Exploratory Factor Analysis Notes

Francis Huang

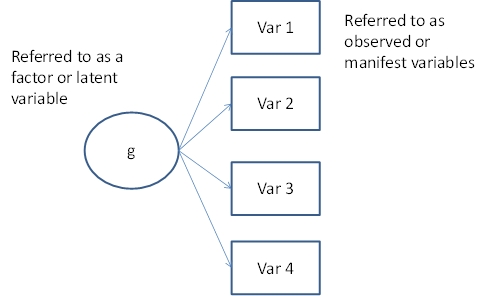
February 12, 2017

### 1. Introduction to EFA

Referred to as an interdependence technique (in contrast to a dependence technique which has DVs and IVs). Nice extension from last week's' discussion on PCA. EFA is used to generate theory "identify the factor structure or model for a set of variables" (Bandalos, 1996).

* "define the underlying structure among variables in the analysis" (Hair et al., 2009)
* Concept can be traced to Charles Spearman's (1904): research on human intellect
* Theorized that each measure of human intelligence can be explained by a general factor, common to all other measures of ability. Measures are correlated primarily through the shared measure (g or the general intelligence factor).

For example:



One factor model

* Remember, with a PCA, the components were a linear combination of the observed variables (arrows pointing from circles to the component).
* With an EFA, the manifest variables are linear combination of the factor.
* Latent constructs or factors are thought to 'cause and summarize responses to observed variables.' (Henson & Roberts, 2006).
* An attempt to summarize several j observed variables into a fewer set of k factors ('researchers commonly attempt to explain the most with the least' (HR, 2006).
* Results in a 'factor' and makes use of the concept of a variate, a linear combination of variables (think of regression weights) formed to maximize the explanation of the entire variable set.

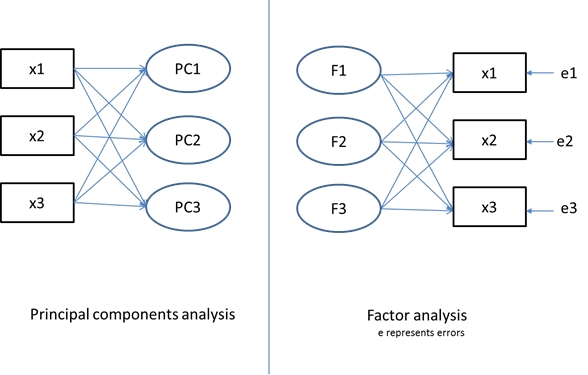
Two main purposes (Hair et al., 2009)

1. Data summarization: investigates the underlying dimensions and structure of the data that when interpreted, describe the data in a smaller number of concepts compared to the original variables (e.g., academic abilities affects math, science, and language skills). Identifying the dimensions or factors is the end in itself.
2. Data reduction: identify a smaller set of variables to use in subsequent analysis or create an entirely new set of variables (variates/factors/components). Identifying the factors for subsequent analysis [sounds more like PCA though]

Practically speaking, sometimes, this distinction is not as clear or as important. In either case, it is reducing a large number of variables with less.

### 2. Differences with a PCA and EFA

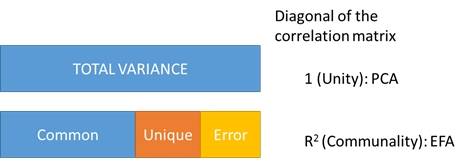
A. EFA is used to determine the underlying factors or dimensions that give an effect to the manifest variables *this is also why the direction of the arrows differ*. PCA is used when the objective is to reduce the number of variables into a fewer number of components for prediction purposes.



Difference in graphical depiction of PCA vs EFA

B. PCA uses TOTAL variance. EFA (which is also why it is called common factor analysis)-uses variability that is ONLY common to all variables. Communalities (the multiple R2) instead of unities (1s) are inserted into the diagonal of the correlation matrix analyzed.

* Variance is partitioned to COMMON (the R2) + UNIQUE (something that is not COMMON really) + ERROR (sometimes scores just vary based on something else) variance



Differences between PCA and EF: Suhr <http://www2.sas.com/proceedings/sugi30/203-30.pdf>

With PAF, the 1s are replaced with the estimates of communality or that proportion of variance in variable X accounted for by all the other variables in the dataset (regression X by all other variables and get the R2 and stick that in the diagonal) - unlike PCA which includes common variance accounted for + unique variance (1- R2).

dat <- read.csv("http://faculty.missouri.edu/huangf/data/mvnotes/READING120n.csv") #from last week  
dat <- dat[,-1] #removed the first gender column  
(Rm <- round(cor(dat), 3))

## rhyme Begsnd ABC LS Spelling COW  
## rhyme 1.000 0.616 0.499 0.677 0.668 0.693  
## Begsnd 0.616 1.000 0.285 0.347 0.469 0.469  
## ABC 0.499 0.285 1.000 0.796 0.589 0.598  
## LS 0.677 0.347 0.796 1.000 0.758 0.749  
## Spelling 0.668 0.469 0.589 0.758 1.000 0.767  
## COW 0.693 0.469 0.598 0.749 0.767 1.000

This is the standard correlation matrix we used for the PCA: 1's were on the diagonals. However, in an EFA, we use a reduced correlation matrix where the diagonals represent the variance account for by the OTHER variables. The variance accounted for is the squared multiple correlation (SMC or R2).

