An introduction to R: a short course
Part II
Multivariate analysis
Zelig is named after a Woody Allen movie about a man who had the strange ability to become the physical and psychological reflection of anyone he met and thus to fit perfectly in any situation.
Multivariate analysis

I. Dimension reduction through factor analysis, principal components analysis, cluster analysis, and multidimensional scaling

II. Multiple measures of reliability

III. Practical use of R for scoring inventories
Dimension Reduction

I. The problem: How best to summarize and think about many variables some of which are moderately correlated

II. The solutions: rank reduction through FA, PCA, CA, MDS

III. Examples will be tests and then items
The Thurstone data set

> data(bifactor)
> colnames(Thurstone) <- c
("Sentences","Vocab","S.comp","F.letter","4.letter","Suffix",
  "Series","Pedi","letters")

> round(Thurstone,2)

<table>
<thead>
<tr>
<th></th>
<th>Sentences</th>
<th>Vocab</th>
<th>S.comp</th>
<th>F.letter</th>
<th>4.letter</th>
<th>Suffix</th>
<th>Series</th>
<th>Pedi</th>
<th>letters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentences</td>
<td>1.00</td>
<td>0.83</td>
<td>0.78</td>
<td>0.44</td>
<td>0.43</td>
<td>0.45</td>
<td>0.45</td>
<td>0.54</td>
<td>0.38</td>
</tr>
<tr>
<td>Vocabulary</td>
<td>0.83</td>
<td>1.00</td>
<td>0.78</td>
<td>0.49</td>
<td>0.46</td>
<td>0.49</td>
<td>0.43</td>
<td>0.54</td>
<td>0.36</td>
</tr>
<tr>
<td>Sent.Completion</td>
<td>0.78</td>
<td>0.78</td>
<td>1.00</td>
<td>0.46</td>
<td>0.42</td>
<td>0.44</td>
<td>0.40</td>
<td>0.53</td>
<td>0.36</td>
</tr>
<tr>
<td>First.Letters</td>
<td>0.44</td>
<td>0.49</td>
<td>0.46</td>
<td>1.00</td>
<td>0.67</td>
<td>0.59</td>
<td>0.38</td>
<td>0.35</td>
<td>0.42</td>
</tr>
<tr>
<td>4.Letter.Words</td>
<td>0.43</td>
<td>0.46</td>
<td>0.42</td>
<td>0.67</td>
<td>1.00</td>
<td>0.54</td>
<td>0.40</td>
<td>0.37</td>
<td>0.45</td>
</tr>
<tr>
<td>Suffixes</td>
<td>0.45</td>
<td>0.49</td>
<td>0.44</td>
<td>0.59</td>
<td>0.54</td>
<td>1.00</td>
<td>0.29</td>
<td>0.32</td>
<td>0.32</td>
</tr>
<tr>
<td>Letter.Series</td>
<td>0.45</td>
<td>0.43</td>
<td>0.40</td>
<td>0.38</td>
<td>0.40</td>
<td>0.29</td>
<td>1.00</td>
<td>0.56</td>
<td>0.60</td>
</tr>
<tr>
<td>Pedigrees</td>
<td>0.54</td>
<td>0.54</td>
<td>0.53</td>
<td>0.35</td>
<td>0.37</td>
<td>0.32</td>
<td>0.56</td>
<td>1.00</td>
<td>0.45</td>
</tr>
<tr>
<td>Letter.Group</td>
<td>0.38</td>
<td>0.36</td>
<td>0.36</td>
<td>0.42</td>
<td>0.45</td>
<td>0.32</td>
<td>0.60</td>
<td>0.45</td>
<td>1.00</td>
</tr>
</tbody>
</table>
How many dimensions

I. Chi Square test
II. Scree plot
III. Parallel analysis of random data
IV. Minimum Average Partial correlation
V. Very Simple Structure
VI. Do not use eigen value > 1 rule!
> factanal(covmat=Thurstone, factors=3, n.obs = 213)

Call:
factanal(factors = 3, covmat = Thurstone, n.obs = 213)

Uniquenesses:

<table>
<thead>
<tr>
<th>Sentences</th>
<th>Vocab</th>
<th>S.comp</th>
<th>F.letter</th>
<th>4.letter</th>
<th>Suffix</th>
<th>Series</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pedi</td>
<td>0.175</td>
<td>0.165</td>
<td>0.268</td>
<td>0.268</td>
<td>0.372</td>
<td>0.504</td>
</tr>
<tr>
<td>letters</td>
<td>0.496</td>
<td>0.473</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Loadings:

<table>
<thead>
<tr>
<th></th>
<th>Factor1</th>
<th>Factor2</th>
<th>Factor3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentences</td>
<td>0.834</td>
<td>0.244</td>
<td>0.264</td>
</tr>
<tr>
<td>Vocabulary</td>
<td>0.827</td>
<td>0.318</td>
<td>0.223</td>
</tr>
<tr>
<td>Sent.Completion</td>
<td>0.775</td>
<td>0.284</td>
<td>0.227</td>
</tr>
<tr>
<td>First.Letters</td>
<td>0.228</td>
<td>0.792</td>
<td>0.230</td>
</tr>
<tr>
<td>4.Letter.Words</td>
<td>0.213</td>
<td>0.706</td>
<td>0.291</td>
</tr>
<tr>
<td>Suffixes</td>
<td>0.314</td>
<td>0.616</td>
<td>0.134</td>
</tr>
<tr>
<td>Letter.Series</td>
<td>0.232</td>
<td>0.179</td>
<td>0.795</td>
</tr>
<tr>
<td>Pedigrees</td>
<td>0.446</td>
<td>0.166</td>
<td>0.527</td>
</tr>
<tr>
<td>Letter.Group</td>
<td>0.154</td>
<td>0.311</td>
<td>0.638</td>
</tr>
</tbody>
</table>

Factor1 Factor2 Factor3

| SS loadings | 2.454 | 1.902 | 1.642 |
| Proportion Var | 0.273 | 0.211 | 0.182 |
| Cumulative Var | 0.273 | 0.484 | 0.666 |

Test of the hypothesis that 3 factors are sufficient.
The chi square statistic is 2.82 on 12 degrees of freedom.
The p-value is 0.997
Call:
factanal(factors = 2, covmat = Thurstone, n.obs = 213)

Uniquenesses:

<table>
<thead>
<tr>
<th></th>
<th>Sentences</th>
<th>Vocab</th>
<th>S.comp</th>
<th>F.letter</th>
<th>4.letter</th>
<th>Suffix</th>
<th>Series</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pedi letters</td>
<td>0.168</td>
<td>0.178</td>
<td>0.269</td>
<td>0.322</td>
<td>0.344</td>
<td>0.537</td>
<td>0.680</td>
</tr>
<tr>
<td></td>
<td>0.612</td>
<td>0.677</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Loadings:

<table>
<thead>
<tr>
<th></th>
<th>Factor1</th>
<th>Factor2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentences</td>
<td>0.866</td>
<td>0.287</td>
</tr>
<tr>
<td>Vocabulary</td>
<td>0.839</td>
<td>0.343</td>
</tr>
<tr>
<td>Sent.Completion</td>
<td>0.795</td>
<td>0.314</td>
</tr>
<tr>
<td>First.Letters</td>
<td>0.255</td>
<td>0.783</td>
</tr>
<tr>
<td>4.Letter.Words</td>
<td>0.235</td>
<td>0.775</td>
</tr>
<tr>
<td>Suffixes</td>
<td>0.317</td>
<td>0.602</td>
</tr>
<tr>
<td>Letter.Series</td>
<td>0.372</td>
<td>0.426</td>
</tr>
<tr>
<td>Pedigrees</td>
<td>0.524</td>
<td>0.336</td>
</tr>
<tr>
<td>Letter.Group</td>
<td>0.269</td>
<td>0.501</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Factor1</th>
<th>Factor2</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS loadings</td>
<td>2.793</td>
<td>2.420</td>
</tr>
<tr>
<td>Proportion Var</td>
<td>0.310</td>
<td>0.269</td>
</tr>
<tr>
<td>Cumulative Var</td>
<td>0.310</td>
<td>0.579</td>
</tr>
</tbody>
</table>

Test of the hypothesis that 2 factors are sufficient.
The chi square statistic is 82.84 on 19 degrees of freedom.
The p-value is 5.99e-10
Parallel analysis

Parallel Analysis Scree Plots

- PC Actual Data
- PC Simulated Data
- FA Actual Data
- FA Simulated Data

Factor Number

Eigenvalues of principal components and factor analysis
```
> vss <- VSS(Thurstone,n.obs=213,SMC=FALSE)
> vss

Very Simple Structure
Call: VSS(x = Thurstone, n.obs = 213, SMC = FALSE)
VSS complexity 1 achieves a maximum of 0.88 with 1 factors
VSS complexity 2 achieves a maximum of 0.92 with 2 factors

The Velicer MAP criterion achieves a minimum of 1 with 3 factors

Velicer MAP
[1] 0.07 0.07 0.07 0.11 0.20 0.31 0.59 1.00

Very Simple Structure Complexity 1
[1] 0.88 0.60 0.54 0.51 0.47 0.47 0.46 0.46

Very Simple Structure Complexity 2
[1] 0.00 0.92 0.86 0.79 0.73 0.72 0.63 0.63
```
Very Simple Structure

![Graph showing factor fit for Very Simple Structure](image)
Principal Axis FA

> pa3 <- factor.pa(Thurstone, nfacators=3, n.obs=213)
> pa3

<table>
<thead>
<tr>
<th></th>
<th>V</th>
<th>PA1</th>
<th>PA2</th>
<th>PA3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentences</td>
<td>1</td>
<td>0.83</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vocab</td>
<td>2</td>
<td>0.83</td>
<td>0.32</td>
<td></td>
</tr>
<tr>
<td>S.comp</td>
<td>3</td>
<td>0.78</td>
<td></td>
<td></td>
</tr>
<tr>
<td>F.letter</td>
<td>4</td>
<td></td>
<td>0.79</td>
<td></td>
</tr>
<tr>
<td>4.letter</td>
<td>5</td>
<td></td>
<td>0.71</td>
<td></td>
</tr>
<tr>
<td>Suffix</td>
<td>6</td>
<td>0.31</td>
<td>0.62</td>
<td></td>
</tr>
<tr>
<td>Series</td>
<td>7</td>
<td></td>
<td>0.79</td>
<td></td>
</tr>
<tr>
<td>Pedi</td>
<td>8</td>
<td>0.45</td>
<td>0.53</td>
<td></td>
</tr>
<tr>
<td>letters</td>
<td>9</td>
<td>0.31</td>
<td>0.64</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>PA1</th>
<th>PA2</th>
<th>PA3</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS loadings</td>
<td>2.46</td>
<td>1.91</td>
<td>1.64</td>
</tr>
<tr>
<td>Proportion Var</td>
<td>0.27</td>
<td>0.21</td>
<td>0.18</td>
</tr>
<tr>
<td>Cumulative Var</td>
<td>0.27</td>
<td>0.49</td>
<td>0.67</td>
</tr>
</tbody>
</table>

Test of the hypothesis that 3 factors are sufficient.

