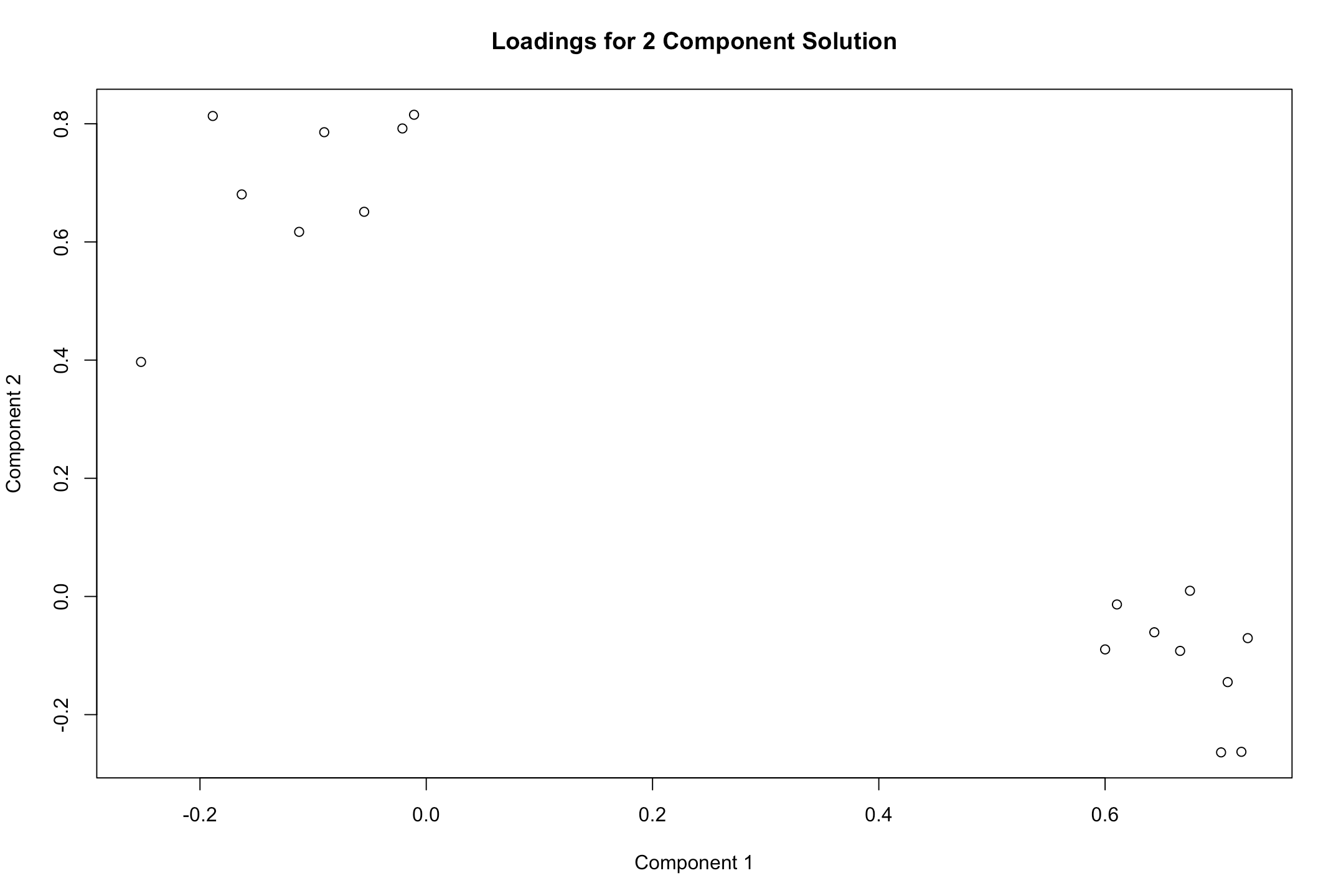
*Example (Calm):* -.1522 + .2802 + .7772 = .659

From this table, we see that component 1 represents negative feelings (i.e., disgust, distressed, embarrassed, guilty, irritable, nervous, sadness, scared, and upset), component 2 represents positive feelings (i.e., amusement, content, enthusiastic, excited, happy, joy, and proud), and component 3 represents calmness.

