Radicalized attitudes and intentions - LPA

Code

Deadpool

2024-09-24

Hide

```
sapply(c("readxl", "psych", "dplyr", "lavaan", "semTools", "tidySEM", "ggplot2", "carel
ess", "VIM"), library, character.only = T)
```

Analytical approach

Analyses in which participants aged 30 or less are retained.

Data preparations

Hide

```
baza <- read_excel("data_final_nolabs.xlsx")
nrow(baza)
```

```
## [1] 1048
```

Additional preparations

Hide

```
baza$grp <- ifelse(baza$Q1 == 1, "nonstudent", "student")
baza <- arrange(baza, Q1)
```

Excluding participants who stopped responding

In line with the Informed consent.

Hide

```
baza[baza == -3] <- NA
baza[baza == -2] <- NA
baza <- baza[complete.cases(baza[, 1:47]), ]
nrow(baza)
```

```
## [1] 922
```

Hide

```
baza[baza == -1] <- NA
baza <- baza[complete.cases(baza$spol), ]
set.seed(365)
bx <- kNN(baza, k = 7, trace = F, addRandom = T, addRF = T)
```

```
## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf

## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf
```

```
## Warning in min(weights[[i]]): no non-missing arguments to min; returning Inf
```

Hide

```
bx <- bx[, 1:51]
```

Excluding participants with repetitive responses

Participants providing the same response 10+ times in a row.

Hide

```
bx <- bx[longstring(bx) < 11, ]
nrow(bx)
```

```
## [1] 910
```

Excluding participants of third gender

Hide

```
bx <- subset(bx, spol < 3)
nrow(bx)
```

```
## [1] 901
```

Excluding participants outside the age range

Hide

```
bx$dob <- bx$dob + 17
bx <- subset(bx, dob < 30)
nrow(bx)
```

```
## [1] 823
```

Descriptives

Hide

```
describe(bx$dob)
```

```
##    vars   n  mean   sd median trimmed  mad min max range skew kurtosis   se
## X1    1 823 21.83 2.58     21   21.55 2.97  18  29    11 0.84     0.18 0.09
```

Hide

```
describe(bx$edu)
```

```
##    vars   n  mean   sd median trimmed  mad min max range  skew kurtosis  se
## X1    1 823 10.88 2.92     11   11.18 1.48   1  16    15 -1.55     3.63 0.1
```

Hide

```
prop.table(table(bx$grp))
```

```
##
## nonstudent    student
##  0.1166464  0.8833536
```

Hide

```
length(grep("psiho", bx$studij, ignore.case = T))
```

```
## [1] 24
```

Hide

```
length(grep("socio", bx$studij, ignore.case = T))
```

```
## [1] 6
```

Hide

```
length(grep("polito", bx$studij, ignore.case = T))
```

```
## [1] 3
```

Main analyses

Factor scores

Hide

```
mod <- 'int_rad =~ rada + radb + radc + radd
        int_act =~ acta + actb + actc + actd'

modmod <- sem(mod, estimator = "MLR", data = bx)
fitmeasures(modmod, fit.measures = c("cfi", "rmsea.robust", "rmsea.ci.lower.robust", "r
msea.ci.upper.robust", "srmr"))
```

```
##                   cfi          rmsea.robust rmsea.ci.lower.robust
##                 0.972                 0.083                 0.064
## rmsea.ci.upper.robust                  srmr
##                 0.102                 0.029
```

Hide

```
reliability(modmod)
```

```
##          int_rad   int_act
## alpha  0.9197430 0.8765525
## omega  0.9213580 0.8805766
## omega2 0.9213580 0.8805766
## omega3 0.9212322 0.8803422
## avevar 0.7465083 0.6514985
```

Hide

```
mod <- 'att_act =~ vatte + vattf + vattg + vatth'

modmod <- sem(mod, estimator = "MLR", data = bx)
fitmeasures(modmod, fit.measures = c("cfi", "rmsea.robust", "rmsea.ci.lower.robust", "r
msea.ci.upper.robust", "srmr"))
```

```
##                   cfi          rmsea.robust rmsea.ci.lower.robust
##                 0.958                 0.129                 0.083
## rmsea.ci.upper.robust                  srmr
##                 0.182                 0.042
```

Hide

```
reliability(modmod)
```

```
##          att_act
## alpha  0.6532366
## omega  0.6649193
## omega2 0.6649193
## omega3 0.6480873
## avevar 0.3597021
```

Hide

```
mod <- 'att_rad =~ vatta + vattb + vattc + vattd'

modmod <- sem(mod, estimator = "MLR", data = bx)
fitmeasures(modmod, fit.measures = c("cfi", "rmsea.robust", "rmsea.ci.lower.robust", "r
msea.ci.upper.robust", "srmr"))
```

```
##                   cfi          rmsea.robust rmsea.ci.lower.robust
##                 0.989                 0.089                 0.046
## rmsea.ci.upper.robust                  srmr
##                 0.139                 0.019
```

Hide

```
reliability(modmod)
```

```
##          att_rad
## alpha  0.8346999
## omega  0.8355334
## omega2 0.8355334
## omega3 0.8353507
## avevar 0.5603166
```

Hide

```
mod <- 'att_rad =~ vatta + vattb + vattc + vattd
        att_act =~ vatte + vattf + vattg + vatth
        int_rad =~ rada + radb + radc + radd
        int_act =~ acta + actb + actc + actd'

modmod <- sem(mod, estimator = "MLR", data = bx)
summary(modmod, standardized = T)
```

```
## lavaan 0.6.15 ended normally after 47 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        38
##
##   Number of observations                           823
##
## Model Test User Model:
##                                              Standard      Scaled
##   Test Statistic                              371.769     285.387
##   Degrees of freedom                               98          98
##   P-value (Chi-square)                          0.000       0.000
##   Scaling correction factor                                 1.303
##     Yuan-Bentler correction (Mplus variant)
##
## Parameter Estimates:
##
##   Standard errors                             Sandwich
##   Information bread                           Observed
##   Observed information based on                Hessian
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   att_rad =~
##     vatta             1.000                               1.053    0.684
##     vattb             1.160    0.055   21.138    0.000    1.222    0.755
##     vattc             1.169    0.063   18.478    0.000    1.231    0.785
##     vattd             1.173    0.056   20.899    0.000    1.236    0.762
##   att_act =~
##     vatte             1.000                               0.559    0.517
##     vattf             0.648    0.111    5.848    0.000    0.362    0.258
##     vattg             1.640    0.170    9.655    0.000    0.916    0.754
##     vatth             1.606    0.152   10.580    0.000    0.897    0.838
##   int_rad =~
##     rada              1.000                               0.975    0.925
##     radb              0.816    0.049   16.573    0.000    0.796    0.780
##     radc              0.925    0.036   25.568    0.000    0.902    0.878
##     radd              0.947    0.035   26.734    0.000    0.924    0.864
##   int_act =~
##     acta              1.000                               1.375    0.868
##     actb              0.721    0.032   22.309    0.000    0.991    0.685
##     actc              0.955    0.031   31.056    0.000    1.313    0.822
##     actd              0.948    0.031   30.766    0.000    1.303    0.822
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   att_rad ~~
##     att_act           0.014    0.030    0.480    0.631    0.024    0.024
##     int_rad           0.629    0.062   10.158    0.000    0.612    0.612
##     int_act           0.074    0.065    1.139    0.255    0.051    0.051
##   att_act ~~
##     int_rad          -0.018    0.026   -0.708    0.479   -0.033   -0.033
##     int_act           0.377    0.043    8.690    0.000    0.491    0.491
##   int_rad ~~
```

```
##     int_act           0.410    0.062    6.651    0.000    0.306    0.306
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .vatta             1.261    0.083   15.105    0.000    1.261    0.532
##    .vattb             1.130    0.091   12.398    0.000    1.130    0.431
##    .vattc             0.946    0.085   11.074    0.000    0.946    0.384
##    .vattd             1.100    0.092   12.015    0.000    1.100    0.419
##    .vatte             0.856    0.068   12.515    0.000    0.856    0.733
##    .vattf             1.841    0.097   19.022    0.000    1.841    0.934
##    .vattg             0.639    0.082    7.766    0.000    0.639    0.432
##    .vatth             0.340    0.053    6.377    0.000    0.340    0.297
##    .rada              0.161    0.030    5.377    0.000    0.161    0.145
##    .radb              0.408    0.053    7.704    0.000    0.408    0.392
##    .radc              0.242    0.037    6.567    0.000    0.242    0.229
##    .radd              0.290    0.050    5.775    0.000    0.290    0.254
##    .acta              0.619    0.069    9.008    0.000    0.619    0.247
##    .actb              1.110    0.065   17.142    0.000    1.110    0.530
##    .actc              0.825    0.075   11.050    0.000    0.825    0.324
##    .actd              0.814    0.073   11.209    0.000    0.814    0.324
##     att_rad           1.109    0.103   10.744    0.000    1.000    1.000
##     att_act           0.312    0.048    6.435    0.000    1.000    1.000
##     int_rad           0.951    0.088   10.763    0.000    1.000    1.000
##     int_act           1.890    0.105   17.969    0.000    1.000    1.000
```

Hide

```
fitmeasures(modmod, fit.measures = c("cfi", "rmsea.robust", "rmsea.ci.lower.robust", "r
msea.ci.upper.robust", "srmr"))
```

```
##                   cfi          rmsea.robust rmsea.ci.lower.robust
##                 0.960                 0.055                 0.048
## rmsea.ci.upper.robust                  srmr
##                 0.062                 0.058
```

Hide

```
reliability(modmod)
```

```
##          att_rad   att_act   int_rad   int_act
## alpha  0.8346999 0.6532366 0.9197430 0.8765525
## omega  0.8352220 0.6702218 0.9215371 0.8804980
## omega2 0.8352220 0.6702218 0.9215371 0.8804980
## omega3 0.8341518 0.6612883 0.9219108 0.8799844
## avevar 0.5599516 0.3620865 0.7469477 0.6513799
```

Hide

```
modmodc <- sem(mod, estimator = "MLR", data = bx, group = "spol")
modmodw <- sem(mod, estimator = "MLR", data = bx, group = "spol", group.equal = "loadin
gs")
modmods <- sem(mod, estimator = "MLR", data = bx, group = "spol", group.equal = c("load
ings", "intercepts"))

summary(compareFit(modmodc, modmodw, modmods, argsLRT = list(method = "satorra.bentler.
2010")))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2010")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 196 37886 38395 495.38
## modmodw 208 37891 38344 524.88     20.942      12    0.05123 .
## modmods 220 37939 38335 596.95     70.838      12  2.232e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ######################## Model Fit Indices ###########################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     407.479†       196          .000        .056       .962†      .953
## modmodw     429.042        208          .000        .056†      .960       .954†
## modmods     493.026        220          .000        .060       .951       .946
##          srmr        aic        bic
## modmodc .059† 37885.678† 38394.678
## modmodw .062  37891.170  38343.614
## modmods .064  37939.241  38335.129†
##
## ################### Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc        12        0.000     -0.002      0.000 0.003  5.492
## modmods - modmodw        12        0.004     -0.009     -0.007 0.002 48.070
##                         bic
## modmodw - modmodc -51.064
## modmods - modmodw  -8.485
```

Hide

```
modindices(modmods, free.remove = F) %>% filter(op == "~1") %>% group_by(lhs, op, rhs)
%>% summarize(mi = mean(mi)) %>% arrange(desc(mi)) %>% as.data.frame() %>% head(20)
```

```
## `summarise()` has grouped output by 'lhs', 'op'. You can override using the
## `.groups` argument.
```

```
##        lhs op rhs           mi
## 1     actc ~1     1.142817e+01
## 2     actd ~1     9.165116e+00
## 3    vattc ~1     7.363550e+00
## 4     radc ~1     6.515112e+00
## 5    vattd ~1     2.125053e+00
## 6     radb ~1     2.087369e+00
## 7     radd ~1     1.058888e+00
## 8    vattb ~1     8.135633e-01
## 9    vatte ~1     7.933773e-01
## 10   vattg ~1     4.688300e-01
## 11   vatta ~1     4.385542e-01
## 12    rada ~1     2.572306e-01
## 13    actb ~1     2.504916e-01
## 14   vattf ~1     5.935979e-02
## 15   vatth ~1     3.630676e-02
## 16    acta ~1     6.849138e-03
## 17 int_rad ~1     1.259876e-09
## 18 int_act ~1     6.660874e-10
## 19 att_rad ~1     4.123153e-10
## 20 att_act ~1     4.630072e-11
```