In general, the uncorrelated factor equation being solved is:

represents the factor loadings = eigenvectors %\*% sqrt(evalues) where the evalues is a square matrix with the eigenvalues on the diagonal. R represents the correlation matrix.

represents the matrix of communalities on the diagonal (with 0s in the off diagonal). In a PCA, these communalities are 1s. In an EFA, the communalities are represented by the SMCs.

Equivalently, in an EFA, an attempt is made to solve for the following equation:

The left side of the equation is then the reduced correlation matrix.

m1 <- lm(rhyme ~ . , data = dat)  
#NOTE: the . is a shortcut indicating all other variables  
summary(m1)

##   
## Call:  
## lm(formula = rhyme ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -5.0470 -0.9528 0.2083 1.2369 3.6783   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.52038 0.69405 2.191 0.030517 \*   
## Begsnd 0.39851 0.06989 5.702 9.48e-08 \*\*\*  
## ABC -0.04937 0.03952 -1.249 0.214127   
## LS 0.16783 0.04762 3.525 0.000612 \*\*\*  
## Spelling 0.04291 0.04896 0.877 0.382576   
## COW 0.08755 0.03983 2.198 0.029978 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.797 on 114 degrees of freedom  
## Multiple R-squared: 0.6542, Adjusted R-squared: 0.6391   
## F-statistic: 43.14 on 5 and 114 DF, p-value: < 2.2e-16

#NOTE: the R2 = .65

The other five variables account for 65% of the variance in the rhyme outcome variable (look at the R2). To get the R2s for all variables, we can use:

(h2 <- diag(1 - (1/solve(Rm))))

## rhyme Begsnd ABC LS Spelling COW   
## 0.6542386 0.4219772 0.6396063 0.8059539 0.6836182 0.6842471

Rreduced <- Rm  
diag(Rreduced) <- h2  
round(Rreduced, 3)

## rhyme Begsnd ABC LS Spelling COW  
## rhyme 0.654 0.616 0.499 0.677 0.668 0.693  
## Begsnd 0.616 0.422 0.285 0.347 0.469 0.469  
## ABC 0.499 0.285 0.640 0.796 0.589 0.598  
## LS 0.677 0.347 0.796 0.806 0.758 0.749  
## Spelling 0.668 0.469 0.589 0.758 0.684 0.767  
## COW 0.693 0.469 0.598 0.749 0.767 0.684

NOTE: this is now the **REDUCED** correlation matrix. This is what is factor analyzed.

Only variance shared with other variables is accounted for:

* "Many factor analysts would not consider PCA as factor analysis at all" (HR, 2006). NOTE: when using the dimension reduction command in SPSS, it defaults to PCA which is probably why many researchers use that option! Easy to see when EFA and PCA may be most similar, when the error variance on the diagonal is low (or the diagonal is very close to one).

PCA is a dimension reduction tool primarily (latent constructs are not really the focus of the analysis). FA using PAF focuses on the common variance. Thompson (1992) argues that differences though between the two are not large (well, the closer the diagonals are to 1 with PAF, the results will be very similar). See notes of PCA session.

NOTE: the different methods may not necessarily suggest the same number of factors!

### 3. Performing an EFA

Using the same dataset, with EFA, slightly different conclusions may still be reached depending on the way EFA is performed! EFA is driven by the "mechanics and mathematics of the method" (Kieffer, 1999). No theory needed? take 'what the data give you' (Hair et al., 2009).

* Most statistical packages use the correlation matrix as the default and most researchers subsequently use the correlation matrix as well (HR, 2006). Not an issue if you have the raw data (because the software generates the correlation matrix anyway). If you run an EFA with a correlation or a covariance matrix, results **will** differ.

#load("c:/users/huangf/documents/efamatrix.rdata")  
con <- url("http://faculty.missouri.edu/huangf/data/mvnotes/efamatrixbook.rdata")  
load(con)  
  
#creates matrix efam  
efam #a 12 x 12 matrix

## TEN1 TEN2 TEN3 WOR1 WOR2 WOR3 IRTHK1 IRTHK2 IRTHK3 BODY1 BODY2  
## TEN1 1.00 0.66 0.65 0.28 0.29 0.36 0.08 0.00 0.03 0.29 0.36  
## TEN2 0.66 1.00 0.66 0.34 0.33 0.46 0.09 0.04 0.10 0.31 0.38  
## TEN3 0.65 0.66 1.00 0.30 0.35 0.44 0.12 0.10 0.10 0.46 0.49  
## WOR1 0.28 0.34 0.30 1.00 0.64 0.66 0.32 0.31 0.30 0.27 0.26  
## WOR2 0.29 0.33 0.35 0.64 1.00 0.57 0.31 0.30 0.34 0.31 0.28  
## WOR3 0.36 0.46 0.44 0.66 0.57 1.00 0.37 0.33 0.31 0.35 0.37  
## IRTHK1 0.08 0.09 0.12 0.32 0.31 0.37 1.00 0.61 0.67 0.12 0.20  
## IRTHK2 0.00 0.04 0.10 0.31 0.30 0.33 0.61 1.00 0.70 0.14 0.19  
## IRTHK3 0.03 0.10 0.10 0.30 0.34 0.31 0.67 0.70 1.00 0.18 0.20  
## BODY1 0.29 0.31 0.46 0.27 0.31 0.35 0.12 0.14 0.18 1.00 0.37  
## BODY2 0.36 0.38 0.49 0.26 0.28 0.37 0.20 0.19 0.20 0.37 1.00  
## BODY3 0.44 0.41 0.52 0.32 0.28 0.38 0.17 0.16 0.10 0.46 0.48  
## BODY3  
## TEN1 0.44  
## TEN2 0.41  
## TEN3 0.52  
## WOR1 0.32  
## WOR2 0.28  
## WOR3 0.38  
## IRTHK1 0.17  
## IRTHK2 0.16  
## IRTHK3 0.10  
## BODY1 0.46  
## BODY2 0.48  
## BODY3 1.00

