Project analysis

Yichi Zhang

2023-04-21

```r
## raw dataset
data("HolzingerSwineford1939")
data <- HolzingerSwineford1939[157:301, -c(1:6)]
rownames(data) <- 1:nrow(data)
datasummary_skim(data)
```

```r
# CFA
path <- '
f1 =~ x1 + x2 + x3
f2 =~ x4 + x5 + x6
f3 =~ x7 + x8 + x9
'
model <- cfa(path, data = data, estimator = "MLM")
summary(model, fit.measures = TRUE)
```

```
## lavaan 0.6.15 ended normally after 34 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        21
##
##   Number of observations                           145
##
## Model Test User Model:
##                                               Standard      Scaled
##   Test Statistic                                51.542      49.373
##   Degrees of freedom                                24          24
```

	Unique (#)	Missing (%)	Mean	SD	Min	Median	Max	
x1	30	0	4.9	1.2	1.8	5.0	8.5	
x2	20	0	6.2	1.1	2.2	6.2	9.2	
x3	32	0	2.0	1.0	0.4	1.9	4.5	
x4	18	0	3.3	1.1	0.3	3.0	6.3	
x5	22	0	4.7	1.2	1.0	4.8	7.0	
x6	35	0	2.5	1.1	0.3	2.3	5.9	
x7	71	0	3.9	1.0	1.3	3.9	6.5	
x8	69	0	5.5	1.0	3.0	5.5	10.0	
x9	85	0	5.3	1.0	3.1	5.3	9.2	

```
##   P-value (Chi-square)                          0.001       0.002
##   Scaling correction factor                                 1.044
##     Satorra-Bentler correction
##
## Model Test Baseline Model:
##
##   Test statistic                              505.767     398.008
##   Degrees of freedom                               36          36
##   P-value                                       0.000       0.000
##   Scaling correction factor                                 1.271
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                   0.941       0.930
##   Tucker-Lewis Index (TLI)                      0.912       0.895
##
##   Robust Comparative Fit Index (CFI)                        0.942
##   Robust Tucker-Lewis Index (TLI)                           0.914
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)             -1734.889   -1734.889
##   Loglikelihood unrestricted model (H1)     -1709.118   -1709.118
##
##   Akaike (AIC)                               3511.778    3511.778
##   Bayesian (BIC)                             3574.289    3574.289
##   Sample-size adjusted Bayesian (SABIC)      3507.838    3507.838
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                         0.089       0.085
##   90 Percent confidence interval - lower        0.055       0.052
##   90 Percent confidence interval - upper        0.122       0.119
##   P-value H_0: RMSEA <= 0.050                   0.031       0.043
##   P-value H_0: RMSEA >= 0.080                   0.695       0.633
##
##   Robust RMSEA                                              0.087
##   90 Percent confidence interval - lower                    0.052
##   90 Percent confidence interval - upper                    0.122
##   P-value H_0: Robust RMSEA <= 0.050                        0.042
##   P-value H_0: Robust RMSEA >= 0.080                        0.661
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                          0.072       0.072
##
## Parameter Estimates:
##
##   Standard errors                             Robust.sem
##   Information                                   Expected
##   Information saturated (h1) model            Structured
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
```

```
##   f1 =~
##     x1                1.000
##     x2                0.736    0.162    4.532    0.000
##     x3                0.925    0.179    5.155    0.000
##   f2 =~
##     x4                1.000
##     x5                0.990    0.088   11.309    0.000
##     x6                0.963    0.091   10.639    0.000
##   f3 =~
##     x7                1.000
##     x8                1.226    0.162    7.563    0.000
##     x9                1.058    0.142    7.442    0.000
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   f1 ~~
##     f2                0.408    0.105    3.883    0.000
##     f3                0.276    0.082    3.357    0.001
##   f2 ~~
##     f3                0.222    0.094    2.367    0.018
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .x1                0.715    0.167    4.270    0.000
##    .x2                0.899    0.140    6.443    0.000
##    .x3                0.557    0.107    5.201    0.000
##    .x4                0.315    0.066    4.781    0.000
##    .x5                0.419    0.071    5.867    0.000
##    .x6                0.406    0.076    5.358    0.000
##    .x7                0.600    0.080    7.543    0.000
##    .x8                0.401    0.112    3.593    0.000
##    .x9                0.535    0.084    6.344    0.000
##     f1                0.604    0.180    3.352    0.001
##     f2                0.942    0.162    5.799    0.000
##     f3                0.461    0.108    4.253    0.000
```

```r
resid <- resid(model)
```

```r
## builtin function from lavaan to extract lambda matrix
Lambda <- lavaan::inspect(model, what = "est")$lambda
## extract factor variance covariance matrix directly
Phi <- lavaan::inspect(model, what = "est")$psi
```

```r
# ## focus on one factor first
mod1 <- 'f1 =~ x1 + x2 + x3'
## add sex as a grouping variable
data_gp <- cbind(data, "sex" = HolzingerSwineford1939[157:301,]$sex)
