

Synthesizing Indirect Effects in Mediation Models with Meta-Analytic Methods: Supplementary Materials 2

Mike Cheung

January 30, 2021

Contents

Data preparation	1
Illustration 1 with one mediator	2
Meta-analyzing the indirect and direct effects	6
Caculation of indirect and direct effects	6
Meta-analysis of indirect and direct effects	6
TSSEM	9
Stage 1 analysis	9
Stage 2 analysis	10
OSMASEM	13
Model without any moderator	13
Model with a moderator	16
Illustration 2 with two parallel mediators (Aut and Cap)	17
Meta-analyzing the indirect and direct effects	20
Caculation of indirect and direct effects	20
Meta-analysis of indirect and direct effects	23
TSSEM	24
Stage 1 analysis	24
Stage 2 analysis	25
Illustration 3 with two serial mediators (Aut and Cap)	27
Meta-analyzing the indirect and direct effects	31
Caculation of indirect and direct effects	31
Meta-analysis of indirect and direct effects	32
TSSEM	33
Stage 1 analysis	33
Stage 2 analysis	34

Data preparation

```
library(metaSEM)

## Get the data from the local file
source("Hagger18.R")

## Check whether the correlation matrices are positive definite
```

```

which(is.pd(Hagger18$data)==FALSE)

## 21
## 21
## Drop the 21th study as the correlation matrix is not positive definite.
Hagger18 <- lapply(Hagger18, function(x) x[-21])

```

Illustration 1 with one mediator

```

## Use df1 as the data file in illustration 1
df1 <- Hagger18

## Select Beh, Int, and PB for the illustration
obs.vars1 <- c("Beh", "Int", "PB")
df1$data <- lapply(df1$data, function(x) x[obs.vars1, obs.vars1])

## NA is not allowed in computing the indirect and direct effects.
## Studies with NA are excluded first.
index1 <- sapply(df1$data, function(x) any(is.na(vechs(x))) )
df1 <- lapply(df1, function(x) x[!index1])

## Show the first few studies
head(df1)

```

```

## $data
## $data$`3`
##      Beh  Int   PB
## Beh 1.000 0.65 0.709
## Int 0.650 1.00 0.650
## PB  0.709 0.65 1.000
##
## $data$`19`
##      Beh  Int   PB
## Beh 1.00 0.84 0.87
## Int 0.84 1.00 0.94
## PB  0.87 0.94 1.00
##
## $data$`20`
##      Beh  Int   PB
## Beh 1.000 0.534 0.768
## Int 0.534 1.000 0.540
## PB  0.768 0.540 1.000
##
## $data$`22`
##      Beh  Int   PB
## Beh 1.00 0.55 0.54
## Int 0.55 1.00 0.32
## PB  0.54 0.32 1.00
##
## $data$`25`
##      Beh  Int   PB
## Beh 1.00 0.53 0.52

```

```

## Int 0.53 1.00 0.78
## PB 0.52 0.78 1.00
##
## $data$`26`
##      Beh  Int   PB
## Beh 1.00 0.53 0.54
## Int 0.53 1.00 0.54
## PB 0.54 0.54 1.00
##
## $data$`29`
##      Beh  Int   PB
## Beh 1.00 0.64 0.63
## Int 0.64 1.00 0.70
## PB 0.63 0.70 1.00
##
## $data$`30`
##      Beh  Int   PB
## Beh 1.000 0.589 0.651
## Int 0.589 1.000 0.539
## PB 0.651 0.539 1.000
##
## $data$`33`
##      Beh  Int   PB
## Beh 1.000 0.742 0.756
## Int 0.742 1.000 0.798
## PB 0.756 0.798 1.000
##
## $data$`34`
##      Beh  Int   PB
## Beh 1.000 0.732 0.780
## Int 0.732 1.000 0.752
## PB 0.780 0.752 1.000
##
## $data$`35`
##      Beh  Int   PB
## Beh 1.00 0.02 0.01
## Int 0.02 1.00 0.22
## PB 0.01 0.22 1.00
##
## $data$`36`
##      Beh  Int   PB
## Beh 1.00 0.3 0.09
## Int 0.30 1.0 0.20
## PB 0.09 0.2 1.00
##
## $data$`37`
##      Beh  Int   PB
## Beh 1.000 0.451 0.746
## Int 0.451 1.000 0.477
## PB 0.746 0.477 1.000
##
## $data$`44`
##      Beh  Int   PB
## Beh 1.00 0.27 0.53

```

```

## Int 0.27 1.00 0.34
## PB 0.53 0.34 1.00
##
## $data$`48`
##      Beh    Int    PB
## Beh 1.000 0.378 0.545
## Int 0.378 1.000 0.485
## PB 0.545 0.485 1.000
##
## $data$`49`
##      Beh    Int    PB
## Beh 1.00 0.22 0.36
## Int 0.22 1.00 0.49
## PB 0.36 0.49 1.00
##
## $data$`51`
##      Beh    Int    PB
## Beh 1.00 0.59 0.6
## Int 0.59 1.00 0.7
## PB 0.60 0.70 1.0
##
## $data$`52`
##      Beh    Int    PB
## Beh 1.00 0.40 0.61
## Int 0.40 1.00 0.41
## PB 0.61 0.41 1.00
##
## $data$`53`
##      Beh    Int    PB
## Beh 1.00 0.42 0.42
## Int 0.42 1.00 0.63
## PB 0.42 0.63 1.00
##
## $data$`54`
##      Beh    Int    PB
## Beh 1.00 0.72 0.46
## Int 0.72 1.00 0.47
## PB 0.46 0.47 1.00
##
## $data$`61`
##      Beh    Int    PB
## Beh 1.00 0.42 0.47
## Int 0.42 1.00 0.34
## PB 0.47 0.34 1.00
##
## $data$`67`
##      Beh    Int    PB
## Beh 1.00 0.44 0.55
## Int 0.44 1.00 0.59
## PB 0.55 0.59 1.00
##
## $data$`73`
##      Beh    Int    PB
## Beh 1.00 0.11 0.05

```

```

## Int 0.11 1.00 0.49
## PB 0.05 0.49 1.00
##
## $data$`74`
##      Beh  Int   PB
## Beh 1.00 0.57 0.71
## Int 0.57 1.00 0.61
## PB 0.71 0.61 1.00
##
## $data$`76`
##      Beh  Int   PB
## Beh 1.000 0.149 0.199
## Int 0.149 1.000 0.224
## PB 0.199 0.224 1.000
##
## $data$`77`
##      Beh  Int   PB
## Beh 1.000 0.261 0.517
## Int 0.261 1.000 0.315
## PB 0.517 0.315 1.000
##
## $data$`78`
##      Beh  Int   PB
## Beh 1.00 0.61 0.37
## Int 0.61 1.00 0.59
## PB 0.37 0.59 1.00
##
## $data$`80`
##      Beh  Int   PB
## Beh 1.00 0.64 0.81
## Int 0.64 1.00 0.78
## PB 0.81 0.78 1.00
##
## $data$`81`
##      Beh  Int   PB
## Beh 1.00 0.52 0.45
## Int 0.52 1.00 0.64
## PB 0.45 0.64 1.00
##
##
## $n
## [1] 413 118 41 146 192 413 1403 133 523 596 174 272 85 365 620
## [16] 743 109 79 273 95 236 153 103 225 139 146 54 225 62
##
## $beh_freq_high
## [1] 1 0 1 1 1 1 1 1 1 1 1 1 1 0 0 1 1 1 0 1 1 1 1 1 1 1 1 1
## Show the no. of studies per correlation
pattern.na(df1$data, show.na = FALSE)

##      Beh Int PB
## Beh 29 29 29
## Int 29 29 29
## PB 29 29 29

```

```
## Show the total sample sizes per correlation
pattern.n(df1$data, df1$n)
```

```
##      Beh  Int  PB
## Beh 8136 8136 8136
## Int 8136 8136 8136
## PB   8136 8136 8136
```

Meta-analyzing the indirect and direct effects

Calculation of indirect and direct effects

```
## Calculate the indirect and direct effects and their sampling covariance matrices
## The variables are arranged as outcome, mediator, and predictor.
IE.df1 <- indirectEffect(df1$data, df1$n)
```

```
## Add behavior frequency to the data
IE.df1 <- data.frame(IE.df1, beh_freq_high=df1$beh_freq_high)
```

```
## Show the first few studies
head(IE.df1)
```

```
##      ind_eff  dir_eff    ind_var  ind_dir_cov    dir_var beh_freq_high
## 3  0.23971461 0.5585981 0.0012158465 -0.0010127721 0.001890922      1
## 19 0.20669102 0.7963369 0.0256034746 -0.0126145194 0.007139375      0
## 20 0.09709888 0.7230498 0.0054607198 -0.0033732871 0.007436056      1
## 22 0.14246754 0.4296310 0.0016172775 -0.0006532121 0.004438969      1
## 25 0.26640325 0.2926724 0.0064450809 -0.0065081243 0.010790892      1
## 26 0.19485689 0.3841558 0.0009136998 -0.0007736947 0.002370634      1
```

