Example 4.4: Across-Group Invariance of the Wechsler Intelligence Scale

We have sample covariance matrices of subtests of the Wechsler Intelligence Scale (WISC) from two samples: a sample of manic depressive patients (N = 81) and a norm group (N = 200). We are going to assess whether the WISC subscales are measurement invariance across these two groups.

Assessing configural invariance

```
manic.cov <- lav_matrix_lower2full(c(
  9.364,
  7.777, 12.461,
  6.422,  8.756, 10.112,
  5.669,  7.445,  6.797, 8.123,
  3.048,  4.922,  4.513, 4.116, 6.200,
  3.505,  4.880,  4.899, 5.178, 5.114, 15.603,
  3.690,  5.440,  5.220, 3.151, 3.587,  6.219, 11.223,
  3.640,  4.641,  4.877, 3.568, 3.819,  5.811,  6.501, 9.797
))
manic.means <- c(10.09, 12.07, 10.25, 9.96, 10.90, 11.24, 10.30, 10.44)

norming.cov <- lav_matrix_lower2full(c(
  9.610,
  5.844, 8.410,
  6.324, 6.264, 9.000,
  4.405, 4.457, 5.046, 8.410,
  4.464, 4.547, 4.512, 3.712, 10.240,
  3.478, 2.967, 2.970, 2.871,  3.802, 10.890,
  5.270, 4.930, 4.080, 3.254,  5.222,  3.590, 11.560,
  4.297, 4.594, 4.356, 3.158,  4.963,  3.594,  6.620, 10.890
))
norming.means <- c(10.10, 10.30, 9.80, 10.10, 10.10, 10.10, 9.90, 10.20)

wisc3.names <- c("Info", "Sim", "Vocab","Comp", "PicComp", "PicArr",
                 "BlkDsgn", "ObjAsmb")
colnames(norming.cov) <- rownames(norming.cov)  <- colnames(manic.cov) <-
  rownames(manic.cov) <- names(norming.means) <- names(manic.means) <-
  wisc3.names

wisc3.model <- '
  VC =~ Info + Sim + Vocab + Comp
  VS =~ PicComp + PicArr + BlkDsgn + ObjAsmb
'

fit.indices <- c("chisq", "df", "pvalue", "cfi", "rmsea", "srmr", "aic",
                 "bic")

manic.fit <- cfa(wisc3.model, sample.cov = manic.cov, sample.nobs = 81,
                 sample.mean = manic.means, meanstructure = TRUE)
summary(manic.fit, standardized = TRUE)
```

```
## lavaan (0.6-1) converged normally after  42 iterations
```

```
##
##   Number of observations                            81
##
##   Estimator                                         ML
##   Model Fit Test Statistic                      29.169
##   Degrees of freedom                                19
##   P-value (Chi-square)                           0.063
##
## Parameter Estimates:
##
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##   Standard Errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VC =~
##     Info              1.000                               2.347    0.772
##     Sim               1.330    0.153    8.687    0.000    3.121    0.890
##     Vocab             1.189    0.138    8.613    0.000    2.791    0.883
##     Comp              1.015    0.125    8.129    0.000    2.382    0.841
##   VS =~
##     PicComp           1.000                               1.788    0.723
##     PicArr            1.437    0.274    5.246    0.000    2.570    0.655
##     BlkDsgn           1.322    0.234    5.641    0.000    2.364    0.710
##     ObjAsmb           1.285    0.220    5.830    0.000    2.297    0.738
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VC ~~
##     VS                3.086    0.772    3.997    0.000    0.735    0.735
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .Info             10.090    0.338   29.861    0.000   10.090    3.318
##    .Sim              12.070    0.390   30.965    0.000   12.070    3.441
##    .Vocab            10.250    0.351   29.191    0.000   10.250    3.243
##    .Comp              9.960    0.315   31.648    0.000    9.960    3.516
##    .PicComp          10.900    0.275   39.643    0.000   10.900    4.405
##    .PicArr           11.240    0.436   25.769    0.000   11.240    2.863
##    .BlkDsgn          10.300    0.370   27.843    0.000   10.300    3.094
##    .ObjAsmb          10.440    0.346   30.206    0.000   10.440    3.356
##     VC                0.000                               0.000    0.000
##     VS                0.000                               0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .Info              3.742    0.673    5.564    0.000    3.742    0.405
##    .Sim               2.564    0.605    4.237    0.000    2.564    0.208
##    .Vocab             2.200    0.502    4.379    0.000    2.200    0.220
##    .Comp              2.348    0.467    5.027    0.000    2.348    0.293
##    .PicComp           2.926    0.592    4.945    0.000    2.926    0.478
##    .PicArr            8.806    1.631    5.398    0.000    8.806    0.571
##    .BlkDsgn           5.497    1.089    5.046    0.000    5.497    0.496
```

```
##     .ObjAsmb           4.399    0.916    4.804    0.000    4.399    0.455
##      VC                5.507    1.369    4.024    0.000    1.000    1.000
##      VS                3.198    0.924    3.462    0.001    1.000    1.000
```

```r
fitMeasures(manic.fit, fit.indices)
```

```
##    chisq       df   pvalue      cfi    rmsea     srmr      aic      bic
##   29.169   19.000    0.063    0.971    0.081    0.047 3019.261 3079.122
```

