Psychology 454: Latent Variable Modeling
Week 4: Latent models: A review of Exploratory Factor Analysis

William Revelle

Department of Psychology
Northwestern University
Evanston, Illinois USA

October, 2024

Outline

Introduction

1. Ockham's razor and data reduction
2. Factor analysis – several examples
   - Data from a correlation matrix
     - Simulated 2 factor data
     - Real data – Ability tests
   - Raw data
     - Simulated 2 factor data
     - Real data – 5 Personality dimensions
3. Principal Components analysis

Models of data

1. (MacCallum, 2004) "A factor analysis model is not an exact representation of real-world phenomena.
2. Always wrong to some degree, even in population.
3. At best, model is an approximation of real world."
4. Box (1979): "Models, of course, are never true, but fortunately it is only necessary that they be useful. For this it is usually needful only that they not be grossly wrong."
5. Tukey (1961): "In a single sentence, the moral is: Admit that complexity always increases, first from the model you fit to the data, thence to the model you use to think and plan about the experiment and its analysis, and thence to the true situation."

(From MacCallum, 2004); http://www.fa100.info/maccallum2.pdf

Observed Variables

X	Y
$\boxed{X_1}$	$\boxed{Y_1}$
$\boxed{X_2}$	$\boxed{Y_2}$
$\boxed{X_3}$	$\boxed{Y_3}$
$\boxed{X_4}$	$\boxed{Y_4}$
$\boxed{X_5}$	$\boxed{Y_5}$
$\boxed{X_6}$	$\boxed{Y_6}$

Latent Variables

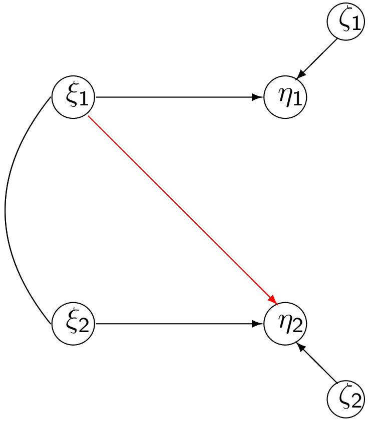
ξ η

ξ_1 η_1

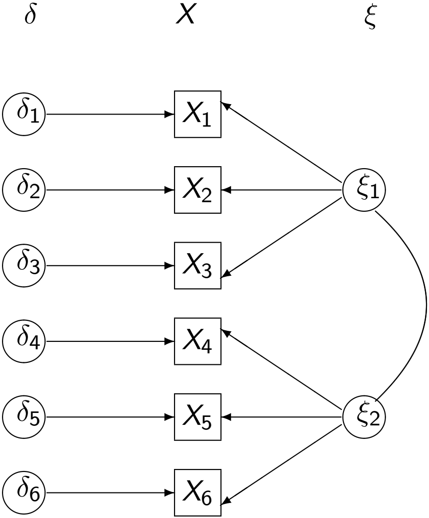
ξ_2 η_2

Theory: A regression model of latent variables

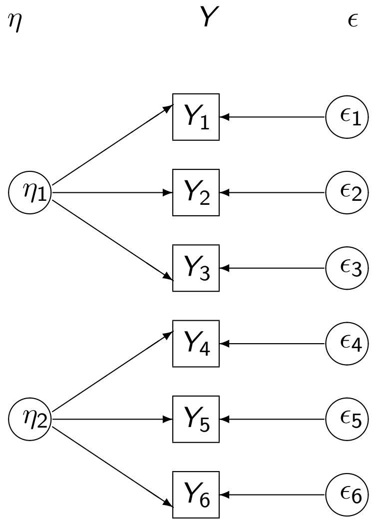
ξ η



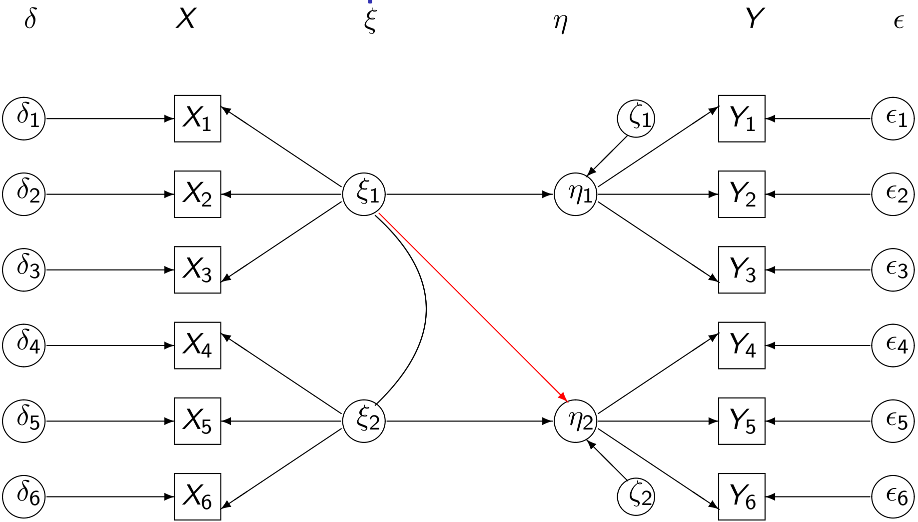
A measurement model for X



A measurement model for Y



A complete structural model



Various measurement models

1. Observed variables models
    - Singular Value Decompostion
    - Eigen Value – Eigen Vector decomposition
    - Principal Components
    - First k principal components as an approximation
2. Latent variable models
    - Exploratory Factor analysis
    - Confirmatory Factor analysis
        - Growth Curve Models
        - Latent Class Models
    - Item Response Theory models
3. Interpretation of models
    - Choosing the appropriate number of components/factors
    - Transforming/rotating towards interpretable structures

Singular Value Decomposition of the data matrix

Consider the matrix **X** of n deviation scores for N subjects, where each element, x_{ij}, represents the responses of the i^{th} individual to the j^{th} item or test. For simplicity, let the x_{ij} scores in each column be deviations from the mean for that column (i.e., they are column centered, perhaps by using `scale`). Let the number of variables be n. Then the `svd` function will find the *Singular Value Decomposition* of **X** which allows us to express **X** as the product of three orthogonal matrices:

$$_{\mathbf{N}}\mathbf{X}_{\mathbf{n}} = {}_{\mathbf{N}}\mathbf{U}_{\mathbf{nn}}\mathbf{D}_{\mathbf{nn}}\mathbf{V}'_{\mathbf{n}}$$

where **D** is a diagonal matrix of the *singular values* and the **U** and **V** matrices are matrices of the *singular vectors*. Although descriptive of the data, what is *meaning* of these vectors?

Decomposition (models) of Correlation and Covariance matrices

With $\mathbf{X}$ defined as before, the covariance matrix, $\mathbf{Cov}$, is

$$\mathbf{Cov} = N^{-1}\mathbf{XX'}$$

and the standard deviations are

$$\mathbf{sd} = \sqrt{diag(\mathbf{Cov})}.$$

Let the matrix $\mathbf{I_{sd}}$ be a diagonal matrix with elements $= \frac{1}{sd_i}$, then the correlation matrix $\mathbf{R}$ is

$$\mathbf{R} = \mathbf{I_{sd}CovI_{sd}}.$$

The problem is how to approximate the matrix, $\mathbf{R}$ of rank n, with a matrix of lower rank? The solution to this problem may be seen if we think about how to create a model matrix to approximate $\mathbf{R}$.

Factor Analysis/Components Analysis/Cluster Analysis

1. Data simplification and Ockham's Razor: "do not multiple entities beyond necessity"
2. Can we describe a data set with a simpler representation of the data.
3. Is it possible to combine subjects and or variables that are redundant?
4. Or almost redundant (without losing very much information)
5. This is a problem in projective geometry. Can we project from a high dimensional space into a lower order space.

An example correlation matrix

Consider the following correlation matrix

	V1	V2	V3	V4	V5	V6
V1	1.00	0.72	0.63	0.54	0.45	0.36
V2	0.72	1.00	0.56	0.48	0.40	0.32
V3	0.63	0.56	1.00	0.42	0.35	0.28
V4	0.54	0.48	0.42	1.00	0.30	0.24
V5	0.45	0.40	0.35	0.30	1.00	0.20
V6	0.36	0.32	0.28	0.24	0.20	1.00

Is it possible to model these 36 correlations and variances with fewer terms? Yes, of course. The diagonal elements are all 1 and the off diagonal elements are symmetric. Thus, we have $n * (n - 1)$ correlations we want to model.

Eigen vector decomposition

Given a *nxn* matrix **R**, each eigenvector, $\mathbf{x_i}$, solves the equation

$$\mathbf{x_i R} = \lambda_i \mathbf{x_i}$$

and the set of n eigenvectors are solutions to the equation

$$\mathbf{XR} = \boldsymbol{\lambda} \mathbf{X}$$

where **X** is a matrix of orthogonal eigenvectors and $\boldsymbol{\lambda}$ is a diagonal matrix of the the eigenvalues, λ_i. Then

$$\mathbf{x_i R} - \lambda_i \mathbf{XI} = 0 <=> \mathbf{x_i}(\mathbf{R} - \lambda_i \mathbf{I}) = 0$$

Finding the eigenvectors and eigenvalues is computationally tedious, but may be done using the `eigen` function. That the vectors making up **X** are orthogonal means that

$$\mathbf{XX'} = \mathbf{I}$$

and because they form the *basis space* for **R** that

$$\mathbf{R} = \mathbf{X} \boldsymbol{\lambda} \mathbf{X'}.$$

Consider the eigen value solution for the example correlation matrix.

```
> e <- eigen(R)
> print(e, digits=2)

$values
[1] 3.16 0.82 0.72 0.59 0.44 0.26

$vectors
      [,1]   [,2]   [,3]  [,4]   [,5]   [,6]
[1,] -0.50 -0.061  0.092  0.14  0.238  0.816
[2,] -0.47 -0.074  0.121  0.21  0.657 -0.533
[3,] -0.43 -0.096  0.182  0.53 -0.675 -0.184
[4,] -0.39 -0.142  0.414 -0.78 -0.201 -0.104
[5,] -0.34 -0.299 -0.860 -0.20 -0.108 -0.067
[6,] -0.28  0.934 -0.178 -0.10 -0.067 -0.045

> round(e$vectors %*% t(e$vectors),2)   #the eigen vectors are ort
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    1    0    0    0    0    0
[2,]    0    1    0    0    0    0
[3,]    0    0    1    0    0    0
[4,]    0    0    0    1    0    0
[5,]    0    0    0    0    1    0
[6,]    0    0    0    0    0    1
```

Eigen Value decomposition and recreation of the original matrix

Find the eigen values (λ) and eigen vectors ($\mathbf{V_i}$).

```
> e <- eigen(R)
> print(e, digits=2)


$values
[1] 3.16 0.82 0.72 0.59 0.44 0.26

$vectors
      [,1]    [,2]    [,3]   [,4]    [,5]
[,6]
[1,] -0.50 -0.061  0.092  0.14   0.238
0.816
[2,] -0.47 -0.074  0.121  0.21
0.657 -0.533
[3,] -0.43 -0.096  0.182
0.53 -0.675 -0.184
[4,] -0.39 -0.142
0.414 -0.78 -0.201 -0.104
[5,] -0.34 -0.299 -0.860 -0.20 -0.108 -0.067
[6,] -0.28
0.934 -0.178 -0.10 -0.067 -0.045

#the eigen vectors are orthogonal

> round(e$vectors %*% t(e$vectors),2)
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    1    0    0    0    0    0
[2,]    0    1    0    0    0    0
[3,]    0    0    1    0    0    0
```

The eigen vectors and values recreate the observed correlations.

$$\mathbf{R} = \mathbf{V}\lambda\mathbf{V}'.$$

```
> round(e$vectors %*% diag(e$values) %*% t(e$vecto


     [,1] [,2] [,3] [,4] [,5] [,6]
[1,] 1.00 0.72 0.63 0.54 0.45 0.36
[2,] 0.72 1.00 0.56 0.48 0.40 0.32
[3,] 0.63 0.56 1.00 0.42 0.35 0.28
[4,] 0.54 0.48 0.42 1.00 0.30 0.24
[5,] 0.45 0.40 0.35 0.30 1.00 0.20
[6,] 0.36 0.32 0.28 0.24 0.20 1.00
```

The eigen values reflect the scale, the vectors the structure

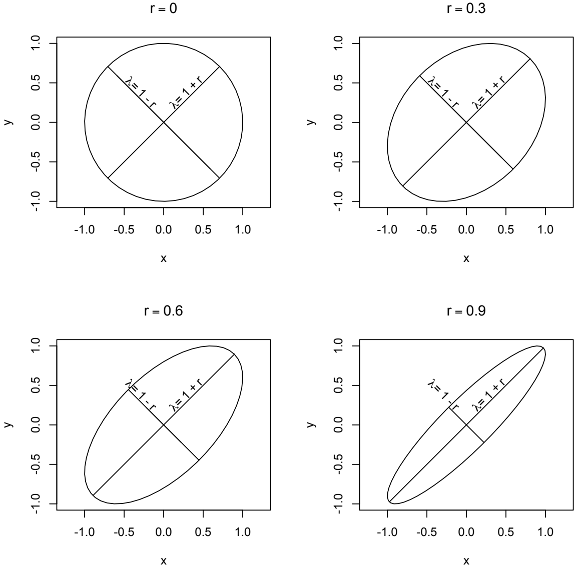
Consider the original data and solution

```
> R
> e <- eigen(R)
> print(e, digits=2)


     V1   V2   V3   V4   V5   V6
V1 1.00 0.72 0.63 0.54 0.45 0.36
V2 0.72 1.00 0.56 0.48 0.40 0.32
V3 0.63 0.56 1.00 0.42 0.35 0.28
V4 0.54 0.48 0.42 1.00 0.30 0.24
V5 0.45 0.40 0.35 0.30 1.00 0.20
V6 0.36 0.32 0.28 0.24 0.20 1.00

$values
[1] 3.16 0.82 0.72 0.59 0.44 0.26

$vectors
      [,1]    [,2]    [,3]  [,4]    [,5]
[,6]
[1,] -0.50 -0.061   0.092   0.14   0.238
0.816
[2,] -0.47 -0.074   0.121   0.21
0.657 -0.533
[3,] -0.43 -0.096   0.182
0.53 -0.675 -0.184
[4,] -0.39 -0.142
0.414 -0.78 -0.201 -0.104
[5,] -0.34 -0.299 -0.860 -0.20 -0.108 -0.067
[6,] -0.28
```