Meta-analysis of indirect and direct effects

```
## Random-effects model
IE0 <- meta(y=cbind(ind_eff, dir_eff),
            v=cbind(ind_var, ind_dir_cov, dir_var),
            data=IE.df1,
            model.name = "Random")
summary(IE0)
```

```
##
## Call:
## meta(y = cbind(ind_eff, dir_eff), v = cbind(ind_var, ind_dir_cov,
##      dir_var), data = IE.df1, model.name = "Random")
##
## 95% confidence intervals: z statistic approximation (robust=FALSE)
## Coefficients:
##      Estimate Std. Error    lbound    ubound z value  Pr(>|z|)
## Intercept1  0.1453716  0.0203468  0.1054926  0.1852507  7.1447 9.019e-13 ***
## Intercept2  0.4275448  0.0409088  0.3473651  0.5077246 10.4512 < 2.2e-16 ***
## Tau2_1_1    0.0095735  0.0030380  0.0036191  0.0155278  3.1513 0.0016257 **
## Tau2_2_1    0.0027014  0.0046064 -0.0063269  0.0117297  0.5865 0.5575693
## Tau2_2_2    0.0432142  0.0128722  0.0179851  0.0684433  3.3572 0.0007875 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```

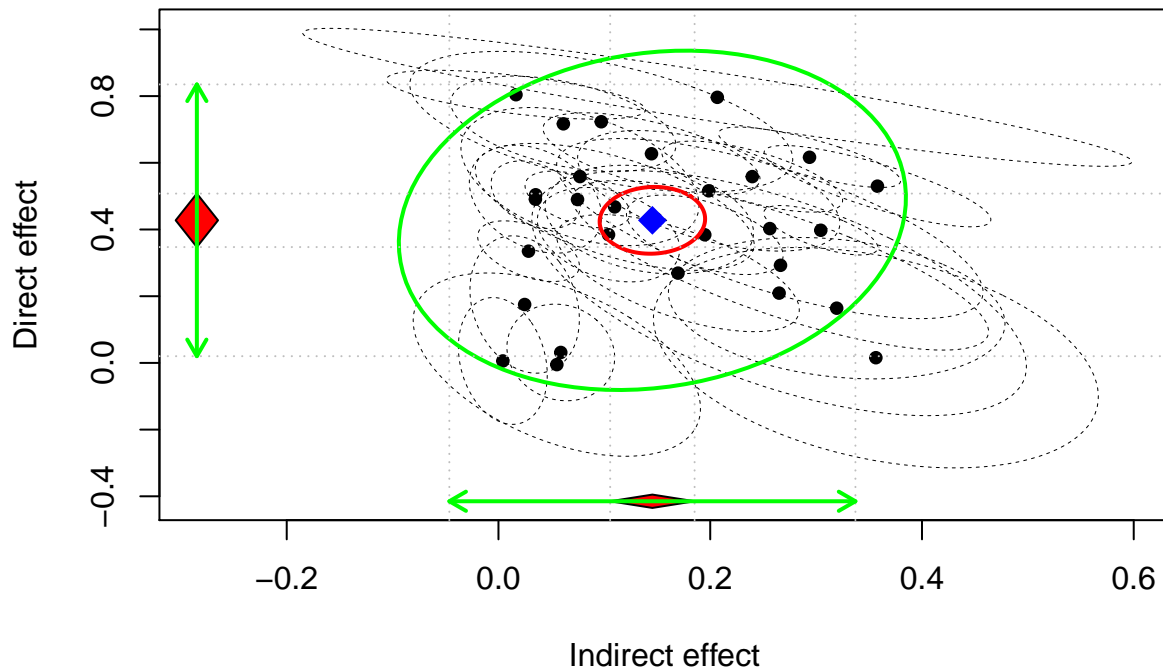
##
## Q statistic on the homogeneity of effect sizes: 1031.864
## Degrees of freedom of the Q statistic: 56
## P value of the Q statistic: 0
##
## Heterogeneity indices (based on the estimated Tau2):
##
##               Estimate
## Intercept1: I2 (Q statistic)  0.9044
## Intercept2: I2 (Q statistic)  0.9349
##
## Number of studies (or clusters): 29
## Number of observed statistics: 58
## Number of estimated parameters: 5
## Degrees of freedom: 53
## -2 log likelihood: -50.01425
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values may indicate problems.)
## Variance-covariance matrix of the random effects
VarCorr(IE0)

##           [,1]      [,2]
## [1,] 0.009573474 0.002701423
## [2,] 0.002701423 0.043214171
## Correlation matrix of the random effects
cov2cor(VarCorr(IE0))

##           [,1]      [,2]
## [1,] 1.0000000 0.1328143
## [2,] 0.1328143 1.0000000
## Plot the effect sizes
plot(IE0, axis.labels = c("Indirect effect", "Direct effect"))

```

Effect Sizes and their Confidence Ellipses



```
## Mixed-effects model with behavior frequency as a moderator
```

```
IE1 <- meta(y=cbind(ind_eff, dir_eff),
            v=cbind(ind_var, ind_dir_cov, dir_var),
            x=beh_freq_high,
            data=IE.df1,
            model.name = "Mixed")
summary(IE1)
```

```
##
## Call:
## meta(y = cbind(ind_eff, dir_eff), v = cbind(ind_var, ind_dir_cov,
##       dir_var), x = beh_freq_high, data = IE.df1, model.name = "Mixed")
##
## 95% confidence intervals: z statistic approximation (robust=FALSE)
## Coefficients:
##      Estimate Std. Error   lbound   ubound z value Pr(>|z|)
## Intercept1  0.1406059  0.0577021  0.0275118  0.2537000  2.4368 0.0148198 *
## Intercept2  0.4604307  0.1076665  0.2494083  0.6714532  4.2765 1.899e-05 ***
## Slope1_1    0.0053929  0.0614966 -0.1151382  0.1259240  0.0877 0.9301193
## Slope2_1   -0.0383525  0.1163143 -0.2663242  0.1896193 -0.3297 0.7416029
## Tau2_1_1    0.0095301  0.0030421  0.0035678  0.0154925  3.1328 0.0017315 **
## Tau2_2_1    0.0026812  0.0045911 -0.0063172  0.0116795  0.5840 0.5592234
## Tau2_2_2    0.0431205  0.0128488  0.0179373  0.0683038  3.3560 0.0007908 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Q statistic on the homogeneity of effect sizes: 1031.864
## Degrees of freedom of the Q statistic: 56
## P value of the Q statistic: 0
##
```



```
## Explained variances (R2):
##              y1      y2
## Tau2 (no predictor)    0.0095735 0.0432
## Tau2 (with predictors) 0.0095301 0.0431
## R2                    0.0045263 0.0022
##
## Number of studies (or clusters): 29
## Number of observed statistics: 58
## Number of estimated parameters: 7
## Degrees of freedom: 51
## -2 log likelihood: -50.13325
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values may indicate problems.)

## Test the statistical significance between the models
anova(IE1, IE0)

##      base comparison ep  minus2LL df      AIC      diffLL diffdf      p
## 1 Mixed      <NA>  7 -50.13325 51 -152.1332      NA      NA      NA
## 2 Mixed      Random  5 -50.01425 53 -156.0143 0.1189926      2 0.942239
```

