Psychology 454: Latent Variable Modeling

Advanced modeling with lavaan

Department of Psychology
Northwestern University
Evanston, Illinois USA

November 16, 2016

Outline

More examples from MPlus manual

- The *lavaan* introduction http://users.ugent.be/~yrosseel/lavaan/lavaanIntroduction.pdf gives examples from the MPlus User's guide http://www.statmodel.com/ugexcerpts.shtml
  - Regression and Path analysis
  - Growth Modeling
  - CFA and SEM
- The data examples may be downloaded and stored in a local directory.
- Then, set the working directory to that folder/directory, e.g.
  - `dir <- "/Volumes/WR/bill/Downloads/chapter3"`
  - `setwd(dir)`

Basic Syntax

formula type	operator	mnemonic
latent variable definition	=~	is measured by
regression	~	is regressed on
(residual) (co)variance	~~	is correlated with
intercept	~1	intercept

Basic Syntax – continued

R code

```
myModel <- '      #regressions
                     y1 + y2 ~ f1 + f2 + x1 + x2
                             f1 ~ f2 + f3
                             f2 ~ f3 + x1 + x2

                        #latent variable defintions
                        f1 =~ y1 + y2 + y3
                        f2 =~ y4 _+ y5 + y6
                        f3 =~ y7 + y8 +
                              y9  + y10

                        #variances and covariances
                           y1 ~~ y1
                           y1 ~~y2
                           f1 ~~ f2

                           #intercepts
                            y1 ~ 1
                            y2 ~ 1
          '
```

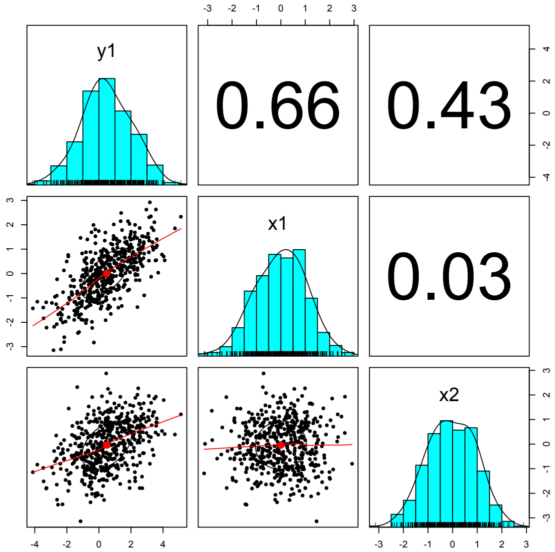
A simple regression example from MPlus

R code

```
fn <- "http://www.statmodel.com/usersguide/chap3/ex3.1.dat"
Data<- read.table(fn)
names(Data) <- c("y1","x1","x2")
describe(Data)
pairs.panels(Data)
```

```
   vars   n  mean   sd median trimmed  mad   min  max range  skew kurtosis   se
V1    1 500  0.48 1.55   0.43    0.49 1.51 -4.12 5.11  9.23 -0.01    -0.15 0.07
V2    2 500  0.00 1.05   0.02    0.01 1.12 -3.15 2.92  6.07 -0.13    -0.17 0.05
V3    3 500 -0.04 0.98  -0.04   -0.04 1.10 -3.14 2.88  6.01 -0.06    -0.37 0.04
```

The example data (Data set 3.1)



Example A.1: Regression and Path Analysis

```
fn <-   "http://www.statmodel.com/usersguide/chap3/ex3.1.dat"
# ex3.1
Data <- read.table(fn)
names(Data) <- c("y1","x1","x2")
model.ex3.1 <- ' y1 ~ x1 + x2 '
fit <- sem(model.ex3.1, data=Data)
summary(fit, standardized=TRUE, fit.measures=TRUE)
```

```
                   Estimate  Std.err  Z-value  P(>|z|)
Regressions:
  y1 ~
    x1                0.969    0.042   23.357    0.000
    x2                0.649    0.044   14.627    0.000
Variances:
    y1                0.941    0.060   15.811    0.000
#compare to linear regression
   > lm(y1~x1+x2,data=Data)
Call:
lm(formula = y1 ~ x1 + x2, data = Data)
Coefficients:
(Intercept)           x1           x2
     0.5110       0.9695       0.6490
```

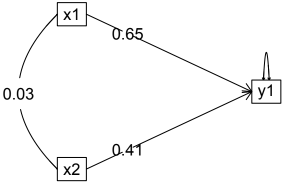
Compare to setCor

R code

```
setCor("y1",c("x1","x2"),data=Data)
```

```
Call: setCor(y = "y1", x = c("x1", "x2"), data = Data)

Multiple Regression from raw data

Beta weights
     y1
x1 0.65
x2 0.41

Multiple R
     y1
y1 0.78
multiple R2
     y1
y1 0.61

 Unweighted multiple R
  y1
0.76
 Unweighted multiple R2
  y1
0.58

 SE of Beta weights
     y1
x1 0.03
x2 0.03

 t of Beta Weights
```

Regression Models



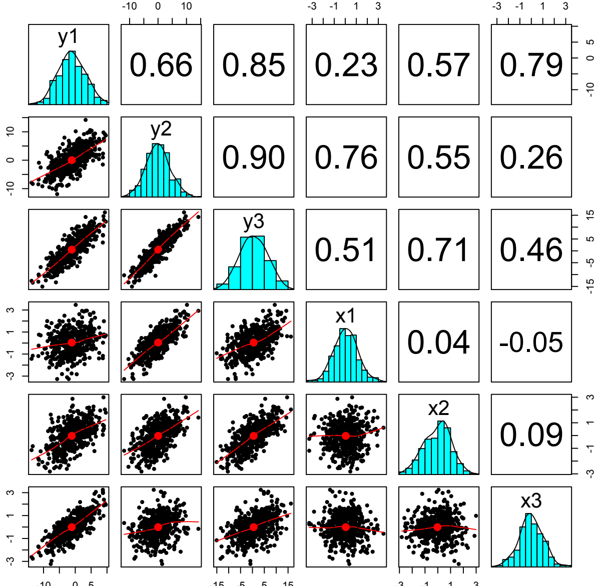
unweighted matrix correlation = 0.76

R code

```
# ex3.11
fn <- "http://www.statmodel.com/usersguide/chap3/ex3.11.dat"
Data <- read.table(fn)
names(Data) <- c("y1","y2","y3",
"x1","x2","x3")
describe(Data)
pairs.panels(Data)
```

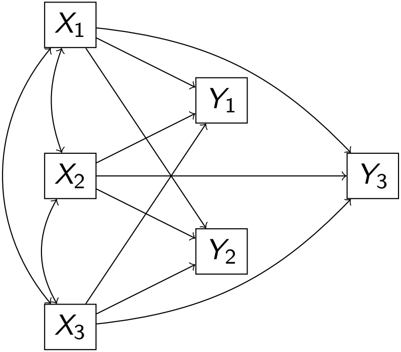
```
 describe(Data)
   vars   n  mean   sd median trimmed  mad    min   max range  skew kurtosis   se
y1    1 500 -1.11 4.18  -1.16   -1.13 4.08 -13.77  9.63 23.40  0.02    -0.20 0.19
y2    2 500  0.03 4.14  -0.07   -0.03 3.82 -11.93 14.19 26.12  0.12     0.02 0.19
y3    3 500  0.50 5.64   0.42    0.51 5.76 -15.07 16.22 31.29 -0.01    -0.39 0.25
x1    4 500  0.05 1.07   0.03    0.04 1.05  -3.27  3.47  6.74  0.01     0.30 0.05
x2    5 500 -0.03 1.03   0.09   -0.01 1.06  -2.82  2.99  5.81 -0.15    -0.29 0.05
x3    6 500 -0.01 1.04  -0.04    0.00 0.99  -3.23  3.25  6.48  0.03     0.27 0.05
```

Example 3.11 – 3 criteria, 3 predictors



More complicated regression - 3 dependent variables

```
# ex3.11
fn <- "http://www.statmodel.com/usersguide/chap3/ex3.11.dat"
Data <- read.table(fn)
names(Data) <- c("y1","y2","y3",
"x1","x2","x3")
model.ex3.11 <- ' y1 + y2  + y3 ~ x1 + x2 + x3 '
fit <- sem(model.ex3.11, data=Data)
summary(fit, standardized=TRUE)
```



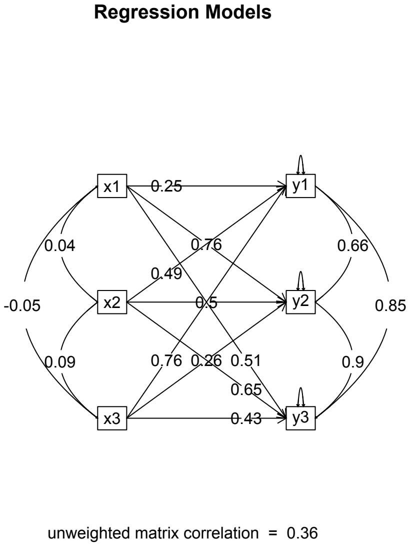
3 criteria, 3 predictors

```
                   Estimate  Std.err  Z-value  P(>|z|)   Std.lv  Std.all
Regressions:
  y1 ~
    x1                0.992    0.043   22.979    0.000    0.992    0.254
    x2                2.001    0.045   44.618    0.000    2.001    0.495
    x3                3.052    0.045   68.274    0.000    3.052    0.758
  y2 ~
    x1                2.935    0.050   59.002    0.000    2.935    0.759
    x2                1.992    0.052   38.556    0.000    1.992    0.497
    x3                1.023    0.051   19.869    0.000    1.023    0.256
  y3 ~
    x1                2.672    0.071   37.819    0.000    2.672    0.507
    x2                3.544    0.073   48.301    0.000    3.544    0.649
    x3                2.318    0.073   31.686    0.000    2.318    0.426

Covariances:
  y1 ~~
    y2                0.002    0.055    0.032    0.975    0.002    0.001
    y3                0.529    0.081    6.511    0.000    0.529    0.304
  y2 ~~
    y3                1.104    0.102   10.803    0.000    1.104    0.552

Variances:
    y1                1.061    0.067   15.811    0.000    1.061    0.061
    y2                1.408    0.089   15.811    0.000    1.408    0.082
    y3                2.842    0.180   15.811    0.000    2.842    0.089
```