The χ^2, CFI, and SRMR indicate good fit in the manic group. The RMSEA indicates less than adequate fit. All loadings are substantial and significant. The two factors also correlate substantially and significantly.

```r
norming.fit <- cfa(wisc3.model, sample.cov = norming.cov, sample.nobs = 200,
                   sample.mean = norming.means, meanstructure = TRUE)
summary(norming.fit, standardized = TRUE)
```

```
## lavaan (0.6-1) converged normally after  45 iterations
##
##   Number of observations                           200
##
##   Estimator                                         ML
##   Model Fit Test Statistic                      24.211
##   Degrees of freedom                                19
##   P-value (Chi-square)                           0.188
##
## Parameter Estimates:
##
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##   Standard Errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VC =~
##     Info              1.000                               2.440    0.789
##     Sim               0.997    0.079   12.607    0.000    2.433    0.841
##     Vocab             1.045    0.082   12.778    0.000    2.550    0.852
##     Comp              0.768    0.083    9.301    0.000    1.874    0.648
##   VS =~
##     PicComp           1.000                               2.182    0.684
##     PicArr            0.715    0.122    5.854    0.000    1.561    0.474
##     BlkDsgn           1.149    0.135    8.542    0.000    2.507    0.739
##     ObjAsmb           1.100    0.130    8.464    0.000    2.401    0.730
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VC ~~
##     VS                4.103    0.661    6.204    0.000    0.771    0.771
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .Info             10.100    0.219   46.192    0.000   10.100    3.266
##    .Sim              10.300    0.205   50.355    0.000   10.300    3.561
##    .Vocab             9.800    0.212   46.314    0.000    9.800    3.275
##    .Comp             10.100    0.205   49.377    0.000   10.100    3.491
##    .PicComp          10.100    0.226   44.748    0.000   10.100    3.164
```

```
##    .PicArr           10.100    0.233   43.392    0.000   10.100    3.068
##    .BlkDsgn           9.900    0.240   41.282    0.000    9.900    2.919
##    .ObjAsmb          10.200    0.233   43.822    0.000   10.200    3.099
##     VC                0.000                               0.000    0.000
##     VS                0.000                               0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .Info              3.609    0.455    7.928    0.000    3.609    0.377
##    .Sim               2.450    0.354    6.925    0.000    2.450    0.293
##    .Vocab             2.453    0.370    6.635    0.000    2.453    0.274
##    .Comp              4.857    0.533    9.114    0.000    4.857    0.580
##    .PicComp           5.426    0.675    8.034    0.000    5.426    0.533
##    .PicArr            8.398    0.897    9.364    0.000    8.398    0.775
##    .BlkDsgn           5.215    0.716    7.279    0.000    5.215    0.453
##    .ObjAsmb           5.069    0.682    7.434    0.000    5.069    0.468
##     VC                5.953    0.928    6.416    0.000    1.000    1.000
##     VS                4.763    0.952    5.006    0.000    1.000    1.000
```

```
fitMeasures(norming.fit, fit.indices)
```

```
##    chisq       df    pvalue      cfi     rmsea      srmr       aic       bic
##   24.211   19.000     0.188    0.992     0.037     0.029  7564.068  7646.526
```

We see better model fit in the norm group. So perhaps we should not just assume configural invariance here. In such a case, I would first inspect the fitted model, residuals and modification indices in each of the two groups.

```
residuals(manic.fit, type = "cor")
```

```
## $type
## [1] "cor.bollen"
##
## $cor
##         Info   Sim    Vocab  Comp   PicCmp PicArr BlkDsg ObjAsm
## Info     0.000
## Sim      0.033  0.000
## Vocab   -0.021 -0.006  0.000
## Comp     0.001 -0.008  0.007  0.000
## PicComp -0.010  0.087  0.101  0.133  0.000
## PicArr  -0.081 -0.078 -0.035  0.055  0.047  0.000
## BlkDsgn -0.043 -0.005  0.029 -0.109 -0.083  0.005  0.000
## ObjAsmb -0.039 -0.063  0.010 -0.057 -0.044 -0.013  0.096  0.000
##
## $mean
##    Info     Sim   Vocab    Comp PicComp  PicArr BlkDsgn ObjAsmb
##       0       0       0       0       0       0       0       0
```

```
modindices(manic.fit, sort = TRUE)[1:10, ]
```

```
##        lhs op     rhs    mi    epc sepc.lv sepc.all sepc.nox
## 65 BlkDsgn ~~ ObjAsmb 7.391  2.227   2.227    0.453    0.453
## 58    Comp ~~ BlkDsgn 7.116 -1.304  -1.304   -0.363   -0.363
## 30      VC =~ PicComp 6.621  0.484   1.137    0.459    0.459
## 61 PicComp ~~ BlkDsgn 4.992 -1.448  -1.448   -0.361   -0.361
## 57    Comp ~~  PicArr 4.737  1.310   1.310    0.288    0.288
## 56    Comp ~~ PicComp 3.153  0.639   0.639    0.244    0.244
```

```
## 38    Info ~~     Sim 2.394  0.796   0.796    0.257    0.257
## 48     Sim ~~  PicArr 1.791 -0.907  -0.907   -0.191   -0.191
## 62 PicComp ~~ ObjAsmb 1.675 -0.792  -0.792   -0.221   -0.221
## 36      VS =~   Vocab 1.583  0.278   0.497    0.157    0.157
```

