Analysis 2011

Contents

Preparations

Load packages

```
library(here)
library(dplyr)
library(labelled)
library(ggplot2)
library(tidyr)
library(stringr)
library(psych)
library(lavaan)
library(semTools)
library(semPlot)
library(haven)
library(sjlabelled)
#library(robumeta)
```

Read data file

```
df2011 <- readRDS("data/final/candsurvey_vaa_2011.rds")
```

Select variables used in the analysis

```
VAA_LR_items<-c("y22","y23","y26","y27","y9","y19")
VAA_LR_items %in% names(df2011)
```

```
## [1] TRUE TRUE TRUE TRUE TRUE TRUE
```

```
VAA_GT_items<-c("y4","y5","y1","y21")
VAA_GT_items %in% names(df2011)
```

```
## [1] TRUE TRUE TRUE TRUE
```

```
CS_LR_items<-c("C1_2","C1_7","C1_8")
CS_LR_items %in% names(df2011)
```

```
## [1] TRUE TRUE TRUE
```

```
CS_GT_items<-c("C1_1","C1_3","C1_4","C1_5","C1_6","C1_10","C1_11")
CS_GT_items %in% names(df2011)
```

```
## [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE
```

```
Party_item<-c("puolue")
Party_item %in% names(df2011)
```

```
## [1] TRUE
```

```
#vector for all item names

all_items<-c(Party_item,
             VAA_LR_items,
             VAA_GT_items,
             CS_LR_items,
             CS_GT_items)

#vector for observed variables in CFA (and party)
```

```r
obs_items<-c(Party_item,
             VAA_LR_items,
             VAA_GT_items,
             CS_LR_items,
             CS_GT_items)
```

Print the responses to the observed items

```r
for (i in 1:length(obs_items)){
  print(obs_items[i])
  print(table(df2011[,obs_items[i]],useNA="always"))
  }
```

```
## [1] "puolue"
##
##   IP   KD KESK  KOK  KTP  M11 Muut  PIR   PS  RKP  SDP  SKP  SSP  STP  VAS VIHR
##   69  192  233  232   46   66   34  127  239   83  238  144   44   47  237  228
##   VP <NA>
##   60    0
## [1] "y22"
##
##    1    2    3    4    5 <NA>
##  574  515   22  573  216  419
## [1] "y23"
##
##    1    2    3    4    5 <NA>
##  744  654   22  360  120  419
## [1] "y26"
##
##    1    2    3    4    5 <NA>
##   73  204    2  618 1002  420
## [1] "y27"
##
##    1    2    3    4    5 <NA>
##  145  398    3  805  550  418
## [1] "y9"
##
##    1    2    3    4    5 <NA>
##  186  463   13  747  496  414
## [1] "y19"
##
##    1    2    3    4    5 <NA>
##   87  426    6  530  854  416
## [1] "y4"
##
##    1    2    3    4    5 <NA>
##  341  294   20  407  849  408
## [1] "y5"
##
##    1    2    3    4    5 <NA>
##  370  538   12  548  441  410
## [1] "y1"
##
##    1    2    3    4    5 <NA>
##  628  611    4  591   81  404
```

```
## [1] "y21"
##
##    1    2    3    4    5 <NA>
##  190  505   80  607  518  419
## [1] "C1_2"
##
##    1    2    3    4    5   99 <NA>
##  424  309   42  100   26   14 1404
## [1] "C1_7"
##
##    1    2    3    4    5   99 <NA>
##   11   66   56  449  320   13 1404
## [1] "C1_8"
##
##    1    2    3    4    5   99 <NA>
##   16   89   66  306  424   14 1404
## [1] "C1_1"
##
##    1    2    3    4    5   99 <NA>
##   15   82   69  440  296   13 1404
## [1] "C1_3"
##
##    1    2    3    4    5   99 <NA>
##   27   86   83  322  387   10 1404
## [1] "C1_4"
##
##    1    2    3    4    5   99 <NA>
##  469  111   83   69  166   17 1404
## [1] "C1_5"
##
##    1    2    3    4    5   99 <NA>
##  208  209  268  161   53   16 1404
## [1] "C1_6"
##
##    1    2    3    4    5   99 <NA>
##   37  162  136  366  200   14 1404
## [1] "C1_10"
##
##    1    2    3    4    5   99 <NA>
##   76  136  114  388  191   10 1404
## [1] "C1_11"
##
##    1    2    3    4    5   99 <NA>
##   28   54   55  190  571   17 1404
```

Recode middle-responses (3) from yle items to NA, and 99 responses from CS to NA

```
VAA_items<-c(VAA_LR_items,VAA_GT_items)
CS_items<-c(CS_LR_items,CS_GT_items)

three.to.na<-function(var){
  return(ifelse(var==3,NA,var))
}

df2011[,VAA_items]<-sapply(df2011[,VAA_items],three.to.na)
```

```
ninenine.to.na<-function(var){
  return(ifelse(var==99,NA,var))
}

df2011[,CS_items]<-sapply(df2011[,CS_items],ninenine.to.na)



for (i in 1:length(all_items)){
  print(all_items[i])
  print(table(df2011[,all_items[i]],useNA="always"))
  }
```

```
## [1] "puolue"
##
##   IP   KD KESK  KOK  KTP  M11 Muut  PIR   PS  RKP  SDP  SKP  SSP  STP  VAS VIHR
##   69  192  233  232   46   66   34  127  239   83  238  144   44   47  237  228
##   VP <NA>
##   60    0
## [1] "y22"
##
##    1    2    4    5 <NA>
##  574  515  573  216  441
## [1] "y23"
##
##    1    2    4    5 <NA>
##  744  654  360  120  441
## [1] "y26"
##
##    1    2    4    5 <NA>
##   73  204  618 1002  422
## [1] "y27"
##
##    1    2    4    5 <NA>
##  145  398  805  550  421
## [1] "y9"
##
##    1    2    4    5 <NA>
##  186  463  747  496  427
## [1] "y19"
##
##    1    2    4    5 <NA>
##   87  426  530  854  422
## [1] "y4"
##
##    1    2    4    5 <NA>
##  341  294  407  849  428
## [1] "y5"
##
##    1    2    4    5 <NA>
##  370  538  548  441  422
## [1] "y1"
##
```

```
##    1    2    4    5 <NA> 
##  628  611  591   81  408 
## [1] "y21"
## 
##    1    2    4    5 <NA> 
##  190  505  607  518  499 
## [1] "C1_2"
## 
##    1    2    3    4    5 <NA> 
##  424  309   42  100   26 1418 
## [1] "C1_7"
## 
##    1    2    3    4    5 <NA> 
##   11   66   56  449  320 1417 
## [1] "C1_8"
## 
##    1    2    3    4    5 <NA> 
##   16   89   66  306  424 1418 
## [1] "C1_1"
## 
##    1    2    3    4    5 <NA> 
##   15   82   69  440  296 1417 
## [1] "C1_3"
## 
##    1    2    3    4    5 <NA> 
##   27   86   83  322  387 1414 
## [1] "C1_4"
## 
##    1    2    3    4    5 <NA> 
##  469  111   83   69  166 1421 
## [1] "C1_5"
## 
##    1    2    3    4    5 <NA> 
##  208  209  268  161   53 1420 
## [1] "C1_6"
## 
##    1    2    3    4    5 <NA> 
##   37  162  136  366  200 1418 
## [1] "C1_10"
## 
##    1    2    3    4    5 <NA> 
##   76  136  114  388  191 1414 
## [1] "C1_11"
## 
##    1    2    3    4    5 <NA> 
##   28   54   55  190  571 1421
```

Exclude completely missing cases

```
df2011$completely_missing<-
  rowSums(is.na(df2011[,obs_items[2:length(obs_items)]]))==length(obs_items)-1

table(df2011$completely_missing)
```

```
##
```

```
## FALSE  TRUE
##  2060   259
```

```
dat2011<-df2011 %>%
  filter(!completely_missing)
```

Transform/Reverse code high scores on observed variable to indicate right and TAN positioning

```
reverse_items<-c("y26","y19",
                 "y4","y1","y21",
                 "C1_7","C1_8",
                 "C1_3","C1_5","C1_10","C1_11")

reverse_items %in% names(df2011)
```

```
##  [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
```

```
for (i in 1:length(reverse_items)){
  dat2011[,reverse_items[i]]<-6-dat2011[,reverse_items[i]]
}
```

Analysis

H1 and H2

H1. Left-Right placement as computed from responses to the pre-election public Voting Advice Applications (VAAs) is positively associated with Left-Right placement as computed from responses to the privately administered post-election Candidate Survey (CS). This association is stronger than any associations between the Left-Right and GAL-TAN dimensions.

H2. GAL-TAN placement as computed from responses to the pre-election public Voting Advice Applications (VAAs) is positively associated with GAL-TAN placement as computed from responses to the privately administered post-election Candidate Survey (CS). This association is stronger than any associations between the Left-Right and GAL-TAN dimensions.

Define the model

```
model_H1H2<-"
#loadings
VAA_LR=~y22+y23+y26+y27+y9+y19
VAA_GT=~y4+y5+y1+y21
CS_LR=~C1_2+C1_7+C1_8
CS_GT=~C1_1+C1_3+C1_4+C1_5+C1_6+C1_10+C1_11

#latent correlations

#cross-dimension same-method
VAA_LR~~r.VAA*VAA_GT
CS_LR~~r.CS*CS_GT

#concurrent validity
VAA_LR~~r.LR*CS_LR
VAA_GT~~r.GT*CS_GT

#cross-dimension cross-method correlations
VAA_LR~~r.d1*CS_GT
VAA_GT~~r.d2*CS_LR

#custom parameters
test.H1:=r.LR-max(r.VAA,r.CS,r.d1,r.d2)
test.H2:=r.GT-max(r.VAA,r.CS,r.d1,r.d2)

"
```

Fit the model

```
fit_H1H2<-cfa(model=model_H1H2,
              data=dat2011,
              missing="fiml")
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: covariance matrix of latent variables
##                 is not positive definite;
##                 use lavInspect(fit, "cov.lv") to investigate.
```

Some problems with latent variable covariance structure

```r
lavInspect(fit_H1H2, "cov.lv")
```

```
##        VAA_LR VAA_GT CS_LR CS_GT
## VAA_LR 1.022
## VAA_GT 0.551  1.468
## CS_LR  0.211  0.040  0.075
## CS_GT  0.266  0.723  0.022 0.282
```

```r
#examine standardized estimates
std.est_H1H2<-standardizedsolution(fit_H1H2)
std.est_H1H2[std.est_H1H2$op=="~~" &
               std.est_H1H2$lhs!=std.est_H1H2$rhs,]
```

```
##       lhs op    rhs est.std    se      z pvalue ci.lower ci.upper
## 21 VAA_LR ~~ VAA_GT   0.450 0.026 17.135  0.000    0.399    0.502
## 22  CS_LR ~~  CS_GT   0.149 0.044  3.411  0.001    0.063    0.235
## 23 VAA_LR ~~  CS_LR   0.761 0.037 20.300  0.000    0.687    0.834
## 24 VAA_GT ~~  CS_GT   1.125 0.015 76.454  0.000    1.096    1.153
## 25 VAA_LR ~~  CS_GT   0.496 0.031 15.980  0.000    0.435    0.557
## 26 VAA_GT ~~  CS_LR   0.121 0.042  2.876  0.004    0.038    0.203
```

There is an impossible correlation between GAL-TAN factors (absolute value > 1)

```r
model_H1H2.re<-paste0(model_H1H2,
                      "y4~~C1_4\n")
```

Respecify the model by introducing the preregistered residual correlation

```r
fit_H1H2.re<-cfa(model=model_H1H2.re,
              data=dat2011,
              missing="fiml")
```

Fit the respecified model

```
## Warning in lav_object_post_check(object): lavaan WARNING: covariance matrix of latent variables
##                 is not positive definite;
##                 use lavInspect(fit, "cov.lv") to investigate.
```

The problem persists, inspect the parameter estimates

```r
summary(fit_H1H2.re,fit=T,standardized=T,rsquare=T)
```

```
## lavaan 0.6-5 ended normally after 82 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of free parameters                         67
##
##   Number of observations                          2060
##   Number of missing patterns                        63
##
## Model Test User Model:
##
##   Test statistic                              1811.451
##   Degrees of freedom                               163
##   P-value (Chi-square)                           0.000
##
## Model Test Baseline Model:
##
##   Test statistic                              8365.164
##   Degrees of freedom                               190
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.798
##   Tucker-Lewis Index (TLI)                       0.765
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)             -43099.896
##   Loglikelihood unrestricted model (H1)     -42194.171
##
##   Akaike (AIC)                               86333.792
##   Bayesian (BIC)                             86711.033
##   Sample-size adjusted Bayesian (BIC)        86498.168
```

```
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.070
##   90 Percent confidence interval - lower         0.067
##   90 Percent confidence interval - upper         0.073
##   P-value RMSEA <= 0.05                          0.000
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.096
##
## Parameter Estimates:
##
##   Information                                 Observed
##   Observed information based on                Hessian
##   Standard errors                             Standard
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR =~
##     y22               1.000                               1.008    0.692
##     y23               0.782    0.036   21.953    0.000    0.788    0.605
##     y26               0.633    0.031   20.614    0.000    0.639    0.569
##     y27               0.609    0.034   18.009    0.000    0.614    0.473
##     y9                0.902    0.037   24.689    0.000    0.910    0.670
##     y19               0.851    0.035   24.315    0.000    0.858    0.651
##   VAA_GT =~
##     y4                1.000                               1.055    0.660
##     y5                1.141    0.050   22.918    0.000    1.204    0.797
##     y1                0.427    0.034   12.675    0.000    0.451    0.338
##     y21               0.770    0.037   20.556    0.000    0.813    0.575
##   CS_LR =~
##     C1_2              1.000                               0.274    0.249
##     C1_7              1.600    0.264    6.061    0.000    0.438    0.487
##     C1_8              3.240    0.537    6.031    0.000    0.887    0.854
##   CS_GT =~
##     C1_1              1.000                               0.616    0.645
##     C1_3              0.755    0.068   11.030    0.000    0.465    0.432
##     C1_4              1.575    0.100   15.728    0.000    0.970    0.616
##     C1_5              0.414    0.072    5.766    0.000    0.255    0.215
##     C1_6              0.985    0.070   14.086    0.000    0.607    0.535
##     C1_10             1.203    0.080   15.112    0.000    0.741    0.612
##     C1_11             0.619    0.066    9.398    0.000    0.382    0.366
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR ~~
##     VAA_GT  (r.VA)    0.480    0.038   12.519    0.000    0.451    0.451
##   CS_LR ~~
##     CS_GT   (r.CS)    0.024    0.009    2.723    0.006    0.143    0.143
##   VAA_LR ~~
##     CS_LR   (r.LR)    0.210    0.037    5.734    0.000    0.762    0.762
##   VAA_GT ~~
```

```
##     CS_GT   (r.GT)    0.662    0.041   16.077    0.000    1.019    1.019
##   VAA_LR ~~
##     CS_GT   (r.d1)    0.296    0.028   10.732    0.000    0.476    0.476
##   VAA_GT ~~
##     CS_LR   (r.d2)    0.040    0.014    2.787    0.005    0.138    0.138
##  .y4 ~~
##    .C1_4              0.942    0.065   14.528    0.000    0.942    0.632
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               2.639    0.033   79.332    0.000    2.639    1.810
##    .y23               2.178    0.030   73.067    0.000    2.178    1.672
##    .y26               1.798    0.026   70.198    0.000    1.798    1.602
##    .y27               3.636    0.030  122.654    0.000    3.636    2.803
##    .y9                3.468    0.031  112.184    0.000    3.468    2.555
##    .y19               2.132    0.030   71.118    0.000    2.132    1.619
##    .y4                2.442    0.036   68.064    0.000    2.442    1.527
##    .y5                3.101    0.034   91.051    0.000    3.101    2.053
##    .y1                3.592    0.030  118.219    0.000    3.592    2.697
##    .y21               2.608    0.033   79.891    0.000    2.608    1.847
##    .C1_2              1.887    0.036   51.940    0.000    1.887    1.717
##    .C1_7              1.896    0.029   65.285    0.000    1.896    2.109
##    .C1_8              1.864    0.031   59.548    0.000    1.864    1.796
##    .C1_1              4.021    0.029  139.603    0.000    4.021    4.209
##    .C1_3              1.947    0.034   56.660    0.000    1.947    1.806
##    .C1_4              2.261    0.043   53.040    0.000    2.261    1.435
##    .C1_5              3.400    0.039   86.610    0.000    3.400    2.860
##    .C1_6              3.593    0.035  101.591    0.000    3.593    3.168
##    .C1_10             2.472    0.037   67.049    0.000    2.472    2.041
##    .C1_11             1.643    0.034   48.693    0.000    1.643    1.577
##     VAA_LR            0.000                               0.000    0.000
##     VAA_GT            0.000                               0.000    0.000
##     CS_LR             0.000                               0.000    0.000
##     CS_GT             0.000                               0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               1.109    0.046   23.949    0.000    1.109    0.522
##    .y23               1.076    0.041   26.186    0.000    1.076    0.634
##    .y26               0.853    0.032   26.911    0.000    0.853    0.676
##    .y27               1.306    0.046   28.646    0.000    1.306    0.776
##    .y9                1.015    0.041   24.566    0.000    1.015    0.551
##    .y19               0.998    0.039   25.406    0.000    0.998    0.576
##    .y4                1.444    0.062   23.128    0.000    1.444    0.565
##    .y5                0.831    0.053   15.631    0.000    0.831    0.364
##    .y1                1.571    0.053   29.720    0.000    1.571    0.885
##    .y21               1.334    0.051   26.063    0.000    1.334    0.669
##    .C1_2              1.133    0.055   20.732    0.000    1.133    0.938
##    .C1_7              0.617    0.032   19.033    0.000    0.617    0.763
##    .C1_8              0.292    0.063    4.647    0.000    0.292    0.271
##    .C1_1              0.533    0.030   17.782    0.000    0.533    0.584
##    .C1_3              0.945    0.047   20.160    0.000    0.945    0.814
##    .C1_4              1.540    0.081   18.976    0.000    1.540    0.621
##    .C1_5              1.348    0.064   20.965    0.000    1.348    0.954
```

```
##    .C1_6              0.918    0.048   19.243    0.000    0.918    0.714
##    .C1_10             0.917    0.050   18.166    0.000    0.917    0.625
##    .C1_11             0.940    0.046   20.362    0.000    0.940    0.866
##     VAA_LR            1.016    0.065   15.607    0.000    1.000    1.000
##     VAA_GT            1.114    0.079   14.129    0.000    1.000    1.000
##     CS_LR             0.075    0.023    3.267    0.001    1.000    1.000
##     CS_GT             0.380    0.037   10.265    0.000    1.000    1.000
##
## R-Square:
##                    Estimate
##     y22               0.478
##     y23               0.366
##     y26               0.324
##     y27               0.224
##     y9                0.449
##     y19               0.424
##     y4                0.435
##     y5                0.636
##     y1                0.115
##     y21               0.331
##     C1_2              0.062
##     C1_7              0.237
##     C1_8              0.729
##     C1_1              0.416
##     C1_3              0.186
##     C1_4              0.379
##     C1_5              0.046
##     C1_6              0.286
##     C1_10             0.375
##     C1_11             0.134
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##     test.H1          -0.269    0.051   -5.336    0.000    0.286    0.286
##     test.H2           0.183    0.044    4.154    0.000    0.543    0.543
```

Inspect fit of the model

```r
round(inspect(fit_H1H2,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##     npar       df    chisq   pvalue      cfi      tli    rmsea     srmr
##   66.000  164.000 2057.729    0.000    0.768    0.732    0.075    0.097
```

```r
round(inspect(fit_H1H2.re,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##     npar       df    chisq   pvalue      cfi      tli    rmsea     srmr
##   67.000  163.000 1811.451    0.000    0.798    0.765    0.070    0.096
```

The fits of the models are poor.