*Question:* What if we forced R to extract only two components?







All of the ***positive feelings*** (now including calm) load highly on component 2 (loadings range from .4 to.8)

All of the ***negative feelings*** load highly on component 1

(loadings range from .5 to .7)

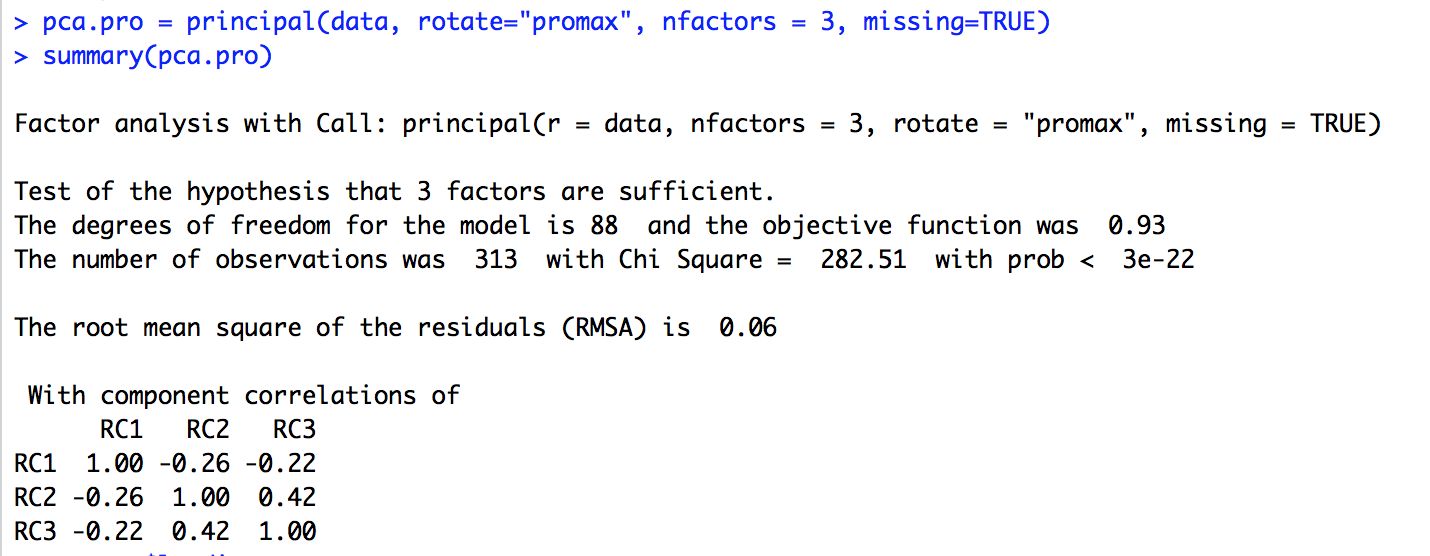
Amount of variance explained by our two components.

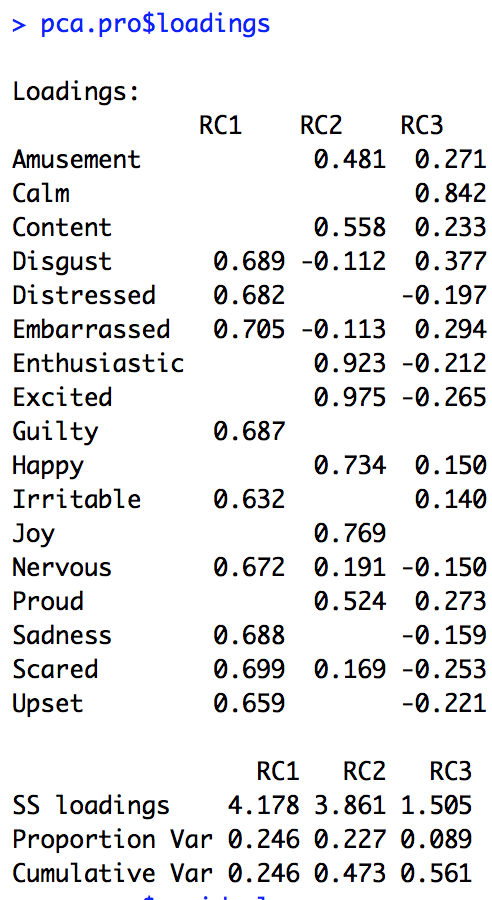
The above analyses used orthogonal rotation (varimax) – meaning the components were not allowed to covary with one another. Would a solution in which we allow the components to correlate be better? Let’s do an oblique rotation (promax) to find out.