## fit a one factor cfa model
fit_dat1 <- cfa(mod1, data = data_gp,
                group = "sex",
                estimator = "MLM",
                std.lv = TRUE)
summary(fit_dat1, fit.measures = TRUE)
```

```
## lavaan 0.6.15 ended normally after 27 iterations
```

```
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        18
##
##   Number of observations per group:
##     1                                               72
##     2                                               73
##
## Model Test User Model:
##                                               Standard      Scaled
##   Test Statistic                                 0.000       0.000
##   Degrees of freedom                                 0           0
##   Test statistic for each group:
##     1                                            0.000       0.000
##     2                                            0.000       0.000
##
## Model Test Baseline Model:
##
##   Test statistic                                70.660      60.735
##   Degrees of freedom                                 6           6
##   P-value                                        0.000       0.000
##   Scaling correction factor                                  1.163
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    1.000       1.000
##   Tucker-Lewis Index (TLI)                       1.000       1.000
##
##   Robust Comparative Fit Index (CFI)                            NA
##   Robust Tucker-Lewis Index (TLI)                               NA
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)               -620.061    -620.061
##   Loglikelihood unrestricted model (H1)       -620.061    -620.061
##
##   Akaike (AIC)                                1276.122    1276.122
##   Bayesian (BIC)                              1329.703    1329.703
##   Sample-size adjusted Bayesian (SABIC)       1272.745    1272.745
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.000          NA
##   90 Percent confidence interval - lower         0.000          NA
##   90 Percent confidence interval - upper         0.000          NA
##   P-value H_0: RMSEA <= 0.050                       NA          NA
##   P-value H_0: RMSEA >= 0.080                       NA          NA
##
##   Robust RMSEA                                               0.000
##   90 Percent confidence interval - lower                     0.000
##   90 Percent confidence interval - upper                     0.000
##   P-value H_0: Robust RMSEA <= 0.050                            NA
##   P-value H_0: Robust RMSEA >= 0.080                            NA
```

```
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.000       0.000
##
## Parameter Estimates:
##
##   Standard errors                             Robust.sem
##   Information                                   Expected
##   Information saturated (h1) model            Structured
##
##
## Group 1 [1]:
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   f1 =~
##     x1                0.500    0.233    2.146    0.032
##     x2                0.403    0.139    2.904    0.004
##     x3                1.004    0.337    2.979    0.003
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .x1                4.975    0.136   36.697    0.000
##    .x2                6.226    0.128   48.543    0.000
##    .x3                2.139    0.126   16.976    0.000
##     f1                0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .x1                1.073    0.294    3.656    0.000
##    .x2                1.022    0.196    5.213    0.000
##    .x3                0.135    0.663    0.204    0.838
##     f1                1.000
##
##
## Group 2 [2]:
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   f1 =~
##     x1                0.795    0.131    6.081    0.000
##     x2                0.783    0.147    5.307    0.000
##     x3                0.695    0.122    5.718    0.000
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .x1                4.886    0.134   36.467    0.000
##    .x2                6.175    0.132   46.877    0.000
##    .x3                1.854    0.115   16.134    0.000
##     f1                0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
```

```
##    .x1                 0.679    0.194    3.506    0.000
##    .x2                 0.654    0.178    3.682    0.000
##    .x3                 0.481    0.124    3.864    0.000
##     f1                 1.000
```

```
## original data
res1 <- alignment(fit_dat1, group_name = c("female", "male"))
```

Here is the second dataset with good leverage points.

```
# Data set 2 -- Good leverage points
h = c(0.3881, 1.3762, 5.6153, 2.4312, 1.7442)
bartlett_predict <- lavPredict(model, method = "Bartlett")

data2_new_ob <- foreach(i = 141:145, j = 1:5) %do%{
  data[i, ] + h[[j]]*Lambda%*%bartlett_predict[i, ]
}

data2_new_ob <-data.frame(t(sapply(data2_new_ob,c)))

#replace the last 5 observations and the whole data set 2 is:
data2 <- rbind(data, data2_new_ob)
data2 <- data2[-(141:145), ]
rownames(data2) <- 1:nrow(data2)
data2 <- data.frame(matrix(unlist(data2), ncol=length(data2), byrow=FALSE))
data2 <- cbind(data2, "sex" = HolzingerSwineford1939[157:301,]$sex)
data2_fe <- data2 %>% filter(sex == 1)
data2_ma <- data2 %>% filter(sex == 2)
```