Hide

```
modmods <- sem(mod, estimator = "MLR", data = bx, group = "spol", group.equal = c("load
ings", "intercepts"), group.partial = c("actc ~ 1"))

summary(compareFit(modmodc, modmodw, modmods, argsLRT = list(method = "satorra.bentler.
2010")))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2010")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 196 37886 38395 495.38
## modmodw 208 37891 38344 524.88     20.942      12    0.05123 .
## modmods 219 37916 38316 571.60     47.059      11  2.099e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ######################## Model Fit Indices ############################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     407.479†       196          .000        .056       .962†      .953
## modmodw     429.042        208          .000        .056†      .960       .954†
## modmods     471.973        219          .000        .058       .954       .950
##          srmr        aic        bic
## modmodc .059† 37885.678† 38394.678
## modmodw .062  37891.170  38343.614
## modmods .063  37915.890  38316.491†
##
## ################### Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc        12        0.000     -0.002      0.000 0.003  5.492
## modmods - modmodw        11        0.002     -0.005     -0.004 0.001 24.720
##                        bic
## modmodw - modmodc -51.064
## modmods - modmodw -27.123
```

Hide

```
modindices(modmods, free.remove = F) %>% filter(op == "~1") %>% group_by(lhs, op, rhs)
%>% summarize(mi = mean(mi)) %>% arrange(desc(mi)) %>% as.data.frame() %>% head(20)
```

```
## `summarise()` has grouped output by 'lhs', 'op'. You can override using the
## `.groups` argument.
```

```
##        lhs op rhs           mi
## 1    vattc ~1     7.363126e+00
## 2     radc ~1     6.514963e+00
## 3     actd ~1     3.539815e+00
## 4     acta ~1     2.308587e+00
## 5    vattd ~1     2.126402e+00
## 6     radb ~1     2.087145e+00
## 7     radd ~1     1.058571e+00
## 8    vattb ~1     8.129133e-01
## 9    vatte ~1     7.936845e-01
## 10   vattg ~1     4.688534e-01
## 11   vatta ~1     4.384409e-01
## 12    rada ~1     2.571577e-01
## 13   vattf ~1     5.923725e-02
## 14    actb ~1     5.856832e-02
## 15   vatth ~1     3.626864e-02
## 16 att_rad ~1     1.889807e-09
## 17 int_rad ~1     3.915534e-10
## 18 att_act ~1     1.859725e-10
## 19    actc ~1     1.022740e-12
## 20 int_act ~1     8.019760e-13
```

Hide

```
modmods <- sem(mod, estimator = "MLR", data = bx, group = "spol", group.equal = c("load
ings", "intercepts"), group.partial = c("actc ~ 1", "vattc ~ 1"))

summary(compareFit(modmodc, modmodw, modmods, argsLRT = list(method = "satorra.bentler.
2010")))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2010")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 196 37886 38395 495.38
## modmodw 208 37891 38344 524.88     20.942      12   0.051229 .
## modmods 218 37900 38305 553.32     29.227      10   0.001145 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ######################## Model Fit Indices ############################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     407.479†       196          .000        .056       .962†      .953
## modmodw     429.042        208          .000        .056†      .960       .954†
## modmods     456.446        218          .000        .057       .957       .953
##          srmr        aic        bic
## modmodc .059† 37885.678† 38394.678
## modmodw .062  37891.170  38343.614
## modmods .062  37899.616  38304.930†
##
## ################### Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr   aic
## modmodw - modmodc        12        0.000     -0.002      0.000 0.003 5.492
## modmods - modmodw        10        0.001     -0.003     -0.001 0.001 8.445
##                        bic
## modmodw - modmodc -51.064
## modmods - modmodw -38.684
```

Hide

```
modindices(modmods, free.remove = F) %>% filter(op == "~1") %>% group_by(lhs, op, rhs)
%>% summarize(mi = mean(mi)) %>% arrange(desc(mi)) %>% as.data.frame() %>% head(20)
```

```
## `summarise()` has grouped output by 'lhs', 'op'. You can override using the
## `.groups` argument.
```

```
##        lhs op rhs           mi
## 1     radc ~1     6.513792e+00
## 2     actd ~1     3.539998e+00
## 3     acta ~1     2.308366e+00
## 4     radb ~1     2.087157e+00
## 5     radd ~1     1.058342e+00
## 6    vatte ~1     7.938094e-01
## 7    vattg ~1     4.685688e-01
## 8     rada ~1     2.568713e-01
## 9    vattd ~1     1.049999e-01
## 10   vattf ~1     5.922109e-02
## 11    actb ~1     5.856336e-02
## 12   vatta ~1     3.695620e-02
## 13   vatth ~1     3.618288e-02
## 14   vattb ~1     2.984472e-02
## 15 int_rad ~1     1.287505e-09
## 16 int_act ~1     4.013438e-11
## 17    actc ~1     2.225867e-11
## 18 att_act ~1     1.944972e-11
## 19 att_rad ~1     1.650123e-11
## 20   vattc ~1     1.517554e-12
```

Hide

```
modmods <- sem(mod, estimator = "MLR", data = bx, group = "spol", group.equal = c("load
ings", "intercepts"), group.partial = c("actc ~ 1", "vattc ~ 1", "radc ~ 1"))

summary(compareFit(modmodc, modmodw, modmods, argsLRT = list(method = "satorra.bentler.
2010")))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2010")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 196 37886 38395 495.38
## modmodw 208 37891 38344 524.88     20.942      12    0.05123 .
## modmods 217 37887 38297 539.17     14.175       9    0.11623
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ######################## Model Fit Indices ############################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     407.479†       196          .000        .056       .962†      .953
## modmodw     429.042        208          .000        .056       .960       .954
## modmods     443.951        217          .000        .056†      .959       .955†
##          srmr        aic        bic
## modmodc .059† 37885.678† 38394.678
## modmodw .062  37891.170  38343.614
## modmods .062  37887.460  38297.487†
##
## ################### Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc        12        0.000     -0.002      0.000 0.003  5.492
## modmods - modmodw         9       -0.001     -0.001      0.001 0.000 -3.710
##                         bic
## modmodw - modmodc -51.064
## modmods - modmodw -46.127
```

Hide

```
dod <- lavPredict(modmods, newdata = bx, assemble = T) %>% as.data.frame()
describe(dod, type = 2)
```

```
##         vars   n  mean   sd median trimmed  mad   min  max range  skew kurtosis
## att_rad    1 823  0.23 0.98   0.11    0.15 0.97 -1.32 3.09  4.41  0.66    -0.11
## att_act    2 823 -0.10 0.50  -0.01   -0.05 0.54 -1.99 0.58  2.57 -0.85     0.39
## int_rad    3 823  0.15 0.95  -0.36   -0.06 0.16 -0.53 4.57  5.11  1.97     3.65
## int_act    4 823 -0.17 1.30  -0.27   -0.24 1.56 -2.31 3.11  5.43  0.39    -0.71
## spol       5 823  1.55 0.50   2.00    1.56 0.00  1.00 2.00  1.00 -0.20    -1.97
##           se
## att_rad 0.03
## att_act 0.02
## int_rad 0.03
## int_act 0.05
## spol    0.02
```

Hide

```
describe(poms(dod), type = 2)
```

```
##         vars   n mean   sd median trimmed  mad min max range  skew kurtosis
## att_rad    1 823 0.35 0.22   0.33    0.33 0.22   0   1     1  0.66    -0.11
## att_act    2 823 0.74 0.19   0.77    0.76 0.21   0   1     1 -0.85     0.39
## int_rad    3 823 0.13 0.19   0.03    0.09 0.03   0   1     1  1.97     3.65
## int_act    4 823 0.40 0.24   0.38    0.38 0.29   0   1     1  0.39    -0.71
## spol       5 823 0.55 0.50   1.00    0.56 0.00   0   1     1 -0.20    -1.97
##           se
## att_rad 0.01
## att_act 0.01
## int_rad 0.01
## int_act 0.01
## spol    0.02
```

Hide

```
names(dod)[5] <- "spo"
bmf <- cbind(bx, dod)
cor(bmf$spol, bmf$spo)
```

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

LPAs

Hide

```
bf <- subset(bmf, spol == 2)
bm <- subset(bmf, spol == 1)

f1 <- mx_profiles(bf[, c("att_rad", "int_rad", "att_act", "int_act")], 1:10, variances
= "equal",covariances = "zero")
```

```
## Running mix1 with 8 parameters
```

```
## Running mix2 with 13 parameters
## Running mix2 with 13 parameters
```

```
## Running mix3 with 18 parameters
## Running mix3 with 18 parameters
```

```
## Running mix4 with 23 parameters
## Running mix4 with 23 parameters
```

```
## Running mix5 with 28 parameters
## Running mix5 with 28 parameters
```

```
## Running mix6 with 33 parameters
## Running mix6 with 33 parameters
```

```
## Running mix7 with 38 parameters
## Running mix7 with 38 parameters
```

```
## Running mix8 with 43 parameters
## Running mix8 with 43 parameters
```

```
## Running mix9 with 48 parameters
## Running mix9 with 48 parameters
```

```
## Running mix10 with 53 parameters
## Running mix10 with 53 parameters
```

Hide

```
table_fit(f1)
```

```
##            Name Classes        LL   n Parameters      AIC      BIC    saBIC
## 1   equal var 1       1 -2229.759 452          8 4475.518 4508.428 4483.039
## 2   equal var 2       2 -2120.783 452         13 4267.567 4321.044 4279.787
## 3   equal var 3       3 -1898.721 452         18 3833.442 3907.488 3850.363
## 4   equal var 4       4 -1868.765 452         23 3783.531 3878.146 3805.152
## 5   equal var 5       5 -1738.301 452         28 3532.602 3647.785 3558.923
## 6   equal var 6       6 -1725.633 452         33 3517.267 3653.018 3548.288
## 7   equal var 7       7 -1627.903 452         38 3331.805 3488.125 3367.527
## 8   equal var 8       8 -1606.814 452         43 3299.629 3476.517 3340.051
## 9   equal var 9       9 -1559.362 452         48 3214.724 3412.181 3259.846
## 10 equal var 10      10 -1528.857 452         53 3163.715 3381.740 3213.537
##      Entropy  prob_min  prob_max      n_min     n_max  np_ratio  np_local
## 1  1.0000000 1.0000000 1.0000000 1.00000000 1.0000000 56.500000 56.500000
## 2  0.7413275 0.9106624 0.9402438 0.44469027 0.5553097 34.769231 33.500000
## 3  0.8031003 0.8611288 0.9728079 0.11061947 0.5176991 25.111111  9.375000
## 4  0.8426442 0.8491842 0.9743149 0.02212389 0.5088496 19.652174  2.000000
## 5  0.8621692 0.8534951 0.9964741 0.02212389 0.4690265 16.142857  2.083333
## 6  0.7879993 0.7880744 0.9964423 0.01769912 0.3185841 13.696970  1.714286
## 7  0.8871957 0.8429193 0.9988312 0.02433628 0.4535398 11.894737  2.406250
## 8  0.8420602 0.8094095 0.9986970 0.01769912 0.3008850 10.511628  1.777778
## 9  0.8232495 0.7855478 0.9978467 0.01548673 0.3097345  9.416667  1.575000
## 10 0.8555427 0.8178785 0.9994596 0.01769912 0.3008850  8.528302  1.818182
```

Hide

```
lr_lmr(f1)
```

```
## Lo-Mendell-Rubin adjusted Likelihood Ratio Test:
##
##  null   alt   lr df        p    w2     p_w2
##  mix1  mix2 8.12  5 2.30e-16 0.398 7.94e-09
##  mix2  mix3 8.91  5 2.22e-16 1.373 3.09e-03
##  mix3  mix4 2.36  5 9.24e-03 0.358 1.76e-07
##  mix4  mix5 6.03  5 8.05e-10 1.035 2.65e-02
##  mix5  mix6 1.74  5 4.06e-02 0.117 6.06e-02
##  mix6  mix7 4.03  5 2.77e-05 1.300 3.11e-02
##  mix7  mix8 1.99  5 2.31e-02 0.247 8.08e-02
##  mix8  mix9 1.91  5 2.79e-02 1.361 4.58e-01
##  mix9 mix10 1.47  5 7.06e-02 0.951 6.32e-01
```

