SEM: measuring the impact of Adopt Me program

LOCAL ANIMAL SHELTER

A local animal shelter has designed a survey to measure the impact of their Adopt Me program. Viewers rated each dog's picture, background story, and other characteristics to indicate the "adoptableness" of each animal.

The adoptsurvey data contains the six items they rated including pictures, background, loveskids that measure a "good story" latent variable, while energy, wagstail, playful measure an "in person" latent variable. We will build a two-factor model of their survey and examine it for Heywood cases.

```
library(lavaan)
```

```
## This is lavaan 0.6-6
```

```
## lavaan is BETA software! Please report any bugs.
```

```
library(data.table)
library(curl)
adoptsurvey <- fread('https://raw.githubusercontent.com/JiaxiangBU/picbackup/master/adoptsurvey02.csv')
```

```
head(adoptsurvey)
```

```
##     pictures background loveskids    energy   wagstail  playful
## 1:  3.708400 -0.9640867  3.859116 -6.728699 -1.1995000 4.097103
## 2:  1.244440  6.3804313  5.951090  1.606351  0.5322139 1.925454
## 3:  1.192845 -4.3286503  8.231443  4.090618  4.5900018 4.035844
## 4: -1.260835  5.1964583  2.457856  7.596427  3.6990812 4.559570
## 5:  4.575658 -0.1453078  9.527073 -3.134994  2.5460263 3.432766
## 6:  1.959739  6.6615860  5.619911  1.289012  3.3453336 9.074500
```

```
str(adoptsurvey)
```

```
## Classes 'data.table' and 'data.frame':   100 obs. of  6 variables:
##  $ pictures  : num  3.71 1.24 1.19 -1.26 4.58 ...
##  $ background: num  -0.964 6.38 -4.329 5.196 -0.145 ...
##  $ loveskids : num  3.86 5.95 8.23 2.46 9.53 ...
##  $ energy    : num  -6.73 1.61 4.09 7.6 -3.13 ...
##  $ wagstail  : num  -1.199 0.532 4.59 3.699 2.546 ...
##  $ playful   : num  4.1 1.93 4.04 4.56 3.43 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

Build the model

```
adopt.model <- 'goodstory =~ pictures + background + loveskids
inperson =~ energy + wagstail + playful'
```

Analyze the model

```
adopt.fit <- cfa(model = adopt.model, data = adoptsurvey)
```

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

we see an error message warning you that the latent variables are not positive definite.

So, correlation > 1 on the latent variable.

You should fix the Heywood case by collapsing the two latent variables into one latent variable. create only one goodstory factor that is measured by all six manifest variables in the adoptsurvey dataset

Edit the original model

```
adopt.model <- 'goodstory =~ pictures + background + loveskids + energy + wagstail + playful'
```

Analyze the updated model

```
adopt.fit <- cfa(model = adopt.model, data = adoptsurvey)
```

Look for Heywood cases

```
summary(adopt.fit, standardized = TRUE, fit.measures = TRUE)
```

```
## lavaan 0.6-6 ended normally after 56 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of free parameters                         12
##
##   Number of observations                           100
##
## Model Test User Model:
##
##   Test statistic                                 9.627
##   Degrees of freedom                                 9
##   P-value (Chi-square)                           0.382
##
## Model Test Baseline Model:
##
##   Test statistic                                25.380
##   Degrees of freedom                                15
##   P-value                                        0.045
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.940
##   Tucker-Lewis Index (TLI)                       0.899
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -1651.202
```

```
##   Loglikelihood unrestricted model (H1)      -1646.389
##
##   Akaike (AIC)                                3326.404
##   Bayesian (BIC)                              3357.666
##   Sample-size adjusted Bayesian (BIC)         3319.767
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.026
##   90 Percent confidence interval - lower         0.000
##   90 Percent confidence interval - upper         0.117
##   P-value RMSEA <= 0.05                          0.569
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.061
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   goodstory =~
##     pictures          1.000                               1.343    0.432
##     background        1.468    0.756    1.942    0.052    1.972    0.513
##     loveskids         1.815    0.936    1.939    0.052    2.438    0.515
##     energy            0.067    0.380    0.177    0.859    0.090    0.025
##     wagstail         -0.306    0.521   -0.588    0.556   -0.412   -0.086
##     playful          -0.009    0.356   -0.025    0.980   -0.012   -0.004
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .pictures          7.860    1.503    5.228    0.000    7.860    0.813
##    .background       10.873    2.659    4.089    0.000   10.873    0.737
##    .loveskids        16.491    4.052    4.069    0.000   16.491    0.735
##    .energy           12.677    1.794    7.066    0.000   12.677    0.999
##    .wagstail         22.674    3.232    7.016    0.000   22.674    0.993
##    .playful          11.181    1.581    7.071    0.000   11.181    1.000
##     goodstory         1.804    1.287    1.402    0.161    1.000    1.000
```

You will look for a Heywood cases on one of the manifest variables, rather than on the latent variable.

(negative variance)

Build the model