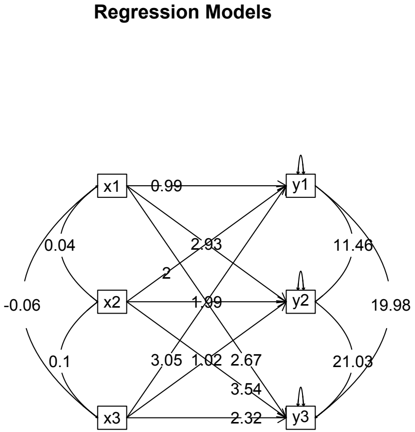
Compare to regression

```
setCor(c("y1","y2","y3"), x = c("x1","x2","x3"),data=Data,std=FALSE)
Call: setCor(y = c("y1", "y2", "y3"), x = c("x1", "x2", "x3"), data = Data,
    std = FALSE)

Beta weights
     y1   y2   y3
x1 0.99 2.93 2.67
x2 2.00 1.99 3.54
x3 3.05 1.02 2.32

Multiple R
  y1   y2   y3
0.97 0.96 0.95

Multiple R2
  y1   y2   y3
0.94 0.92 0.91

 SE of Beta weights
     y1   y2   y3
x1 0.04 0.05 0.07
x2 0.05 0.05 0.07
x3 0.04 0.05 0.07

 t of Beta Weights
      y1    y2    y3
x1 22.89 58.77 37.67
x2 44.44 38.40 48.11
x3 68.00 19.79 31.56

Probability of t <
   y1 y2 y3
x1  0  0  0
x2  0  0  0
x3  0  0  0

 Shrunken R2
  y1   y2   y3
0.94 0.92 0.91

Standard Error of R2
    y1     y2     y3
0.0052 0.0070 0.0076

F
     y1      y2      y3
2550.11 1843.07 1682.39

Probability of F <
y1 y2 y3
 0  0  0

 degrees of freedom of regression
[1]   3 496
```

Set Cor with standardized output



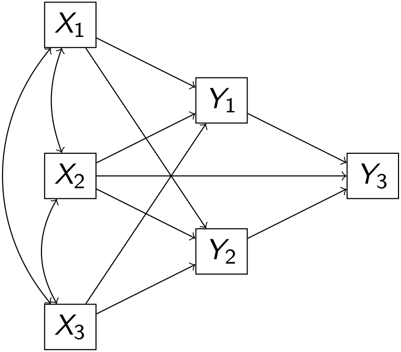
Set Cor with raw output



unweighted matrix correlation = 0.35

More complicated regression - a path model

```
# ex3.11
Data <- read.table("ex3.11.dat")
names(Data) <- c("y1","y2","y3",
"x1","x2","x3")
model.ex3.11 <- ' y1 + y2 ~ x1 + x2 + x3
y3 ~ y1 + y2 + x2 '
fit <- sem(model.ex3.11, data=Data)
summary(fit, standardized=TRUE, fit.measures=TRUE)
```



```
# ex3.11

Lavaan (0.4-7) converged normally after 46 iterations

  Number of observations                           500

  Estimator                                         ML
  Minimum Function Chi-square                    0.757
  Degrees of freedom                                 3
  P-value                                        0.860

Chi-square test baseline model:

  Minimum Function Chi-square                 4107.449
  Degrees of freedom                                12
  P-value                                        0.000

Full model versus baseline model:
  Comparative Fit Index (CFI)                    1.000
  Tucker-Lewis Index (TLI)                       1.002

Loglikelihood and Information Criteria:
  Loglikelihood user model (H0)              -4556.552
  Loglikelihood unrestricted model (H1)      -4556.174

  Number of free parameters                         12
  Akaike (AIC)                                9137.105
  Bayesian (BIC)                              9187.680
  Sample-size adjusted Bayesian (BIC)         9149.591

Root Mean Square Error of Approximation:
  RMSEA                                          0.000
  90 Percent Confidence Interval          0.000  0.040
  P-value RMSEA <= 0.05                          0.972

Standardized Root Mean Square Residual:
```

```
                   Estimate  Std.err  Z-value  P(>|z|)
Regressions:
  y1 ~
    x1                0.992    0.043   22.979    0.000
    x2                2.001    0.045   44.618    0.000
    x3                3.052    0.045   68.274    0.000
  y2 ~
    x1                2.935    0.050   59.002    0.000
    x2                1.992    0.052   38.556    0.000
    x3                1.023    0.051   19.869    0.000
  y3 ~
    y1                0.507    0.020   25.494    0.000
    y2                0.746    0.020   37.918    0.000
    x2                1.046    0.072   14.539    0.000

Variances:
    y1                1.061    0.067   15.811    0.000
    y2                1.408    0.089   15.811    0.000
    y3                1.717    0.109   15.811    0.000
```

But this is just a more tedious regression

```
mat.regress(Data,c("x1","x2","x3"),c("y1","y2"))

Call: mat.regress(m = Data, x = c("x1", "x2", "x3"),
          y = c("y1", "y2"))

Multiple Regression from matrix input

Beta weights
     y1   y2
x1 0.99 2.93
x2 2.00 1.99
x3 3.05 1.02

Multiple R
  y1   y2
0.97 0.96

Multiple R2
  y1   y2
0.94 0.92

 SE of Beta weights
     y1   y2
x1 0.04 0.05
x2 0.05 0.05
x3 0.04 0.05

F
     y1      y2
2550.11 1843.07
```

```
mat.regress(Data,c("y1","y2","x2"),c("y3"))

Call: mat.regress(m = Data, x = c("y1", "y2", "x2"),
            y = c("y3"))

Multiple Regression from matrix input

Beta weights
        y3
y1 0.51
y2 0.75
x2 1.05

Multiple R
   y3
0.97

Multiple R2
<NA>
0.95

 SE of Beta weights
     y3
y1 0.02
y2 0.02
x2 0.07

F
     y3
2892.53
```

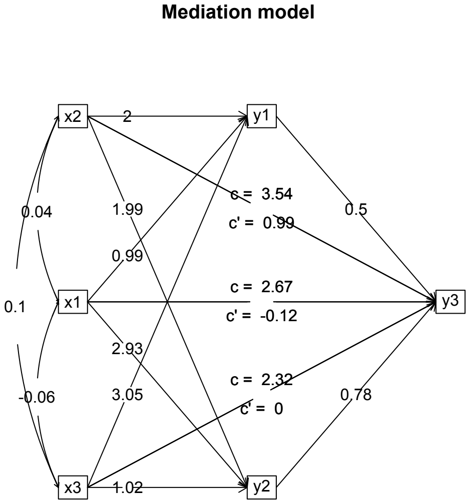
And not quite what we get when we try a mediation model

R code

```
mediate(y="y3", x = c("x1","x2","x3"),m=c("y1","y2"),data=Data)
```

```
Call: mediate(y = "y3", x = c("x1", "x2", "x3"), m = c("y1", "y2"),
    data = Data)

The DV (Y) was  y3 . The IV (X) was  x1 x2 x3 . The mediating variable(s) =  y1 y2 .

Total Direct effect(c) of  x1  on  y3  =  2.67   S.E. =  0.07  t direct =  37.67   with probability =
Direct effect (c') of  x1  on  y3  removing  y1 y2  =  -0.12   S.E. =  0.17  t direct =  -0.72   with p
Indirect effect (ab) of  x1  on  y3  through  y1 y2   =  2.79
Mean bootstrapped indirect effect =  2.79  with standard error =  0.17  Lower CI =  2.46    Upper CI =

Total Direct effect(c) of  x2  on  y3  =  3.54   S.E. =  0.07  t direct =  48.11   with probability =
Direct effect (c') of  x2  on  NA  removing  y1 y2  =  0.99   S.E. =  0.16  t direct =  6.13   with pro
Indirect effect (ab) of  x2  on  y3  through  y1 y2   =  2.55
Mean bootstrapped indirect effect =  2.79  with standard error =  0.17  Lower CI =  2.24    Upper CI =

Total Direct effect(c) of  x3  on  y3  =  2.32   S.E. =  0.07  t direct =  31.56   with probability =
Direct effect (c') of  x3  on  NA  removing  y1 y2  =  0   S.E. =  0.19  t direct =  0   with probabili
Indirect effect (ab) of  x3  on  y3  through  y1 y2   =  2.32
Mean bootstrapped indirect effect =  2.79  with standard error =  0.17  Lower CI =  1.95    Upper CI =
R2 of model =  0.95
 To see the longer output, specify short = FALSE in the print statement
```

Mediation model is not the same as the simple path diagram



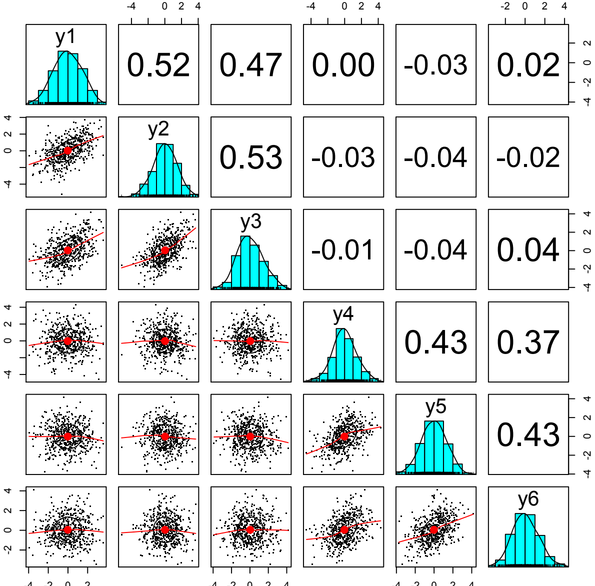
First, get the example data set

R code

```
fn <-  "http://www.statmodel.com/usersguide/chap5/ex5.1.dat"
Data <- read.table(fn)
names(Data) <- paste("y", 1:6, sep="")
describe(Data)
 pairs.panels(Data,pch=".")
```

```
 describe(Data)
   vars   n  mean   sd median trimmed  mad   min  max range  skew kurtosis   se
y1    1 500 -0.02 1.41  -0.04   -0.01 1.49 -3.96 3.59  7.55 -0.05    -0.37 0.06
y2    2 500  0.03 1.40   0.04    0.04 1.35 -5.19 3.70  8.90 -0.14     0.10 0.06
y3    3 500  0.03 1.40  -0.06    0.00 1.41 -3.91 4.16  8.07  0.17    -0.15 0.06
y4    4 500 -0.02 1.43  -0.07   -0.03 1.33 -4.55 4.29  8.83 -0.02     0.23 0.06
y5    5 500 -0.02 1.31  -0.01   -0.01 1.34 -3.73 4.15  7.88 -0.04    -0.11 0.06
y6    6 500  0.05 1.30   0.02    0.02 1.34 -3.42 4.11  7.53  0.21    -0.04 0.06
```