Hide

```
m1 <- mx_profiles(bm[, c("att_rad", "int_rad", "att_act", "int_act")], 1:10, variances
= "equal",covariances = "zero")
```

```
## Running mix1 with 8 parameters
```

```
## Running mix2 with 13 parameters
## Running mix2 with 13 parameters
```

```
## Running mix3 with 18 parameters
## Running mix3 with 18 parameters
```

```
## Running mix4 with 23 parameters
## Running mix4 with 23 parameters
```

```
## Running mix5 with 28 parameters
## Running mix5 with 28 parameters
```

```
## Running mix6 with 33 parameters
## Running mix6 with 33 parameters
```

```
## Running mix7 with 38 parameters
## Running mix7 with 38 parameters
```

```
## Running mix8 with 43 parameters
## Running mix8 with 43 parameters
```

```
## Running mix9 with 48 parameters
## Running mix9 with 48 parameters
```

```
## Running mix10 with 53 parameters
## Running mix10 with 53 parameters
```

Hide

```
table_fit(m1)
```

```
##            Name Classes        LL   n Parameters      AIC      BIC    saBIC
## 1   equal var 1       1 -1943.236 371          8 3902.473 3933.803 3908.421
## 2   equal var 2       2 -1799.088 371         13 3624.175 3675.086 3633.841
## 3   equal var 3       3 -1752.025 371         18 3540.050 3610.541 3553.433
## 4   equal var 4       4 -1676.971 371         23 3399.943 3490.015 3417.044
## 5   equal var 5       5 -1630.327 371         28 3316.653 3426.307 3337.472
## 6   equal var 6       6 -1621.192 371         33 3308.384 3437.618 3332.920
## 7   equal var 7       7 -1561.321 371         38 3198.642 3347.458 3226.896
## 8   equal var 8       8 -1544.692 371         43 3175.384 3343.781 3207.356
## 9   equal var 9       9 -1591.271 371         48 3278.542 3466.520 3314.231
## 10 equal var 10      10 -1520.433 371         53 3146.865 3354.424 3186.272
##      Entropy  prob_min  prob_max      n_min     n_max  np_ratio  np_local
## 1  1.0000000 1.0000000 1.0000000 1.00000000 1.0000000 46.375000 46.375000
## 2  0.9046159 0.9392674 0.9835371 0.22911051 0.7708895 28.538462 14.166667
## 3  0.8217907 0.7952136 0.9508291 0.17520216 0.5983827 20.611111 12.187500
## 4  0.8701590 0.7962334 0.9687313 0.07277628 0.5633423 16.130435  5.400000
## 5  0.8835177 0.8091750 0.9952197 0.03234501 0.5040431 13.250000  2.500000
## 6  0.7901923 0.6523238 0.9680238 0.02964960 0.3908356 11.242424  2.357143
## 7  0.8259610 0.6668367 0.9998101 0.02425876 0.3719677  9.763158  1.968750
## 8  0.8364174 0.6553220 0.9998432 0.02425876 0.3692722  8.627907  2.000000
## 9  0.7921231 0.6600376 0.9425763 0.02425876 0.2479784  7.729167  2.025000
## 10 0.8020894 0.6717211 0.9999567 0.02425876 0.1886792  7.000000  2.045455
```

Hide

```
lr_lmr(m1)
```

```
## Lo-Mendell-Rubin adjusted Likelihood Ratio Test:
##
##  null   alt    lr df        p     w2     p_w2
##  mix1  mix2  9.18  5 2.22e-16 0.6642 2.06e-04
##  mix2  mix3  4.47  5 3.82e-06 0.2981 3.62e-05
##  mix3  mix4  5.05  5 2.25e-07 0.5963 2.29e-02
##  mix4  mix5  3.02  5 1.25e-03 0.6412 1.18e-01
##  mix5  mix6  1.91  5 2.80e-02 0.0616 7.34e-01
##  mix6  mix7  4.53  5 2.94e-06 0.4708 4.86e-01
##  mix7  mix8  2.09  5 1.82e-02 0.1702 1.48e-01
##  mix8  mix9 -2.42  5 9.92e-01 0.9989 4.01e-01
##  mix9 mix10  3.55  5 1.92e-04 1.0729 5.41e-01
```

Visualizations

Hide

```
bf$klase3 <- class_prob(f1$`equal var 3`)$individual %>% as.data.frame(.) %>% select(pr
edicted) %>% unlist()
bf$klase4 <- class_prob(f1$`equal var 4`)$individual %>% as.data.frame(.) %>% select(pr
edicted) %>% unlist()
bf$klase5 <- class_prob(f1$`equal var 5`)$individual %>% as.data.frame(.) %>% select(pr
edicted) %>% unlist()
bf$klase6 <- class_prob(f1$`equal var 6`)$individual %>% as.data.frame(.) %>% select(pr
edicted) %>% unlist()
bf$klase7 <- class_prob(f1$`equal var 7`)$individual %>% as.data.frame(.) %>% select(pr
edicted) %>% unlist()

bm$klase3 <- class_prob(m1$`equal var 3`)$individual %>% as.data.frame(.) %>% select(pr
edicted) %>% unlist()
bm$klase4 <- class_prob(m1$`equal var 4`)$individual %>% as.data.frame(.) %>% select(pr
edicted) %>% unlist()
bm$klase5 <- class_prob(m1$`equal var 5`)$individual %>% as.data.frame(.) %>% select(pr
edicted) %>% unlist()
bm$klase6 <- class_prob(m1$`equal var 6`)$individual %>% as.data.frame(.) %>% select(pr
edicted) %>% unlist()
bm$klase7 <- class_prob(m1$`equal var 7`)$individual %>% as.data.frame(.) %>% select(pr
edicted) %>% unlist()

round(prop.table(table(bf$klase3)), 2)
```

```
##
##    1    2    3
## 0.52 0.37 0.11
```

Hide

```
round(prop.table(table(bf$klase4)), 2)
```

```
##
##    1    2    3    4
## 0.02 0.51 0.36 0.11
```

Hide

```
round(prop.table(table(bf$klase5)), 3)
```

```
##
##     1     2     3     4     5
## 0.022 0.469 0.310 0.144 0.055
```

Hide

```
round(prop.table(table(bf$klase6)), 2)
```

```
##
##    1    2    3    4    5    6
## 0.32 0.25 0.02 0.22 0.14 0.06
```

Hide

```
round(prop.table(table(bm$klase3)), 2)
```

```
##
##    1    2    3
## 0.60 0.18 0.23
```

Hide

```
round(prop.table(table(bm$klase4)), 3)
```

```
##
##     1     2     3     4
## 0.563 0.135 0.073 0.229
```

Hide

```
round(prop.table(table(bm$klase5)), 2)
```

```
##
##    1    2    3    4    5
## 0.50 0.13 0.18 0.15 0.03
```

Hide

```
round(prop.table(table(bm$klase6)), 2)
```

```
##
##    1    2    3    4    5    6
## 0.39 0.14 0.18 0.15 0.11 0.03
```

