Psychology 405: Psychometric Theory
Even more of Factor analyis

William Revelle

Department of Psychology
Northwestern University
Evanston, Illinois USA

NORTHWESTERN UNIVERSITY

May, 2020

Outline

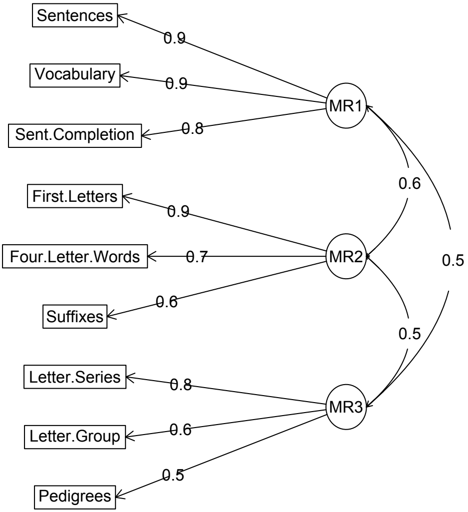
3 factor solution

R code

```
f3 <- fa(Thurstone,3)
f3   #show the results
fa.diagram(f3,main="3 Factors of Thurstone, oblimin transformation")
```

```
Factor Analysis using method =  minres
Call: fa(r = Thurstone, nfactors = 3)
Standardized loadings (pattern matrix) based upon correlation matrix
                   MR1   MR2   MR3   h2   u2 com
Sentences          0.90 -0.03  0.04 0.82 0.18 1.0
Vocabulary         0.89  0.06 -0.03 0.84 0.16 1.0
Sent.Completion    0.84  0.03  0.00 0.74 0.26 1.0
First.Letters      0.00  0.85  0.00 0.73 0.27 1.0
Four.Letter.Words -0.02  0.75  0.10 0.63 0.37 1.0
Suffixes           0.18  0.63 -0.08 0.50 0.50 1.2
Letter.Series      0.03 -0.01  0.84 0.73 0.27 1.0
Pedigrees          0.38 -0.05  0.46 0.51 0.49 2.0
Letter.Group      -0.06  0.21  0.63 0.52 0.48 1.2

                       MR1  MR2  MR3
SS loadings           2.65 1.87 1.49
Proportion Var        0.29 0.21 0.17
Cumulative Var        0.29 0.50 0.67
Proportion Explained  0.44 0.31 0.25
Cumulative Proportion 0.44 0.75 1.00

 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

Mean item complexity =  1.2
```

3 Factors of Thurstone, oblimin transformation



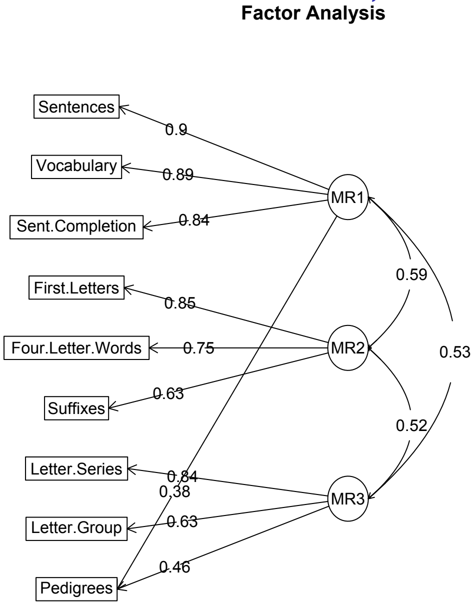
Graphic displays can be deceiving

1. The previous figure was *assuming* simple structure.
2. Allow for cross loadings
3. But once again, how large?

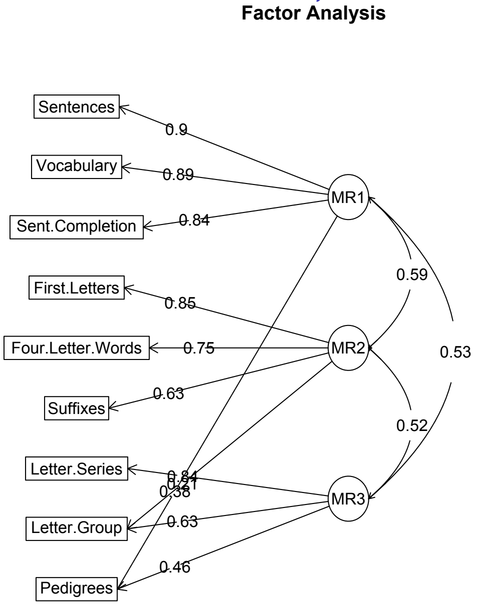
R code

```
fa.diagram(f3,digits=2,simple=FALSE)
> fa.diagram(f3,digits=2,simple=FALSE,cut=.2)
```