The degrees of freedom for the model is 12 and the fit was 0.01
The number of observations was 213 with Chi Square = 2.97 with prob < 1
Factor analysis options

I. factanal (MLE) is part of core R

II. fa is part of psych

   A. minres (ols) (default)

   B. weighted least squares

   C. generalized weighted least squares

   D. maximum likelihood

   E. principal axis
Principal Components

> pc3

<table>
<thead>
<tr>
<th></th>
<th>V</th>
<th>PC1</th>
<th>PC2</th>
<th>PC3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentences</td>
<td>1</td>
<td>0.863</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vocabulary</td>
<td>2</td>
<td>0.854</td>
<td>0.31</td>
<td></td>
</tr>
<tr>
<td>Sent.Completion</td>
<td>3</td>
<td>0.849</td>
<td></td>
<td></td>
</tr>
<tr>
<td>First.Letters</td>
<td>4</td>
<td></td>
<td>0.82</td>
<td></td>
</tr>
<tr>
<td>4.Letter.Words</td>
<td>5</td>
<td></td>
<td>0.79</td>
<td>0.301</td>
</tr>
<tr>
<td>Suffixes</td>
<td>6</td>
<td>0.314</td>
<td>0.77</td>
<td></td>
</tr>
<tr>
<td>Letter.Series</td>
<td>7</td>
<td></td>
<td>0.834</td>
<td></td>
</tr>
<tr>
<td>Pedigrees</td>
<td>8</td>
<td>0.534</td>
<td>0.613</td>
<td></td>
</tr>
<tr>
<td>Letter.Group</td>
<td>9</td>
<td>0.31</td>
<td>0.805</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>PC1</th>
<th>PC2</th>
<th>PC3</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS loadings</td>
<td>2.73</td>
<td>2.25</td>
<td>1.99</td>
</tr>
<tr>
<td>Proportion Var</td>
<td>0.30</td>
<td>0.25</td>
<td>0.22</td>
</tr>
<tr>
<td>Cumulative Var</td>
<td>0.30</td>
<td>0.55</td>
<td>0.78</td>
</tr>
</tbody>
</table>

Test of the hypothesis that 3 factors are sufficient.

The degrees of freedom for the model is 12 and the fit was 0.62.
The number of observations was 213 with Chi Square = 127.9 with prob < 1.6e-21.
Comparing solutions: factor congruence

```r
> factor.congruence(list(f3, pa3, pc3))
```

<table>
<thead>
<tr>
<th></th>
<th>Factor1</th>
<th>Factor2</th>
<th>Factor3</th>
<th>PA1</th>
<th>PA2</th>
<th>PA3</th>
<th>PC1</th>
<th>PC2</th>
<th>PC3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor1</td>
<td>1.00</td>
<td>0.64</td>
<td>0.62</td>
<td>1.00</td>
<td>0.64</td>
<td>0.62</td>
<td>1.00</td>
<td>0.59</td>
<td>0.55</td>
</tr>
<tr>
<td>Factor2</td>
<td>0.64</td>
<td>1.00</td>
<td>0.62</td>
<td>0.63</td>
<td>1.00</td>
<td>0.62</td>
<td>0.61</td>
<td>0.99</td>
<td>0.57</td>
</tr>
<tr>
<td>Factor3</td>
<td>0.62</td>
<td>0.62</td>
<td>1.00</td>
<td>0.62</td>
<td>0.62</td>
<td>1.00</td>
<td>0.61</td>
<td>0.56</td>
<td>0.99</td>
</tr>
<tr>
<td>PA1</td>
<td>1.00</td>
<td>0.63</td>
<td>0.62</td>
<td>1.00</td>
<td>0.64</td>
<td>0.62</td>
<td>1.00</td>
<td>0.59</td>
<td>0.55</td>
</tr>
<tr>
<td>PA2</td>
<td>0.64</td>
<td>1.00</td>
<td>0.62</td>
<td>0.64</td>
<td>1.00</td>
<td>0.62</td>
<td>0.61</td>
<td>0.99</td>
<td>0.57</td>
</tr>
<tr>
<td>PA3</td>
<td>0.62</td>
<td>0.62</td>
<td>1.00</td>
<td>0.62</td>
<td>0.62</td>
<td>1.00</td>
<td>0.61</td>
<td>0.56</td>
<td>0.99</td>
</tr>
<tr>
<td>PC1</td>
<td>1.00</td>
<td>0.61</td>
<td>0.61</td>
<td>1.00</td>
<td>0.61</td>
<td>0.61</td>
<td>1.00</td>
<td>0.56</td>
<td>0.54</td>
</tr>
<tr>
<td>PC2</td>
<td>0.59</td>
<td>0.99</td>
<td>0.56</td>
<td>0.59</td>
<td>0.99</td>
<td>0.56</td>
<td>0.56</td>
<td>1.00</td>
<td>0.50</td>
</tr>
<tr>
<td>PC3</td>
<td>0.55</td>
<td>0.57</td>
<td>0.99</td>
<td>0.55</td>
<td>0.57</td>
<td>0.99</td>
<td>0.54</td>
<td>0.50</td>
<td>1.00</td>
</tr>
</tbody>
</table>
A misleading graph

Factor Analysis

V1
V2
V3
V4
V5
V6
V7
V8
V9

F1
F2
F3
Plot the loadings: shows some cross loadings
Rotations and transformations

I. Orthogonal rotations
   A. Varimax, Quartimax

II. Oblique transformations
   A. Promax, Oblimin, Quartimin, biquartimin, targeted, cluster, ...
> fa3o <- fa(Thurstone,3,rotate="oblimin")
Loading required package: GPArotation
> fa3o

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

item   MR1   MR2   MR3   h2   u2
Sentences          1  0.90             0.82 0.18
Vocabulary         2  0.89             0.84 0.16
Sent.Completion    3  0.84             0.74 0.26
First.Letters      4        0.85       0.73 0.27
4.Letter.Words     5        0.75       0.63 0.37
Suffixes           6        0.63       0.50 0.50
Letter.Series      7              0.84 0.72 0.28
Pedigrees          8  0.38        0.47 0.50 0.50
Letter.Group       9              0.63 0.52 0.48

  MR1  MR2  MR3
SS loadings    2.64 1.87 1.49
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 0.53
MR2  0.59 1.00 0.52
MR3  0.53 0.52 1.00

Test of the hypothesis that 3 factors are sufficient.
The degrees of freedom for the model is 12 and the objective function is 0.01
Show all values!
> pa3p <- Promax(pa3)
> pa3p

V PA1 PA2 PA3
Sentences 1 0.911
Vocab 2 0.904
S.comp 3 0.848
F.letter 4 0.869
4.letter 5 0.759
Suffix 6 0.650
Series 7 0.8865
Pedi 8 0.350 0.4969
letters 9 0.6800

<table>
<thead>
<tr>
<th>PA1</th>
<th>PA2</th>
<th>PA3</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS loadings</td>
<td>2.54</td>
<td>1.80</td>
</tr>
<tr>
<td>Proportion Var</td>
<td>0.28</td>
<td>0.20</td>
</tr>
<tr>
<td>Cumulative Var</td>
<td>0.28</td>
<td>0.48</td>
</tr>
</tbody>
</table>

With factor correlations of

<table>
<thead>
<tr>
<th>PA1</th>
<th>PA2</th>
<th>PA3</th>
</tr>
</thead>
<tbody>
<tr>
<td>PA1</td>
<td>1.00</td>
<td>0.61</td>
</tr>
<tr>
<td>PA2</td>
<td>0.61</td>
<td>1.00</td>
</tr>
<tr>
<td>PA3</td>
<td>0.61</td>
<td>0.58</td>
</tr>
</tbody>
</table>
A more accurate graphic

3 oblique factors from Thurstone

- Sentences
- Vocabulary
- Sent. Completion
- First. Letters
- 4. Letter. Words
- Suffixes
- Letter. Series
- Letter. Group
- Pedigrees

fa.diagran(fa3o)
Hierarchical solutions

Omega with Schmid Leiman Transformation

Hierarchical (multilevel) Structure

> omsl <- omega(Thurstone, title="Bifactor")
> omsl <- omega(Thurstone, sl=FALSE, title="Hierarchical")
Hierarchical Clustering

I. Find the similarity matrix (correlations)

II. Find the most similar pair of items/tests

III. Combine them and repeat II and III until some criterion (alpha, beta) fails to increase
Hierarchical cluster analysis

ICLUST of Thurstone's 9 variables

- C1: Vocabulary, α = 0.91, β = 0.91
- C2: First.Letters, α = 0.81, β = 0.81
- C3: Letter.Group, α = 0.75, β = 0.75
- C4: Sent.Completion, α = 0.92, β = 0.9
- C5: Sentences, α = 0.82, β = 0.77
- C6: 4.Letter.Words, α = 0.82, β = 0.77
- C7: Pedigrees, α = 0.87, β = 0.73
- C8: Suffixes, α = 0.89, β = 0.77

Correlation coefficients:
- 0.77
- 0.77
- 0.8
- 0.93
- 0.96
- 0.84
- 0.89
- 0.77
Multidimensional scaling

I. Convert correlations to distances

II. Multidimensional scaling will remove a general factor since it considers relative ranking of distances
MDS

> Thurs.dist <- sqrt(2*(1-Thurstone))
> mdsT <- cmdscale(Thurs.dist,2)
> round(mdsT,2)

[,1]   [,2]
Sentences       -0.46  0.08
Vocabulary      -0.42  0.16
Sent.Completion -0.44  0.12
First.Letters    0.37  0.32
4.Letter.Words   0.40  0.24
Suffixes         0.24  0.44
Letter.Series    0.14 -0.54
Pedigrees       -0.18 -0.37
Letter.Group     0.34 -0.46
position <- rep(4,9)
plot(mdsT,xlim=c(-.6,.6),ylim=c(-.6,.6),ylab="Axis 2",xlab="Axis 1")
text(mdsT,rownames(mdsT),pos=position)
A more complex example

I. Holzinger-Harman 24 mental ability tests

II. Compare FA, CA, MDS
Parallel analysis

Parallel Analysis Scree Plots

> hh <- Harman74.cor$cov
> fa.parallel(hh,n.obs=228)
VSS says 1 big factor

> vss <- VSS(hh, n.obs=228)
> vss

Very Simple Structure
Call: VSS(x = hh, n.obs = 228)
VSS complexity 1 achieves a maximum of 0.8 with 1 factors
VSS complexity 2 achieves a maximum of 0.85 with 2 factors

The Velicer MAP criterion achieves a minimum of 0.03 with 4 factors

Velicer MAP
[1] 0.02 0.02 0.02 0.02 0.02 0.02 0.02 0.03 0.03

Very Simple Structure Complexity 1
[1] 0.80 0.55 0.46 0.42 0.40 0.40 0.40 0.40 0.41