Hide

```
#f

pf3 <- bf %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase3 = factor(klase3)) %>%
  group_by(klase3, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase3)) +
  geom_histogram(stat = "identity", position = "dodge") +
  theme_bw()  +
  theme(text = element_text(size = 12, family = "serif")) +
  scale_fill_manual("class", values = c("red", "steelblue", "purple")) +
  theme(axis.text.x = element_blank()) +
  xlab("")
```

```
## `summarise()` has grouped output by 'klase3'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge"): Ignoring
## unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pf4 <- bf %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase4 = factor(klase4)) %>%
  group_by(klase4, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase4)) +
  geom_histogram(stat = "identity", position = "dodge") +
  theme_bw()   +
  theme(text = element_text(size = 12, family = "serif")) +
  scale_fill_manual("class", values = c("green", "red", "steelblue", "purple")) +
  theme(axis.text.x = element_blank()) +
  xlab("")
```

```
## `summarise()` has grouped output by 'klase4'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge"): Ignoring
## unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pf5 <- bf %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase5 = factor(klase5)) %>%
  group_by(klase5, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase5)) +
  geom_histogram(stat = "identity", position = "dodge") +
  theme_bw()   +
  theme(text = element_text(size = 12, family = "serif")) +
  scale_fill_manual("class", values = c("green", "red", "steelblue", "goldenrod1", "pur
ple")) +
  theme(axis.text.x = element_text(angle = 90, size = 12, hjust = .5, vjust = .5))
```

```
## `summarise()` has grouped output by 'klase5'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge"): Ignoring
## unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pf6 <- bf %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase6 = factor(klase6)) %>%
  group_by(klase6, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase6)) +
  geom_histogram(stat = "identity", position = "dodge") +
  theme_bw() +
  theme(text = element_text(size = 12, family = "serif")) +
  xlab("") +
  scale_fill_manual("class", values = c("green", "red", "steelblue", "black", "goldenro
d1", "purple")) +
  theme(axis.text.x = element_text(angle = 90, size = 12, hjust = .5, vjust = .5))
```

```
## `summarise()` has grouped output by 'klase6'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge"): Ignoring
## unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pf3b <- bf %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase3 = factor(klase3)) %>%
  group_by(klase3, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase3, linetype = klase3)) +
  geom_histogram(stat = "identity", position = "dodge", linewidth = 1, col = "black") +
  theme_bw()  +
  theme(text = element_text(size = 12, family = "serif")) +
  scale_fill_manual("profile", values = c("gray40", "gray70", "gray100")) +
  scale_linetype_manual("profile", values = c(1, 1, 1)) +
  theme(axis.text.x = element_blank()) +
  xlab("")
```

```
## `summarise()` has grouped output by 'klase3'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge", linewidth = 1,
## : Ignoring unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pf4b <- bf %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase4 = factor(klase4)) %>%
  group_by(klase4, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase4, linetype = klase4)) +
  geom_histogram(stat = "identity", position = "dodge", linewidth = 1, col = "black") +
  theme_bw()   +
  theme(text = element_text(size = 12, family = "serif")) +
  scale_fill_manual("profile", values = c("gray40", "gray40", "gray70", "gray100")) +
  scale_linetype_manual("profile", values = c(2, 1, 1, 1)) +
  theme(axis.text.x = element_blank()) +
  xlab("")
```

```
## `summarise()` has grouped output by 'klase4'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge", linewidth = 1,
## : Ignoring unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pf5b <- bf %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase5 = factor(klase5)) %>%
  group_by(klase5, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase5, linetype = klase5)) +
  geom_histogram(stat = "identity", position = "dodge", linewidth = 1, col = "black") +
  theme_bw()   +
  theme(text = element_text(size = 12, family = "serif")) +
  scale_fill_manual("profile", values = c("gray40", "gray40", "gray70", "gray70", "gray
100")) +
  scale_linetype_manual("profile", values = c(2, 1, 1, 2, 1)) +
  theme(axis.text.x = element_text(angle = 90, size = 12, hjust = .5, vjust = .5)) +
  xlab("")
```

```
## `summarise()` has grouped output by 'klase5'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge", linewidth = 1,
## : Ignoring unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pf6b <- bf %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase6 = factor(klase6)) %>%
  group_by(klase6, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase6, linetype = klase6)) +
  geom_histogram(stat = "identity", position = "dodge", linewidth = 1, col = "black") +
  theme_bw() +
  theme(text = element_text(size = 12, family = "serif")) +
  scale_fill_manual("profile", values = c("gray100", "gray70", "gray40", "gray40", "gra
y70", "gray100")) +
  scale_linetype_manual("profile", values = c(2, 1, 2, 1, 2, 1)) +
  theme(axis.text.x = element_text(angle = 90, size = 12, hjust = .5, vjust = .5)) +
  xlab("")
```

```
## `summarise()` has grouped output by 'klase6'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge", linewidth = 1,
## : Ignoring unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pm3 <- bm %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase3 = factor(klase3)) %>%
  group_by(klase3, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase3)) +
  geom_histogram(stat = "identity", position = "dodge") +
  theme_bw()  +
  theme(text = element_text(size = 12, family = "serif")) +
  scale_fill_manual("class", values = c("red", "steelblue", "purple")) +
  theme(axis.text.x = element_blank()) +
  xlab("")
```

```
## `summarise()` has grouped output by 'klase3'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge"): Ignoring
## unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pm4 <- bm %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase4 = factor(klase4)) %>%
  group_by(klase4, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase4)) +
  geom_histogram(stat = "identity", position = "dodge") +
  theme_bw()   +
  theme(text = element_text(size = 12, family = "serif")) +
  scale_fill_manual("class", values = c("red", "steelblue", "purple", "hotpink")) +
  theme(axis.text.x = element_blank()) +
  xlab("")
```

```
## `summarise()` has grouped output by 'klase4'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge"): Ignoring
## unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pm5 <- bm %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase5 = factor(klase5)) %>%
  group_by(klase5, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase5)) +
  geom_histogram(stat = "identity", position = "dodge") +
  theme_bw()   +
  theme(text = element_text(size = 12, family = "serif")) +
  scale_fill_manual("class", values = c("hotpink", "red", "purple", "goldenrod1", "stee
lblue")) +
  theme(axis.text.x = element_text(angle = 90, size = 12, hjust = .5, vjust = .5))
```

```
## `summarise()` has grouped output by 'klase5'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge"): Ignoring
## unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pm6 <- bm %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase6 = factor(klase6)) %>%
  group_by(klase6, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase6)) +
  geom_histogram(stat = "identity", position = "dodge") +
  theme_bw() +
  theme(text = element_text(size = 12, family = "serif")) +
  xlab("") +
  scale_fill_manual("class", values = c("red", "hotpink", "purple", "brown4", "steelblu
e", "goldenrod1")) +
  theme(axis.text.x = element_text(angle = 90, size = 12, hjust = .5, vjust = .5))
```

```
## `summarise()` has grouped output by 'klase6'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge"): Ignoring
## unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pm3b <- bm %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase3 = factor(klase3)) %>%
  group_by(klase3, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase3, linetype = klase3)) +
  geom_histogram(stat = "identity", position = "dodge", linewidth = 1, col = "black") +
  theme_bw()  +
  theme(text = element_text(size = 12, family = "serif")) +
  theme(axis.text.x = element_blank())  +
  xlab("") +
  scale_fill_manual("profile", values = c("gray40", "gray70", "gray100")) +
  scale_linetype_manual("profile", values = c(1, 1, 1))
```

```
## `summarise()` has grouped output by 'klase3'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge", linewidth = 1,
## : Ignoring unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pm4b <- bm %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase4 = factor(klase4)) %>%
  group_by(klase4, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase4, linetype = klase4)) +
  geom_histogram(stat = "identity", position = "dodge", linewidth = 1, col = "black") +
  theme_bw()   +
  theme(text = element_text(size = 12, family = "serif")) +
  theme(axis.text.x = element_blank()) +
  xlab("") +
  scale_fill_manual("profile", values = c("gray40", "gray70", "gray100", "gray100")) +
  scale_linetype_manual("profile", values = c(1, 1, 1, 6))
```

```
## `summarise()` has grouped output by 'klase4'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge", linewidth = 1,
## : Ignoring unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
pm5b <- bm %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase5 = factor(klase5)) %>%
  group_by(klase5, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase5, linetype = klase5)) +
  geom_histogram(stat = "identity", position = "dodge", linewidth = 1, col = "black") +
  theme_bw()   +
  theme(text = element_text(size = 12, family = "serif")) +
  theme(axis.text.x = element_text(angle = 90, size = 12, hjust = .5, vjust = .5)) +
  xlab("") +
  scale_fill_manual("profile", values = c("gray40", "gray70", "gray100", "gray70", "gra
y100")) +
  scale_linetype_manual("profile", values = c(1, 1, 6, 6, 1)) +
  theme(axis.text.x = element_text(angle = 90, size = 12, hjust = .5, vjust = .5))
```

```
## `summarise()` has grouped output by 'klase5'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge", linewidth = 1,
## : Ignoring unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

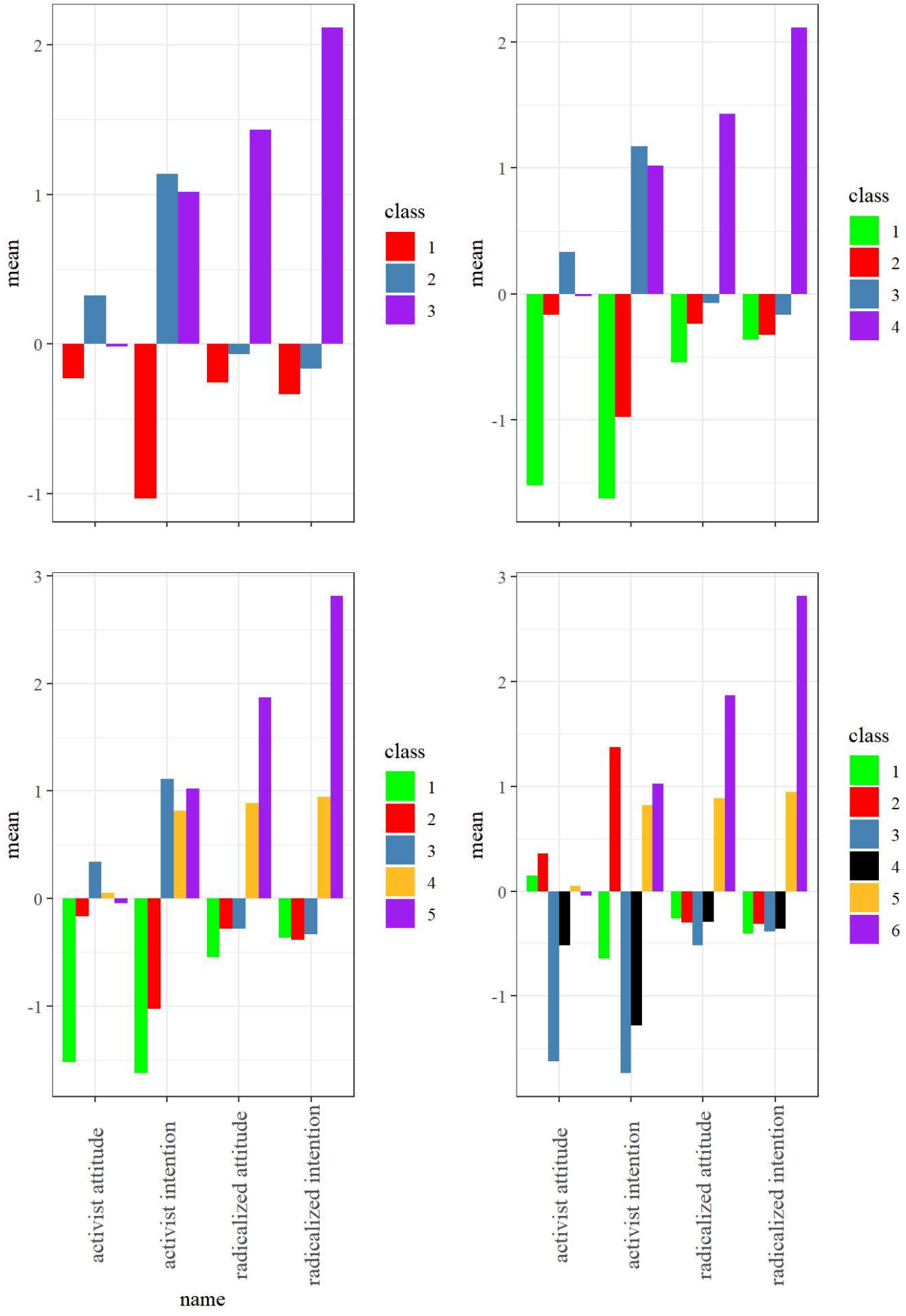
```
pm6b <- bm %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase6 = factor(klase6)) %>%
  group_by(klase6, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase6, linetype = klase6)) +
  geom_histogram(stat = "identity", position = "dodge", linewidth = 1, col = "black") +
  theme_bw() +
  theme(text = element_text(size = 12, family = "serif")) +
  theme(axis.text.x = element_text(angle = 90, size = 12, hjust = .5, vjust = .5)) +
  scale_fill_manual("profile", values = c("gray40", "gray70", "gray100", "gray70", "gra
y70", "gray100")) +
  scale_linetype_manual("profile", values = c(1, 3, 6, 6, 1, 1)) +
  theme(axis.text.x = element_text(angle = 90, size = 12, hjust = .5, vjust = .5)) +
  xlab("")
```

```
## `summarise()` has grouped output by 'klase6'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge", linewidth = 1,
## : Ignoring unknown parameters: `binwidth`, `bins`, and `pad`
```

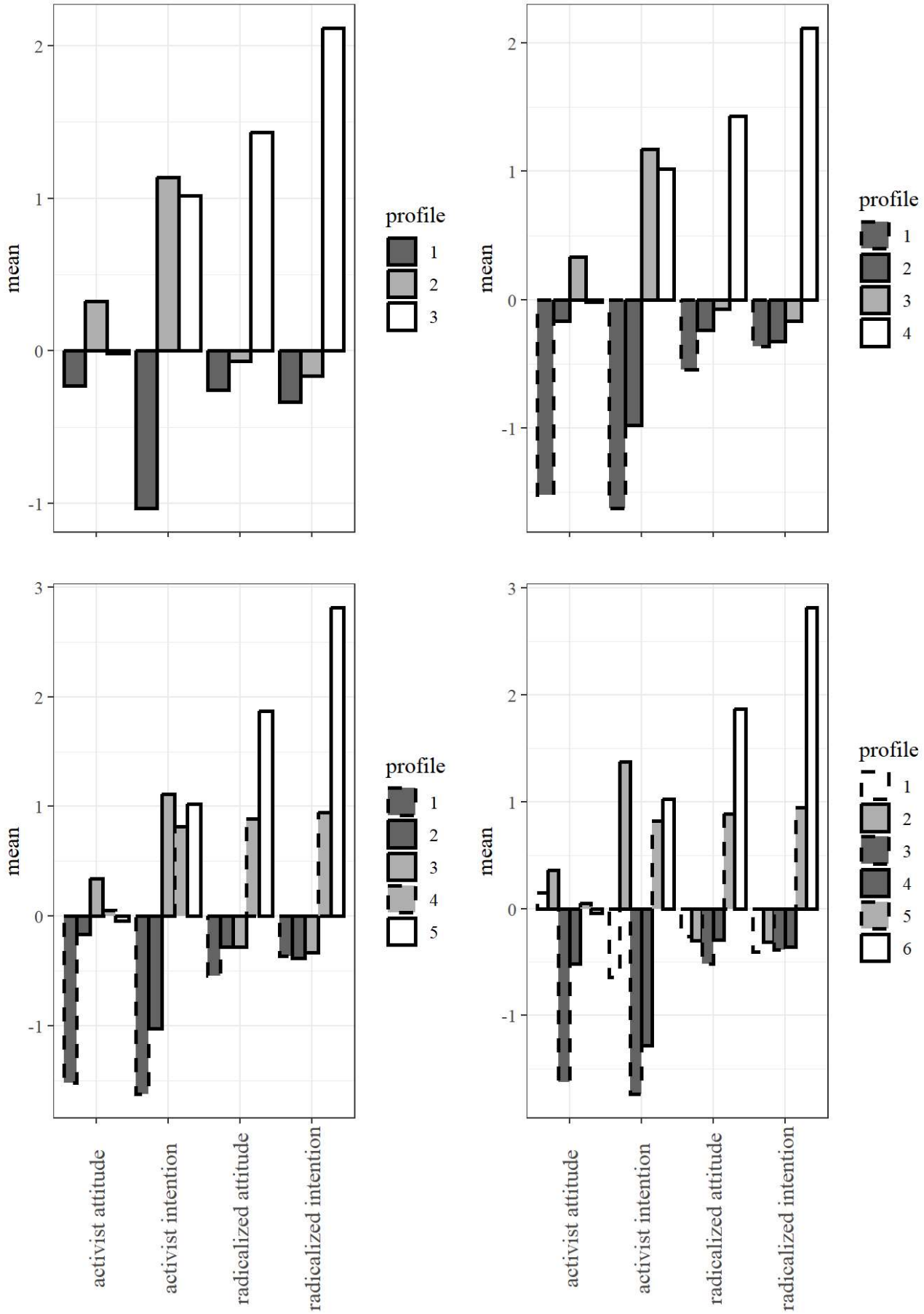
Hide