Consider if all the correlations are divided by 2.

```
> R.5 <- as.matrix(R/2)
> diag(R.5) <- 1
> R.5
> e.5 <- eigen(R.5)
> print(e.5,2)


      V1    V2    V3    V4    V5    V6
V1 1.000 0.36 0.315 0.27 0.225 0.18
V2 0.360 1.00 0.280 0.24 0.200 0.16
V3 0.315 0.28 1.000 0.21 0.175 0.14
V4 0.270 0.24 0.210 1.00 0.150 0.12
V5 0.225 0.20 0.175 0.15 1.000 0.10
V6 0.180 0.16 0.140 0.12 0.100 1.00

$values
[1] 2.08 0.91 0.86 0.80 0.72 0.63

$vectors
      [,1]    [,2]    [,3]  [,4]    [,5]
[,6]
[1,] 0.50 -0.061   0.092   0.14   0.238
0.816
[2,] 0.47 -0.074   0.121   0.21
0.657 -0.533
[3,] 0.43 -0.096   0.182
0.53 -0.675 -0.184
[4,] 0.39 -0.142
0.414 -0.78 -0.201 -0.104
[5,] 0.34 -0.299 -0.860 -0.20 -0.108 -0.067
```

Eigen vectors of a 2 x 2 correlation matrix



Although the length (eigen values) of the axes differ, their orientation (eigen vectors) are the same.

```
> r2 <- matrix(c(1,.6,.6,1),2
> print(eigen(r2),2)

$values
[1] 1.6 0.4

$vectors
     [,1]  [,2]
[1,] 0.71 -0.71
[2,] 0.71  0.71
```

From eigen vectors to Principal Components

1. For n variables, there are n eigen vectors
   - There is no parsimony in thinking of the eigen vectors
   - Except that the vectors provide the orthogonal basis for the variables
2. Principal components are formed from the eigen vectors and eigen values
   - $\mathbf{R} = \mathbf{V}\lambda\mathbf{V}' = \mathbf{C}\mathbf{C}'$
   - $\mathbf{C} = \mathbf{V}\sqrt{\lambda}$
3. But there will still be as many Principal Components as variables, so what is the point?
4. Take just the first k Principal Components and see how well this reduced model fits the data.

The first principal component.

```
> pc1 <- principal(R,1)
> pc1

Uniquenesses:
   V1    V2    V3    V4
V5    V6
0.220 0.307 0.408 0.519 0.635 0.748
Loadings:
   PC1
V1 0.88
V2 0.83
V3 0.77
V4 0.69
V5 0.60
V6 0.50

                 PC1
SS loadings    3.142
Proportion Var 0.524
```

```
#show the model
> round(pc1$loadings %*% t(pc1$loadings),2)

     V1   V2   V3   V4   V5   V6
V1 0.77 0.73 0.68 0.61 0.53 0.44
V2 0.73 0.69 0.64 0.57 0.50 0.42
V3 0.68 0.64 0.59 0.53 0.46 0.38
V4 0.61 0.57 0.53 0.48 0.41 0.34
V5 0.53 0.50 0.46 0.41 0.36 0.30
V6 0.44 0.42 0.38 0.34 0.30 0.25

#find the residuals
> Rresid <-
R - pc1$loadings %*% t(pc1$loadings)
> round(Rresid,2)

      V1    V2    V3    V4    V5    V6
V1  0.23 -0.01 -0.05 -0.07 -0.08 -0.08
V2 -0.01  0.31 -0.08 -0.09 -0.10 -0.09
V3 -0.05 -0.08  0.41 -0.11 -0.11 -0.10
V4 -0.07 -0.09 -0.11  0.52 -0.11 -0.10
V5 -0.08 -0.10 -0.11 -0.11  0.64 -0.10
V6 -0.08 -0.09 -0.10 -0.10 -0.10  0.75
```

The model fits pretty well, except that the diagonal is underestimated and the other correlations are over estimated.

Try 2 and 3 principal components

```
> p2 <- principal(R,2,rotate="none")
> p2
> resid(p2)

Principal Components Analysis
Call: principal(r = R, nfactors = 2, rotate = "none")
Standardized loadings (pattern matrix)
    PC1   PC2   h2    u2
V1 0.88 -0.06 0.78 0.217
V2 0.83 -0.07 0.70 0.302
V3 0.77 -0.09 0.60 0.400
V4 0.69 -0.13 0.50 0.502
V5 0.60 -0.27 0.44 0.561
V6 0.50  0.85 0.97 0.031

                 PC1  PC2
SS loadings     3.16 0.82
Proportion Var  0.53 0.14
Cumulative Var  0.53 0.66

Fit based upon off diagonal values = 0.95

     V1    V2    V3    V4    V5    V6
V1  0.22
V2 -0.02  0.30
V3 -0.05 -0.09  0.40
V4 -0.08 -0.11 -0.13  0.50
V5 -0.10 -0.12 -0.14 -0.15  0.56
V6 -0.04 -0.04 -0.03  0.00  0.13  0.03
```

```
> p3 <- principal(R,3,rotate="none")
> resid(p3)

Principal Components Analysis
Call: principal(r = R, nfactors = 3, rotate =
Standardized loadings (pattern matrix)
    PC1   PC2   PC3   h2     u2
V1 0.88 -0.06 -0.08 0.79 0.2108
V2 0.83 -0.07 -0.10 0.71 0.2917
V3 0.77 -0.09 -0.15 0.62 0.3761
V4 0.69 -0.13 -0.35 0.62 0.3789
V5 0.60 -0.27  0.73 0.97 0.0292
V6 0.50  0.85  0.15 0.99 0.0084

                 PC1  PC2  PC3
SS loadings     3.16 0.82 0.72
Proportion Var  0.53 0.14 0.12
Cumulative Var  0.53 0.66 0.78

Fit based upon off diagonal values = 0.97

     V1    V2    V3    V4    V5    V6
V1  0.21
V2 -0.03  0.29
V3 -0.07 -0.10  0.38
V4 -0.11 -0.14 -0.18  0.38
V5 -0.04 -0.05 -0.03  0.10  0.03
V6 -0.02 -0.03 -0.01  0.05  0.02  0.01
```

Consider the following matrix

Correlations between 6 variables

Variable	V1	V2	V3	V4	V5	V6
V1	1.00	0.72	0.63	0.54	0.45	0.36
V2	0.72	1.00	0.56	0.48	0.40	0.32
V3	0.63	0.56	1.00	0.42	0.35	0.28
V4	0.54	0.48	0.42	1.00	0.30	0.24
V5	0.45	0.40	0.35	0.30	1.00	0.20
V6	0.36	0.32	0.28	0.24	0.20	1.00

Can we represent this in a simpler way?

$$R = FF' + U^2$$

or

$$R = CC'$$

Representing a correlation matrix with factors or components

Correlations between 6 variables

Variable	V1	V2	V3	V4	V5	V6
V1	1.00	0.72	0.63	0.54	0.45	0.36
V2	0.72	1.00	0.56	0.48	0.40	0.32
V3	0.63	0.56	1.00	0.42	0.35	0.28
V4	0.54	0.48	0.42	1.00	0.30	0.24
V5	0.45	0.40	0.35	0.30	1.00	0.20
V6	0.36	0.32	0.28	0.24	0.20	1.00

Table: $R = FF' + U^2$

Variable	loading
V1	0.9
V2	0.8
V3	0.7
V4	0.6
V5	0.5
V6	0.4

Table: $R = CC'$

Variable	
V1	0.88
V2	0.83
V3	0.77
V4	0.69
V5	0.60
V6	0.50

Factors vs. components

Originally developed by Spearman (1904) for the case of one common factor, and then later generalized by Thurstone (1947) and others to the case of multiple factors, factor analysis is probably the most frequently used and sometimes the most controversial psychometric procedure. The factor model, although seemingly very similar to the components model, is in fact very different. For rather than having components as linear sums of variables, in the factor model the variables are themselves linear sums of the unknown factors. That is, while components can be solved for by doing an *eigenvalue* or *singular value decomposition*, factors are estimated as best fitting solutions (Eckart and Young, 1936; Householder and Young, 1938), normally through iterative methods (Jöreskog, 1978; Lawley and Maxwell, 1963). Cattell (1965) referred to components analysis as a closed model and factor analysis as an open model, in that by explaining just the common variance, there was still more variance to explain.

Iterative principal axes factor analysis

Principal components represents a $n * n$ matrix in terms of the first k components. It attempts to reproduce all of the **R** matrix. *Factor analysis* on the other hand, attempts to model just the common part of the matrix, which means all of the off-diagonal elements and the common part of the diagonal (the *communalities*). The non-common part, the *uniquenesses*, are simply that which is left over. An easy to understand procedure is *principal axes* factor analysis. This is similar to principal components, except that it is done with a reduced matrix where the diagonals are the communalities. The communalities can either be specified a priori, estimated by such procedures as multiple linear regression, or found by iteratively doing an eigenvalue decomposition and repeatedly replacing the original 1s on the diagonal with the the value of 1 - u^2 where

$$\mathbf{U}^2 = diag(\mathbf{R} - \mathbf{FF}').$$

Principal axes as eigen values of a reduced matrix

That is, starting with the original correlation or covariance matrix, **R**, find the k largest principal components, reproduce the matrix using those principal components. Find the resulting residual matrix, $\mathbf{R}^*$ and uniqueness matrix, $\mathbf{U}^2$ by

$$\mathbf{R}^* = \mathbf{R} - \mathbf{FF}' \tag{1}$$

$$\mathbf{U}^2 = diag(\mathbf{R}^*)$$

and then, for iteration *i*, find $\mathbf{R}_i$ by replacing the diagonal of the original **R** matrix with 1 - diag($\mathbf{U}^2$) found on the previous step. Repeat this process until the change from one iteration to the next is arbitrarily small.

Comparing 1 with 5 iterations

```
> f1 <- fa(R,1,fm='pa',max.iter=1)
> f1
> resid(f1)
```

```
Factor Analysis using method =  pa
Call: fa(r = R, nfactors = 1, max.iter = 1, fm = "pa")
Standardized loadings (pattern matrix)
    PA1   h2   u2
V1 0.86 0.74 0.26
V2 0.79 0.62 0.38
V3 0.70 0.48 0.52
V4 0.60 0.36 0.64
V5 0.50 0.25 0.75
V6 0.40 0.16 0.84

                PA1
SS loadings    2.62
Proportion Var 0.44

   V1   V2   V3   V4   V5   V6
V1 0.26
V2 0.04 0.38
V3 0.03 0.01 0.52
V4 0.02 0.01 0.00 0.64
V5 0.02 0.00 0.00 0.00 0.75
V6 0.01 0.00 0.00 0.00 0.00 0.84
```

```
> f1 <- fa(R,1,fm='pa',max.iter=5)
> f1
> resid(f1)
```

```
Factor Analysis using method =  pa
Call: fa(r = R, nfactors = 1, max.iter = 5, fr
Standardized loadings (pattern matrix)
    PA1   h2   u2
V1 0.9 0.81 0.19
V2 0.8 0.64 0.36
V3 0.7 0.49 0.51
V4 0.6 0.36 0.64
V5 0.5 0.25 0.75
V6 0.4 0.16 0.84

                PA1
SS loadings    2.71
Proportion Var 0.45

   V1   V2   V3   V4   V5   V6
V1 0.19
V2 0.00 0.36
V3 0.00 0.00 0.51
V4 0.00 0.00 0.00 0.64
V5 0.00 0.00 0.00 0.00 0.75
V6 0.00 0.00 0.00 0.00 0.00 0.84
```

SMCs as initial communality estimates

Rather than starting with initial communality estimates of 1, the process can be started with other estimates of the communality. A conventional starting point is the lower bound estimate of the communalities, the *squared multiple correlation* or *SMC* (Roff, 1936).

The concept here is that a variable's communality must be at least as great as the amount of its variance that can be predicted by all of the other variables. The squared multiple correlations of each variable with the remaining variables are the diagonal elements of

$$\mathbf{I} - (diag(\mathbf{R}^{-1})^{-1}$$

and thus a starting estimate for $\mathbf{R_0}$ would be $\mathbf{R} - (diag(\mathbf{R}^{-1})^{-1}$.

Goodness of fit–simple estimates

At least three indices of goodness of fit of the principal factors model can be considered: One compares the sum of squared residuals to the sum of the squares of the original values:

$$GF_{total} = 1 - \frac{\mathbf{1R^{*2}1'}}{\mathbf{1R^{2}1'}}$$

The second does the same, but does not consider the diagonal of **R**

$$GF_{offdiagonal} = 1 - \frac{\sum_{i \neq j} r_{ij}^{*2}}{\sum_{i \neq j} r_{ij}^{*2}} = 1 - \frac{\mathbf{1R^{*2}1'} - tr(\mathbf{1R^{*2}1'}}{\mathbf{1R^{2}1'} - tr(\mathbf{1R^{2}1'})}$$

Finally, a χ^2 test of the size of the residuals simply sums all the squared residuals and multiplies by the number of observations:

$$\chi^2 = \sum_{i<j} r*_{ij}^2 (N-1)$$

with p * (p-1)/2 degrees of freedom.

OLS

The fundamental factor equation (Equation 1) may be viewed as set of simultaneous equations which may be solved several different ways: *ordinary least squares*, *generalized least squares*, and *maximum likelihood*. Ordinary least squares (*OLS*) or *unweighted least squares* (*ULS*) minimizes the sum of the squared residuals when modeling the sample correlation or covariance matrix, **S**, with $\Sigma = \mathbf{FF'} + \mathbf{U^2}$

$$E = \frac{1}{2} tr(\mathbf{S} - \Sigma)^2 \qquad (2)$$

where the *trace*, `tr`, of a matrix is the sum of the diagonal elements and the division by two reflects the symmetry of the **S** matrix.