Graph example 5.1



Exploratory factor analysis

```
> fa(Data,2)
Loading required package: GPArotation
Factor Analysis using method =  minres
Call: fa(r = Data, nfactors = 2)
Standardized loadings based upon correlation matrix
     MR1   MR2   h2   u2
y1  0.68  0.01 0.46 0.54
y2  0.77 -0.02 0.59 0.41
y3  0.70  0.01 0.49 0.51
y4  0.00  0.61 0.37 0.63
y5 -0.02  0.70 0.50 0.50
y6  0.04  0.61 0.37 0.63

                MR1  MR2
SS loadings    1.54 1.24
Proportion Var 0.26 0.21
Cumulative Var 0.26 0.46

 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  15  and the
          objective function was  1.19 with Chi Square of  5
The degrees of freedom for the model are 4  and the
             objective function was  0

The root mean square of the residuals is  0.01
The df corrected root mean square of the residuals is  0.02
The number of observations was  500  with
            Chi Square =  2.18  with prob <  0.7

Tucker Lewis Index of factoring reliability =  1.012
RMSEA index =  0  and the 90 % confidence intervals are  0 (
BIC =  -22.68
Fit based upon off diagonal values = 1
Measures of factor score adequacy
                                                MR1  MR2
Correlation of scores with factors             0.87 0.83
Multiple R square of scores with factors       0.76 0.68
Minimum correlation of possible factor scores  0.53 0.37
```

Confirmatory Factor Analysis is almost identical to EFA because cross loadings are trivial

```
model.ex5.1 <- ' f1 =~ y1 + y2 + y3
f2 =~ y4 + y5 + y6 '
fit <- cfa(model.ex5.1, data=Data,std.ov=TRUE,std.lv=TRUE)
summary(fit, standardized=TRUE, fit.measures=TRUE)
```

```
Lavaan (0.4-7) converged normally after 16 iterations
  Number of observations                             500
  Estimator                                           ML
  Minimum Function Chi-square                      3.896
  Degrees of freedom                                   8
  P-value                                          0.866
Chi-square test baseline model:
  Minimum Function Chi-square                    596.921
  Degrees of freedom                                  15
  P-value                                          0.000
Full model versus baseline model:
  Comparative Fit Index (CFI)                      1.000
  Tucker-Lewis Index (TLI)                         1.013
Loglikelihood and Information Criteria:
  Loglikelihood user model (H0)                -3957.300
  Loglikelihood unrestricted model (H1)        -3955.352
  Number of free parameters                           13
  Akaike (AIC)                                  7940.600
  Bayesian (BIC)                                7995.390
  Sample-size adjusted Bayesian (BIC)           7954.127
Root Mean Square Error of Approximation:
  RMSEA                                            0.000
  90 Percent Confidence Interval          0.000    0.027
  P-value RMSEA <= 0.05                            0.995
Standardized Root Mean Square Residual:
  SRMR                                             0.016
```

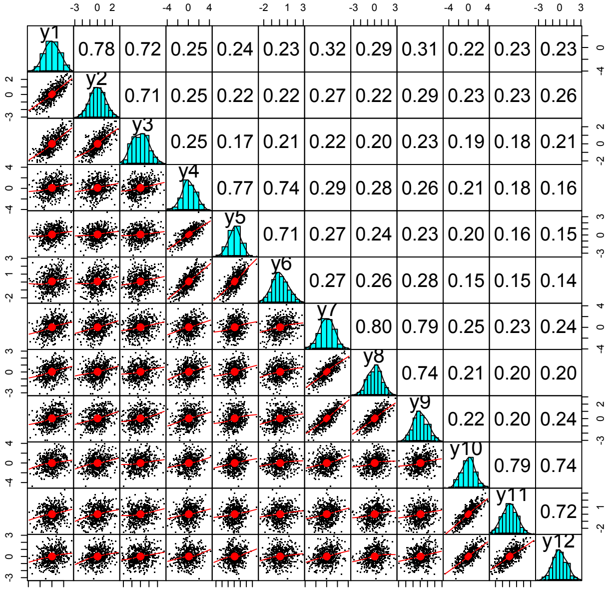
```
                   Estimate  Std.err  Z-value  P(>|z|)
Latent variables:
  f1 =~
    y1                0.678    0.047   14.508    0.000
    y2                0.768    0.047   16.244    0.000
    y3                0.694    0.047   14.832    0.000
  f2 =~
    y4                0.608    0.053   11.481    0.000
    y5                0.706    0.056   12.700    0.000
    y6                0.603    0.053   11.411    0.000

Covariances:
  f1 ~~
    f2               -0.036    0.062   -0.585    0.559

Variances:
    y1                0.539    0.048   11.124    0.000
    y2                0.408    0.051    7.981    0.000
    y3                0.516    0.049   10.600    0.000
    y4                0.628    0.058   10.880    0.000
    y5                0.500    0.065    7.736    0.000
    y6                0.634    0.057   11.035    0.000
    f1                1.000
    f2                1.000
```

R code

```
fn <- "http://www.statmodel.com/usersguide/chap5/ex5.6.dat"
Data <- read.table(fn)
names(Data) <- paste("y", 1:12, sep="")
describe(Data)
pairs.panels(Data,gap=0,pch=".")
```

	var	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
y1	1	500	0.01	1.31	0.02	0.01	1.38	-3.87	3.39	7.27	-0.04	-0.25	0.06
y2	2	500	0.03	1.02	0.01	0.03	1.01	-2.93	2.75	5.68	0.00	-0.17	0.05
y3	3	500	0.00	0.97	0.01	-0.02	1.03	-2.22	2.79	5.01	0.16	-0.36	0.04
y4	4	500	0.10	1.35	0.04	0.11	1.26	-3.91	4.19	8.11	-0.07	0.04	0.06
y5	5	500	0.08	1.01	0.11	0.09	1.00	-3.34	3.70	7.04	-0.12	0.18	0.05
y6	6	500	0.08	1.02	0.02	0.06	1.04	-2.46	2.95	5.41	0.16	-0.20	0.05
y7	7	500	0.02	1.38	0.05	0.04	1.37	-3.78	4.16	7.94	-0.06	-0.12	0.06
y8	8	500	0.03	1.04	0.08	0.03	1.04	-3.17	2.99	6.16	-0.11	-0.03	0.05
y9	9	500	0.03	1.03	-0.03	0.03	1.05	-2.98	2.92	5.90	0.04	-0.19	0.05
y10	10	500	-0.02	1.38	0.05	-0.02	1.31	-4.79	3.95	8.74	0.01	-0.04	0.06
y11	11	500	0.01	1.06	0.01	0.01	1.07	-2.62	3.46	6.08	-0.01	-0.16	0.05
y12	12	500	0.01	1.01	-0.07	0.00	1.01	-3.18	2.88	6.06	0.06	-0.07	0.05

Example 5.6: 12 variables, ? factors



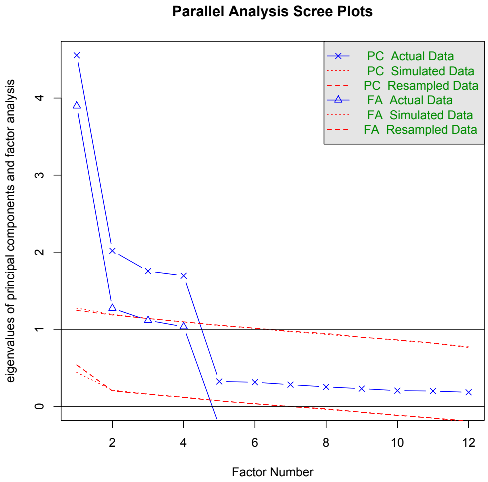
Better yet, just show the correlations

```
pairs.panels(Data,gap=0,pch=".")
loweCor(Data)
```

```
    V1   V2   V3   V4   V5   V6   V7   V8   V9   V10  V11  V12
V1  1.00
V2  0.78 1.00
V3  0.72 0.71 1.00
V4  0.25 0.25 0.25 1.00
V5  0.24 0.22 0.17 0.77 1.00
V6  0.23 0.22 0.21 0.74 0.71 1.00
V7  0.32 0.27 0.22 0.29 0.27 0.27 1.00
V8  0.29 0.22 0.20 0.28 0.24 0.26 0.80 1.00
V9  0.31 0.29 0.23 0.26 0.23 0.28 0.79 0.74 1.00
V10 0.22 0.23 0.19 0.21 0.20 0.15 0.25 0.21 0.22 1.00
V11 0.23 0.23 0.18 0.18 0.16 0.15 0.23 0.20 0.20 0.79 1.00
V12 0.23 0.26 0.21 0.16 0.15 0.14 0.24 0.20 0.24 0.74 0.72 1.00
```

Parallel Analysis

```
fa.parallel(Data)
Parallel analysis suggests that the number of factors = 4 and the number
of components = 4
```



Extract 4 factors using EFA

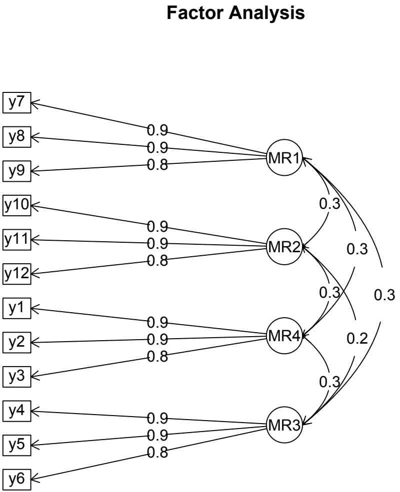
fa(Data,4)

```
Factor Analysis using method =  minres
Call: fa(r = Data, nfactors = 4)
Standardized loadings based upon correlation matrix
      MR1   MR2   MR4   MR3   h2   u2
y1    0.05 -0.01  0.87  0.00 0.79 0.21
y2   -0.01  0.02  0.87  0.00 0.77 0.23
y3   -0.04 -0.01  0.83  0.01 0.67 0.33
y4    0.00  0.01  0.01  0.88 0.80 0.20
y5   -0.02  0.01 -0.01  0.87 0.74 0.26
y6    0.02 -0.03  0.00  0.82 0.69 0.31
y7    0.92  0.01 -0.01  0.00 0.86 0.14
y8    0.87 -0.01 -0.02  0.01 0.74 0.26
y9    0.85  0.00  0.03 -0.01 0.74 0.26
y10   0.00  0.90 -0.03  0.02 0.81 0.19
y11  -0.01  0.88  0.00  0.00 0.76 0.24
y12   0.02  0.82  0.04 -0.03 0.68 0.32

                MR1  MR2  MR4  MR3
SS loadings    2.34 2.26 2.23 2.22
Proportion Var 0.19 0.19 0.19 0.19
Cumulative Var 0.19 0.38 0.57 0.75

 With factor correlations of
    MR1  MR2  MR4  MR3
MR1 1.00 0.29 0.34 0.35
MR2 0.29 1.00 0.29 0.22
MR4 0.34 0.29 1.00 0.30
MR3 0.35 0.22 0.30 1.00
```

```
Test of the hypothesis that 4 factors are sufficient.

The degrees of freedom for the null model are  66  and the
          objective function was  8.02 with Chi Square of  3965.23
The degrees of freedom for the model are 24  and the
            objective function was  0.06

The root mean square of the residuals is  0.01
The df corrected root mean square of the residuals is  0.01
The number of observations was  500  with
                     Chi Square =  27.97  with prob <  0.26

Tucker Lewis Index of factoring reliability =  0.997
RMSEA index =  0.019  and the
            90 % confidence intervals are  0.019 0.023
BIC =  -121.18
Fit based upon off diagonal values = 1
Measures of factor score adequacy
                                                 MR1  MR2  MR4  MR3
Correlation of scores with factors              0.96 0.95 0.95 0.95
Multiple R square of scores with factors        0.92 0.91 0.90 0.90
Minimum correlation of possible factor scores   0.84 0.82 0.80 0.80
```