```
adopt.model <- 'goodstory =~ pictures + background + loveskids
inperson =~ energy + wagstail + playful'
```

Analyze the model and include the data argument

```
adopt.fit <- cfa(adopt.model, adoptsurvey)
```

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

Summarize the model to view the negative variances

```
summary(adopt.fit, standardized=TRUE, fit.measures = TRUE)
```

```
## lavaan 0.6-6 ended normally after 300 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of free parameters                         13
##
##   Number of observations                           100
##
## Model Test User Model:
##
##   Test statistic                                 7.134
##   Degrees of freedom                                 8
##   P-value (Chi-square)                           0.522
##
## Model Test Baseline Model:
##
##   Test statistic                                25.380
##   Degrees of freedom                                15
##   P-value                                        0.045
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    1.000
##   Tucker-Lewis Index (TLI)                       1.156
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -1649.956
##   Loglikelihood unrestricted model (H1)      -1646.389
##
##   Akaike (AIC)                                3325.912
##   Bayesian (BIC)                              3359.779
##   Sample-size adjusted Bayesian (BIC)         3318.722
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.000
##   90 Percent confidence interval - lower         0.000
##   90 Percent confidence interval - upper         0.109
##   P-value RMSEA <= 0.05                          0.686
##
## Standardized Root Mean Square Residual:
##
```

```
##   SRMR                                           0.050
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   goodstory =~
##     pictures          1.000                               1.360    0.437
##     background        1.471    0.763    1.928    0.054    2.000    0.521
##     loveskids         1.746    0.892    1.958    0.050    2.375    0.501
##   inperson =~
##     energy            1.000                               0.208    0.058
##     wagstail         45.278 1090.877    0.042    0.967    9.410    1.969
##     playful           0.869    1.110    0.783    0.434    0.181    0.054
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   goodstory ~~
##     inperson         -0.014    0.332   -0.041    0.967   -0.048   -0.048
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .pictures           7.814    1.514    5.162    0.000    7.814    0.809
##    .background        10.762    2.695    3.993    0.000   10.762    0.729
##    .loveskids         16.791    3.936    4.266    0.000   16.791    0.749
##    .energy            12.642    2.066    6.119    0.000   12.642    0.997
##    .wagstail         -65.707 2125.647   -0.031    0.975  -65.707   -2.876
##    .playful           11.148    1.760    6.335    0.000   11.148    0.997
##     goodstory          1.850    1.310    1.411    0.158    1.000    1.000
##     inperson           0.043    1.046    0.041    0.967    1.000    1.000
```

we can see variance is negative for wagstail variable, which is a Heywood case. (-65.707)

HEIWOOD CASES=> Correlations that are out of bounds, Negative variances

Fix the Manifest Heywood Model:

To fix the error in the last model, we can use the var() function to calculate the variance of the manifest variable that is estimated as negative.

Summarize the model to view the negative variances

```
summary(adopt.fit, standardized = TRUE, fit.measures = TRUE, rsquare=TRUE)
```

```
## lavaan 0.6-6 ended normally after 300 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of free parameters                         13
##
##   Number of observations                           100
##
```

```
## Model Test User Model:
##
##   Test statistic                                 7.134
##   Degrees of freedom                                 8
##   P-value (Chi-square)                           0.522
##
## Model Test Baseline Model:
##
##   Test statistic                                25.380
##   Degrees of freedom                                15
##   P-value                                        0.045
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    1.000
##   Tucker-Lewis Index (TLI)                       1.156
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -1649.956
##   Loglikelihood unrestricted model (H1)      -1646.389
##
##   Akaike (AIC)                                3325.912
##   Bayesian (BIC)                              3359.779
##   Sample-size adjusted Bayesian (BIC)         3318.722
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.000
##   90 Percent confidence interval - lower         0.000
##   90 Percent confidence interval - upper         0.109
##   P-value RMSEA <= 0.05                          0.686
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.050
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   goodstory =~
##     pictures          1.000                               1.360    0.437
##     background        1.471    0.763    1.928    0.054    2.000    0.521
##     loveskids         1.746    0.892    1.958    0.050    2.375    0.501
##   inperson =~
##     energy            1.000                               0.208    0.058
##     wagstail         45.278 1090.877    0.042    0.967    9.410    1.969
##     playful           0.869    1.110    0.783    0.434    0.181    0.054
##
```