Thurstone 3 factor solution, don't force simple structure



Thurstone 3 factor solution, don't force simple structure, cut = .2



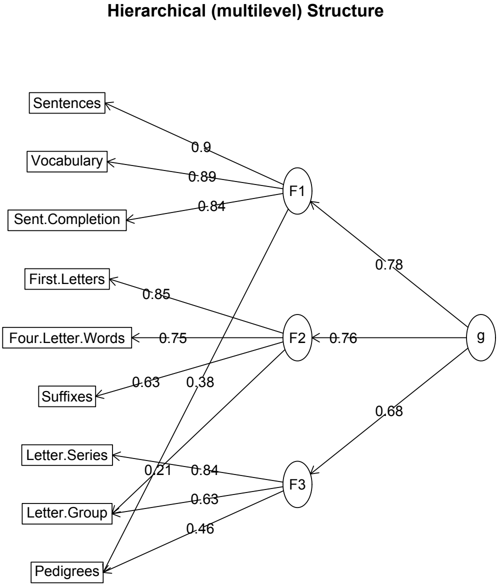
Try a hierarchical/higher order solution

R code

```
om <- omega(Thurstone)  #by default, chooses 3
omega.diagram(om,sl=FALSE,digits=2) #show the higher order solution
```

```
Omega
Call: omega(m = Thurstone)
Alpha:                  0.89
G.6:                    0.91
Omega Hierarchical:     0.74
Omega H asymptotic:     0.79
Omega Total             0.93

Schmid Leiman Factor loadings greater than  0.2
                     g   F1*   F2*   F3*   h2   u2   p2
Sentences         0.71  0.56              0.82 0.18 0.61
Vocabulary        0.73  0.55              0.84 0.16 0.63
Sent.Completion   0.68  0.52              0.74 0.26 0.63
First.Letters     0.65        0.56        0.73 0.27 0.57
Four.Letter.Words 0.62        0.49        0.63 0.37 0.61
Suffixes          0.56        0.41        0.50 0.50 0.63
Letter.Series     0.59              0.62 0.73 0.27 0.48
Pedigrees         0.58  0.24        0.34 0.51 0.49 0.66
Letter.Group      0.54              0.46 0.52 0.48 0.56

With eigenvalues of:
   g  F1*  F2*  F3*
3.58 0.96 0.74 0.72
```

The Omega higher order solution



Bifactor solution using omegaSem

R code

```
om.sem <- omegaSem(Thurstone,n.obs=213)
```

```
The following analyses were done using the  lavaan  package

 Omega Hierarchical from a confirmatory model using sem =  0.79
 Omega Total  from a confirmatory model using sem =  0.93
With loadings of
                   g  F1*  F2*  F3*   h2   u2   p2
Sentences         0.77 0.49             0.82 0.18 0.72
Vocabulary        0.79 0.45             0.83 0.17 0.75
Sent.Completion   0.75 0.40             0.73 0.27 0.77
First.Letters     0.61      0.61        0.74 0.26 0.50
Four.Letter.Words 0.60      0.50        0.61 0.39 0.59
Suffixes          0.57      0.39        0.48 0.52 0.68
Letter.Series     0.57           0.73 0.85 0.15 0.38
Pedigrees         0.66           0.25 0.50 0.50 0.87
Letter.Group      0.53           0.41 0.45 0.55 0.62

With eigenvalues of:
   g  F1*  F2*  F3*
3.86 0.60 0.78 0.75
```

omega from SEM versus omega from EFA

1. The `omega` function does an EFA of the data, and then does a Schmid Leiman Transformation to get the general factor
2. `omegaSem` function does the EFA, takes that converts it to a cluster structure of a general + group,and then calls the *lavaan* package to do the anaysis.
3. It will return (invisibly) the commands for the *lavaan* bifactor model
4. These loadings will differ somewhat, because the SEM model is a confirmatory model with 0 cross loadings.

```
$lavaan
[1] g =~ +Sentences+Vocabulary+Sent.Completion+First.Letters+Four.Letter.Words
             +Suffixes+Letter.Series+Pedigrees+Letter.Group
[2] F1=~  + Sentences + Vocabulary + Sent.Completion
[3] F2=~  + First.Letters + Four.Letter.Words + Suffixes
[4] F3=~  + Letter.Series + Pedigrees + Letter.Group
```