To make a reduced correlation matrix, we need to put the SMC in the diagonals

(h2 <- diag(1 - (1/solve(efam))))

## TEN1 TEN2 TEN3 WOR1 WOR2 WOR3 IRTHK1   
## 0.5339805 0.5626029 0.6126476 0.5491588 0.4807034 0.5666763 0.5168444   
## IRTHK2 IRTHK3 BODY1 BODY2 BODY3   
## 0.5478043 0.6082119 0.3212328 0.3417454 0.4175286

Rreduced <- efam  
diag(Rreduced) <- h2

NOTE: this is from the book. See table 9.7. We have 12 variables. How many factors should we extract? Often referred to as one of the most important questions in FA. Too much or too little may not be good (results in failure to reproduce in other datasets).

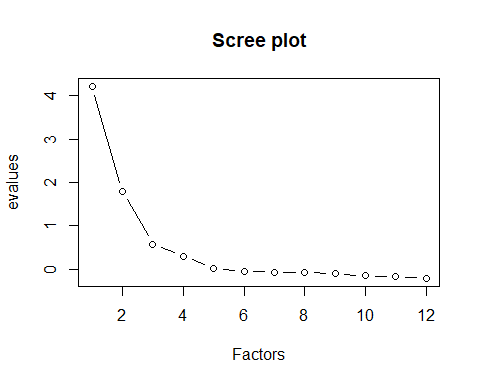
Kaiser's rule that we discussed in the context of PCA is for PCA only (i.e., the eigenvalue > 1). **That is not applicable for a reduced correlation matrix.** At times, others may use a PA in reference to the unreduced correlation matrix (i.e., the original matrix) but then perform a factor analysis. Some may say that this is a contradiction (Fabrigar & Wegener) as the number of factors may differ.

Can still use a scree plot which is very subjective and often not recommended as a sole reason for determining the number of factors to extract. From Google: n. a mass of small loose stones that form or cover a slope on a mountain.

(evalues <- eigen(Rreduced)$values)

## [1] 4.21342747 1.78678709 0.57947874 0.30025512 0.01574536  
## [6] -0.04635722 -0.06900068 -0.08280248 -0.09661914 -0.15682151  
## [11] -0.17184064 -0.21311500

plot(evalues, type = 'b', main = "Scree plot", xlab = 'Factors')



#type = 'b' specifies that the points and the lines are drawn

Hard to say? Say three or four meaningful factors? Parallel analysis is a good way to determine the number of factors to retain.

Horn's (1965) parallel analysis (PA):

1. FA will summarize correlations if no factors truly exist in the data
2. Imagine an equally complicated dataset with the same number of cases, same number of variables, no true correlation in the data (i.e., garbage)
3. How would factors look if only random noise were included in the data?
4. PA provides a benchmark-what if we were examining garbage? We would still come up with x number of factors
5. Retained components should be larger than in situations where there were no true factor (just randomness)
6. Several methodologists agree that this is the most accurate technique (there is another procedure referred to as MAP but is referred to for PCA not for EFA)

library(hornpa) #I think the sample size is 318??  
hornpa(k = 12, size = 318, reps = 500, method = 'pa')

##   
## Parallel Analysis Results   
##   
## Method: pa   
## Number of variables: 12   
## Sample size: 318   
## Number of correlation matrices: 500   
## Percentile: 0.95   
##   
## Compare your observed eigenvalues from your original dataset to the 95 percentile in the table below generated using random data. If your eigenvalue is greater than the percentile indicated (not the mean), you have support to retain that factor/component.   
##   
## Factor Mean 0.95  
## 1 0.370 0.468  
## 2 0.277 0.348  
## 3 0.211 0.265  
## 4 0.151 0.198  
## 5 0.095 0.137  
## 6 0.044 0.085  
## 7 -0.004 0.032  
## 8 -0.051 -0.013  
## 9 -0.096 -0.059  
## 10 -0.143 -0.108  
## 11 -0.190 -0.152  
## 12 -0.249 -0.210

evalues

## [1] 4.21342747 1.78678709 0.57947874 0.30025512 0.01574536  
## [6] -0.04635722 -0.06900068 -0.08280248 -0.09661914 -0.15682151  
## [11] -0.17184064 -0.21311500

The four eigen values are higher than the 95% percentile found in the simulated datasets. Supports the retention of four factors. The fifth eigenvalue is .015, which is lower than ~.13. Eigen values 6 to 12 are also negative which can happen with reduced correlation matrices.