MLE

Equation 2 can be generalized to weight the residuals $(\mathbf{S} - \Sigma)$ by the inverse of the sample matrix, $\mathbf{S}$, and thus to minimize

$$E = \frac{1}{2}tr((\mathbf{S} - \Sigma)\mathbf{S}^{-1})^2 = \frac{1}{2}tr(\mathbf{I} - \Sigma\mathbf{S}^{-\mathbf{1}})^2. \tag{3}$$

This is known as *generalized least squares* (*GLS*) or *weighted least squares* (*WLS*). Similarly, if the residuals are weighted by the inverse of the model, Σ, minimizing

$$E = \frac{1}{2}tr((\mathbf{S} - \Sigma)\Sigma^{-1})^2 = \frac{1}{2}tr(\mathbf{S}\Sigma^{-1} - I)^2 \tag{4}$$

will result in a model that maximizes the likelihood of the data. This procedure, *maximum likelihood estimation* (*MLE*) is also seen as finding the minimum of

$$E = \frac{1}{2}\left(tr(\Sigma^{-1}\mathbf{S}) - ln\left|\Sigma^{-1}\mathbf{S}\right| - p\right) \tag{5}$$

where p is the number of variables (Jöreskog, 1978). Perhaps a helpful intuitive explanation of Equation 8 is that if the model is

Maximum Likelihood Estimation

(MLE) is also seen as finding the minimum of

$$E = \frac{1}{2}\left(tr(\Sigma^{-1}\mathbf{S}) - ln\left|\Sigma^{-1}\mathbf{S}\right| - p\right) \quad (8)$$

where p is the number of variables (Jöreskog, 1978). Perhaps a helpful intuitive explanation of Equation 8 is that if the model is correct, then $\Sigma = \mathbf{S}$ and thus $\Sigma^{-1}\mathbf{S} = \mathbf{I}$. The trace of an identity matrix of rank p is p, and the logarithm of $|\mathbf{I}|$ is 0. Thus, the value of E if the model has perfect fit is 0. With the assumption of multivariate normality of the residuals, and for large samples, a χ^2 statistic can be estimated for a model with p variables and f factors:

$$\chi^2 = \left(tr(\Sigma^{-1}\mathbf{S}) - ln\left|\Sigma^{-1}\mathbf{S}\right| - p\right)\left(N - 1 - (2p+5)/6 - (2f)/3\right). \quad (9)$$

This χ^2 has degrees of freedom:

$$df = p * (p-1)/2 - p * f + f * (f-1)/2. \quad (10)$$

That is, the number of lower off diagonal correlations - the number

Minimum Residual Factor Analysis

The previous factor analysis procedures attempt to optimize the fit of the model matrix (Σ) to the correlation or covariance matrix ($\mathbf{S}$). The diagonal of the matrix is treated as mixture of common variance and unique variance and the problem becomes one of estimating the common variance (the *communality* of each variable). An alternative is to ignore the diagonal and to find that model which minimizes the squared residuals of the off diagonal elements. This is done in the `fa` function using the "minres" option by finding the solution that minimizes

$$\frac{1}{2}\mathbf{1}\left((\mathbf{S}-\mathbf{I})-(\Sigma-tr(\Sigma)\right)^2\mathbf{1}'. \tag{11}$$

The advantage of the *minres* solution is that it does not require finding the inverse of either the original correlation matrix (as do *GLS* and *WLS*) nor of the model matrix (as does *MLE*, and thus can be performed on non-positive definite matrices or matrices that are not invertible.

Solutions with more than 1 factor or component

Nothing in the previous algebra restricted the dimensionality of the **F** matrix or **C** matrix to be one column. That is, why limit ourselves to a one dimensional solution? Consider the following correlation matrix (constructed by creating a factor matrix and then finding its inner product).

```
>F <- matrix(c(.9,.8,.7,rep(0,6),.8,.7,.6),ncol=2) #the model
> rownames(F) <- paste("V",seq(1:6),sep="")  #add labels
> colnames(F) <- c("F1", "F2")
> R <- F %*% t(F) #create the correlation matrix
> diag(R) <- 1  #adjust the diagonal of the matrix
> R
```

	V1	V2	V3	V4	V5	V6
V1	1.00	0.72	0.63	0.00	0.00	0.00
V2	0.72	1.00	0.56	0.00	0.00	0.00
V3	0.63	0.56	1.00	0.00	0.00	0.00
V4	0.00	0.00	0.00	1.00	0.56	0.48
V5	0.00	0.00	0.00	0.56	1.00	0.42
V6	0.00	0.00	0.00	0.48	0.42	1.00

Try one principal component to this model.

```
> pc1 <- principal(R)
> pc1
```

The residuals are large for the second set of variables.

```
Principal Components Analysis
Call: principal(r = R)
Standardized loadings (pattern matrix) based upon correlation matrix
    PC1   h2   u2
V1 0.90 0.82 0.18
V2 0.88 0.77 0.23
V3 0.83 0.69 0.31
V4 0.00 0.00 1.00
V5 0.00 0.00 1.00
V6 0.00 0.00 1.00

                PC1
SS loadings    2.28
Proportion Var 0.38

Test of the hypothesis that 1 component is sufficient.

The degrees of freedom for the null model are
15  and the objective function was  1.96
The degrees of freedom for the model are 9
and the objective function was  0.87

Fit based upon off diagonal values = 0.61
```

```
> resid(pc1)
```

```
     V1    V2    V3    V4
V5    V6
V1  0.18
V2 -0.07  0.23
V3 -0.12 -0.17  0.31
V4  0.00  0.00  0.00  1.00
V5  0.00  0.00  0.00  0.56  1.00
V6  0.00  0.00  0.00  0.48
0.42  1.00
```

Two Principal Components

```
> pc2 <- principal(R,2)
> pc2

Uniquenesses:
   V1    V2    V3    V4    V5    V6
0.182 0.234 0.309 0.282 0.332 0.409
Loadings:
   PC1  PC2
V1 0.90
V2 0.88
V3 0.83
V4      0.85
V5      0.82
V6      0.77
                 PC1   PC2
SS loadings    2.273 1.988
Proportion Var 0.379 0.331
Cumulative Var 0.379 0.710

> round(pc2$loadings %*% t(pc2$loadings),2)

   V1   V2   V3   V4   V5   V6
V1 0.81 0.79 0.75 0.00 0.00 0.00
V2 0.79 0.77 0.73 0.00 0.00 0.00
V3 0.75 0.73 0.69 0.00 0.00 0.00
V4 0.00 0.00 0.00 0.72 0.70 0.65
V5 0.00 0.00 0.00 0.70 0.67 0.63
V6 0.00 0.00 0.00 0.65 0.63 0.59
```

```
> Rresid <- R - pc2$loadings %*% t(pc2$loadin
> round(Rresid,2)

      V1    V2    V3    V4    V5    V6
V1  0.19 -0.07 -0.12  0.00  0.00  0.00
V2 -0.07  0.23 -0.17  0.00  0.00  0.00
V3 -0.12 -0.17  0.31  0.00  0.00  0.00
V4  0.00  0.00  0.00  0.28 -0.14 -0.17
V5  0.00  0.00  0.00 -0.14  0.33 -0.21
V6  0.00  0.00  0.00 -0.17 -0.21  0.41

> resid(pc2)

     V1    V2    V3    V4    V5    V6
V1  0.18
V2 -0.07  0.23
V3 -0.12 -0.17  0.31
V4  0.00  0.00  0.00  0.28
V5  0.00  0.00  0.00 -0.13  0.33
V6  0.00  0.00  0.00 -0.17 -0.21  0.41
```

Try two factors

```
> f2 <- fa(R,2,rotate="none")
> f2
```

```
Factor Analysis using method =  minres
Call: fa(r = R, nfactors = 2, rotate = "none")
Standardized loadings (pattern matrix) based upon correlation matrix
   MR1 MR2   h2   u2
V1 0.9 0.0 0.81 0.19
V2 0.8 0.0 0.64 0.36
V3 0.7 0.0 0.49 0.51
V4 0.0 0.8 0.64 0.36
V5 0.0 0.7 0.49 0.51
V6 0.0 0.6 0.36 0.64

                MR1  MR2
SS loadings     1.94 1.49
Proportion Var 0.32 0.25
Cumulative Var 0.32 0.57

Test of the hypothesis that 2 factors are sufficient.

The degrees of freedom for the null model are
15  and the objective function was  1.96
The degrees of freedom for the model are 4
and the objective function was  0

The root mean square of the residuals (RMSR) is
0
The df corrected root mean square of the residuals is
0
```

```
> resid(f2)
```

```
     V1   V2   V3   V4   V5   V6
V1 0.19
V2 0.00 0.36
V3 0.00 0.00 0.51
V4 0.00 0.00 0.00 0.36
V5 0.00 0.00 0.00 0.00 0.51
V6 0.00 0.00 0.00 0.00 0.00 0.64
```

Add two more variables (with a factor model)

```
#the model
> f <- matrix(c(.9,.8,.7,rep(0,3),.7,rep(0,4),.8,.7,.6,0,.5),ncc
> rownames(f) <- paste("V",seq(1:8),sep="")  #add labels
>  colnames(f) <- c("F1", "F2")
>  R <- f %*% t(f) #create the correlation matrix
>  diag(R) <- 1  #adjust the diagonal of the matrix
>  R
```

	V1	V2	V3	V4	V5	V6	V7	V8
V1	1.00	0.72	0.63	0.00	0.00	0.00	0.63	0.00
V2	0.72	1.00	0.56	0.00	0.00	0.00	0.56	0.00
V3	0.63	0.56	1.00	0.00	0.00	0.00	0.49	0.00
V4	0.00	0.00	0.00	1.00	0.56	0.48	0.00	0.40
V5	0.00	0.00	0.00	0.56	1.00	0.42	0.00	0.35
V6	0.00	0.00	0.00	0.48	0.42	1.00	0.00	0.30
V7	0.63	0.56	0.49	0.00	0.00	0.00	1.00	0.00
V8	0.00	0.00	0.00	0.40	0.35	0.30	0.00	1.00

Factors loadings do not change, component loadings do

```
> R

     V1   V2   V3   V4   V5   V6   V7
V8
V1 1.00 0.72 0.63 0.00 0.00 0.00 0.63 0.00
V2 0.72 1.00 0.56 0.00 0.00 0.00 0.56 0.00
V3 0.63 0.56 1.00 0.00 0.00 0.00 0.49 0.00
V4 0.00 0.00 0.00 1.00 0.56 0.48 0.00 0.40
V5 0.00 0.00 0.00 0.56 1.00 0.42 0.00 0.35
V6 0.00 0.00 0.00 0.48 0.42 1.00 0.00 0.30
V7 0.63 0.56 0.49 0.00 0.00 0.00 1.00 0.00
V8 0.00 0.00 0.00 0.40 0.35 0.30 0.00 1.00

> f2 <- factanal(covmat=R, factors=2)
> f2

Call:
factanal(factors = 2, covmat = R)
Uniquenesses:
  V1   V2   V3   V4   V5   V6   V7   V8
0.19 0.36 0.51 0.36 0.51 0.64 0.51 0.75
Loadings:
   Factor1 Factor2
V1 0.9
V2 0.8
V3 0.7
V4         0.8
V5         0.7
V6         0.6
V7 0.7
```

```
pc2 <- principal(R,2)
pc2

Uniquenesses:
   V1    V2    V3    V4    V5    V6
V7    V8
0.194 0.271 0.367 0.311 0.379 0.468 0.367 0.57
Loadings:
   PC1  PC2
V1 0.90
V2 0.85
V3 0.80
V4      0.83
V5      0.79
V6      0.73
V7 0.80
V8      0.65
                 PC1   PC2
SS loadings     2.812 2.268
Proportion Var  0.352 0.284
Cumulative Var  0.352 0.635
```

Simple Structure

The original solution of a principal components or principal axes factor analysis is a set of vectors that best account for the observed covariance or correlation matrix, and where the components or factors account for progressively less and less variance. But such a solution, although maximally efficient in describing the data, is rarely easy to interpret. But what makes a structure easy to interpret? Thurstone's answer, *simple structure*, consists of five rules (Thurstone, 1947, p 335):

> *(1) Each row of the oblique factor matrix* **V** *should have at least one zero.*
> *(2) For each column p of the factor matrix* **V** *there should be a distinct set of r linearly independent tests whose factor loadings* v_{ip} *are zero.*
> *(3) For every pair of columns of* **V** *there should be several tests whose entries* v_{ip} *vanish in one column but not in the other.*
> *(4) For every pair of columns of* **V***, a large proportion of the tests should have zero entries in both columns. This applies to factor problems with four or five or more common factors.*
> *(5) For every pair of columns there should preferably be only a small number of tests with non-vanishing entries in both columns.*

Thurstone proposed to rotate the original solution to achieve simple structure.

Simple structure

A matrix is said to be *rotated* if it is multiplied by a matrix of orthogonal vectors that preserves the communalities of each variable. Just as the original matrix was orthogonal, so is the rotated solution. For two factors, the *rotation* matrix **T** will rotate the two factors θ radians in a counterclockwise direction.

$$T = \begin{pmatrix} cos(\theta) & sin(\theta) \\ -sin(\theta) & cos(\theta) \end{pmatrix} \tag{12}$$

Generalizing equation 12 to larger matrices is straight forward:

$$T = \begin{pmatrix} 1 & ... & 0 & ... & 0 & ... & 0 \\ 0 & ... & cos(\theta) & ... & sin(\theta) & ... & 0 \\ ... & ... & 0 & 1 & 0 & ... & 0 \\ 0 & ... & -sin(\theta) & ... & cos(\theta) & ... & 0 \\ ... & ... & 0 & ... & 0 & ... & ... \\ 0 & ... & 0 & ... & 0 & ... & 1 \end{pmatrix}. \tag{13}$$

Rotating to simple structure

When **F** is post-multiplied by **T**, **T** will rotate the i^{th} and j^{th} columns of **F** by θ radians in a counterclockwise direction.

$$\mathbf{F}_r = \mathbf{FT} \tag{14}$$

The `factor.rotate` function from the **psych** package will do this rotation for arbitrary angles (in degrees) for any pairs of factors. This is useful if there is a particular rotation that is desired. An entire package devoted to rotations is the *GPArotation* by Robert Jennrich (Jennrich, 2004).

Analytic Simple Structure

As pointed out by Carroll (1953) when discussing Thurstone's (1947) simple structure as a rotational criterion "it is obvious that there could hardly be any single mathematical expression which could embody all these characteristics." (p 24). Carroll's solution to this was to minimize the sum of the inner products of the squared (rotated) loading matrix. An alternative, discussed by Ferguson (1954) is to consider the *parsimony* of a group of n tests with r factors to be defined as the average parsimony of the individual tests (I_j) where

$$I_j = \sum_m^r a_{jm}^4 \tag{15}$$

(the squared communality) and thus the average parsimony is

$$I_. = n^{-1} \sum_j^n \sum_m^r a_{jm}^4$$

and to choose a rotation that maximizes parsimony.