In the manic group, there are standardized residuals > .1 for PicComp ~~ Comp, Piccomp ~~ Vocab, Comp ~~ BlkDsgn. Modification indices suggest adding BlkDsgn ~~ ObjAsmb.

```
residuals(norming.fit, type = "cor")
```

```
## $type
## [1] "cor.bollen"
##
## $cor
##         Info   Sim    Vocab  Comp   PicCmp PicArr BlkDsg ObjAsm
## Info     0.000
## Sim     -0.014  0.000
## Vocab    0.008  0.003  0.000
## Comp    -0.021 -0.015  0.028  0.000
## PicComp  0.034  0.047  0.021  0.059  0.000
## PicArr   0.052  0.003 -0.011  0.063  0.036  0.000
## BlkDsgn  0.051  0.021 -0.085 -0.039 -0.026 -0.031  0.000
## ObjAsmb -0.023  0.007 -0.039 -0.034 -0.029 -0.016  0.051  0.000
##
## $mean
##    Info     Sim   Vocab    Comp PicComp  PicArr BlkDsgn ObjAsmb
##       0       0       0       0       0       0       0       0
```

```
modindices(norming.fit, sort = TRUE)[1:10, ]
```

```
##        lhs op     rhs    mi    epc sepc.lv sepc.all sepc.nox
## 54   Vocab ~~ BlkDsgn 7.962 -0.988  -0.988   -0.276   -0.276
## 65 BlkDsgn ~~ ObjAsmb 7.148  1.670   1.670    0.325    0.325
## 36      VS =~   Vocab 5.765 -0.359  -0.784   -0.262   -0.262
## 43    Info ~~ BlkDsgn 4.150  0.794   0.794    0.183    0.183
## 30      VC =~ PicComp 3.196  0.326   0.796    0.249    0.249
## 51   Vocab ~~    Comp 1.936  0.475   0.475    0.137    0.137
## 35      VS =~     Sim 1.805  0.195   0.425    0.147    0.147
## 62 PicComp ~~ ObjAsmb 1.528 -0.694  -0.694   -0.132   -0.132
## 57    Comp ~~  PicArr 1.468  0.586   0.586    0.092    0.092
## 34      VS =~    Info 1.447  0.190   0.416    0.134    0.134
```

In the norming group, there are no standardized residuals > .1. Modification indices for the norming group also suggest adding BlkDsgn ~~ ObjAsmb. These may indeed have something in common that is not shared by the other indictors of the Visuo-Spatial factor. Should we add a correlated error to the model? That is a decision you, the researcher, have to make. Adding correlated errors violates the assumption of conditional independence: that conditional on the common factor(s), the observed indicators are independent. So I would rather not add a correlated error. Later on, one could perform a sensitivity analysis to check whether the addition of a correlated error would have led to a different conclusion.

For now, let's carry on with the multigroup analysis, while assuming configural invariance.

We first have to combine the covariance matrices, sample sizes and means of both groups into lists:

```
combined.cov <- list(manic = manic.cov, norming = norming.cov)
combined.n <- list(manic = 81, norming = 200)
combined.means <- list(manic = manic.means, norming = norming.means)
```

Note that in practice, you will often analyse the whole dataset, so combining the means and covariances into lists is not necessary. With the raw data, you would specify the `data` and `group` arguments, instead of the `sample.cov`, `sample.nobs` and `sample.mean` arguments.

```
configural.fit <- cfa(wisc3.model, sample.cov = combined.cov,
                      sample.nobs = combined.n, sample.mean = combined.means,
                      meanstructure = TRUE)
fitMeasures(configural.fit, fit.indices)
```

```
##     chisq        df    pvalue       cfi     rmsea      srmr       aic
##    53.380    38.000     0.050     0.985     0.054     0.034 10583.329
##       bic
## 10765.247
```

These fit indices indicate configural invariances is tenable: Although the chi-sqaure and RMSEA values are on the borderline of good fit, CFI and SRMR indicate good fit. Now we will constrain the loadings to be equal across groups:

Assessing metric invariance

```
metric.fit <- cfa(wisc3.model, sample.cov = combined.cov,
                  sample.nobs = combined.n, sample.mean = combined.means,
                  meanstructure = TRUE, group.equal = "loadings")
fitMeasures(metric.fit, fit.indices)
```

```
##     chisq        df    pvalue       cfi     rmsea      srmr       aic
##    65.992    44.000     0.018     0.979     0.060     0.055 10583.942
##       bic
## 10744.029
```

```
lavTestLRT(metric.fit, configural.fit)
```

```
## Chi Square Difference Test
##
##                Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## configural.fit 38 10583 10765 53.380
## metric.fit     44 10584 10744 65.992     12.613       6    0.04961 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

According to the fit indices, the fit of the metric invariant model is acceptable. Also, by restricting the loadings to be equal across groups, ΔCFI was $< .01$. However, the $\Delta\chi^2$ test indicates a significant difference in model fit, and BIC and AIC also indicate a deterioration of fit.

We can use modification indices to find out which parameter restriction causes the misfit. In newer versions of the lavaan package, the modificationindices() function does not give modification indices for parameters that are restricted to equality anymore. You have to use the lavTestScore() function for that:

```
lavTestScore(metric.fit)
```

```
## $test
##
## total score test:
##
##    test     X2 df p.value
## 1 score 12.477  6   0.052
##
```

```
## $uni
##
## univariate score tests:
##
##    lhs op   rhs    X2 df p.value
## 1 .p2. == .p31. 2.501  1   0.114
## 2 .p3. == .p32. 0.557  1   0.455
## 3 .p4. == .p33. 0.874  1   0.350
## 4 .p6. == .p35. 6.144  1   0.013
## 5 .p7. == .p36. 0.042  1   0.837
## 6 .p8. == .p37. 0.001  1   0.971
```

```
pars <- parameterestimates(metric.fit)
pars[pars$label == ".p6.",]
```

```
##    lhs op    rhs block group label   est    se     z pvalue ci.lower
## 6   VS =~ PicArr     1     1  .p6. 0.888 0.118 7.554      0    0.658
## 35  VS =~ PicArr     2     2  .p6. 0.888 0.118 7.554      0    0.658
##    ci.upper
## 6     1.118
## 35    1.118
```

The modification indices suggest lifting the restriction on parameter 6, which is the factor loading of Picture Arrangement on the VS factor. We specify this parameter with the `group.partial` argument, to release the equality restriction imposed by the `group.equal` command:

```
metric.fit2 <- cfa(wisc3.model, sample.cov = combined.cov,
                   sample.nobs = combined.n, sample.mean = combined.means,
                   meanstructure = TRUE, group.equal = "loadings",
                   group.partial = "VS =~ PicArr")
fitmeasures(metric.fit2, fit.indices)
```

```
##     chisq        df    pvalue       cfi     rmsea      srmr       aic
##    59.500    43.000     0.048     0.984     0.052     0.045 10579.449
##       bic
## 10743.175
```

```
lavTestLRT(metric.fit2, configural.fit)
```

```
## Chi Square Difference Test
##
##                Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## configural.fit 38 10583 10765 53.380
## metric.fit2    43 10579 10743 59.499       6.12       5     0.2947
```

We obtained an adequately fitting partial metric invariance model.

Assessing scalar invariance

```
scalar.fit <- cfa(wisc3.model, sample.cov = combined.cov,
                  sample.nobs = combined.n, sample.mean = combined.means,
                  meanstructure = TRUE,
                  group.equal = c("loadings", "intercepts"),
                  group.partial = "VS =~ PicArr")
fitMeasures(scalar.fit, fit.indices)
```

```
##     chisq        df    pvalue       cfi     rmsea      srmr       aic 
##   103.076    49.000     0.000     0.947     0.089     0.056 10611.025 
##       bic 
## 10752.921
```

```r
lavTestLRT(metric.fit2, scalar.fit)
```

```
## Chi Square Difference Test
## 
##             Df   AIC   BIC   Chisq Chisq diff Df diff Pr(>Chisq)    
## metric.fit2 43 10579 10743  59.499                                  
## scalar.fit  49 10611 10753 103.076     43.576       6   8.97e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

According to all fit indices but the SRMR, the fit of the scalar invariance model is not acceptable. Also, ΔCFI was $> .01$ and $\Delta\chi^2$ was significant.

```r
lavTestScore(scalar.fit)
```

```
## $test
## 
## total score test:
## 
##    test     X2 df p.value
## 1 score 45.098 13       0
## 
## $uni
## 
## univariate score tests:
## 
##      lhs op   rhs     X2 df p.value
## 1   .p2. == .p31.  1.157  1   0.282
## 2   .p3. == .p32.  0.199  1   0.655
## 3   .p4. == .p33.  1.384  1   0.239
## 4   .p7. == .p36.  0.076  1   0.782
## 5   .p8. == .p37.  0.218  1   0.641
## 6  .p20. == .p49.  5.123  1   0.024
## 7  .p21. == .p50. 29.321  1   0.000
## 8  .p22. == .p51.  0.819  1   0.366
## 9  .p23. == .p52.  7.470  1   0.006
## 10 .p24. == .p53.  1.739  1   0.187
## 11 .p25. == .p54.  3.491  1   0.062
## 12 .p26. == .p55.  0.742  1   0.389
## 13 .p27. == .p56.  2.279  1   0.131
```