TSSEM

Stage 1 analysis

```
## Random-effects model
random1 <- tssem1(df1$data, df1$n, method="REM")
summary(random1)

##
## Call:
## meta(y = ES, v = acovR, RE.constraints = Diag(paste0(RE.startvalues,
##      "*Tau2_", 1:no.es, "_", 1:no.es)), RE.lbound = RE.lbound,
##      I2 = I2, model.name = model.name, suppressWarnings = TRUE,
##      silent = silent, run = run)
##
## 95% confidence intervals: z statistic approximation (robust=FALSE)
## Coefficients:
##      Estimate Std. Error    lbound    ubound z value Pr(>|z|)
## Intercept1 0.4766330 0.0354367 0.4071784 0.5460876 13.4503 < 2.2e-16 ***
## Intercept2 0.5264034 0.0392050 0.4495630 0.6032437 13.4270 < 2.2e-16 ***
## Intercept3 0.5371427 0.0338542 0.4707897 0.6034956 15.8664 < 2.2e-16 ***
## Tau2_1_1    0.0327619 0.0094370 0.0142658 0.0512580  3.4717 0.0005173 ***
## Tau2_2_2    0.0414127 0.0116982 0.0184846 0.0643408  3.5401 0.0004000 ***
## Tau2_3_3    0.0302936 0.0085934 0.0134508 0.0471365  3.5252 0.0004231 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Q statistic on the homogeneity of effect sizes: 1199.938
## Degrees of freedom of the Q statistic: 84
## P value of the Q statistic: 0
##
## Heterogeneity indices (based on the estimated Tau2):
##      Estimate
## Intercept1: I2 (Q statistic) 0.9403
```

```
## Intercept2: I2 (Q statistic) 0.9590
## Intercept3: I2 (Q statistic) 0.9480
##
## Number of studies (or clusters): 29
## Number of observed statistics: 87
## Number of estimated parameters: 6
## Degrees of freedom: 81
## -2 log likelihood: -35.1454
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values may indicate problems.)
```

```
## Average correlation matrix under a random-effects model
averageR <- vec2symMat(coef(random1, select="fixed"), diag = FALSE)
dimnames(averageR) <- list(obs.vars1, obs.vars1)
averageR
```

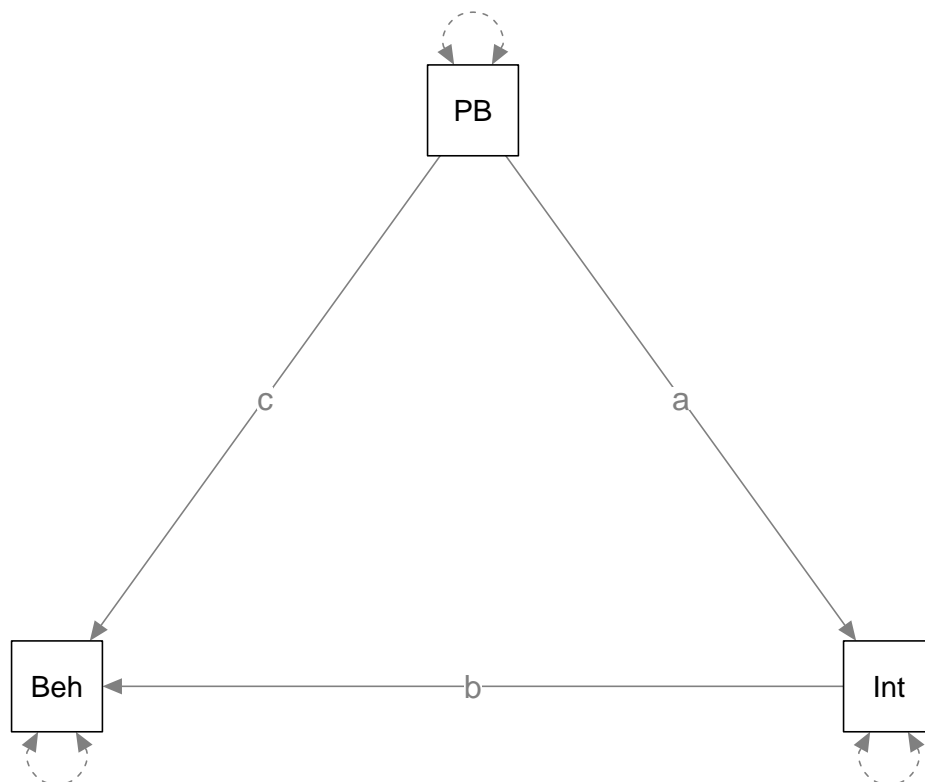
```
##           Beh           Int           PB
## Beh 1.0000000 0.4766330 0.5264034
## Int 0.4766330 1.0000000 0.5371427
## PB 0.5264034 0.5371427 1.0000000
```

```
## Heterogeneity variances of the random-effects
coef(random1, select="random")
```

```
##   Tau2_1_1   Tau2_2_2   Tau2_3_3
## 0.03276187 0.04141267 0.03029364
```

Stage 2 analysis

```
## Proposed model in lavaan syntax
model1 <- "Beh ~ c*PB + b*Int
           Int ~ a*PB
           PB ~~ 1*PB"
plot(model1)
```



Convert the lavaan syntax to RAM specification used in metaSEM

```
RAM1 <- lavaan2RAM(model1, obs.variables=obs.vars1)
```

```
RAM1
```

```
## $A
```

```
##      Beh Int  PB
## Beh "0" "0*b" "0*c"
## Int "0" "0"   "0*a"
## PB  "0" "0"   "0"
```

```
##
```

```
## $S
```

```
##      Beh          Int          PB
## Beh "0*BehWITHBeh" "0"          "0"
## Int "0"             "0*IntWITHInt" "0"
## PB  "0"             "0"          "1"
```

```
##
```

```
## $F
```

```
##      Beh Int PB
## Beh   1  0  0
## Int   0  1  0
## PB    0  0  1
```

```
##
```

```
## $M
```

```
##      Beh Int PB
## 1     0  0  0
```

Request the likelihood-based confidence interval

Indirect effect: ind = a*b

Direct effect: dir = c

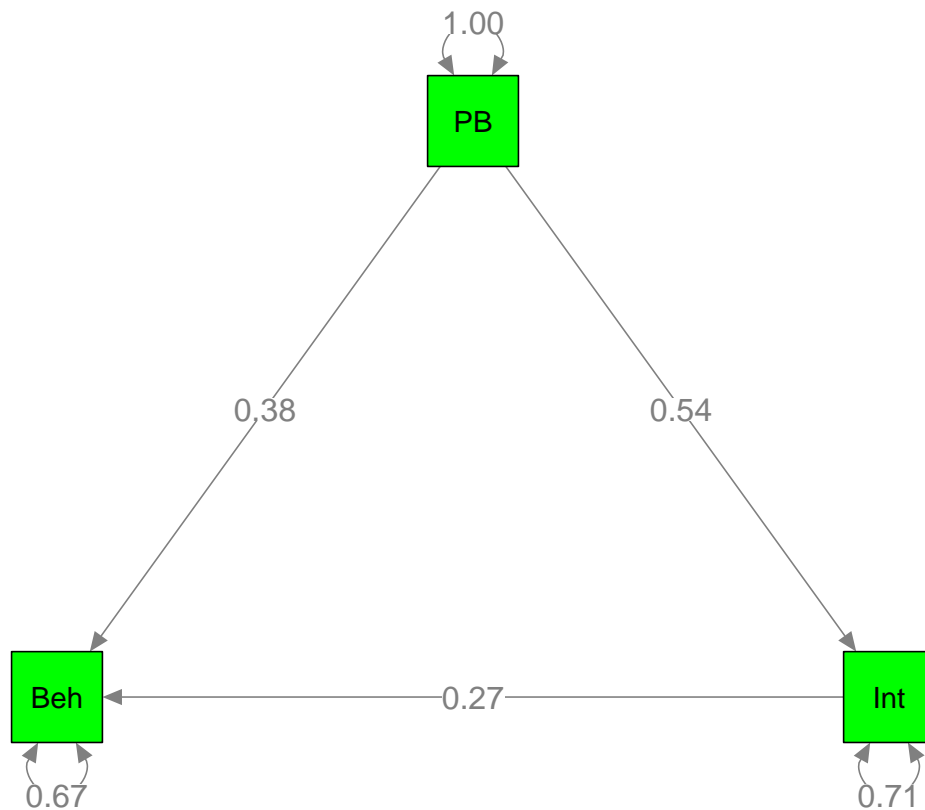
```
tssem.fit <- tssem2(random1, RAM=RAM1, intervals.type = "LB",
```

```

mx.algebras = list(ind=mxAlgebra(a*b, name="ind"),
                    dir=mxAlgebra(c, name="dir"))
summary(tssem.fit)

##
## Call:
## wls(Cov = pooledS, aCov = aCov, n = tssem1.obj$total.n, RAM = RAM,
##      Amatrix = Amatrix, Smatrix = Smatrix, Fmatrix = Fmatrix,
##      diag.constraints = diag.constraints, cor.analysis = cor.analysis,
##      intervals.type = intervals.type, mx.algebras = mx.algebras,
##      model.name = model.name, suppressWarnings = suppressWarnings,
##      silent = silent, run = run)
##
## 95% confidence intervals: Likelihood-based statistic
## Coefficients:
##      Estimate Std.Error  lbound  ubound z value Pr(>|z|)
## b  0.27250      NA 0.15566 0.38428      NA      NA
## c  0.38003      NA 0.26006 0.49868      NA      NA
## a  0.53714      NA 0.47079 0.60350      NA      NA
##
## mxAlgebras objects (and their 95% likelihood-based CIs):
##              lbound Estimate  ubound
## ind[1,1] 0.08584873 0.1463726 0.210334
## dir[1,1] 0.26006012 0.3800308 0.498680
##
## Goodness-of-fit indices:
##
##                                     Value
## Sample size                        8136.00
## Chi-square of target model          0.00
## DF of target model                  0.00
## p value of target model             0.00
## Number of constraints imposed on "Smatrix" 0.00
## DF manually adjusted                0.00
## Chi-square of independence model     574.38
## DF of independence model            3.00
## RMSEA                              0.00
## RMSEA lower 95% CI                  0.00
## RMSEA upper 95% CI                  0.00
## SRMR                              0.00
## TLI                                -Inf
## CFI                                1.00
## AIC                                0.00
## BIC                                0.00
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values indicate problems.)
plot(tssem.fit, color="green")