Very Simple Structure Complexity 2
[1] 0.00 0.85 0.79 0.74 0.71 0.70 0.70 0.69
Chi square says $> 8$

```r
> vss$vss.stats[,1:3]
     dof   chisq         prob
1  252 1012.9000 9.259262e-92
2  229  693.9911 2.649458e-48
3  207  485.1213 2.232607e-24
4  186  370.8825 2.174718e-14
5  166  307.7273 1.479600e-10
6  147  266.8324 5.628353e-09
7  129  226.6602 2.396296e-07
8  112  182.6407 2.832707e-05
> 
```
```r
> fa4o <- factor.pa(hh,4,n.obs=228,rotate='oblimin')
> fa4o

<table>
<thead>
<tr>
<th>V</th>
<th>PA1</th>
<th>PA3</th>
<th>PA2</th>
<th>PA4</th>
</tr>
</thead>
<tbody>
<tr>
<td>VisualPerception</td>
<td>1</td>
<td>0.69</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cubes</td>
<td>2</td>
<td>0.46</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PaperFormBoard</td>
<td>3</td>
<td>0.54</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Flags</td>
<td>4</td>
<td>0.52</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GeneralInformation</td>
<td>5</td>
<td>0.76</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ParagraphComprehension</td>
<td>6</td>
<td>0.80</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SentenceCompletion</td>
<td>7</td>
<td>0.87</td>
<td></td>
<td></td>
</tr>
<tr>
<td>WordClassification</td>
<td>8</td>
<td>0.56</td>
<td></td>
<td></td>
</tr>
<tr>
<td>WordMeaning</td>
<td>9</td>
<td>0.86</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Addition</td>
<td>10</td>
<td>0.86</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Code</td>
<td>11</td>
<td>0.49</td>
<td>0.30</td>
<td></td>
</tr>
<tr>
<td>CountingDots</td>
<td>12</td>
<td>0.70</td>
<td></td>
<td></td>
</tr>
<tr>
<td>StraightCurvedCapitals</td>
<td>13</td>
<td>0.42</td>
<td>0.47</td>
<td></td>
</tr>
<tr>
<td>WordRecognition</td>
<td>14</td>
<td></td>
<td>0.58</td>
<td></td>
</tr>
<tr>
<td>NumberRecognition</td>
<td>15</td>
<td></td>
<td>0.55</td>
<td></td>
</tr>
<tr>
<td>FigureRecognition</td>
<td>16</td>
<td>0.33</td>
<td>0.52</td>
<td></td>
</tr>
<tr>
<td>ObjectNumber</td>
<td>17</td>
<td></td>
<td>0.59</td>
<td></td>
</tr>
<tr>
<td>NumberFigure</td>
<td>18</td>
<td></td>
<td>0.43</td>
<td></td>
</tr>
<tr>
<td>FigureWord</td>
<td>19</td>
<td></td>
<td>0.32</td>
<td></td>
</tr>
<tr>
<td>Deduction</td>
<td>20</td>
<td>0.33</td>
<td>0.31</td>
<td></td>
</tr>
<tr>
<td>NumericalPuzzles</td>
<td>21</td>
<td>0.37</td>
<td>0.33</td>
<td></td>
</tr>
<tr>
<td>ProblemReasoning</td>
<td>22</td>
<td>0.31</td>
<td>0.30</td>
<td></td>
</tr>
<tr>
<td>SeriesCompletion</td>
<td>23</td>
<td>0.30</td>
<td>0.44</td>
<td></td>
</tr>
<tr>
<td>ArithmeticProblems</td>
<td>24</td>
<td>0.41</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

4 oblique factors

With factor correlations

```r
<table>
<thead>
<tr>
<th></th>
<th>PA1</th>
<th>PA3</th>
<th>PA2</th>
<th>PA4</th>
</tr>
</thead>
<tbody>
<tr>
<td>PA1</td>
<td>1.00</td>
<td>0.41</td>
<td>0.30</td>
<td>0.40</td>
</tr>
<tr>
<td>PA3</td>
<td>0.41</td>
<td>1.00</td>
<td>0.27</td>
<td>0.38</td>
</tr>
<tr>
<td>PA2</td>
<td>0.30</td>
<td>0.27</td>
<td>1.00</td>
<td>0.32</td>
</tr>
<tr>
<td>PA4</td>
<td>0.41</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
4 oblique factors of the Holzinger-Harman problem
om4 <- omega(hh, 4, title="Holzinger Harman problem -- Bifactor")
Holzinger Harman problem -- Hierarchical
ic4 <- ICLUST(hh,title="ICLUST of Holzinger Harman 24 mental measurements")
Hierarchical cluster analysis

ICLUST of Holzinger-Harman problem
MDS of HH problem

Multidimensional Scaling of 24 ability tests

- SentenceCompletion
- GeneralInformation
- WordMeaning
- ParagraphComprehension
- WordClassification
- ProblemReasoning
- Deduction
- Flags
- VisualPerception
- Cubes
- PaperFormBoard
- SeriesCompletion
- NumericalPuzzles
- FigureWord
- ObjectNumber
- FigureRecognition
- WordRecognition
- WordMeaning
- CountingDots
- StraightCurvedCapitals
- WordComprehension
- Code
- Addition
- ProblemReasoning
- SeriesCompletion
- Deduction
- Flags
- VisualPerception
- Cubes
- PaperFormBoard
- NumericalPuzzles
- FigureWord
- ObjectNumber
- FigureRecognition
- WordRecognition
- WordMeaning
- CountingDots
code for MDS plot

```r
> dis24 <- sqrt(2*(1-Harman74.cor$cov))
> mds24 <- cmdscale(dis24,2)
> plot.char <- c( 19, 19, 19, 19, 21, 21, 21, 21, 21, 20, 20, 20,
+ 20, 23, 23, 23, 23, 23, 23, 19, 22, 19, 19, 22 )
> plot(mds24,xlim=c(-.6,.6),ylim=c(-.6,.6),xlab="Dimension 1",ylab="Dimension 2",asp=1,pch=plot.char)
> position <- c(2,2,3,4, 4,4,3,4, 3,2,3,2, 3,3,1,4, 4,1,3,1, 1,2,3,4)
> text(mds24,rownames(mds24),cex=.6,pos=position)
> abline(v=0,h=0)
> title("Multidimensional Scaling of 24 ability tests")
> #draw circles at .25 and .50 units away from the center
> segments = 51
> angles <- (0:segments) * 2 * pi/segments
> unit.circle <- cbind(cos(angles), sin(angles))
> lines(unit.circle*.25)
> lines(unit.circle*.5)
```
Reliability of scales and raters

I. First consider the reliability of raters using the IntraClass Correlation (see Shrout and Fleiss for the definitive discussion)

II. Types of reliability of ratings (1 or n per target)

   A. each target rated by a different judge, judges are random

   B. random sample of k judges rate targets

   C. Fixed set of k judges give ratings
### 4 judges rate 6 subjects

<table>
<thead>
<tr>
<th>Subject</th>
<th>J1</th>
<th>J2</th>
<th>J3</th>
<th>J4</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>9</td>
<td>2</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>S2</td>
<td>6</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>S3</td>
<td>8</td>
<td>4</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>S4</td>
<td>7</td>
<td>1</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>S5</td>
<td>10</td>
<td>5</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>S6</td>
<td>6</td>
<td>2</td>
<td>4</td>
<td>7</td>
</tr>
</tbody>
</table>
Simple correlations
(these will remove means for raters)

> round(cor(sf),2)

<table>
<thead>
<tr>
<th></th>
<th>J1</th>
<th>J2</th>
<th>J3</th>
<th>J4</th>
</tr>
</thead>
<tbody>
<tr>
<td>J1</td>
<td>1.00</td>
<td>0.75</td>
<td>0.73</td>
<td>0.75</td>
</tr>
<tr>
<td>J2</td>
<td>0.75</td>
<td>1.00</td>
<td>0.89</td>
<td>0.73</td>
</tr>
<tr>
<td>J3</td>
<td>0.73</td>
<td>0.89</td>
<td>1.00</td>
<td>0.72</td>
</tr>
<tr>
<td>J4</td>
<td>0.75</td>
<td>0.73</td>
<td>0.72</td>
<td>1.00</td>
</tr>
</tbody>
</table>
ICC

> ICC(sf)

<table>
<thead>
<tr>
<th>type</th>
<th>ICC</th>
<th>F</th>
<th>df1</th>
<th>df2</th>
<th>p</th>
<th>lower bound</th>
<th>upper bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single_raters_absolute</td>
<td>ICC1</td>
<td>0.17</td>
<td>1.79</td>
<td>5</td>
<td>18</td>
<td>0.16</td>
<td>-0.13</td>
</tr>
<tr>
<td>Single_random_raters</td>
<td>ICC2</td>
<td>0.29</td>
<td>11.03</td>
<td>5</td>
<td>15</td>
<td>0.00</td>
<td>0.02</td>
</tr>
<tr>
<td>Single_fixed_raters</td>
<td>ICC3</td>
<td>0.71</td>
<td>11.03</td>
<td>5</td>
<td>15</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>Average_raters_absolute</td>
<td>ICC1k</td>
<td>0.44</td>
<td>1.79</td>
<td>5</td>
<td>18</td>
<td>0.16</td>
<td>-0.88</td>
</tr>
<tr>
<td>Average_random_raters</td>
<td>ICC2k</td>
<td>0.62</td>
<td>11.03</td>
<td>5</td>
<td>15</td>
<td>0.00</td>
<td>0.07</td>
</tr>
<tr>
<td>Average_fixed_raters</td>
<td>ICC3k</td>
<td>0.91</td>
<td>11.03</td>
<td>5</td>
<td>15</td>
<td>0.00</td>
<td>0.68</td>
</tr>
</tbody>
</table>

> alpha(sf)

Alpha of raters

Reliability analysis

Call: alpha(x = sf)

raw_alpha std.alpha G6(smc) average_r mean sd
0.91    0.93    0.92     0.76   21 6.7

Reliability if an item is dropped:

<table>
<thead>
<tr>
<th>raw_alpha</th>
<th>std.alpha</th>
<th>G6(smc)</th>
<th>average_r</th>
</tr>
</thead>
<tbody>
<tr>
<td>J1</td>
<td>0.88</td>
<td>0.91</td>
<td>0.89</td>
</tr>
<tr>
<td>J2</td>
<td>0.87</td>
<td>0.89</td>
<td>0.85</td>
</tr>
<tr>
<td>J3</td>
<td>0.87</td>
<td>0.90</td>
<td>0.85</td>
</tr>
<tr>
<td>J4</td>
<td>0.92</td>
<td>0.92</td>
<td>0.90</td>
</tr>
</tbody>
</table>