Show the 4 factor exploratory solution

```
fa.diagram(fa(Data,4),cut=.2)
```



4 factors using CFA

```
Data <- read.table("ex5.6.dat")
names(Data) <- paste("y", 1:12, sep="")
model.ex5.6 <- ' f1 =~ y1 + y2 + y3
f2 =~ y4 + y5 + y6
f3 =~ y7 + y8 + y9
f4 =~ y10 + y11 + y12'
fit <- cfa(model.ex5.6, data=Data, estimator="ML",
          std.ov=TRUE,std.lv=TRUE)
summary(fit, standardized=TRUE, fit.measures=TRUE)

  Number of observations                          500

  Estimator                                        ML
  Minimum Function Chi-square                  45.780
  Degrees of freedom                               48
  P-value                                       0.564
Chi-square test baseline model:
  Minimum Function Chi-square                4012.035
  Degrees of freedom                               66
  P-value                                       0.000
Full model versus baseline model:
  Comparative Fit Index (CFI)                   1.000
  Tucker-Lewis Index (TLI)                      1.001

...
Root Mean Square Error of Approximation:
  RMSEA                                         0.000
  90 Percent Confidence Interval        0.000   0.027
  P-value RMSEA <= 0.05                         1.000
```

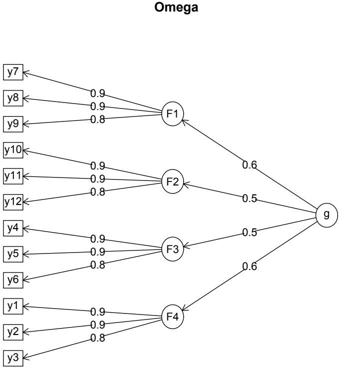
```
                   Estimate  Std.err  Z-value  P(>|z|)
Latent variables:
  f1 =~
    y1                0.892    0.037   24.413    0.000
    y2                0.872    0.037   23.576    0.000
    y3                0.811    0.038   21.238    0.000
  f2 =~
    y4                0.893    0.037   24.421    0.000
    y5                0.857    0.037   22.997    0.000
    y6                0.827    0.038   21.846    0.000
  f3 =~
    y7                0.924    0.035   26.284    0.000
    y8                0.859    0.037   23.425    0.000
    y9                0.857    0.037   23.367    0.000
  f4 =~
    y10               0.899    0.036   24.819    0.000
    y11               0.872    0.037   23.721    0.000
    y12               0.826    0.038   21.896    0.000
Covariances:
  f1 ~~
    f2                0.310    0.045    6.817    0.000
    f3                0.351    0.044    8.050    0.000
    f4                0.292    0.046    6.374    0.000
  f2 ~~
    f3                0.348    0.044    7.952    0.000
    f4                0.226    0.047    4.790    0.000
  f3 ~~
    f4                0.292    0.045    6.456    0.000
Variances:
    y1                0.202    0.024    8.288    0.000
```

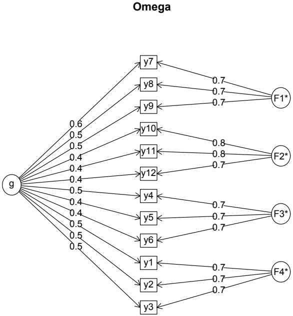
Omega solution

```
om.5_6 <- omega(Data,4,sl=FALSE)

Omega
Call: omega(m = Data, nfactors = 4)
Alpha:                  0.85
G.6:                    0.92
Omega Hierarchical:     0.6
Omega H asymptotic:     0.63
Omega Total             0.95

Schmid Leiman Factor loadings greater than  0.2
       g   F1*   F2*   F3*   F4*   h2   u2   p2
y1  0.53                    0.72 0.79 0.21 0.35
y2  0.50                    0.71 0.77 0.23 0.33
y3  0.45                    0.68 0.67 0.33 0.30
y4  0.48              0.75       0.80 0.20 0.29
y5  0.45              0.74       0.74 0.26 0.27
y6  0.44              0.70       0.68 0.32 0.28
y7  0.57  0.72                   0.86 0.14 0.39
y8  0.53  0.68                   0.74 0.26 0.38
y9  0.54  0.67                   0.74 0.26 0.40
y10 0.42        0.80             0.82 0.18 0.21
y11 0.40        0.78             0.76 0.24 0.21
y12 0.40        0.72             0.68 0.32 0.23

With eigenvalues of:
  g F1* F2* F3* F4*
2.8 1.4 1.8 1.6 1.5
```

```
general/max  1.56   max/min =   1.23
mean percent general =  0.3    with sd =  0.07 and cv of  0.22

The degrees of freedom are 24  and the fit is  0.06
The number of observations was  500  with Chi Square =  27.97
        with prob <  0.26
The root mean square of the residuals is  0.01
The df corrected root mean square of the residuals is  0.01
RMSEA index =  0.019  and the 90 %
           confidence intervals are  0.019 0.023
BIC =  -121.18

Compare this with the adequacy of just a general factor and
            no group factors
The degrees of freedom for just the general factor are 54
                        and the fit is  5.33
The number of observations was  500  with
             Chi Square =  2630.39  with prob <  0
The root mean square of the residuals is  0.16
The df corrected root mean square of the residuals is  0.25

RMSEA index =  0.311  and the 90 % confidence intervals are  0.3
BIC =  2294.8

Measures of factor score adequacy
                                                g  F1* F2*  F3*
Correlation of scores with factors           0.78 0.83 0.9 0.87
Multiple R square of scores with factors     0.61 0.69 0.8 0.75
Minimum correlation of factor score estimates 0.22 0.38 0.6 0.50
```

Two omega solutions





Higher order model with CFA

```
Data <- read.table("ex5.6.dat")
names(Data) <- paste("y", 1:12, sep="")
model.ex5.6 <- ' f1 =~ y1 + y2 + y3
f2 =~ y4 + y5 + y6
f3 =~ y7 + y8 + y9
f4 =~ y10 + y11 + y12
f5 =~ f1 + f2 + f3 + f4'
fit <- cfa(model.ex5.6, data=Data, estimator="ML",
           std.ov=TRUE,std.lv=TRUE)
summary(fit, standardized=TRUE, fit.measures=TRUE)
```

```
 Number of observations                           500
  Estimator                                        ML
  Minimum Function Chi-square                  46.743
  Degrees of freedom                               50
  P-value                                       0.605
Chi-square test baseline model:
  Minimum Function Chi-square                4012.035
  Degrees of freedom                               66
  P-value                                       0.000
Full model versus baseline model:
  Comparative Fit Index (CFI)                   1.000
  Tucker-Lewis Index (TLI)                      1.001

...

Root Mean Square Error of Approximation:
  RMSEA                                         0.000
  90 Percent Confidence Interval          0.000  0.026
  P-value RMSEA <= 0.05                         1.000
```

```
                   Estimate  Std.err  Z-value  P(>|z|)   S
Latent variables:
  f1 =~
    y1                0.727    0.041   17.731    0.000
    y2                0.709    0.041   17.433    0.000
    y3                0.660    0.040   16.439    0.000
  f2 =~
    y4                0.755    0.039   19.212    0.000
    y5                0.725    0.039   18.553    0.000
    y6                0.699    0.039   17.932    0.000
  f3 =~
    y7                0.721    0.044   16.557    0.000
    y8                0.670    0.042   15.845    0.000
    y9                0.669    0.042   15.829    0.000
  f4 =~
    y10               0.795    0.038   21.154    0.000
    y11               0.771    0.038   20.491    0.000
    y12               0.730    0.038   19.283    0.000
  f5 =~
    f1                0.713    0.098    7.270    0.000
    f2                0.632    0.088    7.192    0.000
    f3                0.802    0.111    7.214    0.000
    f4                0.528    0.078    6.785    0.000

Variances:
    y1                0.201    0.024    8.252    0.000
    y2                0.239    0.025    9.570    0.000
    y3                0.341    0.028   12.265    0.000
    y4                0.201    0.024    8.221    0.000
    y5                0.263    0.026   10.298    0.000
    y6                0.314    0.027   11.666    0.000
```

Try a Schmid Leiman model

```
Data <- read.table("ex5.6.dat")
names(Data) <- paste("y", 1:12, sep="")
model.ex5.6 <- ' f1 =~ y1 + y2 + y3
f2 =~ y4 + y5 + y6
f3 =~ y7 + y8 + y9
f4 =~ y10 + y11 + y12
g =~  y1 +  y2 + y3 +  y4 + y5 + y6 + y7 +
                y8 + y9 + y10 + y11 + y12'
fit <- cfa(model.ex5.6, data=Data, estimator="ML",
          std.ov=TRUE,std.lv=TRUE,orthogonal=TRUE)
```

```
Number of observations                           500

  Estimator                                         ML
  Minimum Function Chi-square                   41.640
  Degrees of freedom                                42
  P-value                                        0.487

Chi-square test baseline model:

  Minimum Function Chi-square                 4012.035
  Degrees of freedom                                66
  P-value                                        0.000

Full model versus baseline model:

  Comparative Fit Index (CFI)                    1.000
  Tucker-Lewis Index (TLI)                       1.000

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)              -6522.428
```

```
                   Estimate  Std.err  Z-value  P(>|z|)
Latent variables:
  f1 =~
    y1                0.704    0.045   15.768    0.000
    y2                0.709    0.045   15.887    0.000
    y3                0.694    0.045   15.314    0.000
  f2 =~
    y4                0.752    0.042   18.112    0.000
    y5                0.742    0.042   17.667    0.000
    y6                0.705    0.043   16.453    0.000
  f3 =~
    y7                0.713    0.047   15.248    0.000
    y8                0.679    0.048   14.207    0.000
    y9                0.646    0.049   13.211    0.000
  f4 =~
    y10               0.798    0.039   20.266    0.000
    y11               0.781    0.040   19.546    0.000
    y12               0.718    0.041   17.473    0.000
  g =~
    y1                0.543    0.057    9.490    0.000
    y2                0.508    0.058    8.798    0.000
    y3                0.431    0.059    7.356    0.000
    y4                0.476    0.056    8.448    0.000
    y5                0.435    0.057    7.644    0.000
    y6                0.433    0.057    7.618    0.000
    y7                0.588    0.059   10.039    0.000
    y8                0.528    0.059    8.898    0.000
    y9                0.563    0.059    9.565    0.000
    y10               0.414    0.056    7.374    0.000
    y11               0.392    0.056    6.944    0.000
    y12               0.405    0.056    7.201    0.000
```