Hypotheses 1 and 2

Print standardized estimates to test the difference between correlations

```r
std.est_H1H2<-standardizedsolution(fit_H1H2.re)
std.est_H1H2[std.est_H1H2$op==":=" |
```

```r
              std.est_H1H2$op=="~~" &
              std.est_H1H2$lhs!=std.est_H1H2$rhs,]
```

```
##        lhs op                           rhs est.std    se      z pvalue
## 21  VAA_LR ~~                        VAA_GT   0.451 0.026 17.600  0.000
## 22   CS_LR ~~                         CS_GT   0.143 0.044  3.272  0.001
## 23  VAA_LR ~~                         CS_LR   0.762 0.038 20.291  0.000
## 24  VAA_GT ~~                         CS_GT   1.019 0.015 69.261  0.000
## 25  VAA_LR ~~                         CS_GT   0.476 0.032 15.000  0.000
## 26  VAA_GT ~~                         CS_LR   0.138 0.041  3.350  0.001
## 27      y4 ~~                          C1_4   0.632 0.025 25.047  0.000
## 76 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)   0.286 0.051  5.656  0.000
## 77 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.543 0.035 15.680  0.000
##    ci.lower ci.upper
## 21    0.401    0.501
## 22    0.057    0.228
## 23    0.688    0.836
## 24    0.990    1.048
## 25    0.414    0.538
## 26    0.057    0.219
## 27    0.582    0.681
## 76    0.187    0.385
## 77    0.475    0.610
```

```
mis_H1H2<-miPowerFit(fit_H1H2.re,stdLoad=.40,cor=.20)
```

Exploratory analysis for H1 and H2: Seek misspecification to improve the overall model fit

```
## Warning in lav_start_check_cov(lavpartable = lavpartable, start = START): lavaan WARNING: starting va
##                        variables involved are:  VAA_GT   CS_GT
```

```
mis_H1H2<-mis_H1H2[mis_H1H2$op=="=~" | mis_H1H2$op=="~~",]
#see summary of the decisions
table(mis_H1H2$decision.pow)
```

```
##
##  EPC:M EPC:NM      M     NM
##      7    111      2    131
```

```
#there are 9 misspecifications

rounded.vars<-c("mi","epc","target.epc",
                "std.epc","se.epc")

num.round<-function(var){
  var<-as.numeric(var)
  var<-round(var,2)
  return(var)
}

mis_H1H2[,rounded.vars]<-sapply(mis_H1H2[,rounded.vars],num.round)

printed.vars<-c("lhs","op","rhs","mi","epc","target.epc",
                "std.epc","std.target.epc","significant.mi",
                "high.power","decision.pow","se.epc")

#print the output

mis_H1H2 %>%
  filter(mis_H1H2$decision.pow=="M" |
              mis_H1H2$decision.pow=="EPC:M") %>%
  dplyr::select(all_of(printed.vars))
```

```
##      lhs op   rhs     mi   epc target.epc std.epc std.target.epc significant.mi
## 1 VAA_LR =~    y1 401.16 -0.76       0.53   -0.58            0.4           TRUE
## 2 VAA_LR =~  C1_2  50.46  0.58       0.44    0.53            0.4           TRUE
## 3 VAA_LR =~  C1_8  24.27 -0.88       0.41   -0.85            0.4           TRUE
## 4  CS_LR =~   y26 136.95  2.92       1.64    0.71            0.4           TRUE
## 5  CS_LR =~    y9  59.26 -2.22       1.98   -0.45            0.4           TRUE
## 6  CS_LR =~    y1 366.17 -2.72       1.95   -0.56            0.4           TRUE
## 7  CS_GT =~    y1  17.43 -1.06       0.86   -0.49            0.4           TRUE
## 8     y1 ~~ C1_10  76.05  0.41       0.32    0.25            0.2           TRUE
## 9   C1_7 ~~  C1_8   5.52  0.17       0.19    0.18            0.2           TRUE
##   high.power decision.pow se.epc
## 1       TRUE        EPC:M   0.04
## 2       TRUE        EPC:M   0.08
## 3      FALSE            M   0.18
## 4       TRUE        EPC:M   0.25
## 5       TRUE        EPC:M   0.29
```

```
## 6       TRUE        EPC:M   0.14
## 7       TRUE        EPC:M   0.25
## 8       TRUE        EPC:M   0.05
## 9      FALSE            M   0.07
```

Item y1 "Finland should continue to financially assist EU countries that are facing economic hardship (r.)" is proposed to load to all other factors besides loading on its specified factor (VAA_GT). It is also proposed to have residual correlation with C1_10 ("Maahanmuutto on hyvä asia Suomen taloudelle"). Exclude item y1 entirely

```
model_H1H2.exp.re<-"
#loadings
VAA_LR=~y22+y23+y26+y27+y9+y19
VAA_GT=~y4+y5+y21
CS_LR=~C1_2+C1_7+C1_8
CS_GT=~C1_1+C1_3+C1_4+C1_5+C1_6+C1_10+C1_11

#latent correlations

#cross-dimension same-method
VAA_LR~~r.VAA*VAA_GT
CS_LR~~r.CS*CS_GT

#concurrent validity
VAA_LR~~r.LR*CS_LR
VAA_GT~~r.GT*CS_GT

#cross-dimension cross-method correlations
VAA_LR~~r.d1*CS_GT
VAA_GT~~r.d2*CS_LR

#custom parameters
test.H1:=r.LR-max(r.VAA,r.CS,r.d1,r.d2)
test.H2:=r.GT-max(r.VAA,r.CS,r.d1,r.d2)


y4~~C1_4
"
```

```
fit_H1H2.exp.re<-cfa(model=model_H1H2.exp.re,
              data=dat2011,
              missing="fiml")
```

Exploratory respecification

```
## Warning in lav_data_full(data = data, group = group, cluster = cluster, : lavaan WARNING: some cases
##   573
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: covariance matrix of latent variables
##                 is not positive definite;
##                 use lavInspect(fit, "cov.lv") to investigate.
```

Problems are still there

Inspect fit of the model

```r
round(inspect(fit_H1H2.re,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##     npar       df    chisq   pvalue      cfi      tli    rmsea     srmr
##   67.000  163.000 1811.451    0.000    0.798    0.765    0.070    0.096
```

```r
round(inspect(fit_H1H2.exp.re,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##     npar       df    chisq   pvalue      cfi      tli    rmsea     srmr
##   64.000  145.000 1292.194    0.000    0.847    0.820    0.062    0.084
```

The fit of the model is improved by removal of one item (these are not really comparable, because of non-nested modeling).

Retest Hypotheses 1 and 2

Print standardized estimates to test the difference between correlations

```r
std.est_H1H2.exp<-standardizedsolution(fit_H1H2.exp.re)
std.est_H1H2.exp[std.est_H1H2.exp$op==":=" |
               std.est_H1H2.exp$op=="~~" &
               std.est_H1H2.exp$lhs!=std.est_H1H2.exp$rhs,]
```

```
##         lhs op                           rhs est.std    se      z pvalue
## 20   VAA_LR ~~                        VAA_GT   0.508 0.025 20.740      0
## 21    CS_LR ~~                         CS_GT   0.154 0.044  3.514      0
## 22   VAA_LR ~~                         CS_LR   0.763 0.038 20.337      0
## 23   VAA_GT ~~                         CS_GT   1.024 0.015 66.834      0
## 24   VAA_LR ~~                         CS_GT   0.490 0.032 15.501      0
## 25   VAA_GT ~~                         CS_LR   0.187 0.042  4.477      0
## 26       y4 ~~                          C1_4   0.630 0.025 24.785      0
## 73 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)   0.254 0.046  5.563      0
## 74 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.515 0.028 18.150      0
##    ci.lower ci.upper
## 20    0.460    0.557
## 21    0.068    0.240
## 22    0.689    0.836
## 23    0.994    1.054
## 24    0.428    0.552
## 25    0.105    0.269
## 26    0.580    0.680
## 73    0.165    0.344
## 74    0.459    0.571
```

```
mis.H1H2<-miPowerFit(fit_H1H2.exp.re,stdLoad=.40,cor=.20)
```

Seek for additional misspecifications

```
## Warning in lav_start_check_cov(lavpartable = lavpartable, start = START): lavaan WARNING: starting va
##                      variables involved are:  VAA_GT   CS_GT
```

```
mis.H1H2<-mis.H1H2[mis.H1H2$op=="=~" |
                     (mis.H1H2$op=="~~" &
                        mis.H1H2$lhs!=mis.H1H2$rhs),]
#see summary of the decisions
table(mis.H1H2$decision.pow)
```

```
##
##  EPC:M EPC:NM      M     NM
##      3    102      2    122
```

```
#there are several misspecifications

rounded.vars<-c("mi","epc","target.epc",
                "std.epc","se.epc")

num.round<-function(var){
  var<-as.numeric(var)
  var<-round(var,2)
  return(var)
}

mis.H1H2[,rounded.vars]<-sapply(mis.H1H2[,rounded.vars],num.round)

printed.vars<-c("lhs","op","rhs","mi","epc","target.epc",
                "std.epc","std.target.epc","significant.mi",
                "high.power","decision.pow","se.epc")

#print the output

mis.H1H2 %>%
  filter(mis.H1H2$decision.pow=="M" |
               mis.H1H2$decision.pow=="EPC:M") %>%
  dplyr::select(all_of(printed.vars))
```

```
##      lhs op  rhs     mi   epc target.epc std.epc std.target.epc significant.mi
## 1 VAA_LR =~ C1_2  50.53  0.58       0.44    0.53            0.4           TRUE
## 2 VAA_LR =~ C1_8  23.69 -0.86       0.41   -0.84            0.4           TRUE
## 3  CS_LR =~  y26 135.63  2.91       1.63    0.71            0.4           TRUE
## 4  CS_LR =~   y9  59.16 -2.22       1.97   -0.45            0.4           TRUE
## 5   C1_7 ~~ C1_8   5.97  0.17       0.19    0.18            0.2           TRUE
##   high.power decision.pow se.epc
## 1       TRUE        EPC:M   0.08
## 2      FALSE            M   0.18
## 3       TRUE        EPC:M   0.25
## 4       TRUE        EPC:M   0.29
## 5      FALSE            M   0.07
```

item y26 is indicated to cross-lead quite strongly, it is removed as well.

```
model_H1H2.exp.re.2<-"
#loadings
VAA_LR=~y22+y23+y27+y9+y19
VAA_GT=~y4+y5+y21
CS_LR=~C1_2+C1_7+C1_8
CS_GT=~C1_1+C1_3+C1_4+C1_5+C1_6+C1_10+C1_11

#latent correlations

#cross-dimension same-method
VAA_LR~~r.VAA*VAA_GT
CS_LR~~r.CS*CS_GT

#concurrent validity
VAA_LR~~r.LR*CS_LR
VAA_GT~~r.GT*CS_GT

#cross-dimension cross-method correlations
VAA_LR~~r.d1*CS_GT
VAA_GT~~r.d2*CS_LR

#custom parameters
test.H1:=r.LR-max(r.VAA,r.CS,r.d1,r.d2)
test.H2:=r.GT-max(r.VAA,r.CS,r.d1,r.d2)


y4~~C1_4
"
```

```
fit_H1H2.exp.re.2<-cfa(model=model_H1H2.exp.re.2,
              data=dat2011,
              missing="fiml")
```

```
## Warning in lav_data_full(data = data, group = group, cluster = cluster, : lavaan WARNING: some cases
##   573
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: covariance matrix of latent variables
##                 is not positive definite;
##                 use lavInspect(fit, "cov.lv") to investigate.
```

```
round(inspect(fit_H1H2.exp.re,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##     npar       df    chisq   pvalue      cfi      tli    rmsea     srmr
##   64.000  145.000 1292.194    0.000    0.847    0.820    0.062    0.084
```

```
round(inspect(fit_H1H2.exp.re.2,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##     npar       df    chisq   pvalue      cfi      tli    rmsea     srmr
##   61.000  128.000 1017.272    0.000    0.867    0.841    0.058    0.079
```

Repeat the misspecification identification again

```
mis.H1H2<-miPowerFit(fit_H1H2.exp.re.2,stdLoad=.40,cor=.20)
```

```
## Warning in lav_start_check_cov(lavpartable = lavpartable, start = START): lavaan WARNING: starting va
```

```
##                      variables involved are:  VAA_GT   CS_GT
```

```
mis.H1H2<-mis.H1H2[mis.H1H2$op=="=~" |
                     (mis.H1H2$op=="~~" &
                        mis.H1H2$lhs!=mis.H1H2$rhs),]
#see summary of the decisions
table(mis.H1H2$decision.pow)
```

```
##
##  EPC:M EPC:NM      M     NM
##      4     86      2    116
```

```
#there are several misspecifications

rounded.vars<-c("mi","epc","target.epc",
                "std.epc","se.epc")

num.round<-function(var){
  var<-as.numeric(var)
  var<-round(var,2)
  return(var)
}

mis.H1H2[,rounded.vars]<-sapply(mis.H1H2[,rounded.vars],num.round)

printed.vars<-c("lhs","op","rhs","mi","epc","target.epc",
                "std.epc","std.target.epc","significant.mi",
                "high.power","decision.pow","se.epc")

#print the output

mis.H1H2 %>%
  filter(mis.H1H2$decision.pow=="M" |
               mis.H1H2$decision.pow=="EPC:M") %>%
  dplyr::select(all_of(printed.vars))
```

```
##      lhs op   rhs    mi   epc target.epc std.epc std.target.epc significant.mi
## 1 VAA_LR =~  C1_2 47.97  0.49       0.43    0.46            0.4           TRUE
## 2 VAA_LR =~  C1_8 22.69 -0.70       0.40   -0.70            0.4           TRUE
## 3 VAA_LR =~ C1_10 98.93 -0.49       0.47   -0.42            0.4           TRUE
## 4  CS_LR =~   y23 55.61  2.00       1.89    0.42            0.4           TRUE
## 5     y5 ~~ C1_10 71.52  0.37       0.37    0.20            0.2           TRUE
## 6   C1_7 ~~  C1_8  5.31  0.18       0.19    0.20            0.2           TRUE
##   high.power decision.pow se.epc
## 1       TRUE        EPC:M   0.07
## 2      FALSE            M   0.15
## 3       TRUE        EPC:M   0.05
## 4       TRUE        EPC:M   0.27
## 5       TRUE        EPC:M   0.04
## 6      FALSE            M   0.08
```

cross-loading and residual correlations are suggested for C1_10 (Maahanmuutto on hyvä asia Suomen taloudelle). Add the residual correlation with y5 (Tax funds should not be used in the current extent for taking in immigrants)

```r
model_H1H2.exp.re.3<-"
#loadings
VAA_LR=~y22+y23+y27+y9+y19
VAA_GT=~y4+y5+y21
CS_LR=~C1_2+C1_7+C1_8
CS_GT=~C1_1+C1_3+C1_4+C1_5+C1_6+C1_10+C1_11

#latent correlations

#cross-dimension same-method
VAA_LR~~r.VAA*VAA_GT
CS_LR~~r.CS*CS_GT

#concurrent validity
VAA_LR~~r.LR*CS_LR
VAA_GT~~r.GT*CS_GT

#cross-dimension cross-method correlations
VAA_LR~~r.d1*CS_GT
VAA_GT~~r.d2*CS_LR

#custom parameters
test.H1:=r.LR-max(r.VAA,r.CS,r.d1,r.d2)
test.H2:=r.GT-max(r.VAA,r.CS,r.d1,r.d2)


y4~~C1_4
y5~~C1_10

"
```

```r
fit_H1H2.exp.re.3<-cfa(model=model_H1H2.exp.re.3,
              data=dat2011,
              missing="fiml")
```

```
## Warning in lav_data_full(data = data, group = group, cluster = cluster, : lavaan WARNING: some cases
##   573
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: covariance matrix of latent variables
##                 is not positive definite;
##                 use lavInspect(fit, "cov.lv") to investigate.
```

Problem still persists

```r
round(inspect(fit_H1H2.exp.re.2,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##     npar       df    chisq   pvalue      cfi      tli    rmsea     srmr
##   61.000  128.000 1017.272    0.000    0.867    0.841    0.058    0.079
```

```r
round(inspect(fit_H1H2.exp.re.3,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  62.000 127.000 948.797   0.000   0.877   0.852   0.056   0.076
```