Item statistics

<table>
<thead>
<tr>
<th>n</th>
<th>r</th>
<th>r.cor</th>
<th>mean</th>
<th>sd</th>
</tr>
</thead>
<tbody>
<tr>
<td>J1</td>
<td>6</td>
<td>0.89</td>
<td>0.83</td>
<td>7.7</td>
</tr>
<tr>
<td>J2</td>
<td>6</td>
<td>0.93</td>
<td>0.92</td>
<td>2.5</td>
</tr>
<tr>
<td>J3</td>
<td>6</td>
<td>0.92</td>
<td>0.91</td>
<td>4.3</td>
</tr>
<tr>
<td>J4</td>
<td>6</td>
<td>0.88</td>
<td>0.82</td>
<td>6.7</td>
</tr>
</tbody>
</table>
Reliability of a single scale

```r
> round(cor(bfi[,1:10], use="pairwise"), 2)

<table>
<thead>
<tr>
<th></th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>1.00</td>
<td>-0.30</td>
<td>-0.23</td>
<td>-0.12</td>
<td>-0.19</td>
<td>-0.03</td>
<td>-0.06</td>
<td>0.01</td>
<td>0.19</td>
<td>0.08</td>
</tr>
<tr>
<td>A2</td>
<td>-0.30</td>
<td>1.00</td>
<td>0.39</td>
<td>0.24</td>
<td>0.41</td>
<td>0.06</td>
<td>0.06</td>
<td>0.22</td>
<td>-0.15</td>
<td>-0.12</td>
</tr>
<tr>
<td>A3</td>
<td>-0.23</td>
<td>0.39</td>
<td>1.00</td>
<td>0.27</td>
<td>0.45</td>
<td>0.07</td>
<td>0.12</td>
<td>0.16</td>
<td>-0.14</td>
<td>-0.12</td>
</tr>
<tr>
<td>A4</td>
<td>-0.12</td>
<td>0.24</td>
<td>0.27</td>
<td>1.00</td>
<td>0.22</td>
<td>0.07</td>
<td>0.17</td>
<td>0.08</td>
<td>-0.15</td>
<td>-0.17</td>
</tr>
<tr>
<td>A5</td>
<td>-0.19</td>
<td>0.41</td>
<td>0.45</td>
<td>0.22</td>
<td>1.00</td>
<td>0.10</td>
<td>0.06</td>
<td>0.20</td>
<td>-0.14</td>
<td>-0.10</td>
</tr>
<tr>
<td>C1</td>
<td>-0.03</td>
<td>0.06</td>
<td>0.07</td>
<td>0.07</td>
<td>0.10</td>
<td>1.00</td>
<td>0.44</td>
<td>0.41</td>
<td>-0.39</td>
<td>-0.23</td>
</tr>
<tr>
<td>C2</td>
<td>-0.06</td>
<td>0.06</td>
<td>0.12</td>
<td>0.17</td>
<td>0.06</td>
<td>0.44</td>
<td>1.00</td>
<td>0.35</td>
<td>-0.36</td>
<td>-0.24</td>
</tr>
<tr>
<td>C3</td>
<td>0.01</td>
<td>0.22</td>
<td>0.16</td>
<td>0.08</td>
<td>0.20</td>
<td>0.41</td>
<td>0.35</td>
<td>1.00</td>
<td>-0.37</td>
<td>-0.36</td>
</tr>
<tr>
<td>C4</td>
<td>0.19</td>
<td>-0.15</td>
<td>-0.14</td>
<td>-0.15</td>
<td>-0.14</td>
<td>-0.39</td>
<td>-0.36</td>
<td>-0.37</td>
<td>1.00</td>
<td>0.53</td>
</tr>
<tr>
<td>C5</td>
<td>0.08</td>
<td>-0.12</td>
<td>-0.12</td>
<td>-0.17</td>
<td>-0.10</td>
<td>-0.23</td>
<td>-0.24</td>
<td>-0.36</td>
<td>0.53</td>
<td>1.00</td>
</tr>
</tbody>
</table>
```
> alpha(bfi[1:10])

Reliability analysis
Call: alpha(x = bfi[1:10])

<table>
<thead>
<tr>
<th>raw_alpha</th>
<th>std.alpha</th>
<th>G6(smc)</th>
<th>average_r</th>
<th>mean</th>
<th>sd</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.19</td>
<td>0.25</td>
<td>0.44</td>
<td>0.032</td>
<td>40</td>
<td>4.7</td>
</tr>
</tbody>
</table>

Reliability if an item is dropped:

<table>
<thead>
<tr>
<th>raw_alpha</th>
<th>std.alpha</th>
<th>G6(smc)</th>
<th>average_r</th>
<th>Item statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>0.290</td>
<td>0.354</td>
<td>0.51</td>
<td>A1 1000 0.099 -0.21 2.3 1.3</td>
</tr>
<tr>
<td>A2</td>
<td>0.082</td>
<td>0.133</td>
<td>0.35</td>
<td>A2 994 0.508 0.47 4.8 1.1</td>
</tr>
<tr>
<td>A3</td>
<td>0.045</td>
<td>0.101</td>
<td>0.33</td>
<td>A3 989 0.552 0.54 4.6 1.2</td>
</tr>
<tr>
<td>A4</td>
<td>0.108</td>
<td>0.173</td>
<td>0.40</td>
<td>A4 993 0.448 0.31 4.8 1.4</td>
</tr>
<tr>
<td>A5</td>
<td>0.042</td>
<td>0.093</td>
<td>0.32</td>
<td>A5 988 0.563 0.55 4.6 1.2</td>
</tr>
<tr>
<td>C1</td>
<td>0.133</td>
<td>0.190</td>
<td>0.38</td>
<td>C1 997 0.421 0.34 4.4 1.2</td>
</tr>
<tr>
<td>C2</td>
<td>0.123</td>
<td>0.182</td>
<td>0.38</td>
<td>C2 997 0.434 0.35 4.2 1.2</td>
</tr>
<tr>
<td>C3</td>
<td>0.105</td>
<td>0.154</td>
<td>0.36</td>
<td>C3 995 0.478 0.44 4.3 1.3</td>
</tr>
<tr>
<td>C4</td>
<td>0.336</td>
<td>0.392</td>
<td>0.50</td>
<td>C4 986 0.005 -0.24 2.6 1.4</td>
</tr>
<tr>
<td>C5</td>
<td>0.329</td>
<td>0.364</td>
<td>0.49</td>
<td>C5 997 0.077 -0.17 3.5 1.5</td>
</tr>
</tbody>
</table>
somewhat better reliability

```r
> keys <- make.keys(10,list(all=c(-1,2:8,-9,-10)))
> alpha(bfi[1:10],keys)

Reliability analysis
Call: alpha(x = bfi[1:10], keys = keys)

    raw_alpha std.alpha  G6(smc) average_r    mean    sd
 0.72      0.72    0.75      0.21   40 4.7

Reliability if an item is dropped:

    raw_alpha std.alpha  G6(smc) average_r
A1  0.72      0.72    0.74      0.23
A2  0.70      0.70    0.72      0.20
A3  0.70      0.70    0.72      0.20
A4  0.71      0.71    0.74      0.22
A5  0.70      0.70    0.72      0.21
C1  0.70      0.70    0.72      0.21
C2  0.70      0.70    0.72      0.21
C3  0.69      0.69    0.71      0.20
C4  0.67      0.68    0.70      0.19
C5  0.70      0.70    0.72      0.20

Item statistics

    n   r  r.cor mean  sd
A1 1000 0.41  0.29  2.3 1.3
A2  994 0.55  0.49  4.8 1.1
A3  989 0.55  0.49  4.6 1.2
A4  993 0.46  0.35  4.8 1.4
A5  988 0.53  0.46  4.6 1.2
C1  997 0.52  0.46  4.4 1.2
C2  997 0.54  0.47  4.2 1.3
C3  995 0.59  0.61  2.6 1.4
C4  986 0.64  0.61  2.6 1.4
C5  997 0.55  0.49  3.5 1.5
```
Examine the items

The items

A1  Am indifferent to the feelings of others.
A2  Inquire about others’ well-being.
A3  Know how to comfort others.
A4  Love children.
A5  Make people feel at ease.
C1  Am exacting in my work.
C2  Continue until everything is perfect.
C3  Do things according to a plan.
C4  Do things in a half-way manner.
C5  Waste my time.

VSS suggests 2 factors!
Omega reliability

> om2 <- omega(bfi[1:10],2)
Warning messages:
1: In schmid(m, nfactors, pc, digits, rotate = rotate, n.obs = n.obs,  :
   Three factors are required for identification -- general factor loadings set to be equal. Proceed with caution.
2: In schmid(m, nfactors, pc, digits = digits, n.obs = n.obs, ...) :
   Three factors are required for identification -- general factor loadings set to be equal. Proceed with caution.
> om2
Omega
Call: omega(m = bfi[1:10], nfactors = 2)
Alpha: 0.72
G.6: 0.75
Omega Hierarchical: 0.36
Omega Total 0.77

Schmid Leiman Factor loadings greater than 0.2

<table>
<thead>
<tr>
<th></th>
<th>g</th>
<th>F1*</th>
<th>F2*</th>
<th>h2</th>
<th>u2</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1-</td>
<td>0.21</td>
<td>0.30</td>
<td>0.87</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A2</td>
<td>0.36</td>
<td>0.53</td>
<td>0.41</td>
<td>0.59</td>
<td></td>
</tr>
<tr>
<td>A3</td>
<td>0.36</td>
<td>0.55</td>
<td>0.43</td>
<td>0.57</td>
<td></td>
</tr>
<tr>
<td>A4</td>
<td>0.24</td>
<td>0.28</td>
<td>0.86</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A5</td>
<td>0.36</td>
<td>0.54</td>
<td>0.42</td>
<td>0.58</td>
<td></td>
</tr>
<tr>
<td>C1</td>
<td>0.30</td>
<td>0.51</td>
<td>0.36</td>
<td>0.64</td>
<td></td>
</tr>
<tr>
<td>C2</td>
<td>0.30</td>
<td>0.48</td>
<td>0.32</td>
<td>0.68</td>
<td></td>
</tr>
<tr>
<td>C3</td>
<td>0.37</td>
<td>0.48</td>
<td>0.37</td>
<td>0.63</td>
<td></td>
</tr>
<tr>
<td>C4-</td>
<td>0.40</td>
<td>0.58</td>
<td>0.50</td>
<td>0.50</td>
<td></td>
</tr>
<tr>
<td>C5-</td>
<td>0.33</td>
<td>0.48</td>
<td>0.33</td>
<td>0.67</td>
<td></td>
</tr>
</tbody>
</table>

With eigenvalues of:

<table>
<thead>
<tr>
<th></th>
<th>g</th>
<th>F1*</th>
<th>F2*</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>1.3</td>
<td>1.1</td>
<td></td>
</tr>
</tbody>
</table>
> keys <- make.keys(10,list(all=c(-1,2:8,-9,-10),agree=c(-1,2:5),con=c(6:8,-9,-10)))
> score.items(keys,bfi[1:10])