**Part 6: PCA with oblique rotation**

Compared to PCA with varimax rotation, the output for the PCA with promax rotation will be the same except:

1. The component loadings after rotation will differ
2. The components will now correlate (the correlation(s) are given in the summary)





*Question:* Based on the above output, should we be using an orthogonal or oblique rotation?

***Structural Equation Modeling (SEM)***

Terminology/Notation:

* A variable is ***exogenous*** if its causes are not represented in the model – therefore, all exogenous variables are free to vary, which is represented by the following symbol:
* A variable is ***endogenous*** if its causes are represented in the model – these variables have an error term that is free to vary, which is represented by the following symbol:

e1

Indicator

* ***Indicators*** are represented as boxes

Disturbance

Factor

* ***Factors***are represented as circles or ovals
* Latent factor variances are called ***disturbances.***

Characteristics of CFA:

1. Each indicator (rectangle) has two causes
   1. A ***single*** factor
   2. All other sources of influence (omitted variables and measurement artifacts) represented by an error term
2. The factors (ovals) covary. In CFA, typically the only factors included are the ones that directly cause indicators. If you have a bigger model with more latent factors influencing each other, then you would refer to that as a structural model (SEM) rather than confirmatory factor analysis. CFA is only one part of SEM; models describing how latent factors cause one another fall outside CFA.
3. The errors are independent of each other and of the factors.
4. In addition to specifying the number of factors in a standard CFA you also specify indicator-factor correspondence.

*Standard CFA model:*

e3

1

e1

1

e6

1

e5

e4

1

1

e2

1

I5

I1

I2

I3

I4

I6

1

B

A

1

*Q:* In the above measurement model, what variables are exogenous? Which ones are endogenous?

*Running CFA in R*

*The story:* A researcher has collected test data from 1000 subjects on three different abilities: visual-spatial ability, short-term memory, and social-cognition. She has 4 different tests for each ability (so each participant completed 12 tasks). She’s got all kinds of fun ideas about how these abilities relate to eachother, but before diving into complicated model stuff, she just wants to see how well her items load onto the factors they’re supposed to measure. She goes to Elliot’s office hours, and he tells her to run CFA.

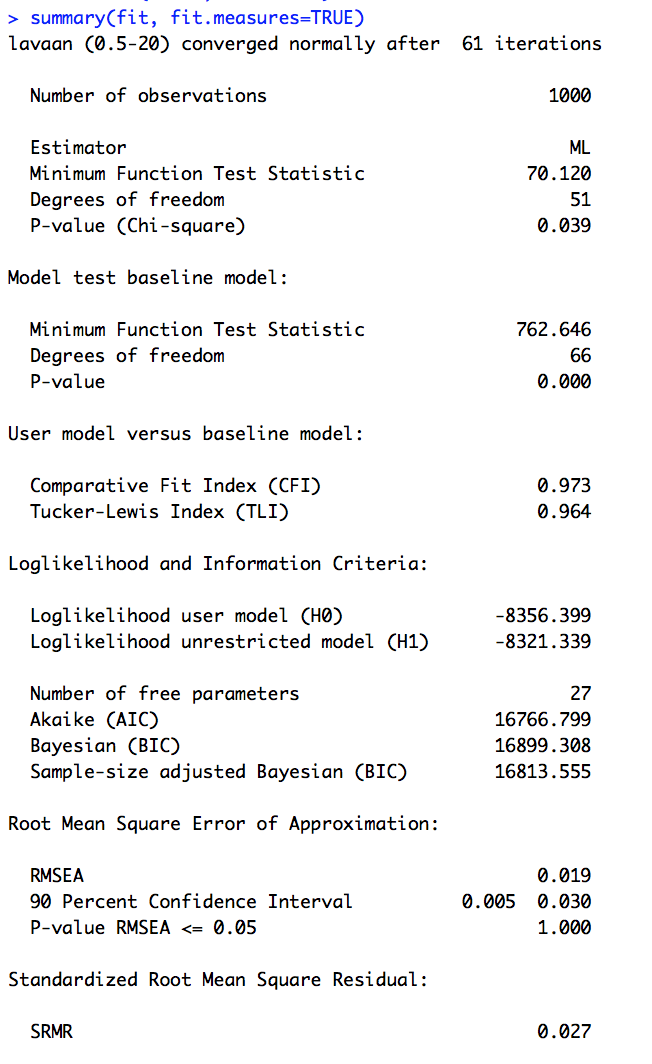
So we have three latent factors: visual-spatial ability (VS), short-term memory (STM), and social-cognition (SC). Run a basic CFA. That means each indicator loads on 1 and only 1 factor, and the factors are allowed to covary with each other.