Especially the equality restriction on parameter p21 causes misfit. To a lesser extent also the equality restrictions on p23 and p20.

```r
pars <- parameterestimates(scalar.fit)
pars[pars$label == ".p21.",]
```

```
##    lhs op rhs block group label    est    se      z pvalue ci.lower
## 21 Sim ~1         1     1 .p21. 10.976 0.344 31.938      0   10.303
## 50 Sim ~1         2     2 .p21. 10.976 0.344 31.938      0   10.303
##    ci.upper
## 21    11.65
```

```
## 50    11.65
```

The intercepts of the Similarities subtest are probably not equal across the two groups. We release that equality restriction:

```
scalar.fit2 <- cfa(wisc3.model, sample.cov = combined.cov,
                   sample.nobs = combined.n, sample.mean = combined.means,
                   group.equal = c("loadings", "intercepts"),
                   group.partial = c("VS =~ PicArr", "Sim~1"),
                   meanstructure = TRUE)
fitMeasures(scalar.fit2, fit.indices)
```

```
##     chisq        df    pvalue       cfi     rmsea      srmr       aic
##    69.921    48.000     0.021     0.979     0.057     0.049 10579.870
##       bic
## 10725.404
```

```
lavTestLRT(metric.fit2, scalar.fit2)
```

```
## Chi Square Difference Test
##
##             Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## metric.fit2 43 10579 10743 59.499
## scalar.fit2 48 10580 10725 69.921     10.421       5    0.06415 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Now, the model fits well and the difference with the partial metric invariance model is not significant anymore. Thus, partial scalar invariance is tenable. We now proceed with testing across-group equality of residual variances:

Assessing uniqueness invariance

```
uniqueness.fit <- cfa(wisc3.model, sample.cov = combined.cov,
                      sample.nobs = combined.n,
                      sample.mean = combined.means,
                      group.equal=c("loadings", "intercepts", "residuals"),
                      group.partial = c("VS =~ PicArr", "Sim~1"))
fitMeasures(uniqueness.fit, fit.indices)
```

```
##     chisq        df    pvalue       cfi     rmsea      srmr       aic
##    88.019    56.000     0.004     0.969     0.064     0.057 10581.969
##       bic
## 10698.396
```

```
lavTestLRT(uniqueness.fit, scalar.fit2)
```

```
## Chi Square Difference Test
##
##                Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## scalar.fit2    48 10580 10725 69.921
## uniqueness.fit 56 10582 10698 88.019     18.099       8     0.0205 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ΔCFI is just on the cut-off value of $> .01$ and $\Delta\chi^2$ is significant.

```r
lavTestScore(uniqueness.fit)
```

```
## $test
##
## total score test:
##
##    test     X2 df p.value
## 1 score 30.506 20   0.062
##
## $uni
##
## univariate score tests:
##
##      lhs op   rhs    X2 df p.value
## 1    .p2. == .p31. 3.967  1   0.046
## 2    .p3. == .p32. 0.694  1   0.405
## 3    .p4. == .p33. 0.122  1   0.727
## 4    .p7. == .p36. 0.634  1   0.426
## 5    .p8. == .p37. 0.301  1   0.584
## 6    .p9. == .p38. 0.027  1   0.869
## 7   .p10. == .p39. 0.200  1   0.655
## 8   .p11. == .p40. 0.241  1   0.624
## 9   .p12. == .p41. 7.293  1   0.007
## 10  .p13. == .p42. 6.547  1   0.011
## 11  .p14. == .p43. 0.143  1   0.706
## 12  .p15. == .p44. 0.015  1   0.904
## 13  .p16. == .p45. 0.326  1   0.568
## 14  .p20. == .p49. 0.820  1   0.365
## 15  .p22. == .p51. 3.119  1   0.077
## 16  .p23. == .p52. 1.490  1   0.222
## 17  .p24. == .p53. 1.401  1   0.237
## 18  .p25. == .p54. 3.786  1   0.052
## 19  .p26. == .p55. 0.653  1   0.419
## 20  .p27. == .p56. 1.948  1   0.163
```

Equality restrictions on p12 and p13 seem problematic. What parameters are they?

```r
pars <- parameterestimates(uniqueness.fit)
pars[pars$label %in% c(".p12.", ".p13."),]
```

```
##        lhs op     rhs block group label   est    se      z pvalue ci.lower
## 12    Comp ~~    Comp     1     1 .p12. 4.166 0.395 10.537      0    3.392
## 13 PicComp ~~ PicComp     1     1 .p13. 4.669 0.497  9.389      0    3.694
## 41    Comp ~~    Comp     2     2 .p12. 4.166 0.395 10.537      0    3.392
## 42 PicComp ~~ PicComp     2     2 .p13. 4.669 0.497  9.389      0    3.694
##    ci.upper
## 12    4.941
## 13    5.644
## 41    4.941
## 42    5.644
```