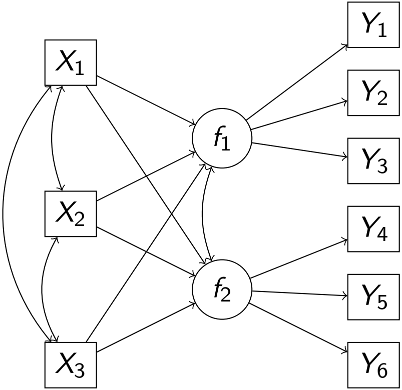
Compare the omega solution with cfa higher order model

```
Schmid Leiman Factor loadings greater than  0.2
       g   F1*   F2*   F3*   F4*   h2   u2   p2
y1  0.53                    0.72 0.79 0.21 0.35
y2  0.50                    0.71 0.77 0.23 0.33
y3  0.45                    0.68 0.67 0.33 0.30
y4  0.48              0.75       0.80 0.20 0.29
y5  0.45              0.74       0.74 0.26 0.27
y6  0.44              0.70       0.68 0.32 0.28
y7  0.57  0.72                   0.86 0.14 0.39
y8  0.53  0.68                   0.74 0.26 0.38
y9  0.54  0.67                   0.74 0.26 0.40
y10 0.42        0.80             0.82 0.18 0.21
y11 0.40        0.78             0.76 0.24 0.21
y12 0.40        0.72             0.68 0.32 0.23

With eigenvalues of:
  g F1* F2* F3* F4*
2.8 1.4 1.8 1.6 1.5
```

```
Latent variables:
  f1 =~
    y1                0.704    0.045   15.768    0.000
    y2                0.709    0.045   15.887    0.000
    y3                0.694    0.045   15.314    0.000
  f2 =~
    y4                0.752    0.042   18.112    0.000
    y5                0.742    0.042   17.667    0.000
    y6                0.705    0.043   16.453    0.000
  f3 =~
    y7                0.713    0.047   15.248    0.000
    y8                0.679    0.048   14.207    0.000
    y9                0.646    0.049   13.211    0.000
  f4 =~
    y10               0.798    0.039   20.266    0.000
    y11               0.781    0.040   19.546    0.000
    y12               0.718    0.041   17.473    0.000
  g =~
    y1                0.543    0.057    9.490    0.000
    y2                0.508    0.058    8.798    0.000
    y3                0.431    0.059    7.356    0.000
    y4                0.476    0.056    8.448    0.000
    y5                0.435    0.057    7.644    0.000
    y6                0.433    0.057    7.618    0.000
    y7                0.588    0.059   10.039    0.000
    y8                0.528    0.059    8.898    0.000
    y9                0.563    0.059    9.565    0.000
    y10               0.414    0.056    7.374    0.000
    y11               0.392    0.056    6.944    0.000
    y12               0.405    0.056    7.201    0.000
```

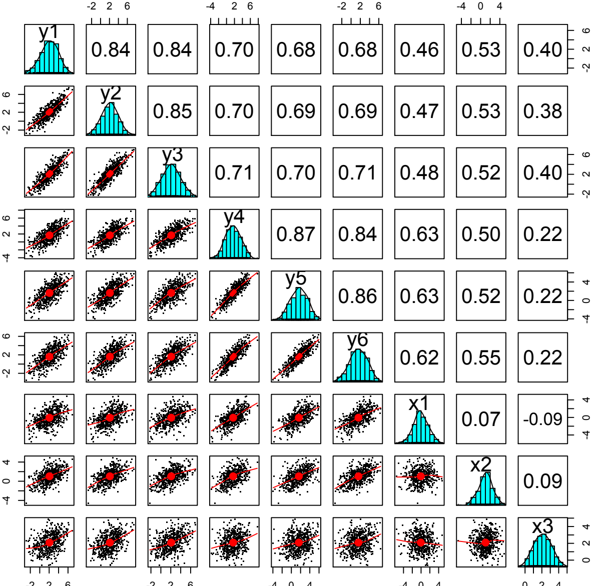
A MIMIC model

Consider the case of 6 variables with two latent factors that have 3 covariates (MPlus Example 5.8)




```
fn <- "http://www.statmodel.com/usersguid
Data <- read.table(fn)
names(Data) <- c(paste("y", 1:6, sep=""),
          paste("x", 1:3, sep=""))
 describe(Data)
 pairs.panels(Data,pch=".")
model.ex5.8 <- ' f1 =~ y1 + y2 + y3
f2 =~ y4 + y5 + y6
f1 + f2 ~ x1 + x2 + x3 '
fit <- cfa(model.ex5.8, data=Data,
           estimator="ML",std.lv=TRUE)
summary(fit, standardized=TRUE,
                 fit.measures=TRUE)
```

The MIMIC correlations 2 - factors



A MIMIC model

```
lavaan (0.5-22) converged normally after  30 iterations

  Number of observations                           500

  Estimator                                         ML
  Minimum Function Test Statistic               20.023
  Degrees of freedom                                20
  P-value (Chi-square)                           0.457

Model test baseline model:

  Minimum Function Test Statistic             4176.601
  Degrees of freedom                                33
  P-value                                        0.000

User model versus baseline model:

  Comparative Fit Index (CFI)                    1.000
  Tucker-Lewis Index (TLI)                       1.000

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)              -6585.177
  Loglikelihood unrestricted model (H1)      -6575.166

  Number of free parameters                         19
  Akaike (AIC)                               13208.355
  Bayesian (BIC)                             13288.432
  Sample-size adjusted Bayesian (BIC)        13228.125

Root Mean Square Error of Approximation:
```

```
                   Estimate  Std.err  Z-value  P(>|z|)
Latent variables:
  f1 =~
    y1                0.849    0.036   23.407    0.000
    y2                0.876    0.037   23.627    0.000
    y3                0.867    0.037   23.733    0.000
  f2 =~
    y4                0.802    0.035   23.067    0.000
    y5                0.780    0.033   23.504    0.000
    y6                0.783    0.034   23.046    0.000

Regressions:
  f1 ~
    x1                0.585    0.037   15.786    0.000
    x2                0.667    0.043   15.414    0.000
    x3                0.814    0.058   13.950    0.000
  f2 ~
    x1                0.846    0.045   18.783    0.000
    x2                0.750    0.046   16.237    0.000
    x3                0.539    0.054    9.975    0.000

Covariances:
  f1 ~~
    f2                0.336    0.051    6.621    0.000

Variances:
    y1                0.529    0.046   11.505    0.000
    y2                0.512    0.046   11.056    0.000
```

Consider the two factor measurement model

```
f2 <- fa(Data[1:6],2)
```

```
Factor Analysis using method =  minres
Call: fa(r = Data[1:6], nfactors = 2)
Standardized loadings based
        upon correlation matrix
     MR1   MR2   h2   u2
y1 -0.01  0.93 0.84 0.16
y2  0.00  0.93 0.85 0.15
y3  0.03  0.90 0.85 0.15
y4  0.88  0.05 0.85 0.15
y5  0.99 -0.05 0.90 0.10
y6  0.89  0.03 0.84 0.16

                MR1  MR2
SS loadings    2.56 2.56
Proportion Var 0.43 0.43
Cumulative Var 0.43 0.85

 With factor correlations of
     MR1  MR2
MR1 1.00 0.81
MR2 0.81 1.00
```

```
Test of the hypothesis that 2 factors are sufficient.

The degrees of freedom for the null model are  15  and the
       objective function  was  6.59 with Chi Square of  3270
The degrees of freedom for the model are 4  and the
                        objective function was  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  500  with
                              Chi Square =  3.33  with prob <

Tucker Lewis Index of factoring reliability =  1.001
RMSEA index =  0  and the 90 %
                   confidence intervals are  0 0.024
BIC =  -21.53
Fit based upon off diagonal values = 1
Measures of factor score adequacy
                                                 MR1  MR2
Correlation of scores with factors              0.98 0.97
Multiple R square of scores with factors        0.95 0.95
Minimum correlation of possible factor scores  0.91 0.90
```

Note the drastic difference in correlations from the two factor model versus the MIMIC model.

Partial x1 ... x3 from y1 ... y6 and then factor – closer to the MIMIC solution

```
ex5.8 <- Data
 lowerCor(ex5.8[1:6])
 yp <- partial.r(ex5.8,1:6,7:9)
 lowerMat(yp)

#the original
   y1   y2   y3   y4   y5   y6
y1 1.00
y2 0.84 1.00
y3 0.84 0.85 1.00
y4 0.70 0.70 0.71 1.00
y5 0.68 0.69 0.70 0.87 1.00
y6 0.68 0.69 0.71 0.84 0.86 1.00


#the partialed
   y1   y2   y3   y4   y5   y6
y1 1.00
y2 0.59 1.00
y3 0.59 0.61 1.00
y4 0.24 0.23 0.24 1.00
y5 0.16 0.19 0.20 0.60 1.00
y6 0.14 0.16 0.21 0.51 0.57 1.00
```

```
 yp <-unclass(yp)
fa(yp,2)

Factor Analysis using method =  minres
Call: fa(r = yp, nfactors = 2)
Standardized loadings (pattern matrix) based
                upon correlation matrix
     MR1   MR2   h2   u2
y1  0.76 -0.02 0.58 0.42
y2  0.78 -0.01 0.61 0.39
y3  0.77  0.03 0.60 0.40
y4  0.07  0.71 0.54 0.46
y5 -0.04  0.83 0.67 0.33
y6 -0.01  0.70 0.48 0.52

                       MR1  MR2
SS loadings           1.80 1.69
Proportion Var        0.30 0.28
Cumulative Var        0.30 0.58
Proportion Explained  0.52 0.48
Cumulative Proportion 0.52 1.00

 With factor correlations of
     MR1  MR2
MR1 1.00 0.32
MR2 0.32 1.00
```

Compare this to the MIMIC solution and the 2 factor measurement model without covariates.