```



OSMASEM

Model without any moderator

```
## Convert the data format from TSSEM to OSMASEM
osmasem.df1 <- Cor2DataFrame(df1)
```

```
## Show the first few studies
head(osmasem.df1)
```

```
## $data
##      Int_Beh PB_Beh PB_Int C(Int_Beh Int_Beh) C(PB_Beh Int_Beh) C(PB_Int Int_Beh)
## 3      0.650  0.709  0.650      0.0013638687      0.0005774477      0.0005261467
## 19     0.840  0.870  0.940      0.0047735398      0.0020210664      0.0018415129
## 20     0.534  0.768  0.540      0.0137384784      0.0058167248      0.0052999616
## 22     0.550  0.540  0.320      0.0038580654      0.0016334632      0.0014883450
## 25     0.530  0.520  0.780      0.0029337385      0.0012421144      0.0011317638
## 26     0.530  0.540  0.540      0.0013638687      0.0005774478      0.0005261469
## 29     0.640  0.630  0.700      0.0004014810      0.0001699828      0.0001548815
## 30     0.589  0.651  0.539      0.0042351716      0.0017931278      0.0016338249
## 33     0.742  0.756  0.798      0.0010770132      0.0004559963      0.0004154852
## 34     0.732  0.780  0.752      0.0009450970      0.0004001443      0.0003645951
## 35     0.020  0.010  0.220      0.0032372293      0.0013706097      0.0012488439
## 36     0.300  0.090  0.200      0.0020708737      0.0008767859      0.0007988914
## 37     0.451  0.746  0.477      0.0066267975      0.0028057162      0.0025564536
## 44     0.270  0.530  0.340      0.0015432267      0.0006533860      0.0005953386
## 48     0.378  0.545  0.485      0.0009085126      0.0003846549      0.0003504817
## 49     0.220  0.360  0.490      0.0007581128      0.0003209771      0.0002924612
```

```

## 51 0.590 0.600 0.700 0.0051676843 0.0021879422 0.0019935633
## 52 0.400 0.610 0.410 0.0071300982 0.0030188095 0.0027506155
## 53 0.420 0.420 0.630 0.0020632898 0.0008735764 0.0007959672
## 54 0.720 0.460 0.470 0.0059292388 0.0025103771 0.0022873527
## 61 0.420 0.470 0.340 0.0023867701 0.0010105336 0.0009207567
## 67 0.440 0.550 0.590 0.0036815530 0.0015587303 0.0014202511
## 73 0.110 0.050 0.490 0.0054687150 0.0023153962 0.0021096948
## 74 0.570 0.710 0.610 0.0025034566 0.0010599375 0.0009657715
## 76 0.149 0.199 0.224 0.0040523589 0.0017157269 0.0015632999
## 77 0.261 0.517 0.315 0.0038580643 0.0016334630 0.0014883438
## 78 0.610 0.370 0.590 0.0104310703 0.0044164072 0.0040240482
## 80 0.640 0.810 0.780 0.0025034554 0.0010599362 0.0009657705
## 81 0.520 0.450 0.640 0.0090851269 0.0038465497 0.0035048180
## C(PB_Beh PB_Beh) C(PB_Int PB_Beh) C(PB_Int PB_Int) beh_freq_high
## 3 0.0011605001 0.0003951004 0.0010885801 1
## 19 0.0040617498 0.0013828506 0.0038100296 0
## 20 0.0116899132 0.0039799099 0.0109654496 1
## 22 0.0032827831 0.0011176451 0.0030793381 1
## 25 0.0024962842 0.0008498778 0.0023415812 1
## 26 0.0011605001 0.0003951005 0.0010885801 1
## 29 0.0003416155 0.0001163054 0.0003204445 1
## 30 0.0036036585 0.0012268911 0.0033803279 1
## 33 0.0009164181 0.0003120012 0.0008596246 1
## 34 0.0008041722 0.0002737861 0.0007543350 1
## 35 0.0027545212 0.0009377968 0.0025838142 1
## 36 0.0017620824 0.0005999130 0.0016528803 1
## 37 0.0056386644 0.0019197221 0.0052892179 1
## 44 0.0013131137 0.0004470588 0.0012317356 1
## 48 0.0007730429 0.0002631878 0.0007251348 0
## 49 0.0006450694 0.0002196184 0.0006050923 0
## 51 0.0043971230 0.0014970298 0.0041246184 1
## 52 0.0060669189 0.0020655256 0.0056909315 1
## 53 0.0017556295 0.0005977176 0.0016468276 1
## 54 0.0050451207 0.0017176458 0.0047324578 0
## 61 0.0020308752 0.0006914256 0.0019050151 1
## 67 0.0031325911 0.0010665118 0.0029384539 1
## 73 0.0046532665 0.0015842367 0.0043648882 1
## 74 0.0021301623 0.0007252287 0.0019981492 1
## 76 0.0034481054 0.0011739321 0.0032344146 1
## 77 0.0032827829 0.0011176448 0.0030793374 1
## 78 0.0088756774 0.0030217863 0.0083256210 1
## 80 0.0021301614 0.0007252278 0.0019981483 1
## 81 0.0077304301 0.0026318800 0.0072513489 1
##
## $n
## [1] 413 118 41 146 192 413 1403 133 523 596 174 272 85 365 620
## [16] 743 109 79 273 95 236 153 103 225 139 146 54 225 62
##
## $obslabels
## [1] "Beh" "Int" "PB"
##
## $ylabels
## [1] "Int_Beh" "PB_Beh" "PB_Int"
##