Can now use the psych package and the function fa. #you may also have to install a package 'GPArotation' if it is not already installed.

library(psych)  
  
efa1 <- fa(efam, n.obs = 318, nfactors = 4, fm = 'pa', rotate = 'oblimin' )

## Loading required namespace: GPArotation

efa1

## Factor Analysis using method = pa  
## Call: fa(r = efam, nfactors = 4, n.obs = 318, rotate = "oblimin", fm = "pa")  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## PA2 PA1 PA3 PA4 h2 u2 com  
## TEN1 -0.03 0.79 -0.01 0.02 0.64 0.36 1.0  
## TEN2 0.01 0.84 0.07 -0.06 0.70 0.30 1.0  
## TEN3 0.01 0.59 -0.03 0.35 0.72 0.28 1.6  
## WOR1 -0.04 -0.04 0.92 -0.02 0.78 0.22 1.0  
## WOR2 0.08 0.05 0.65 0.04 0.54 0.46 1.0  
## WOR3 0.09 0.14 0.61 0.10 0.63 0.37 1.2  
## IRTHK1 0.74 0.03 0.05 -0.02 0.59 0.41 1.0  
## IRTHK2 0.78 -0.10 0.03 0.07 0.64 0.36 1.1  
## IRTHK3 0.90 0.03 -0.03 -0.04 0.77 0.23 1.0  
## BODY1 0.00 -0.06 0.10 0.61 0.39 0.61 1.1  
## BODY2 0.11 0.11 -0.01 0.53 0.41 0.59 1.2  
## BODY3 -0.02 0.04 0.04 0.69 0.54 0.46 1.0  
##   
## PA2 PA1 PA3 PA4  
## SS loadings 2.06 1.94 1.83 1.52  
## Proportion Var 0.17 0.16 0.15 0.13  
## Cumulative Var 0.17 0.33 0.49 0.61  
## Proportion Explained 0.28 0.26 0.25 0.21  
## Cumulative Proportion 0.28 0.54 0.79 1.00  
##   
## With factor correlations of   
## PA2 PA1 PA3 PA4  
## PA2 1.00 0.09 0.45 0.24  
## PA1 0.09 1.00 0.44 0.65  
## PA3 0.45 0.44 1.00 0.45  
## PA4 0.24 0.65 0.45 1.00  
##   
## Mean item complexity = 1.1  
## Test of the hypothesis that 4 factors are sufficient.  
##   
## The degrees of freedom for the null model are 66 and the objective function was 5.58 with Chi Square of 1741.38  
## The degrees of freedom for the model are 24 and the objective function was 0.11   
##   
## The root mean square of the residuals (RMSR) is 0.02   
## The df corrected root mean square of the residuals is 0.03   
##   
## The harmonic number of observations is 318 with the empirical chi square 9.92 with prob < 0.99   
## The total number of observations was 318 with Likelihood Chi Square = 33.04 with prob < 0.1   
##   
## Tucker Lewis Index of factoring reliability = 0.985  
## RMSEA index = 0.001 and the 90 % confidence intervals are NA 0.061  
## BIC = -105.25  
## Fit based upon off diagonal values = 1  
## Measures of factor score adequacy   
## PA2 PA1 PA3 PA4  
## Correlation of scores with factors 0.93 0.93 0.93 0.88  
## Multiple R square of scores with factors 0.87 0.86 0.86 0.78  
## Minimum correlation of possible factor scores 0.75 0.73 0.73 0.56

#this specifies:  
#dataset, number of observations (when a correlation matrix is used), number of factors, factor method (pa = principal axis factoring or PAF), rotation method

Rotation is used to make sense of the factors in an attempt to find simple structure. By default, a rotation will be specified. Most often, rotation will be used to make sense of the loadings. Can compare if rotate = 'none' is used. Can determine if oblimin (oblique) or varimax (orthogonal) is to be used. Often more realistic to specify correlated. If the correlation among factors is low (e.g., < .20), the using Varimax is reasonable.

Some researchers just recommend starting off with an oblique rotation (correlated factors). If the correlation between factors is not too small (e.g., r > .32), then proceed. If the correlations are very high (e.g., r > .80), then this may suggest reducing the number of factors too since the factors may be measuring the same thing with such high correlations. NOTE: This is the default of the fa function too.

Given all the choices, which should you run? Tabachnick & Fidell recommend starting with an oblique rotation (correlated). If the correlation is .32 or higher, then the correlation is not negligible and may be better off running an oblique rotation.

PRACTICAL ADVICE: Kim and Mueller (1978, p. 50): "We advise that beginners choose one of the commonly available methods of rotation, such as Varimax if orthogonal rotation is sought or Direct Oblimin if oblique rotation is sought.

More about rotation: <http://jalt.org/test/PDF/Brown31.pdf>

The results differ very slightly with what is in the book, largely the ordering of the factors but the differences are not substantive. Loadings are not different my more than |.02|.

Generally supports simple structure. (high one one but low on all others.). NOTE: SS does not say that one measure does not load on two or more factors. SS indicates that a measured variable does not load strongly on ALL factors. However, the double loadings make measurement challenging.