Call: score.items(keys = keys, items = bfi[1:10])

(Unstandardized) Alpha:

<table>
<thead>
<tr>
<th>all</th>
<th>agree</th>
<th>con</th>
</tr>
</thead>
<tbody>
<tr>
<td>alpha</td>
<td>0.72</td>
<td>0.65</td>
</tr>
</tbody>
</table>

Average item correlation:

<table>
<thead>
<tr>
<th>all</th>
<th>agree</th>
<th>con</th>
</tr>
</thead>
<tbody>
<tr>
<td>average.r</td>
<td>0.2</td>
<td>0.27</td>
</tr>
</tbody>
</table>

Guttman 6* reliability:

<table>
<thead>
<tr>
<th>all</th>
<th>agree</th>
<th>con</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lambda.6</td>
<td>0.74</td>
<td>0.62</td>
</tr>
</tbody>
</table>

Scale intercorrelations corrected for attenuation

raw correlations below the diagonal, alpha on the diagonal

<table>
<thead>
<tr>
<th>all</th>
<th>agree</th>
<th>con</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>0.72</td>
<td>1.10</td>
</tr>
<tr>
<td>agree</td>
<td>0.75</td>
<td>0.65</td>
</tr>
<tr>
<td>con</td>
<td>0.83</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Score 3 scales
Item scale correlations

Item by scale correlations:
corrected for item overlap and scale reliability

<table>
<thead>
<tr>
<th></th>
<th>all agree</th>
<th>con</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>-0.28</td>
<td>-0.37</td>
</tr>
<tr>
<td>A2</td>
<td>0.47</td>
<td>0.63</td>
</tr>
<tr>
<td>A3</td>
<td>0.47</td>
<td>0.63</td>
</tr>
<tr>
<td>A4</td>
<td>0.34</td>
<td>0.37</td>
</tr>
<tr>
<td>A5</td>
<td>0.44</td>
<td>0.58</td>
</tr>
<tr>
<td>C1</td>
<td>0.46</td>
<td>0.13</td>
</tr>
<tr>
<td>C2</td>
<td>0.47</td>
<td>0.19</td>
</tr>
<tr>
<td>C3</td>
<td>0.54</td>
<td>0.25</td>
</tr>
<tr>
<td>C4</td>
<td>-0.62</td>
<td>-0.30</td>
</tr>
<tr>
<td>C5</td>
<td>-0.50</td>
<td>-0.23</td>
</tr>
</tbody>
</table>
ICLUST shows 2 scales
Structural Equation modeling in R

I. sem by John Fox

II. does not do multiple group analyses

III. Mx in R is a coming attraction

IV. Using psych as a front end to sem to generate the model commands
Using psych as front end to sem

I. Do the exploratory analysis (fa or omega) in psych

II. output includes the sem model instructions

III. run sem

IV. (see the vignette on using psych for sem)
The model

```r
> om <- omega(Thurstone) # creates the path model and the model commands
> om$model

<table>
<thead>
<tr>
<th>Value</th>
<th>Path</th>
<th>Parameter</th>
<th>Initial</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1,]</td>
<td>&quot;g-&gt;Sentences&quot;</td>
<td>&quot;Sentences&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[2,]</td>
<td>&quot;g-&gt;Vocabulary&quot;</td>
<td>&quot;Vocabulary&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[3,]</td>
<td>&quot;g-&gt;Sent.Completion&quot;</td>
<td>&quot;Sent.Completion&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[4,]</td>
<td>&quot;g-&gt;First.Letters&quot;</td>
<td>&quot;First.Letters&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[5,]</td>
<td>&quot;g-&gt;4.Letter.Words&quot;</td>
<td>&quot;4.Letter.Words&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[6,]</td>
<td>&quot;g-&gt;Suffixes&quot;</td>
<td>&quot;Suffixes&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[7,]</td>
<td>&quot;g-&gt;Letter.Series&quot;</td>
<td>&quot;Letter.Series&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[8,]</td>
<td>&quot;g-&gt;Pedigrees&quot;</td>
<td>&quot;Pedigrees&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[9,]</td>
<td>&quot;g-&gt;Letter.Group&quot;</td>
<td>&quot;Letter.Group&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[10,]</td>
<td>&quot;F1*-&gt;Sentences&quot;</td>
<td>&quot;F1*Sentences&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[11,]</td>
<td>&quot;F1*-&gt;Vocabulary&quot;</td>
<td>&quot;F1*Vocabulary&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[12,]</td>
<td>&quot;F1*-&gt;Sent.Completion&quot;</td>
<td>&quot;F1*Sent.Completion&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[13,]</td>
<td>&quot;F2*-&gt;First.Letters&quot;</td>
<td>&quot;F2*First.Letters&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[15,]</td>
<td>&quot;F2*-&gt;Suffixes&quot;</td>
<td>&quot;F2*Suffixes&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[16,]</td>
<td>&quot;F3*-&gt;Letter.Series&quot;</td>
<td>&quot;F3*Letter.Series&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[17,]</td>
<td>&quot;F3*-&gt;Pedigrees&quot;</td>
<td>&quot;F3*Pedigrees&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[18,]</td>
<td>&quot;F3*-&gt;Letter.Group&quot;</td>
<td>&quot;F3*Letter.Group&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[19,]</td>
<td>&quot;Sentences&lt;-&gt;Sentences&quot;</td>
<td>&quot;e1&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[20,]</td>
<td>&quot;Vocabulary&lt;-&gt;Vocabulary&quot;</td>
<td>&quot;e2&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[21,]</td>
<td>&quot;Sent.Completion&lt;-&gt;Sent.Completion&quot;</td>
<td>&quot;e3&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[22,]</td>
<td>&quot;First.Letters&lt;-&gt;First.Letters&quot;</td>
<td>&quot;e4&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[23,]</td>
<td>&quot;4.Letter.Words&lt;-&gt;4.Letter.Words&quot;</td>
<td>&quot;e5&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[24,]</td>
<td>&quot;Suffixes&lt;-&gt;Suffixes&quot;</td>
<td>&quot;e6&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[25,]</td>
<td>&quot;Letter.Series&lt;-&gt;Letter.Series&quot;</td>
<td>&quot;e7&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[26,]</td>
<td>&quot;Pedigrees&lt;-&gt;Pedigrees&quot;</td>
<td>&quot;e8&quot;</td>
<td>NA</td>
</tr>
<tr>
<td>[27,]</td>
<td>&quot;Letter.Group&lt;-&gt;Letter.Group&quot;</td>
<td>&quot;e9&quot;</td>
<td>NA</td>
</tr>
</tbody>
</table>
```
The model

Omega

- Sentences
- Vocabulary
- Sent.Completion
- First.Leters
- 4.Letter.Words
- Suffixes
- Letter.Series
- Pedigrees
- Letter.Group

F1*
- 0.7
- 0.7
- 0.6
- 0.5
- 0.6
- 0.5
- 0.5
- 0.6

F2*
- 0.6
- 0.6
- 0.6
- 0.5
- 0.5
- 0.4
- 0.3
- 0.5

F3*
Do the sem

```r
> library(sem)
> sem.bf <- sem(om$model,Thurstone,213)
> summary(sem.bf,digits=2)

Model Chisquare =  24   Df =  18  Pr(>Chisq) = 0.15
Chisquare (null model) =  1102   Df =  36
Goodness-of-fit index =  0.98
Adjusted goodness-of-fit index =  0.94
RMSEA index =  0.04   90% CI: (NA, 0.078)
Bentler-Bonnett NFI =  0.98
Tucker-Lewis NNFI =  0.99
Bentler CFI =  1
SRMR =  0.035
BIC =  -72

Normalized Residuals
    Min.  1st Qu.   Median      Mean  3rd Qu.     Max.
-8.2e-01 -3.3e-01 -8.9e-07   2.8e-02  1.6e-01  1.8e+00
```
## Parameter values

| Parameter                  | Estimate | Std Error | z value | Pr(>|z|) | Relationship       |
|----------------------------|----------|-----------|---------|---------|--------------------|
| Sentences                  | 0.77     | 0.073     | 10.57   | 0.0e+00 | Sentences <---- g  |
| Vocabulary                 | 0.79     | 0.072     | 10.92   | 0.0e+00 | Vocabulary <---- g |
| Sent.Completion            | 0.75     | 0.073     | 10.27   | 0.0e+00 | Sent.Completion <---- g |
| First.Letters             | 0.61     | 0.072     | 8.43    | 0.0e+00 | First.Letters <---- g |
| 4.Letter.Words            | 0.60     | 0.074     | 8.09    | 6.7e-16 | 4.Letter.Words <---- g |
| Suffixes                  | 0.57     | 0.071     | 8.00    | 1.3e-15 | Suffixes <---- g  |
| Letter.Series             | 0.57     | 0.074     | 7.63    | 2.3e-14 | Letter.Series <---- g |
| Pedigrees                 | 0.66     | 0.069     | 9.55    | 0.0e+00 | Pedigrees <---- g  |
| Letter.Group              | 0.53     | 0.079     | 6.71    | 2.0e-11 | Letter.Group <---- g |
| F1*Sentences              | 0.49     | 0.085     | 5.71    | 1.1e-08 | Sentences <---- F1*|
| F1*Vocabulary             | 0.45     | 0.090     | 5.00    | 5.7e-07 | Vocabulary <---- F1*|
| F1*Sent.Completion        | 0.40     | 0.093     | 4.33    | 1.5e-05 | Sent.Completion <---- F1*|
| F2*First.Letters          | 0.61     | 0.086     | 7.16    | 8.2e-13 | First.Letters <---- F2*|
| F2*4.Letter.Words         | 0.51     | 0.085     | 5.96    | 2.5e-09 | 4.Letter.Words <---- F2*|
| F2*Suffixes               | 0.39     | 0.078     | 5.04    | 4.7e-07 | Suffixes <---- F2* |
| F3*Letter.Series          | 0.73     | 0.159     | 4.56    | 5.1e-06 | Letter.Series <---- F3*|
| F3*Pedigrees              | 0.25     | 0.089     | 2.77    | 5.6e-03 | Pedigrees <---- F3* |
| F3*Letter.Group           | 0.41     | 0.122     | 3.35    | 8.1e-04 | Letter.Group <---- F3*|
| e1                        | 0.17     | 0.034     | 5.05    | 4.4e-07 | Sentences <---- Sentences |
| e2                        | 0.17     | 0.030     | 5.65    | 1.6e-08 | Vocabulary <---- Vocabulary |
| e3                        | 0.27     | 0.033     | 8.09    | 6.7e-16 | Sent.Completion <---- Sent.Com |
| e4                        | 0.25     | 0.079     | 3.18    | 1.5e-03 | First.Letters <---- First.Letters |
| e5                        | 0.39     | 0.063     | 6.13    | 8.8e-10 | 4.Letter.Words <---- 4.Letter.W |
| e6                        | 0.52     | 0.060     | 8.68    | 0.0e+00 | Suffixes <---- Suffixes |
| e7                        | 0.15     | 0.223     | 0.67    | 5.0e-01 | Letter.Series <---- Letter.Series |
| e8                        | 0.50     | 0.060     | 8.39    | 0.0e+00 | Pedigrees <---- Pedigrees |
Programming in R