```

```

## $vlabels
## [1] "C(Int_Beh Int_Beh)" "C(PB_Beh Int_Beh)" "C(PB_Int Int_Beh)"
## [4] "C(PB_Beh PB_Beh)" "C(PB_Int PB_Beh)" "C(PB_Int PB_Int)"

## Fit a model without any moderator
osmasem.fit0 <- osmasem(model.name="No moderator", RAM=RAM1, data=osmasem.df1)
summary(osmasem.fit0)

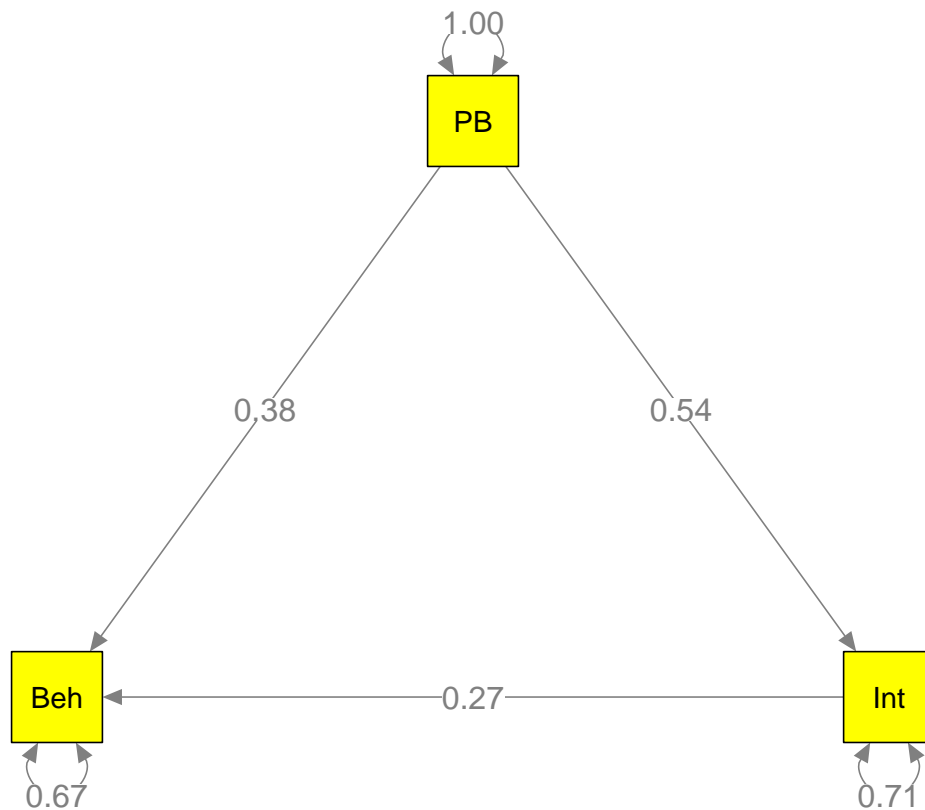
## Summary of No moderator
##
## free parameters:
##      name matrix row col Estimate Std.Error A z value Pr(>|z|)
## 1      b      A0 Beh Int 0.2725023 0.05783959 4.711345 2.460872e-06
## 2      c      A0 Beh PB 0.3800308 0.06042091 6.289722 3.180343e-10
## 3      a      A0 Int PB 0.5371427 0.03385418 15.866362 0.000000e+00
## 4 Tau1_1 vecTau1 1 1 -1.7092449 0.14402351 -11.867819 0.000000e+00
## 5 Tau1_2 vecTau1 2 1 -1.5920843 0.14123961 -11.272222 0.000000e+00
## 6 Tau1_3 vecTau1 3 1 -1.7484087 0.14183547 -12.327020 0.000000e+00
##
## Model Statistics:
##      | Parameters | Degrees of Freedom | Fit (-2lnL units)
##      Model:      6 81 -35.1454
##      Saturated: 9 78 NA
##      Independence: 6 81 NA
## Number of observations/statistics: 8136/87
##
## Information Criteria:
##      | df Penalty | Parameters Penalty | Sample-Size Adjusted
##      AIC: -197.1454 -23.14540 -23.1350671
##      BIC: -764.4738 18.87892 -0.1879251
## To get additional fit indices, see help(mxRefModels)
## timestamp: 2021-01-30 09:54:44
## Wall clock time: 0.07120895 secs
## optimizer: SLSQP
## OpenMx version number: 2.18.1
## Need help? See help(mxSummary)

## Get the heterogeneity of variances
VarCorr(osmasem.fit0)

##      Tau2_1      Tau2_2      Tau2_3
## Tau2_1 0.03276187 0.00000000 0.00000000
## Tau2_2 0.00000000 0.04141267 0.00000000
## Tau2_3 0.00000000 0.00000000 0.03029364

## Plot the fitted model
plot(osmasem.fit0, color="yellow")

```



Model with a moderator

```

## Create A1 to represent the moderator on the A matrix
A1 <- create.modMatrix(RAM1, output="A", "beh_freq_high")
A1

##      Beh Int          PB
## Beh "0" "0*data.beh_freq_high" "0*data.beh_freq_high"
## Int "0" "0"                "0*data.beh_freq_high"
## PB  "0" "0"                "0"

## Fit a model with female as a moderator
osmasem.fit1 <- osmasem(model.name="Female as a moderator", RAM=RAM1, Ax=A1, data=osmasem.df1)
summary(osmasem.fit1)

## Summary of Female as a moderator
##
## free parameters:
##      name  matrix row col   Estimate Std.Error A      z value      Pr(>|z|)
## 1      b      A0 Beh Int  0.30516661 0.17061592    1.78861744 7.367645e-02
## 2      c      A0 Beh PB   0.36918343 0.17839178    2.06950920 3.849833e-02
## 3      a      A0 Int PB   0.58544894 0.09000500    6.50462683 7.788659e-11
## 4     b_1     A1 Beh Int -0.03769750 0.18135154   -0.20786974 8.353307e-01
## 5     c_1     A1 Beh PB   0.01207899 0.18958937    0.06371133 9.492001e-01
## 6     a_1     A1 Int PB  -0.05626572 0.09708943   -0.57952474 5.622352e-01
## 7 Tau1_1 vecTau1 1 1 -1.70930736 0.14338787  -11.92086441 0.000000e+00
## 8 Tau1_2 vecTau1 2 1 -1.59196769 0.14115508  -11.27814632 0.000000e+00
## 9 Tau1_3 vecTau1 3 1 -1.75242337 0.14159263  -12.37651526 0.000000e+00
##

```



```
## Model Statistics:
##           | Parameters | Degrees of Freedom | Fit (-2lnL units)
##      Model:           9           78           -35.76057
##    Saturated:           9           78           NA
## Independence:           6           81           NA
## Number of observations/statistics: 8136/87
##
## Information Criteria:
##           | df Penalty | Parameters Penalty | Sample-Size Adjusted
## AIC:      -191.7606           -17.76057           -17.73841
## BIC:      -738.0768           45.27592           16.67565
## CFI: NA
## TLI: 1 (also known as NNFI)
## RMSEA: 0 [95% CI (NA, NA)]
## Prob(RMSEA <= 0.05): NA
## To get additional fit indices, see help(mxRefModels)
## timestamp: 2021-01-30 09:54:44
## Wall clock time: 0.1399055 secs
## optimizer: SLSQP
## OpenMx version number: 2.18.1
## Need help? See help(mxSummary)

## Test the statistical significance between the models
anova(osmasem.fit1, osmasem.fit0)

##           base comparison ep minus2LL df      AIC      diffLL diffdf
## 1 Female as a moderator      <NA>  9 -35.76057 78 -191.7606      NA      NA
## 2 Female as a moderator No moderator  6 -35.14540 81 -197.1454 0.6151653      3
##           p
## 1      NA
## 2 0.892952

## Get the R2 on the correlation coefficients
osmasemR2(osmasem.fit0, osmasem.fit1)

## $Tau2.0
##   Tau2_1_1  Tau2_2_2  Tau2_3_3
## 0.03275778 0.04142232 0.03005138
##
## $Tau2.1
##   Tau2_1_1  Tau2_2_2  Tau2_3_3
## 0.03276187 0.04141267 0.03029364
##
## $R2
##   Tau2_1_1  Tau2_2_2  Tau2_3_3
## 0.0000000000 0.0002331214 0.0000000000
```

Illustration 2 with two parallel mediators (Aut and Cap)

```
## Use new.df2 as the data file in illustration 2
df2 <- Hagger18

## Select Aut, Cap, Beh, and PB for the illustration
obs.vars2 <- c("Aut", "Cap", "Beh", "PB")
df2$data <- lapply(df2$data, function(x) x[obs.vars2, obs.vars2])
```

```
## Drop studies do not include all correlation in c("Aut", "Cap", "Beh", "PB")
index2 <- sapply(df2$data, function(x) any(is.na(vechs(x))))
df2 <- lapply(df2, function(x) x[!index2])
```

```
## Show the first few studies
head(df2)
```

```
## $data
## $data$`3`
##      Aut    Cap    Beh    PB
## Aut 1.000 0.380 0.264 0.264
## Cap 0.380 1.000 0.591 0.591
## Beh 0.264 0.591 1.000 0.709
## PB  0.264 0.591 0.709 1.000
##
## $data$`20`
##      Aut    Cap    Beh    PB
## Aut 1.000 0.513 -0.138 -0.244
## Cap 0.513 1.000 -0.249 0.057
## Beh -0.138 -0.249 1.000 0.768
## PB -0.244 0.057 0.768 1.000
##
## $data$`29`
##      Aut    Cap    Beh    PB
## Aut 1.00 0.42 0.28 0.25
## Cap 0.42 1.00 0.55 0.51
## Beh 0.28 0.55 1.00 0.63
## PB  0.25 0.51 0.63 1.00
##
## $data$`33`
##      Aut    Cap    Beh    PB
## Aut 1.000 0.697 0.266 0.297
## Cap 0.697 1.000 0.274 0.284
## Beh 0.266 0.274 1.000 0.756
## PB  0.297 0.284 0.756 1.000
##
## $data$`34`
##      Aut    Cap    Beh    PB
## Aut 1.000 0.797 0.527 0.562
## Cap 0.797 1.000 0.570 0.608
## Beh 0.527 0.570 1.000 0.780
## PB  0.562 0.608 0.780 1.000
##
## $data$`37`
##      Aut    Cap    Beh    PB
## Aut 1.000 0.241 0.114 0.101
## Cap 0.241 1.000 0.326 0.335
## Beh 0.114 0.326 1.000 0.746
## PB  0.101 0.335 0.746 1.000
##
## $data$`51`
##      Aut    Cap    Beh    PB
## Aut 1.00 0.18 -0.05 0.15
```

```

## Cap  0.18 1.00  0.37 0.44
## Beh -0.05 0.37  1.00 0.60
## PB   0.15 0.44  0.60 1.00
##
## $data$`52`
##      Aut  Cap  Beh  PB
## Aut  1.00 0.20 -0.19 0.03
## Cap  0.20 1.00  0.26 0.24
## Beh -0.19 0.26  1.00 0.61
## PB   0.03 0.24  0.61 1.00
##
## $data$`53`
##      Aut  Cap  Beh  PB
## Aut  1.00 -0.11 -0.12 -0.27
## Cap -0.11  1.00  0.42 0.46
## Beh -0.12  0.42  1.00 0.42
## PB  -0.27  0.46  0.42 1.00
##
## $data$`54`
##      Aut  Cap  Beh  PB
## Aut  1.00 0.32 0.17 0.07
## Cap  0.32 1.00 0.51 0.32
## Beh  0.17 0.51 1.00 0.46
## PB   0.07 0.32 0.46 1.00
##
## $data$`78`
##      Aut  Cap  Beh  PB
## Aut  1.00 0.55 -0.13 0.22
## Cap  0.55 1.00  0.40 0.43
## Beh -0.13 0.40  1.00 0.37
## PB   0.22 0.43  0.37 1.00
##
## $data$`80`
##      Aut  Cap  Beh  PB
## Aut  1.00 0.21 0.35 0.35
## Cap  0.21 1.00 0.53 0.64
## Beh  0.35 0.53 1.00 0.81
## PB   0.35 0.64 0.81 1.00
##
## $data$`81`
##      Aut  Cap  Beh  PB
## Aut  1.00 0.01 -0.30 0.51
## Cap  0.01 1.00  0.20 0.27
## Beh -0.30 0.20  1.00 0.45
## PB   0.51 0.27  0.45 1.00
##
##
## $n
## [1] 413 41 1403 523 596 85 109 79 273 95 54 225 62
##
## $beh_freq_high
## [1] 1 1 1 1 1 1 1 1 1 0 1 1 1