```
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   goodstory ~~
##     inperson         -0.014    0.332   -0.041    0.967   -0.048   -0.048
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .pictures          7.814    1.514    5.162    0.000    7.814    0.809
##    .background       10.762    2.695    3.993    0.000   10.762    0.729
##    .loveskids        16.791    3.936    4.266    0.000   16.791    0.749
##    .energy           12.642    2.066    6.119    0.000   12.642    0.997
##    .wagstail        -65.707 2125.647   -0.031    0.975  -65.707   -2.876
##    .playful          11.148    1.760    6.335    0.000   11.148    0.997
##     goodstory         1.850    1.310    1.411    0.158    1.000    1.000
##     inperson          0.043    1.046    0.041    0.967    1.000    1.000
##
## R-Square:
##                    Estimate
##     pictures          0.191
##     background        0.271
##     loveskids         0.251
##     energy            0.003
##     wagstail             NA
##     playful           0.003
```

View the variance of the problem manifest variable

```
var(adoptsurvey$wagstail)
```

```
## [1] 23.07446
```

Update the model using 5 decimal places

```
adopt.model2 <- 'goodstory =~ pictures + background + loveskids
inperson =~ energy + wagstail + playful
wagstail ~~ 23.07446 * wagstail'              #THIS LINE
```

Analyze and summarize the updated model

```
adopt.fit2 <- cfa(model = adopt.model2, data = adoptsurvey)
```

Summarize the model to view the negative variances

```
summary(adopt.fit2, standardized = TRUE, fit.measures = TRUE, rsquare=TRUE)
```

```
## lavaan 0.6-6 ended normally after 69 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of free parameters                         12
##
##   Number of observations                           100
##
```

```
## Model Test User Model:
##
##   Test statistic                                8.493
##   Degrees of freedom                                9
##   P-value (Chi-square)                          0.485
##
## Model Test Baseline Model:
##
##   Test statistic                               25.380
##   Degrees of freedom                               15
##   P-value                                       0.045
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                   1.000
##   Tucker-Lewis Index (TLI)                      1.081
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)             -1650.635
##   Loglikelihood unrestricted model (H1)     -1646.389
##
##   Akaike (AIC)                               3325.270
##   Bayesian (BIC)                             3356.532
##   Sample-size adjusted Bayesian (BIC)        3318.633
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                         0.000
##   90 Percent confidence interval - lower        0.000
##   90 Percent confidence interval - upper        0.108
##   P-value RMSEA <= 0.05                         0.664
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                          0.058
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   goodstory =~
##     pictures          1.000                               1.344    0.432
##     background        1.461    0.758    1.928    0.054    1.964    0.511
##     loveskids         1.818    0.947    1.919    0.055    2.444    0.516
##   inperson =~
##     energy            1.000                               0.959    0.269
##     wagstail          1.391    2.244    0.620    0.535    1.334    0.268
##     playful           0.807    1.640    0.492    0.623    0.774    0.231
##
```