I. Very high level language
   A. interpreted at run time
   B. can integrate Fortran or C++ code

II. 3 ways of developing code
   A. cut and paste from an editor
   B. Modify prior code by adding or changing a function
   C. source from a file
   D. build a package for local or global distribution
Programming in R

I. functions and data structures

A. The output of all functions is an object that will have a certain structure

B. Part of the structure might be invisible but can be shown with the str() command.

1. f3 <- fa(Thurstone, 3, rotate=oblimin)

2. str(f3)  #will show more than just asking for f3
> f3 <- fa(Thurstone,3,rotate="oblimin")
> f3

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

<table>
<thead>
<tr>
<th>item</th>
<th>MR1</th>
<th>MR2</th>
<th>MR3</th>
<th>h2</th>
<th>u2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentences</td>
<td>1  0.90</td>
<td></td>
<td></td>
<td>0.82</td>
<td>0.18</td>
</tr>
<tr>
<td>Vocabulary</td>
<td>2  0.89</td>
<td></td>
<td></td>
<td>0.84</td>
<td>0.16</td>
</tr>
<tr>
<td>Sent.Completion</td>
<td>3  0.84</td>
<td></td>
<td></td>
<td>0.74</td>
<td>0.26</td>
</tr>
<tr>
<td>First.Letters</td>
<td>4  0.85</td>
<td>0.73</td>
<td>0.27</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.Letter.Words</td>
<td>5  0.75</td>
<td>0.63</td>
<td>0.37</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Suffixes</td>
<td>6  0.63</td>
<td>0.50</td>
<td>0.50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Letter.Series</td>
<td>7  0.84</td>
<td>0.72</td>
<td>0.28</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pedigrees</td>
<td>8  0.38</td>
<td>0.47</td>
<td>0.50</td>
<td>0.50</td>
<td></td>
</tr>
<tr>
<td>Letter.Group</td>
<td>9  0.63</td>
<td>0.52</td>
<td>0.48</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>MR1</th>
<th>MR2</th>
<th>MR3</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS loadings</td>
<td>2.64</td>
<td>1.87</td>
<td>1.49</td>
</tr>
<tr>
<td>Proportion Var</td>
<td>0.29</td>
<td>0.21</td>
<td>0.17</td>
</tr>
<tr>
<td>Cumulative Var</td>
<td>0.29</td>
<td>0.50</td>
<td>0.67</td>
</tr>
</tbody>
</table>

With factor correlations of

<table>
<thead>
<tr>
<th></th>
<th>MR1</th>
<th>MR2</th>
<th>MR3</th>
</tr>
</thead>
<tbody>
<tr>
<td>MR1</td>
<td>1.00</td>
<td>0.59</td>
<td>0.53</td>
</tr>
<tr>
<td>MR2</td>
<td>0.59</td>
<td>1.00</td>
<td>0.52</td>
</tr>
<tr>
<td>MR3</td>
<td>0.53</td>
<td>0.52</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Test of the hypothesis that 3 factors are sufficient.
The degrees of freedom for the model is 12 and the objective function was 0.01
Fit based upon off diagonal values = 1
Measures of factor score adequacy

<table>
<thead>
<tr>
<th></th>
<th>[,1]</th>
<th>[,2]</th>
<th>[,3]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correlation of scores with factors</td>
<td>0.96</td>
<td>0.92</td>
<td>0.90</td>
</tr>
<tr>
<td>Multiple R square of scores with factors</td>
<td>0.93</td>
<td>0.85</td>
<td>0.82</td>
</tr>
<tr>
<td>Minimum correlation of factor score estimates</td>
<td>0.86</td>
<td>0.71</td>
<td>0.63</td>
</tr>
</tbody>
</table>
All the output (too much)

```
> print(f3, all=TRUE)
Factor Analysis using method = minres
Call: fa(r = Thurstone, nfactors = 3, rotate = "oblimin")

<table>
<thead>
<tr>
<th></th>
<th>MR1</th>
<th>MR2</th>
<th>MR3</th>
<th>h2</th>
<th>u2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentences</td>
<td>1</td>
<td>0.90</td>
<td></td>
<td>0.82</td>
<td>0.18</td>
</tr>
<tr>
<td>Vocabulary</td>
<td>2</td>
<td>0.89</td>
<td></td>
<td>0.84</td>
<td>0.16</td>
</tr>
<tr>
<td>Sent.Completion</td>
<td>3</td>
<td>0.84</td>
<td></td>
<td>0.74</td>
<td>0.26</td>
</tr>
<tr>
<td>First.Letters</td>
<td>4</td>
<td>0.85</td>
<td></td>
<td>0.73</td>
<td>0.27</td>
</tr>
<tr>
<td>4.Letter.Words</td>
<td>5</td>
<td>0.75</td>
<td></td>
<td>0.63</td>
<td>0.37</td>
</tr>
<tr>
<td>Suffixes</td>
<td>6</td>
<td>0.63</td>
<td></td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>Pedigrees</td>
<td>7</td>
<td></td>
<td>0.84</td>
<td>0.72</td>
<td>0.28</td>
</tr>
<tr>
<td>Letter.Series</td>
<td>8</td>
<td>0.38</td>
<td></td>
<td>0.47</td>
<td>0.50</td>
</tr>
<tr>
<td>Letter.Group</td>
<td>9</td>
<td></td>
<td>0.63</td>
<td>0.52</td>
<td>0.48</td>
</tr>
</tbody>
</table>

SS loadings
2.484 1.731 1.343

<table>
<thead>
<tr>
<th></th>
<th>MR1</th>
<th>MR2</th>
<th>MR3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Letter.Group</td>
<td>0.21</td>
<td>0.63</td>
<td></td>
</tr>
<tr>
<td>Pedigrees</td>
<td>0.38</td>
<td></td>
<td>0.47</td>
</tr>
<tr>
<td>Letter.Series</td>
<td></td>
<td>0.84</td>
<td>0.72</td>
</tr>
<tr>
<td>Suffixes</td>
<td>0.18</td>
<td></td>
<td>0.47</td>
</tr>
<tr>
<td>4.Letter.Words</td>
<td>0.74</td>
<td></td>
<td>0.50</td>
</tr>
<tr>
<td>First.Letters</td>
<td></td>
<td>0.84</td>
<td>0.72</td>
</tr>
<tr>
<td>Sent.Completion</td>
<td>0.22</td>
<td></td>
<td>0.47</td>
</tr>
<tr>
<td>Vocabulary</td>
<td>0.38</td>
<td></td>
<td>0.47</td>
</tr>
</tbody>
</table>

With factor correlations of

<table>
<thead>
<tr>
<th></th>
<th>MR1</th>
<th>MR2</th>
<th>MR3</th>
</tr>
</thead>
<tbody>
<tr>
<td>MR1 1.00 0.59 0.53</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MR2 0.59 1.00 0.52</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MR3 0.53 0.52 1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Test of the hypothesis that 3 factors are sufficient.

The degrees of freedom for the model is 12 and the objective function was 0.01

Fit based upon off diagonal values = 1

Measures of factor score adequacy

<table>
<thead>
<tr>
<th></th>
<th>[,1]</th>
<th>[,2]</th>
<th>[,3]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correlation of scores with factors</td>
<td>0.96</td>
<td>0.92</td>
<td>0.90</td>
</tr>
<tr>
<td>Multiple R square of scores with factors</td>
<td>0.93</td>
<td>0.85</td>
<td>0.82</td>
</tr>
</tbody>
</table>

Minimum correlation of factor score estimates 0.86 0.71 0.63

$r.scores$

[,1] [,2] [,3]
[1,] 1.0000000 0.6615062 0.6122765
[2,] 0.6615062 1.0000000 0.6134362
[3,] 0.6122765 0.6134362 1.0000000

$r^2$

[,1] [,2] [,3]
[1,] 0.9285797 0.8525374 0.8161127

$valid$

[,1] [,2] [,3]
[1,] 0.9598034 0.9078568 0.8823116
> f3 <- fa(Thurstone, 3, rotate="oblimin")
> str(f3)
List of 22
$ residual   : num [1:9, 1:9] 0.1798 0.00357 0.00106 -0.00535 0.00549 ...
   ..- attr(*, "dimnames")=List of 2
   .. ..$ : chr [1:9] "Sentences" "Vocabulary" "Sent.Completion" "First.Letters" ...
   .. ..$ : chr [1:9] "Sentences" "Vocabulary" "Sent.Completion" "First.Letters" ...
$ fit         : num 0.957
$ fit.off     : num 1
$ factors     : num 3
$ n.obs       : logi NA
$ PVAL        : logi NA
$ dof         : num 12
$ objective   : num 0.0139
$ criteria    : Named num [1:3] 0.0139 NA NA
   ..- attr(*, "names")= chr [1:3] "objective" "" "" 
$ Call        : language fa(r = Thurstone, nfactors = 3, rotate = "oblimin")
$ r.scores    : num [1:3, 1:3] 1 0.662 0.612 0.662 1 ...
$ R2          : num [1:3] 0.929 0.853 0.816
$ valid       : num [1:3] 0.96 0.908 0.882
$ score.cor   : num [1:3, 1:3] 1 0.571 0.574 0.571 1 ...
$ weights     : num [1:9, 1:3] 0.35332 0.38596 0.2284 0.01086 0.00503 ...
   ..- attr(*, "dimnames")=List of 2
   .. ..$ : chr [1:9] "Sentences" "Vocabulary" "Sent.Completion" "First.Letters" ...
   .. ..$ : NULL
$ communality : Named num [1:9] 0.82 0.84 0.74 0.73 0.63 0.5 0.72 0.5 0.52
   ..- attr(*, "names")= chr [1:9] "Sentences" "Vocabulary" "Sent.Completion" "First.Letters" ...
$ uniquenesses: Named num [1:9] 0.18 0.16 0.26 0.27 0.37 0.5 0.28 0.5 0.48
   ..- attr(*, "names")= chr [1:9] "Sentences" "Vocabulary" "Sent.Completion" "First.Letters" ...
$ values      : num [1:9] 4.85 1.09 1.04 0.48 0.45 0.37 0.32 0.23 0.17
$ loadings    : loadings [1:9, 1:3] 0.90356 0.889 0.83522 -0.00297 -0.01535 ...
   ..- attr(*, "dimnames")=List of 2
   .. ..$ : chr [1:9] "Sentences" "Vocabulary" "Sent.Completion" "First.Letters" ...
   .. ..$ : chr [1:3] "MR1" "MR2" "MR3"
$ fm          : chr "minres"
$ Phi         : num [1:3, 1:3] 1 0.592 0.535 0.592 1 ...
$ fn          : chr "fa"
- attr(*, "class")= chr [1:2] "psych" "fa"
Programming in R

I. Data types
II. Operators
III. Simple functions
IV. Writing functions
Data structures

I. elements: logical, integer, real, character, factor

II. vectors: ordered sets of elements of one type

III. matrices: ordered sets of vectors (all of one type)

IV. data.frames: ordered sets of vectors, may be different types

V. lists: ordered set of anything
Operators

I. arithmetical

1. +, - , *, /, ^, %%

2. a + b, a-b, a*b, a/b, a^b, a %%b

II. Logical

A. a==b, !a, a!=b, a>b, a<b, a>=b, a <=b

III. Matrix

A. %*% is matrix multiplication

B. %o% is outer product
example operations