mve

```
res_mve <- robalign(method = "mve_mah", data_g1 = data2_fe[,1:3], data_g2 = data2_ma[,1:3],
                    mod = mod2, group_name = c("female", "male"))
```

```
##
## Attaching package: 'MASS'

## The following object is masked _by_ '.GlobalEnv':
##
##     ltsreg

## The following object is masked from 'package:dplyr':
##
##     select
```

mcd

```
res_mcd <- robalign(method = "mcd_mah", data_g1 = data2_fe[,1:3], data_g2 = data2_ma[,1:3],
                    mod = mod2, group_name = c("female", "male"))
# mean_rob_mcd <- mcd_mahalanobis(data2[,1:3])$weighted.mean
# cov_rob_mcd <- mcd_mahalanobis(data2[,1:3])$weighted.covariance
# fit_dat3 <- cfa(mod2,
#                  sample.cov = cov_rob_mcd,
#                  sample.mean = mean_rob_mcd,
#                  sample.nobs = 145,
#                  group = "sex",
```

```r
#                  std.lv = TRUE)
# coef(fit_dat3)
# res3 <- alignment(fit_dat3)
```

projection mve

```r
res_pro_mve <- robalign(method = "pro_mve", data_g1 = data2_fe[,1:3], data_g2 = data2_ma[,1:3],
                     mod = mod2, group_name = c("female", "male"))
# mean_rob_pmve <- projection_mve(x = data2[,1:3])$weighted.mean
# cov_rob_pmve <- projection_mve(data2[,1:3])$weighted.covariance
# fit_dat4 <- cfa(mod2,
#                  data = data2,
#                  sample.cov = cov_rob_pmve,
#                  sample.mean = mean_rob_pmve,
#                  sample.nobs = 145,
#                  group = "sex",
#                  std.lv = TRUE)
# # summary(fit_dat2, fit.measures = TRUE)
# res4 <- alignment(fit_dat4)
```

projection mcd

```r
res_pro_mcd <- robalign(method = "pro_mcd", data_g1 = data2_fe[,1:3], data_g2 = data2_ma[,1:3],
                     mod = mod2, group_name = c("female", "male"))
# mean_rob_pmcd <- projection_mcd(x = data2[,1:9])$weighted.mean
# cov_rob_pmcd <- projection_mcd(data2[,1:9])$weighted.covariance
# fit_dat5 <- cfa(mod2,
#                  data = data2,
#                  sample.cov = cov_rob_pmcd,
#                  sample.mean = mean_rob_pmcd,
#                  sample.nobs = 145,
#                  group = "sex",
#                  std.lv = TRUE)
# # summary(fit_dat2, fit.measures = TRUE)
# res5 <- alignment(fit_dat5)
```

outpro mve

```r
res_outmve <- robalign(method = "outpro_mve", data_g1 = data2_fe[,1:3], data_g2 = data2_ma[,1:3], mod =
# mean_rob_omve <- outpro_mcd(data2[,1:9])$weighted.mean
# cov_rob_omve <- outpro_mcd(data2[,1:9])$weighted.covariance
# fit_dat6 <- cfa(mod2,
#                  data = data2,
#                  sample.cov = cov_rob_omve,
#                  sample.mean = mean_rob_omve,
#                  sample.nobs = 145,
#                  group = "sex",
#                  std.lv = TRUE)
# # summary(fit_dat2, fit.measures = TRUE)
# res6 <- alignment(fit_dat6)
```

outpro mcd

```
res_outmcd <- robalign(method = "outpro_mcd", data_g1 = data2_fe[,1:3], data_g2 = data2_ma[,1:3], mod =
# mean_rob_omcd <- outpro_mve(data2[,1:9])$weighted.mean
# cov_rob_omcd <- outpro_mve(data2[,1:9])$weighted.covariance
# fit_dat7 <- cfa(mod2,
#                  data = data2,
#                  sample.cov = cov_rob_omcd,
#                  sample.mean = mean_rob_omcd,
#                  sample.nobs = 145,
#                  group = "sex",
#                  std.lv = TRUE)
# # summary(fit_dat2, fit.measures = TRUE)
# res7 <- alignment(fit_dat7)
```