```
cowplot::plot_grid(pf3, pf4, pf5, pf6, nrow = 2, rel_heights = c(3, 4))
```



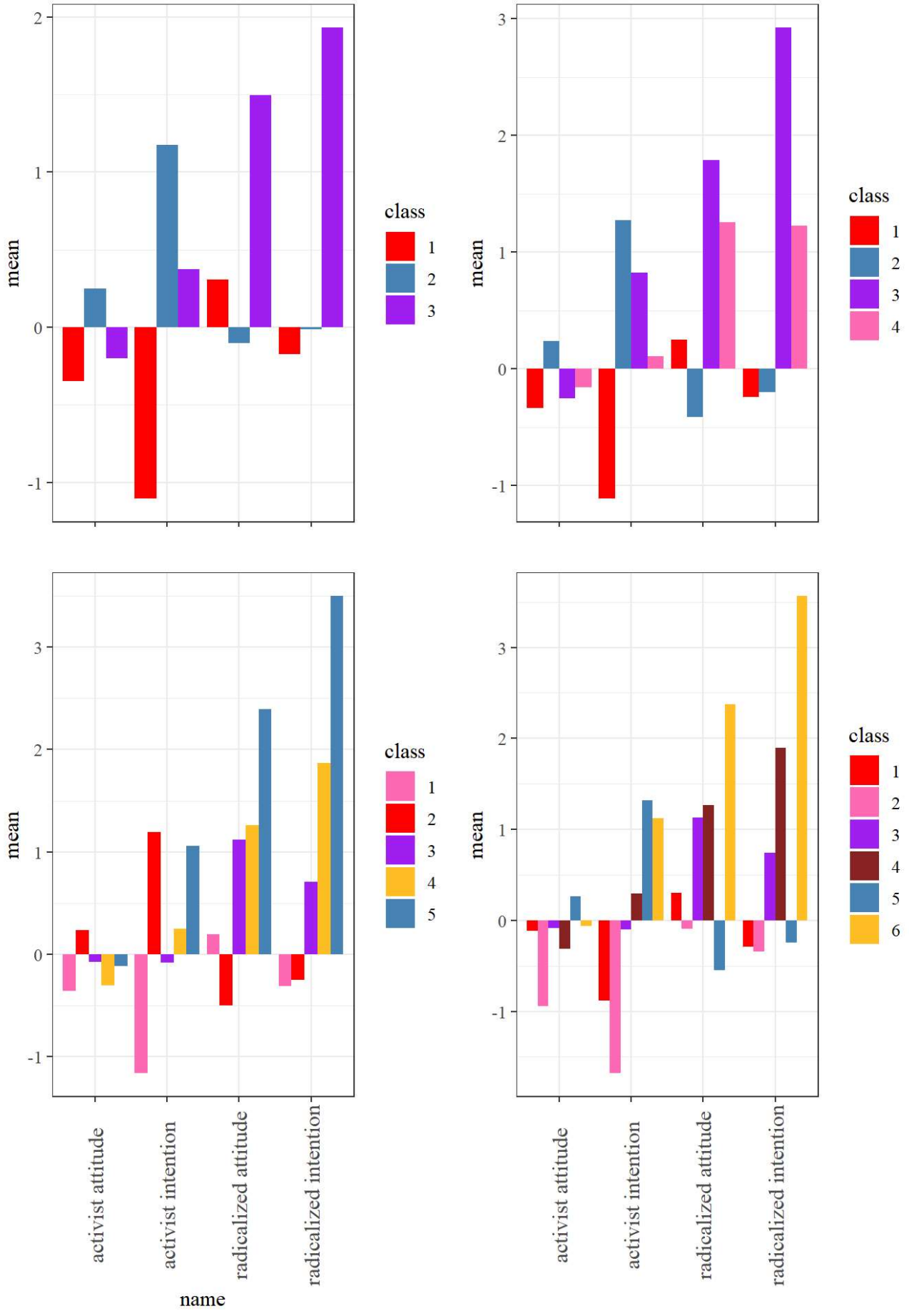
Hide

```
cowplot::plot_grid(pf3b, pf4b, pf5b, pf6b, nrow = 2, rel_heights = c(3, 4))
```



Hide

```
cowplot::plot_grid(pm3, pm4, pm5, pm6, nrow = 2, rel_heights = c(3, 4))
```



Hide

```
cowplot::plot_grid(pm3b, pm4b, pm5b, pm6b, nrow = 2, rel_heights = c(3, 4))
```

mean
profile
1
2
3
activist attitude
activist intention
radicalized attitude
radicalized intention

mean
profile
1
2
3
4
activist attitude
activist intention
radicalized attitude
radicalized intention

mean
profile
1
2
3
4
5
activist attitude
activist intention
radicalized attitude
radicalized intention

mean
profile
1
2
3
4
5
6
activist attitude
activist intention
radicalized attitude
radicalized intention

Hide

```
pm4b <- bm %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase4 = factor(klase4)) %>%
  group_by(klase4, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase4, linetype = klase4)) +
  geom_histogram(stat = "identity", position = "dodge", linewidth = 1, col = "black") +
  theme_bw()   +
  theme(text = element_text(size = 12, family = "serif"), axis.text.x = element_text(an
gle = 90, size = 12, hjust = .5, vjust = .5)) +
  xlab("") +
  ylim(c(-2, 3)) +
  scale_fill_manual("profile", values = c("gray40", "gray70", "gray100", "gray100")) +
  scale_linetype_manual("profile", values = c(1, 1, 1, 6))
```

```
## `summarise()` has grouped output by 'klase4'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge", linewidth = 1,
## : Ignoring unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

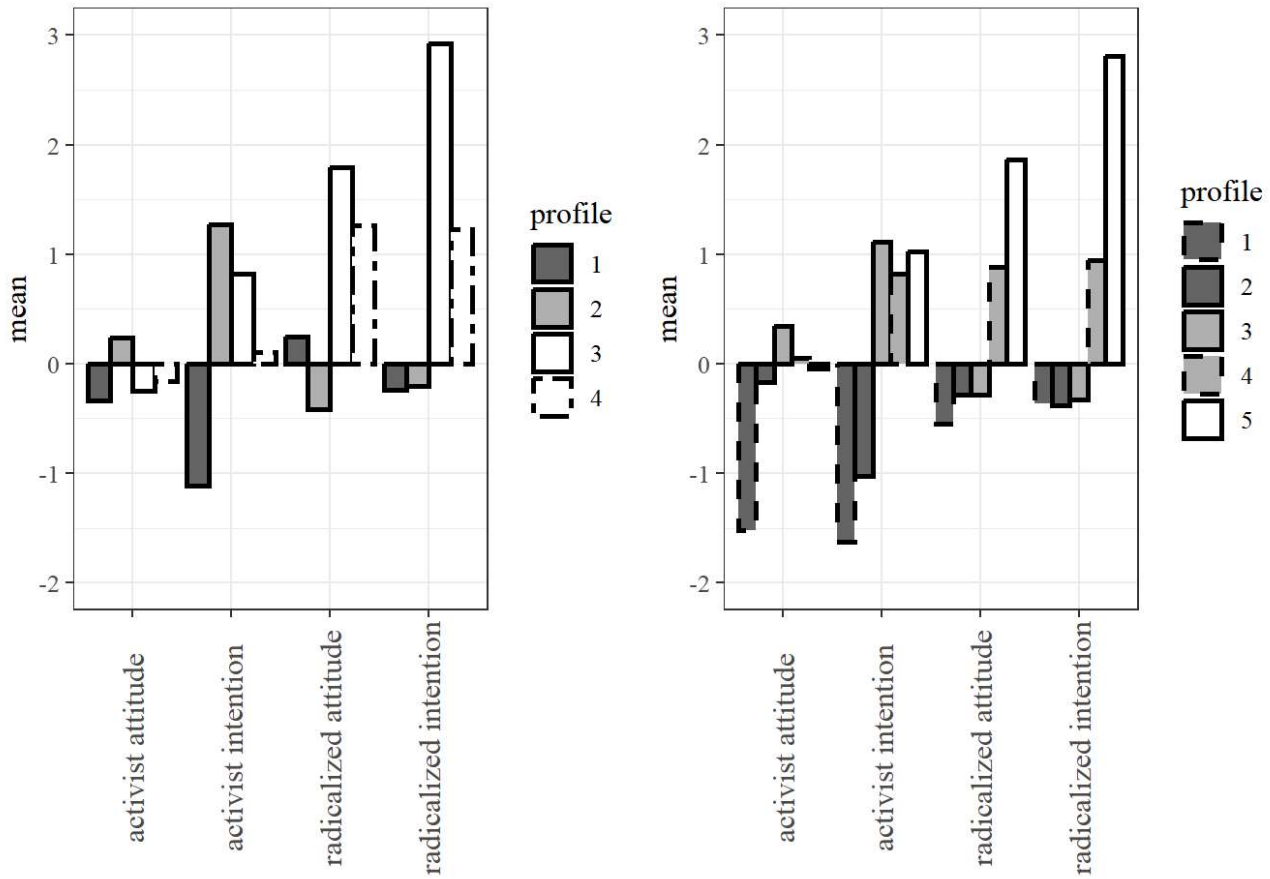
```
pf5b <- bf %>%
  tidyr::pivot_longer(., c("att_act", "int_act", "att_rad", "int_rad")) %>%
  mutate(name = factor(name, levels = c("att_act", "int_act", "att_rad", "int_rad"), la
bels = c("activist attitude", "activist intention", "radicalized attitude", "radicalize
d intention")),
         klase5 = factor(klase5)) %>%
  group_by(klase5, name) %>%
  summarize(mean = mean(value)) %>%
  ggplot(aes(x = name, y = mean, fill = klase5, linetype = klase5)) +
  geom_histogram(stat = "identity", position = "dodge", linewidth = 1, col = "black") +
  theme_bw() +
  theme(text = element_text(size = 12, family = "serif")) +
  scale_fill_manual("profile", values = c("gray40", "gray40", "gray70", "gray70", "gray
100")) +
  scale_linetype_manual("profile", values = c(2, 1, 1, 2, 1)) +
  theme(axis.text.x = element_text(angle = 90, size = 12, hjust = .5, vjust = .5)) +
  xlab("") +
  ylim(c(-2, 3))
```

```
## `summarise()` has grouped output by 'klase5'. You can override using the
## `.groups` argument.
```

```
## Warning in geom_histogram(stat = "identity", position = "dodge", linewidth = 1,
## : Ignoring unknown parameters: `binwidth`, `bins`, and `pad`
```

Hide

```
cowplot::plot_grid(pm4b, pf5b, nrow = 1)
```



Hide

```
table(bf$klase5)
```

```
##
##   1   2   3   4   5
##  10 212 140  65  25
```

Hide

```
table(bm$klase4)
```

```
##
##   1   2   3   4
## 209  50  27  85
```

Hide

```
round(prop.table(table(bf$klase5)), 2)
```

```
##
##    1    2    3    4    5
## 0.02 0.47 0.31 0.14 0.06
```

Hide

```
round(prop.table(table(bm$klase4)), 2)
```

```
##
##    1    2    3    4
## 0.56 0.13 0.07 0.23
```

Hide

```
#prototypes
lavPredict(modmods, newdata = data.frame(vatta = c(3, 2, 4, 3, 2, 4), vattb = c(3, 2,
4, 3, 2, 4), vattc = c(3, 2, 4, 3, 2, 4), vattd = c(3, 2, 4, 3, 2, 4),
                                         vatte = c(3, 2, 4, 3, 2, 4), vattf = c(3, 2,
4, 3, 2, 4), vattg = c(3, 2, 4, 3, 2, 4), vatth = c(3, 2, 4, 3, 2, 4),
                                         acta = c(3, 2, 4, 3, 2, 4), actb = c(3, 2, 4,
3, 2, 4), actc = c(3, 2, 4, 3, 2, 4), actd = c(3, 2, 4, 3, 2, 4),
                                         rada = c(3, 2, 4, 3, 2, 4), radb = c(3, 2, 4,
3, 2, 4), radc = c(3, 2, 4, 3, 2, 4), radd = c(3, 2, 4, 3, 2, 4), spol = c(1, 1, 1, 2,
2, 2)), assemble = T) %>% as.data.frame()
```

```
##      att_rad    att_act   int_rad     int_act spol
## 1  0.3580470 -1.1106708 1.5846781 -0.03445769    1
## 2 -0.4854702 -1.6404490 0.5888419 -1.07623754    1
## 3  1.2015642 -0.5808927 2.5805143  1.00732217    1
## 4  0.5299235 -1.0931749 1.5349592 -0.15181436    2
## 5 -0.3191788 -1.6170986 0.5393486 -1.24794521    2
## 6  1.3790258 -0.5692512 2.5305698  0.94431648    2
```