```
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   goodstory ~~
##     inperson         -0.077    0.450   -0.172    0.863   -0.060   -0.060
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .wagstail         23.074                              23.074    0.928
##    .pictures          7.857    1.510    5.203    0.000    7.857    0.813
##    .background       10.906    2.672    4.082    0.000   10.906    0.739
##    .loveskids        16.461    4.103    4.012    0.000   16.461    0.734
##    .energy           11.765    2.683    4.385    0.000   11.765    0.928
##    .playful          10.582    2.082    5.084    0.000   10.582    0.946
##     goodstory         1.807    1.296    1.395    0.163    1.000    1.000
##     inperson          0.920    2.209    0.416    0.677    1.000    1.000
##
## R-Square:
##                    Estimate
##     wagstail          0.072
##     pictures          0.187
##     background        0.261
##     loveskids         0.266
##     energy            0.072
##     playful           0.054
```

problem fixed

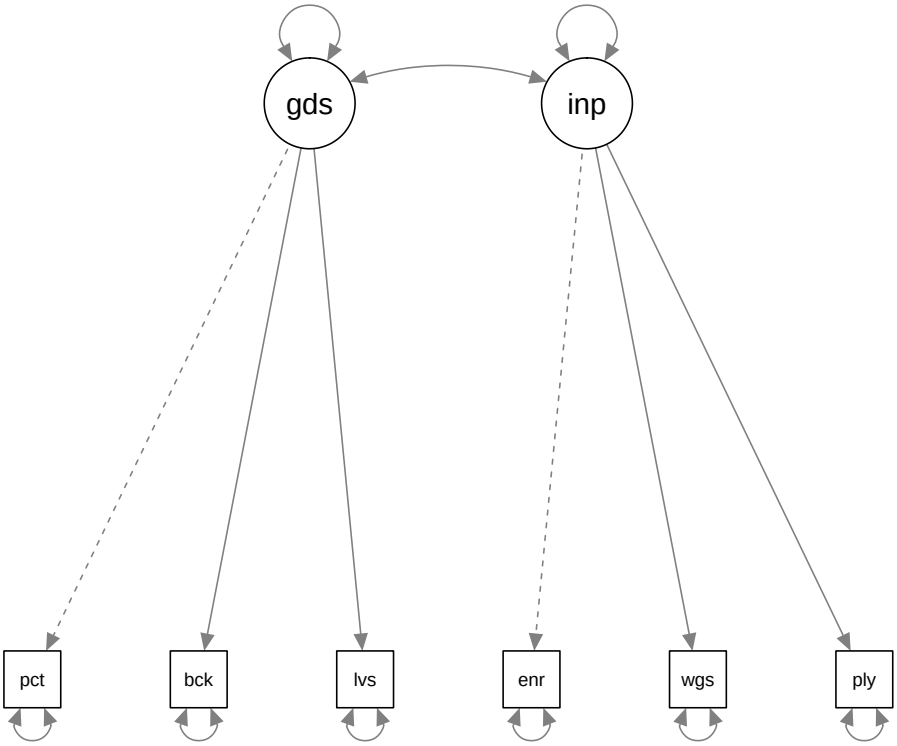
#CREATE DIAGRAMS w/ semPlot library and semPaths() function

Basic diagram

```
# Load the semPlot library
library(semPlot)
```

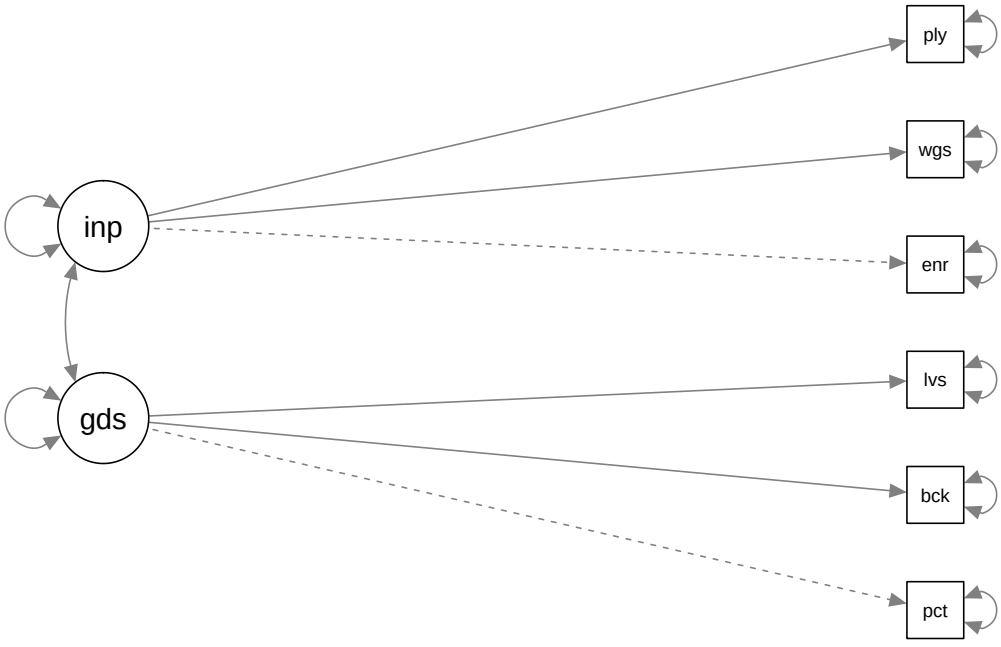
```
## Registered S3 methods overwritten by 'huge':
##   method    from
##   plot.sim  BDgraph
##   print.sim BDgraph
```

```
# Create a default picture
semPaths(adopt.fit)
```



Update the default picture

```
semPaths(object = adopt.fit,
         layout= 'tree',
         rotation = 2)
```



Update the default picture

```
semPaths(object = adopt.fit,
         layout = "tree",
         rotation = 2,
         whatLabels= 'std',
         edge.label.cex = 1,
         what = 'std',
         edge.color = 'blue')
```