The residual variances of Picture Completion and Comprehension do not seem equal across groups. We add those parameters to the `group.partial` argument:

```r
uniqueness.fit2 <- cfa(wisc3.model, sample.cov = combined.cov,
                       sample.nobs = combined.n,
```

```r
                     sample.mean = combined.means,
                     group.equal = c("loadings", "intercepts", "residuals"),
                     group.partial = c("Sim~1", "VS=~PicArr", "PicComp~~PicComp",
                                       "Comp~~Comp"))
fitMeasures(uniqueness.fit2, fit.indices)
```

```
##     chisq        df    pvalue       cfi     rmsea      srmr       aic
##    71.017    54.000     0.060     0.983     0.047     0.050 10568.966
##       bic
## 10692.670
```

```r
lavTestLRT(scalar.fit2, uniqueness.fit2)
```

```
## Chi Square Difference Test
##
##                 Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## scalar.fit2     48 10580 10725 69.921
## uniqueness.fit2 54 10569 10693 71.017      1.096       6     0.9817
```

We have obtained an adequately fitting partial uniqueness invariance model. We now proceed to test structural invariance across the two groups.

Assessing structural invariance

```r
factor.var.fit <- cfa(wisc3.model, sample.cov = combined.cov,
                      sample.nobs = combined.n,
                      sample.mean = combined.means,
                      group.equal = c("loadings", "intercepts", "residuals",
                                      "lv.variances"),
                      group.partial = c("Sim~1", "VS=~PicArr", "PicComp~~PicComp",
                                        "Comp~~Comp"))
fitMeasures(factor.var.fit, fit.indices)
```

```
##     chisq        df    pvalue       cfi     rmsea      srmr       aic
##    75.387    56.000     0.043     0.981     0.050     0.065 10569.336
##       bic
## 10685.763
```

```r
lavTestLRT(factor.var.fit, uniqueness.fit2)
```

```
## Chi Square Difference Test
##
##                 Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## uniqueness.fit2 54 10569 10693 71.017
## factor.var.fit  56 10569 10686 75.387     4.3701       2     0.1125
```

Equality of factor variances is also tenable. We proceed to test the equality of factor covariances:

```r
factor.covar.fit <- cfa(wisc3.model, sample.cov = combined.cov,
                        sample.nobs = combined.n,
                        sample.mean = combined.means,
                        group.equal = c("loadings", "intercepts", "residuals",
                                        "lv.variances", "lv.covariances"),
                        group.partial = c("Sim~1", "VS=~PicArr", "PicComp~~PicComp",
                                          "Comp~~Comp"))
fitMeasures(factor.covar.fit, fit.indices)
```

```
##     chisq        df    pvalue       cfi     rmsea      srmr       aic
##    75.889    57.000     0.048     0.982     0.049     0.065 10567.839
##       bic
## 10680.628
```

```r
lavTestLRT(factor.var.fit, factor.covar.fit)
```

```
## Chi Square Difference Test
##
##                  Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## factor.var.fit   56 10569 10686 75.387
## factor.covar.fit 57 10568 10681 75.889    0.50264       1     0.4783
```

Equality of the factor covariance(s) is also tenable. We proceed to test the equality of factor means:

```r
factor.means.fit <- cfa(wisc3.model, sample.cov = combined.cov,
                        sample.nobs = combined.n,
                        sample.mean = combined.means,
                        group.equal = c("loadings", "intercepts", "residuals",
                                        "lv.variances", "lv.covariances",
                                        "means"),
                        group.partial = c("Sim~1", "VS=~PicArr", "PicComp~~PicComp",
                                          "Comp~~Comp"))
fitMeasures(factor.means.fit, fit.indices)
```

```
##     chisq        df    pvalue       cfi     rmsea      srmr       aic
##    78.163    59.000     0.048     0.981     0.048     0.069 10566.112
##       bic
## 10671.624
```

```r
lavTestLRT(factor.covar.fit, factor.means.fit)
```

```
## Chi Square Difference Test
##
##                  Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## factor.covar.fit 57 10568 10681 75.889
## factor.means.fit 59 10566 10672 78.162     2.2732       2     0.3209
```

Equality of factor means is also tenable. Thus, no structural parameters are significantly different across groups. Thus, we can assume the manic and norm group have equal means, variances and associations between the VS and VC variables.