Repeat the misspecification identification again

```r
mis.H1H2<-miPowerFit(fit_H1H2.exp.re.3,stdLoad=.40,cor=.20)
mis.H1H2<-mis.H1H2[mis.H1H2$op=="=~" |
                     (mis.H1H2$op=="~~" &
                        mis.H1H2$lhs!=mis.H1H2$rhs),]
#see summary of the decisions
table(mis.H1H2$decision.pow)
```

```
##
##  EPC:M EPC:NM      I      M     NM
##      2     85      6      6    106
```

```r
#there are several misspecifications

rounded.vars<-c("mi","epc","target.epc",
                "std.epc","se.epc")

num.round<-function(var){
  var<-as.numeric(var)
  var<-round(var,2)
  return(var)
}

mis.H1H2[,rounded.vars]<-sapply(mis.H1H2[,rounded.vars],num.round)

printed.vars<-c("lhs","op","rhs","mi","epc","target.epc",
                "std.epc","std.target.epc","significant.mi",
                "high.power","decision.pow","se.epc")

#print the output

mis.H1H2 %>%
  filter(mis.H1H2$decision.pow=="M" |
               mis.H1H2$decision.pow=="EPC:M") %>%
  dplyr::select(all_of(printed.vars))
```

```
##      lhs op   rhs    mi   epc target.epc std.epc std.target.epc significant.mi
## 1 VAA_LR =~  C1_2 45.80  0.48       0.43    0.45            0.4           TRUE
## 2 VAA_LR =~  C1_8 20.55 -0.66       0.40   -0.66            0.4           TRUE
## 3 VAA_GT =~  C1_3 27.06  3.67       0.40    3.66            0.4           TRUE
## 4 VAA_GT =~  C1_5 58.42  6.24       0.44    5.64            0.4           TRUE
## 5 VAA_GT =~ C1_10 64.63 -5.92       0.45   -5.26            0.4           TRUE
## 6  CS_LR =~   y23 56.23  2.02       1.88    0.43            0.4           TRUE
## 7  CS_GT =~    y5  8.61 -3.84       0.97   -1.58            0.4           TRUE
## 8   C1_7 ~~  C1_8  5.37  0.18       0.19    0.19            0.2           TRUE
##   high.power decision.pow se.epc
## 1       TRUE        EPC:M   0.07
## 2      FALSE            M   0.15
## 3      FALSE            M   0.71
## 4      FALSE            M   0.82
## 5      FALSE            M   0.74
## 6       TRUE        EPC:M   0.27
## 7      FALSE            M   1.31
## 8      FALSE            M   0.08
```

```r
summary(fit_H1H2.exp.re.3,fit=T,standardized=T)
```

```
## lavaan 0.6-5 ended normally after 84 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of free parameters                         62
##
##                                                   Used       Total
##   Number of observations                          2059        2060
##   Number of missing patterns                        57
##
## Model Test User Model:
##
##   Test statistic                               948.797
##   Degrees of freedom                               127
##   P-value (Chi-square)                           0.000
##
## Model Test Baseline Model:
##
##   Test statistic                              6849.302
##   Degrees of freedom                               153
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.877
##   Tucker-Lewis Index (TLI)                       0.852
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)             -37252.454
##   Loglikelihood unrestricted model (H1)     -36778.055
##
##   Akaike (AIC)                               74628.908
##   Bayesian (BIC)                             74977.966
##   Sample-size adjusted Bayesian (BIC)        74780.987
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.056
##   90 Percent confidence interval - lower         0.053
##   90 Percent confidence interval - upper         0.059
##   P-value RMSEA <= 0.05                          0.001
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.076
##
## Parameter Estimates:
##
##   Information                                 Observed
##   Observed information based on                Hessian
##   Standard errors                             Standard
```

```
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR =~
##     y22               1.000                               1.033    0.708
##     y23               0.705    0.034   20.773    0.000    0.728    0.558
##     y27               0.613    0.034   18.270    0.000    0.633    0.488
##     y9                0.907    0.037   24.726    0.000    0.936    0.690
##     y19               0.824    0.034   24.099    0.000    0.850    0.646
##   VAA_GT =~
##     y4                1.000                               1.075    0.672
##     y5                1.063    0.047   22.795    0.000    1.142    0.755
##     y21               0.766    0.038   20.423    0.000    0.823    0.583
##   CS_LR =~
##     C1_2              1.000                               0.277    0.252
##     C1_7              1.615    0.268    6.026    0.000    0.447    0.498
##     C1_8              3.133    0.528    5.937    0.000    0.868    0.836
##   CS_GT =~
##     C1_1              1.000                               0.624    0.653
##     C1_3              0.746    0.070   10.706    0.000    0.466    0.432
##     C1_4              1.614    0.105   15.403    0.000    1.007    0.639
##     C1_5              0.412    0.072    5.688    0.000    0.257    0.216
##     C1_6              1.014    0.071   14.316    0.000    0.633    0.558
##     C1_10             1.010    0.077   13.164    0.000    0.630    0.521
##     C1_11             0.654    0.068    9.626    0.000    0.408    0.392
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR ~~
##     VAA_GT  (r.VA)    0.650    0.043   15.095    0.000    0.586    0.586
##   CS_LR ~~
##     CS_GT   (r.CS)    0.029    0.010    2.950    0.003    0.166    0.166
##   VAA_LR ~~
##     CS_LR   (r.LR)    0.204    0.036    5.616    0.000    0.714    0.714
##   VAA_GT ~~
##     CS_GT   (r.GT)    0.670    0.042   16.001    0.000    1.000    1.000
##   VAA_LR ~~
##     CS_GT   (r.d1)    0.353    0.030   11.841    0.000    0.548    0.548
##   VAA_GT ~~
##     CS_LR   (r.d2)    0.064    0.018    3.604    0.000    0.214    0.214
##  .y4 ~~
##    .C1_4              0.898    0.065   13.882    0.000    0.898    0.627
##  .y5 ~~
##    .C1_10             0.376    0.047    7.946    0.000    0.376    0.367
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               2.641    0.033   79.395    0.000    2.641    1.812
##    .y23               2.180    0.030   72.987    0.000    2.180    1.672
##    .y27               3.638    0.030  122.720    0.000    3.638    2.804
##    .y9                3.471    0.031  112.286    0.000    3.471    2.557
##    .y19               2.135    0.030   71.158    0.000    2.135    1.621
##    .y4                2.440    0.036   68.014    0.000    2.440    1.527
##    .y5                3.102    0.034   91.052    0.000    3.102    2.050
```

```
##    .y21              2.606    0.033   79.835    0.000    2.606    1.846
##    .C1_2             1.884    0.036   51.807    0.000    1.884    1.714
##    .C1_7             1.892    0.029   64.895    0.000    1.892    2.104
##    .C1_8             1.854    0.032   58.020    0.000    1.854    1.785
##    .C1_1             4.019    0.029  138.703    0.000    4.019    4.206
##    .C1_3             1.945    0.034   56.426    0.000    1.945    1.805
##    .C1_4             2.261    0.043   53.015    0.000    2.261    1.436
##    .C1_5             3.399    0.039   86.530    0.000    3.399    2.859
##    .C1_6             3.590    0.035  101.565    0.000    3.590    3.166
##    .C1_10            2.471    0.036   68.447    0.000    2.471    2.044
##    .C1_11            1.642    0.034   48.757    0.000    1.642    1.576
##     VAA_LR           0.000                               0.000    0.000
##     VAA_GT           0.000                               0.000    0.000
##     CS_LR            0.000                               0.000    0.000
##     CS_GT            0.000                               0.000    0.000
##
## Variances:
##                  Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22              1.060    0.047   22.708    0.000    1.060    0.498
##    .y23              1.170    0.043   27.003    0.000    1.170    0.688
##    .y27              1.283    0.045   28.287    0.000    1.283    0.762
##    .y9               0.966    0.041   23.378    0.000    0.966    0.524
##    .y19              1.011    0.040   25.033    0.000    1.011    0.583
##    .y4               1.400    0.062   22.429    0.000    1.400    0.548
##    .y5               0.985    0.054   18.208    0.000    0.985    0.430
##    .y21              1.316    0.052   25.544    0.000    1.316    0.660
##    .C1_2             1.131    0.055   20.655    0.000    1.131    0.936
##    .C1_7             0.608    0.033   18.396    0.000    0.608    0.752
##    .C1_8             0.326    0.067    4.893    0.000    0.326    0.302
##    .C1_1             0.524    0.031   16.901    0.000    0.524    0.574
##    .C1_3             0.945    0.048   19.894    0.000    0.945    0.813
##    .C1_4             1.465    0.082   17.916    0.000    1.465    0.591
##    .C1_5             1.347    0.064   20.914    0.000    1.347    0.953
##    .C1_6             0.886    0.048   18.600    0.000    0.886    0.689
##    .C1_10            1.065    0.056   18.866    0.000    1.065    0.728
##    .C1_11            0.919    0.046   20.026    0.000    0.919    0.847
##     VAA_LR           1.066    0.067   15.884    0.000    1.000    1.000
##     VAA_GT           1.155    0.080   14.357    0.000    1.000    1.000
##     CS_LR            0.077    0.024    3.249    0.001    1.000    1.000
##     CS_GT            0.389    0.039   10.076    0.000    1.000    1.000
##
## Defined Parameters:
##                  Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##     test.H1         -0.446    0.053   -8.375    0.000    0.128    0.128
##     test.H2          0.020    0.046    0.440    0.660    0.414    0.414
```

Remove the weak loading items C1_2 and C1_5

```
model_H1H2.exp.re.4<-"
#loadings
VAA_LR=~y22+y23+y27+y9+y19
VAA_GT=~y4+y5+y21
CS_LR=~C1_7+C1_8
CS_GT=~C1_1+C1_3+C1_4+C1_6+C1_10+C1_11
```

```
#latent correlations

#cross-dimension same-method
VAA_LR~~r.VAA*VAA_GT
CS_LR~~r.CS*CS_GT

#concurrent validity
VAA_LR~~r.LR*CS_LR
VAA_GT~~r.GT*CS_GT

#cross-dimension cross-method correlations
VAA_LR~~r.d1*CS_GT
VAA_GT~~r.d2*CS_LR

#custom parameters
test.H1:=r.LR-max(r.VAA,r.CS,r.d1,r.d2)
test.H2:=r.GT-max(r.VAA,r.CS,r.d1,r.d2)


y4~~C1_4
y5~~C1_10

"
```

```
fit_H1H2.exp.re.4<-cfa(model=model_H1H2.exp.re.4,
              data=dat2011,
              missing="fiml")
```

```
## Warning in lav_data_full(data = data, group = group, cluster = cluster, : lavaan WARNING: some cases
##   573
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: covariance matrix of latent variables
##                 is not positive definite;
##                 use lavInspect(fit, "cov.lv") to investigate.
```

```
round(inspect(fit_H1H2.exp.re.3,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  62.000 127.000 948.797   0.000   0.877   0.852   0.056   0.076
```

```
round(inspect(fit_H1H2.exp.re.4,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  56.000  96.000 737.799   0.000   0.900   0.875   0.057   0.065
```

```
std.est.H1H2.exp.re.4<-standardizedsolution(fit_H1H2.exp.re.4)

std.est.H1H2.exp.re.4[std.est.H1H2.exp.re.4$op == "=~" |
                        (std.est.H1H2.exp.re.4$op == "~~" &
                           std.est.H1H2.exp.re.4$lhs != std.est.H1H2.exp.re.4$rhs),]
```

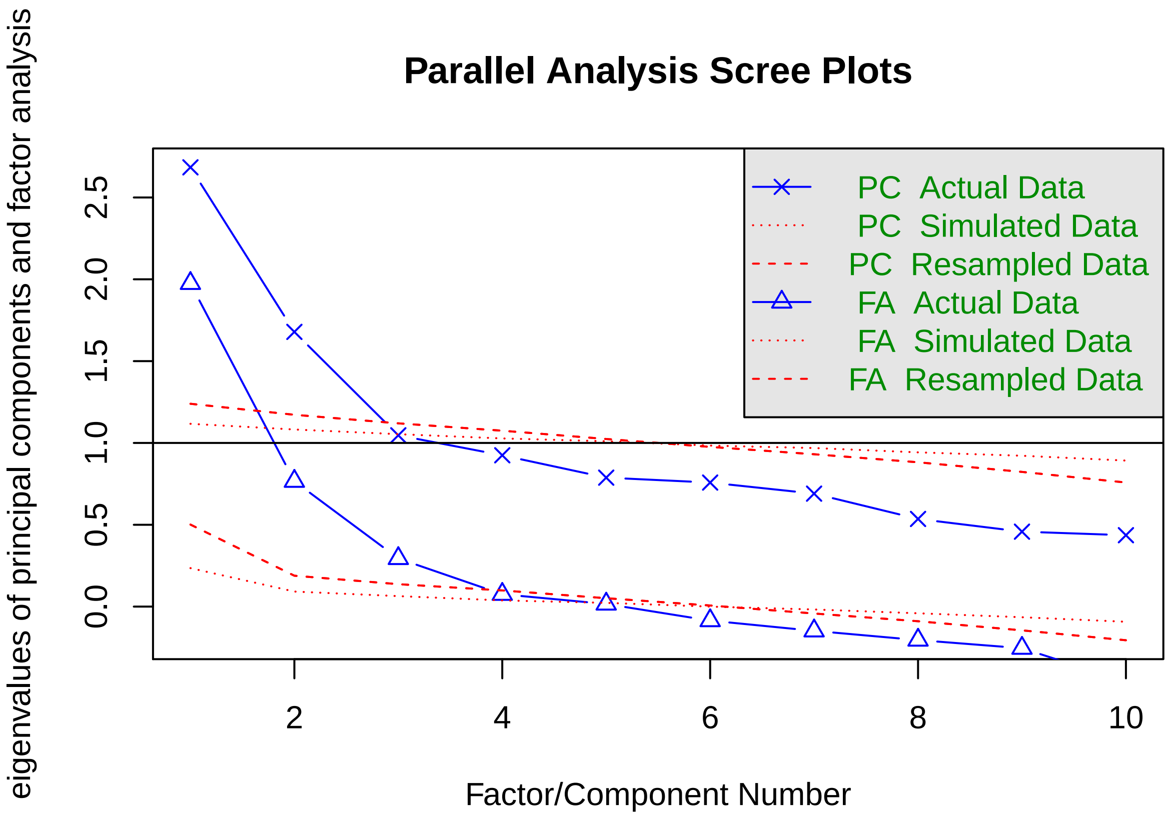
```
##       lhs op    rhs est.std    se      z pvalue ci.lower ci.upper
## 1  VAA_LR =~    y22   0.708 0.015 45.934  0.000    0.678    0.738
## 2  VAA_LR =~    y23   0.558 0.019 29.307  0.000    0.521    0.596
## 3  VAA_LR =~    y27   0.487 0.021 23.717  0.000    0.447    0.527
```

```
## 4  VAA_LR =~     y9   0.690 0.016 43.303  0.000    0.659    0.721
## 5  VAA_LR =~    y19   0.646 0.017 38.197  0.000    0.613    0.679
## 6  VAA_GT =~     y4   0.674 0.018 38.307  0.000    0.640    0.709
## 7  VAA_GT =~     y5   0.755 0.016 47.216  0.000    0.723    0.786
## 8  VAA_GT =~    y21   0.583 0.019 29.961  0.000    0.545    0.621
## 9   CS_LR =~   C1_7   0.499 0.034 14.707  0.000    0.432    0.565
## 10  CS_LR =~   C1_8   0.847 0.040 21.207  0.000    0.768    0.925
## 11  CS_GT =~   C1_1   0.653 0.024 26.843  0.000    0.606    0.701
## 12  CS_GT =~   C1_3   0.416 0.032 13.182  0.000    0.355    0.478
## 13  CS_GT =~   C1_4   0.645 0.023 28.577  0.000    0.601    0.689
## 14  CS_GT =~   C1_6   0.564 0.027 20.691  0.000    0.511    0.618
## 15  CS_GT =~  C1_10   0.522 0.029 18.307  0.000    0.466    0.578
## 16  CS_GT =~  C1_11   0.392 0.033 11.979  0.000    0.328    0.456
## 17 VAA_LR ~~ VAA_GT   0.584 0.024 24.822  0.000    0.538    0.630
## 18  CS_LR ~~  CS_GT   0.107 0.043  2.493  0.013    0.023    0.191
## 19 VAA_LR ~~  CS_LR   0.694 0.040 17.181  0.000    0.614    0.773
## 20 VAA_GT ~~  CS_GT   0.998 0.015 67.587  0.000    0.969    1.027
## 21 VAA_LR ~~  CS_GT   0.538 0.031 17.212  0.000    0.476    0.599
## 22 VAA_GT ~~  CS_LR   0.176 0.042  4.243  0.000    0.095    0.258
## 23     y4 ~~   C1_4   0.624 0.026 23.664  0.000    0.572    0.676
## 24     y5 ~~  C1_10   0.369 0.038  9.718  0.000    0.295    0.444
```

Exploratory factor analysis for CS-data

```r
all_CS_items<-c("C1_2","C1_7","C1_8","C1_1","C1_3","C1_4","C1_5","C1_6","C1_10","C1_11")

#conduct a parallel analysis to explore the number of factors
fa.parallel(dat2011[,all_CS_items])
```




```
## Parallel analysis suggests that the number of factors =  4  and the number of components =  2
```

```r
#four factor solution
fa(dat2011[,all_CS_items],nfactors = 4,fm = "ml",rotate = "oblimin")
```

```
## Loading required namespace: GPArotation
```

```
## Factor Analysis using method =  ml
## Call: fa(r = dat2011[, all_CS_items], nfactors = 4, rotate = "oblimin",
##     fm = "ml")
## Standardized loadings (pattern matrix) based upon correlation matrix
##         ML1   ML3   ML2   ML4   h2    u2 com
## C1_2   0.10  0.26  0.20 -0.03 0.14 0.860 2.3
## C1_7   0.04 -0.12  0.34  0.30 0.25 0.745 2.3
## C1_8   0.00  0.01  1.00  0.01 1.00 0.005 1.0
## C1_1  -0.03  0.74  0.02  0.04 0.55 0.450 1.0
## C1_3   0.14  0.13  0.14  0.34 0.26 0.741 2.1
## C1_4   1.00  0.01  0.00 -0.02 1.00 0.005 1.0
## C1_5  -0.04  0.06  0.15  0.42 0.24 0.760 1.3
## C1_6   0.07  0.62  0.03 -0.11 0.41 0.594 1.1
## C1_10  0.08  0.44 -0.19  0.25 0.37 0.627 2.1
## C1_11  0.47 -0.06 -0.03  0.12 0.23 0.770 1.2
##
##                        ML1  ML3  ML2  ML4
## SS loadings           1.32 1.30 1.27 0.55
## Proportion Var        0.13 0.13 0.13 0.06
## Cumulative Var        0.13 0.26 0.39 0.44
```

```
## Proportion Explained  0.30 0.29 0.29 0.12
## Cumulative Proportion 0.30 0.59 0.88 1.00
##
##  With factor correlations of
##      ML1  ML3  ML2  ML4
## ML1 1.00 0.51 0.03 0.21
## ML3 0.51 1.00 0.02 0.27
## ML2 0.03 0.02 1.00 0.29
## ML4 0.21 0.27 0.29 1.00
##
## Mean item complexity =  1.5
## Test of the hypothesis that 4 factors are sufficient.
##
## The degrees of freedom for the null model are  45  and the objective function was  1.65 with Chi Squa
## The degrees of freedom for the model are 11  and the objective function was  0.02
##
## The root mean square of the residuals (RMSR) is  0.02
## The df corrected root mean square of the residuals is  0.03
##
## The harmonic number of observations is  897 with the empirical chi square  18.22  with prob <  0.077
## The total number of observations was  2060  with Likelihood Chi Square =  42.16  with prob <  1.5e-05
##
## Tucker Lewis Index of factoring reliability =  0.962
## RMSEA index =  0.037  and the 90 % confidence intervals are  0.026 0.049
## BIC =  -41.77
## Fit based upon off diagonal values = 1
## Measures of factor score adequacy
##                                                    ML1  ML3  ML2   ML4
## Correlation of (regression) scores with factors   1.00 0.86 1.00  0.67
## Multiple R square of scores with factors          0.99 0.73 0.99  0.45
## Minimum correlation of possible factor scores     0.99 0.46 0.99 -0.09
```

First factor is defined by “Samaa sukupuolta olevien avioliitot pitäisi kieltää laissa” to which only other substantial loading is from “Naisilla pitäisi olla vapaus päättää aborttikysymyksistä”. So this represents so conservative dimension.

Second factor (ML3) has loadings from immigration items.

Third factor is defined by “Tuloja ja vaurautta pitäisi uudelleenjakaa tavallisten ihmisten suuntaan” with no substantial loadings from other items.

Some of the items also have very weak variance explained, and high communalities

Fourth factor is some sort of mixed factor. This is not an ideal solution to which VAA responses in GT and LR could be compared to.

Try 3 factor solution.

```r
fa(dat2011[,all_CS_items],nfactors = 3,fm = "ml",rotate = "oblimin")
```

```
## Factor Analysis using method =  ml
## Call: fa(r = dat2011[, all_CS_items], nfactors = 3, rotate = "oblimin",
##     fm = "ml")
## Standardized loadings (pattern matrix) based upon correlation matrix
##         ML3   ML1   ML2   h2    u2 com
## C1_2   0.23  0.10  0.20 0.14 0.862 2.3
## C1_7  -0.03  0.03  0.51 0.26 0.736 1.0
## C1_8  -0.02 -0.01  0.83 0.69 0.312 1.0
```

```
## C1_1   0.77 -0.04  0.02 0.57 0.435 1.0
## C1_3   0.23  0.14  0.30 0.21 0.795 2.3
## C1_4   0.00  1.00  0.00 1.00 0.005 1.0
## C1_5   0.19 -0.04  0.34 0.15 0.847 1.6
## C1_6   0.56  0.08 -0.02 0.36 0.639 1.0
## C1_10  0.52  0.09 -0.11 0.33 0.670 1.1
## C1_11 -0.01  0.47  0.02 0.22 0.784 1.0
##
##                        ML3  ML1  ML2
## SS loadings           1.38 1.31 1.22
## Proportion Var        0.14 0.13 0.12
## Cumulative Var        0.14 0.27 0.39
## Proportion Explained  0.35 0.34 0.31
## Cumulative Proportion 0.35 0.69 1.00
##
##  With factor correlations of
##      ML3  ML1  ML2
## ML3 1.00 0.53 0.07
## ML1 0.53 1.00 0.06
## ML2 0.07 0.06 1.00
##
## Mean item complexity =  1.4
## Test of the hypothesis that 3 factors are sufficient.
##
## The degrees of freedom for the null model are  45  and the objective function was  1.65 with Chi Squa
## The degrees of freedom for the model are 18  and the objective function was  0.08
##
## The root mean square of the residuals (RMSR) is  0.03
## The df corrected root mean square of the residuals is  0.05
##
## The harmonic number of observations is  897 with the empirical chi square  78.71  with prob <  1.4e-0
## The total number of observations was  2060  with Likelihood Chi Square =  166.49  with prob <  4.4e-2
##
## Tucker Lewis Index of factoring reliability =  0.889
## RMSEA index =  0.063  and the 90 % confidence intervals are  0.055 0.072
## BIC =  29.14
## Fit based upon off diagonal values = 0.98
## Measures of factor score adequacy
##                                                    ML3  ML1  ML2
## Correlation of (regression) scores with factors   0.86 1.00 0.86
## Multiple R square of scores with factors          0.74 1.00 0.74
## Minimum correlation of possible factor scores     0.48 0.99 0.49
```

First factor (ML3) has loadings from immigration items.