First, draw your model:

Run it (see R script and lavaan tutorial at <http://lavaan.ugent.be/tutorial/cfa.html>)

Output: Fit & Parameter estimates

*Does the model fit well?*

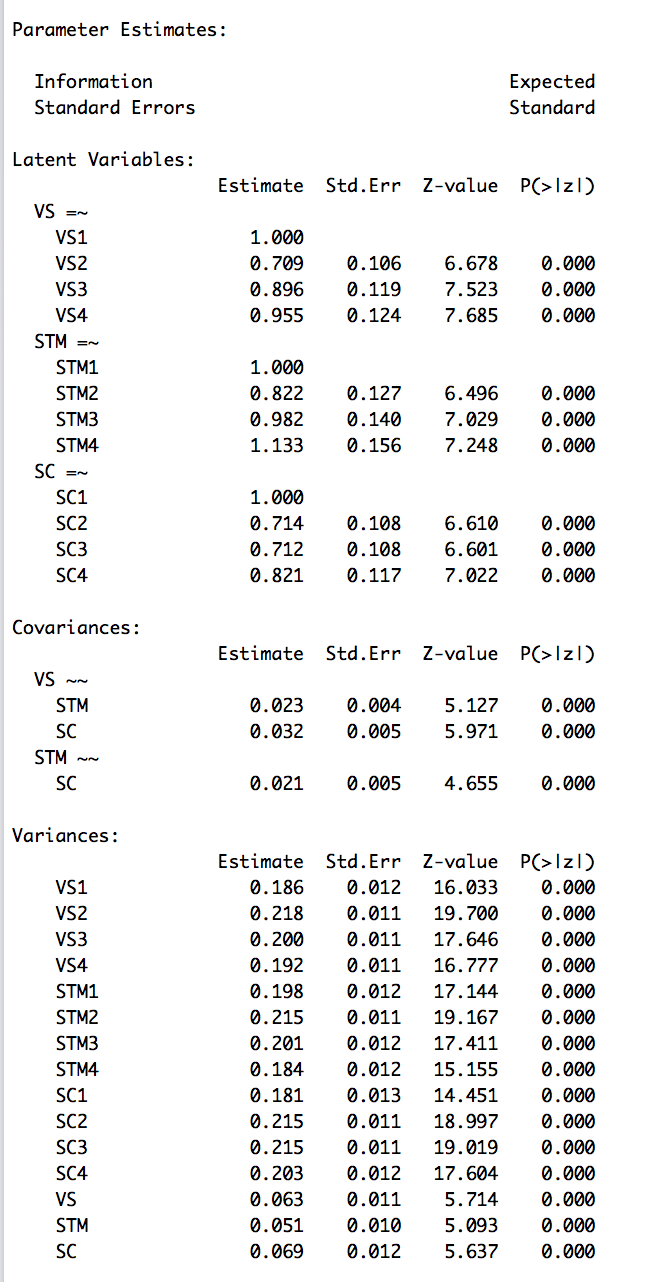


R gives you lots of different fit statistics, and they each tell you slightly different information. See this webpage for help interpreting common measures of model fit: <http://davidakenny.net/cm/fit.htm>

For example, RMSEA is probably the most popular one. Smaller is better (smaller = less error), and .05 has been used as a *rough* cutoff for a “good” fit.