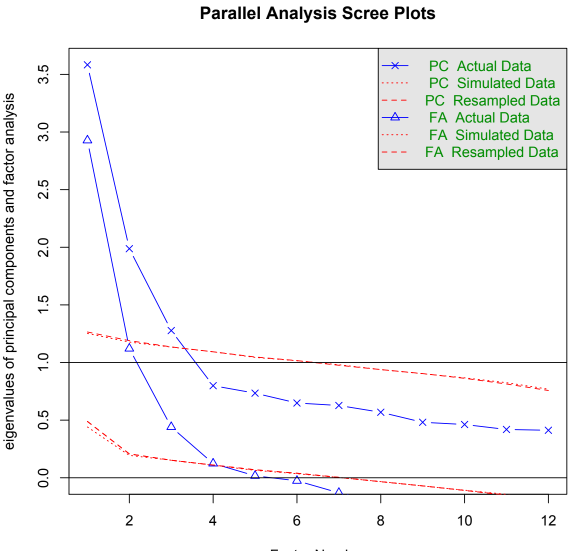
Consider the following 12 variables

```
> Data <- read.table("ex5.11.dat")
> names(Data) <- paste("y", 1:12, sep="")

> round(cor(Data),2)
```

```
       y1    y2    y3    y4    y5    y6   y7   y8   y9  y10  y11  y12
y1   1.00  0.52  0.41 -0.02  0.00 -0.02 0.27 0.24 0.26 0.11 0.14 0.13
y2   0.52  1.00  0.52  0.01 -0.03  0.00 0.23 0.18 0.28 0.14 0.15 0.18
y3   0.41  0.52  1.00  0.02 -0.11 -0.01 0.16 0.17 0.20 0.09 0.08 0.12
y4  -0.02  0.01  0.02  1.00  0.44  0.39 0.34 0.34 0.30 0.16 0.12 0.08
y5   0.00 -0.03 -0.11  0.44  1.00  0.42 0.28 0.27 0.25 0.15 0.09 0.08
y6  -0.02  0.00 -0.01  0.39  0.42  1.00 0.30 0.31 0.34 0.11 0.13 0.12
y7   0.27  0.23  0.16  0.34  0.28  0.30 1.00 0.56 0.55 0.32 0.27 0.27
y8   0.24  0.18  0.17  0.34  0.27  0.31 0.56 1.00 0.53 0.28 0.25 0.23
y9   0.26  0.28  0.20  0.30  0.25  0.34 0.55 0.53 1.00 0.29 0.24 0.29
y10  0.11  0.14  0.09  0.16  0.15  0.11 0.32 0.28 0.29 1.00 0.42 0.31
y11  0.14  0.15  0.08  0.12  0.09  0.13 0.27 0.25 0.24 0.42 1.00 0.33
y12  0.13  0.18  0.12  0.08  0.08  0.12 0.27 0.23 0.29 0.31 0.33 1.00
```

12 variables, how many factors?

fa.parallel(Data)



Try 3 factors

f3 <- fa(Data,3)

```
> f3 <- fa(Data,3)
> f3
Factor Analysis using method =  minres
Call: fa(r = Data, nfactors = 3)
Standardized loadings based upon correlation matrixTest of the hypothesis that 3 factors are sufficient.
      MR1   MR2   MR3   h2   u2
y1    0.03  0.68 -0.02 0.46 0.54                The degrees of freedom for the null model are  66  and the
y2   -0.02  0.76  0.00 0.57 0.43                          objective function was  3.05 with Chi Square of  15
y3   -0.03  0.67 -0.06 0.42 0.58                The degrees of freedom for the model are 33  and the
y4    0.65 -0.09 -0.07 0.38 0.62                                 objective function was  0.17
y5    0.62 -0.15 -0.08 0.35 0.65
y6    0.63 -0.10 -0.07 0.36 0.64                The root mean square of the residuals is  0.02
y7    0.54  0.18  0.21 0.53 0.47                The df corrected root mean square of the residuals is  0.04
y8    0.55  0.17  0.16 0.48 0.52                The number of observations was  500  with
y9    0.53  0.24  0.16 0.51 0.49                                             Chi Square =  81.42  with prob
y10   0.03 -0.06  0.63 0.39 0.61
y11  -0.04 -0.05  0.66 0.39 0.61                Tucker Lewis Index of factoring reliability =  0.932
y12   0.02  0.04  0.49 0.26 0.74                RMSEA index =  0.055  and the 90 % confidence intervals are
                                                BIC =  -123.66
                MR1  MR2  MR3                   Fit based upon off diagonal values = 0.99
SS loadings    2.17 1.68 1.27                   Measures of factor score adequacy
Proportion Var 0.18 0.14 0.11                                                                MR1  MR2  MR3
Cumulative Var 0.18 0.32 0.43                   Correlation of scores with factors           0.89 0.88 0.83
                                                Multiple R square of scores with factors     0.80 0.77 0.70
 With factor correlations of                    Minimum correlation of possible factor scores  0.59 0.55 0.39
     MR1  MR2  MR3
MR1 1.00 0.17 0.43
MR2 0.17 1.00 0.37
MR3 0.43 0.37 1.00
```

Try 4 factor solution

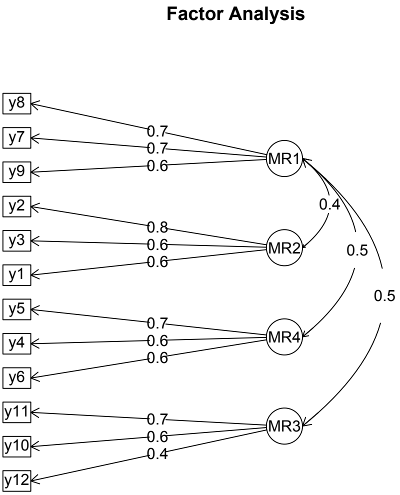
```
> f4 <- fa(Data,4)
> f4
```

```
Factor Analysis using method =  minres
Call: fa(r = Data, nfactors = 4)
Standardized loadings based upon correlation matrix
      MR1   MR2   MR4   MR3   h2   u2
y1    0.21  0.56 -0.11 -0.04 0.43 0.57
y2   -0.07  0.85  0.05  0.04 0.69 0.31
y3    0.07  0.61 -0.07 -0.04 0.40 0.60
y4    0.12  0.00  0.58 -0.02 0.42 0.58
y5   -0.05  0.01  0.71  0.01 0.47 0.53
y6    0.11 -0.01  0.55 -0.01 0.38 0.62
y7    0.72  0.00  0.03  0.06 0.59 0.41
y8    0.74 -0.04  0.02  0.00 0.55 0.45
y9    0.62  0.10  0.08  0.04 0.53 0.47
y10   0.06 -0.02  0.02  0.60 0.40 0.60
y11  -0.03  0.00 -0.01  0.68 0.44 0.56
y12   0.11  0.05 -0.04  0.44 0.26 0.74

                MR1  MR2  MR4  MR3
SS loadings    1.75 1.48 1.25 1.08
Proportion Var 0.15 0.12 0.10 0.09
Cumulative Var 0.15 0.27 0.37 0.46

 With factor correlations of
     MR1   MR2   MR4  MR3
MR1 1.00  0.38  0.54 0.52
MR2 0.38  1.00 -0.07 0.26
MR4 0.54 -0.07  1.00 0.26
MR3 0.52  0.26  0.26 1.00
```

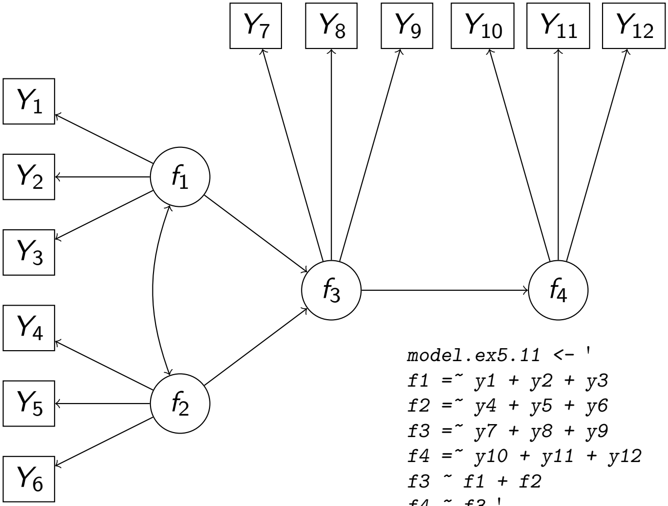
```
Test of the hypothesis that 4 factors are sufficient.

The degrees of freedom for the null model are  66  and the
                 objective function was  3.05 with Chi Square
The degrees of freedom for the model are 24  and
                       the objective function was  0.06

The root mean square of the residuals is  0.01
The df corrected root mean square of the residuals is  0.03
The number of observations was  500  with
                   Chi Square =  27.21  with prob <  0.29

Tucker Lewis Index of factoring reliability =  0.994
RMSEA index =  0.017  and the 90 %
                     confidence intervals are  0.017 0.021
BIC =  -121.94
Fit based upon off diagonal values = 1
Measures of factor score adequacy
                                                 MR1  MR2  MR4
Correlation of scores with factors              0.90 0.89 0.85
Multiple R square of scores with factors        0.81 0.79 0.72
Minimum correlation of possible factor scores   0.63 0.58 0.44
```

4 latent variables – somewhat correlated

diagram(f4,cut=.3)



Four latent variables with a particular structure



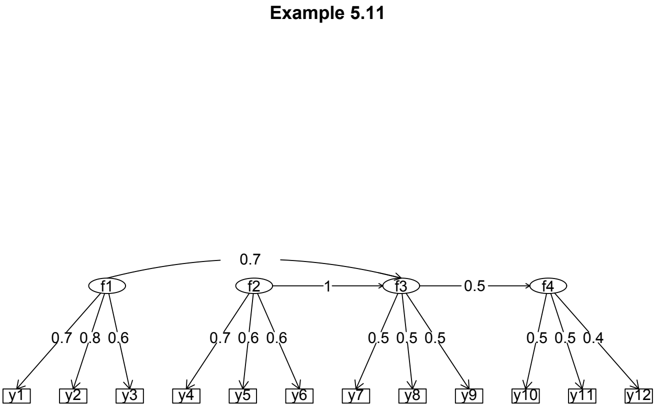

```
model.ex5.11 <- '
f1 =~ y1 + y2 + y3
f2 =~ y4 + y5 + y6
f3 =~ y7 + y8 + y9
f4 =~ y10 + y11 + y12
f3 ~ f1 + f2
f4 ~ f3 '
```

Fit the sem model

```
fit <- sem(model.ex5.11, data=Data, estimator="ML",
                                     std.ov=TRUE,std.lv=TRUE)
summary(fit, standardized=TRUE, fit.measures=TRUE)

Number of observations                           500

  Estimator                                       ML
  Minimum Function Chi-square                 53.704
  Degrees of freedom                              50
  P-value                                      0.334

Chi-square test baseline model:

  Minimum Function Chi-square               1524.403
  Degrees of freedom                              66
  P-value                                      0.000

Full model versus baseline model:

  Comparative Fit Index (CFI)                  0.997
  Tucker-Lewis Index (TLI)                     0.997

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)            -7772.275
  Loglikelihood unrestricted model (H1)    -7745.424

  Number of free parameters                       28
  Akaike (AIC)                             15600.551
  Bayesian (BIC)                           15718.560
  Sample-size adjusted Bayesian (BIC)      15629.686

Root Mean Square Error of Approximation:
```

```
                   Estimate  Std.err  Z-value  P(>|z|)
Latent variables:
  f1 =~
    y1                0.678    0.046   14.621    0.000
    y2                0.779    0.046   16.826    0.000
    y3                0.637    0.047   13.680    0.000
  f2 =~
    y4                0.659    0.048   13.706    0.000
    y5                0.643    0.048   13.349    0.000
    y6                0.632    0.048   13.107    0.000
  f3 =~
    y7                0.487    0.040   12.109    0.000
    y8                0.459    0.039   11.769    0.000
    y9                0.468    0.039   11.884    0.000
  f4 =~
    y10               0.518    0.048   10.806    0.000
    y11               0.502    0.047   10.636    0.000
    y12               0.418    0.045    9.298    0.000

Regressions:
  f3 ~
    f1                0.713    0.102    6.962    0.000
    f2                1.004    0.127    7.906    0.000
  f4 ~
    f3                0.471    0.066    7.128    0.000

Covariances:
  f1 ~~
    f2               -0.034    0.062   -0.546    0.585
```