Second (ML1) is a conservativeness factor (factors 1 and 2 also correlate with .50)

Third factor is right-wing factor, although environment and some women's right items seem to weakly load there as well.

See what the two-factor solution looks like

```
fa(dat2011[,all_CS_items],nfactors = 2,fm = "ml",rotate = "oblimin")
```

```
## Factor Analysis using method =  ml
## Call: fa(r = dat2011[, all_CS_items], nfactors = 2, rotate = "oblimin",
##     fm = "ml")
```

```
## Standardized loadings (pattern matrix) based upon correlation matrix
##         ML2   ML1   h2   u2 com
## C1_2   0.31  0.20 0.14 0.86 1.7
## C1_7   0.01  0.50 0.25 0.75 1.0
## C1_8  -0.02  0.86 0.73 0.27 1.0
## C1_1   0.64  0.02 0.41 0.59 1.0
## C1_3   0.35  0.28 0.22 0.78 1.9
## C1_4   0.66  0.01 0.44 0.56 1.0
## C1_5   0.16  0.32 0.14 0.86 1.5
## C1_6   0.58 -0.02 0.33 0.67 1.0
## C1_10  0.59 -0.12 0.35 0.65 1.1
## C1_11  0.39  0.01 0.15 0.85 1.0
##
##                        ML2  ML1
## SS loadings           1.94 1.24
## Proportion Var        0.19 0.12
## Cumulative Var        0.19 0.32
## Proportion Explained  0.61 0.39
## Cumulative Proportion 0.61 1.00
##
##  With factor correlations of
##      ML2  ML1
## ML2 1.00 0.07
## ML1 0.07 1.00
##
## Mean item complexity =  1.2
## Test of the hypothesis that 2 factors are sufficient.
##
## The degrees of freedom for the null model are  45  and the objective function was  1.65 with Chi Squa
## The degrees of freedom for the model are 26  and the objective function was  0.24
##
## The root mean square of the residuals (RMSR) is  0.05
## The df corrected root mean square of the residuals is  0.07
##
## The harmonic number of observations is  897 with the empirical chi square  210.65  with prob <  8e-3
## The total number of observations was  2060  with Likelihood Chi Square =  485.41  with prob <  3.6e-8
##
## Tucker Lewis Index of factoring reliability =  0.762
## RMSEA index =  0.093  and the 90 % confidence intervals are  0.086 0.1
## BIC =  287.02
## Fit based upon off diagonal values = 0.95
## Measures of factor score adequacy
##                                                    ML2  ML1
## Correlation of (regression) scores with factors   0.87 0.88
## Multiple R square of scores with factors          0.75 0.77
## Minimum correlation of possible factor scores     0.50 0.54
```

This, at least partially, reflects the intended structure. Some loadings, however, are very weak (<.40), and some items show also larger (but < .40) cross-loadings than intended loadings.

Such items are: C1_2: Politiikan ei pitäisi puuttua talouden toimintaan C1_3: Luonnon suojelemiseksi pitäisi ryhtyä vahvempiin toimenpiteisiin C1_5: Naisia pitäisi suosia työhönotossa ja ylennyksissä C1_11: Naisilla pitäisi olla vapaus päättää aborttikysymyksistä

Exclude these items and see if the solution produces a better criteria to which VAA responses can be compared

```r
subset_CS_items<-c("C1_7","C1_8","C1_1","C1_4","C1_6","C1_10")

fa(dat2011[,subset_CS_items],nfactors = 2,fm = "ml",rotate = "oblimin")
```

```
## Factor Analysis using method =  ml
## Call: fa(r = dat2011[, subset_CS_items], nfactors = 2, rotate = "oblimin",
##     fm = "ml")
## Standardized loadings (pattern matrix) based upon correlation matrix
##        ML2   ML1   h2    u2 com
## C1_7  0.01  0.42 0.18 0.822 1.0
## C1_8  0.00  1.00 1.00 0.005 1.0
## C1_1  0.71  0.04 0.51 0.493 1.0
## C1_4  0.56  0.02 0.31 0.686 1.0
## C1_6  0.63  0.01 0.40 0.600 1.0
## C1_10 0.56 -0.12 0.33 0.672 1.1
##
##                        ML2  ML1
## SS loadings           1.53 1.19
## Proportion Var        0.26 0.20
## Cumulative Var        0.26 0.45
## Proportion Explained  0.56 0.44
## Cumulative Proportion 0.56 1.00
##
##  With factor correlations of
##      ML2  ML1
## ML2 1.00 0.02
## ML1 0.02 1.00
##
## Mean item complexity =  1
## Test of the hypothesis that 2 factors are sufficient.
##
## The degrees of freedom for the null model are  15  and the objective function was  0.92 with Chi Squa
## The degrees of freedom for the model are 4  and the objective function was  0.02
##
## The root mean square of the residuals (RMSR) is  0.02
## The df corrected root mean square of the residuals is  0.05
##
## The harmonic number of observations is  898 with the empirical chi square  16.45  with prob <  0.0025
## The total number of observations was  2060  with Likelihood Chi Square =  44.44  with prob <  5.2e-09
##
## Tucker Lewis Index of factoring reliability =  0.919
## RMSEA index =  0.07  and the 90 % confidence intervals are  0.052 0.089
## BIC =  13.92
## Fit based upon off diagonal values = 0.99
## Measures of factor score adequacy
##                                                   ML2  ML1
## Correlation of (regression) scores with factors   0.85 1.00
## Multiple R square of scores with factors          0.72 1.00
## Minimum correlation of possible factor scores     0.45 0.99
```

It's better, although, the CS-LR dimension seems to be very strongly defined by a single item C1_8: Tuloja ja vaurautta pitäisi uudelleenjakaa tavallisten ihmisten suuntaan

H1 and H2 with a subset of CS-items

Model

```
model_H1H2.sub<-"
#loadings
VAA_LR=~y22+y23+y26+y27+y9+y19
VAA_GT=~y4+y5+y1+y21
CS_LR=~C1_7+C1_8
CS_GT=~C1_1+C1_4+C1_6+C1_10

#latent correlations

#cross-dimension same-method
VAA_LR~~r.VAA*VAA_GT
CS_LR~~r.CS*CS_GT

#concurrent validity
VAA_LR~~r.LR*CS_LR
VAA_GT~~r.GT*CS_GT

#cross-dimension cross-method correlations
VAA_LR~~r.d1*CS_GT
VAA_GT~~r.d2*CS_LR

#custom parameters
test.H1:=r.LR-max(r.VAA,r.CS,r.d1,r.d2)
test.H2:=r.GT-max(r.VAA,r.CS,r.d1,r.d2)

"
```

```
fit_H1H2.sub<-cfa(model=model_H1H2.sub,
              data=dat2011,
              missing="fiml")
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: covariance matrix of latent variables
##                  is not positive definite;
##                  use lavInspect(fit, "cov.lv") to investigate.
```

Problems again

Add the preregistered residual correlation.

```
model_H1H2.sub.re<-paste0(model_H1H2.sub,
                      "y4~~C1_4\n")



fit_H1H2.sub.re<-cfa(model=model_H1H2.sub.re,
              data=dat2011,
              missing="fiml")
```

```

round(inspect(fit_H1H2.sub,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

Fit the respecified model

```
##     npar       df    chisq   pvalue      cfi      tli    rmsea     srmr
##   54.000   98.000 1583.723    0.000    0.799    0.753    0.086    0.094
```

```
round(inspect(fit_H1H2.sub.re,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##     npar       df    chisq   pvalue      cfi      tli    rmsea     srmr
##   55.000   97.000 1256.316    0.000    0.843    0.805    0.076    0.090
```

Fit is quite poor

Hypotheses 1 and 2

Print standardized estimates to test the difference between correlations

```
std.est_H1H2.sub.re<-standardizedsolution(fit_H1H2.sub.re)
std.est_H1H2.sub.re[std.est_H1H2.sub.re$op==":=" |
               std.est_H1H2.sub.re$op=="~~" &
               std.est_H1H2.sub.re$lhs!=std.est_H1H2.sub.re$rhs,]
```

```
##        lhs op                            rhs est.std    se      z pvalue
## 17  VAA_LR ~~                         VAA_GT   0.446 0.025 17.531  0.000
## 18   CS_LR ~~                          CS_GT   0.033 0.042  0.785  0.432
## 19  VAA_LR ~~                          CS_LR   0.741 0.038 19.722  0.000
## 20  VAA_GT ~~                          CS_GT   0.987 0.016 62.249  0.000
## 21  VAA_LR ~~                          CS_GT   0.426 0.034 12.520  0.000
## 22  VAA_GT ~~                          CS_LR   0.099 0.039  2.509  0.012
## 23      y4 ~~                           C1_4   0.673 0.022 30.634  0.000
## 64 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)   0.295 0.047  6.295  0.000
## 65 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.541 0.030 18.175  0.000
##    ci.lower ci.upper
## 17    0.396    0.496
## 18   -0.049    0.114
## 19    0.667    0.814
## 20    0.956    1.018
## 21    0.359    0.492
## 22    0.022    0.176
## 23    0.630    0.716
## 64    0.203    0.386
## 65    0.482    0.599
```

H1: There is strong (.741, $p < .001$) correlation between VAA-LR and CS-LR, and it is notably stronger (difference in correlations .295, $p < .001$) than the strongest of correlations between different dimensions (.446 between VAA_LR and VAA_GT, $p < .001$)

H2: There is very strong (.987, $p < .001$) correlation between VAA-GT and CS-GT, and it is notably stronger (difference in correlations .541, $p < .001$) than the strongest of correlations between different dimensions (.446 between VAA_LR and VAA_GT, $p < .001$)

Seek misspecifications

```
mis.H1H2<-miPowerFit(fit_H1H2.sub.re,stdLoad=.40,cor=.20)
mis.H1H2<-mis.H1H2[mis.H1H2$op=="=~" |
                      (mis.H1H2$op=="~~" &
                         mis.H1H2$lhs!=mis.H1H2$rhs),]
#see summary of the decisions
table(mis.H1H2$decision.pow)
```

```
##
```

```
##  EPC:M EPC:NM      I      M     NM
##      5     73      6      3     79
```

```r
#there are eight misspecifications

rounded.vars<-c("mi","epc","target.epc",
                "std.epc","se.epc")

num.round<-function(var){
  var<-as.numeric(var)
  var<-round(var,2)
  return(var)
}

mis.H1H2[,rounded.vars]<-sapply(mis.H1H2[,rounded.vars],num.round)

printed.vars<-c("lhs","op","rhs","mi","epc","target.epc",
                "std.epc","std.target.epc","significant.mi",
                "high.power","decision.pow","se.epc")

#print the output

mis.H1H2 %>%
  filter(mis.H1H2$decision.pow=="M" |
              mis.H1H2$decision.pow=="EPC:M") %>%
  dplyr::select(all_of(printed.vars))
```

```
##      lhs op   rhs     mi   epc target.epc std.epc std.target.epc significant.mi
## 1 VAA_LR =~    y1 398.57 -0.76       0.53   -0.57            0.4           TRUE
## 2 VAA_GT =~  C1_6   4.43 -1.08       0.45   -0.96            0.4           TRUE
## 3  CS_LR =~   y26 134.67  1.66       1.03    0.65            0.4           TRUE
## 4  CS_LR =~    y9  61.16 -1.29       1.24   -0.42            0.4           TRUE
## 5  CS_LR =~    y1 355.24 -1.66       1.22   -0.55            0.4           TRUE
## 6  CS_GT =~    y1  91.57  9.72       0.83    4.68            0.4           TRUE
## 7  CS_GT =~   y21  14.56 -3.93       0.88   -1.78            0.4           TRUE
## 8     y1 ~~ C1_10  63.48  0.37       0.32    0.23            0.2           TRUE
##   high.power decision.pow se.epc
## 1       TRUE        EPC:M   0.04
## 2      FALSE            M   0.51
## 3       TRUE        EPC:M   0.14
## 4       TRUE        EPC:M   0.17
## 5       TRUE        EPC:M   0.09
## 6      FALSE            M   1.02
## 7      FALSE            M   1.03
## 8       TRUE        EPC:M   0.05
```

In this model as well, similarly to the model with the original set of items, y1 (Finland should continue to financially assist EU countries that are facing economic hardship (r.)) is indicated to crossload on all factor, and it also has a residual correlation with C1_10 (Maahanmuutto on hyvä asia Suomen taloudelle). Add this residual correlation.

```r
model_H1H2.sub.exp.re<-paste0(model_H1H2.sub.re,
                              "y1~~C1_10\n")
```

```r
fit_H1H2.sub.exp.re<-cfa(model=model_H1H2.sub.exp.re,
              data=dat2011,
```

```
              missing="fiml")
```

```
round(inspect(fit_H1H2.sub.re,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##     npar       df    chisq   pvalue      cfi      tli    rmsea     srmr
##   55.000   97.000 1256.316    0.000    0.843    0.805    0.076    0.090
```

```
round(inspect(fit_H1H2.sub.exp.re,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##     npar       df    chisq   pvalue      cfi      tli    rmsea     srmr
##   56.000   96.000 1188.944    0.000    0.852    0.815    0.074    0.089
```

Fit is not improved much

Hypotheses 1 and 2

Print standardized estimates to test the difference between correlations

```
std.est_H1H2.sub.exp.re<-standardizedsolution(fit_H1H2.sub.exp.re)
std.est_H1H2.sub.exp.re[std.est_H1H2.sub.exp.re$op==":=" |
               std.est_H1H2.sub.exp.re$op=="~~" &
               std.est_H1H2.sub.exp.re$lhs!=std.est_H1H2.sub.exp.re$rhs,]
```

```
##        lhs op                           rhs est.std    se      z pvalue
## 17  VAA_LR ~~                        VAA_GT   0.456 0.025 18.015  0.000
## 18   CS_LR ~~                         CS_GT   0.067 0.041  1.616  0.106
## 19  VAA_LR ~~                         CS_LR   0.740 0.038 19.687  0.000
## 20  VAA_GT ~~                         CS_GT   0.974 0.015 63.019  0.000
## 21  VAA_LR ~~                         CS_GT   0.468 0.033 14.242  0.000
## 22  VAA_GT ~~                         CS_LR   0.103 0.039  2.603  0.009
## 23      y4 ~~                          C1_4   0.680 0.021 31.639  0.000
## 24      y1 ~~                         C1_10   0.312 0.034  9.092  0.000
## 65 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)   0.273 0.052  5.208  0.000
## 66 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.507 0.036 14.222  0.000
##    ci.lower ci.upper
## 17    0.407    0.506
## 18   -0.014    0.148
## 19    0.667    0.814
## 20    0.944    1.004
## 21    0.403    0.532
## 22    0.025    0.180
## 23    0.638    0.722
## 24    0.245    0.379
## 65    0.170    0.375
## 66    0.437    0.576
```

Hypothesis inference seems almost identical

H1: There is strong (.740, $p < .001$) correlation between VAA-LR and CS-LR, and it is notably stronger (difference in correlations .273, $p < .001$) than the strongest of correlations between different dimensions (.456 between VAA_LR and VAA_GT, $p < .001$)

H2: There is very strong (.974, $p < .001$) correlation between VAA-GT and CS-GT, and it is notably stronger (difference in correlations .507, $p < .001$) than the strongest of correlations between different dimensions (.456 between VAA_LR and VAA_GT, $p < .001$)

Seek misspecifications

```r
mis.H1H2<-miPowerFit(fit_H1H2.sub.exp.re,stdLoad=.40,cor=.20)
mis.H1H2<-mis.H1H2[mis.H1H2$op=="=~" |
                     (mis.H1H2$op=="~~" &
                        mis.H1H2$lhs!=mis.H1H2$rhs),]
#see summary of the decisions
table(mis.H1H2$decision.pow)
```

```
##
##  EPC:M EPC:NM      I      M     NM
##      4     65      4      5     87
```

```r
#there are eight misspecifications

rounded.vars<-c("mi","epc","target.epc",
                "std.epc","se.epc")

num.round<-function(var){
  var<-as.numeric(var)
  var<-round(var,2)
  return(var)
}

mis.H1H2[,rounded.vars]<-sapply(mis.H1H2[,rounded.vars],num.round)

printed.vars<-c("lhs","op","rhs","mi","epc","target.epc",
                "std.epc","std.target.epc","significant.mi",
                "high.power","decision.pow","se.epc")

#print the output

mis.H1H2 %>%
  filter(mis.H1H2$decision.pow=="M" |
               mis.H1H2$decision.pow=="EPC:M") %>%
  dplyr::select(all_of(printed.vars))
```

```
##      lhs op   rhs     mi   epc target.epc std.epc std.target.epc significant.mi
## 1 VAA_LR =~    y1 341.78 -0.70       0.53   -0.53            0.4           TRUE
## 2 VAA_GT =~  C1_6  18.38 -1.94       0.45   -1.73            0.4           TRUE
## 3 VAA_GT =~ C1_10  25.31  2.38       0.48    2.00            0.4           TRUE
## 4  CS_LR =~   y26 134.90  1.65       1.03    0.64            0.4           TRUE
## 5  CS_LR =~    y9  61.54 -1.29       1.24   -0.42            0.4           TRUE
## 6  CS_LR =~    y1 311.10 -1.53       1.22   -0.50            0.4           TRUE
## 7  CS_GT =~    y5  23.64  5.37       0.92    2.33            0.4           TRUE
## 8  CS_GT =~    y1  25.81 -4.12       0.81   -2.02            0.4           TRUE
## 9  CS_GT =~   y21   7.84 -2.37       0.86   -1.10            0.4           TRUE
##   high.power decision.pow se.epc
## 1       TRUE        EPC:M   0.04
## 2      FALSE            M   0.45
## 3      FALSE            M   0.47
## 4       TRUE        EPC:M   0.14
## 5       TRUE        EPC:M   0.16
## 6       TRUE        EPC:M   0.09
## 7      FALSE            M   1.10
## 8      FALSE            M   0.81
```

```
## 9      FALSE            M   0.85
```

The cross-loadings with y1 did not go away with the residual correlation. Exclude the item entirely.