```

```
## Show the no. of studies per correlation
pattern.na(df2$data, show.na = FALSE)
```

```
##      Aut Cap Beh PB
## Aut   13   13   13 13
## Cap   13   13   13 13
## Beh   13   13   13 13
## PB    13   13   13 13
```

```
## Show the total sample sizes per correlation
pattern.n(df2$data, df2$n)
```

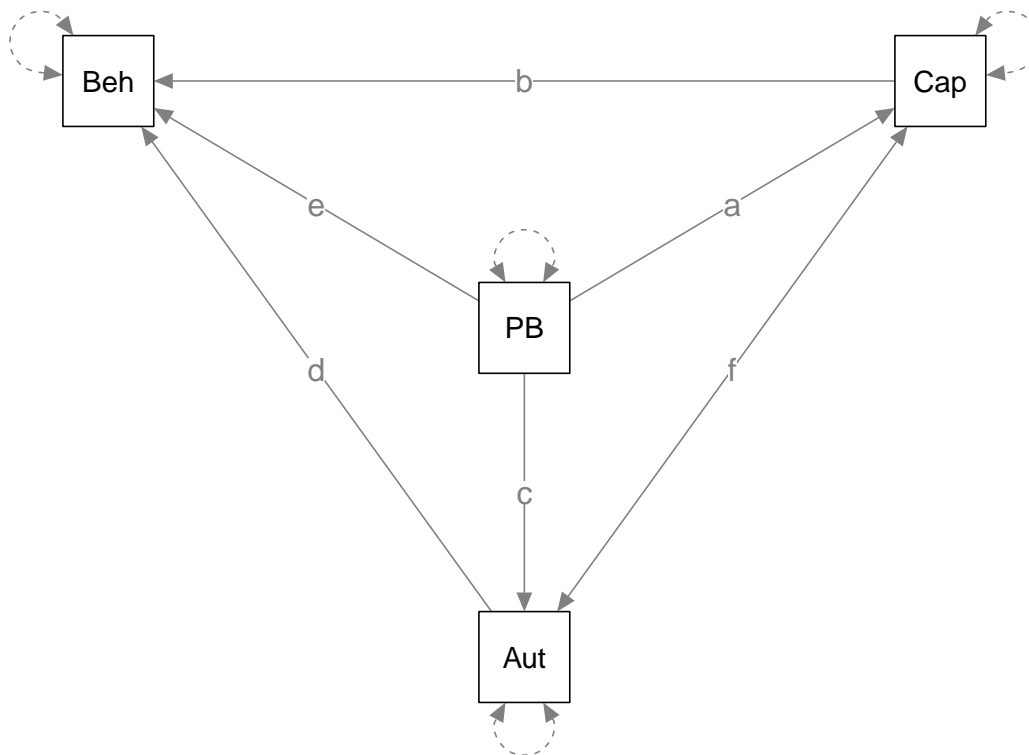
```
##      Aut Cap Beh PB
## Aut 3958 3958 3958 3958
## Cap 3958 3958 3958 3958
## Beh 3958 3958 3958 3958
## PB  3958 3958 3958 3958
```

Meta-analyzing the indirect and direct effects

Calculation of indirect and direct effects

```
## Calculate indirect and direct effects as effect sizes in each study
## PB -> Cap -> Beh
## PB -> Aut -> Beh
model2 <- "Cap ~ a*PB
          Aut ~ c*PB
          Beh ~ b*Cap + d*Aut + e*PB
          Cap ~~ f*Aut
          PB ~~ 1*PB
          ## Define indirect and direct effects
          Ind_Cap := a*b
          Ind_Aut := c*d
          Dir_PB := f"

## Display the proposed model
plot(model2, layout="circle")
```



```
## Estimate the indirect and direct effects
```

```
IE.df2 <- mapply(function(x, y) {calEffSizes(model=model2, n=y, Cov=x)},
  df2$data, df2$n, SIMPLIFY = FALSE)
```

```
head(IE.df2)
```

```
## $`3`
## $`3`$ES
##      Ind_Cap      Ind_Aut      Dir_PB
## 0.151994676 0.005454605 0.223975994
##
## $`3`$VCOV
##           Ind_Cap      Ind_Aut      Dir_PB
## Ind_Cap  7.440345e-04 -6.592824e-05 1.694948e-19
## Ind_Aut -6.592824e-05  9.026331e-05 2.263434e-19
## Dir_PB   2.515114e-19  2.454108e-19 1.587244e-03
##
##
## $`20`
## $`20`$ES
##      Ind_Cap      Ind_Aut      Dir_PB
## -0.02603626 -0.07525792  0.52690799
##
## $`20`$VCOV
##           Ind_Cap      Ind_Aut      Dir_PB
## Ind_Cap  5.101867e-03 -1.739814e-03 -3.599878e-17
## Ind_Aut -1.739814e-03  2.755167e-03  8.966078e-18
## Dir_PB  -3.154713e-17  7.401104e-18  2.963513e-02
##
##
```

```

## $`29`
## $`29`$ES
##      Ind_Cap      Ind_Aut      Dir_PB
## 0.149766899 0.009761141 0.292499971
##
## $`29`$VCOV
##      Ind_Cap      Ind_Aut      Dir_PB
## Ind_Cap  1.972890e-04 -2.082765e-05  1.404948e-19
## Ind_Aut -2.082765e-05  2.980877e-05 -1.487306e-20
## Dir_PB   1.833588e-19 -3.031061e-21  5.553902e-04
##
##
## $`33`
## $`33`$ES
##      Ind_Cap      Ind_Aut      Dir_PB
## 0.017583378 0.001152015 0.612652009
##
## $`33`$VCOV
##      Ind_Cap      Ind_Aut      Dir_PB
## Ind_Cap  1.357261e-04 -9.035574e-05  3.194362e-19
## Ind_Aut -9.035574e-05  1.422618e-04  4.153252e-20
## Dir_PB   2.896467e-19  6.576327e-20  2.320444e-03
##
##
## $`34`
## $`34`$ES
##      Ind_Cap      Ind_Aut      Dir_PB
## 0.06829862 0.03079963 0.45530400
##
## $`34`$VCOV
##      Ind_Cap      Ind_Aut      Dir_PB
## Ind_Cap  7.257764e-04 -4.335468e-04  5.637706e-19
## Ind_Aut -4.335468e-04  5.642697e-04  3.303910e-19
## Dir_PB   5.141592e-19  1.587066e-19  1.071392e-03
##
##
## $`37`
## $`37`$ES
##      Ind_Cap      Ind_Aut      Dir_PB
## 0.026977135 0.002241968 0.207164996
##
## $`37`$VCOV
##      Ind_Cap      Ind_Aut      Dir_PB
## Ind_Cap  7.502225e-04 -3.871017e-05  8.330654e-19
## Ind_Aut -3.871017e-05  6.138024e-05  3.590069e-19
## Dir_PB   7.567297e-19  2.890898e-19  1.084278e-02
##
## Rename the variances and covariances of the effect sizes from Cov1 to Cov6 for ease of reference
IE.df2 <- t(sapply(IE.df2,
  function(x) { acov <- vech(x$VCOV)
                 names(acov) <- paste0("Cov", 1:6)
                 c(x$ES, acov)} ))
## Show the first few studies