Yet another approach: the bifactor rotation

R code

```
bf <- fa(Thurstone,4,rotate="bifactor")
```

```
actor Analysis using method =  minres
Call: fa(r = Thurstone, nfactors = 4, rotate = "bifactor")
Standardized loadings (pattern matrix) based upon correlation matrix
                   MR1   MR2   MR3   MR4   h2     u2 com
Sentences         0.81 -0.04  0.41 -0.01 0.83 0.1677 1.5
Vocabulary        0.80  0.05  0.44  0.00 0.84 0.1639 1.6
Sent.Completion   0.75  0.03  0.40  0.04 0.73 0.2724 1.5
First.Letters     0.58  0.62  0.00 -0.03 0.73 0.2712 2.0
Four.Letter.Words 0.58  0.53 -0.06 -0.02 0.63 0.3704 2.0
Suffixes          0.51  0.47  0.13 -0.01 0.50 0.5021 2.1
Letter.Series     0.72 -0.07 -0.34  0.08 0.65 0.3513 1.5
Pedigrees         0.68 -0.03  0.00  0.73 1.00 0.0049 2.0
Letter.Group      0.66  0.07 -0.37  0.01 0.58 0.4236 1.6

                       MR1  MR2  MR3  MR4
SS loadings           4.21 0.91 0.80 0.55
Proportion Var        0.47 0.10 0.09 0.06
Cumulative Var        0.47 0.57 0.66 0.72
Proportion Explained  0.65 0.14 0.12 0.08
Cumulative Proportion 0.65 0.79 0.92 1.00

Mean item complexity =  1.8
Test of the hypothesis that 4 factors 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 6  and the objective function was  0.01

The root mean square of the residuals (RMSR) is  0
The df corrected root mean square of the residuals is  0.01

Fit based upon off diagonal values = 1
```

Bass Ackward as summary procedure

1. As suggested by Goldberg (2006) we can compare solutions of different number of factors
2. As originally suggested, this was a method of finding principal components, finding the component scores, and then correlating these scores
3. Waller (2007) pointed out that we don't need to find the component scores, but can do this from the linear algebra of the component solutions.
4. Because of the factor indeterminancy issue, Waller does not do this for factors.
5. We can find the factor correlation of different solutions, and the draw these

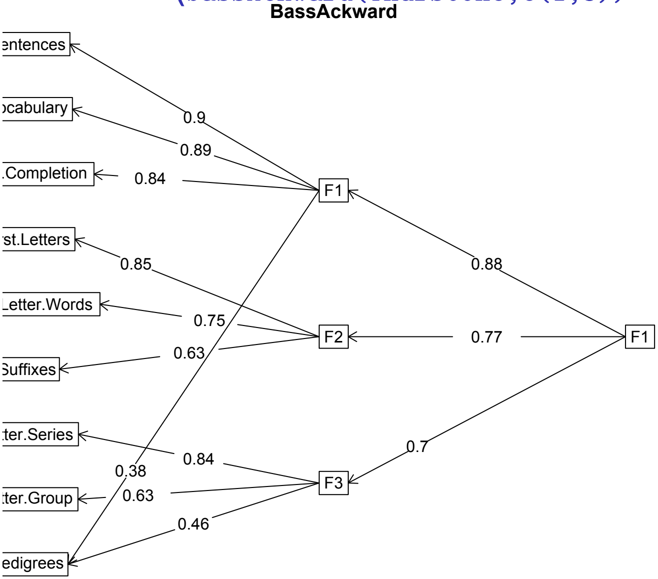
Factor correlations compared to factor score correlations