Rotation to parsimony

Parsimony as defined in equation 15 is a function of the variance as well as the mean of the squared loadings of a particular test on all the factors. For fixed communality h^2, it will be maximized if all but one loading is zero; a variable's parsimony will be maximal if one loading is 1.0 and the rest are zero. In path notation, parsimony is maximized if one and only one arrow is associated with a variable. This criterion, as well as the criterion of maximum variance taken over factors has been operationalized as the *quartimax* criterion by Neuhaus and Wrigley (1954). As pointed out by Kaiser (1958), the criterion can rotate towards a solution with one general factor, ignoring other, smaller factors.

Varimax and alternatives

If a general factor is not desired, an alternative measure of the parsimony of a factor, similar to equation 15 is to maximize the variance of the squared loadings taken over items instead of over factors. This, the *varimax* criterion was developed by Kaiser (1958) to avoid the tendency to yield a general factor. Both of these standard rotations as well as many others are available in the **GPArotation** package of rotations and transformations which uses the *Gradient Projection Algorithms* developed by Jennrich (2001, 2002, 2004).

Harmon 8 physical measures

```
> data(Harman23.cor)
> lower.mat(Harman23.cor$cov)
```

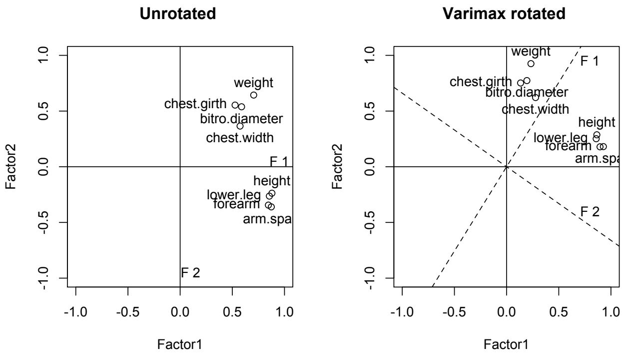
```
               heght arm.s forrm lwr.l weght btr.d chst.g chst.w
height         1.00
arm.span       0.85  1.00
forearm        0.80  0.88  1.00
lower.leg      0.86  0.83  0.80  1.00
weight         0.47  0.38  0.38  0.44  1.00
bitro.diameter 0.40  0.33  0.32  0.33  0.76  1.00
chest.girth    0.30  0.28  0.24  0.33  0.73  0.58  1.00
chest.width    0.38  0.42  0.34  0.36  0.63  0.58  0.54   1.00
```

Two solutions – loadings change, goodness of fits do not

```
> f2 <- fa(Harman23.cor$cov,2,rotate="none")
> f2


Factor Analysis using method =  minres
Call: fa(r = Harman23.cor$cov, nfactors = 2,
    rotate = "none")
Standardized loadings (pattern matrix)
                MR1   MR2   h2   u2
height          0.89 -0.19 0.83 0.17
arm.span        0.89 -0.31 0.89 0.11
forearm         0.86 -0.30 0.83 0.17
lower.leg       0.87 -0.22 0.80 0.20
weight          0.67  0.67 0.89 0.11
bitro.diameter  0.56  0.58 0.65 0.35
chest.girth     0.50  0.59 0.59 0.41
chest.width     0.56  0.40 0.47 0.53

                MR1  MR2
SS loadings     4.40 1.56
Proportion Var  0.55 0.19
Cumulative Var  0.55 0.74
Test of the hypothesis that 2 factors are sufficient.
The root mean square of the residuals (RMSR) is
0.02
The df corrected root mean square of the residuals
             is  0.03
Fit based upon off diagonal values = 1
```

```
> f2 <- fa(Harman23.cor$cov,2,rotate="varimax
> f2


Factor Analysis using method =  minres
Call: fa(r = Harman23.cor$cov, nfactors = 2,
    rotate = "varimax")
Standardized loadings (pattern matrix)
                MR1  MR2  h2   u2
height          0.86 0.30 0.83 0.17
arm.span        0.92 0.20 0.89 0.11
forearm         0.89 0.19 0.83 0.17
lower.leg       0.86 0.26 0.80 0.20
weight          0.22 0.92 0.89 0.11
bitro.diameter  0.18 0.78 0.65 0.35
chest.girth     0.12 0.76 0.59 0.41
chest.width     0.27 0.63 0.47 0.53

                MR1  MR2
SS loadings     3.30 2.66
Proportion Var  0.41 0.33
Cumulative Var  0.41 0.74

The root mean square of the residuals (RMSR)
    is  0.02
The df corrected root mean square of the resi
    is  0.03
```

Alternative rotations



Oblique transformations

Many of those who use factor analysis use it to identify theoretically meaningful constructs which they have no reason to believe are orthogonal. This has lead to the use of *oblique transformations* which allow the factors to be correlated. Although the term rotation is sometimes used for both *orthogonal* and *oblique* solutions, in the oblique case the factor matrix is not rotated so much as *transformed*.

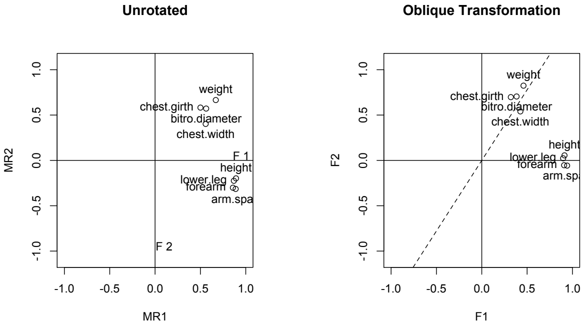
Oblique transformations lead to the distinction between the *factor pattern* and *factor structure* matrices. The *factor pattern* matrix is the set of *regression weights* (loadings) from the latent factors to the observed variables. The *factor structure* matrix is the matrix of *correlations* between the factors and the observed variables. If the factors are uncorrelated, structure and pattern are identical. But, if the factors are correlated, the structure matrix (**S**) is the pattern matrix (**F**) times the factor intercorrelations ϕ

$\mathbf{S} = \mathbf{F}\phi <=> \mathbf{F} = \mathbf{S}\phi^{-1}$:

An oblique transformation of the Harman 8 physical variables

```
> f2t <- fa(Harman23.cor$cov,2,rotate="oblimin",n.obs=305)
> print(f2t)

Factor Analysis using method =  minres
Call: fa(r = Harman23.cor$cov, nfactors = 2, rotate = "oblimin", n
                item   MR1   MR2   h2   u2
height             1  0.87  0.08 0.84 0.16
arm.span           2  0.96 -0.05 0.89 0.11
forearm            3  0.93 -0.04 0.83 0.17
lower.leg          4  0.88  0.04 0.81 0.19
weight             5  0.01  0.94 0.89 0.11
bitro.diameter     6  0.00  0.80 0.64 0.36
chest.girth        7 -0.06  0.79 0.59 0.41
chest.width        8  0.13  0.62 0.47 0.53

                 MR1  MR2
SS loadings     3.37 2.58
Proportion Var  0.42 0.32
Cumulative Var  0.42 0.74

 With factor correlations of
     MR1  MR2
MR1 1.00 0.46
MR2 0.46 1.00
```

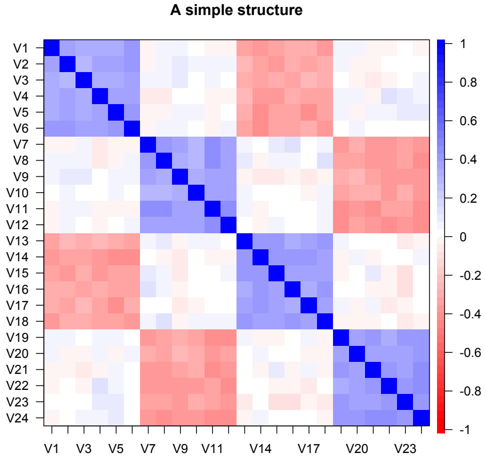
Oblique Transformations



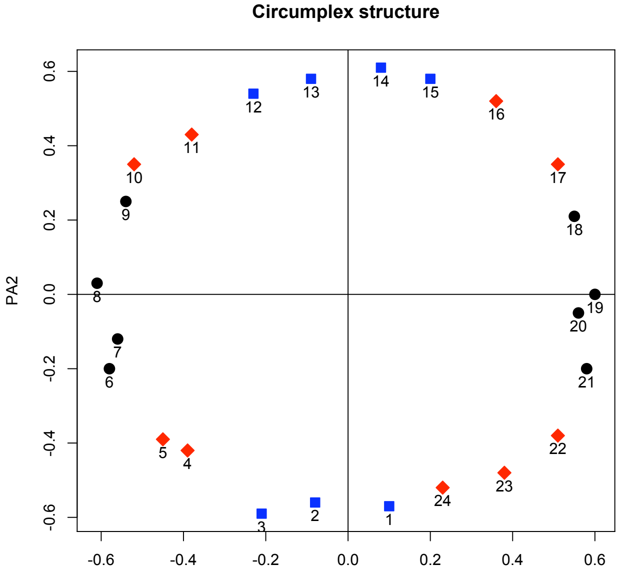
Another way to show simple structure

R code

```
simp24 <- sim.item(24,circum=FALSE)
cor.plot(cor(simp24),main="A simple structure")
```

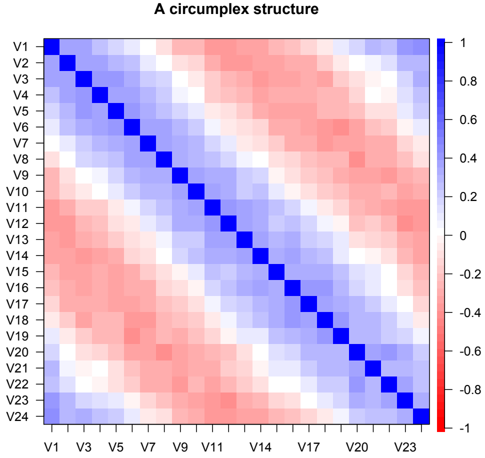


A circumplex is one alternative to simple structure



Another way of showing a circumplex – cor.plot

```
> circ24 <- (24,circum=TRUE)
> cor.plot(cor(circ24),main="A circumplex structure")
```



The Thurstone 9 variable problem

```
> lower.mat(Thurstone)
```

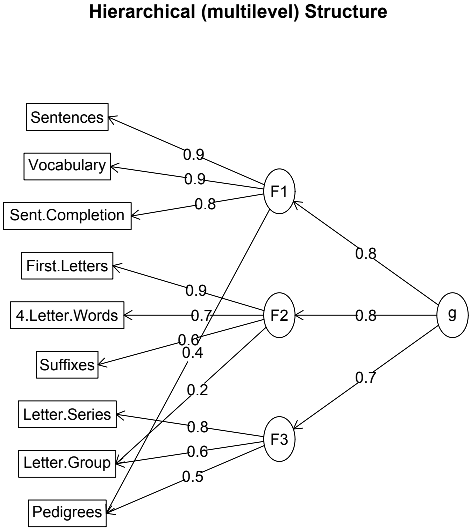
```
                Sntnc Vcblr Snt.C Frs.L 4.L.W Sffxs Ltt.S Pdgrs Lt
Sentences       1.00
Vocabulary      0.83  1.00
Sent.Completion 0.78  0.78  1.00
First.Letters   0.44  0.49  0.46  1.00
4.Letter.Words  0.43  0.46  0.42  0.67  1.00
Suffixes        0.45  0.49  0.44  0.59  0.54  1.00
Letter.Series   0.45  0.43  0.40  0.38  0.40  0.29  1.00
Pedigrees       0.54  0.54  0.53  0.35  0.37  0.32  0.56  1.00
Letter.Group    0.38  0.36  0.36  0.42  0.45  0.32  0.60  0.45
1.00
```

Three factors from Thurstone 9 variables

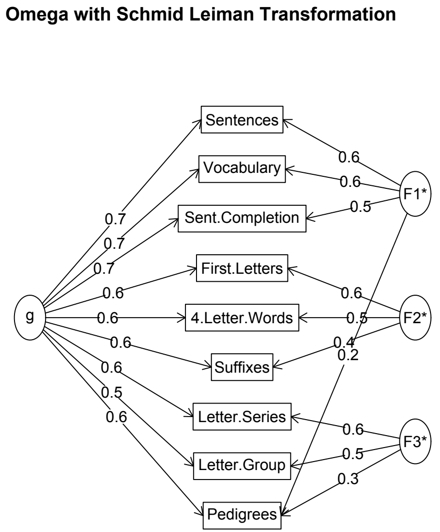
```
> f3 <- fa(Thurstone,3)
> f3
```

```
Factor Analysis using method =  minres
Call: fa(r = Thurstone, nfactors = 3)
Standardized loadings (pattern matrix) based upon correlation matrix
                  MR1   MR2   MR3   h2   u2
Sentences         0.91 -0.04  0.04 0.82 0.18
Vocabulary        0.89  0.06 -0.03 0.84 0.16
Sent.Completion   0.83  0.04  0.00 0.73 0.27
First.Letters     0.00  0.86  0.00 0.73 0.27
4.Letter.Words   -0.01  0.74  0.10 0.63 0.37
Suffixes          0.18  0.63 -0.08 0.50 0.50
Letter.Series     0.03 -0.01  0.84 0.72 0.28
Pedigrees         0.37 -0.05  0.47 0.50 0.50
Letter.Group     -0.06  0.21  0.64 0.53 0.47

                MR1  MR2  MR3
SS loadings     2.64 1.86 1.50
Proportion Var  0.29 0.21 0.17
Cumulative Var  0.29 0.50 0.67

 With factor correlations of
     MR1  MR2  MR3
MR1 1.00 0.59 0.54
MR2 0.59 1.00 0.52
MR3 0.54 0.52 1.00
```

A hierarchical/multilevel solution to the Thurstone 9 variables



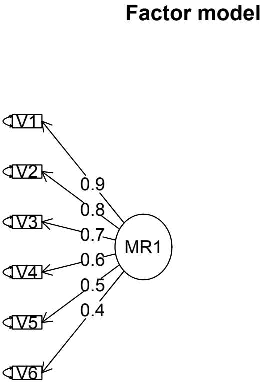
A bifactor solution using the Schmid Leiman transformation