```

```
head(IE.df2)
```

```
##           Ind_Cap      Ind_Aut   Dir_PB      Cov1      Cov2      Cov3
## 3    0.15199468  0.005454605 0.223976 0.0007440345 -6.592824e-05 2.515114e-19
## 20 -0.02603626 -0.075257923 0.526908 0.0051018674 -1.739814e-03 -3.154713e-17
## 29  0.14976690  0.009761141 0.292500 0.0001972890 -2.082765e-05 1.833588e-19
## 33  0.01758338  0.001152015 0.612652 0.0001357261 -9.035574e-05 2.896467e-19
## 34  0.06829862  0.030799631 0.455304 0.0007257764 -4.335468e-04 5.141592e-19
## 37  0.02697713  0.002241968 0.207165 0.0007502225 -3.871017e-05 7.567297e-19
##           Cov4      Cov5      Cov6
## 3    9.026331e-05 2.454108e-19 0.0015872439
## 20  2.755167e-03 7.401104e-18 0.0296351332
## 29  2.980877e-05 -3.031061e-21 0.0005553902
## 33  1.422618e-04 6.576327e-20 0.0023204437
## 34  5.642697e-04 1.587066e-19 0.0010713924
## 37  6.138024e-05 2.890898e-19 0.0108427780
```

Meta-analysis of indirect and direct effects

```
## Random-effects model with independent random effects
## Tau2_2_2 is close to negative. It is fixed at 0.
IE2 <- meta(y=IE.df2[, c("Ind_Cap", "Ind_Aut", "Dir_PB")],
            v=IE.df2[, paste0("Cov", 1:6)],
            RE.constraints = Diag(c("0.01*Tau2_1_1", "0", "0.01*Tau2_3_3")))
summary(IE2)
```

```
##
## Call:
## meta(y = IE.df2[, c("Ind_Cap", "Ind_Aut", "Dir_PB")], v = IE.df2[,
##   paste0("Cov", 1:6)], RE.constraints = Diag(c("0.01*Tau2_1_1",
##   "0", "0.01*Tau2_3_3")))
##
## 95% confidence intervals: z statistic approximation (robust=FALSE)
## Coefficients:
##           Estimate   Std.Error    lbound    ubound  z value  Pr(>|z|)
## Intercept1 6.9096e-02 1.7891e-02 3.4030e-02 1.0416e-01 3.8620 0.0001125
## Intercept2 4.6650e-03 3.0421e-03 -1.2975e-03 1.0627e-02 1.5335 0.1251644
## Intercept3 2.4447e-01 5.9063e-02 1.2871e-01 3.6023e-01 4.1391 3.487e-05
## Tau2_1_1    3.0429e-03 1.5942e-03 -8.1684e-05 6.1674e-03 1.9087 0.0562976
## Tau2_3_3    3.8025e-02 1.7341e-02 4.0376e-03 7.2012e-02 2.1928 0.0283215
##
## Intercept1 ***
## Intercept2
## Intercept3 ***
## Tau2_1_1 .
## Tau2_3_3 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Q statistic on the homogeneity of effect sizes: 305.2991
## Degrees of freedom of the Q statistic: 36
## P value of the Q statistic: 0
##
## Heterogeneity indices (based on the estimated Tau2):
```

```
##                                Estimate
## Intercept1: I2 (Q statistic)    0.8273
## Intercept2: I2 (Q statistic)    0.0000
## Intercept3: I2 (Q statistic)    0.9302
##
## Number of studies (or clusters): 13
## Number of observed statistics: 39
## Number of estimated parameters: 5
## Degrees of freedom: 34
## -2 log likelihood: -88.50845
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values may indicate problems.)
```

TSSEM

Stage 1 analysis

```
## Random-effects model
random1 <- tssem1(df2$data, df2$n)
summary(random1)

##
## Call:
## meta(y = ES, v = acovR, RE.constraints = Diag(paste0(RE.startvalues,
##      "*Tau2_", 1:no.es, "_", 1:no.es)), RE.lbound = RE.lbound,
##      I2 = I2, model.name = model.name, suppressWarnings = TRUE,
##      silent = silent, run = run)
##
## 95% confidence intervals: z statistic approximation (robust=FALSE)
## Coefficients:
##              Estimate      Std.Error      lbound      ubound z value  Pr(>|z|)
## Intercept1  0.35451117  0.07004356  0.21722831  0.49179402  5.0613 4.164e-07
## Intercept2  0.11727541  0.06190972 -0.00406542  0.23861624  1.8943 0.0581856
## Intercept3  0.20814639  0.06281680  0.08502771  0.33126506  3.3135 0.0009212
## Intercept4  0.40319809  0.04605582  0.31293033  0.49346584  8.7546 < 2.2e-16
## Intercept5  0.43477744  0.03722533  0.36181714  0.50773775 11.6796 < 2.2e-16
## Intercept6  0.63769889  0.04056597  0.55819105  0.71720674 15.7200 < 2.2e-16
## Tau2_1_1    0.05827991  0.02439390  0.01046875  0.10609107  2.3891 0.0168889
## Tau2_2_2    0.04228562  0.01895276  0.00513888  0.07943235  2.2311 0.0256741
## Tau2_3_3    0.04418457  0.01972983  0.00551482  0.08285432  2.2395 0.0251246
## Tau2_4_4    0.02232387  0.01159256 -0.00039713  0.04504487  1.9257 0.0541410
## Tau2_5_5    0.01309635  0.00660328  0.00015415  0.02603855  1.9833 0.0473329
## Tau2_6_6    0.01879292  0.00857040  0.00199524  0.03559060  2.1928 0.0283240
##
## Intercept1 ***
## Intercept2 .
## Intercept3 ***
## Intercept4 ***
## Intercept5 ***
## Intercept6 ***
## Tau2_1_1 *
## Tau2_2_2 *
## Tau2_3_3 *
## Tau2_4_4 .
```



```
## Tau2_5_5      *
## Tau2_6_6      *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Q statistic on the homogeneity of effect sizes: 889.6961
## Degrees of freedom of the Q statistic: 72
## P value of the Q statistic: 0
##
## Heterogeneity indices (based on the estimated Tau2):
##                               Estimate
## Intercept1: I2 (Q statistic)  0.9600
## Intercept2: I2 (Q statistic)  0.9278
## Intercept3: I2 (Q statistic)  0.9315
## Intercept4: I2 (Q statistic)  0.9096
## Intercept5: I2 (Q statistic)  0.8563
## Intercept6: I2 (Q statistic)  0.9420
##
## Number of studies (or clusters): 13
## Number of observed statistics: 78
## Number of estimated parameters: 12
## Degrees of freedom: 66
## -2 log likelihood: -34.37846
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values may indicate problems.)
```

```
## Average correlation matrix under a random-effects model
averagerR <- vec2symMat(coef(random1, select="fixed"), diag = FALSE)
dimnames(averagerR) <- list(obs.vars2, obs.vars2)
averagerR
```

```
##           Aut      Cap      Beh      PB
## Aut 1.0000000 0.3545112 0.1172754 0.2081464
## Cap 0.3545112 1.0000000 0.4031981 0.4347774
## Beh 0.1172754 0.4031981 1.0000000 0.6376989
## PB  0.2081464 0.4347774 0.6376989 1.0000000
```

```
## Heterogeneity variances of the random-effects
coef(random1, select="random")
```

```
##   Tau2_1_1  Tau2_2_2  Tau2_3_3  Tau2_4_4  Tau2_5_5  Tau2_6_6
## 0.05827991 0.04228562 0.04418457 0.02232387 0.01309635 0.01879292
```