R code

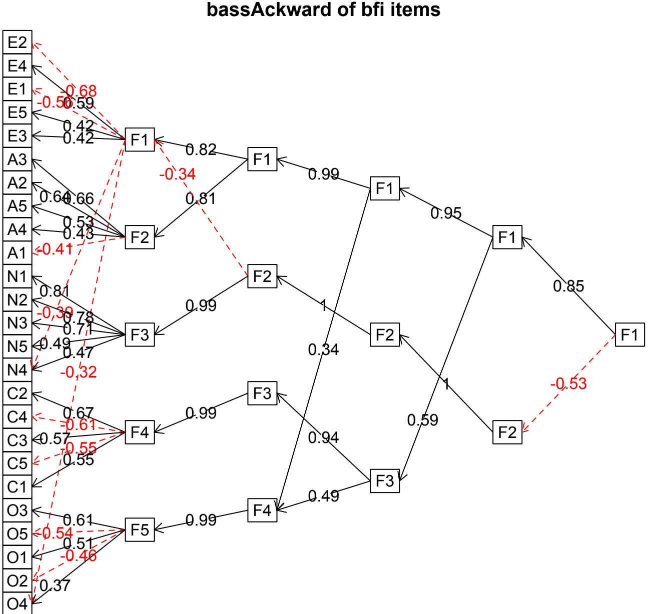
```
faCor(bfi[1:25],nfactors=c(3,5)) #based upon linear algebra
f3 <- fa(bfi[1:25],3,scores="tenBerge")
f5 <- fa(bfi[1:25],5 ,scores="tenBerge")
cor2(f3$scores,f5$scores) #the correlation between the factor score es
```

```
Call: faCor(r = bfi[1:25], nfactors = c(3, 5))

 Factor correlations between the two solutions
      MR2   MR1   MR3    MR5   MR4
MR1 -0.16  0.82  0.30  0.788 0.371
MR2  0.98 -0.34 -0.25 -0.013 0.039
MR3 -0.14  0.16  0.94  0.206 0.494

 Factor congruence between the two solutions
      MR2   MR1   MR3  MR5  MR4
MR1 -0.04  0.77  0.11 0.75 0.28
MR2  0.97 -0.22 -0.13 0.08 0.08
MR3 -0.04 -0.01  0.92 0.06 0.44



> cor2(f3$scores,f5$scores) #the correlation between the factor score estimates
      MR2   MR1   MR3   MR5  MR4
MR1 -0.16  0.82  0.30  0.79 0.37
MR2  0.98 -0.35 -0.25 -0.02 0.05
MR3 -0.13  0.18  0.94  0.20 0.49
```

Organize the correlations and draw them (bassAckward(Thurstone,c(1,3))



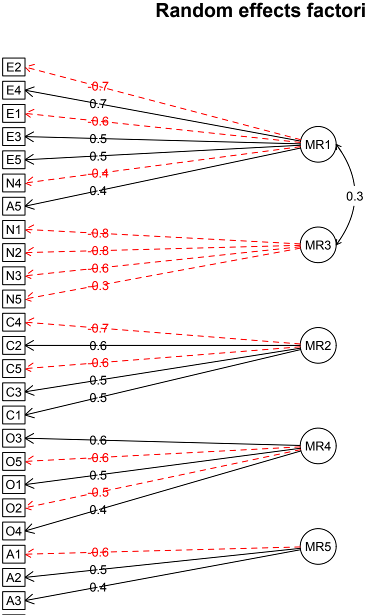
The Big 5 item set (bfi) has multiple solutions



Alternatives to bassAckward

1. `bassAckward` factors the data and compares alternative solutions
2. `omega` factors the data, correlates these factors and extracts a general factor
3. `fa.multi` factors the data, correlates these factors, and factors those (doesn't require a general factor)
4. `fa.random` is a random effects model suggesting that people differ in their overall level of respone.
   - The solution is to *ipsatize* each person (remove their total score) and then factor the ipsatized scores
   - Unfortunately, this solution is improper because the matrix is no longer of full rank.
   - Add a tiny bit of random noise to each subject
   - Functionally, this removes a general factor

A Random effects factoring of the bfi data set



Factors are a decomposition of correlations, Multidimensional Scaling examines distances

1. Convert correlations to distances
2. Apply multidimensional scaling procedures
3. Show them
4. The solutions differ because mds removes the overall general similarity

Convert Thurstone to distances

R code

```
T.dist <- cor2dist(Thurstone)
lowerMat(T.dist)
```

```
lowerMat(T.dist)
                  Sntnc Vcblr Snt.C Frs.L F.L.W Sffxs Ltt.S Pdgrs Ltt.G
Sentences         0.00
Vocabulary        0.59  0.00
Sent.Completion   0.67  0.66  0.00
First.Letters     1.06  1.01  1.04  0.00
Four.Letter.Words 1.07  1.04  1.07  0.81  0.00
Suffixes          1.05  1.01  1.06  0.91  0.96  0.00
Letter.Series     1.05  1.07  1.09  1.11  1.09  1.19  0.00
Pedigrees         0.96  0.96  0.97  1.14  1.13  1.17  0.94  0.00
Letter.Group      1.11  1.13  1.13  1.07  1.05  1.16  0.90  1.05  0.00
```