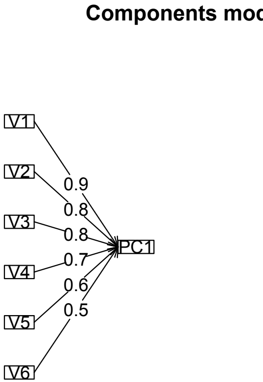


How many factors – no right answer, one wrong answer

1. Statistical
   - Extracting factors until the χ^2 of the residual matrix is not significant.
   - Extracting factors until the change in χ^2 from factor n to factor n+1 is not significant.
2. Rules of Thumb
   - Parallel Extracting factors until the eigenvalues of the real data are less than the corresponding eigenvalues of a random data set of the same size (*parallel analysis*)
   - Plotting the magnitude of the successive eigenvalues and applying the *scree test*.
3. Interpretability
   - Extracting factors as long as they are interpretable.
   - Using the *Very Simple Structure* Criterion (VSS)
   - Using the Minimum Average Partial criterion (MAP).
4. Eigen Value of 1 rule

The factor model versus the components model





Simulation as tool for creating structure

1. The problem of all real data sets is that we do not know the "true" structure.
2. But by artificially creating simulated structures, we know "truth"
3. Simulate two factors with simple structure
4. Simulate two factors with circumplex structure
5. In both cases, ask now many factors are in the data and what is the best fit to the data

Simulate 2 factor data

Using the sim.item function

R code

```
set.seed(42)  #to generate a reproducible example
my.data <- sim.item(12)
 my.cor <- lowerCor(my.data)
```

```
> set.seed(42)  #to generate a reproducible example
>  my.data <- sim.item(12)
>   my.cor <- lowerCor(my.data)
    V1    V2    V3    V4    V5    V6    V7    V8    V9    V10   V11   V12
V1   1.00
V2   0.36  1.00
V3   0.38  0.37  1.00
V4  -0.01 -0.04 -0.01  1.00
V5   0.05 -0.02  0.01  0.34  1.00
V6   0.03  0.01  0.01  0.37  0.35  1.00
V7  -0.35 -0.37 -0.38 -0.09 -0.01 -0.05  1.00
V8  -0.40 -0.34 -0.39  0.00  0.08  0.11  0.34  1.00
V9  -0.41 -0.36 -0.32  0.00  0.02 -0.03  0.32  0.39  1.00
V10  0.06  0.07  0.01 -0.33 -0.32 -0.39 -0.04 -0.11 -0.06  1.00
V11  0.02  0.03  0.05 -0.37 -0.35 -0.32  0.02 -0.12 -0.01  0.41  1.00
V12  0.01  0.01 -0.11 -0.31 -0.30 -0.33  0.08 -0.02  0.00  0.36  0.39  1.00
```

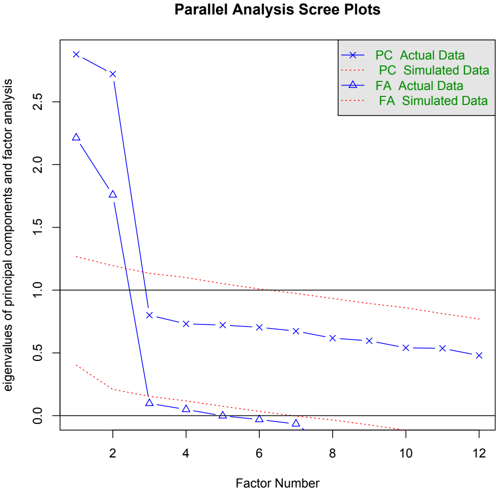
Multiple ways to determine how many factors are in the data

No one answer. Many are good, one should be avoided.

1. Statistical tests
   - χ^2 test of residuals (sensitive to sample size and non-normality of data)
   - χ^2 test of change from nf=n to nf=n+1 (sensitive to sample size)
   - RMSEA, BIC, AIC, SABIC are not as sensitive to sample size, but are to non-normality
2. Rules of Thumb
   - Scree Test of eigen values (Cattell, 1966)
   - Minimum Average Partial (MAP) (Velicer, 1976)
   - Very Simple Structure (Revelle and Rocklin, 1979)
   - Parallel Analysis of random data (Horn, 1965)
   - As many as can be interpreted
3. One test to avoid: Eigen value of 1 (Many programs default to this)

How many factors in my.cor

```
> fa.parallel(my.cor,n.obs=500)
Parallel analysis suggests that the number of factors =  2
                and the number of components =  2
```



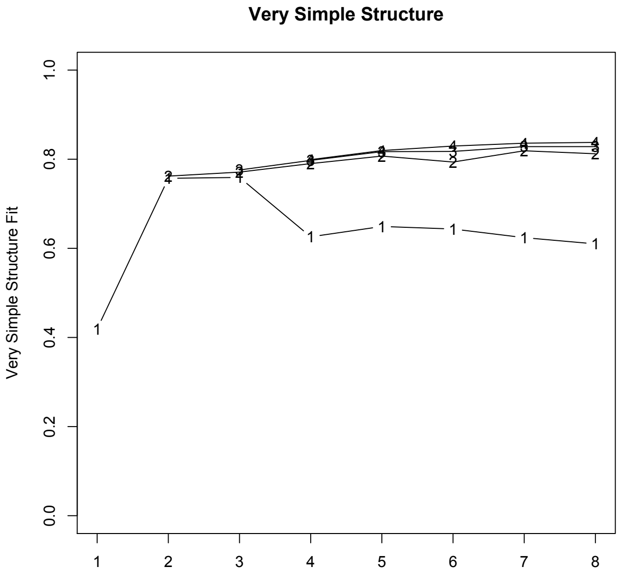
Try Very Simple Structure as well as MAP

```
> vss(my.cor,n.obs=500)

Very Simple Structure
Call: VSS(x = x, n = n, rotate = rotate, diagonal = diagonal, fm =
    n.obs = n.obs, plot = plot, title = title)
VSS complexity 1 achieves a maximimum of 0.74  with  3  factors
VSS complexity 2 achieves a maximimum of 0.8  with  8  factors

The Velicer MAP criterion achieves a minimum of 0.02  with  2
factors

Velicer MAP
[1] 0.05 0.02 0.03 0.05 0.07 0.10 0.13 0.19

Very Simple Structure Complexity 1
[1] 0.39 0.74 0.74 0.63 0.70 0.66 0.58 0.57

Very Simple Structure Complexity 2
[1] 0.00 0.75 0.76 0.78 0.79 0.79 0.80 0.80
```

Even more VSS output

```
> my.vss <- vss(my.cor, n=12)
> my.vss


n.obs was not specified and was arbitrarily set to 1000.  This only affects the chi squa
Very Simple Structure
Call: vss(x = my.cor, n = 12)
VSS complexity 1 achieves a maximimum of 0.74  with  3  factors
VSS complexity 2 achieves a maximimum of 0.81  with  10  factors

The Velicer MAP achieves a minimum of NA  with  2  factors
BIC achieves a minimum of  NA  with  2  factors
Sample Size adjusted BIC achieves a minimum of  NA  with  2  factors

Statistics by number of factors
   vss1 vss2   map dof   chisq     prob sqresid  fit RMSEA  BIC  SABIC complex
1  0.39 0.00 0.055  54 1.3e+03 1.5e-229    12.2 0.39 0.150  894 1065.3     1.0
2  0.74 0.75 0.021  43 1.1e+02  9.0e-08     5.0 0.75 0.040 -187  -50.6     1.0
3  0.74 0.76 0.033  33 7.6e+01  3.1e-05     4.7 0.76 0.036 -152  -47.2     1.1
4  0.63 0.78 0.049  24 5.3e+01  6.0e-04     4.2 0.79 0.035 -113  -36.7     1.2
5  0.70 0.79 0.069  16 3.4e+01  5.6e-03     3.9 0.80 0.034  -77  -25.8     1.3
6  0.66 0.79 0.096   9 1.1e+01  2.6e-01     3.5 0.82 0.016  -51  -22.3     1.3
7  0.58 0.80 0.130   3 1.8e+00  6.2e-01     3.3 0.83 0.000  -19   -9.4     1.5
8  0.57 0.80 0.186  -2 8.4e-02       NA     2.8 0.86    NA   NA     NA     1.5
9  0.43 0.74 0.278  -6 8.8e-07       NA     2.6 0.87    NA   NA     NA     1.9
10 0.62 0.81 0.456  -9 1.7e-09       NA     3.1 0.85    NA   NA     NA     1.8
11 0.55 0.81 1.000 -11 0.0e+00       NA     3.0 0.85    NA   NA     NA     1.8
12 0.55 0.81    NA -12 0.0e+00       NA     3.0 0.85    NA   NA     NA     1.8
>
```

Examine the output



But if we had more subjects?

R code

```
set.seed(42)  #to generate a reproducible example
my.data <- sim.item(12, nsub=10000)
 my.cor <- lowerCor(my.data)
```

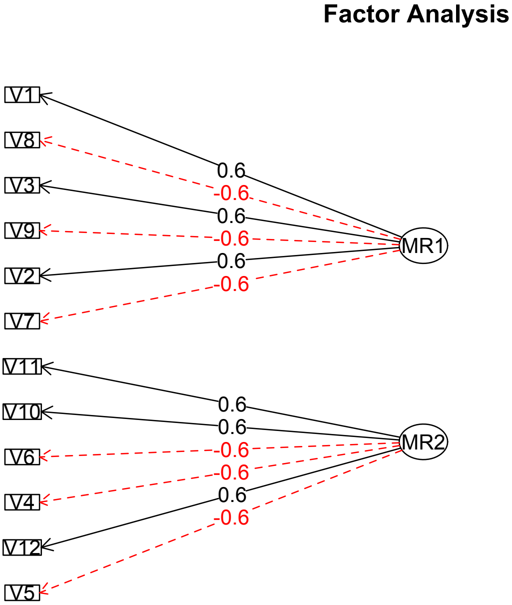
Extract 2 factors –part 1

```
> fa(my.cor,2,n.obs=500)
 Factor Analysis using method =  minres
 Call: fa(r = my.cor, nfactors = 2, n.obs = 500)
 Standardized loadings based upon correlation matrix
       MR1   MR2   h2   u2
V1    0.64 -0.02 0.41 0.59
V2    0.59  0.02 0.35 0.65
V3    0.61 -0.04 0.37 0.63
V4    0.03 -0.58 0.34 0.66
V5    0.01 -0.55 0.30 0.70
V6    0.03 -0.60 0.36 0.64
V7   -0.58  0.08 0.34 0.66
V8   -0.62 -0.10 0.40 0.60
V9   -0.59  0.00 0.35 0.65
V10   0.07  0.61 0.39 0.61
V11   0.03  0.63 0.39 0.61
V12  -0.06  0.57 0.33 0.67

                MR1  MR2
SS loadings     2.21 2.12
 Proportion Var 0.18 0.18
 Cumulative Var 0.18 0.36
```

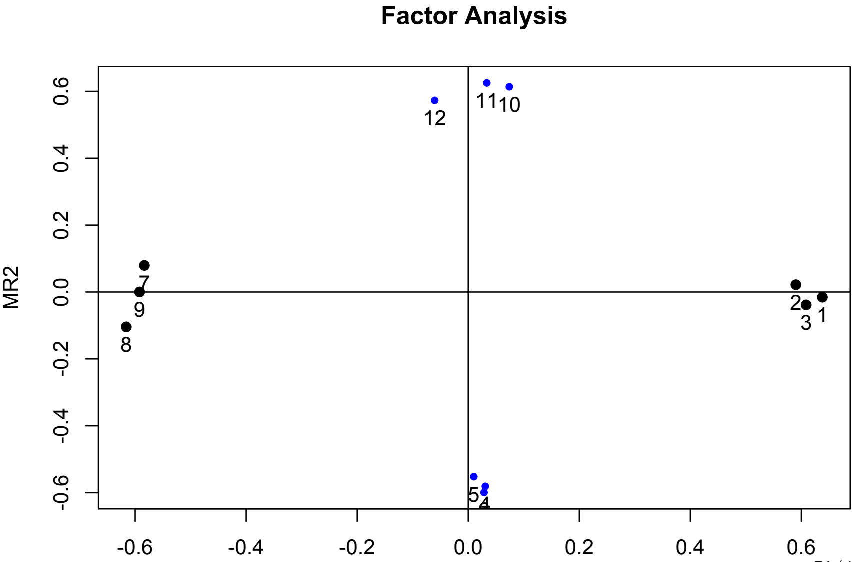
2 artificial factors part 2

```
 With factor correlations of
     MR1  MR2
MR1 1.00 0.04
MR2 0.04 1.00

Test of the hypothesis that 2 factors are sufficient.

The degrees of freedom for the null model are  66  and the objecti
2.52 with Chi Square of  1246.71
The degrees of freedom for the model are 43  and the objective fun
0.11

The root mean square of the residuals is  0.02
The df corrected root mean square of the residuals is  0.03
The number of observations was  500  with Chi Square =  54.56
with prob <  0.11

Tucker Lewis Index of factoring reliability =  0.985
RMSEA index =  0.024  and the 90 % confidence intervals are
0.023 0.026
BIC =  -212.67
Fit based upon off diagonal values = 0.99
Measures of factor score adequacy
                                              MR1  MR2
```

The factor diagram shows the structure



The factor plot also shows the structure



An alternative data structure is a circumplex

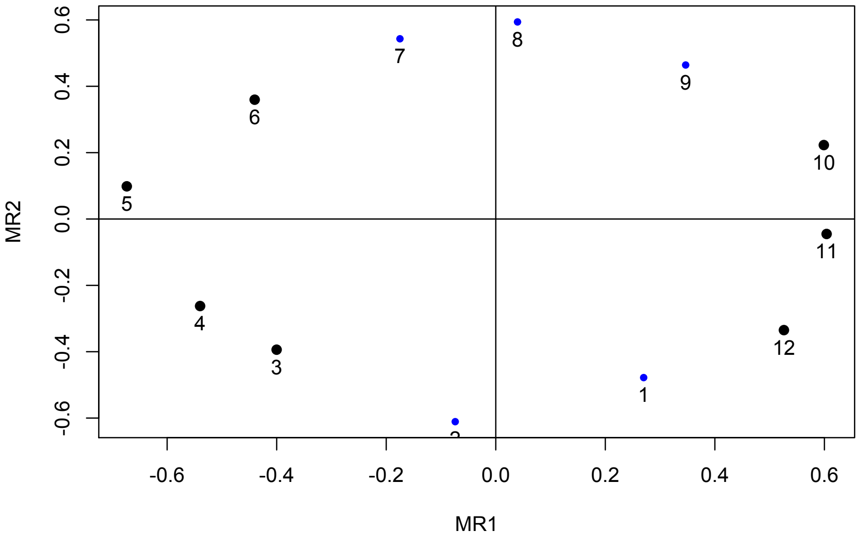
Seen in measures of emotion, interpersonal problems

```
> circ <- sim.circ(12)
> f2 <- fa(circ,2)
> fa.plot(f2,title="A circumplex structure")
> fa.diagram(f2,simple=FALSE,main="A circumplex structure")
```