comparison

```
## factor means and variances
list(res1$pars,
    res_mve$align_res$pars, res_mcd$align_res$pars,
    res_pro_mve$align_res$pars, res_pro_mcd$align_res$pars,
    res_outmve$align_res$pars, res_outmcd$align_res$pars)
```

```
## [[1]]
##            alpha0     psi0
## female  0.0000000 1.000000
## male   -0.1659235 1.631462
##
## [[2]]
##            alpha0     psi0
## female  0.0000000 1.000000
## male   -0.1016879 1.536574
##
## [[3]]
##            alpha0     psi0
## female  0.0000000 1.000000
## male   -0.1145397 1.211929
##
## [[4]]
##            alpha0     psi0
## female  0.0000000 1.000000
## male   -0.8115563 1.153057
##
## [[5]]
##            alpha0     psi0
## female  0.0000000 1.000000
## male   -0.7601359 1.008565
##
## [[6]]
##            alpha0     psi0
## female  0.0000000 1.000000
## male   -0.2026377 1.739594
##
## [[7]]
```

```
##            alpha0     psi0
## female  0.0000000 1.000000
## male   -0.2026376 1.667647
```

```
# ## factor loadings
list(res1$lambda.aligned,
     res_mve$align_res$lambda.aligned, res_mcd$align_res$lambda.aligned,
     res_pro_mve$align_res$lambda.aligned, res_pro_mcd$align_res$lambda.aligned,
     res_outmve$align_res$lambda.aligned, res_outmcd$align_res$lambda.aligned)
```

```
## [[1]]
##               x1        x2        x3
## female 0.4998331 0.4028319 1.0039360
## male   0.4872053 0.4797327 0.4262364
##
## [[2]]
##               V1        V2        V3
## female 0.4728559 0.3212712 0.7791798
## male   0.4663781 0.4398348 0.4182736
##
## [[3]]
##               V1        V2        V3
## female 0.5575191 0.3465450 0.8122413
## male   0.5536329 0.5171441 0.4851572
##
## [[4]]
##               V1        V2        V3
## female 0.3235331 0.2615937 0.6683662
## male   0.3216403 0.2745031 0.2829121
##
## [[5]]
##               V1        V2        V3
## female 0.3480609 0.2776676 0.7136026
## male   0.3420345 0.2984976 0.3105619
##
## [[6]]
##               V1        V2        V3
## female 0.5226324 0.3916314 1.0355731
## male   0.5115216 0.4764864 0.4335098
##
## [[7]]
##               V1        V2        V3
## female 0.5226324 0.3916314 1.0355731
## male   0.5115216 0.4764864 0.4335098
```

```
## factor intercepts
list(res1$nu.aligned,
     res_mve$align_res$nu.aligned, res_mcd$align_res$nu.aligned,
     res_pro_mve$align_res$nu.aligned, res_pro_mcd$align_res$nu.aligned,
     res_outmve$align_res$nu.aligned, res_outmcd$align_res$nu.aligned)
```

```
## [[1]]
##              x1       x2       x3
## female 4.974537 6.225694 2.138889
## male   4.966684 6.254256 1.925175
```

```
##
## [[2]]
##              V1       V2       V3
## female 4.859567 6.034272 2.077100
## male   4.853830 6.131473 1.837711
##
## [[3]]
##              V1       V2       V3
## female 4.860159 6.027849 2.067091
## male   4.854595 6.123279 1.844412
##
## [[4]]
##              V1       V2       V3
## female 4.921436 6.170343 2.106988
## male   4.947723 6.165212 1.983767
##
## [[5]]
##              V1       V2       V3
## female 4.928407 6.179101 2.109541
## male   4.951294 6.176175 1.992370
##
## [[6]]
##              V1       V2       V3
## female 4.953455 6.210175 2.119392
## male   4.946127 6.239284 1.902188
##
## [[7]]
##              V1       V2       V3
## female 4.953455 6.210175 2.119392
## male   4.946127 6.239284 1.902188
```

covariance matrix

```
cov(data_gp[1:3])
```

```
##           x1        x2        x3
## x1 1.3278044 0.4171875 0.5374561
## x2 0.4171875 1.2348958 0.4817708
## x3 0.5374561 0.4817708 1.0808190
```

```
res_outmve$rob_est_g1$weighted.covariance
```

```
##           V1        V2        V3
## V1 1.2335465 0.2075621 0.5488470
## V2 0.2075621 1.0397231 0.4112751
## V3 0.5488470 0.4112751 1.1369259
```

```
res_outmve$rob_est_g2$weighted.covariance
```

```
##           V1        V2        V3
## V1 1.5115165 0.7478262 0.6803761
## V2 0.7478262 1.3805430 0.6337757
## V3 0.6803761 0.6337757 1.0664843
```