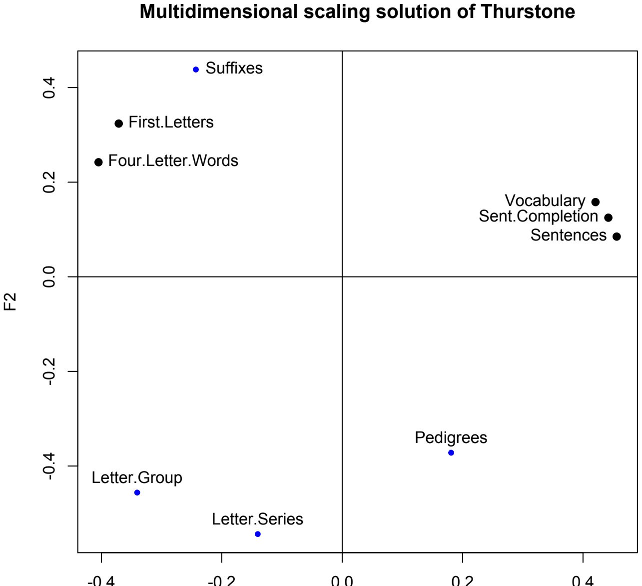
Do the Multidimensional scaling on the Thurstone Distances

R code

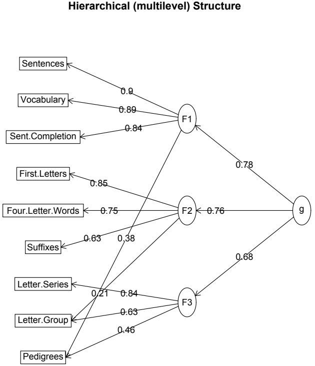
```
t.mds <- cmdscale(T.dist)
t.mds
fa.plot(t.mds,labels=colnames(T.dist),
         title="Multidimensional scaling solution of Thurtone",
           pos=c(2,2,2,4,4,4,3,3,3))
```

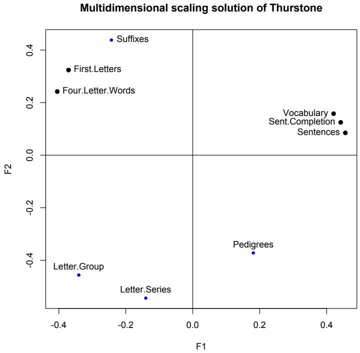
```
t.mds
                         [,1]        [,2]
Sentences           0.4559690  0.08496806
Vocabulary          0.4207616  0.15788958
Sent.Completion     0.4421132  0.12491838
First.Letters      -0.3711708  0.32388569
Four.Letter.Words -0.4047641  0.24217056
Suffixes           -0.2430563  0.43814906
Letter.Series      -0.1402873 -0.54391209
Pedigrees           0.1809522 -0.37192387
Letter.Group       -0.3405175 -0.45614537
```

Multidimensional solution to the Thurstone data



Compare solutions





The components model versus the factor model– what is the big deal?

1. Components are a model of complete data
   - Components are linear sums of the item
   - Component scores are defined by the data
   - $\boldsymbol{R} = \boldsymbol{CC'}$
2. Factors are models of the reliable part of the data
   - Items are linear sums of factors + error
   - Factor scores are *estimated* by the data
   - $\boldsymbol{R} = \boldsymbol{FF'} + \boldsymbol{U^2}$
   - Factors are the components of a reduced covariance matrix = $R* = R - U^2$
3. But most of R is the same for these two models, the difference is the diagonal (1 or $1 - U^2$)
4. The relevant importance of the diagonal decreases as the size of the matrix increases
5. $\boldsymbol{C} \longrightarrow_{\infty} \boldsymbol{F}$ as the number of variables increases