Also, note the correlations among factors. Some are stronger than others (e.g., .66, .45 vs .09).

efa1$loadings

##   
## Loadings:  
## PA2 PA1 PA3 PA4   
## TEN1 0.788   
## TEN2 0.842   
## TEN3 0.592 0.348  
## WOR1 0.923   
## WOR2 0.653   
## WOR3 0.145 0.611 0.103  
## IRTHK1 0.745   
## IRTHK2 0.777   
## IRTHK3 0.900   
## BODY1 0.612  
## BODY2 0.107 0.111 0.534  
## BODY3 0.693  
##   
## PA2 PA1 PA3 PA4  
## SS loadings 1.996 1.733 1.675 1.287  
## Proportion Var 0.166 0.144 0.140 0.107  
## Cumulative Var 0.166 0.311 0.450 0.558

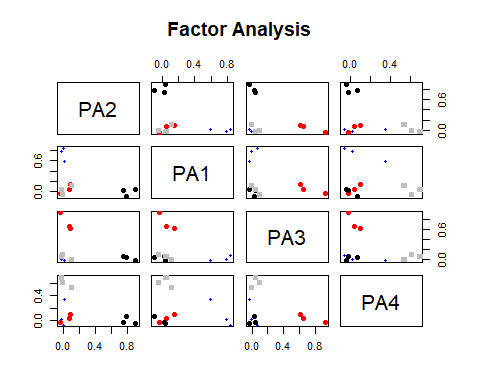
print(efa1$loadings, cutoff = .4) #you can do this to omit

##   
## Loadings:  
## PA2 PA1 PA3 PA4   
## TEN1 0.788   
## TEN2 0.842   
## TEN3 0.592   
## WOR1 0.923   
## WOR2 0.653   
## WOR3 0.611   
## IRTHK1 0.745   
## IRTHK2 0.777   
## IRTHK3 0.900   
## BODY1 0.612  
## BODY2 0.534  
## BODY3 0.693  
##   
## PA2 PA1 PA3 PA4  
## SS loadings 1.996 1.733 1.675 1.287  
## Proportion Var 0.166 0.144 0.140 0.107  
## Cumulative Var 0.166 0.311 0.450 0.558

#the other lower loadings  
round(efa1$weights, 2)

## PA2 PA1 PA3 PA4  
## TEN1 -0.02 0.30 0.00 0.03  
## TEN2 -0.03 0.38 0.04 -0.02  
## TEN3 0.00 0.30 0.00 0.30  
## WOR1 0.01 0.00 0.56 -0.02  
## WOR2 0.02 0.02 0.19 0.02  
## WOR3 0.06 0.06 0.23 0.07  
## IRTHK1 0.22 0.00 0.03 -0.01  
## IRTHK2 0.27 -0.05 0.02 0.04  
## IRTHK3 0.50 -0.01 0.02 0.00  
## BODY1 -0.01 0.00 0.03 0.21  
## BODY2 0.02 0.03 0.00 0.20  
## BODY3 0.01 0.03 0.00 0.35

plot(efa1) #can also plot the loadings using several 2x2 matrices



The standardized (pattern matrix) loadings show a clean structure. Three items per factor go along well very nicely. Can also view the structure matrix (zero order correlations of the factors with the observed variables).

print(efa1$Structure, cutoff = 0) #indicate zero to show all the correlations

##   
## Loadings:  
## PA2 PA1 PA3 PA4   
## TEN1 0.045 0.797 0.338 0.526  
## TEN2 0.099 0.833 0.419 0.519  
## TEN3 0.134 0.805 0.391 0.722  
## WOR1 0.371 0.353 0.881 0.363  
## WOR2 0.384 0.369 0.727 0.387  
## WOR3 0.407 0.489 0.764 0.496  
## IRTHK1 0.765 0.102 0.389 0.202  
## IRTHK2 0.798 0.029 0.367 0.209  
## IRTHK3 0.878 0.072 0.369 0.185  
## BODY1 0.181 0.385 0.346 0.618  
## BODY2 0.242 0.465 0.330 0.629  
## BODY3 0.170 0.511 0.366 0.736  
##   
## PA2 PA1 PA3 PA4  
## SS loadings 2.594 3.120 3.116 3.032  
## Proportion Var 0.216 0.260 0.260 0.253  
## Cumulative Var 0.216 0.476 0.736 0.988

Often, the recommendation is to interpret the **standardized pattern matrix** which are like the regression weights.

To compare without rotation or rotation using 'varimax' only.

fa(efam, n.obs = 318, nfactors = 4, fm = 'pa', rotate = 'none' )