```
model_H1H2.sub.exp.re.2<-"
#loadings
VAA_LR=~y22+y23+y26+y27+y9+y19
VAA_GT=~y4+y5+y21
CS_LR=~C1_7+C1_8
CS_GT=~C1_1+C1_4+C1_6+C1_10

#latent correlations

#cross-dimension same-method
VAA_LR~~r.VAA*VAA_GT
CS_LR~~r.CS*CS_GT

#concurrent validity
VAA_LR~~r.LR*CS_LR
VAA_GT~~r.GT*CS_GT

#cross-dimension cross-method correlations
VAA_LR~~r.d1*CS_GT
VAA_GT~~r.d2*CS_LR

#custom parameters
test.H1:=r.LR-max(r.VAA,r.CS,r.d1,r.d2)
test.H2:=r.GT-max(r.VAA,r.CS,r.d1,r.d2)

y4  ~~  C1_4
"
```

```
fit_H1H2.sub.exp.re.2<-cfa(model=model_H1H2.sub.exp.re.2,
              data=dat2011,
              missing="fiml")
```

```
## Warning in lav_data_full(data = data, group = group, cluster = cluster, : lavaan WARNING: some cases
##   573
```

```
round(inspect(fit_H1H2.sub.exp.re,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##     npar       df    chisq   pvalue      cfi      tli    rmsea     srmr
##   56.000   96.000 1188.944    0.000    0.852    0.815    0.074    0.089
```

```
round(inspect(fit_H1H2.sub.exp.re.2,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  52.000  83.000 752.004   0.000   0.900   0.874   0.063   0.069
```

Now the fit seems to be adequate (These models are not nested, and can't be compared, therefore)

Hypotheses 1 and 2

Print standardized estimates to test the difference between correlations

```
std.est_H1H2.sub.exp.re.2<-standardizedsolution(fit_H1H2.sub.exp.re.2)
std.est_H1H2.sub.exp.re.2[std.est_H1H2.sub.exp.re.2$op==":=" |
```

```r
              std.est_H1H2.sub.exp.re.2$op=="~~" &
              std.est_H1H2.sub.exp.re.2$lhs!=std.est_H1H2.sub.exp.re.2$rhs,]
```

```
##        lhs op                          rhs est.std    se      z pvalue
## 16  VAA_LR ~~                       VAA_GT   0.501 0.025 20.373    0.0
## 17   CS_LR ~~                        CS_GT   0.043 0.042  1.037    0.3
## 18  VAA_LR ~~                        CS_LR   0.741 0.038 19.758    0.0
## 19  VAA_GT ~~                        CS_GT   0.988 0.016 60.121    0.0
## 20  VAA_LR ~~                        CS_GT   0.442 0.034 13.018    0.0
## 21  VAA_GT ~~                        CS_LR   0.143 0.040  3.594    0.0
## 22      y4 ~~                         C1_4   0.671 0.022 30.211    0.0
## 61 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)   0.240 0.046  5.211    0.0
## 62 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.488 0.029 16.739    0.0
##    ci.lower ci.upper
## 16    0.453    0.549
## 17   -0.038    0.125
## 18    0.668    0.815
## 19    0.956    1.021
## 20    0.375    0.508
## 21    0.065    0.221
## 22    0.627    0.714
## 61    0.150    0.331
## 62    0.430    0.545
```

Hypothesis inference seems almost identical

H1: There is strong (.741, p < .001) correlation between VAA-LR and CS-LR, and it is notably stronger (difference in correlations .240, p < .001) than the strongest of correlations between different dimensions (.442 between VAA_LR and VAA_GT, p < .001)

H2: There is very strong (.988, p < .001) correlation between VAA-GT and CS-GT, and it is notably stronger (difference in correlations .488, p < .001) than the strongest of correlations between different dimensions (.442 between VAA_LR and VAA_GT, p < .001)

Print all model parameters

```r
std.est_H1H2.sub.exp.re.2
```

```
##        lhs op                          rhs est.std    se      z pvalue
## 1   VAA_LR =~                          y22   0.692 0.015 44.991    0.0
## 2   VAA_LR =~                          y23   0.604 0.018 33.963    0.0
## 3   VAA_LR =~                          y26   0.566 0.019 30.154    0.0
## 4   VAA_LR =~                          y27   0.475 0.020 23.192    0.0
## 5   VAA_LR =~                           y9   0.674 0.016 42.279    0.0
## 6   VAA_LR =~                          y19   0.650 0.016 39.664    0.0
## 7   VAA_GT =~                           y4   0.625 0.018 34.091    0.0
## 8   VAA_GT =~                           y5   0.810 0.015 52.273    0.0
## 9   VAA_GT =~                          y21   0.562 0.020 28.253    0.0
## 10   CS_LR =~                         C1_7   0.488 0.032 15.141    0.0
## 11   CS_LR =~                         C1_8   0.866 0.036 23.873    0.0
## 12   CS_GT =~                         C1_1   0.681 0.023 29.599    0.0
## 13   CS_GT =~                         C1_4   0.588 0.023 25.212    0.0
## 14   CS_GT =~                         C1_6   0.574 0.027 21.332    0.0
## 15   CS_GT =~                        C1_10   0.620 0.025 24.409    0.0
## 16  VAA_LR ~~                       VAA_GT   0.501 0.025 20.373    0.0
## 17   CS_LR ~~                        CS_GT   0.043 0.042  1.037    0.3
```

```
## 18  VAA_LR ~~                           CS_LR   0.741 0.038 19.758    0.0
## 19  VAA_GT ~~                           CS_GT   0.988 0.016 60.121    0.0
## 20  VAA_LR ~~                           CS_GT   0.442 0.034 13.018    0.0
## 21  VAA_GT ~~                           CS_LR   0.143 0.040  3.594    0.0
## 22      y4 ~~                            C1_4   0.671 0.022 30.211    0.0
## 23     y22 ~~                             y22   0.522 0.021 24.528    0.0
## 24     y23 ~~                             y23   0.636 0.021 29.607    0.0
## 25     y26 ~~                             y26   0.680 0.021 32.042    0.0
## 26     y27 ~~                             y27   0.775 0.019 39.846    0.0
## 27      y9 ~~                              y9   0.546 0.021 25.445    0.0
## 28     y19 ~~                             y19   0.578 0.021 27.177    0.0
## 29      y4 ~~                              y4   0.609 0.023 26.599    0.0
## 30      y5 ~~                              y5   0.345 0.025 13.743    0.0
## 31     y21 ~~                             y21   0.684 0.022 30.589    0.0
## 32    C1_7 ~~                            C1_7   0.762 0.031 24.268    0.0
## 33    C1_8 ~~                            C1_8   0.250 0.063  3.982    0.0
## 34    C1_1 ~~                            C1_1   0.536 0.031 17.106    0.0
## 35    C1_4 ~~                            C1_4   0.654 0.027 23.862    0.0
## 36    C1_6 ~~                            C1_6   0.670 0.031 21.656    0.0
## 37   C1_10 ~~                           C1_10   0.615 0.032 19.520    0.0
## 38  VAA_LR ~~                          VAA_LR   1.000 0.000     NA     NA
## 39  VAA_GT ~~                          VAA_GT   1.000 0.000     NA     NA
## 40   CS_LR ~~                           CS_LR   1.000 0.000     NA     NA
## 41   CS_GT ~~                           CS_GT   1.000 0.000     NA     NA
## 42     y22 ~1                                   1.810 0.037 48.564    0.0
## 43     y23 ~1                                   1.671 0.036 47.079    0.0
## 44     y26 ~1                                   1.601 0.035 46.306    0.0
## 45     y27 ~1                                   2.803 0.051 55.082    0.0
## 46      y9 ~1                                   2.555 0.047 54.061    0.0
## 47     y19 ~1                                   1.619 0.035 46.623    0.0
## 48      y4 ~1                                   1.527 0.033 46.240    0.0
## 49      y5 ~1                                   2.053 0.040 51.498    0.0
## 50     y21 ~1                                   1.846 0.038 48.624    0.0
## 51    C1_7 ~1                                   2.109 0.059 35.776    0.0
## 52    C1_8 ~1                                   1.797 0.051 35.436    0.0
## 53    C1_1 ~1                                   4.208 0.100 41.992    0.0
## 54    C1_4 ~1                                   1.434 0.040 35.409    0.0
## 55    C1_6 ~1                                   3.167 0.079 39.909    0.0
## 56   C1_10 ~1                                   2.040 0.056 36.585    0.0
## 57  VAA_LR ~1                                   0.000 0.000     NA     NA
## 58  VAA_GT ~1                                   0.000 0.000     NA     NA
## 59   CS_LR ~1                                   0.000 0.000     NA     NA
## 60   CS_GT ~1                                   0.000 0.000     NA     NA
## 61 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)   0.240 0.046  5.211    0.0
## 62 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.488 0.029 16.739    0.0
##    ci.lower ci.upper
## 1     0.662    0.722
## 2     0.569    0.639
## 3     0.529    0.602
## 4     0.435    0.515
## 5     0.642    0.705
## 6     0.617    0.682
## 7     0.589    0.661
## 8     0.779    0.840
```

```
## 9     0.523    0.601
## 10    0.425    0.551
## 11    0.795    0.937
## 12    0.636    0.726
## 13    0.542    0.634
## 14    0.522    0.627
## 15    0.570    0.670
## 16    0.453    0.549
## 17   -0.038    0.125
## 18    0.668    0.815
## 19    0.956    1.021
## 20    0.375    0.508
## 21    0.065    0.221
## 22    0.627    0.714
## 23    0.480    0.563
## 24    0.593    0.678
## 25    0.638    0.722
## 26    0.736    0.813
## 27    0.504    0.588
## 28    0.536    0.620
## 29    0.565    0.654
## 30    0.295    0.394
## 31    0.640    0.728
## 32    0.701    0.824
## 33    0.127    0.373
## 34    0.475    0.598
## 35    0.601    0.708
## 36    0.609    0.731
## 37    0.554    0.677
## 38    1.000    1.000
## 39    1.000    1.000
## 40    1.000    1.000
## 41    1.000    1.000
## 42    1.737    1.883
## 43    1.602    1.741
## 44    1.534    1.669
## 45    2.703    2.903
## 46    2.463    2.648
## 47    1.551    1.687
## 48    1.463    1.592
## 49    1.974    2.131
## 50    1.771    1.920
## 51    1.994    2.225
## 52    1.697    1.896
## 53    4.012    4.405
## 54    1.354    1.513
## 55    3.012    3.323
## 56    1.931    2.149
## 57    0.000    0.000
## 58    0.000    0.000
## 59    0.000    0.000
## 60    0.000    0.000
## 61    0.150    0.331
## 62    0.430    0.545
```

H3 and H4

Exclude other than members of the eight parties that have multiple members in the parliament

```
dat2011.party<-dat2011 %>%
  filter(puolue=="KD" |
           puolue=="KESK" |
           puolue=="KOK" |
           puolue=="PS" |
           puolue=="RKP" |
           puolue=="SDP" |
           puolue=="VAS" |
           puolue=="VIHR")

table(dat2011.party$puolue)
```

```
##
##   KD KESK  KOK   PS  RKP  SDP  VAS VIHR
##  170  220  227  213   77  222  217  226
```

Define the model

Use the subset of items that were used for H1 and H2

```
model_H3H4<-"
#loadings
VAA_LR=~y22+y23+y26+y27+y9+y19
VAA_GT=~y4+y5+y21
CS_LR=~C1_7+C1_8
CS_GT=~C1_1+C1_4+C1_6+C1_10

#cross-dimension same-method
VAA_LR~~c(r.VAA.KD,r.VAA.KESK,r.VAA.KOK,r.VAA.PS,r.VAA.RKP,r.VAA.SDP,r.VAA.VAS,r.VAA.VIHR)*VAA_GT
CS_LR~~c(r.CS.KD,r.CS.KESK,r.CS.KOK,r.CS.PS,r.CS.RKP,r.CS.SDP,r.CS.VAS,r.CS.VIHR)*CS_GT

#concurrent validity
VAA_LR~~c(r.LR.KD,r.LR.KESK,r.LR.KOK,r.LR.PS,r.LR.RKP,r.LR.SDP,r.LR.VAS,r.LR.VIHR)*CS_LR
VAA_GT~~c(r.GT.KD,r.GT.KESK,r.GT.KOK,r.GT.PS,r.GT.RKP,r.GT.SDP,r.GT.VAS,r.GT.VIHR)*CS_GT

#cross-dimension cross-method correlations
VAA_LR~~c(r.d1.KD,r.d1.KESK,r.d1.KOK,r.d1.PS,r.d1.RKP,r.d1.SDP,r.d1.VAS,r.d1.VIHR)*CS_GT
VAA_GT~~c(r.d2.KD,r.d2.KESK,r.d2.KOK,r.d2.PS,r.d2.RKP,r.d2.SDP,r.d2.VAS,r.d2.VIHR)*CS_LR

#custom parameters
mean.r.VAA:=mean(r.VAA.KD,r.VAA.KESK,r.VAA.KOK,r.VAA.PS,r.VAA.RKP,r.VAA.SDP,r.VAA.VAS,r.VAA.VIHR)
mean.r.CS:=mean(r.CS.KD,r.CS.KESK,r.CS.KOK,r.CS.PS,r.CS.RKP,r.CS.SDP,r.CS.VAS,r.CS.VIHR)
mean.r.LR:=mean(r.LR.KD,r.LR.KESK,r.LR.KOK,r.LR.PS,r.LR.RKP,r.LR.SDP,r.LR.VAS,r.LR.VIHR)
mean.r.GT:=mean(r.GT.KD,r.GT.KESK,r.GT.KOK,r.GT.PS,r.GT.RKP,r.GT.SDP,r.GT.VAS,r.GT.VIHR)
mean.r.d1:=mean(r.d1.KD,r.d1.KESK,r.d1.KOK,r.d1.PS,r.d1.RKP,r.d1.SDP,r.d1.VAS,r.d1.VIHR)
mean.r.d2:=mean(r.d2.KD,r.d2.KESK,r.d2.KOK,r.d2.PS,r.d2.RKP,r.d2.SDP,r.d2.VAS,r.d2.VIHR)

test.H3:=mean.r.LR-max(mean.r.VAA,mean.r.CS,mean.r.d1,mean.r.d2)
test.H4:=mean.r.GT-max(mean.r.VAA,mean.r.CS,mean.r.d1,mean.r.d2)
```

```
"
```

Fit the configural model

```
fit_H3H4<-cfa(model=model_H3H4,
              data=dat2011.party,
              group=c("puolue"),
              group.label=c("KD","KESK","KOK","PS","RKP","SDP","VAS","VIHR"),
              missing="fiml")
```

```
## Warning in lav_data_full(data = data, group = group, cluster = cluster, : lavaan WARNING: some cases
##   452
```

```
## Warning in lav_model_estimate(lavmodel = lavmodel, lavpartable = lavpartable, : lavaan WARNING: the 
##                     but not all elements of the gradient are (near) zero;
##                     the optimizer may not have found a local solution
##                     use check.gradient = FALSE to skip this check.
```

Add the preregistered residual correlation

```
model_H3H4.re<-paste0(model_H3H4,
                      "y4~~C1_4\n")
```

```

fit_H3H4.re<-cfa(model=model_H3H4.re,
              data=dat2011.party,
              group=c("puolue"),
              group.label=c("KD","KESK","KOK","PS","RKP","SDP","VAS","VIHR"),
              missing="fiml")
```

Fit the respecified model

```
## Warning in lav_data_full(data = data, group = group, cluster = cluster, : lavaan WARNING: some cases
##   452
```

```
## Warning in lav_model_estimate(lavmodel = lavmodel, lavpartable = lavpartable, : lavaan WARNING: the 
##                     but not all elements of the gradient are (near) zero;
##                     the optimizer may not have found a local solution
##                     use check.gradient = FALSE to skip this check.
```

The model does not converge.

```
summary(fit_H3H4.re,fit=T,standardized=T)
```

```
## lavaan 0.6-5 did NOT end normally after 1906 iterations
## ** WARNING ** Estimates below are most likely unreliable
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of free parameters                        416
##
##   Number of observations per group:
##     KD                                             170
##     KESK                                           220
##     KOK                                            227
```

```
##     PS                                          213
##     RKP                                          77
##     SDP                                         222
##     VAS                                         216
##     VIHR                                        226
##   Number of missing patterns per group:
##     KD                                           11
##     KESK                                         13
##     KOK                                          11
##     PS                                           15
##     RKP                                          17
##     SDP                                          14
##     VAS                                          11
##     VIHR                                         14
##
## Model Test User Model:
##
##   Test statistic                                 NA
##   Degrees of freedom                             NA
##   Test statistic for each group:
##     KD                                           NA
##     KESK                                         NA
##     KOK                                          NA
##     PS                                           NA
##     RKP                                          NA
##     SDP                                          NA
##     VAS                                          NA
##     VIHR                                         NA
```

```
## Warning in .local(object, ...): lavaan WARNING: fit measures not available if model did not converge
```

```
## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in sqrt(ETA2): NaNs produced
```

```
##
## Parameter Estimates:
##
##   Information                              Observed
##   Observed information based on             Hessian
##   Standard errors                          Standard
##
##
## Group 1 [KD]:
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
```

```
##   VAA_LR =~
##     y22               1.000                                0.364    0.277
##     y23               1.252       NA                       0.456    0.392
##     y26               1.673       NA                       0.609    0.642
##     y27               0.084       NA                       0.031    0.035
##     y9                0.485       NA                       0.176    0.172
##     y19               0.889       NA                       0.323    0.257
##   VAA_GT =~
##     y4                1.000                                0.024    0.025
##     y5                1.117       NA                       0.026    0.021
##     y21             340.250       NA                       8.059    6.340
##   CS_LR =~
##     C1_7              1.000                                0.233    0.273
##     C1_8              3.191       NA                       0.742    0.907
##   CS_GT =~
##     C1_1              1.000                                0.364    0.646
##     C1_4              0.055       NA                       0.020    0.020
##     C1_6              1.129       NA                       0.411    0.610
##     C1_10             0.859       NA                       0.312    0.355
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR ~~
##     VAA_GT  (r.VA)   -0.000       NA                      -0.002   -0.002
##   CS_LR ~~
##     CS_GT   (r.CS)    0.038       NA                       0.452    0.452
##   VAA_LR ~~
##     CS_LR   (r.LR)    0.094       NA                       1.113    1.113
##   VAA_GT ~~
##     CS_GT   (r.GT)   -0.000       NA                      -0.031   -0.031
##   VAA_LR ~~
##     CS_GT   (r.1.)    0.051       NA                       0.386    0.386
##   VAA_GT ~~
##     CS_LR   (r.2.)    0.000       NA                       0.018    0.018
##  .y4 ~~
##    .C1_4              0.476       NA                       0.476    0.504
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               3.248       NA                       3.248    2.476
##    .y23               2.385       NA                       2.385    2.050
##    .y26               1.809       NA                       1.809    1.907
##    .y27               4.199       NA                       4.199    4.757
##    .y9                3.834       NA                       3.834    3.742
##    .y19               2.888       NA                       2.888    2.295
##    .y4                4.448       NA                       4.448    4.606
##    .y5                3.314       NA                       3.314    2.610
##    .y21               2.948       NA                       2.948    2.319
##    .C1_7              2.039       NA                       2.039    2.392
##    .C1_8              1.832       NA                       1.832    2.239
##    .C1_1              4.395       NA                       4.395    7.807
##    .C1_4              4.366       NA                       4.366    4.467
##    .C1_6              4.038       NA                       4.038    5.996
##    .C1_10             2.454       NA                       2.454    2.788
```

```
##     VAA_LR            0.000                                0.000    0.000
##     VAA_GT            0.000                                0.000    0.000
##     CS_LR             0.000                                0.000    0.000
##     CS_GT             0.000                                0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               1.588       NA                      1.588    0.923
##    .y23               1.146       NA                      1.146    0.847
##    .y26               0.529       NA                      0.529    0.588
##    .y27               0.778       NA                      0.778    0.999
##    .y9                1.019       NA                      1.019    0.970
##    .y19               1.480       NA                      1.480    0.934
##    .y4                0.932       NA                      0.932    0.999
##    .y5                1.612       NA                      1.612    1.000
##    .y21             -63.339       NA                    -63.339  -39.201
##    .C1_7              0.673       NA                      0.673    0.926
##    .C1_8              0.118       NA                      0.118    0.177
##    .C1_1              0.184       NA                      0.184    0.582
##    .C1_4              0.955       NA                      0.955    1.000
##    .C1_6              0.285       NA                      0.285    0.628
##    .C1_10             0.677       NA                      0.677    0.874
##     VAA_LR            0.132       NA                      1.000    1.000
##     VAA_GT            0.001       NA                      1.000    1.000
##     CS_LR             0.054       NA                      1.000    1.000
##     CS_GT             0.132       NA                      1.000    1.000
##
##
## Group 2 [KESK]:
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR =~
##     y22               1.000                               0.770    0.652
##     y23              -0.213       NA                     -0.164   -0.130
##     y26              -0.127       NA                     -0.098   -0.103
##     y27               0.403       NA                      0.310    0.255
##     y9                0.381       NA                      0.293    0.354
##     y19               0.757       NA                      0.583    0.467
##   VAA_GT =~
##     y4                1.000                               0.738    0.506
##     y5                1.257       NA                      0.927    0.841
##     y21               0.391       NA                      0.289    0.234
##   CS_LR =~
##     C1_7              1.000                               6.275    6.552
##     C1_8              0.005       NA                      0.034    0.037
##   CS_GT =~
##     C1_1              1.000                               0.303    0.478
##     C1_4              2.318       NA                      0.703    0.458
##     C1_6              1.132       NA                      0.343    0.386
##     C1_10             1.363       NA                      0.413    0.454
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
```