Factors and components 9 variables

R code

```
f3 <- fa(Thurstone,3)
p3 <- pca(Thurstone,3,rotate="oblimin")
```

```
Call: fa(r = Thurstone, nfactors = 3)
Standardized loadings (pattern matrix) based upon correlation matrix
                   MR1   MR2   MR3   h2   u2 com
Sentences         0.90 -0.03  0.04 0.82 0.18 1.
Vocabulary        0.89  0.06 -0.03 0.84 0.16 1.
Sent.Completion   0.84  0.03  0.00 0.74 0.26 1.
First.Letters     0.00  0.85  0.00 0.73 0.27 1.
Four.Letter.Words -0.02  0.75  0.10 0.63 0.37 1.
Suffixes          0.18  0.63 -0.08 0.50 0.50 1.
Letter.Series     0.03 -0.01  0.84 0.73 0.27 1.
Pedigrees         0.38 -0.05  0.46 0.51 0.49 2.
Letter.Group     -0.06  0.21  0.63 0.52 0.48 1.

                       MR1  MR2  MR3
SS loadings           2.65 1.87 1.49
Proportion Var        0.29 0.21 0.17
Cumulative Var        0.29 0.50 0.67
Proportion Explained  0.44 0.31 0.25
Cumulative Proportion 0.44 0.75 1.00

 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
```

```
Call: principal(r = r,)
Standardized loadings (pattern matrix) based u
                    TC1   TC2   TC3   h2   u2 c
Sentences          0.90  0.01  0.03 0.86 0.14 1
Vocabulary         0.88  0.10 -0.02 0.86 0.14 1
Sent.Completion    0.89  0.04 -0.01 0.83 0.17 1
First.Letters      0.03  0.84  0.07 0.78 0.22 1
Four.Letter.Words -0.03  0.81  0.16 0.75 0.25 1
Suffixes           0.17  0.79 -0.14 0.70 0.30 1
Letter.Series      0.10 -0.01  0.84 0.78 0.22 1
Pedigrees          0.49 -0.14  0.55 0.67 0.33 2
Letter.Group      -0.11  0.21  0.82 0.75 0.25 1

                       TC1  TC2  TC3
SS loadings           2.88 2.20 1.90
Proportion Var        0.32 0.24 0.21
Cumulative Var        0.32 0.56 0.78
Proportion Explained  0.41 0.32 0.27
Cumulative Proportion 0.41 0.73 1.00

 With component correlations of
     TC1  TC2  TC3
TC1 1.00 0.49 0.43
TC2 0.49 1.00 0.39
TC3 0.43 0.39 1.00
```

Factor and component correlations for Thurstone

R code

```
faCor(Thurstone,c(3,3),fm=c("minres","pca"))
```

```
Call: faCor(r = Thurstone, nfactors = c(3, 3), fm = c("minres", "pca"))

 Factor correlations between the two solutions
     TC1  TC2  TC3
MR1 0.99 0.53 0.45
MR2 0.55 0.98 0.48
MR3 0.51 0.44 0.98

 Factor congruence between the two solutions
     TC1  TC2  TC3
MR1 1.00 0.09 0.08
MR2 0.05 0.99 0.13
MR3 0.13 0.04 0.99
```

Factor and component correlations for 16 ability items

R code

```
faCor(ability,c(4,4),fm=c("minres","pca"))
```

```
Call: faCor(r = ability, nfactors = c(4, 4), fm = c("minres", "pca"))

 Factor correlations between the two solutions
     TC2  TC4  TC1  TC3
MR2 0.99 0.39 0.35 0.32
MR1 0.36 0.61 0.97 0.45
MR4 0.38 0.96 0.52 0.40
MR3 0.26 0.40 0.33 0.91

 Factor congruence between the two solutions
     TC2  TC4  TC1  TC3
MR2 1.00 0.08 0.06 0.09
MR1 0.04 0.21 0.97 0.19
MR4 0.07 0.95 0.09 0.12
MR3 0.01 0.13 0.06 0.92
```

Factors versus components for 25 `bfi` items

R code

```
faCor(bfi[1:25],c(5,5),fm=c("minres","pca"))
```

```
Call: faCor(r = bfi[1:25], nfactors = c(5, 5), fm = c("minres", "pca"))

 Factor correlations between the two solutions
        TC2   TC1   TC3    TC5    TC4
MR2  0.9910 -0.11 -0.13 -0.092 -0.024
MR1 -0.2533  0.98  0.20  0.173  0.025
MR3 -0.1905  0.22  0.99  0.151  0.142
MR5  0.0061  0.39  0.20  0.966  0.098
MR4  0.0085  0.24  0.17  0.160  0.980

 Factor congruence between the two solutions
      TC2   TC1   TC3   TC4   TC4
MR2  0.99 -0.03 -0.03 -0.08 -0.05
MR1 -0.16  0.98  0.06  0.09  0.00
MR3 -0.09  0.07  1.00  0.06  0.07
MR5  0.03  0.31  0.10  0.98  0.01
MR4 -0.01  0.20  0.07  0.06  0.99
```