## Factor Analysis using method = pa  
## Call: fa(r = efam, nfactors = 4, n.obs = 318, rotate = "none", fm = "pa")  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## PA1 PA2 PA3 PA4 h2 u2 com  
## TEN1 0.60 -0.45 0.11 -0.25 0.64 0.36 2.3  
## TEN2 0.66 -0.41 0.06 -0.31 0.70 0.30 2.2  
## TEN3 0.71 -0.42 0.20 -0.04 0.72 0.28 1.8  
## WOR1 0.68 0.16 -0.54 0.03 0.78 0.22 2.0  
## WOR2 0.64 0.13 -0.33 0.01 0.54 0.46 1.6  
## WOR3 0.74 0.07 -0.27 0.00 0.63 0.37 1.3  
## IRTHK1 0.47 0.58 0.16 -0.08 0.59 0.41 2.1  
## IRTHK2 0.45 0.64 0.19 0.00 0.64 0.36 2.0  
## IRTHK3 0.48 0.69 0.25 -0.11 0.77 0.23 2.1  
## BODY1 0.52 -0.15 0.10 0.30 0.39 0.61 1.9  
## BODY2 0.56 -0.14 0.19 0.20 0.41 0.59 1.6  
## BODY3 0.61 -0.24 0.16 0.30 0.54 0.46 2.0  
##   
## PA1 PA2 PA3 PA4  
## SS loadings 4.32 1.90 0.72 0.41  
## Proportion Var 0.36 0.16 0.06 0.03  
## Cumulative Var 0.36 0.52 0.58 0.61  
## Proportion Explained 0.59 0.26 0.10 0.06  
## Cumulative Proportion 0.59 0.85 0.94 1.00  
##   
## Mean item complexity = 1.9  
## Test of the hypothesis that 4 factors are sufficient.  
##   
## The degrees of freedom for the null model are 66 and the objective function was 5.58 with Chi Square of 1741.38  
## The degrees of freedom for the model are 24 and the objective function was 0.11   
##   
## The root mean square of the residuals (RMSR) is 0.02   
## The df corrected root mean square of the residuals is 0.03   
##   
## The harmonic number of observations is 318 with the empirical chi square 9.92 with prob < 0.99   
## The total number of observations was 318 with Likelihood Chi Square = 33.04 with prob < 0.1   
##   
## Tucker Lewis Index of factoring reliability = 0.985  
## RMSEA index = 0.001 and the 90 % confidence intervals are NA 0.061  
## BIC = -105.25  
## Fit based upon off diagonal values = 1  
## Measures of factor score adequacy   
## PA1 PA2 PA3 PA4  
## Correlation of scores with factors 0.96 0.93 0.84 0.70  
## Multiple R square of scores with factors 0.93 0.86 0.71 0.49  
## Minimum correlation of possible factor scores 0.85 0.72 0.42 -0.02

#with none, note a lot of double loadings, fourth factor does not have anything that loads well  
efa2 <- fa(efam, n.obs = 318, nfactors = 4, fm = 'pa', rotate = 'varimax' )  
print(efa2$loadings, cutoff = .4)

##   
## Loadings:  
## PA2 PA4 PA3 PA1   
## TEN1 0.726   
## TEN2 0.763   
## TEN3 0.648 0.519  
## WOR1 0.836   
## WOR2 0.638   
## WOR3 0.634   
## IRTHK1 0.736   
## IRTHK2 0.772   
## IRTHK3 0.864   
## BODY1 0.566  
## BODY2 0.541  
## BODY3 0.661  
##   
## PA2 PA4 PA3 PA1  
## SS loadings 2.078 1.843 1.787 1.637  
## Proportion Var 0.173 0.154 0.149 0.136  
## Cumulative Var 0.173 0.327 0.476 0.612

#actually, the orthogonal rotation is not bad too except that TEN3 loads on two factors

### 4. Computing scale reliabilities

After running an EFA and you have empirical support of grouping certain items together in a scale, can compute Cronbach's alpha which is an often used measure of internal consistency (for better or for worse). NOTE: at times, researchers just report alphas and think this is enough to say that they have a good measure. Not really.

Read: <http://www.ats.ucla.edu/stat/spss/faq/alpha.html>. Many articles that talk about the limitations of alpha.

To compute this, can use the alpha function in the psych package. Pass a dataset with the items to compute the alphas for.

colnames(efam)

## [1] "TEN1" "TEN2" "TEN3" "WOR1" "WOR2" "WOR3" "IRTHK1"  
## [8] "IRTHK2" "IRTHK3" "BODY1" "BODY2" "BODY3"

alpha(efam[1:3,1:3])

##   
## Reliability analysis   
## Call: alpha(x = efam[1:3, 1:3])  
##   
## raw\_alpha std.alpha G6(smc) average\_r S/N  
## 0.85 0.85 0.79 0.66 5.7  
##   
## Reliability if an item is dropped:  
## raw\_alpha std.alpha G6(smc) average\_r S/N  
## TEN1 0.80 0.80 0.66 0.66 3.9  
## TEN2 0.79 0.79 0.65 0.65 3.7  
## TEN3 0.80 0.80 0.66 0.66 3.9  
##   
## Item statistics   
## r r.cor r.drop  
## TEN1 0.88 0.78 0.72  
## TEN2 0.88 0.79 0.73  
## TEN3 0.88 0.78 0.72

alpha(efam[4:6,4:6])

##   
## Reliability analysis   
## Call: alpha(x = efam[4:6, 4:6])  
##   
## raw\_alpha std.alpha G6(smc) average\_r S/N  
## 0.83 0.83 0.77 0.62 5  
##   
## Reliability if an item is dropped:  
## raw\_alpha std.alpha G6(smc) average\_r S/N  
## WOR1 0.73 0.73 0.57 0.57 2.7  
## WOR2 0.80 0.80 0.66 0.66 3.9  
## WOR3 0.78 0.78 0.64 0.64 3.6  
##   
## Item statistics   
## r r.cor r.drop  
## WOR1 0.89 0.81 0.73  
## WOR2 0.85 0.73 0.66  
## WOR3 0.86 0.75 0.68

alpha(efam[7:9,7:9])