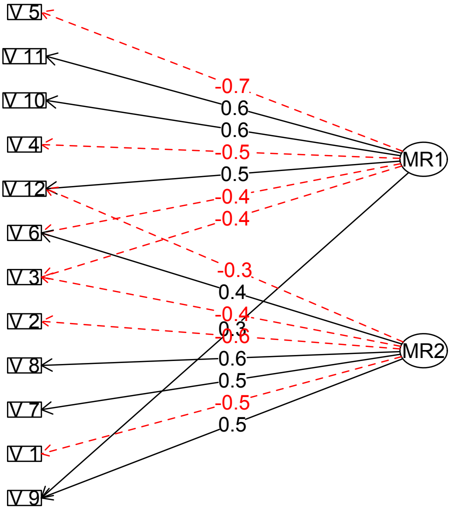
```
Factor Analysis using method = minres
Call: fa(r = circ, nfactors = 2)
Standardized loadings (pattern matrix)
     MR1   MR2   h2   u2  com
1   0.27 -0.48 0.30 0.70 1.6
2  -0.07 -0.61 0.38 0.62 1.0
3  -0.40 -0.39 0.32 0.68 2.0
4  -0.54 -0.26 0.36 0.64 1.4
5  -0.67  0.10 0.46 0.54 1.0
6  -0.44  0.36 0.32 0.68 1.9
7  -0.17  0.54 0.32 0.68 1.2
8   0.04  0.59 0.35 0.65 1.0
9   0.35  0.46 0.34 0.66 1.9
10  0.60  0.22 0.41 0.59 1.3
11  0.60 -0.05 0.37 0.63 1.0
12  0.53 -0.33 0.38 0.62 1.7
```

```
                                      MR1 MR2
SS loadings           2.33 1.99
Proportion Var        0.19 0.17
Cumulative Var        0.19 0.36
Proportion Explained  0.54 0.46
Cumulative Proportion 0.54 1.00

 With factor correlations of
     MR1  MR2
MR1 1.00 0.01
MR2 0.01 1.00

Mean item complexity = 1.4
```

A circumplex structure



A circumplex structure



9 cognitive tests from Thurstone

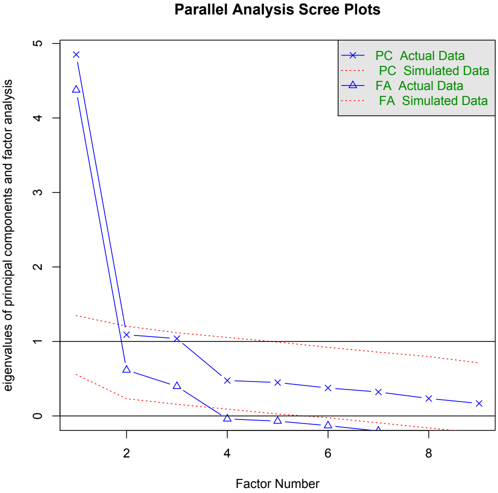
```
> lowerMat(Thurstone)

                Sntnc Vcblr Snt.C Frs.L 4.L.W Sffxs Ltt.S Pdgrs Lt
Sentences       1.00
Vocabulary      0.83  1.00
Sent.Completion 0.78  0.78  1.00
First.Letters   0.44  0.49  0.46  1.00
4.Letter.Words  0.43  0.46  0.42  0.67  1.00
Suffixes        0.45  0.49  0.44  0.59  0.54  1.00
Letter.Series   0.45  0.43  0.40  0.38  0.40  0.29  1.00
Pedigrees       0.54  0.54  0.53  0.35  0.37  0.32  0.56  1.00
Letter.Group    0.38  0.36  0.36  0.42  0.45  0.32  0.60  0.45
1.00
```

Choicies in dimension reduction

1. Factors or Components
   - Components are maximally efficient to describe the data (including the error)
   - Factors model the shared common variance, but not the errors
2. How many factors to extract
3. Which factor extraction technique
   - maximum likelihood is "optimal" but not if model of mean residual $= 0$ is false
   - minres minimizes the residuals using Ordinary Least Squares – fits are almost as good as mle
   - principal axis is an iterative procedure used by SPSS
   - minchi minimizes the sample size weighted residual if number of pairwise observations differ
4. rotation (orthogonal) or transformation (oblique)
   - Orthogonal rotations (e.g.,) varimax, quartimax, bifactor
   - Oblique transformations (e.g.,) oblimin, oblimax, geomin, biquaritimin,
   - Higher order structures (e.g.,) schmid leiman, bifactor

9 mental tests from Thurstone

```
data(bifactor)
fa.parallel(Thurstone,n.obs=213)
```



Exract 3 factors

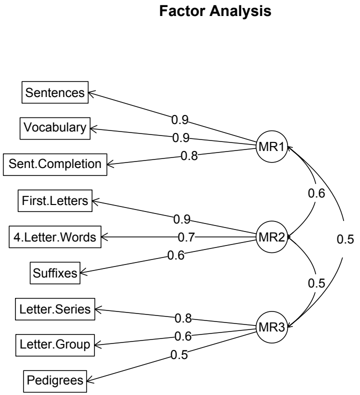
```
> fa3 <-  fa(Thurstone,3,n.obs=213)
> fa3
Factor Analysis using method =  minres
Call: fa(r = Thurstone, nfactors = 3, n.obs = 213)
Standardized loadings based upon correlation matrix
                   MR1   MR2   MR3   h2   u2
Sentences         0.91 -0.04  0.04 0.82 0.18
Vocabulary        0.89  0.06 -0.03 0.84 0.16
Sent.Completion   0.83  0.04  0.00 0.73 0.27
First.Letters     0.00  0.86  0.00 0.73 0.27
4.Letter.Words   -0.01  0.74  0.10 0.63 0.37
Suffixes          0.18  0.63 -0.08 0.50 0.50
Letter.Series     0.03 -0.01  0.84 0.72 0.28
Pedigrees         0.37 -0.05  0.47 0.50 0.50
Letter.Group     -0.06  0.21  0.64 0.53 0.47

                 MR1  MR2  MR3
SS loadings     2.64 1.86 1.50
Proportion Var  0.29 0.21 0.17
Cumulative Var  0.29 0.50 0.67
```

Thurstone 3 factors part 2

```
 With factor correlations of
      MR1  MR2  MR3
MR1 1.00 0.59 0.54
MR2 0.59 1.00 0.52
MR3 0.54 0.52 1.00

Test of the hypothesis that 3 factors are sufficient.

The degrees of freedom for the null model are  36  and the objecti
5.2 with Chi Square of  1081.97
The degrees of freedom for the model are 12  and the objective fun
0.01

The root mean square of the residuals is  0
The df corrected root mean square of the residuals is  0.01
The number of observations was  213  with Chi Square =  2.82
with prob <  1

Tucker Lewis Index of factoring reliability =  1.027
RMSEA index =  0  and the 90 % confidence intervals are  0 0.023
BIC =  -61.51
Fit based upon off diagonal values = 1
Measures of factor score adequacy
                                             MR1  MR2  MR3
```

A factor diagram

```
fa3 <- fa(Thurstone,3,n.obs=213)
```

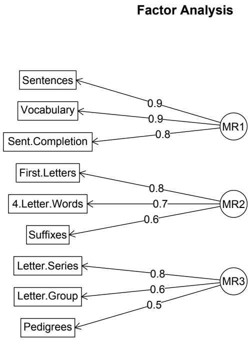


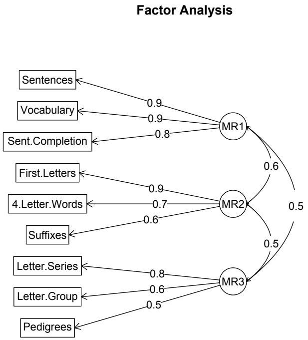
Thurstone, 3 factors Varimax rotated

```
> v3 <- fa(Thurstone,3,rotate="Varimax",n.obs=213)
> fa.diagram(v3)
> v3
Factor Analysis using method =  minres
Call: fa(r = Thurstone, nfactors = 3, n.obs = 213, rotate = "Varim
Standardized loadings based upon correlation matrix
                  MR1  MR2  MR3   h2   u2
Sentences        0.86 0.20 0.22 0.82 0.18
Vocabulary       0.85 0.27 0.18 0.84 0.16
Sent.Completion  0.80 0.24 0.19 0.73 0.27
First.Letters    0.29 0.78 0.20 0.73 0.27
4.Letter.Words   0.27 0.70 0.26 0.63 0.37
Suffixes         0.36 0.60 0.10 0.50 0.50
Letter.Series    0.28 0.18 0.78 0.72 0.28
Pedigrees        0.48 0.15 0.50 0.50 0.50
Letter.Group     0.20 0.32 0.62 0.53 0.47

                 MR1  MR2  MR3
SS loadings     2.73 1.78 1.48
Proportion Var  0.30 0.20 0.16
Cumulative Var  0.30 0.50 0.67
```

Compare the two solutions

```
> v3 <- fa(Thurstone,3,rotate="Varimax",n.obs=213)
> fa.diagram(v3)
```





Factor Congruence is a measure of how much the factor loadings agree

1. Developed by Burt (1948) but known as the "Tucker coefficient" it is just $\frac{\Sigma(F_1 F_2)}{\sqrt{\Sigma_{{F_1}^2} \Sigma_{{F_2}^2}}}$

```
> factor.congruence(list(v3,f3))
```

	MR1	MR2	MR3	MR1	MR2	MR3
MR1	1.00	0.64	0.63	0.95	0.30	0.28
MR2	0.64	1.00	0.62	0.41	0.92	0.28
MR3	0.63	0.62	1.00	0.41	0.36	0.90
MR1	0.95	0.41	0.41	1.00	0.06	0.09
MR2	0.30	0.92	0.36	0.06	1.00	0.08
MR3	0.28	0.28	0.90	0.09	0.08	1.00

Principal Components of the Thurstone data set

```
> p3 <- principal(Thurstone,3)
> p3


Principal Components Analysis
Call: principal(r = Thurstone, nfactors = 3)
Standardized loadings (pattern matrix) based upon correlation matrix
                 RC1  RC2  RC3   h2   u2
Sentences        0.86 0.24 0.23 0.86 0.14
Vocabulary       0.85 0.31 0.19 0.86 0.14
Sent.Completion  0.85 0.26 0.19 0.83 0.17
First.Letters    0.23 0.82 0.23 0.78 0.22
4.Letter.Words   0.18 0.79 0.30 0.75 0.25
Suffixes         0.31 0.77 0.06 0.70 0.30
Letter.Series    0.25 0.16 0.83 0.78 0.22
Pedigrees        0.53 0.08 0.61 0.67 0.33
Letter.Group     0.10 0.31 0.80 0.75 0.25

                       RC1  RC2  RC3
SS loadings            2.73 2.25 1.99
Proportion Var         0.30 0.25 0.22
Cumulative Var         0.30 0.55 0.78
Proportion Explained   0.39 0.32 0.29
Cumulative Proportion  0.39 0.71 1.00

Test of the hypothesis that 3 components are sufficient.

The degrees of freedom for the null model are  36  and the objective function was
5.2
The degrees of freedom for the model are 12  and the objective function was  0.62

Fit based upon off diagonal values = 0.98
```

R has many built in data sets

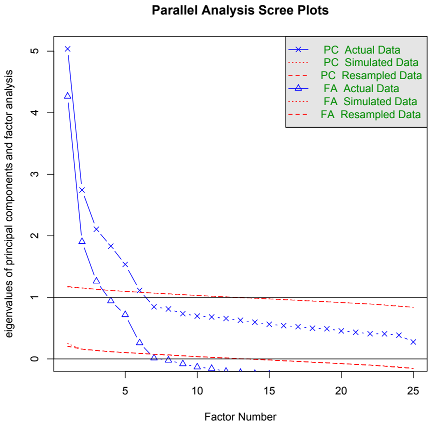
- data(bfi)
- 25 personality items from the Big 5
  - Collected as part of the SAPA project
- Thought to represent 5 dimensions
  - Agreeableness
  - Extraversion
  - Conscientiousness
  - Openness
  - Neuroticism

Describe the Big 5

```
> data(bfi)
> describe(bfi)
           var    n  mean    sd median trimmed   mad min max range  skew kurtosis
se
A1           1 2784  2.41  1.41      2    2.23  1.48   1   6     5  0.83    -0.31 0.03
A2           2 2773  4.80  1.17      5    4.98  1.48   1   6     5 -1.12     1.05 0.02
A3           3 2774  4.60  1.30      5    4.79  1.48   1   6     5 -1.00     0.44 0.02
A4           4 2781  4.70  1.48      5    4.93  1.48   1   6     5 -1.03     0.04 0.03
A5           5 2784  4.56  1.26      5    4.71  1.48   1   6     5 -0.85     0.16 0.02
C1           6 2779  4.50  1.24      5    4.64  1.48   1   6     5 -0.85     0.30 0.02
C2           7 2776  4.37  1.32      5    4.50  1.48   1   6     5 -0.74    -0.14 0.03
C3           8 2780  4.30  1.29      5    4.42  1.48   1   6     5 -0.69    -0.13 0.02
C4           9 2774  2.55  1.38      2    2.41  1.48   1   6     5  0.60    -0.62 0.03
C5          10 2784  3.30  1.63      3    3.25  1.48   1   6     5  0.07    -1.22 0.03
E1          11 2777  2.97  1.63      3    2.86  1.48   1   6     5  0.37    -1.09 0.03
E2          12 2784  3.14  1.61      3    3.06  1.48   1   6     5  0.22    -1.15 0.03
E3          13 2775  4.00  1.35      4    4.07  1.48   1   6     5 -0.47    -0.47 0.03
E4          14 2791  4.42  1.46      5    4.59  1.48   1   6     5 -0.82    -0.30 0.03
E5          15 2779  4.42  1.33      5    4.56  1.48   1   6     5 -0.78    -0.09 0.03
N1          16 2778  2.93  1.57      3    2.82  1.48   1   6     5  0.37    -1.01 0.03
N2          17 2779  3.51  1.53      4    3.51  1.48   1   6     5 -0.08    -1.05 0.03
N3          18 2789  3.22  1.60      3    3.16  1.48   1   6     5  0.15    -1.18 0.03
N4          19 2764  3.19  1.57      3    3.12  1.48   1   6     5  0.20    -1.09 0.03
N5          20 2771  2.97  1.62      3    2.85  1.48   1   6     5  0.37    -1.06 0.03
O1          21 2778  4.82  1.13      5    4.96  1.48   1   6     5 -0.90     0.43 0.02
O2          22 2800  2.71  1.57      2    2.56  1.48   1   6     5  0.59    -0.81 0.03
O3          23 2772  4.44  1.22      5    4.56  1.48   1   6     5 -0.77     0.30 0.02
O4          24 2786  4.89  1.22      5    5.10  1.48   1   6     5 -1.22     1.08 0.02
O5          25 2780  2.49  1.33      2    2.34  1.48   1   6     5  0.74    -0.24 0.03
gender      26 2800  1.67  0.47      2    1.71  0.00   1   2     1 -0.73    -1.47 0.01
education   27 2577  3.19  1.11      3    3.22  1.48   1   5     4 -0.05    -0.32 0.02
age         28 2800 28.78 11.13     26   27.43 10.38   3  86    83  1.02     0.56 0.21
```

How many factors?