A SEM diagram using lavaan.diagram (from psych)

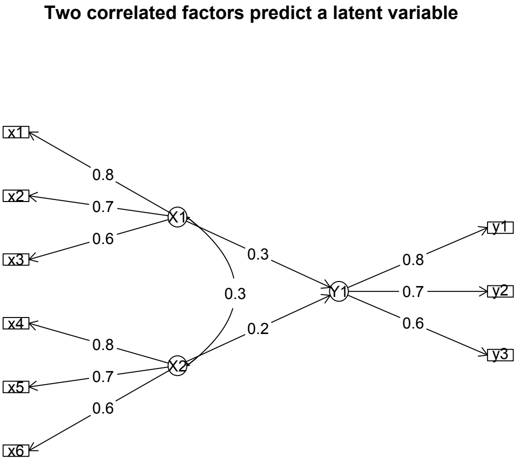


Constructing a model with two correlated factors predicting a latent variable

```
fx <- matrix(c(.8,.7,.6,rep(0,6),.8,.7,.6),ncol=2)
 fy <- matrix(c(.8,.7,.6))
Phi <-matrix(c(1,.3,.3,.3,1,.2,0,0,1),3,3)
set.seed(42)
sim2f <- sim.structure(fx,Phi,fy,n=500)

> fx
     [,1] [,2]
[1,]  0.8  0.0
[2,]  0.7  0.0
[3,]  0.6  0.0
[4,]  0.0  0.8
[5,]  0.0  0.7
[6,]  0.0  0.6
> fy
     [,1]
[1,]  0.8
[2,]  0.7
[3,]  0.6
> Phi
     [,1] [,2] [,3]
[1,]  1.0  0.3    0
[2,]  0.3  1.0    0
[3,]  0.3  0.2    1
```

```
 lowerMat(sim2f$model)
 simdata <- data.frame(sim2f$observed)
lowerCor(simdata)

#model
    Vx1  Vx2  Vx3  Vx4  Vx5  Vx6  Vy1  Vy2  Vy3
Vx1 1.00
Vx2 0.56 1.00
Vx3 0.48 0.42 1.00
Vx4 0.19 0.17 0.14 1.00
Vx5 0.17 0.15 0.13 0.56 1.00
Vx6 0.14 0.13 0.11 0.48 0.42 1.00
Vy1 0.19 0.17 0.14 0.13 0.11 0.10 1.00
Vy2 0.17 0.15 0.13 0.11 0.10 0.08 0.56 1.00
Vy3 0.14 0.13 0.11 0.10 0.08 0.07 0.48 0.42 1.00

#data
 lowerCor(simdata)
    Vx1   Vx2   Vx3   Vx4   Vx5   Vx6   Vy1   Vy2   Vy
Vx1  1.00
Vx2  0.50  1.00
Vx3  0.45  0.35  1.00
Vx4 -0.19 -0.02 -0.09  1.00
Vx5  0.00 -0.29 -0.02  0.43  1.00
Vx6  0.01 -0.01 -0.34  0.41  0.29  1.00
Vy1  0.32  0.19  0.10  0.48  0.28  0.23  1.00
Vy2  0.26  0.23  0.16  0.21  0.51  0.18  0.59  1.00
Vy3  0.17  0.10  0.19  0.21  0.22  0.50  0.49  0.45  1
```

The structural model



An EFA model – doesn't quite get it

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

Factor Analysis using method =  minres
Call: fa(r = simdata, nfactors = 3)
Standardized loadings (pattern matrix) based upon correlation matrix
      MR2   MR3   MR1   h2    u2 com
Vx1  0.14  0.65  0.02 0.47 0.527 1.1
Vx2 -0.05  0.76  0.06 0.56 0.435 1.0
Vx3  0.21  0.45 -0.36 0.40 0.605 2.4
Vx4  0.39 -0.20  0.28 0.32 0.681 2.4
Vx5  0.69 -0.39  0.06 0.58 0.422 1.6
Vx6  0.00  0.02  1.00 1.00 0.005 1.0
Vy1  0.65  0.20  0.06 0.52 0.479 1.2
Vy2  0.80  0.14 -0.04 0.67 0.327 1.1
Vy3  0.37  0.23  0.41 0.46 0.543 2.6

                       MR2  MR3  MR1
SS loadings           1.98 1.54 1.46
Proportion Var        0.22 0.17 0.16
Cumulative Var        0.22 0.39 0.55
Proportion Explained  0.40 0.31 0.29
Cumulative Proportion 0.40 0.71 1.00

 With factor correlations of
     MR2   MR3   MR1
MR2 1.00  0.15  0.29
MR3 0.15  1.00 -0.09
MR1 0.29 -0.09  1.00
```

Compare two lavaan models with either one or two predictors

```
model <- 'f1 =~ Vx1 + Vx2 + Vx3
        f2 =~ Vx4 + Vx5 + Vx6
         Yf =~ Vy1 + Vy2 + Vy3
         Yf ~ f1 + f2'
fit <- sem(model,data=simdata,std.lv=TRUE,orthogonal=TRUE)
 summary(fit)

  Number of observations                           500

  Estimator                                         ML
  Minimum Function Test Statistic              768.689
  Degrees of freedom                                25
  P-value (Chi-square)                           0.000

Parameter Estimates:

  Information                                 Expected
  Standard Errors                             Standard

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  f1 =~
    Vx1               0.644    0.038   16.931    0.000
    Vx2               0.492    0.037   13.131    0.000
    Vx3               0.456    0.039   11.650    0.000
  f2 =~
    Vx4               0.540    0.036   15.160    0.000
    Vx5               0.499    0.038   13.048    0.000
    Vx6               0.432    0.039   11.014    0.000
  Yf =~
    Vy1               0.514    0.070    7.327    0.000
    Vy2               0.459    0.062    7.352    0.000
    Vy3               0.381    0.054    7.053    0.000
```

```
model1 <- 'f1 =~ Vx1 + Vx2 + Vx3
        f2 =~ Vx4 + Vx5 + Vx6
         Yf =~ Vy1 + Vy2 + Vy3
         Yf ~  f2'
fit1 <- sem(model1,data=simdata,std.lv=TRUE,orthogonal
 summary(fit1)

  Number of observations                           500

  Estimator                                         ML
  Minimum Function Test Statistic              886.555
  Degrees of freedom                                26
  P-value (Chi-square)                           0.000

Parameter Estimates:

  Information                                 Expected
  Standard Errors                             Standard

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  f1 =~
    Vx1               0.639    0.044   14.372    0.000
    Vx2               0.493    0.041   12.097    0.000
    Vx3               0.460    0.041   11.143    0.000
  f2 =~
    Vx4               0.538    0.038   14.340    0.000
    Vx5               0.495    0.040   12.434    0.000
    Vx6               0.440    0.040   10.882    0.000
  Yf =~
    Vy1               0.786    0.058   13.500    0.000
    Vy2               0.701    0.054   12.965    0.000
    Vy3               0.591    0.051   11.476    0.000
```

Allow the factors to be correlated – fits are better

```
model <- 'f1 =~ Vx1 + Vx2 + Vx3
        f2 =~ Vx4 + Vx5 + Vx6
         Yf =~ Vy1 + Vy2 + Vy3
         Yf ~ f1 + f2'
 fit <- sem(model,data=simdata,std.lv=TRUE)
summary(fit)

 Number of observations                            500
  Estimator                                         ML
  Minimum Function Chi-square                   26.474
  Degrees of freedom                                24
  P-value                                        0.330
                   Estimate  Std.err  Z-value  P(>|z|)
Latent variables:
  f1 =~
    Vx1               0.694    0.046   15.077    0.000
    Vx2               0.661    0.048   13.703    0.000
    Vx3               0.591    0.049   12.031    0.000
  f2 =~
    Vx4               0.867    0.050   17.341    0.000
    Vx5               0.664    0.049   13.482    0.000
    Vx6               0.594    0.047   12.638    0.000
  Yf =~
    Vy1               0.755    0.048   15.653    0.000
    Vy2               0.649    0.047   13.939    0.000
    Vy3               0.584    0.047   12.421    0.000
Regressions:
  Yf ~
    f1                0.309    0.069    4.480    0.000
    f2                0.051    0.063    0.817    0.414
Covariances:
  f1 ~~
    f2                0.287    0.055    5.186    0.000
```

```
model1 <- 'f1 =~ Vx1 + Vx2 + Vx3
        f2 =~ Vx4 + Vx5 + Vx6
         Yf =~ Vy1 + Vy2 + Vy3
         Yf ~  f2'
fit1 <- sem(model1,data=simdata,std.lv=TRUE)
 summary(fit1)

Number of observations                             500
 Estimator                                          ML
 Minimum Function Chi-square                    48.118
 Degrees of freedom                                 25
 P-value                                         0.004
                   Estimate  Std.err  Z-value  P(>|z|)
Latent variables:
 f1 =~
   Vx1                0.692    0.047   14.735    0.000
   Vx2                0.659    0.049   13.480    0.000
   Vx3                0.596    0.050   12.033    0.000
 f2 =~
   Vx4                0.859    0.049   17.353    0.000
   Vx5                0.670    0.049   13.640    0.000
   Vx6                0.596    0.047   12.717    0.000
 Yf =~
   Vy1                0.786    0.050   15.771    0.000
   Vy2                0.672    0.048   13.872    0.000
   Vy3                0.610    0.049   12.528    0.000
Regressions:
 Yf ~
   f2                 0.153    0.059    2.611    0.009
Covariances:
 f1 ~~
   f2                 0.298    0.055    5.372    0.000
```

Multiple group models can improve power

- Can improve power by "borrowing strength" from other samples
- Consider the case of figural invariance across groups
  - Items have same loadings on latent variables
  - Latent variable correlations differ across groups
  - Borrow strength in fixing figural invariance, and then study the difference in latent correlations

Consider the MIMIC model of Example 5.8, but with a grouping variable

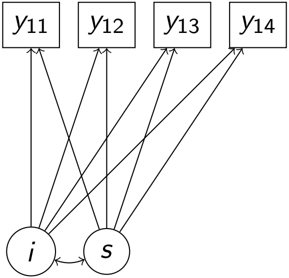
```
Data <- read.table("ex5.14.dat")
names(Data) <- c("y1","y2","y3","y4","y5","y6",
                 "x1","x2","x3", "g")
model.ex5.14 <- ' f1 =~ y1 + y2 + y3
f2 =~ y4 + y5 + y6
f1 + f2 ~ x1 + x2 + x3 '
fit <- cfa(model.ex5.14, data=Data, group="g",
  meanstructure=FALSE,group.equal=c("loadings"),
        group.partial=c("f1=~y3"),std.ov=TRUE)
summary(fit, standardized=TRUE, fit.measures=TRUE)