##   
## Reliability analysis   
## Call: alpha(x = efam[7:9, 7:9])  
##   
## raw\_alpha std.alpha G6(smc) average\_r S/N  
## 0.85 0.85 0.8 0.66 5.8  
##   
## Reliability if an item is dropped:  
## raw\_alpha std.alpha G6(smc) average\_r S/N  
## IRTHK1 0.82 0.82 0.70 0.70 4.7  
## IRTHK2 0.80 0.80 0.67 0.67 4.1  
## IRTHK3 0.76 0.76 0.61 0.61 3.1  
##   
## Item statistics   
## r r.cor r.drop  
## IRTHK1 0.86 0.75 0.69  
## IRTHK2 0.88 0.78 0.72  
## IRTHK3 0.90 0.83 0.76

alpha(efam[10:12,10:12])

##   
## Reliability analysis   
## Call: alpha(x = efam[10:12, 10:12])  
##   
## raw\_alpha std.alpha G6(smc) average\_r S/N  
## 0.7 0.7 0.61 0.44 2.3  
##   
## Reliability if an item is dropped:  
## raw\_alpha std.alpha G6(smc) average\_r S/N  
## BODY1 0.65 0.65 0.48 0.48 1.8  
## BODY2 0.63 0.63 0.46 0.46 1.7  
## BODY3 0.54 0.54 0.37 0.37 1.2  
##   
## Item statistics   
## r r.cor r.drop  
## BODY1 0.77 0.58 0.48  
## BODY2 0.78 0.60 0.50  
## BODY3 0.82 0.68 0.57

We are doing it this way because we just have the correlation matrix. You can pass off a dataset with just the variable to compute the alphas for (but we don't have the raw data here).

Often, alphas > .70 are taken as signs of good internal consistency. [again, debates about this but we will not touch on this]

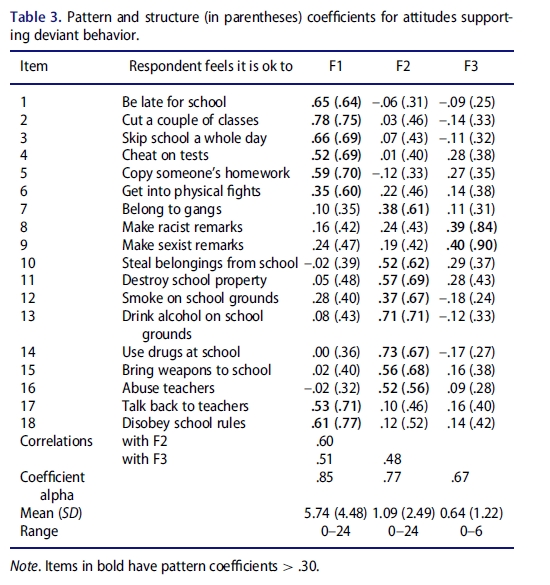
### 4. Presenting results

Need to interpret the factors (Often which measured variable has the largest association with the factor? Other times, look at how the items go together and describe what the factor is). Writeup should include:

1. Factor model used (e.g., use of PAF, etc.)
2. Whether correlation or covariance matrix was used
3. Rule(s) for determining number of factors
4. Use of rotation (what type)
5. Size of factor coefficient required for interpretation (e.g., > .35)

Should also include:

* Sample size, number of variables
* Correlation table (and M and SD)
* Results of the number of factors
* Pattern and structure matrix (separate for oblique, one and the same for orthogonal)



Example from: Huang, F. (2016): Do Black students misbehave more? Investigating the differential involvement hypothesis and out-of-school suspensions, The Journal of Educational Research, DOI: 10.1080/00220671.2016.1253538

### 5. Additional stuff

Many, many books and articles have been written on the topic (read your book chapter too! These notes are not a replacement for the required reading):

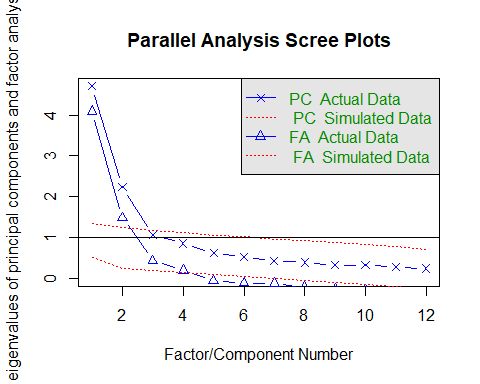
* Fabrigar & Wegener (2012), Exploratory Factor Analysis. New York, NY: Oxford University Press.
* Costello & Osborne (2005). Best practices in EFA. Practical Assessment, Research, & Evaluation.
* Henson & Roberts (2006). Use of EFA in published research. Educational and Psychological Measurement.
* Pohlmann, J. (2004). Use and interpretation of factor analysis in the Journal of Educational Research: 1992-2002. Journal of Educational Research.

There are lots of different issues that should be dealt with (e.g., dealing with categorical indicators, multilevel factor analysis, etc.).