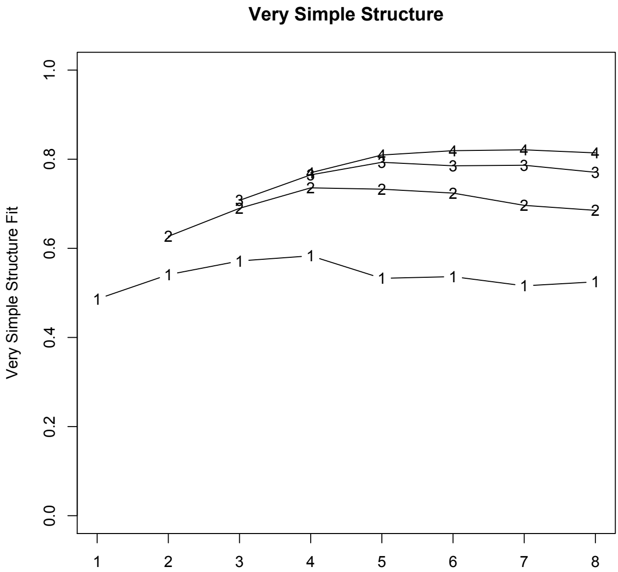
```
> fa.parallel(bfi[1:25]) #just the items
Parallel analysis suggests that the number of factors = 6 and th
6
```



How many factors part 2: VSS

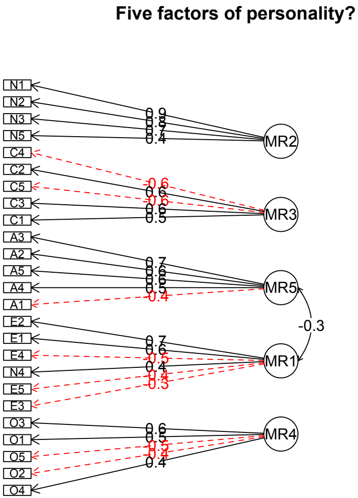
```
> VSS(bfi[1:25])

Very Simple Structure
Call: VSS(x = bfi[1:25])
VSS complexity 1 achieves a maximimum of 0.58  with  4  factors
VSS complexity 2 achieves a maximimum of 0.74  with  4  factors

The Velicer MAP criterion achieves a minimum of 0.01  with  5
factors

Velicer MAP
[1] 0.02 0.02 0.02 0.02 0.01 0.02 0.02 0.02

Very Simple Structure Complexity 1
[1] 0.49 0.54 0.57 0.58 0.53 0.54 0.52 0.52

Very Simple Structure Complexity 2
[1] 0.00 0.63 0.69 0.74 0.73 0.72 0.70 0.69
```

VSS plot


Very Simple Structure
Very Simple Structure Fit
0.0
0.2
0.4
0.6
0.8
1.0
1
2
3
4
5
6
7
8

Extract 5 factors from the BFI

```
> f5 <- fa(bfi[1:25],5)
fa.diagram(f5,main="Five factors of personality?")
```



Analyzing from an external file

- Data may reside on a local or a remote computer
- Option A: Using read.clipboard and its alternatives
    - Open the other other file using a text editor or spreadsheet program
    - Select all and copy (to the clipboard)
    - my.data <- read.clipboard() or my.data <- read.clipboard.csv() or read.clipboard.tab()
- Read the information directly
    - find the file and call it something fn <- file.choose()
    - Read in the data my.data <- read.table(fn, header=TRUE)
- Read from an SPSS file using the foreign package
    - library(foreign)
    - find the file and call it something fn <- file.choose()
    - my.data <- read.spss(fn,to.data.frame=TRUE)

A simplex

- In developmental, or any time process, nearby items are more correlated
  - An underlying growth process
  - localized errors
- grades in progressive quarters
- reaction times during a long session

Simulate a simplex

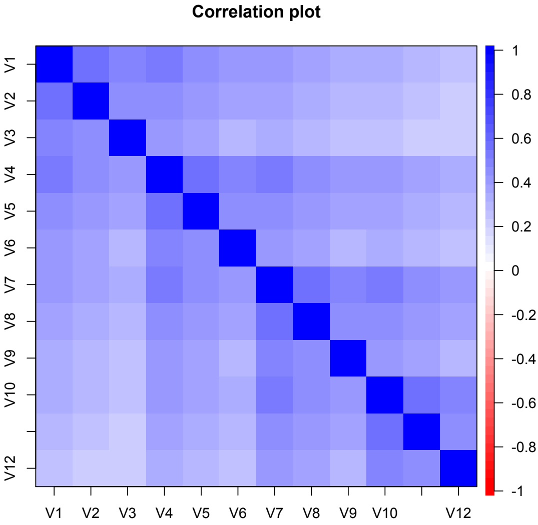
```
> set.seed(42)  # for reproducible results
> s9 <- sim.simplex(9,n=1000)
> str(s9) #show the structure
 List of 4
  $ model    : num [1:9, 1:9] 1 0.8 0.64 0.512 0.41 ...
   ..- attr(*, "dimnames")=List of 2
   .. ..$ : chr [1:9] "V1" "V2" "V3" "V4" ...
   .. ..$ : chr [1:9] "V1" "V2" "V3" "V4" ...
  $ r        : num [1:9, 1:9] 1 0.789 0.625 0.492 0.42 ...
   ..- attr(*, "dimnames")=List of 2
   .. ..$ : chr [1:9] "V1" "V2" "V3" "V4" ...
   .. ..$ : chr [1:9] "V1" "V2" "V3" "V4" ...
  $ observed: num [1:1000, 1:9] -0.659 -0.858 0.241 0.714 1.268 ...
   ..- attr(*, "dimnames")=List of 2
   .. ..$ : NULL
   .. ..$ : chr [1:9] "V1" "V2" "V3" "V4" ...
  $ Call     : language sim.simplex(nvar = 9, n = 1000)
  - attr(*, "class")= chr [1:2] "psych" "sim"
```

A simplex correlation matrix

```
> round(s9$model,2)
     V1   V2   V3   V4   V5   V6   V7   V8   V9
V1 1.00 0.80 0.64 0.51 0.41 0.33 0.26 0.21 0.17
V2 0.80 1.00 0.80 0.64 0.51 0.41 0.33 0.26 0.21
V3 0.64 0.80 1.00 0.80 0.64 0.51 0.41 0.33 0.26
V4 0.51 0.64 0.80 1.00 0.80 0.64 0.51 0.41 0.33
V5 0.41 0.51 0.64 0.80 1.00 0.80 0.64 0.51 0.41
V6 0.33 0.41 0.51 0.64 0.80 1.00 0.80 0.64 0.51
V7 0.26 0.33 0.41 0.51 0.64 0.80 1.00 0.80 0.64
V8 0.21 0.26 0.33 0.41 0.51 0.64 0.80 1.00 0.80
V9 0.17 0.21 0.26 0.33 0.41 0.51 0.64 0.80 1.00
```

How many factors?

fa.parallel(s9$observed)

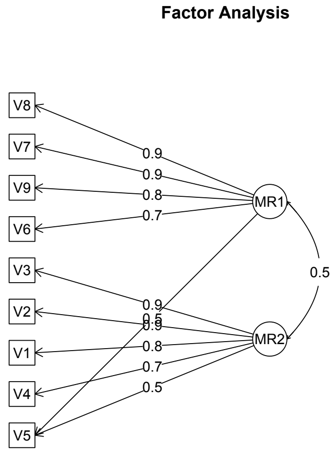


Factor a simplex

```
> f2 <- fa(s9$observed,2)
> f2
Factor Analysis using method =  minres
Call: fa(r = s9$observed, nfactors = 2)
Standardized loadings based upon correlation matrix
     MR1   MR2   h2   u2
V1 -0.07  0.76 0.53 0.47
V2 -0.06  0.89 0.75 0.25
V3 -0.01  0.92 0.83 0.17
V4  0.21  0.72 0.71 0.29
V5  0.48  0.49 0.69 0.31
V6  0.72  0.23 0.74 0.26
V7  0.90  0.04 0.85 0.15
V8  0.91 -0.10 0.76 0.24
V9  0.79 -0.09 0.57 0.43
                MR1  MR2
SS loadings     3.22 3.19
Proportion Var 0.36 0.35
Cumulative Var 0.36 0.71
 With factor correlations of
     MR1  MR2
MR1 1.00 0.47
MR2 0.47 1.00
```

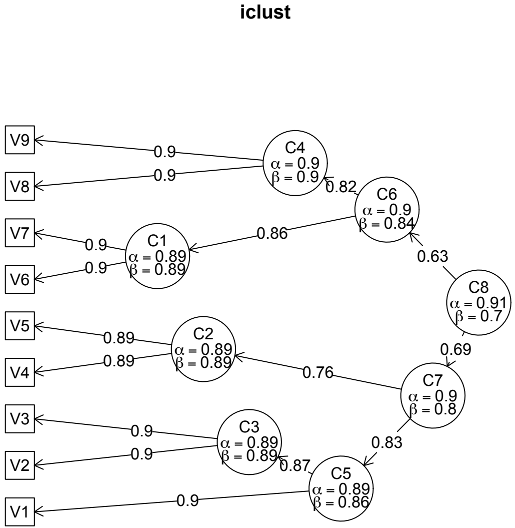
factor diagram

> fa.diagram(f2,simple=FALSE) #show large cross loadings



ICLUST of a simplex

> iclust(s9$observed) #cluster analyze the data



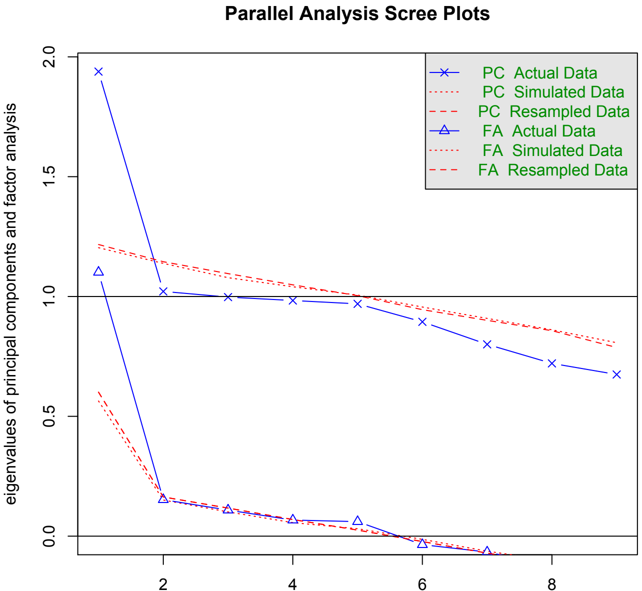
Item difficulty leads to a simplex structure

- Dichotomous items (e.g., ability items) differ in difficulty
  - Easy items have high endorsement rates
  - Hard items have low endorsement rates
- Φ coefficient is sensitive to differences in response
- Items with similar difficulties will correlate more highly

How many factors

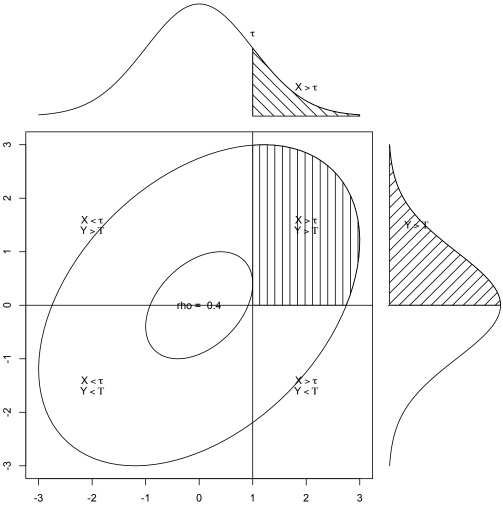
```
> set.seed(42)
> v9 <- sim.rasch(9)
> round(cor(v9$items),2)
     V1   V2   V3   V4   V5   V6   V7   V8   V9
V1 1.00 0.11 0.01 0.12 0.06 0.09 0.11 0.03 0.06
V2 0.11 1.00 0.16 0.23 0.09 0.08 0.09 0.10 0.09
V3 0.01 0.16 1.00 0.14 0.07 0.17 0.10 0.08 0.07
V4 0.12 0.23 0.14 1.00 0.23 0.23 0.13 0.12 0.12
V5 0.06 0.09 0.07 0.23 1.00 0.21 0.07 0.11 0.06
V6 0.09 0.08 0.17 0.23 0.21 1.00 0.21 0.05 0.16
V7 0.11 0.09 0.10 0.13 0.07 0.21 1.00 0.12 0.09
V8 0.03 0.10 0.08 0.12 0.11 0.05 0.12 1.00 0.02
V9 0.06 0.09 0.07 0.12 0.06 0.16 0.09 0.02 1.00
> fa.parallel(v9$items)
 Parallel analysis suggests that the number of facto
5
  and the number of components = 1
```