```
##   VAA_LR ~~
##     VAA_GT  (r.VA)    0.191       NA                         0.337    0.337
##   CS_LR ~~
##     CS_GT   (r.CS)   -0.014       NA                        -0.007   -0.007
##   VAA_LR ~~
##     CS_LR   (r.LR)    0.143       NA                         0.030    0.030
##   VAA_GT ~~
##     CS_GT   (r.GT)    0.190       NA                         0.848    0.848
##   VAA_LR ~~
##     CS_GT   (r.1.)    0.126       NA                         0.540    0.540
##   VAA_GT ~~
##     CS_LR   (r.2.)   -0.120       NA                        -0.026   -0.026
##  .y4 ~~
##    .C1_4              1.338       NA                         1.338    0.781
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               3.496       NA                         3.496    2.961
##    .y23               2.721       NA                         2.721    2.160
##    .y26               2.001       NA                         2.001    2.107
##    .y27               3.251       NA                         3.251    2.671
##    .y9                4.241       NA                         4.241    5.122
##    .y19               2.869       NA                         2.869    2.300
##    .y4                3.060       NA                         3.060    2.100
##    .y5                3.496       NA                         3.496    3.170
##    .y21               3.586       NA                         3.586    2.912
##    .C1_7              2.221       NA                         2.221    2.319
##    .C1_8              2.294       NA                         2.294    2.499
##    .C1_1              4.064       NA                         4.064    6.401
##    .C1_4              2.694       NA                         2.694    1.757
##    .C1_6              3.815       NA                         3.815    4.289
##    .C1_10             2.222       NA                         2.222    2.442
##     VAA_LR            0.000                                  0.000    0.000
##     VAA_GT            0.000                                  0.000    0.000
##     CS_LR             0.000                                  0.000    0.000
##     CS_GT             0.000                                  0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               0.802       NA                         0.802    0.575
##    .y23               1.561       NA                         1.561    0.983
##    .y26               0.892       NA                         0.892    0.989
##    .y27               1.385       NA                         1.385    0.935
##    .y9                0.600       NA                         0.600    0.875
##    .y19               1.215       NA                         1.215    0.782
##    .y4                1.578       NA                         1.578    0.744
##    .y5                0.357       NA                         0.357    0.293
##    .y21               1.433       NA                         1.433    0.945
##    .C1_7            -38.463       NA                       -38.463  -41.932
##    .C1_8              0.842       NA                         0.842    0.999
##    .C1_1              0.311       NA                         0.311    0.772
##    .C1_4              1.857       NA                         1.857    0.790
##    .C1_6              0.673       NA                         0.673    0.851
##    .C1_10             0.658       NA                         0.658    0.794
```

```
##     VAA_LR            0.592       NA                            1.000    1.000
##     VAA_GT            0.544       NA                            1.000    1.000
##     CS_LR            39.380       NA                            1.000    1.000
##     CS_GT             0.092       NA                            1.000    1.000
##
##
## Group 3 [KOK]:
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR =~
##     y22               1.000                               0.653    0.662
##     y23               0.596       NA                      0.389    0.310
##     y26               0.062       NA                      0.041    0.034
##     y27               0.600       NA                      0.392    0.421
##     y9                0.354       NA                      0.231    0.351
##     y19               0.954       NA                      0.623    0.507
##   VAA_GT =~
##     y4                1.000                               0.571    0.383
##     y5                1.415       NA                      0.808    0.705
##     y21               0.756       NA                      0.432    0.341
##   CS_LR =~
##     C1_7              1.000                               0.685    0.725
##     C1_8              0.263       NA                      0.180    0.173
##   CS_GT =~
##     C1_1              1.000                               0.371    0.565
##     C1_4              2.016       NA                      0.748    0.481
##     C1_6              1.109       NA                      0.411    0.407
##     C1_10             0.537       NA                      0.199    0.278
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR ~~
##     VAA_GT  (r.VA)    0.157       NA                      0.421    0.421
##   CS_LR ~~
##     CS_GT   (r.CS)   -0.016       NA                     -0.062   -0.062
##   VAA_LR ~~
##     CS_LR   (r.LR)    0.066       NA                      0.147    0.147
##   VAA_GT ~~
##     CS_GT   (r.GT)    0.215       NA                      1.017    1.017
##   VAA_LR ~~
##     CS_GT   (r.1.)    0.092       NA                      0.380    0.380
##   VAA_GT ~~
##     CS_LR   (r.2.)    0.085       NA                      0.217    0.217
##  .y4 ~~
##    .C1_4              1.123       NA                      1.123    0.599
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               3.846       NA                      3.846    3.898
##    .y23               3.585       NA                      3.585    2.851
##    .y26               3.076       NA                      3.076    2.580
##    .y27               4.186       NA                      4.186    4.492
##    .y9                4.531       NA                      4.531    6.871
```

```
##    .y19              3.354       NA                          3.354    2.728
##    .y4               2.558       NA                          2.558    1.717
##    .y5               3.565       NA                          3.565    3.109
##    .y21              2.686       NA                          2.686    2.122
##    .C1_7             2.416       NA                          2.416    2.556
##    .C1_8             3.448       NA                          3.448    3.317
##    .C1_1             4.210       NA                          4.210    6.417
##    .C1_4             2.358       NA                          2.358    1.517
##    .C1_6             3.787       NA                          3.787    3.745
##    .C1_10            1.688       NA                          1.688    2.359
##     VAA_LR           0.000                                   0.000    0.000
##     VAA_GT           0.000                                   0.000    0.000
##     CS_LR            0.000                                   0.000    0.000
##     CS_GT            0.000                                   0.000    0.000
##
## Variances:
##                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22              0.547       NA                          0.547    0.562
##    .y23              1.430       NA                          1.430    0.904
##    .y26              1.419       NA                          1.419    0.999
##    .y27              0.714       NA                          0.714    0.823
##    .y9               0.381       NA                          0.381    0.877
##    .y19              1.123       NA                          1.123    0.743
##    .y4               1.893       NA                          1.893    0.853
##    .y5               0.662       NA                          0.662    0.503
##    .y21              1.416       NA                          1.416    0.884
##    .C1_7             0.424       NA                          0.424    0.475
##    .C1_8             1.049       NA                          1.049    0.970
##    .C1_1             0.293       NA                          0.293    0.680
##    .C1_4             1.859       NA                          1.859    0.769
##    .C1_6             0.853       NA                          0.853    0.835
##    .C1_10            0.472       NA                          0.472    0.923
##     VAA_LR           0.427       NA                          1.000    1.000
##     VAA_GT           0.326       NA                          1.000    1.000
##     CS_LR            0.470       NA                          1.000    1.000
##     CS_GT            0.138       NA                          1.000    1.000
##
##
## Group 4 [PS]:
##
## Latent Variables:
##                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR =~
##     y22              1.000                                   0.854    0.600
##     y23              0.571       NA                          0.488    0.444
##     y26              0.398       NA                          0.340    0.390
##     y27              0.390       NA                          0.333    0.462
##     y9               0.708       NA                          0.605    0.578
##     y19              0.529       NA                          0.452    0.369
##   VAA_GT =~
##     y4               1.000                                     NaN      NaN
##     y5            -318.172       NA                            NaN      NaN
##     y21              1.348       NA                            NaN      NaN
##   CS_LR =~
```

```
##     C1_7              1.000                                   0.358    0.484
##     C1_8              1.196       NA                          0.428    0.598
##   CS_GT =~
##     C1_1              1.000                                   0.247    0.614
##     C1_4              2.362       NA                          0.584    0.428
##     C1_6              2.102       NA                          0.519    0.599
##     C1_10             1.789       NA                          0.442    0.463
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR ~~
##     VAA_GT  (r.VA)   -0.000       NA                         -0.036   -0.036
##   CS_LR ~~
##     CS_GT   (r.CS)    0.002       NA                          0.022    0.022
##   VAA_LR ~~
##     CS_LR   (r.LR)    0.187       NA                          0.611    0.611
##   VAA_GT ~~
##     CS_GT   (r.GT)   -0.000       NA                         -0.058   -0.058
##   VAA_LR ~~
##     CS_GT   (r.1.)    0.072       NA                          0.340    0.340
##   VAA_GT ~~
##     CS_LR   (r.2.)    0.000       NA                          0.014    0.014
##  .y4 ~~
##    .C1_4              0.745       NA                          0.745    0.463
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               3.001       NA                          3.001    2.109
##    .y23               2.075       NA                          2.075    1.888
##    .y26               1.521       NA                          1.521    1.746
##    .y27               4.459       NA                          4.459    6.189
##    .y9                3.924       NA                          3.924    3.745
##    .y19               2.353       NA                          2.353    1.918
##    .y4                3.907       NA                          3.907    2.987
##    .y5                4.870       NA                          4.870   10.757
##    .y21               3.613       NA                          3.613    2.920
##    .C1_7              1.829       NA                          1.829    2.469
##    .C1_8              1.620       NA                          1.620    2.260
##    .C1_1              4.797       NA                          4.797   11.930
##    .C1_4              3.557       NA                          3.557    2.611
##    .C1_6              4.184       NA                          4.184    4.826
##    .C1_10             3.761       NA                          3.761    3.940
##     VAA_LR            0.000                                   0.000    0.000
##     VAA_GT            0.000                                     NaN      NaN
##     CS_LR             0.000                                   0.000    0.000
##     CS_GT             0.000                                   0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               1.296       NA                          1.296    0.640
##    .y23               0.970       NA                          0.970    0.803
##    .y26               0.644       NA                          0.644    0.848
##    .y27               0.408       NA                          0.408    0.786
##    .y9                0.732       NA                          0.732    0.666
```

```
##    .y19               1.301       NA                              1.301    0.864
##    .y4                1.711       NA                              1.711    1.000
##    .y5               12.498       NA                             12.498   60.970
##    .y21               1.531       NA                              1.531    1.000
##    .C1_7              0.420       NA                              0.420    0.766
##    .C1_8              0.330       NA                              0.330    0.643
##    .C1_1              0.101       NA                              0.101    0.623
##    .C1_4              1.515       NA                              1.515    0.817
##    .C1_6              0.482       NA                              0.482    0.641
##    .C1_10             0.716       NA                              0.716    0.786
##     VAA_LR            0.730       NA                              1.000    1.000
##     VAA_GT           -0.000       NA                                NaN      NaN
##     CS_LR             0.128       NA                              1.000    1.000
##     CS_GT             0.061       NA                              1.000    1.000
##
##
## Group 5 [RKP]:
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR =~
##     y22               1.000                               0.696    0.522
##     y23               0.872       NA                      0.607    0.476
##     y26               1.042       NA                      0.725    0.535
##     y27               0.282       NA                      0.196    0.198
##     y9                0.588       NA                      0.409    0.406
##     y19               0.665       NA                      0.463    0.347
##   VAA_GT =~
##     y4                1.000                               0.301    0.308
##     y5                0.492       NA                      0.148    0.173
##     y21               2.732       NA                      0.823    0.657
##   CS_LR =~
##     C1_7              1.000                               0.290    0.303
##     C1_8              5.616       NA                      1.631    1.449
##   CS_GT =~
##     C1_1              1.000                                 NaN      NaN
##     C1_4              0.697       NA                        NaN      NaN
##     C1_6             -0.149       NA                        NaN      NaN
##     C1_10            -5.766       NA                        NaN      NaN
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR ~~
##     VAA_GT  (r.VA)   -0.023       NA                     -0.108   -0.108
##   CS_LR ~~
##     CS_GT   (r.CS)   -0.001       NA                     -0.025   -0.025
##   VAA_LR ~~
##     CS_LR   (r.LR)    0.068       NA                      0.339    0.339
##   VAA_GT ~~
##     CS_GT   (r.GT)   -0.022       NA                     -0.361   -0.361
##   VAA_LR ~~
##     CS_GT   (r.1.)   -0.019       NA                     -0.135   -0.135
##   VAA_GT ~~
##     CS_LR   (r.2.)   -0.001       NA                     -0.013   -0.013
```

```
##  .y4 ~~
##    .C1_4             0.599       NA                          0.599    0.627
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22             3.034       NA                          3.034    2.276
##    .y23             3.299       NA                          3.299    2.588
##    .y26             3.071       NA                          3.071    2.268
##    .y27             4.080       NA                          4.080    4.121
##    .y9              3.935       NA                          3.935    3.910
##    .y19             2.751       NA                          2.751    2.065
##    .y4              1.461       NA                          1.461    1.496
##    .y5              1.590       NA                          1.590    1.854
##    .y21             2.605       NA                          2.605    2.080
##    .C1_7            2.213       NA                          2.213    2.311
##    .C1_8            3.069       NA                          3.069    2.727
##    .C1_1            3.298       NA                          3.298    3.212
##    .C1_4            1.357       NA                          1.357    1.333
##    .C1_6            3.220       NA                          3.220    3.073
##    .C1_10           1.402       NA                          1.402    2.031
##     VAA_LR          0.000                                   0.000    0.000
##     VAA_GT          0.000                                   0.000    0.000
##     CS_LR           0.000                                   0.000    0.000
##     CS_GT           0.000                                     NaN      NaN
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22             1.293       NA                          1.293    0.728
##    .y23             1.257       NA                          1.257    0.773
##    .y26             1.309       NA                          1.309    0.713
##    .y27             0.942       NA                          0.942    0.961
##    .y9              0.846       NA                          0.846    0.835
##    .y19             1.561       NA                          1.561    0.879
##    .y4              0.863       NA                          0.863    0.905
##    .y5              0.713       NA                          0.713    0.970
##    .y21             0.892       NA                          0.892    0.569
##    .C1_7            0.833       NA                          0.833    0.908
##    .C1_8           -1.394       NA                         -1.394   -1.101
##    .C1_1            1.097       NA                          1.097    1.040
##    .C1_4            1.058       NA                          1.058    1.020
##    .C1_6            1.099       NA                          1.099    1.001
##    .C1_10           1.885       NA                          1.885    3.955
##     VAA_LR          0.484       NA                          1.000    1.000
##     VAA_GT          0.091       NA                          1.000    1.000
##     CS_LR           0.084       NA                          1.000    1.000
##     CS_GT          -0.042       NA                            NaN      NaN
##
##
## Group 6 [SDP]:
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR =~
##     y22             1.000                                   0.601    0.558
```

```
##     y23               0.221       NA                              0.133    0.231
##     y26               0.466       NA                              0.280    0.474
##     y27               0.283       NA                              0.170    0.166
##     y9                0.903       NA                              0.543    0.411
##     y19               0.495       NA                              0.298    0.543
##   VAA_GT =~
##     y4                1.000                                       0.514    0.486
##     y5                1.369       NA                              0.704    0.584
##     y21               0.953       NA                              0.490    0.407
##   CS_LR =~
##     C1_7              1.000                                       0.463    0.587
##     C1_8              0.739       NA                              0.343    0.535
##   CS_GT =~
##     C1_1              1.000                                       0.446    0.512
##     C1_4              1.504       NA                              0.671    0.632
##     C1_6              1.117       NA                              0.498    0.486
##     C1_10             0.988       NA                              0.441    0.583
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR ~~
##     VAA_GT  (r.VA)    0.085       NA                              0.276    0.276
##   CS_LR ~~
##     CS_GT   (r.CS)    0.004       NA                              0.021    0.021
##   VAA_LR ~~
##     CS_LR   (r.LR)    0.073       NA                              0.262    0.262
##   VAA_GT ~~
##     CS_GT   (r.GT)    0.217       NA                              0.945    0.945
##   VAA_LR ~~
##     CS_GT   (r.1.)    0.072       NA                              0.269    0.269
##   VAA_GT ~~
##     CS_LR   (r.2.)    0.017       NA                              0.072    0.072
##  .y4 ~~
##    .C1_4              0.422       NA                              0.422    0.554
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               1.879       NA                              1.879    1.744
##    .y23               1.328       NA                              1.328    2.304
##    .y26               1.301       NA                              1.301    2.199
##    .y27               3.744       NA                              3.744    3.652
##    .y9                3.047       NA                              3.047    2.304
##    .y19               1.160       NA                              1.160    2.115
##    .y4                1.784       NA                              1.784    1.686
##    .y5                2.798       NA                              2.798    2.321
##    .y21               2.396       NA                              2.396    1.989
##    .C1_7              1.805       NA                              1.805    2.287
##    .C1_8              1.455       NA                              1.455    2.273
##    .C1_1              3.874       NA                              3.874    4.453
##    .C1_4              1.596       NA                              1.596    1.502
##    .C1_6              3.386       NA                              3.386    3.303
##    .C1_10             2.067       NA                              2.067    2.737
##     VAA_LR            0.000                                       0.000    0.000
##     VAA_GT            0.000                                       0.000    0.000
```

```
##     CS_LR             0.000                                 0.000    0.000
##     CS_GT             0.000                                 0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               0.799       NA                      0.799    0.688
##    .y23               0.315       NA                      0.315    0.947
##    .y26               0.271       NA                      0.271    0.776
##    .y27               1.022       NA                      1.022    0.972
##    .y9                1.455       NA                      1.455    0.831
##    .y19               0.212       NA                      0.212    0.705
##    .y4                0.855       NA                      0.855    0.764
##    .y5                0.958       NA                      0.958    0.659
##    .y21               1.211       NA                      1.211    0.835
##    .C1_7              0.408       NA                      0.408    0.655
##    .C1_8              0.293       NA                      0.293    0.714
##    .C1_1              0.558       NA                      0.558    0.737
##    .C1_4              0.678       NA                      0.678    0.601
##    .C1_6              0.803       NA                      0.803    0.764
##    .C1_10             0.376       NA                      0.376    0.660
##     VAA_LR            0.362       NA                      1.000    1.000
##     VAA_GT            0.264       NA                      1.000    1.000
##     CS_LR             0.215       NA                      1.000    1.000
##     CS_GT             0.199       NA                      1.000    1.000
##
##
## Group 7 [VAS]:
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR =~
##     y22               1.000                               0.162    0.203
##     y23               0.993       NA                      0.161    0.281
##     y26               0.443       NA                      0.072    0.339
##     y27               3.421       NA                      0.554    0.489
##     y9                4.646       NA                      0.753    0.601
##     y19               0.138       NA                      0.022    0.063
##   VAA_GT =~
##     y4                1.000                               0.582    0.563
##     y5                1.683       NA                      0.979    0.841
##     y21               1.103       NA                      0.642    0.544
##   CS_LR =~
##     C1_7              1.000                               0.336    0.444
##     C1_8              1.306       NA                      0.439    0.677
##   CS_GT =~
##     C1_1              1.000                               0.650    0.665
##     C1_4              0.629       NA                      0.409    0.442
##     C1_6              0.993       NA                      0.645    0.583
##     C1_10             0.711       NA                      0.462    0.472
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR ~~
##     VAA_GT  (r.VA)    0.063       NA                      0.669    0.669
```