Hide

```
class_prob(f1$`equal var 5`)$avg.mostlikely %>% round(., 2)
```

```
##            meanprob.class1 meanprob.class2 meanprob.class3 meanprob.class4
## assigned.1            0.89            0.11            0.00            0.00
## assigned.2            0.01            0.90            0.09            0.00
## assigned.3            0.00            0.11            0.88            0.01
## assigned.4            0.00            0.01            0.03            0.97
## assigned.5            0.00            0.00            0.00            0.02
##            meanprob.class5
## assigned.1            0.00
## assigned.2            0.00
## assigned.3            0.00
## assigned.4            0.00
## assigned.5            0.98
```

Hide

```
class_prob(m1$`equal var 4`)$avg.mostlikely %>% round(., 2)
```

```
##            meanprob.class1 meanprob.class2 meanprob.class3 meanprob.class4
## assigned.1            0.94            0.05            0.00            0.02
## assigned.2            0.12            0.87            0.00            0.01
## assigned.3            0.00            0.00            0.95            0.05
## assigned.4            0.04            0.02            0.01            0.93
```

Comparison with respect to D

Hide

```
bmf$dtb <- 6-bmf$dtb
bmf$dtd <- 6-bmf$dtd
bmf$dtg <- 6-bmf$dtg
bmf$dti <- 6-bmf$dti
bmf$dtl <- 6-bmf$dtl
bmf$dtm <- 6-bmf$dtm
bmf$dtn <- 6-bmf$dtn
bmf$dto <- 6-bmf$dto

mod <- 'dark =~ dtp + dta + dtc + dtd + dte + dtf + dtg + dth + dti + dtj + dtk + dtl +
dtm + dtn + dto + dtb'

modmod <- sem(mod, estimator = "MLR", data = bmf)
summary(modmod, standardized = T)
```

```
## lavaan 0.6.15 ended normally after 32 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        32
##
##   Number of observations                           823
##
## Model Test User Model:
##                                              Standard      Scaled
##   Test Statistic                              558.997     487.598
##   Degrees of freedom                              104         104
##   P-value (Chi-square)                          0.000       0.000
##   Scaling correction factor                                 1.146
##     Yuan-Bentler correction (Mplus variant)
##
## Parameter Estimates:
##
##   Standard errors                             Sandwich
##   Information bread                           Observed
##   Observed information based on                Hessian
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   dark =~
##     dtp               1.000                               0.513    0.453
##     dta               0.848    0.127    6.666    0.000    0.435    0.369
##     dtc               0.744    0.093    8.010    0.000    0.381    0.400
##     dtd               0.795    0.110    7.211    0.000    0.408    0.380
##     dte               0.826    0.079   10.489    0.000    0.423    0.494
##     dtf               1.068    0.119    8.996    0.000    0.548    0.453
##     dtg               0.738    0.115    6.421    0.000    0.379    0.395
##     dth               0.420    0.071    5.923    0.000    0.215    0.283
##     dti               0.744    0.109    6.808    0.000    0.382    0.528
##     dtj               1.048    0.104   10.072    0.000    0.538    0.497
##     dtk               0.837    0.103    8.139    0.000    0.429    0.405
##     dtl               0.853    0.129    6.606    0.000    0.438    0.434
##     dtm               0.661    0.105    6.275    0.000    0.339    0.452
##     dtn               0.987    0.124    7.935    0.000    0.506    0.619
##     dto               0.792    0.114    6.924    0.000    0.406    0.563
##     dtb               0.808    0.118    6.843    0.000    0.414    0.546
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .dtp               1.019    0.074   13.722    0.000    1.019    0.795
##    .dta               1.203    0.060   19.931    0.000    1.203    0.864
##    .dtc               0.762    0.042   18.132    0.000    0.762    0.840
##    .dtd               0.984    0.061   16.175    0.000    0.984    0.856
##    .dte               0.555    0.049   11.291    0.000    0.555    0.756
##    .dtf               1.160    0.056   20.702    0.000    1.160    0.795
##    .dtg               0.778    0.047   16.494    0.000    0.778    0.844
##    .dth               0.531    0.042   12.606    0.000    0.531    0.920
##    .dti               0.377    0.040    9.392    0.000    0.377    0.721
##    .dtj               0.881    0.055   15.925    0.000    0.881    0.753
##    .dtk               0.941    0.045   21.010    0.000    0.941    0.836
```

```
##    .dtl               0.827    0.069   11.933    0.000    0.827    0.812
##    .dtm               0.449    0.046    9.766    0.000    0.449    0.796
##    .dtn               0.412    0.036   11.447    0.000    0.412    0.617
##    .dto               0.355    0.034   10.431    0.000    0.355    0.683
##    .dtb               0.404    0.039   10.480    0.000    0.404    0.702
##     dark              0.263    0.055    4.797    0.000    1.000    1.000
```

Hide

```
fitmeasures(modmod, fit.measures = c("cfi", "rmsea.robust", "rmsea.ci.lower.robust", "r
msea.ci.upper.robust", "srmr"))
```

```
##                   cfi          rmsea.robust rmsea.ci.lower.robust
##                 0.798                 0.072                 0.065
## rmsea.ci.upper.robust                  srmr
##                 0.078                 0.062
```

Hide

```
reliability(modmod)
```

```
##             dark
## alpha  0.7989131
## omega  0.7967372
## omega2 0.7967372
## omega3 0.7848822
## avevar 0.2022979
```

Hide

```
modmodc <- sem(mod, estimator = "MLR", data = bmf, group = "spol")
modmodw <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = "loadi
ngs")
modmods <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = c("loa
dings", "intercepts"))

summary(compareFit(modmodc, modmodw, modmods, argsLRT = list(method ="satorra.bentler.2
010")))
```

```
## Warning in lav_test_diff_SatorraBentler2010(mods[[m]], mods[[m + 1]], H1 = FALSE): l
avaan WARNING: information matrix of the M10 model is not positive definite.

## Warning in lav_test_diff_SatorraBentler2010(mods[[m]], mods[[m + 1]], H1 = FALSE): l
avaan WARNING: information matrix of the M10 model is not positive definite.
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2010")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 208 33174 33626 661.75
## modmodw 223 33212 33594 730.35    -22.112      15          1
## modmods 238 33249 33560 796.74     53.229      15  3.532e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ####################### Model Fit Indices ############################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     595.210†       208          .000        .071†      .771†      .736†
## modmodw     656.181        223          .000        .072       .744       .724
## modmods     716.850        238          .000        .074       .717       .715
##          srmr        aic        bic
## modmodc .064† 33173.885† 33626.329
## modmodw .074  33212.486  33594.236
## modmods .079  33248.882  33559.937†
##
## ################### Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc        15        0.002     -0.027     -0.012 0.010 38.601
## modmods - modmodw        15        0.001     -0.027     -0.009 0.005 36.396
##                        bic
## modmodw - modmodc -32.093
## modmods - modmodw -34.299
```

Hide

```
summary(compareFit(modmodc, modmodw, modmods))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2001")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 208 33174 33626 661.75
## modmodw 223 33212 33594 730.35     60.696      15  1.914e-07 ***
## modmods 238 33249 33560 796.74     61.025      15  1.680e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ####################### Model Fit Indices ############################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     595.210†       208          .000        .071†      .771†      .736†
## modmodw     656.181        223          .000        .072       .744       .724
## modmods     716.850        238          .000        .074       .717       .715
##          srmr        aic        bic
## modmodc .064† 33173.885† 33626.329
## modmodw .074  33212.486  33594.236
## modmods .079  33248.882  33559.937†
##
## ################## Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc        15        0.002     -0.027     -0.012 0.010 38.601
## modmods - modmodw        15        0.001     -0.027     -0.009 0.005 36.396
##                         bic
## modmodw - modmodc -32.093
## modmods - modmodw -34.299
```

Hide

```
modindices(modmodw, free.remove = F) %>% filter(op == "=~") %>% group_by(lhs, op, rhs)
%>% summarize(mi = mean(mi)) %>% arrange(desc(mi)) %>% as.data.frame() %>% head(5)
```

```
## `summarise()` has grouped output by 'lhs', 'op'. You can override using the
## `.groups` argument.
```

```
##    lhs op rhs       mi
## 1 dark =~ dtk 4.942227
## 2 dark =~ dta 3.147290
## 3 dark =~ dti 3.133561
## 4 dark =~ dtl 2.860909
## 5 dark =~ dtm 2.009474
```

Hide

```
modmodw <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = "loadi
ngs", group.partial = c("dark =~ dtk"))
modmods <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = c("loa
dings", "intercepts"), group.partial = c("dark =~ dtk"))

summary(compareFit(modmodc, modmodw, modmods))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2001")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 208 33174 33626 661.75
## modmodw 222 33204 33590 719.69     49.373      14  7.773e-06 ***
## modmods 237 33235 33551 781.31     56.417      15  1.030e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ######################## Model Fit Indices ############################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     595.210†       208          .000        .071†      .771†      .736†
## modmodw     645.067        222          .000        .072       .749       .729
## modmods     701.230        237          .000        .073       .725       .722
##          srmr        aic        bic
## modmodc .064† 33173.885† 33626.329
## modmodw .072  33203.833  33590.295
## modmods .077  33235.446  33551.214†
##
## ################### Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc        14        0.001     -0.022     -0.007 0.008 29.947
## modmods - modmodw        15        0.001     -0.024     -0.007 0.005 31.613
##                         bic
## modmodw - modmodc -36.034
## modmods - modmodw -39.081
```

Hide

```
modindices(modmodw, free.remove = F) %>% filter(op == "=~") %>% group_by(lhs, op, rhs)
%>% summarize(mi = mean(mi)) %>% arrange(desc(mi)) %>% as.data.frame() %>% head(5)
```

```
## `summarise()` has grouped output by 'lhs', 'op'. You can override using the
## `.groups` argument.
```

```
##    lhs op rhs       mi
## 1 dark =~ dta 4.262305
## 2 dark =~ dti 2.885353
## 3 dark =~ dtl 2.846511
## 4 dark =~ dtp 2.703513
## 5 dark =~ dtm 1.821947
```

Hide

```
modmodw <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = "loadi
ngs", group.partial = c("dark =~ dtk", "dark =~ dta"))
modmods <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = c("loa
dings", "intercepts"), group.partial = c("dark =~ dtk", "dark =~ dta"))

summary(compareFit(modmodc, modmodw, modmods))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2001")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 208 33174 33626 661.75
## modmodw 221 33197 33588 710.40     40.709      13  0.0001062 ***
## modmods 236 33219 33539 762.61     47.347      15   3.24e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ######################## Model Fit Indices ############################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     595.210†       208          .000        .071†      .771†      .736†
## modmodw     636.166        221          .000        .071       .754       .732
## modmods     683.463        236          .000        .072       .735       .730
##          srmr        aic        bic
## modmodc .064† 33173.885† 33626.329
## modmodw .071  33196.542  33587.717
## modmods .076  33218.751  33539.232†
##
## ################### Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc        13            0     -0.018     -0.004 0.007 22.656
## modmods - modmodw        15            0     -0.019     -0.002 0.004 22.210
##                        bic
## modmodw - modmodc -38.612
## modmods - modmodw -48.485
```

Hide

```
modindices(modmodw, free.remove = F) %>% filter(op == "=~") %>% group_by(lhs, op, rhs)
%>% summarize(mi = mean(mi)) %>% arrange(desc(mi)) %>% as.data.frame() %>% head(5)
```

```
## `summarise()` has grouped output by 'lhs', 'op'. You can override using the
## `.groups` argument.
```

```
##    lhs op rhs       mi
## 1 dark =~ dtp 3.899379
## 2 dark =~ dtl 2.907460
## 3 dark =~ dti 2.386876
## 4 dark =~ dtf 1.794732
## 5 dark =~ dtm 1.671609
```