```r
> a <- 2
> b <- 3
> v <- 5:10
> w <- 6:7
> v
[1]  5  6  7  8  9 10
> w
[1] 6 7
> v ^ a
[1]  25  36  49  64  81 100
> w * b
[1] 18 21
> w * v
[1] 30 42 42 56 54 70
```
Matrix operations

> v
[1] 5 6 7 8 9 10
> t(v)
[1,]  5   6   7   8   9  10
> t(v) %*% v
   [,1]
[1,] 355
> v %*% t(v)
[1,]  25  30  35  40  45  50
[2,]  30  36  42  48  54  60
[3,]  35  42  49  56  63  70
[4,]  40  48  56  64  72  80
[5,]  45  54  63  72  81  90
[6,]  50  60  70  80  90 100
### Additional matrix operators

#### outer product

```r
> v
[1] 5 6 7 8 9 10
> w
[1] 6 7
> v %o% w
[,1] [,2]
[1,]  30  35
[2,]  36  42
[3,]  42  49
[4,]  48  56
[5,]  54  63
[6,]  60  70
```

#### matrix “addition” (psych)

```r
> x <- seq(4,8,2)
> x
[1] 4 6 8
> x %+% t(x)
[,1] [,2] [,3]
[1,]  8  10  12
[2,] 10  12  14
[3,] 12  14  16
```

#### kronecker

```r
```

```r
```
Functions

I. Operate on an object and provide a new object

II. e.g., f <- function(x) {x * 2}
```r
> f <- function(x) {x * 2}
> f(43)
[1] 86
> x
[1] 4 6 8
> f(x)
[1]  8 12 16
> f(v %o% w)
          [,1] [,2]
[1,]      60   70
[2,]      72   84
[3,]      84   98
[4,]      96  112
[5,]     108  126
[6,]     120  140
```
a subset of useful functions

I. is.na(), is.null(), is.vector(), is.matrix(), is.list()

II. sum(), rowSums(), colSums(), mean(x), rowMeans(), colMeans(), max, min, median
   (these work on the entire matrix)

III. var, cov, cor, sd (these work on the columns of the matrix/data.frame)

IV. help.start() brings up a web page of manuals
More useful functions

I. rep(x,n) (repeats the value x n times)
II. c(x,y) (combines x with y)
III. cbind(x,y) combines column wise
IV. rbind(x,y) combines rowwise
V. seq(a,b,c) sequence from a to b stepping by c
sums on matrices and data.frames

> z <- f(v %o% w)
> z
 [,1] [,2]
[1,]  60  70
[2,]  72  84
[3,]  84  98
[4,]  96 112
[5,] 108 126
[6,] 120 140

> sum(z)
[1] 1170
> min(z)
[1] 60
> max(z)
[1] 140
> median(z)
[1] 97

> rowSums(z)
[1] 130 156 182 208 234 260
> colSums(z)
[1] 540 630
> mean(z)
[1] 97.5
> rowMeans(z)
[1] 65 78 91 104 117 130
Basic stats functions, part 2
?cor

var(x, y = NULL, na.rm = FALSE, use)

cov(x, y = NULL, use = "everything",
    method = c("pearson", "kendall",
    "spearman"))

cor(x, y = NULL, use = "everything",
    method = c("pearson", "kendall",
    "spearman"))

cov2cor(V)
More on cor

x
a numeric vector, matrix or data frame.
y
NULL (default) or a vector, matrix or data frame with compatible dimensions to x. The default is equivalent to y = x (but more efficient).
na.rm
logical. Should missing values be removed?
use
an optional character string giving a method for computing covariances in the presence of missing values. This must be (an abbreviation of) one of the strings "everything", "all.obs", "complete.obs", "na.or.complete", or "pairwise.complete.obs".
method
a character string indicating which correlation coefficient (or covariance) is to be computed. One of "pearson" (default), "kendall", or "spearman", can be abbreviated.
V
symmetric numeric matrix, usually positive definite such as a covariance matrix.
row and col as functions

> r <- .8
> R <- diag(1,8)
> R

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>[1,]</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>[2,]</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>[3,]</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>[4,]</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>[5,]</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>[6,]</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>[7,]</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>[8,]</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

> R <- r^(abs(row(R)-col(R)))
> round(R,2)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>[1,]</td>
<td>1.00</td>
<td>0.80</td>
<td>0.64</td>
<td>0.51</td>
<td>0.41</td>
<td>0.33</td>
<td>0.26</td>
</tr>
<tr>
<td>[2,]</td>
<td>0.80</td>
<td>1.00</td>
<td>0.80</td>
<td>0.64</td>
<td>0.51</td>
<td>0.41</td>
<td>0.33</td>
</tr>
<tr>
<td>[3,]</td>
<td>0.64</td>
<td>0.80</td>
<td>1.00</td>
<td>0.80</td>
<td>0.64</td>
<td>0.51</td>
<td>0.41</td>
</tr>
<tr>
<td>[4,]</td>
<td>0.51</td>
<td>0.64</td>
<td>0.80</td>
<td>1.00</td>
<td>0.80</td>
<td>0.64</td>
<td>0.51</td>
</tr>
<tr>
<td>[5,]</td>
<td>0.41</td>
<td>0.51</td>
<td>0.64</td>
<td>0.80</td>
<td>1.00</td>
<td>0.80</td>
<td>0.64</td>
</tr>
<tr>
<td>[6,]</td>
<td>0.33</td>
<td>0.41</td>
<td>0.51</td>
<td>0.64</td>
<td>0.80</td>
<td>1.00</td>
<td>0.80</td>
</tr>
<tr>
<td>[7,]</td>
<td>0.26</td>
<td>0.33</td>
<td>0.41</td>
<td>0.51</td>
<td>0.64</td>
<td>0.80</td>
<td>1.00</td>
</tr>
<tr>
<td>[8,]</td>
<td>0.21</td>
<td>0.26</td>
<td>0.33</td>
<td>0.41</td>
<td>0.51</td>
<td>0.64</td>
<td>0.80</td>
</tr>
</tbody>
</table>
Yet more stats functions

I. `sample(n, N, replace=TRUE)`

II. `eigen(X)` (eigen value decomposition of X)

III. `solve(X)` (inverse of X)

IV. `solve(X,Y)` Regression of Y on X
Creating a matrix

```r
> x <- matrix(sample(10,50,replace=TRUE),ncol=5)
> x

[1,]  10   3   4   4   6
[2,]   3  10   8   8   9
[3,]   1   6   5   8   5
[4,]   9   1   3   5   3
[5,]   6   8   3   5   1
[6,]   8   6  10   1  10
[7,]  10   5  10   2   1
[8,]   9   3   2   2   9
[9,]   6  10   2   9   4
[10,]  1   8   8   2   6
```

standardize it

```r
> z <- scale(x)
> z

[1,] 1.0483835 -0.9819805 -0.4678877 -0.2059329  0.1852621
[2,] -0.9350447  1.3093073  0.7798129  1.1669533  1.1115724
[3,] -1.5017385  0.0000000 -0.1559626  1.1669533 -0.1235080
...[9,] -0.0850040  1.3093073 -1.0917380  1.5101749 -0.4322782
[10,] -1.5017385  0.6546537  0.7798129 -0.8923761  0.1852621
attr("scaled:center")
[1] 6.3 6.0 5.5 4.6 5.4
attr("scaled:scale")
```
Just center it

```r
> c <- scale(x, scale=FALSE)

> c

[1,]  3.7  -3  -1.5 -0.6  0.6
[2,] -3.3   4   2.5  3.4  3.6
[3,] -5.3   0  -0.5  3.4 -0.4
[4,]  2.7  -5  -2.5  0.4 -2.4
[5,] -0.3   2  -2.5  0.4 -4.4
[6,]  1.7   0   4.5 -3.6  4.6
[7,]  3.7  -1   4.5 -2.6 -4.4
[8,]  2.7  -3  -3.5 -2.6  3.6
[9,] -0.3   4  -3.5  4.4 -1.4
[10,] -5.3   2  2.5 -2.6  0.6

attr(, "scaled:center")
[1]  6.3  6.0  5.5  4.6  5.4
```
Find the covariance and inverse

```r
> c <- cov(x)
> round(c, 2)
[1,] 12.46 -6.89 -1.61 -4.53 -1.58
[2,] -6.89  9.33  2.11  4.11  0.56
[3,] -1.61  2.11 10.28 -3.89  2.22
[4,] -4.53  4.11 -3.89  8.49 -1.60
[5,] -1.58  0.56  2.22 -1.60 10.49
```

```r
> round(solve(c), 2)
[1,] 0.15  0.08  0.02  0.06  0.02
[2,] 0.08  0.23  0.07 -0.10  0.00
[3,] 0.02  0.07  0.16  0.12 -0.01
[4,] 0.06 -0.10  0.12  0.26  0.03
[5,] 0.02  0.00 -0.01  0.03  0.10
```
Flow control

I. if(condition) {then do this}  else {do this}

II. for (condition) do {expression}
   A. for (i in 1:n} do {x <- x + 1}

III. while (condition) {expression}
conditionals

I. \((a \& b)\) vs. \((a \&\& b)\)

II. \((a \mid b)\) vs. \((a \| b)\)

```r
> a <- 1
> if (a & b) {print("hello")} else {print("goodby")}
Error: object 'b' not found
> if (a && b) {print("hello")} else {print("goodby")}
[1] "goodby"
> if (a | b) {print("hello")} else {print("goodby")}
Error: object 'b' not found
> if (a || b) {print("hello")} else {print("goodby")}
Error: object 'b' not found
> a <- 1
> if (a || b) {print("hello")} else {print("goodby")}
[1] "hello"
> 
```
simple control

```r
> a <- 1
> b <- 2
> c <- 3
> k <- 10
> x <- 1
> if(x == a) {print("x is the same as a and has value",x)} else {print ("x is not equal to a")}

> x <- 3
> if(x == a) {print("x is the same as a and has value",x)} else {print ("x is not equal to a")}
[1] "x is not equal to a"
>```
Make that a function

```r
> f1 <- function(x,y) {if(x == y) {print ("x is the same as y and has value",x)} else {print ("x is not equal to y")}}
> f1(3,4)
[1] "x is not equal to y"
> f1(5,5)
[1] "x is the same as y and has value"
```
Simple functions: part 2