```
##   CS_LR ~~
##     CS_GT   (r.CS)    0.038       NA                       0.176    0.176
##   VAA_LR ~~
##     CS_LR   (r.LR)    0.035       NA                       0.640    0.640
##   VAA_GT ~~
##     CS_GT   (r.GT)    0.336       NA                       0.887    0.887
##   VAA_LR ~~
##     CS_GT   (r.1.)    0.065       NA                       0.613    0.613
##   VAA_GT ~~
##     CS_LR   (r.2.)    0.020       NA                       0.103    0.103
##  .y4 ~~
##    .C1_4              0.338       NA                       0.338    0.476
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               1.363       NA                       1.363    1.706
##    .y23               1.227       NA                       1.227    2.142
##    .y26               1.051       NA                       1.051    4.958
##    .y27               2.283       NA                       2.283    2.013
##    .y9                2.320       NA                       2.320    1.852
##    .y19               1.069       NA                       1.069    3.015
##    .y4                1.539       NA                       1.539    1.488
##    .y5                2.028       NA                       2.028    1.740
##    .y21               2.086       NA                       2.086    1.769
##    .C1_7              1.649       NA                       1.649    2.182
##    .C1_8              1.209       NA                       1.209    1.867
##    .C1_1              3.700       NA                       3.700    3.790
##    .C1_4              1.333       NA                       1.333    1.440
##    .C1_6              3.377       NA                       3.377    3.053
##    .C1_10             2.252       NA                       2.252    2.302
##     VAA_LR            0.000                                0.000    0.000
##     VAA_GT            0.000                                0.000    0.000
##     CS_LR             0.000                                0.000    0.000
##     CS_GT             0.000                                0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               0.613       NA                       0.613    0.959
##    .y23               0.302       NA                       0.302    0.921
##    .y26               0.040       NA                       0.040    0.885
##    .y27               0.979       NA                       0.979    0.761
##    .y9                1.002       NA                       1.002    0.639
##    .y19               0.125       NA                       0.125    0.996
##    .y4                0.731       NA                       0.731    0.683
##    .y5                0.398       NA                       0.398    0.293
##    .y21               0.980       NA                       0.980    0.704
##    .C1_7              0.459       NA                       0.459    0.802
##    .C1_8              0.227       NA                       0.227    0.541
##    .C1_1              0.531       NA                       0.531    0.557
##    .C1_4              0.690       NA                       0.690    0.805
##    .C1_6              0.807       NA                       0.807    0.660
##    .C1_10             0.744       NA                       0.744    0.777
##     VAA_LR            0.026       NA                       1.000    1.000
##     VAA_GT            0.339       NA                       1.000    1.000
```

```
##     CS_LR             0.113       NA                              1.000    1.000
##     CS_GT             0.422       NA                              1.000    1.000
##
##
## Group 8 [VIHR]:
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR =~
##     y22               1.000                               0.813    0.636
##     y23               0.150       NA                      0.122    0.105
##     y26               0.362       NA                      0.294    0.339
##     y27               0.513       NA                      0.417    0.354
##     y9                0.892       NA                      0.725    0.584
##     y19               0.559       NA                      0.454    0.415
##   VAA_GT =~
##     y4                1.000                               0.174    0.384
##     y5                3.018       NA                      0.525    0.576
##     y21               1.341       NA                      0.233    0.424
##   CS_LR =~
##     C1_7              1.000                               0.451    0.553
##     C1_8              0.547       NA                      0.247    0.328
##   CS_GT =~
##     C1_1              1.000                               0.542    0.615
##     C1_4              0.532       NA                      0.289    0.499
##     C1_6              1.185       NA                      0.643    0.588
##     C1_10             0.478       NA                      0.260    0.445
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   VAA_LR ~~
##     VAA_GT  (r.VA)    0.087       NA                      0.614    0.614
##   CS_LR ~~
##     CS_GT   (r.CS)   -0.194       NA                     -0.794   -0.794
##   VAA_LR ~~
##     CS_LR   (r.LR)    0.060       NA                      0.164    0.164
##   VAA_GT ~~
##     CS_GT   (r.GT)    0.116       NA                      1.231    1.231
##   VAA_LR ~~
##     CS_GT   (r.1.)    0.213       NA                      0.483    0.483
##   VAA_GT ~~
##     CS_LR   (r.2.)   -0.043       NA                     -0.542   -0.542
##  .y4 ~~
##    .C1_4              0.062       NA                      0.062    0.295
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               2.450       NA                      2.450    1.916
##    .y23               2.096       NA                      2.096    1.810
##    .y26               1.580       NA                      1.580    1.818
##    .y27               3.560       NA                      3.560    3.019
##    .y9                3.028       NA                      3.028    2.437
##    .y19               1.957       NA                      1.957    1.787
##    .y4                1.146       NA                      1.146    2.527
```

```
##    .y5                1.704       NA                            1.704    1.870
##    .y21               1.216       NA                            1.216    2.210
##    .C1_7              1.920       NA                            1.920    2.355
##    .C1_8              1.808       NA                            1.808    2.407
##    .C1_1              3.571       NA                            3.571    4.049
##    .C1_4              1.129       NA                            1.129    1.953
##    .C1_6              3.027       NA                            3.027    2.771
##    .C1_10             1.591       NA                            1.591    2.725
##     VAA_LR            0.000                                     0.000    0.000
##     VAA_GT            0.000                                     0.000    0.000
##     CS_LR             0.000                                     0.000    0.000
##     CS_GT             0.000                                     0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .y22               0.975       NA                            0.975    0.596
##    .y23               1.326       NA                            1.326    0.989
##    .y26               0.669       NA                            0.669    0.885
##    .y27               1.217       NA                            1.217    0.875
##    .y9                1.018       NA                            1.018    0.659
##    .y19               0.994       NA                            0.994    0.828
##    .y4                0.175       NA                            0.175    0.853
##    .y5                0.555       NA                            0.555    0.668
##    .y21               0.248       NA                            0.248    0.820
##    .C1_7              0.461       NA                            0.461    0.694
##    .C1_8              0.503       NA                            0.503    0.892
##    .C1_1              0.483       NA                            0.483    0.622
##    .C1_4              0.251       NA                            0.251    0.751
##    .C1_6              0.780       NA                            0.780    0.654
##    .C1_10             0.274       NA                            0.274    0.802
##     VAA_LR            0.661       NA                            1.000    1.000
##     VAA_GT            0.030       NA                            1.000    1.000
##     CS_LR             0.203       NA                            1.000    1.000
##     CS_GT             0.294       NA                            1.000    1.000
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##     mean.r.VAA       -0.000                                    -0.002   -0.002
##     mean.r.CS         0.038                                     0.452    0.452
##     mean.r.LR         0.094                                     1.113    1.113
##     mean.r.GT        -0.000                                    -0.031   -0.031
##     mean.r.d1         0.051                                     0.386    0.386
##     mean.r.d2         0.000                                     0.018    0.018
##     test.H3           0.043                                     0.661    0.661
##     test.H4          -0.051                                    -0.483   -0.483
```

Try to fit the model separately for each group

```r
fit_H3H4.re.KD<-cfa(model=model_H1H2.re,
                    data=dat2011.party,
                    group=c("puolue"),
                    group.label=c("KD"),
                    missing="fiml")
```

Model for KD

```
## Warning in lav_model_estimate(lavmodel = lavmodel, lavpartable = lavpartable, : lavaan WARNING: the o
##                    but not all elements of the gradient are (near) zero;
##                    the optimizer may not have found a local solution
##                    use check.gradient = FALSE to skip this check.
```

Model for KD does not converge

Fit is poor

Print standardized estimates to test the difference between correlations

```r
std.est_H3H4.re.KD<-standardizedsolution(fit_H3H4.re.KD)
```

```
## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in computeOmega(Sigma.hat = Sigma.hat, Mu.hat = Mu.hat, lavsamplestats = lavsamplestats, : la
```

```
## Error in chol.default(S) :
##   the leading minor of order 13 is not positive definite
```

```r
std.est_H3H4.re.KD[std.est_H3H4.re.KD$op==":=" |
               std.est_H3H4.re.KD$op=="~~" &
               std.est_H3H4.re.KD$lhs!=std.est_H3H4.re.KD$rhs,]
```

```
##        lhs op                            rhs est.std se  z pvalue ci.lower
## 21  VAA_LR ~~                         VAA_GT   0.122 NA NA     NA       NA
## 22   CS_LR ~~                          CS_GT  -0.028 NA NA     NA       NA
## 23  VAA_LR ~~                          CS_LR  -0.071 NA NA     NA       NA
## 24  VAA_GT ~~                          CS_GT   3.010 NA NA     NA       NA
## 25  VAA_LR ~~                          CS_GT   0.480 NA NA     NA       NA
## 26  VAA_GT ~~                          CS_LR   0.022 NA NA     NA       NA
## 27      y4 ~~                           C1_4   0.510 NA NA     NA       NA
## 76 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)  -0.551 NA NA     NA       NA
## 77 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   2.530 NA NA     NA       NA
##    ci.upper
## 21       NA
## 22       NA
## 23       NA
## 24       NA
## 25       NA
## 26       NA
## 27       NA
## 76       NA
## 77       NA
```

```r
fit_H3H4.re.KESK<-cfa(model=model_H1H2.re,
                      data=dat2011.party,
                      group=c("puolue"),
                      group.label=c("KESK"),
                      missing="fiml")
```

Model for KESK Model for KESK does converge

```r
round(inspect(fit_H3H4.re.KESK,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  67.000 163.000 287.143   0.000   0.662   0.606   0.059   0.114
```

Fit is poor

Print standardized estimates to test the difference between correlations

```r
std.est_H3H4.re.KESK<-standardizedsolution(fit_H3H4.re.KESK)
std.est_H3H4.re.KESK[std.est_H3H4.re.KESK$op==":=" |
               std.est_H3H4.re.KESK$op=="~~" &
               std.est_H3H4.re.KESK$lhs!=std.est_H3H4.re.KESK$rhs,]
```

```
##        lhs op                           rhs est.std    se      z pvalue
## 21  VAA_LR ~~                        VAA_GT   0.254 0.113  2.250  0.024
## 22   CS_LR ~~                         CS_GT   0.241 0.276  0.873  0.383
## 23  VAA_LR ~~                         CS_LR  -0.192 0.348 -0.552  0.581
## 24  VAA_GT ~~                         CS_GT   0.884 0.092  9.593  0.000
## 25  VAA_LR ~~                         CS_GT   0.392 0.176  2.229  0.026
## 26  VAA_GT ~~                         CS_LR   0.025 0.250  0.100  0.921
## 27      y4 ~~                          C1_4   0.766 0.063 12.131  0.000
## 76 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)  -0.584 0.351 -1.665  0.096
## 77 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.492 0.209  2.353  0.019
##    ci.lower ci.upper
## 21    0.033    0.475
## 22   -0.300    0.781
## 23   -0.874    0.490
## 24    0.704    1.065
## 25    0.047    0.737
## 26   -0.465    0.515
## 27    0.643    0.890
## 76   -1.272    0.104
## 77    0.082    0.902
```

```
fit_H3H4.re.KOK<-cfa(model=model_H1H2.re,
                    data=dat2011.party,
                    group=c("puolue"),
                    group.label=c("KOK"),
                    missing="fiml")
```

Model for KOK

```
## Warning in lav_object_post_check(object): lavaan WARNING: covariance matrix of latent variables
##                 is not positive definite;
##                 use lavInspect(fit, "cov.lv") to investigate.
```

Model for KOK has problems

```
round(inspect(fit_H3H4.re.KOK,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  67.000 163.000 214.323   0.004   0.799   0.766   0.037   0.108
```

Fit is poor

Print standardized estimates to test the difference between correlations

```
std.est_H3H4.re.KOK<-standardizedsolution(fit_H3H4.re.KOK)
std.est_H3H4.re.KOK[std.est_H3H4.re.KOK$op==":=" |
               std.est_H3H4.re.KOK$op=="~~" &
               std.est_H3H4.re.KOK$lhs!=std.est_H3H4.re.KOK$rhs,]
```

```
##        lhs op                             rhs est.std    se      z pvalue
## 21  VAA_LR ~~                          VAA_GT   0.431 0.112  3.838  0.000
## 22   CS_LR ~~                           CS_GT   0.533 0.290  1.838  0.066
## 23  VAA_LR ~~                           CS_LR   0.348 0.243  1.431  0.152
## 24  VAA_GT ~~                           CS_GT   1.295 0.162  7.980  0.000
## 25  VAA_LR ~~                           CS_GT   0.512 0.187  2.730  0.006
## 26  VAA_GT ~~                           CS_LR   0.282 0.261  1.079  0.280
## 27      y4 ~~                            C1_4   0.549 0.100  5.497  0.000
## 76 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)  -0.185 0.359 -0.515  0.607
## 77 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.762 0.326  2.339  0.019
##    ci.lower ci.upper
## 21    0.211    0.652
## 22   -0.035    1.101
## 23   -0.129    0.824
## 24    0.977    1.613
## 25    0.144    0.879
## 26   -0.230    0.794
## 27    0.353    0.745
## 76   -0.889    0.519
## 77    0.124    1.401
```

Correlations larger than 1.00

```r
fit_H3H4.re.PS<-cfa(model=model_H1H2.re,
                    data=dat2011.party,
                    group=c("puolue"),
                    group.label=c("PS"),
                    missing="fiml")
```

Model for PS

```
## Warning in lav_object_post_check(object): lavaan WARNING: some estimated ov
## variances are negative
```

Model for PS has problems with negative variances

```r
round(inspect(fit_H3H4.re.PS,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  67.000 163.000 204.859   0.015   0.849   0.823   0.035   0.092
```

Fit is poor, but not very poor

Print standardized estimates to test the difference between correlations

```r
std.est_H3H4.re.PS<-standardizedsolution(fit_H3H4.re.PS)
std.est_H3H4.re.PS[std.est_H3H4.re.PS$op==":=" |
               std.est_H3H4.re.PS$op=="~~" &
               std.est_H3H4.re.PS$lhs!=std.est_H3H4.re.PS$rhs,]
```

```
##        lhs op                          rhs est.std    se      z pvalue
## 21  VAA_LR ~~                       VAA_GT   0.380 0.132  2.880  0.004
## 22   CS_LR ~~                        CS_GT  -0.042 0.122 -0.345  0.730
## 23  VAA_LR ~~                        CS_LR   0.300 0.242  1.242  0.214
## 24  VAA_GT ~~                        CS_GT   0.921 0.158  5.834  0.000
## 25  VAA_LR ~~                        CS_GT   0.378 0.143  2.639  0.008
## 26  VAA_GT ~~                        CS_LR  -0.077 0.154 -0.499  0.618
## 27      y4 ~~                         C1_4   0.403 0.096  4.194  0.000
## 76 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)  -0.080 0.282 -0.285  0.776
## 77 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.541 0.194  2.785  0.005
##    ci.lower ci.upper
## 21    0.122    0.639
## 22   -0.282    0.198
## 23   -0.174    0.774
## 24    0.612    1.231
## 25    0.097    0.658
## 26   -0.378    0.225
## 27    0.214    0.591
## 76   -0.632    0.472
## 77    0.160    0.921
```

```r
fit_H3H4.re.RKP<-cfa(model=model_H1H2.re,
                     data=dat2011.party,
                     group=c("puolue"),
                     group.label=c("RKP"),
                     missing="fiml")
```

Model for RKP

```
## Warning in lav_object_post_check(object): lavaan WARNING: some estimated ov
## variances are negative
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: some estimated lv
## variances are negative
```

Model for RKP converges, but has problem with negative variances

```r
round(inspect(fit_H3H4.re.RKP,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  67.000 163.000 251.461   0.000   0.463   0.374   0.084   0.132
```

Fit is poor

Print standardized estimates to test the difference between correlations

```r
std.est_H3H4.re.RKP<-standardizedsolution(fit_H3H4.re.RKP)
```

```
## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced

## Warning in sqrt(ETA2): NaNs produced
```

```
std.est_H3H4.re.RKP[std.est_H3H4.re.RKP$op==":=" |
               std.est_H3H4.re.RKP$op=="~~" &
               std.est_H3H4.re.RKP$lhs!=std.est_H3H4.re.RKP$rhs,]
```

```
##        lhs op                            rhs est.std    se      z pvalue
## 21  VAA_LR ~~                         VAA_GT   0.352 0.245  1.439  0.150
## 22   CS_LR ~~                          CS_GT   0.069 0.238  0.289  0.773
## 23  VAA_LR ~~                          CS_LR  -0.269 0.281 -0.960  0.337
## 24  VAA_GT ~~                          CS_GT   0.421 0.289  1.454  0.146
## 25  VAA_LR ~~                          CS_GT   0.656 0.251  2.611  0.009
## 26  VAA_GT ~~                          CS_LR   0.003 0.093  0.037  0.970
## 27      y4 ~~                           C1_4   0.556 0.102  5.463  0.000
## 76 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)  -0.925 0.370 -2.505  0.012
## 77 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)  -0.235 0.369 -0.638  0.524
##    ci.lower ci.upper
## 21   -0.128    0.832
## 22   -0.398    0.535
## 23   -0.819    0.281
## 24   -0.146    0.988
## 25    0.164    1.149
## 26   -0.179    0.186
## 27    0.357    0.756
## 76   -1.650   -0.201
## 77   -0.959    0.488
```

```r
fit_H3H4.re.SDP<-cfa(model=model_H1H2.re,
                    data=dat2011.party,
                    group=c("puolue"),
                    group.label=c("SDP"),
                    missing="fiml")
```

Model for SDP Model for SDP converges

```r
round(inspect(fit_H3H4.re.SDP,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  67.000 163.000 227.918   0.001   0.781   0.745   0.042   0.095
```

Fit is poor

Print standardized estimates to test the difference between correlations

```r
std.est_H3H4.re.SDP<-standardizedsolution(fit_H3H4.re.SDP)
std.est_H3H4.re.SDP[std.est_H3H4.re.SDP$op==":=" |
               std.est_H3H4.re.SDP$op=="~~" &
               std.est_H3H4.re.SDP$lhs!=std.est_H3H4.re.SDP$rhs,]
```

```
##        lhs op                            rhs est.std    se     z pvalue
## 21  VAA_LR ~~                         VAA_GT   0.195 0.126 1.542  0.123
## 22   CS_LR ~~                          CS_GT   0.174 0.211 0.824  0.410
## 23  VAA_LR ~~                          CS_LR   0.328 0.241 1.364  0.173
## 24  VAA_GT ~~                          CS_GT   0.932 0.097 9.602  0.000
## 25  VAA_LR ~~                          CS_GT   0.298 0.152 1.961  0.050
## 26  VAA_GT ~~                          CS_LR   0.138 0.233 0.593  0.553
## 27      y4 ~~                           C1_4   0.537 0.094 5.695  0.000
## 76 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)   0.031 0.269 0.114  0.910
## 77 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.635 0.186 3.411  0.001
##    ci.lower ci.upper
## 21   -0.053    0.442
## 22   -0.240    0.589
## 23   -0.143    0.800
## 24    0.742    1.123
## 25    0.000    0.595
## 26   -0.319    0.595
## 27    0.352    0.722
## 76   -0.496    0.558
## 77    0.270    0.999
```

```r
fit_H3H4.re.VAS<-cfa(model=model_H1H2.re,
                     data=dat2011.party,
                     group=c("puolue"),
                     group.label=c("VAS"),
                     missing="fiml")
```

Model for VAS

```
## Warning in lav_object_post_check(object): lavaan WARNING: covariance matrix of latent variables
##                 is not positive definite;
##                 use lavInspect(fit, "cov.lv") to investigate.
```

Model for VAS has problems

```r
round(inspect(fit_H3H4.re.VAS,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  67.000 163.000 260.989   0.000   0.760   0.720   0.053   0.113
```

Print standardized estimates to test the difference between correlations