Hide

```
modmodw <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = "loadi
ngs", group.partial = c("dark =~ dtk", "dark =~ dtl", "dark =~ dta"))
modmods <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = c("loa
dings", "intercepts"), group.partial = c("dark =~ dtk", "dark =~ dtl", "dark =~ dta"))

summary(compareFit(modmodc, modmodw, modmods))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2001")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 208 33174 33626 661.75
## modmodw 220 33192 33588 704.34     35.698      12  0.0003623 ***
## modmods 235 33208 33533 749.63     41.979      15  0.0002264 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ######################## Model Fit Indices ###########################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     595.210†       208          .000        .071†      .771†      .736†
## modmodw     631.002        220          .000        .071       .756       .734
## modmods     673.014        235          .000        .071       .741       .735
##          srmr        aic        bic
## modmodc .064† 33173.885† 33626.329
## modmodw .070  33192.476  33588.365
## modmods .074  33207.770  33532.964†
##
## ################## Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc        12            0     -0.015     -0.002 0.006 18.591
## modmods - modmodw        15            0     -0.015      0.001 0.004 15.294
##                         bic
## modmodw - modmodc -37.964
## modmods - modmodw -55.401
```

Hide

```
modindices(modmodw, free.remove = F) %>% filter(op == "=~") %>% group_by(lhs, op, rhs)
%>% summarize(mi = mean(mi)) %>% arrange(desc(mi)) %>% as.data.frame() %>% head(5)
```

```
## `summarise()` has grouped output by 'lhs', 'op'. You can override using the
## `.groups` argument.
```

```
##    lhs op rhs       mi
## 1 dark =~ dtp 3.613504
## 2 dark =~ dti 2.761792
## 3 dark =~ dtm 1.988072
## 4 dark =~ dtf 1.602803
## 5 dark =~ dtn 1.408271
```

Hide

```
modmodw <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = "loadi
ngs", group.partial = c("dark =~ dtk", "dark =~ dti", "dark =~ dta", "dark =~ dtl"))
modmods <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = c("loa
dings", "intercepts"), group.partial = c("dark =~ dtk", "dark =~ dti", "dark =~ dta",
"dark =~ dtl"))

summary(compareFit(modmodc, modmodw, modmods))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2001")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 208 33174 33626 661.75
## modmodw 219 33188 33589 698.25     30.107      11   0.001525 **
## modmods 234 33204 33534 744.32     43.081      15   0.000153 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ######################## Model Fit Indices ############################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     595.210†       208          .000        .071       .771†      .736
## modmodw     625.200        219          .000        .071       .759       .736
## modmods     668.270        234          .000        .071†      .743       .736†
##          srmr        aic        bic
## modmodc .064† 33173.885† 33626.329
## modmodw .069  33188.387  33588.988
## modmods .073  33204.460  33534.367†
##
## ################### Differences in Fit Indices ########################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc        11            0     -0.012      0.000 0.005 14.502
## modmods - modmodw        15            0     -0.016      0.001 0.004 16.073
##                         bic
## modmodw - modmodc -37.341
## modmods - modmodw -54.621
```

Hide

```
modindices(modmodw, free.remove = F) %>% filter(op == "=~") %>% group_by(lhs, op, rhs)
%>% summarize(mi = mean(mi)) %>% arrange(desc(mi)) %>% as.data.frame() %>% head(5)
```

```
## `summarise()` has grouped output by 'lhs', 'op'. You can override using the
## `.groups` argument.
```

```
##    lhs op rhs       mi
## 1 dark =~ dtp 3.200015
## 2 dark =~ dtm 2.639744
## 3 dark =~ dtn 2.033937
## 4 dark =~ dtb 1.865375
## 5 dark =~ dtf 1.469068
```

Hide

```
modmodw <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = "loadi
ngs", group.partial = c("dark =~ dtk", "dark =~ dti", "dark =~ dta", "dark =~ dtl", "da
rk =~ dtm"))
modmods <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = c("loa
dings", "intercepts"), group.partial = c("dark =~ dtk", "dark =~ dti", "dark =~ dta",
"dark =~ dtl", "dark =~ dtm"))

summary(compareFit(modmodc, modmodw, modmods))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2001")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 208 33174 33626 661.75
## modmodw 218 33185 33590 692.55     25.759      10  0.0040775 **
## modmods 233 33197 33532 735.20     39.902      15  0.0004692 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ######################## Model Fit Indices ###########################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     595.210†       208          .000        .071       .771†      .736
## modmodw     620.764        218          .000        .071       .761       .737
## modmods     660.777        233          .000        .070†      .747       .739†
##          srmr        aic        bic
## modmodc .064† 33173.885† 33626.329
## modmodw .068  33184.693  33590.007
## modmods .072  33197.340  33531.959†
##
## ################### Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc        10            0     -0.010      0.001 0.004 10.807
## modmods - modmodw        15            0     -0.014      0.002 0.004 12.647
##                         bic
## modmodw - modmodc -36.322
## modmods - modmodw -58.047
```

Hide

```
modindices(modmodw, free.remove = F) %>% filter(op == "=~") %>% group_by(lhs, op, rhs)
%>% summarize(mi = mean(mi)) %>% arrange(desc(mi)) %>% as.data.frame() %>% head(5)
```

```
## `summarise()` has grouped output by 'lhs', 'op'. You can override using the
## `.groups` argument.
```

```
##    lhs op rhs        mi
## 1 dark =~ dtn 2.7389827
## 2 dark =~ dtp 2.7177402
## 3 dark =~ dtb 2.4679390
## 4 dark =~ dtf 1.3530789
## 5 dark =~ dto 0.9296221
```

Hide

```
modmodw <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = "loadi
ngs", group.partial = c("dark =~ dtk", "dark =~ dti", "dark =~ dta", "dark =~ dtl", "da
rk =~ dtm", "dark =~ dtn"))
modmods <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = c("loa
dings", "intercepts"), group.partial = c("dark =~ dtk", "dark =~ dti", "dark =~ dta",
"dark =~ dtm", "dark =~ dtb", "dark =~ dtn"))

summary(compareFit(modmodc, modmodw, modmods))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2001")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 208 33174 33626 661.75
## modmodw 217 33180 33590 685.82     19.799       9    0.01919 *
## modmods 232 33208 33548 744.31     56.146      15  1.145e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ######################## Model Fit Indices ###########################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     595.210†       208          .000        .071       .771†      .736
## modmodw     614.476        217          .000        .070†      .764       .739†
## modmods     669.767        232          .000        .071       .741       .732
##          srmr        aic        bic
## modmodc .064† 33173.885† 33626.329
## modmodw .067  33179.963  33589.991
## modmods .073  33208.453  33547.786†
##
## ################### Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc         9        0.000     -0.007      0.003 0.003  6.078
## modmods - modmodw        15        0.001     -0.023     -0.007 0.005 28.490
##                        bic
## modmodw - modmodc -36.339
## modmods - modmodw -42.204
```

Hide

```
modindices(modmodw, free.remove = F) %>% filter(op == "=~") %>% group_by(lhs, op, rhs)
%>% summarize(mi = mean(mi)) %>% arrange(desc(mi)) %>% as.data.frame() %>% head(5)
```

```
## `summarise()` has grouped output by 'lhs', 'op'. You can override using the
## `.groups` argument.
```

```
##    lhs op rhs        mi
## 1 dark =~ dtb 3.9268256
## 2 dark =~ dtp 2.1185082
## 3 dark =~ dto 2.1036459
## 4 dark =~ dtf 0.9543726
## 5 dark =~ dtj 0.4284381
```

Hide

```
modmodw <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = "loadi
ngs", group.partial = c("dark =~ dtk", "dark =~ dti", "dark =~ dta", "dark =~ dtl", "da
rk =~ dtm", "dark =~ dtn", "dark =~ dtb"))
modmods <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = c("loa
dings", "intercepts"), group.partial = c("dark =~ dtk", "dark =~ dti", "dark =~ dta",
"dark =~ dtl", "dark =~ dtm", "dark =~ dtn", "dark =~ dtb"))

summary(compareFit(modmodc, modmodw, modmods))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2001")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 208 33174 33626 661.75
## modmodw 216 33172 33587 676.10     12.270       8     0.1396
## modmods 231 33189 33533 722.79     44.997      15  7.667e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ######################## Model Fit Indices ############################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     595.210†       208          .000        .071       .771†       .736
## modmodw     606.944        216          .000        .070†      .769        .743†
## modmods     651.757        231          .000        .070       .752        .742
##          srmr        aic        bic
## modmodc .064† 33173.885  33626.329
## modmodw .066  33172.243† 33586.983
## modmods .070  33188.934  33532.980†
##
## ################### Differences in Fit Indices ########################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc         8       -0.001     -0.003      0.007 0.002 -1.642
## modmods - modmodw        15        0.000     -0.017      0.000 0.004 16.691
##                         bic
## modmodw - modmodc -39.346
## modmods - modmodw -54.004
```

Hide

```
modindices(modmods, free.remove = F) %>% filter(op == "~1") %>% group_by(lhs, op, rhs)
%>% summarize(mi = mean(mi)) %>% arrange(desc(mi)) %>% as.data.frame() %>% head(5)
```

```
## `summarise()` has grouped output by 'lhs', 'op'. You can override using the
## `.groups` argument.
```

```
##    lhs op rhs       mi
## 1  dtf ~1     7.274054
## 2  dth ~1     3.872502
## 3  dta ~1     2.209063
## 4  dte ~1     1.898110
## 5 dark ~1     1.330189
```

Hide

```
modmods <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = c("loa
dings", "intercepts"), group.partial = c("dark =~ dtk", "dark =~ dti", "dark =~ dta",
"dark =~ dtl", "dark =~ dtb", "dark =~ dtn", "dark =~ dtm", "dtf ~ 1"))

summary(compareFit(modmodc, modmodw, modmods))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2001")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 208 33174 33626 661.75
## modmodw 216 33172 33587 676.10     12.270       8    0.13957
## modmods 230 33173 33522 704.84     28.229      14    0.01326 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ######################## Model Fit Indices ###########################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     595.210†       208          .000        .071       .771†       .736
## modmodw     606.944        216          .000        .070       .769        .743
## modmods     636.078        230          .000        .069†      .761        .750†
##          srmr        aic        bic
## modmodc .064† 33173.885  33626.329
## modmodw .066  33172.243† 33586.983
## modmods .068  33172.976  33521.735†
##
## ################### Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc         8       -0.001     -0.003      0.007 0.002 -1.642
## modmods - modmodw        14       -0.001     -0.008      0.008 0.003  0.733
##                         bic
## modmodw - modmodc -39.346
## modmods - modmodw -65.248
```

Hide

```
modindices(modmods, free.remove = F) %>% filter(op == "~1") %>% group_by(lhs, op, rhs)
%>% summarize(mi = mean(mi)) %>% arrange(desc(mi)) %>% as.data.frame() %>% head(5)
```

```
## `summarise()` has grouped output by 'lhs', 'op'. You can override using the
## `.groups` argument.
```

```
##    lhs op rhs       mi
## 1  dth ~1     4.422775
## 2  dta ~1     1.936121
## 3 dark ~1     1.906784
## 4  dte ~1     1.356895
## 5  dtl ~1     1.252925
```

Hide

```
modmods <- sem(mod, estimator = "MLR", data = bmf, group = "spol", group.equal = c("loa
dings", "intercepts"), group.partial = c("dark =~ dtk", "dark =~ dti", "dark =~ dta",
"dark =~ dtl", "dark =~ dtb", "dark =~ dtn", "dark =~ dtm", "dtf ~ 1", "dth ~ 1"))

summary(compareFit(modmodc, modmodw, modmods))
```

```
## ################### Nested Model Comparison #########################
##
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2001")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##          Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## modmodc 208 33174 33626 661.75
## modmodw 216 33172 33587 676.10     12.270       8     0.1396
## modmods 229 33164 33518 694.35     17.994      13     0.1577
##
## ######################## Model Fit Indices ############################
##         chisq.scaled df.scaled pvalue.scaled rmsea.robust cfi.robust tli.robust
## modmodc     595.210†       208          .000        .071       .771†       .736
## modmodw     606.944        216          .000        .070       .769        .743
## modmods     626.519        229          .000        .068†      .766        .755†
##          srmr        aic        bic
## modmodc .064† 33173.885  33626.329
## modmodw .066  33172.243  33586.983
## modmods .068  33164.485† 33517.956†
##
## ################## Differences in Fit Indices ######################
##                   df.scaled rmsea.robust cfi.robust tli.robust  srmr    aic
## modmodw - modmodc         8       -0.001     -0.003      0.007 0.002 -1.642
## modmods - modmodw        13       -0.002     -0.003      0.012 0.002 -7.758
##                        bic
## modmodw - modmodc -39.346
## modmods - modmodw -69.027
```