We inspect the differences in intercepts and uniquenesses between groups in our final, best-fitting model:

```r
summary(factor.means.fit, standardized = TRUE)
```

```
## lavaan (0.6-1) converged normally after  64 iterations
##
##   Number of observations per group
##   manic                                             81
##   norming                                          200
##
##   Estimator                                         ML
##   Model Fit Test Statistic                      78.163
##   Degrees of freedom                                59
##   P-value (Chi-square)                           0.048
##
```

```
## Chi-square for each group:
##
##   manic                                       45.688
##   norming                                     32.474
##
## Parameter Estimates:
##
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##   Standard Errors                             Standard
##
##
## Group 1 [manic]:
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VC =~
##     Info              1.000                               2.396    0.779
##     Sim     (.p2.)    1.104    0.073   15.224    0.000    2.645    0.857
##     Vocab   (.p3.)    1.099    0.072   15.343    0.000    2.634    0.864
##     Comp    (.p4.)    0.864    0.069   12.553    0.000    2.071    0.795
##   VS =~
##     PicComp           1.000                               2.046    0.770
##     PicArr            1.341    0.203    6.612    0.000    2.743    0.683
##     BlkDsgn (.p7.)    1.208    0.118   10.268    0.000    2.472    0.732
##     ObjAsmb (.p8.)    1.164    0.113   10.300    0.000    2.382    0.735
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VC ~~
##     VS      (.19.)    3.683    0.503    7.327    0.000    0.751    0.751
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .Info    (.20.)   10.063    0.183   55.085    0.000   10.063    3.270
##    .Sim              11.863    0.252   47.147    0.000   11.863    3.846
##    .Vocab   (.22.)    9.892    0.181   54.728    0.000    9.892    3.245
##    .Comp    (.23.)    9.981    0.169   58.896    0.000    9.981    3.833
##    .PicComp (.24.)   10.355    0.174   59.598    0.000   10.355    3.895
##    .PicArr  (.25.)   10.347    0.207   49.899    0.000   10.347    2.577
##    .BlkDsgn (.26.)    9.969    0.200   49.848    0.000    9.969    2.950
##    .ObjAsmb (.27.)   10.224    0.192   53.277    0.000   10.224    3.153
##     VC                0.000                               0.000    0.000
##     VS                0.000                               0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .Info    (.p9.)    3.731    0.381    9.792    0.000    3.731    0.394
##    .Sim     (.10.)    2.520    0.313    8.054    0.000    2.520    0.265
##    .Vocab   (.11.)    2.357    0.301    7.835    0.000    2.357    0.254
##    .Comp              2.493    0.470    5.300    0.000    2.493    0.368
##    .PicComp           2.881    0.594    4.850    0.000    2.881    0.408
##    .PicArr  (.14.)    8.597    0.797   10.780    0.000    8.597    0.533
##    .BlkDsgn (.15.)    5.308    0.600    8.841    0.000    5.308    0.465
```

```
##    .ObjAsmb (.16.)    4.840    0.551    8.784    0.000    4.840    0.460
##     VC      (.17.)    5.741    0.765    7.502    0.000    1.000    1.000
##     VS      (.18.)    4.188    0.688    6.088    0.000    1.000    1.000
##
##
## Group 2 [norming]:
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VC =~
##     Info              1.000                               2.396    0.779
##     Sim     (.p2.)    1.104    0.073   15.224    0.000    2.645    0.857
##     Vocab   (.p3.)    1.099    0.072   15.343    0.000    2.634    0.864
##     Comp    (.p4.)    0.864    0.069   12.553    0.000    2.071    0.686
##   VS =~
##     PicComp           1.000                               2.046    0.654
##     PicArr            0.758    0.127    5.971    0.000    1.551    0.468
##     BlkDsgn (.p7.)    1.208    0.118   10.268    0.000    2.472    0.732
##     ObjAsmb (.p8.)    1.164    0.113   10.300    0.000    2.382    0.735
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VC ~~
##     VS      (.19.)    3.683    0.503    7.327    0.000    0.751    0.751
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .Info    (.20.)   10.063    0.183   55.085    0.000   10.063    3.270
##    .Sim              10.331    0.196   52.636    0.000   10.331    3.349
##    .Vocab   (.22.)    9.892    0.181   54.728    0.000    9.892    3.245
##    .Comp    (.23.)    9.981    0.169   58.896    0.000    9.981    3.306
##    .PicComp (.24.)   10.355    0.174   59.598    0.000   10.355    3.308
##    .PicArr  (.25.)   10.347    0.207   49.899    0.000   10.347    3.119
##    .BlkDsgn (.26.)    9.969    0.200   49.848    0.000    9.969    2.950
##    .ObjAsmb (.27.)   10.224    0.192   53.277    0.000   10.224    3.153
##     VC                0.000                               0.000    0.000
##     VS                0.000                               0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .Info    (.p9.)    3.731    0.381    9.792    0.000    3.731    0.394
##    .Sim     (.10.)    2.520    0.313    8.054    0.000    2.520    0.265
##    .Vocab   (.11.)    2.357    0.301    7.835    0.000    2.357    0.254
##    .Comp              4.830    0.535    9.031    0.000    4.830    0.530
##    .PicComp           5.611    0.669    8.381    0.000    5.611    0.573
##    .PicArr  (.14.)    8.597    0.797   10.780    0.000    8.597    0.781
##    .BlkDsgn (.15.)    5.308    0.600    8.841    0.000    5.308    0.465
##    .ObjAsmb (.16.)    4.840    0.551    8.784    0.000    4.840    0.460
##     VC      (.17.)    5.741    0.765    7.502    0.000    1.000    1.000
##     VS      (.18.)    4.188    0.688    6.088    0.000    1.000    1.000
```

The loading for Picture Arrangement is higher in the manic group than in the norm group. The intercept for Similarities is higher in the manic than in the norm group. The residual variances for Comprehension and Picture Completion are higher in the norm than in the manic group.