```r
std.est_H3H4.re.VAS<-standardizedsolution(fit_H3H4.re.VAS)
std.est_H3H4.re.VAS[std.est_H3H4.re.VAS$op==":=" |
               std.est_H3H4.re.VAS$op=="~~" &
               std.est_H3H4.re.VAS$lhs!=std.est_H3H4.re.VAS$rhs,]
```

```
##        lhs op                          rhs est.std    se      z pvalue
## 21  VAA_LR ~~                       VAA_GT   0.658 0.097  6.786  0.000
## 22   CS_LR ~~                        CS_GT   1.107 0.603  1.835  0.067
## 23  VAA_LR ~~                        CS_LR   1.401 0.724  1.934  0.053
## 24  VAA_GT ~~                        CS_GT   0.965 0.071 13.635  0.000
## 25  VAA_LR ~~                        CS_GT   0.582 0.149  3.908  0.000
## 26  VAA_GT ~~                        CS_LR   0.716 0.498  1.439  0.150
## 27      y4 ~~                         C1_4   0.437 0.099  4.430  0.000
## 76 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)   0.294 0.430  0.683  0.495
## 77 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)  -0.142 0.607 -0.234  0.815
##    ci.lower ci.upper
## 21    0.468    0.848
## 22   -0.076    2.290
## 23   -0.019    2.821
## 24    0.826    1.104
## 25    0.290    0.875
## 26   -0.259    1.692
## 27    0.244    0.631
## 76   -0.550    1.137
## 77   -1.332    1.047
```

Correlations are impossible

```
fit_H3H4.re.VIHR<-cfa(model=model_H1H2.re,
                     data=dat2011.party,
                     group=c("puolue"),
                     group.label=c("VIHR"),
                     missing="fiml")
```

Model for VIHR

```
## Warning in lav_model_estimate(lavmodel = lavmodel, lavpartable = lavpartable, : lavaan WARNING: the o
##                      but not all elements of the gradient are (near) zero;
##                      the optimizer may not have found a local solution
##                      use check.gradient = FALSE to skip this check.
```

Model for VIHR does not converge

Fit is poor

Print standardized estimates to test the difference between correlations

```
std.est_H3H4.re.VIHR<-standardizedsolution(fit_H3H4.re.VIHR)
```

```
## Warning in sqrt(ETA2): NaNs produced
```

```
## Warning in computeOmega(Sigma.hat = Sigma.hat, Mu.hat = Mu.hat, lavsamplestats = lavsamplestats, : la
```

```
## Error in chol.default(S) :
##   the leading minor of order 13 is not positive definite
```

```
std.est_H3H4.re.VIHR[std.est_H3H4.re.VIHR$op==":=" |
               std.est_H3H4.re.VIHR$op=="~~" &
               std.est_H3H4.re.VIHR$lhs!=std.est_H3H4.re.VIHR$rhs,]
```

```
##        lhs op                            rhs est.std se  z pvalue ci.lower
## 21  VAA_LR ~~                         VAA_GT   0.556 NA NA     NA       NA
## 22   CS_LR ~~                          CS_GT   0.004 NA NA     NA       NA
## 23  VAA_LR ~~                          CS_LR  -0.011 NA NA     NA       NA
## 24  VAA_GT ~~                          CS_GT   1.144 NA NA     NA       NA
## 25  VAA_LR ~~                          CS_GT   0.591 NA NA     NA       NA
## 26  VAA_GT ~~                          CS_LR  -0.001 NA NA     NA       NA
## 27      y4 ~~                           C1_4   0.317 NA NA     NA       NA
## 76 test.H1 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)  -0.602 NA NA     NA       NA
## 77 test.H2 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.553 NA NA     NA       NA
##    ci.upper
## 21       NA
## 22       NA
## 23       NA
## 24       NA
## 25       NA
## 26       NA
## 27       NA
## 76       NA
## 77       NA
```

Summary of H3-H4 with MG-CFA approach

The configural model did not converge, even after respecification. Single group model also were non-converging or had other type of problems, except for KD and KOK, for which the fit of the model nevertheless was poor, and therefore not interpretable.

This most likely is an indication that the sample sizes of the parties are too small for this model with 21 indicators and 4 factors.

The alternative way to test hypotheses 4-6 is presented below. It unconfounds the associations in the model by using party-mean centered observed variables for estimating the similar type of model that was used for H1 and H2, and H5, respectively. Because this approach does not have any grouping structure, it uses the overall sample size for the eight parties, which is 1572. It is nevertheless only conducted among the eight focal parties, and other parties are excluded. Because the misspecification in the model with centered variables might be entirely different to raw score variables, the modeling is again started with no residual correlations and they are examined if the fit of the model is inadequate.

H3 and H4 with group-mean centered variables and no grouping structure

Estimate how much of the variation in each item is between-groups

```
#there was problems running the mult.icc function to the data structure so
#data observed data was extracted from one of the previously fitted models
#to get rid of all labels etc.

num.dat.2011<-data.frame(fit_H1H2@Data@X,dat2011$puolue)
names(num.dat.2011)<-c(fit_H1H2@Data@ov$name,"puolue")
num.dat.2011<-num.dat.2011 %>%
  filter(puolue=="KD" |
           puolue=="KESK" |
           puolue=="KOK" |
           puolue=="PS" |
           puolue=="RKP" |
           puolue=="SDP" |
           puolue=="VAS" |
           puolue=="VIHR")


ICC<-data.frame(multilevel::mult.icc(x=num.dat.2011[,obs_items[2:length(obs_items)]],
                                     grpid=num.dat.2011$puolue))
ICC[,2:3]<-round(ICC[,2:3],3)
ICC
```

```
##    Variable  ICC1  ICC2
## 1       y22 0.338 0.990
## 2       y23 0.385 0.992
## 3       y26 0.421 0.993
## 4       y27 0.318 0.989
## 5        y9 0.326 0.990
## 6       y19 0.379 0.992
## 7        y4 0.525 0.995
## 8        y5 0.531 0.996
## 9        y1 0.505 0.995
## 10      y21 0.319 0.989
## 11     C1_2 0.078 0.943
## 12     C1_7 0.071 0.937
## 13     C1_8 0.477 0.994
## 14     C1_1 0.266 0.986
## 15     C1_3 0.205 0.981
## 16     C1_4 0.479 0.994
## 17     C1_5 0.054 0.918
## 18     C1_6 0.137 0.969
## 19    C1_10 0.428 0.993
## 20    C1_11 0.480 0.995
```

```
describe(ICC$ICC1,fast=T)
```

```
##    vars  n mean   sd  min  max range   se
## X1    1 20 0.34 0.16 0.05 0.53  0.48 0.03
```

```
ICC$label<-get_label(df2011[,as.character(ICC[,1])])

#export to .csv file
```

```r
write.csv2(ICC,"ICC_2011.csv")
```

ICC gives the proportion (%) of variance that is between the parties. There is quite a lot between-party variance.

Center all observed variables

```r
ind.items<-obs_items[2:length(obs_items)]

dat2011.gmc<-data.frame(dat2011.party)

na.mean<-function(var){
  mean(var,na.rm=T)
}

group.means<-dat2011.gmc %>%
  group_by(puolue) %>%
  summarise_at(ind.items,na.mean)

dat2011.gmc<-left_join(x=dat2011.gmc,
                       y=group.means,
                       by=c("puolue"),
                       suffix=c("",".pm"))

ind.items %in% names(dat2011.gmc)
```

```
##  [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [16] TRUE TRUE TRUE TRUE TRUE
```

```r
paste0(ind.items,".pm") %in% names(dat2011.gmc)
```

```
##  [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [16] TRUE TRUE TRUE TRUE TRUE
```

```r
for(i in 1:length(ind.items)){
  dat2011.gmc[,which(names(dat2011.gmc)==ind.items[i])]<-
    dat2011.gmc[,which(names(dat2011.gmc)==ind.items[i])]-
    dat2011.gmc[,which(names(dat2011.gmc)==paste0(ind.items[i],".pm"))]
}


psych::describe(dat2011.gmc[,ind.items],fast=T)
```

```
##       vars    n mean   sd   min  max range   se
## y22      1 1479    0 1.17 -2.85 3.64  6.49 0.03
## y23      2 1482    0 1.07 -2.58 3.78  6.37 0.03
## y26      3 1494    0 0.89 -2.08 3.48  5.56 0.02
## y27      4 1496    0 1.03 -3.46 2.74  6.20 0.03
## y9       5 1490    0 1.07 -3.53 2.72  6.25 0.03
## y19      6 1495    0 1.07 -2.35 3.93  6.29 0.03
## y4       7 1490    0 1.16 -3.44 3.53  6.97 0.03
## y5       8 1496    0 1.05 -2.87 3.30  6.17 0.03
## y1       9 1503    0 1.00 -3.80 2.69  6.49 0.03
## y21     10 1439    0 1.15 -2.61 3.79  6.40 0.03
## C1_2    11  639    0 0.91 -1.16 3.50  4.65 0.04
## C1_7    12  637    0 0.84 -1.41 3.20  4.61 0.03
```

```
## C1_8    13  638    0 0.81 -2.08 3.75  5.83 0.03
## C1_1    14  639    0 0.77 -2.88 1.70  4.58 0.03
## C1_3    15  640    0 0.96 -1.64 3.89  5.53 0.04
## C1_4    16  636    0 1.18 -3.39 3.87  7.25 0.05
## C1_5    17  639    0 1.09 -2.76 2.00  4.76 0.04
## C1_6    18  639    0 0.97 -3.17 1.98  5.15 0.04
## C1_10   19  640    0 0.84 -2.75 2.80  5.55 0.03
## C1_11   20  633    0 0.81 -2.56 3.75  6.31 0.03
```

Define the model

Use the subset of items used for H1 and H2

```
model_H3H4<-"
#loadings
VAA_LR=~y22+y23+y26+y27+y9+y19
VAA_GT=~y4+y5+y21
CS_LR=~C1_7+C1_8
CS_GT=~C1_1+C1_4+C1_6+C1_10

#latent correlations

#cross-dimension same-method
VAA_LR~~r.VAA*VAA_GT
CS_LR~~r.CS*CS_GT

#concurrent validity
VAA_LR~~r.LR*CS_LR
VAA_GT~~r.GT*CS_GT

#cross-dimension cross-method correlations
VAA_LR~~r.d1*CS_GT
VAA_GT~~r.d2*CS_LR

#custom parameters
test.H3:=r.LR-max(r.VAA,r.CS,r.d1,r.d2)
test.H4:=r.GT-max(r.VAA,r.CS,r.d1,r.d2)

"
```

Fit the model

```
fit_H3H4<-cfa(model=model_H3H4,
              data=dat2011.gmc,
              missing="fiml")
```

```
## Warning in lav_data_full(data = data, group = group, cluster = cluster, : lavaan WARNING: some cases
##   452
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: covariance matrix of latent variables
##                  is not positive definite;
##                  use lavInspect(fit, "cov.lv") to investigate.
```

Problems with the covariance structure, add the preregistered residual correlation.

```r
model_H3H4.re<-paste0(model_H3H4,
                      "y4~~C1_4\n")
```

```r
fit_H3H4.re<-cfa(model=model_H3H4.re,
              data=dat2011.gmc,
              missing="fiml")
```

Fit the respecified model

```
## Warning in lav_data_full(data = data, group = group, cluster = cluster, : lavaan WARNING: some cases
##   452
```

Problem was solved with the residual correlation

Inspect fit of the model

```r
round(inspect(fit_H3H4.re,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  52.000  83.000 268.505   0.000   0.873   0.840   0.038   0.048
```

The fit of the model is adequate.

Hypotheses 3 and 4

Print standardized estimates to test the difference between correlations

```r
std.est_H3H4.re<-standardizedsolution(fit_H3H4.re)
std.est_H3H4.re[std.est_H3H4.re$op==":=" |
               std.est_H3H4.re$op=="~~" &
               std.est_H3H4.re$lhs!=std.est_H3H4.re$rhs,]
```

```
##        lhs op                            rhs est.std    se      z pvalue
## 16  VAA_LR ~~                         VAA_GT   0.373 0.046  8.094  0.000
## 17   CS_LR ~~                          CS_GT   0.031 0.074  0.422  0.673
## 18  VAA_LR ~~                          CS_LR   0.424 0.093  4.582  0.000
## 19  VAA_GT ~~                          CS_GT   0.863 0.053 16.277  0.000
## 20  VAA_LR ~~                          CS_GT   0.439 0.063  6.939  0.000
## 21  VAA_GT ~~                          CS_LR   0.060 0.077  0.773  0.439
## 22      y4 ~~                           C1_4   0.587 0.028 21.117  0.000
## 61 test.H3 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)  -0.015 0.114 -0.129  0.897
## 62 test.H4 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.425 0.082  5.209  0.000
##    ci.lower ci.upper
## 16    0.283    0.463
## 17   -0.114    0.177
## 18    0.243    0.605
## 19    0.759    0.967
## 20    0.315    0.562
## 21   -0.092    0.211
## 22    0.532    0.641
## 61   -0.238    0.208
## 62    0.265    0.584
```

H3: There is a moderately strong (.424, p < .001) correlation between VAA-LR and CS-LR, and it is not stronger (difference in correlations -.015, p = .897) than the strongest of correlations between different dimensions (.439 between VAA_LR and CS_GT, p < .001)

H4: There is a strong (.863, $p < .001$) correlation between VAA-GT and CS-GT, and it is notably stronger (difference in correlations .425, $p < .001$) than the strongest of correlations between different dimensions (.439 between VAA_LR and CS_GT, $p < .001$)

```
mis.H3H4<-miPowerFit(fit_H3H4.re,stdLoad=.40,cor=.20)
mis.H3H4<-mis.H3H4[mis.H3H4$op=="=~" |  mis.H3H4$op=="~~",]
#see summary of the decisions
table(mis.H3H4$decision.pow)
```

Exploratory for H3 and H4: Seek misspecification to improve the overall model fit

```
##
##  EPC:M EPC:NM      I      M     NM
##      2     33      4      4    105
```

```
#there are 6 misspecifications

rounded.vars<-c("mi","epc","target.epc",
                "std.epc","se.epc")

num.round<-function(var){
  var<-as.numeric(var)
  var<-round(var,2)
  return(var)
}

mis.H3H4[,rounded.vars]<-sapply(mis.H3H4[,rounded.vars],num.round)

printed.vars<-c("lhs","op","rhs","mi","epc","target.epc",
                "std.epc","std.target.epc","significant.mi",
                "high.power","decision.pow","se.epc")

#print the output

mis.H3H4 %>%
  filter(mis.H3H4$decision.pow=="M" |
                mis.H3H4$decision.pow=="EPC:M") %>%
  dplyr::select(all_of(printed.vars))
```

```
##      lhs op   rhs    mi   epc target.epc std.epc std.target.epc significant.mi
## 1 VAA_GT =~  C1_6 15.56 -1.81       0.86   -0.85            0.4           TRUE
## 2 VAA_GT =~ C1_10 24.92  1.87       0.74    1.01            0.4           TRUE
## 3  CS_GT =~    y5 21.85  4.99       0.93    2.14            0.4           TRUE
## 4  CS_GT =~   y21 11.85 -1.63       1.02   -0.64            0.4           TRUE
## 5    y26 ~~  C1_8 25.76  0.15       0.14    0.20            0.2           TRUE
## 6   C1_1 ~~  C1_6 21.63  0.15       0.15    0.20            0.2           TRUE
##   high.power decision.pow se.epc
## 1      FALSE            M   0.46
## 2      FALSE            M   0.37
## 3      FALSE            M   1.07
## 4      FALSE            M   0.47
## 5       TRUE        EPC:M   0.03
## 6       TRUE        EPC:M   0.03
```

All the proposed loadings would be cross-loadings across methods (from VAA to CS or vice versa), and therefore not applicable for the present approach. Also, the expected parameter changes are indicative that most of these respecification would be Heywood -cases (standardized loadings that would be larger than 1 in absolute magnitude).

There were two misspecified residual correlations: y26. "Taxation of high earners should be increased in the next electoral cycle (r.)" and C1_8. "Tuloja ja vaurautta pitäisi uudelleenjakaa tavallisten ihmisten suuntaan" and between C1.1 "Maahanmuuttajien pitäisi sopeutua suomalaisiin tapoihin" and "Lakia rikkovia ihmisiä pitäisi rangaista kovemmin"

Add these residual correlations to the model.

```
model_H3H4.exp.re<-paste0(model_H3H4.re,
                      "y26~~C1_8\n",
                      "C1_1~~C1_6\n")
```

```
fit_H3H4.exp.re<-cfa(model=model_H3H4.exp.re,
              data=dat2011.gmc,
              missing="fiml")
```

Exploratory respecification

```
## Warning in lav_data_full(data = data, group = group, cluster = cluster, : lavaan WARNING: some cases
##   452
```

Inspect fit of the model

```
round(inspect(fit_H3H4.re,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  52.000  83.000 268.505   0.000   0.873   0.840   0.038   0.048
```

```
round(inspect(fit_H3H4.exp.re,"fit")
      [c("npar","df","chisq","pvalue","cfi","tli","rmsea","srmr")],3)
```

```
##    npar      df   chisq  pvalue     cfi     tli   rmsea    srmr
##  54.000  81.000 219.828   0.000   0.905   0.877   0.033   0.042
```

The fit of the model is improved

Retest Hypotheses 4 and 5

Print standardized estimates to test the difference between correlations

```
std.est_H3H4.exp<-standardizedsolution(fit_H3H4.exp.re)
std.est_H3H4.exp[std.est_H3H4.exp$op==":=" |
               std.est_H3H4.exp$op=="~~" &
               std.est_H3H4.exp$lhs!=std.est_H3H4.exp$rhs,]
```

```
##        lhs op                            rhs est.std    se      z pvalue
## 16  VAA_LR ~~                         VAA_GT   0.373 0.046  8.153  0.000
## 17   CS_LR ~~                          CS_GT   0.080 0.085  0.947  0.344
## 18  VAA_LR ~~                          CS_LR   0.379 0.093  4.094  0.000
## 19  VAA_GT ~~                          CS_GT   0.963 0.056 17.167  0.000
## 20  VAA_LR ~~                          CS_GT   0.444 0.070  6.340  0.000
## 21  VAA_GT ~~                          CS_LR   0.080 0.081  0.986  0.324
## 22      y4 ~~                           C1_4   0.570 0.029 19.349  0.000
```

```
## 23      y26 ~~                           C1_8    0.274 0.055  4.942  0.000
## 24     C1_1 ~~                           C1_6    0.237 0.045  5.285  0.000
## 63 test.H3 := r.LR-max(r.VAA,r.CS,r.d1,r.d2)  -0.065 0.117 -0.561  0.575
## 64 test.H4 := r.GT-max(r.VAA,r.CS,r.d1,r.d2)   0.519 0.088  5.924  0.000
##    ci.lower ci.upper
## 16    0.283    0.462
## 17   -0.086    0.247
## 18    0.197    0.560
## 19    0.853    1.073
## 20    0.307    0.582
## 21   -0.079    0.240
## 22    0.512    0.628
## 23    0.165    0.383
## 24    0.149    0.325
## 63   -0.294    0.163
## 64    0.347    0.691
```

The results are virtually identical to those without the additional residual correlations.

H3: There is a moderately strong (.379, $p < .001$) correlation between VAA-LR and CS-LR, and it is not stronger (difference in correlations -.065, $p = .575$) than the strongest of correlations between different dimensions (.444 between VAA_LR and CS_GT, $p < .001$)

H4: There is a very strong (.963, $p < .001$) correlation between VAA-GT and CS-GT, and it is notably stronger (difference in correlations .519, $p < .001$) than the strongest of correlations between different dimensions (.444 between VAA_LR and CS_GT, $p < .001$)

Examine how self-placement on