Standardized Root Mean Square Residual is handy (and very interpretable!). It’s the average difference between your observed and reproduced correlations.

*What are the parameter estimates?*



Sanity check: How many parameters were we expecting?

Notice that R automatically set the first indicator’s path to 1 for each latent variable. Why?

Write the estimates (with SEs) for the paths and covariances into the model you drew before.

From the model, what is the direct path (pattern correlation) between SC3 and the latent social cognition factor?

What is the direct path (pattern correlation) between SC3 and the latent short-term memory factor?

What is the total path (structure correlation) between SC3 and the latent

short-term memory factor? *.712\*.021 = .0142*

What is the total path (structure correlation) between SC3 and the latent short-term memory factor?

What is the total path between SC3 and VS1? (Note, this is what shows up in the “reproduced correlation matrix”.)

How about between VS2 and STM1?

*Reminders about path tracings:*

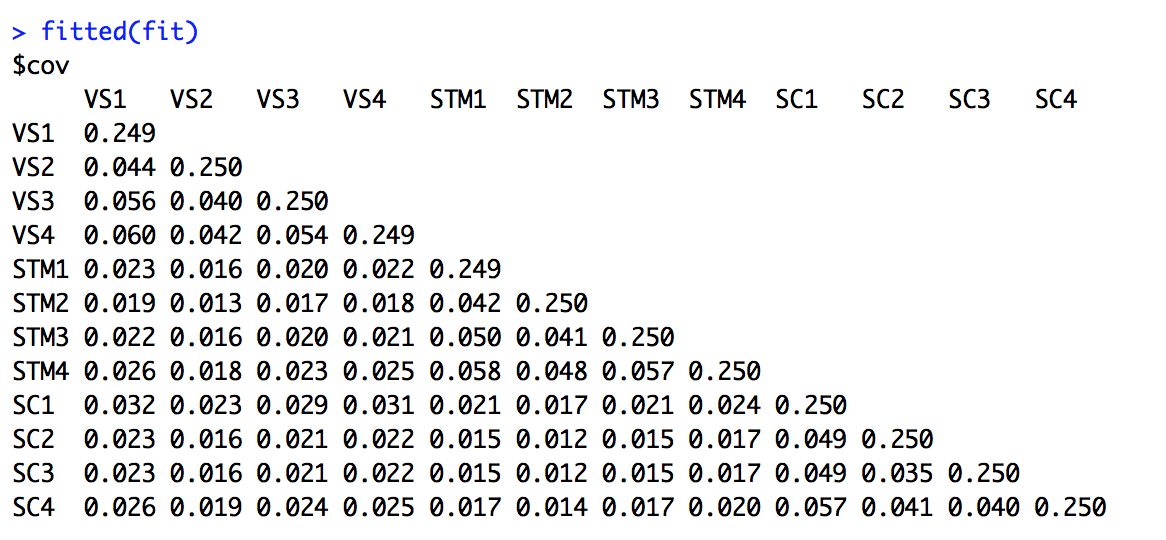
To calculate implied correlations from your model, use tracings. A tracing is a path that...