Factor versus component solutions for the `spi` 135

R code

```
faCor(spi[11:145],c(5,5),fm=c("minres","pca"))
```

```
Call: faCor(r = spi[11:145], nfactors = c(5, 5), fm = c("minres", "pca"))

 Factor correlations between the two solutions
        TC2    TC1     TC5    TC4    TC3
MR2  0.9994 -0.085 -0.0042  0.155  0.060
MR1 -0.0859  0.999 -0.1899 -0.083 -0.055
MR5 -0.0089 -0.202  0.9990  0.038  0.195
MR4  0.1776 -0.093  0.0524  0.999  0.016
MR3  0.0577 -0.058  0.1805  0.030  0.999

 Factor congruence between the two solutions
      TC2   TC1   TC5   TC4   TC3
MR2  1.00 -0.05 -0.03  0.09  0.08
MR1 -0.05  1.00 -0.11 -0.06 -0.05
MR5 -0.03 -0.12  1.00  0.00  0.10
MR4  0.12 -0.07  0.01  1.00  0.01
MR3  0.08 -0.05  0.09  0.02  1.00
```

Factor versus components for the 72 mood items from the msqR

R code

```
faCor(msqR[1:72],c(2,2),fm=c("minres","pca"))
faCor(msqR[1:72],c(4,4),fm=c("minres","pca"))
```

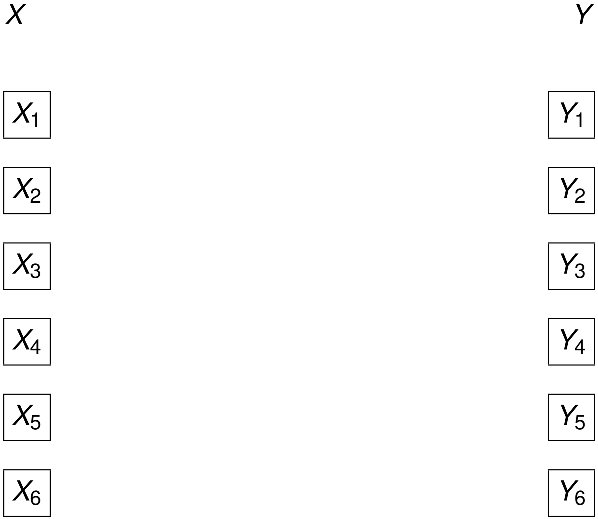
```
Call: faCor(r = msqR[1:72], nfactors = c(2, 2),
 fm = c("minres", "pca"))

 Factor correlations between the two solutions
      TC1   TC2
MR1  1.00 -0.11
MR2 -0.12  1.00

 Factor congruence between the two solutions
      TC1   TC2
MR1  1.00 -0.04
MR2 -0.04  1.00
```

```
Call: faCor(r = msqR[1:72], nfactors = c(4, 4),
 fm = c("minres", "pca"))

 Factor correlations between the two solutions
       TC1    TC2    TC3    TC4
MR1  0.998 -0.061  0.127 -0.418
MR2 -0.048  0.998 -0.319  0.253
MR3  0.125 -0.285  0.997  0.059
MR4 -0.436  0.192  0.057  0.990

 Factor congruence between the two solutions
      TC1   TC2   TC3   TC4
MR1  1.00  0.01  0.09 -0.18
MR2  0.02  1.00 -0.13  0.12
MR3  0.09 -0.09  1.00  0.08
MR4 -0.19  0.06  0.07  0.99
```

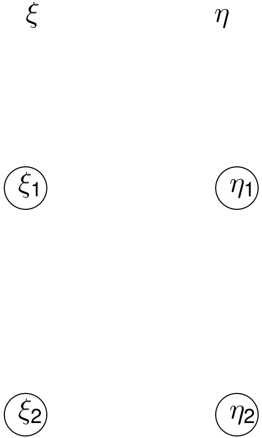
Why bother?