### OTHERS

While earlier, the use of PA was illustrated to be able to conceptually explain what goes on when a PA is done (comparing this with the eigenvalues), the psych package also has a PA feature:

fa.parallel(efam, n.obs = 318)



## Parallel analysis suggests that the number of factors = 4 and the number of components = 2

Results are the same as what we derived earlier. However, it is important to have an understanding of what it is really doing (now you should).

Another way to determine the number of factors is to use fm = 'ml' (maximum likelihood) and compare model fit indices by incrementally adding factors and seeing which one fits best using a likelihood ratio test or comparing RMSEA statistics. Maximum likelihood assumes multivariate normality (principal axis factoring does not) and has the advantage that it provides the fit indices (like in a traditional CFA).

* Central to ML is the **likelihood function** that reflects the relative likelihood of the observed data given a set of estimates for the model parameters (F&W, p. 47).
* Goal of ML is to find a set of parameters for a given model that are most likely to have produced the observed data- attempts to find a set of values for the factor loadings and unique variances that will produce the largest likelihood function
* Produces the test of model fit with degrees of freedom df = (p - m)^2 - (p + m)/2 where p is the numbmer of measured variabbles and m the number of common factors in the model.
* The test is an index of fit where the H0 is that the common factor model with m factors holds perfectly in the population (thus is a good fit). Not rejecting the null (p > .05) is a good thing!
* Often criticized because sensitive to sample size (larger sample, easier to find statistical significance)

This way though is less clear and not always recommended.

t1 <- fa(efam, n.obs = 318, nfactors = 1, fm = 'ml')  
t2 <- fa(efam, n.obs = 318, nfactors = 2, fm = 'ml')  
t3 <- fa(efam, n.obs = 318, nfactors = 3, fm = 'ml')  
t4 <- fa(efam, n.obs = 318, nfactors = 4, fm = 'ml')  
t5 <- fa(efam, n.obs = 318, nfactors = 5, fm = 'ml')

## Warning in fac(r = r, nfactors = nfactors, n.obs = n.obs, rotate =  
## rotate, : A Heywood case was detected. Examine the loadings carefully.

#heywood case suggests an error somewhere, such as a negative variance or communalities > 1 (or very near 1) which is not possible  
#Structure matrices cannot have values > 1 since it is a correlation. However, pattern matrices CAN have values > 1 since they are like regression coefficients.   
#notice also the loadings with five factors, the last one only had one measured variable loading  
  
t4$communalities #this is fine

## TEN1 TEN2 TEN3 WOR1 WOR2 WOR3 IRTHK1   
## 0.6287763 0.7089159 0.7169294 0.7965700 0.5339236 0.6199313 0.5824200   
## IRTHK2 IRTHK3 BODY1 BODY2 BODY3   
## 0.6410613 0.7841143 0.3917790 0.4167536 0.5318376

t5$communalities #you can see it with IRTHNK3

## TEN1 TEN2 TEN3 WOR1 WOR2 WOR3 IRTHK1   
## 0.6376726 0.7056674 0.7172005 0.7686710 0.5522912 0.6385581 0.6433564   
## IRTHK2 IRTHK3 BODY1 BODY2 BODY3   
## 0.6064849 0.9950000 0.4765138 0.4058217 0.5200716

t1$STATISTIC

## [1] 753.0624

#t2$STATISTIC  
#t3$STATISTIC  
#t4$STATISTIC #t4$dof  
#t5$STATISTIC

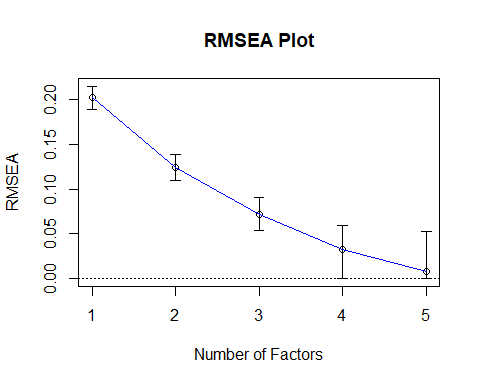
You can do the model fitting one model at a time. A quicker way is to use a special function that you can load below from James Steiger <http://www.statpower.net/Content/312/Handout/Advanced%20Exploratory%20Factor%20Analysis%20in%20R.pdf>

source("http://www.statpower.net/Content/312/R%20Stuff/AdvancedFactorFunctions.txt")

##   
## Attaching package: 'plotrix'

## The following object is masked from 'package:psych':  
##   
## rescale

#this will load in some new functions like FA.stats  
round(FA.Stats(efam, n.factors= 1:5, n.obs = 318), 3)



## Factors Cum.Eigen Chi-Square Df p.value RMSEA.Pt RMSEA.Lo RMSEA.Hi  
## [1,] 1 4.703 753.062 54 0.000 0.202 0.189 0.215  
## [2,] 2 6.939 253.379 43 0.000 0.124 0.110 0.139  
## [3,] 3 8.007 86.600 33 0.000 0.072 0.053 0.090  
## [4,] 4 8.860 32.138 24 0.124 0.033 0.000 0.060  
## [5,] 5 9.477 16.306 16 0.432 0.008 0.000 0.053

This is a neat function since it can run the model with multiple indicated factors. You will note that will four factors, the . A good fit. Adding an additional factor does not improve the model anymore so a more parsimonious model is suitable. Also, the RMSEA is suitable (desired < .08, better < .05, note there is also a confidence interval).