* does not go through the same variable twice
* does not enter and leave the same variable through two arrow heads

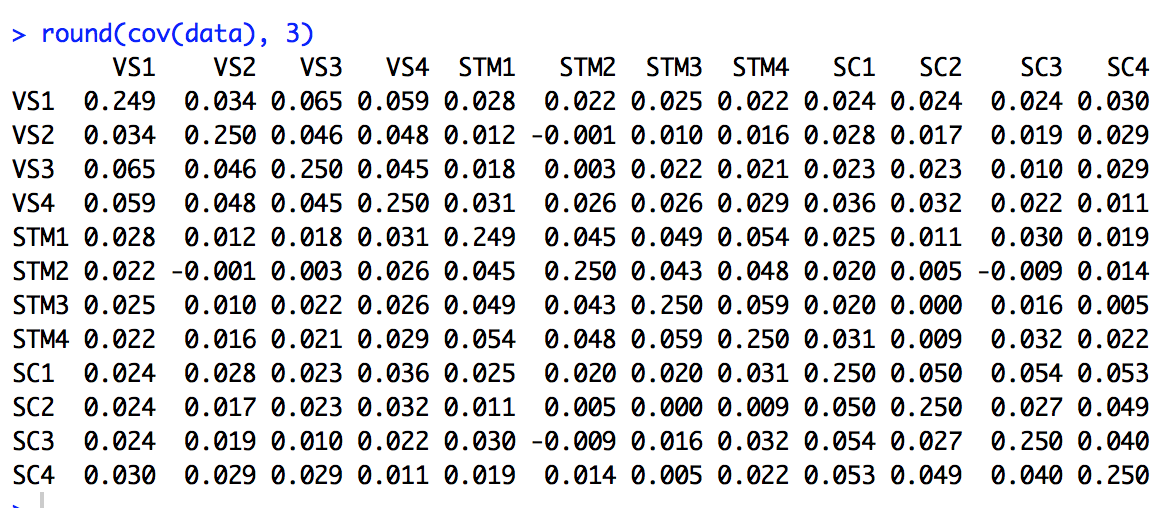
Note: You don't have to follow the direction of arrows (it’s okay to go the “wrong” way on an arrow).

A tracing can be direct (one arrow) or compound (more than one arrow). For compound tracings, you multiply the weights of each path.

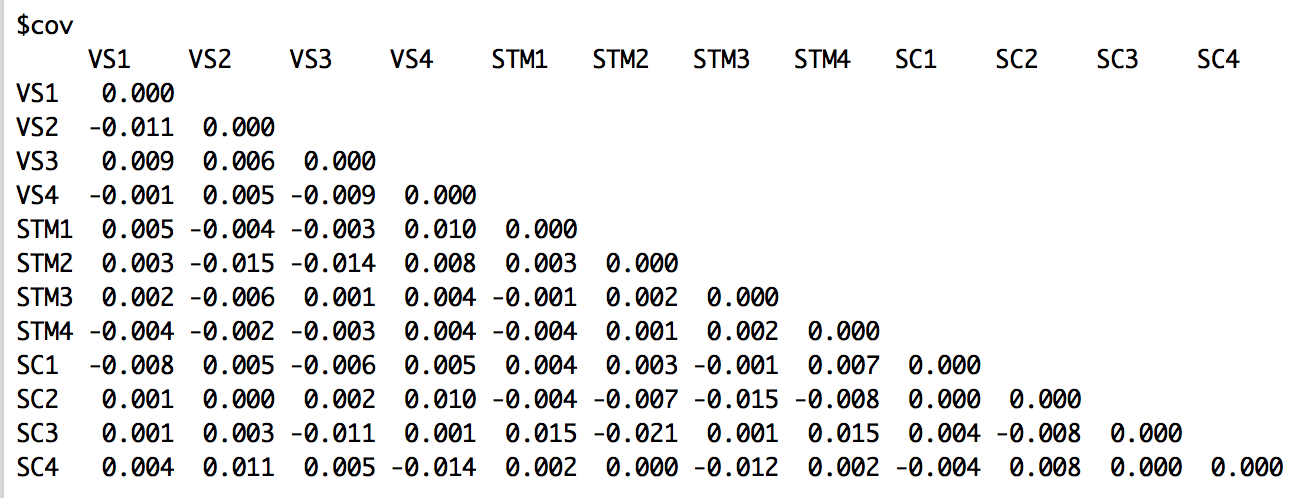
*What are the reproduced/implied correlations among the indicators?*



*How do they compare to the original/observed covariances?*



*Here are the residuals (original – reproduced):*



Do any stick out as larger than the others?