Example 4.6: Genetically Informative Design

```
library(lavaan)
```

We will analyze BMI values of MZ and DZ twins to assess the extent to which BMI is determined by additive and non-additive genetic effects.

First, we create the covariance matrices:

```
MZ <- lav_matrix_lower2full(c(
  .725,
  .589, .792
))
DZ <- lav_matrix_lower2full(c(
  .779,
  .246, .837
))
rownames(MZ) <- colnames(MZ) <- rownames(DZ)  <- colnames(DZ) <- c("P1", "P2")

bmi.cov <- list(MZ=MZ, DZ=DZ)
bmi.n <- list(MZ=534, DZ=328)

bmi.ade.mod <- '

  # build the factor model with group constraints:
  A1=~ NA*P1 + c(a,a)*P1
  A2=~ NA*P2 + c(a,a)*P2
  D1 =~ NA*P1 + c(d,d)*P1
  D2 =~ NA*P2 + c(d,d)*P2

  # constrain the factor variances:
  A1 ~~ 1*A1
  A2 ~~ 1*A2
  D1 ~~ 1*D1
  D2 ~~ 1*D2
  P1 ~~ c(e2,e2)*P1
  P2 ~~ c(e2,e2)*P2

  # constrain the factor covariances:
  A1 ~~ c(1,.5)*A2
  A1 ~~ 0*D1 + 0*D2
  A2 ~~ 0*D1 + 0*D2
  D1 ~~ c(1,.25)*D2
'

bmi.ade.fit <- cfa(bmi.ade.mod, sample.cov=bmi.cov, sample.nobs=bmi.n)
summary(bmi.ade.fit, standardized=TRUE)
```

```
## lavaan (0.6-1) converged normally after  17 iterations
##
##   Number of observations per group
##   MZ                                               534
##   DZ                                               328
##
##   Estimator                                         ML
```

```
##   Model Fit Test Statistic                      3.704
##   Degrees of freedom                                3
##   P-value (Chi-square)                          0.295
##
## Chi-square for each group:
##
##   MZ                                            2.927
##   DZ                                            0.778
##
## Parameter Estimates:
##
##   Information                                Expected
##   Information saturated (h1) model         Structured
##   Standard Errors                            Standard
##
##
## Group 1 [MZ]:
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   A1 =~
##     P1        (a)    0.562    0.139    4.053    0.000    0.562    0.636
##   A2 =~
##     P2        (a)    0.562    0.139    4.053    0.000    0.562    0.636
##   D1 =~
##     P1        (d)    0.543    0.140    3.874    0.000    0.543    0.615
##   D2 =~
##     P2        (d)    0.543    0.140    3.874    0.000    0.543    0.615
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   A1 ~~
##     A2               1.000                               1.000    1.000
##     D1               0.000                               0.000    0.000
##     D2               0.000                               0.000    0.000
##   A2 ~~
##     D1               0.000                               0.000    0.000
##     D2               0.000                               0.000    0.000
##   D1 ~~
##     D2               1.000                               1.000    1.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##     A1               1.000                               1.000    1.000
##     A2               1.000                               1.000    1.000
##     D1               1.000                               1.000    1.000
##     D2               1.000                               1.000    1.000
##    .P1       (e2)    0.170    0.010   16.398    0.000    0.170    0.218
##    .P2       (e2)    0.170    0.010   16.398    0.000    0.170    0.218
##
##
## Group 2 [DZ]:
##
## Latent Variables:
```

```
##                     Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   A1 =~
##     P1         (a)    0.562    0.139    4.053    0.000    0.562    0.636
##   A2 =~
##     P2         (a)    0.562    0.139    4.053    0.000    0.562    0.636
##   D1 =~
##     P1         (d)    0.543    0.140    3.874    0.000    0.543    0.615
##   D2 =~
##     P2         (d)    0.543    0.140    3.874    0.000    0.543    0.615
##
## Covariances:
##                     Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   A1 ~~
##     A2                0.500                               0.500    0.500
##     D1                0.000                               0.000    0.000
##     D2                0.000                               0.000    0.000
##   A2 ~~
##     D1                0.000                               0.000    0.000
##     D2                0.000                               0.000    0.000
##   D1 ~~
##     D2                0.250                               0.250    0.250
##
## Variances:
##                     Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##     A1                1.000                               1.000    1.000
##     A2                1.000                               1.000    1.000
##     D1                1.000                               1.000    1.000
##     D2                1.000                               1.000    1.000
##    .P1        (e2)    0.170    0.010   16.398    0.000    0.170    0.218
##    .P2        (e2)    0.170    0.010   16.398    0.000    0.170    0.218
```