Lavaan (0.4-7) converged normally after 47 iterations

  Number of observations per group
  1                                               500
  2                                               600
  Estimator                                        ML
  Minimum Function Chi-square                  39.132
  Degrees of freedom                               45
  P-value                                       0.718
Chi-square for each group:
  1                                            20.522
  2                                            18.610
Chi-square test baseline model:
  Minimum Function Chi-square                8731.537
  Degrees of freedom                               72
  P-value                                       0.000
Full model versus baseline model:
  Comparative Fit Index (CFI)                   1.000
```

```
Group 1 [1]:

                   Estimate  Std.err  Z-value  P(>|z|)
Latent variables:
  f1 =~
    y1                0.480    0.014   33.915    0.000
    y2                0.481    0.014   34.007    0.000
    y3                0.483    0.016   29.612    0.000
  f2 =~
    y4                0.423    0.012   34.369    0.000
    y5                0.431    0.012   35.073    0.000
    y6                0.429    0.012   34.882    0.000

Regressions:
  f1 ~
    x1                0.988    0.057   17.388    0.000
    x2                0.951    0.056   16.886    0.000
    x3                0.821    0.055   14.979    0.000
  f2 ~
    x1                1.438    0.064   22.624    0.000
    x2                1.076    0.058   18.504    0.000
    x3                0.547    0.053   10.398    0.000

Covariances:
  f1 ~~
    f2                0.339    0.050    6.787    0.000

Variances:
    y1                0.163    0.014   11.538    0.000
```

Growth models

- If traits change over time, one can try to model the growth in the means structure
    - Intercepts are set to be equal
    - Slope parameter models changes in the means
    - Variables can be observed or latent
- Can also model more complicated growth functions
    - Quadratic growth
    - Piecewise growth

Consider the following growth model




```
Data <- read.table("ex6.1.dat")
names(Data) <- c("y11","y12","y13","y14")

> round(cor(Data),2)
      y11  y12  y13  y14
y11 1.00 0.67 0.59 0.56
y12 0.67 1.00 0.78 0.74
y13 0.59 0.78 1.00 0.85
y14 0.56 0.74 0.85 1.00
> describe(Data)
    var   n mean   sd median trimmed  mad   min  max rang
y11   1 500 0.51 1.20   0.56    0.54 1.27 -2.69 3.60  6.2
y12   2 500 1.57 1.41   1.56    1.58 1.28 -3.06 5.96  9.0
y13   3 500 2.57 1.71   2.52    2.58 1.82 -2.75 8.43 11.1
y14   4 500 3.60 2.08   3.61    3.62 2.10 -2.36 9.18 11.5
```

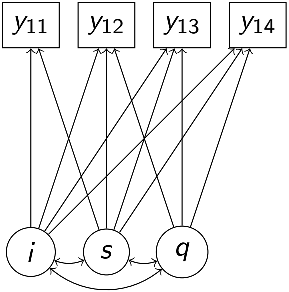
Growth model for example 6.1

```
model.ex6.1 <- ' i =~ 1*y11 + 1*y12 + 1*y13 + 1*y14
s =~ 0*y11 + 1*y12 + 2*y13 + 3*y14 '
fit <- growth(model.ex6.1, data=Data)
summary(fit, standardized=TRUE, fit.measures=TRUE)
```

```
Lavaan (0.4-7) converged normally after 27 iterations

  Number of observations                           500

  Estimator                                         ML
  Minimum Function Chi-square                    4.593
  Degrees of freedom                                 5
  P-value                                        0.468

Chi-square test baseline model:

  Minimum Function Chi-square                 1439.722
  Degrees of freedom                                 6
  P-value                                        0.000

Full model versus baseline model:

  Comparative Fit Index (CFI)                    1.000
  Tucker-Lewis Index (TLI)                       1.000

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)              -3016.386
  Loglikelihood unrestricted model (H1)      -3014.089

  Number of free parameters                          9
  Akaike (AIC)                                6050.772
  Bayesian (BIC)                              6088.703
```

```
                   Estimate  Std.err  Z-value  P(>|z|)
Latent variables:
  i =~
    y11               1.000
    y12               1.000
    y13               1.000
    y14               1.000
  s =~
    y11               0.000
    y12               1.000
    y13               2.000
    y14               3.000

Covariances:
  i ~~
    s                 0.133    0.033    4.057    0.000

Intercepts:
    y11               0.000
    y12               0.000
    y13               0.000
    y14               0.000
    i                 0.523    0.051   10.153    0.000
    s                 1.026    0.025   40.268    0.000

Variances:
    y11               0.475    0.059    7.989    0.000
    y12               0.482    0.040   11.994    0.000
    y13               0.473    0.047   10.007    0.000
```

Consider the following growth model



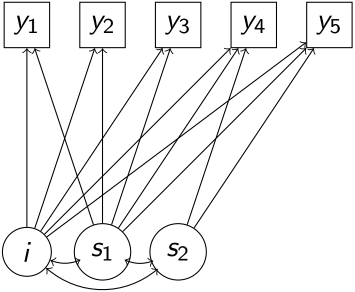

```
Data <- read.table("ex6.9.dat")
names(Data) <- c("y11","y12","y13","y14")

> round(cor(Data),2)
     y11  y12  y13  y14
y11 1.00 0.50 0.33 0.24
y12 0.50 1.00 0.68 0.58
y13 0.33 0.68 1.00 0.89
y14 0.24 0.58 0.89 1.00
> describe(Data)
    var   n mean   sd median trimmed  mad   min   max ran
y11   1 500 0.52 1.39   0.57    0.55 1.46 -3.19  4.08  7.
y12   2 500 2.09 1.71   2.08    2.10 1.59 -4.25  7.61 11.
y13   3 500 4.62 2.82   4.38    4.64 3.04 -3.56 14.28 17.
y14   4 500 8.24 4.99   8.19    8.29 5.20 -6.18 22.45 28.
```

Estimating quadratic and linear growth

```
model.ex6.9 <- '
i =~ 1*y11 + 1*y12 + 1*y13 + 1*y14
s =~ 0*y11 + 1*y12 + 2*y13 + 3*y14
q =~ 0*y11 + 1*y12 + 4*y13 + 9*y14
'
fit <- growth(model.ex6.9, data=Data)
summary(fit, standardized=TRUE, fit.measures=TRUE)

Lavaan (0.4-7) converged normally after 61 iterations

  Number of observations                           500

  Estimator                                         ML
  Minimum Function Chi-square                    0.475
  Degrees of freedom                                 1
  P-value                                        0.490

Chi-square test baseline model:

  Minimum Function Chi-square                 1231.275
  Degrees of freedom                                 6
  P-value                                        0.000

Full model versus baseline model:

  Comparative Fit Index (CFI)                    1.000
  Tucker-Lewis Index (TLI)                       1.003

Loglikelihood and Information Criteria:
```

```
                   Estimate  Std.err  Z-value  P(>|z|)
Latent variables:
  i =~
    y11               1.000
    y12               1.000
    y13               1.000
    y14               1.000
  s =~
    y11               0.000
    y12               1.000
    y13               2.000
    y14               3.000
  q =~
    y11               0.000
    y12               1.000
    y13               4.000
    y14               9.000

Covariances:
  i ~~
    s                -0.173    0.300   -0.578    0.563
    q                 0.101    0.080    1.269    0.204
  s ~~
    q                -0.179    0.117   -1.525    0.127

Intercepts:
    y11               0.000
    y12               0.000
    y13               0.000
    y14               0.000
```

Piecewise growth




```
> Data <- read.table("ex6.11.dat")
> names(Data) <- c("y1","y2","y3","y4","y5")
> round(cor(Data),2)
> describe(Data)

     y1   y2   y3   y4   y5
y1 1.00 0.63 0.51 0.53 0.49
y2 0.63 1.00 0.70 0.68 0.62
y3 0.51 0.70 1.00 0.75 0.66
y4 0.53 0.68 0.75 1.00 0.79
y5 0.49 0.62 0.66 0.79 1.00

   var   n mean   sd median trimmed  mad   min   max ra
y1   1 500 0.44 1.17   0.37    0.44 1.17 -2.64  3.61
y2   2 500 1.58 1.32   1.52    1.58 1.29 -2.28  5.76
y3   3 500 2.59 1.52   2.58    2.60 1.56 -1.46  7.15
y4   4 500 4.54 1.58   4.46    4.54 1.60 -0.25  9.11
y5   5 500 6.54 1.81   6.63    6.54 1.79  1.54 12.29
```

Modeling piecewise growth

```
modelex6.11 <- '
i =~ 1*y1 + 1*y2 + 1*y3 + 1*y4 + 1*y5
s1 =~ 0*y1 + 1*y2 + 2*y3 + 2*y4 + 2*y5
s2 =~ 0*y1 + 0*y2 + 0*y3 + 1*y4 + 2*y5
'
fit <- growth(modelex6.11, data=Data)
summary(fit, standardized=TRUE, fit.measures=TRUE)
```

```
  Number of observations                           500

  Estimator                                         ML
  Minimum Function Chi-square                    5.244
  Degrees of freedom                                 6
  P-value                                        0.513

Chi-square test baseline model:

  Minimum Function Chi-square                 1587.697
  Degrees of freedom                                10
  P-value                                        0.000

Full model versus baseline model:

  Comparative Fit Index (CFI)                    1.000
  Tucker-Lewis Index (TLI)                       1.001

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)              -3706.171
  Loglikelihood unrestricted model (H1)      -3703.549

  Number of free parameters                         14
```

```
                   Estimate  Std.err  Z-value  P(>|z|)
Latent variables:
  i =~
    y1                1.000
    y2                1.000
    y3                1.000
    y4                1.000
    y5                1.000
  s1 =~
    y1                0.000
    y2                1.000
    y3                2.000
    y4                2.000
    y5                2.000
  s2 =~
    y1                0.000
    y2                0.000
    y3                0.000
    y4                1.000
    y5                2.000

Covariances:
  i ~~
    s1               -0.029    0.049   -0.592    0.554
    s2                0.059    0.036    1.644    0.100
  s1 ~~
    s2               -0.031    0.026   -1.186    0.236
```