I. Find the squared multiple correlation of a variable with all the other variables in a matrix.

II. The $R^2$ is 1 - the residual variance
The essence of the function

```r
SMC <- function(R) {
  R.inv <- solve(R)
  SMC <- 1 - 1/diag(R.inv)}

> S <- cor(attitude)
> SMC(S)      #does not show anything

> round(SMC(S),2) #but this does
  rating complaints privileges learning raises critical advance
  0.73  0.77  0.38  0.62  0.68
  0.19  0.52
```
SMC <- function(R) {
  R.inv <- solve(R)
  SMC <- 1 - 1/diag(R.inv)
  return(SMC)}

> SMC(S)
     rating complaints privileges learning raises
critical  0.7326020  0.7700868  0.3831176  0.6194561  0.6770498
    advance  0.1881465  0.5186447

Allow it to find R

SMC <- function(R) {
  if(dim(R)[1] != dim(R)[2]) {R <<- cor(R)}
  R.inv <- solve(R)
  SMC <- 1 - 1/diag(R.inv)
  return(SMC)}

> SMC(attitude)
       rating complaints privileges learning raises
critical advance
     0.7326020  0.7700868  0.3831176  0.6194561  0.6770498
     0.1881465  0.5186447
SMC <- function(R,digits=2) {
  if(dim(R)[1] != dim(R)[2]) {R <- cor(R)}
  R.inv <- solve(R)
  SMC <- 1 - 1/diag(R.inv)
  return(round(SMC, digits))
}

> SMC(attitude)
    rating complaints privileges learning raises critical advance
     0.73    0.77      0.38     0.62    0.68     0.19
     0.52
Check for poor input

> att <- data.frame(attitude[1:3],attitude[1:3])
> SMC(att)
Error in solve.default(R) :
  Lapack routine dgesv: system is exactly singular
Add checks for weird matrices

SMC <- function(R,digits=2) {
  p <- dim(R)[2]
  if (dim(R)[1] != p) {R <- cor(R)}
  R.inv <- try(solve(R),TRUE)
  if(class(R.inv)== as.character("try-error")) {SMC <- rep(1,p)
    warning("Correlation matrix not invertible, smc's returned as 1s")}
  else {smc <- 1 -1/diag(R.inv)
    SMC <- 1 - 1/diag(R.inv)}
  return(round(SMC,digits))}

> SMC(att)
[1] 1 1 1 1 1 1 1
Warning message:
In SMC(att) : Correlation matrix not invertible, smc's returned as 1s

> SMC(attitude)
        rating complaints privileges learning raises critical
rating   0.73        0.77       0.38   0.62   0.68     0.19
Further checks

Input is a covariance matrix

```
> SMC(cov(attitude))
  rating complaints privileges learning raises critical advance
    -38.62    -39.76   -91.35   -51.42   -33.91   -78.49
    -49.96
```

Input is raw data or correlations

```
> SMC(cor(attitude))
  rating complaints privileges learning raises critical advance
    0.73      0.77      0.38     0.62     0.68     0.19
    0.52
> SMC(attitude)
  rating complaints privileges learning raises critical advance
    0.73      0.77      0.38     0.62     0.68     0.19
    0.52
```
#modified Dec 10, 2008 to return 1 on diagonal if non-invertible
#modified March 20, 2009 to return smcs * variance if covariance matrix is desired
#modified April 8, 2009 to remove bug introduced March 10 when using covar from data
"smc" <- function(R,covar =FALSE) {
  failed=FALSE
  p <- dim(R)[2]
  if (dim(R)[1] != p) {if(covar) {C <- cov(R, use="pairwise")
          vari <- diag(C)
          R <- cov2cor(C)
      } else {R <- cor(R,use="pairwise")}}
  else {vari <- diag(R)
      R <- cov2cor(R)
      if (!is.matrix(R)) R <- as.matrix(R)}
  R.inv <- try(solve(R),TRUE)
  if(class(R.inv)== as.character("try-error")) {smc <- rep(1,p)
    warning("Correlation matrix not invertible, smc's returned as 1s")
  } else {smc <- 1 -1/diag(R.inv)
    if(covar) {smc <- smc * vari}
    return(smc)
  }
}
Creating a new function

I. Is there a base function to modify?

II. Consider the case of modifying Promax rotation to allow for any target matrix

III. Original promax (inside the factanal package) had been modified to report the factor correlation.

IV. This version was created with the assistance of Pat Shrout and Steve Miller
promax

> promax
function (x, m = 4)
{
  if (ncol(x) < 2)
    return(x)
  dn <- dimnames(x)
  xx <- varimax(x)
  x <- xx$loadings
  Q <- x * abs(x)^(m - 1)
  U <- lm.fit(x, Q)$coefficients
  d <- diag(solve(t(U) %*% U))
  U <- U %*% diag(sqrt(d))
  dimnames(U) <- NULL
  z <- x %*% U
  U <- xx$rotmat %*% U
  dimnames(z) <- dn
  class(z) <- "loadings"
  list(loadings = z, rotmat = U)
}
<environment: namespace:stats>
> Promax
function (x, m = 4) {
    if (!is.matrix(x) & !is.data.frame(x)) {
        if (!is.null(x$loadings))
            x <- as.matrix(x$loadings)
    }
    else {
        x <- x
    }
    if (ncol(x) < 2)
        return(x)
    dn <- dimnames(x)
    xx <- varimax(x)
    x <- xx$loadings
    Q <- x * abs(x)^(m - 1)
    U <- lm.fit(x, Q)$coefficients
    d <- diag(solve(t(U) ** U))
    U <- U ** diag(sqrt(d))
    dimnames(U) <- NULL
    z <- x ** U
    U <- xx$rotmat ** U
    ui <- solve(U)
    Phi <- ui ** t(ui)
    dimnames(z) <- dn
    class(z) <- "loadings"
    result <- list(loadings = z, rotmat = U, Phi = Phi)
    class(result) <- c("psych", "fa")
    return(result)}
"target.rot" <-
function (x, keys=NULL, m = 4) {
  if(!is.matrix(x) & !is.data.frame(x) ) {
    if(!is.null(x$loadings)) x <- as.matrix(x$loadings)
  } else {x <- x}
  if (ncol(x) < 2)
    return(x)
  dn <- dimnames(x)
  if(is.null(keys)) {xx <- varimax(x)
    x <- xx$loadings
    Q <- x * abs(x)^(m - 1)} else {Q <- keys}
  U <- lm.fit(x, Q)$coefficients
  d <- diag(solve(t(U) %*% U))
  U <- U %*% diag(sqrt(d))
  dimnames(U) <- NULL
  z <- x %*% U
  if (is.null(keys)) {U <- xx$rotmat %*% U } else {U <- U}
  ui <- solve(U)
  Phi <- ui %*% t(ui)
  dimnames(z) <- dn
  class(z) <- "loadings"
  result <- list(loadings = z, rotmat = U, Phi = Phi)
  class(result) <- c("psych","fa")
  return(result)
}
optim as “solver” for R

I. Many statistical functions are not closed form but rather are solved iteratively.

II. We start with a good guess and then minimize the function

III. optim will do this for functions where you manipulate one vector (which can of course actually be a matrix)
optim

optim() minimising 'wild function'
initial solution of traveling salesman problem
optim() 'solving' traveling salesman problem
Trying to make a new function to do OLS FA

I. First, look at current ML FA function
   A. `factanal`
   B. It turns out that the critical optimization is done in `factanal.fit.mle`, but where is that?

II. `getAnywhere(factanal.fit.mle)`
   A. then look at the code
   B. scratch your head and try running it
Sharing your code

I. Post the source code on your web site

II. develop a package which you keep on a “repository” on your web site

III. develop a package and upload it to CRAN
Package development

I. a somewhat dated tutorial is found at http://personality-project.org/r/makingpackages.html
Package development

I. package.skeleton(yourpackage)  #creates directories and subdirectories for a package
   A. this includes a number of different subdirectories and files.

II. prompt(yourfunction)  #creates a draft help file for your function

III. Document as you go
I. R CMD check mypackage  #makes sure all the code is correct, checks for matches with documentation, runs all the example,

II. R CMD make mypackage #builds the package, but does not check except for working R

   A. this will convert the helpfiles from Rd files to html, LaTeX, and pdf

III. R CMD install mypackage will add any changes to your current package
Using X11 to check packages

bash-3.2$ cd /Volumes/Test
bash-3.2$ R CMD check psych
* checking working pdflatex ... OK
* using log directory '/Volumes/Test/psych.Rcheck'
* using R version 2.9.2 Patched (2009-09-24 r43893)
* using session charset: ASCII
* checking for file 'psych/DESCRIPTION' ... OK
* this is package 'psych' version '1.0-82'
* checking package namespace information ... OK
* checking package dependencies ... OK
* checking if this is a source package ... OK
* checking for executable files ... OK
* checking whether package 'psych' can be installed ... OK
* checking package directory ... OK
* checking for portable file names ... OK
* checking for sufficient/correct file permissions ... OK
* checking DESCRIPTION meta-information ... OK
* checking top-level files ... OK
* checking index information ... OK
* checking package subdirectories ... OK
* checking R files for non-ASCII characters ... OK
* checking R files for syntax errors ... OK
* checking whether the package can be loaded ... OK
* checking whether the package can be loaded with stated dependencies ... OK
* checking whether the name space can be loaded with stated dependencies ... OK
* checking for unstated dependencies in R code ... OK
* checking S3 generic/method consistency ... OK
* checking replacement functions ... OK
* checking foreign function calls ... OK
* checking R code for possible problems ... OK
* checking Rd files ... OK
* checking Rd files against version 2 parser ... OK
* checking Rd cross-references ... OK
* checking for missing documentation entries ... OK
* checking for code/documentation mismatches ... OK
* checking Rd 'usage' sections ... OK
* checking data for non-ASCII characters ... OK
* checking examples ... OK
* checking package vignettes in 'inst/doc' ... OK
* checking PDF version of manual ... OK
The structure of a package
The package
R CMD build psych

** building package indices ...**
* DONE (psych)
* creating vignettes ... OK
* removing junk files
* excluding invalid files from 'psych'
subdirectory 'R' contains invalid file names:
correlations neo pearson power reliability.data.sets
checking for LF line-endings in source and make files
checking for empty or unneeded directories
* building 'psych-1.0-92.tar.gz'
R in the classroom

I. Undergraduate statistics and research methods
   
   A. `describe`, `pairs.panels`, `anova`, `lm`
   
   B. `plot`, `curve`, etc.

   C. see tutorials for 205 and 371

   D. simulations of data for simulated studies

   E. Examples of complex models
R in the classroom

I. Graduate
   A. data simulations
   B. data analysis
   C. longer tutorial