1. If the models are so similar, why do we care?
2. Factors introduce the concept of the latent score
3. This is relevant when we correct for reliability, particularly in Structural Equation Modeling
4. Factors force us to realize that what we see is not quite what we mean.

Observed Variables

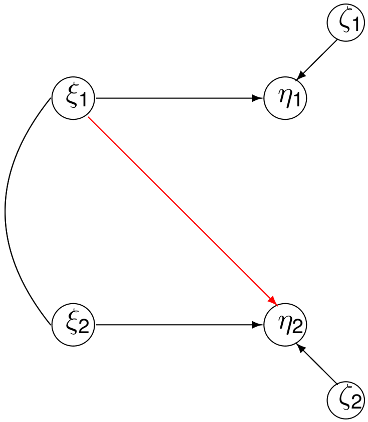


Latent Variables

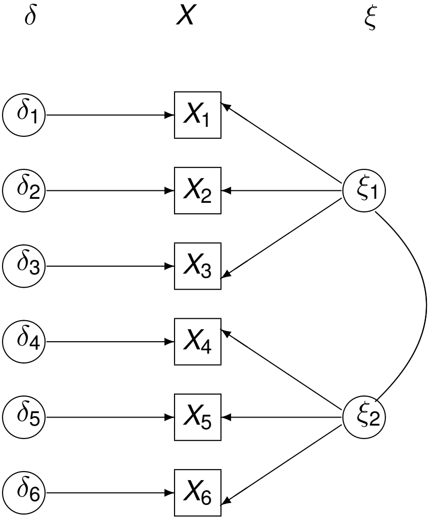


Theory: A regression model of latent variables

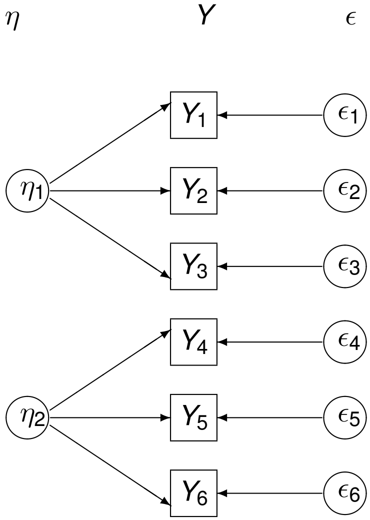
ξ η



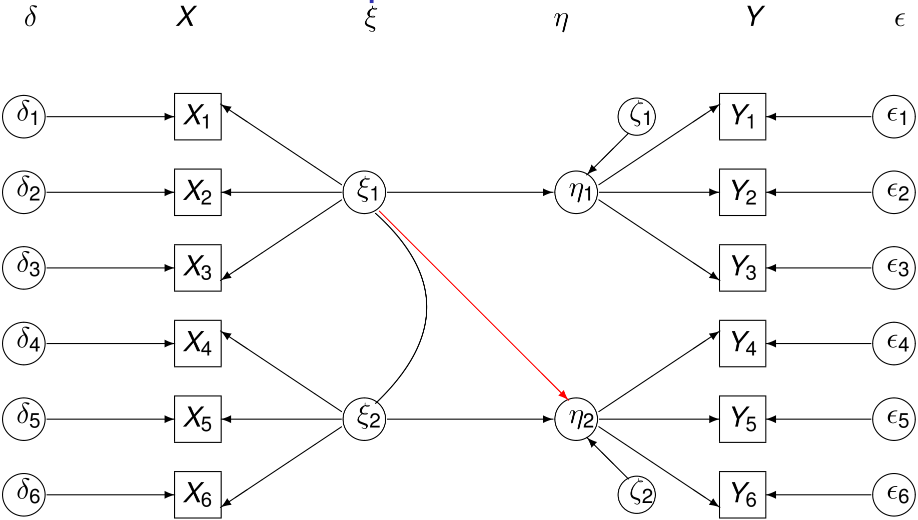
A measurement model for X



A measurement model for Y



A complete structural model



Goldberg, L. R. (2006). Doing it all bass-ackwards: The development of hierarchical factor structures from the top down. *Journal of Research in Personality*, *40*(4), 347 – 358.

Waller, N. (2007). A general method for computing hierarchical component structures by Goldberg's Bass-Ackwards method. *Journal of Research in Personality*, *41*(4), 745 – 752.