Hide

```
dod2 <- lavPredict(modmods, newdata = bmf, assemble = T) %>% as.data.frame()
describe(poms(dod2))
```

```
##      vars   n mean   sd median trimmed  mad min max range  skew kurtosis   se
## dark    1 823 0.27 0.15   0.25    0.26 0.13   0   1     1  1.02     2.10 0.01
## spol    2 823 0.55 0.50   1.00    0.56 0.00   0   1     1 -0.20    -1.96 0.02
```

Hide

```
bmf2 <- cbind(bmf, dod2)

bf2 <- subset(bmf2, spol == 2)
dim(bf2)
```

```
## [1] 452  58
```

Hide

```
bm2 <- subset(bmf2, spol == 1)
dim(bm2)
```

```
## [1] 371  58
```

Women

Hide

```
names(bf)
```

```
##  [1] "dta"      "dtb"      "dtc"      "dtd"      "dte"      "dtf"
##  [7] "dtg"      "dth"      "dti"      "dtj"      "dtk"      "dtl"
## [13] "dtm"      "dtn"      "dto"      "dtp"      "erda"     "erdb"
## [19] "erdc"     "erdd"     "erde"     "frda"     "frdb"     "frdc"
## [25] "frdd"     "frde"     "odg"      "vatta"    "vattb"    "vattc"
## [31] "vattd"    "vatte"    "vattf"    "vattg"    "vatth"    "acta"
## [37] "actb"     "actc"     "actd"     "rada"     "radb"     "radc"
## [43] "radd"     "spol"     "dob"      "edu"      "Q1"       "studij"
## [49] "standard" "Q2"       "grp"      "att_rad"  "att_act"  "int_rad"
## [55] "int_act"  "spo"      "klase3"   "klase4"   "klase5"   "klase6"
## [61] "klase7"
```

Hide

```
res <- BCH(f1$`equal var 5`, "dark ~ 1", data = bf2$dark)
```

```
## Running aux with 10 parameters
```

Hide

```
lr_test(res)
```

```
## Running aux with 2 parameters
```

```
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
```

```
## BCH test for equality of means across classes
##
## Overall likelihood ratio test:
##  LL_baseline LL_restricted   LL_dif df            p
##     409.4216      478.7343 69.31269  8 6.733512e-12
##
## Pairwise comparisons using likelihood ratio tests:
##  Model1 Model2 LL_baseline LL_restricted     LL_dif df            p
##  class1 class2    409.4216      410.0083  0.5867508  2 7.457421e-01
##  class1 class3    409.4216      410.6451  1.2235146  2 5.423969e-01
##  class2 class3    409.4216      412.2023  2.7807038  2 2.489877e-01
##  class1 class4    409.4216      412.3912  2.9696067  2 2.265469e-01
##  class2 class4    409.4216      445.0762 35.6546763  2 1.810028e-08
##  class3 class4    409.4216      439.8539 30.4323670  2 2.464311e-07
##  class1 class5    409.4216      415.1219  5.7002792  2 5.783625e-02
##  class2 class5    409.4216      444.5253 35.1036983  2 2.384124e-08
##  class3 class5    409.4216      440.0539 30.6323593  2 2.229809e-07
##  class4 class5    409.4216      411.4456  2.0240282  2 3.634861e-01
```

Hide

```
table_results(res) %>% filter(label == "Means.y")
```

```
##     label est_sig   se pval        confint  group
## 1 Means.y    0.05 0.11 0.67  [-0.17, 0.26] class1
## 2 Means.y   -0.04 0.02 0.11  [-0.09, 0.01] class2
## 3 Means.y  -0.08* 0.03 0.01 [-0.14, -0.02] class3
## 4 Means.y 0.25*** 0.06 0.00   [0.13, 0.36] class4
## 5 Means.y 0.40*** 0.10 0.00   [0.22, 0.59] class5
```

Hide

```
bf2$dd <- rowMeans(bf2[, 1:16])
res <- BCH(f1$`equal var 5`, "dd ~ 1", data = bf2$dd)
```

```
## Running aux with 10 parameters
```

Hide

```
lr_test(res)
```

```
## Running aux with 2 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
## Running aux with 8 parameters
```

```
## BCH test for equality of means across classes
##
## Overall likelihood ratio test:
##  LL_baseline LL_restricted   LL_dif df            p
##      393.532      459.7215 66.18957  8 2.806543e-11
##
## Pairwise comparisons using likelihood ratio tests:
##  Model1 Model2 LL_baseline LL_restricted     LL_dif df            p
##  class1 class2     393.532      393.9952  0.4632065  2 7.932608e-01
##  class1 class3     393.532      394.7490  1.2170306  2 5.441582e-01
##  class2 class3     393.532      397.8016  4.2696893  2 1.182630e-01
##  class1 class4     393.532      396.5457  3.0137887  2 2.215971e-01
##  class2 class4     393.532      426.8119 33.2799290  2 5.934105e-08
##  class3 class4     393.532      420.5701 27.0381077  2 1.345084e-06
##  class1 class5     393.532      399.6101  6.0781119  2 4.788007e-02
##  class2 class5     393.532      428.3235 34.7915832  2 2.786787e-08
##  class3 class5     393.532      421.8545 28.3225850  2 7.076668e-07
##  class4 class5     393.532      395.8671  2.3351104  2 3.111267e-01
```

Hide

```
table_results(res) %>% filter(label == "Means.y")
```

```
##     label est_sig   se pval      confint  group
## 1 Means.y 1.78*** 0.10 0.00 [1.57, 1.98] class1
## 2 Means.y 1.70*** 0.02 0.00 [1.66, 1.75] class2
## 3 Means.y 1.66*** 0.03 0.00 [1.60, 1.72] class3
## 4 Means.y 1.98*** 0.05 0.00 [1.87, 2.08] class4
## 5 Means.y 2.14*** 0.09 0.00 [1.96, 2.31] class5
```

Men

Hide

```
names(bm)
```

```
##  [1] "dta"      "dtb"      "dtc"      "dtd"      "dte"      "dtf"
##  [7] "dtg"      "dth"      "dti"      "dtj"      "dtk"      "dtl"
## [13] "dtm"      "dtn"      "dto"      "dtp"      "erda"     "erdb"
## [19] "erdc"     "erdd"     "erde"     "frda"     "frdb"     "frdc"
## [25] "frdd"     "frde"     "odg"      "vatta"    "vattb"    "vattc"
## [31] "vattd"    "vatte"    "vattf"    "vattg"    "vatth"    "acta"
## [37] "actb"     "actc"     "actd"     "rada"     "radb"     "radc"
## [43] "radd"     "spol"     "dob"      "edu"      "Q1"       "studij"
## [49] "standard" "Q2"       "grp"      "att_rad"  "att_act"  "int_rad"
## [55] "int_act"  "spo"      "klase3"   "klase4"   "klase5"   "klase6"
## [61] "klase7"
```

Hide

```
res <- BCH(m1$`equal var 4`, "dark ~ 1", data = bm2$dark)
```

```
## Running aux with 8 parameters
```

Hide

```
lr_test(res)
```

```
## Running aux with 2 parameters
```

```
## Running aux with 6 parameters
## Running aux with 6 parameters
## Running aux with 6 parameters
## Running aux with 6 parameters
## Running aux with 6 parameters
## Running aux with 6 parameters
```

```
## BCH test for equality of means across classes
##
## Overall likelihood ratio test:
##  LL_baseline LL_restricted   LL_dif df           p
##     503.0987      522.6984 19.59965  6 0.003262124
##
## Pairwise comparisons using likelihood ratio tests:
##  Model1 Model2 LL_baseline LL_restricted     LL_dif df            p
##  class1 class2    503.0987      515.4785 12.3798282  2 0.0020500028
##  class1 class3    503.0987      503.3469  0.2481511  2 0.8833130988
##  class2 class3    503.0987      510.5457  7.4470019  2 0.0241492746
##  class1 class4    503.0987      506.1217  3.0230264  2 0.2205759511
##  class2 class4    503.0987      521.1331 18.0343926  2 0.0001213058
##  class3 class4    503.0987      503.7384  0.6397009  2 0.7262576242
```

Hide

```
table_results(res) %>% filter(label == "Means.y")
```

```
##     label est_sig   se pval      confint  group
## 1 Means.y 0.40*** 0.03 0.00 [0.33, 0.47] class1
## 2 Means.y   0.14* 0.06 0.03 [0.02, 0.26] class2
## 3 Means.y 0.45*** 0.10 0.00 [0.26, 0.64] class3
## 4 Means.y 0.48*** 0.05 0.00 [0.39, 0.58] class4
```

Hide

```
bm2$dd <- rowMeans(bm2[, 1:16])
res <- BCH(m1$`equal var 4`, "dd ~ 1", data = bm2$dd)
```

```
## Running aux with 8 parameters
```

Hide

```
lr_test(res)
```

```
## Running aux with 2 parameters
## Running aux with 6 parameters
## Running aux with 6 parameters
## Running aux with 6 parameters
## Running aux with 6 parameters
## Running aux with 6 parameters
## Running aux with 6 parameters
```

```
## BCH test for equality of means across classes
##
## Overall likelihood ratio test:
##  LL_baseline LL_restricted   LL_dif df            p
##     510.1963      532.7845 22.58821  6 0.0009468055
##
## Pairwise comparisons using likelihood ratio tests:
##  Model1 Model2 LL_baseline LL_restricted     LL_dif df            p
##  class1 class2    510.1963      523.6702 13.4738982  2 1.186261e-03
##  class1 class3    510.1963      510.7776  0.5812526  2 7.477951e-01
##  class2 class3    510.1963      518.8877  8.6914173  2 1.296232e-02
##  class1 class4    510.1963      513.7495  3.5532273  2 1.692102e-01
##  class2 class4    510.1963      530.8027 20.6063916  2 3.352578e-05
##  class3 class4    510.1963      510.8562  0.6598966  2 7.189609e-01
```

Hide

```
table_results(res) %>% filter(label == "Means.y")
```

```
##     label est_sig   se pval      confint  group
## 1 Means.y 2.10*** 0.03 0.00 [2.03, 2.17] class1
## 2 Means.y 1.82*** 0.07 0.00 [1.69, 1.95] class2
## 3 Means.y 2.18*** 0.10 0.00 [1.99, 2.37] class3
## 4 Means.y 2.19*** 0.05 0.00 [2.10, 2.29] class4
```

Additional comparisons

Male vs female subsample

Hide

```
t.test(bmf2$dob ~ bmf2$spol)
```

```
##
##  Welch Two Sample t-test
##
## data:  bmf2$dob by bmf2$spol
## t = 0.36405, df = 809.26, p-value = 0.7159
## alternative hypothesis: true difference in means between group 1 and group 2 is not
equal to 0
## 95 percent confidence interval:
##  -0.2868331  0.4174536
## sample estimates:
## mean in group 1 mean in group 2
##        21.87062        21.80531
```

Hide

```
t.test(bmf2$edu ~ bmf2$spol)
```

```
##
##  Welch Two Sample t-test
##
## data:  bmf2$edu by bmf2$spol
## t = -0.46558, df = 783.7, p-value = 0.6416
## alternative hypothesis: true difference in means between group 1 and group 2 is not
equal to 0
## 95 percent confidence interval:
##  -0.4984469  0.3073347
## sample estimates:
## mean in group 1 mean in group 2
##        10.82480        10.92035
```

Hide

```
t.test(bmf2$standard ~ bmf2$spol)
```

```
##
##  Welch Two Sample t-test
##
## data:  bmf2$standard by bmf2$spol
## t = 1.9435, df = 761.13, p-value = 0.05232
## alternative hypothesis: true difference in means between group 1 and group 2 is not
equal to 0
## 95 percent confidence interval:
##  -0.0009096447  0.1815861230
## sample estimates:
## mean in group 1 mean in group 2
##        3.269542        3.179204
```