Parallel analysis of dichotomous items



Find the tetrachoric correlations

> draw.tetra(.4,1,0) #rho, cut 1, cut 2



The tetrachoric correlation matrix

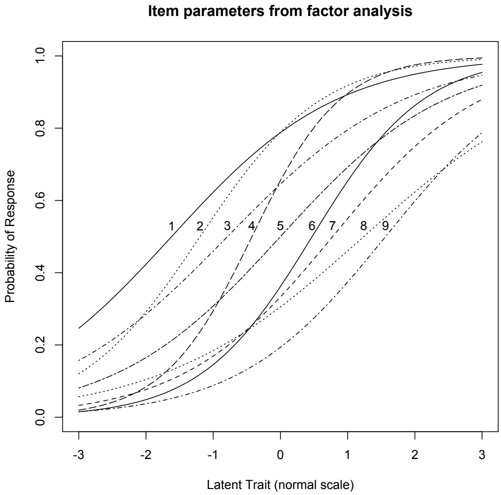
```
> rtet <- tetrachoric(v9$items)
Loading required package: mvtnorm
> rtet
Call: tetrachoric(x = v9$items)
tetrachoric correlation
     V1    V2   V3    V4   V5   V6   V7   V8    V9
V1 1.000 0.27 0.024 0.27 0.15 0.24 0.33 0.115 0.253
V2 0.268 1.00 0.299 0.41 0.18 0.15 0.21 0.281 0.322
V3 0.024 0.30 1.000 0.24 0.11 0.32 0.20 0.195 0.182
V4 0.273 0.41 0.242 1.00 0.36 0.38 0.23 0.253 0.288
V5 0.147 0.18 0.114 0.36 1.00 0.33 0.12 0.221 0.130
V6 0.239 0.15 0.316 0.38 0.33 1.00 0.35 0.111 0.335
V7 0.330 0.21 0.195 0.23 0.12 0.35 1.00 0.247 0.212
V8 0.115 0.28 0.195 0.25 0.22 0.11 0.25 1.000 0.048
V9 0.253 0.32 0.182 0.29 0.13 0.34 0.21 0.048 1.000

 with tau of
   V1    V2    V3    V4   V5   V6   V7   V8   V9
-1.46 -1.00 -0.72 -0.32 0.00 0.39 0.69 1.16 1.38
```

Factor analyze the items using tetrachorics

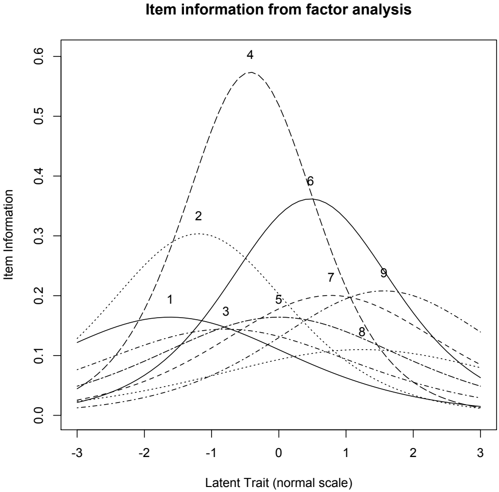
```
> f.irt <- irt.fa(v9$items)
> f.irt
Item Response Analysis using Factor Analysis =
Call: irt.fa(x = v9$items)
   Location  Discrimination   tau  Loading
V1    -1.62            0.48 -1.46     0.43
V2    -1.19            0.65 -1.00     0.54
V3    -0.79            0.45 -0.72     0.41
V4    -0.42            0.89 -0.32     0.66
V5     0.00            0.48  0.00     0.43
V6     0.47            0.71  0.39     0.58
V7     0.78            0.53  0.69     0.47
V8     1.24            0.39  1.16     0.36
V9     1.56            0.54  1.38     0.47
```

Show the items

> plot(f.irt,type="ICC")



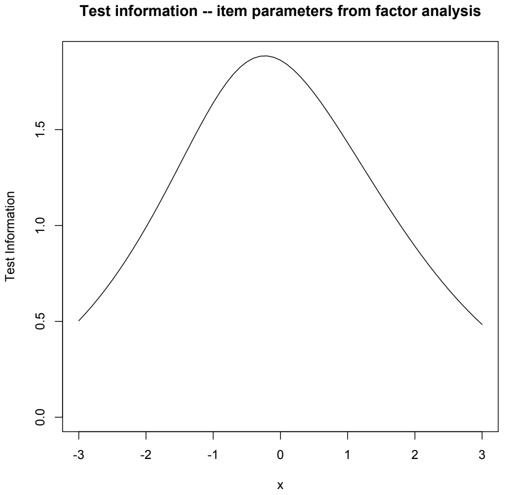
Show the item information functions



> plot(f.irt)

Show the Test information function

> plot(f.irt,type="test")



Polytomous items

- Most personality items have 3-6 alternatives
  - The fewer the alternatives, the more the correlation is restricted
  - For 6 choice items this is not too serious, but for 4, it probably is
- Find the polychoric correlation (What would be the Pearson if the data were bivariate normal?
- polychoric function in R

Find polychoric correlations for Big 5 items

```
data(bfi)
rbfi <-polychoric(bfi[1:25]) #this takes awhile
```

Hierarchical Clustering (e.g., `iclust`)

1. Find the proximity (e.g. correlation) matrix,
2. Identify the most similar pair of items
3. Combine this most similar pair of items to form a new variable (cluster),
4. Find the similarity of this cluster to all other items and clusters,
5. Repeat steps 2 and 3 until some criterion is reached (e.g., typically, if only one cluster remains or in `ICLUST` if there is a failure to increase reliability coefficients α or β).
6. Purify the solution by reassigning items to the most similar cluster center.

Factor score estimates are estimates, not precise

That factor scores are indeterminate has been taken by some (e.g., Schonemann, 1996) to represent psychopathology on the part of psychometricians, for the problem of indeterminacy has been known (and either ignored or suppressed) since Wilson (1928). To others, factor indeterminacy is a problem at the data level but not the structural level, and at the data level it is adequate to report the degree of indeterminacy. This degree of indeterminacy is indeed striking (Schonemann and Wang, 1972; Velicer and Jackson, 1990), and should be reported routinely. It is reported for each factor in the `fa` and `omega` functions.

Factor score estimates

1. The problem is, factors model items, loadings are β of factor onto item
2. We need to predict factor from the items.
3. Most naive model: unit weight the variables that have salient loadings
4. Regression weights based upon **R** and loadings.
   - $w = \mathbf{R}^{-1}\mathbf{F}$
5. Ten Berge weights
   - $L = F\phi^{1/2}$
   - $C = R^{-1/2}L(L'R^{-})L)^{-1/2}$
   - $W = R^{-1/2}C\phi^{1/2}$
6. Bartlett
   - $W = U^{-2}F(F'U^{-2}F)^{-1}$

Factor analysis and factor score estimates

```
> f3 <- fa(Thurstone,3)
> f3

loading required package: GPArotation

Factor Analysis using method =  minres
Call: fa(r = Thurstone, nfactors = 3)
...

The root mean square of the residuals (RMSR) is  0.01
The df corrected root mean square of the residuals is  0.01

Fit based upon off diagonal values = 1
Measures of factor score adequacy
                                                   MR1  MR2  MR3
Correlation of scores with factors                0.96 0.92 0.90
Multiple R square of scores with factors          0.93 0.85 0.81
Minimum correlation of possible factor scores     0.86 0.71 0.63
```

The weights are smaller than the loadings but highly similar

```
> round(f3$weights,2)

                   [,1]  [,2]  [,3]
Sentences          0.37 -0.01  0.05
Vocabulary         0.38  0.08 -0.02
Sent.Completion    0.22  0.03  0.01
First.Letters      0.01  0.47  0.02
4.Letter.Words     0.01  0.29  0.06
Suffixes           0.03  0.18 -0.02
Letter.Series      0.01  0.01  0.55
Pedigrees          0.05 -0.01  0.18
Letter.Group      -0.01  0.07  0.25

> factor.congruence(f3$weights,f3$loadings)
      MR1  MR2  MR3
[1,] 0.97 0.06 0.06
[2,] 0.11 0.97 0.09
[3,] 0.10 0.10 0.97
```

Factor extension

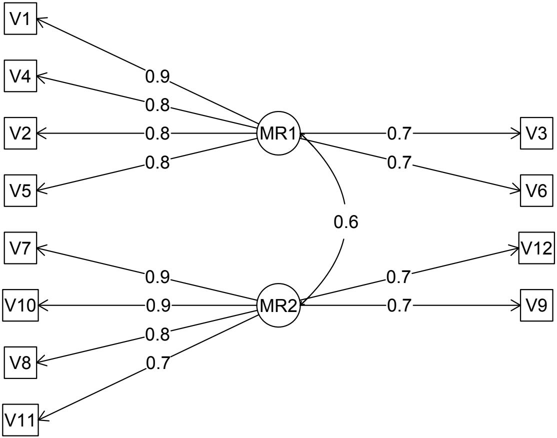
1. Originally developed for the problem of more variables being added after a tedious factor analysis was carried out.
2. Dwyer (1937) introduced a method for finding factor loadings for variables not included in the original analysis. This is basically finding the unattenuated correlation of the extension variables with the factor scores. An alternative, which does not correct for factor reliability was proposed by Gorsuch (1997). Both options are an application of exploratory factor analysis with extensions to new variables.

Factor Extension

Table: Create 12 variables with a clear two factor structure. Remove variables 3, 6, 9, and 12 from the matrix and factor the other variables. Then extend this solutio to the deleted variables. Compare this solution to the solution with all variables included (not shown). A graphic representation is in Figure 1.

```
set.seed(42)
fx <- matrix(c(.9,.8,.7,.85,.75,.65,rep(0,12),.9,.8,.7,.85,.75,.65),ncol=2)
 Phi <- matrix(c(1,.6,.6,1),2)
 sim.data <- sim.structure(fx,Phi,n=1000,raw=TRUE)
 R <- cor(sim.data$observed)
 Ro <- R[c(1,2,4,5,7,8,10,11),c(1,2,4,5,7,8,10,11)]
 Roe <- R[c(1,2,4,5,7,8,10,11),c(3,6,9,12)]
 fo <- fa(Ro,2)
 fe <- fa.extension(Roe,fo)
 fa.diagram(fo,fe=fe)

Call: fa.extension(Roe = Roe, fo = fo)
Standardized loadings based upon correlation matrix
      MR1   MR2   h2   u2
V3    0.69  0.01 0.49 0.51
V6    0.66 -0.02 0.42 0.58
V9    0.01  0.66 0.44 0.56
V12  -0.06  0.70 0.44 0.56

                MR1  MR2
SS loadings     0.89 0.89
```

Factor analysis and extension



Factor analysis/Components analysis/MDS/cluster analysis as models of data

1. Factor analysis $\mathbf{R} = \mathbf{FF'} + \mathbf{U}^2$ Models the common variance
2. Components $\mathbf{R} = \mathbf{CC'}$ Models all the variance
3. Cluster analysis – can be a blend of factor idea and component idea
4. MDS Similar results to factor, but with the general factor removed
5. All are models, they should be compared!
6. Confirmatory factor models to be discussed later

A useful tutorial: https://personality-project.org/r/psych/HowTo/factor.pdf

Steps toward data reduction/theory clarification

1. Verify the quality of the data
2. Choose a method for extraction – components vs. factors
3. Choose the number of factors to extract (statistically, pragmatically, rules of thumb)
4. Rotate/transform the results to a solution that is interpretable and useful
5. Estimate the factor scores for analysis with criteria or just find the factor correlations using structural equation techniques
6. Compare the model to alternative models
7. Don't panic!

Bartlett, M. S. (1951). The effect of standardization on a χ^2 approximation in factor analysis. *Biometrika*, 38(3/4):337–344.

Burt, C. (1948). The factorial study of temperamental traits. *British Journal of Statistical Psychology*, 1(3):178–203.

Carroll, J. (1953). An analytical solution for approximating simple structure in factor analysis. *Psychometrika*, 18(1):23–38.

Cattell, R. B. (1965). A biometrics invited paper. factor analysis: An introduction to essentials i. the purpose and underlying models. *Biometrics*, 21(1):190–215.

Cattell, R. B. (1966). The scree test for the number of factors. *Multivariate Behavioral Research*, 1(2):245–276.

Eckart, C. and Young, G. (1936). The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218.

Ferguson, G. (1954). The concept of parsimony in factor analysis. *Psychometrika*, 19(4):281–290.

Horn, J. L. (1965). A rationale and test for the number of factors in factor analysis. *Psychometrika*, 30(2):179–185.

Householder, A. S. and Young, G. (1938). Matrix approximation and latent roots. *The American Mathematical Monthly*, 45(3):165–171.

Jennrich, R. (2001). A simple general procedure for orthogonal rotation. *Psychometrika*, 66(2):289–306.

Jennrich, R. (2002). A simple general method for oblique rotation. *Psychometrika*, 67(1):7–19.

Jennrich, R. (2004). Derivative free gradient projection algorithms for rotation. *Psychometrika*, 69(3):475–480.

Jöreskog, K. G. (1978). Structural analysis of covariance and correlation matrices. *Psychometrika*, 43(4):443–477.

Kaiser, H. F. (1958). The varimax criterion for analytic rotation in factor analysis. *Psychometrika*, 23(3):187–200.

Lawley, D. N. and Maxwell, A. E. (1962). Factor analysis as a statistical method. *The Statistician*, 12(3):209–229.

Lawley, D. N. and Maxwell, A. E. (1963). *Factor analysis as a statistical method*. Butterworths, London.

Neuhaus, J. and Wrigley, C. (1954). The quartimax method: an analytical approach to orthogonal simple structure. *British Journal of Statistical Psychology*, 7:81–91.

Revelle, W. and Rocklin, T. (1979). Very Simple Structure - alternative procedure for estimating the optimal number of interpretable factors. *Multivariate Behavioral Research*, 14(4):403–414.

Roff, M. (1936). Some properties of the communality in multiple factor theory. *Psychometrika*, 1(2):1–6.

Schonemann, P. H. (1996). The psychopathology of factor indeterminacy. *Multivariate Behavioral Research*, 31(4):571–577.

Schonemann, P. H. and Wang, M.-M. (1972). Some new results on factor indeterminacy. *Psychometrika*, . 37(1, Pt. 1):61–91.

Spearman, C. (1904). "General Intelligence," objectively determined and measured. *American Journal of Psychology*, 15(2):201–292.

Thurstone, L. L. (1947). *Multiple-factor analysis: a development* *and expansion of The vectors of the mind*. The University of Chicago Press, Chicago, Ill.

Velicer, W. (1976). Determining the number of components from the matrix of partial correlations. *Psychometrika*, 41(3):321–327.

Velicer, W. F. and Jackson, D. N. (1990). Component analysis versus common factor analysis: Some issues in selecting an appropriate procedure. *Multivariate Behavioral Research*, 25(1):1–28.

Wilson, E. B. (1928). On hierarchical correlation systems. *Proceedings of the National Academy of Sciences*, 14(3):283–291.