Stage 2 analysis

```
RAM2 <- lavaan2RAM(model2, obs.variables=obs.vars2)
RAM2
```

```
## $A
##      Aut  Cap  Beh PB
## Aut "0"  "0"  "0" "0*c"
## Cap "0"  "0"  "0" "0*a"
## Beh "0*d" "0*b" "0" "0*e"
## PB  "0"  "0"  "0" "0"
##
## $S
```

```

##      Aut      Cap      Beh      PB
## Aut "0*AutWITHAut" "0*f"      "0"      "0"
## Cap "0*f"          "0*CapWITHCap" "0"      "0"
## Beh "0"            "0"          "0*BehWITHBeh" "0"
## PB  "0"            "0"          "0"          "1"
##
## $F
##      Aut Cap Beh PB
## Aut   1   0   0   0
## Cap   0   1   0   0
## Beh   0   0   1   0
## PB    0   0   0   1
##
## $M
##      Aut Cap Beh PB
## 1     0   0   0   0

```

```

tssem.fit <- tssem2(random1, RAM=RAM2, intervals.type = "LB",
                    mx.algebras = list(Ind_Cap=mxAlgebra(a*b, name="Ind_Cap"),
                                       Ind_Aut=mxAlgebra(c*d, name="Ind_Aut"),
                                       Dir_PB=mxAlgebra(f, name="Dir_PB")))
summary(tssem.fit)

```

```

##
## Call:
## wls(Cov = pooledS, aCov = aCov, n = tssem1.obj$total.n, RAM = RAM,
##      Amatrix = Amatrix, Smatrix = Smatrix, Fmatrix = Fmatrix,
##      diag.constraints = diag.constraints, cor.analysis = cor.analysis,
##      intervals.type = intervals.type, mx.algebras = mx.algebras,
##      model.name = model.name, suppressWarnings = suppressWarnings,
##      silent = silent, run = run)
##
## 95% confidence intervals: Likelihood-based statistic
## Coefficients:
##      Estimate Std. Error    lbound    ubound z value Pr(>|z|)
## c  0.208146      NA  0.085023  0.331265      NA      NA
## d -0.064842      NA -0.232386  0.092916      NA      NA
## b  0.176407      NA  0.032724  0.319057      NA      NA
## e  0.574498      NA  0.466478  0.684753      NA      NA
## a  0.434777      NA  0.361817  0.507737      NA      NA
## f  0.264014      NA  0.117668  0.409699      NA      NA
##
## mxAlgebras objects (and their 95% likelihood-based CIs):
##              lbound      Estimate      ubound
## Ind_Cap[1,1]  0.01498672  0.07669765  0.13854490
## Ind_Aut[1,1] -0.06639614 -0.01349671  0.01630176
## Dir_PB[1,1]   0.11766835  0.26401381  0.40969865
##
## Goodness-of-fit indices:
##
##              Value
## Sample size      3958.00
## Chi-square of target model      0.00
## DF of target model      0.00
## p value of target model      0.00
## Number of constraints imposed on "Smatrix"      0.00

```

```
## DF manually adjusted 0.00
## Chi-square of independence model 433.99
## DF of independence model 6.00
## RMSEA 0.00
## RMSEA lower 95% CI 0.00
## RMSEA upper 95% CI 0.00
## SRMR 0.00
## TLI -Inf
## CFI 1.00
## AIC 0.00
## BIC 0.00
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values indicate problems.)
```

```
plot(tssem.fit, layout="circle", color="green")
```

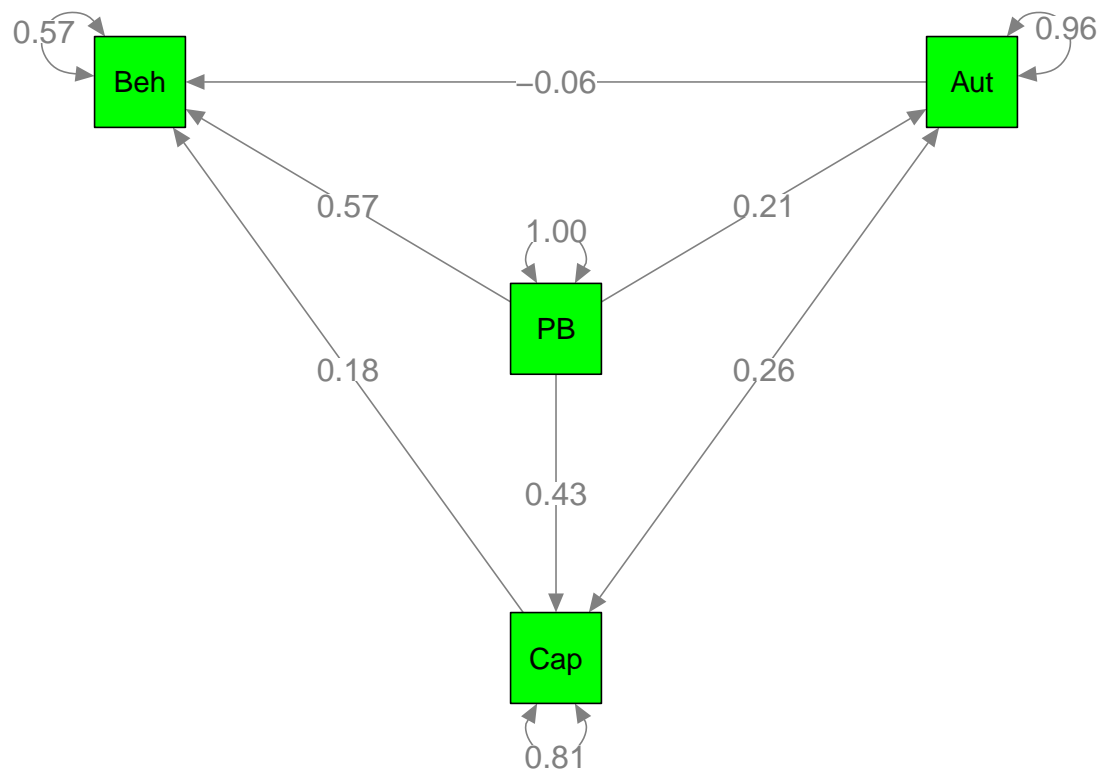


Illustration 3 with two serial mediators (Aut and Cap)

```
## Use df3 as the data file in illustration 3
df3 <- Hagger18

## Select Cap, Int, Beh, and PB for the illustration
obs.vars3 <- c("Cap", "Int", "Beh", "PB")
df3$data <- lapply(df3$data, function(x) x[obs.vars3, obs.vars3])

## Drop studies do not include all correlations in c("Cap", "Int", "Beh", "PB")
index3 <- sapply(df3$data, function(x) any(is.na(vechs(x))))
df3 <- lapply(df3, function(x) x[!index3])
```

```
## Show the first few studies
head(df3)
```

```
## $data
## $data$`3`
##      Cap    Int    Beh    PB
## Cap 1.000 0.702 0.591 0.591
## Int 0.702 1.000 0.650 0.650
## Beh 0.591 0.650 1.000 0.709
## PB  0.591 0.650 0.709 1.000
##
## $data$`20`
##      Cap    Int    Beh    PB
## Cap  1.000 0.376 -0.249 0.057
## Int  0.376 1.000  0.534 0.540
## Beh -0.249 0.534  1.000 0.768
## PB   0.057 0.540  0.768 1.000
##
## $data$`29`
##      Cap    Int    Beh    PB
## Cap 1.00 0.59 0.55 0.51
## Int 0.59 1.00 0.64 0.70
## Beh 0.55 0.64 1.00 0.63
## PB  0.51 0.70 0.63 1.00
##
## $data$`30`
##      Cap    Int    Beh    PB
## Cap 1.000 0.483 0.299 0.310
## Int 0.483 1.000 0.589 0.539
## Beh 0.299 0.589 1.000 0.651
## PB  0.310 0.539 0.651 1.000
##
## $data$`33`
##      Cap    Int    Beh    PB
## Cap 1.000 0.386 0.274 0.284
## Int 0.386 1.000 0.742 0.798
## Beh 0.274 0.742 1.000 0.756
## PB  0.284 0.798 0.756 1.000
##
## $data$`34`
##      Cap    Int    Beh    PB
## Cap 1.000 0.742 0.570 0.608
## Int 0.742 1.000 0.732 0.752
## Beh 0.570 0.732 1.000 0.780
## PB  0.608 0.752 0.780 1.000
##
## $data$`37`
##      Cap    Int    Beh    PB
## Cap 1.000 0.446 0.326 0.335
## Int 0.446 1.000 0.451 0.477
## Beh 0.326 0.451 1.000 0.746
## PB  0.335 0.477 0.746 1.000
##
## $data$`44`
```

```

##      Cap  Int  Beh  PB
## Cap 1.00 0.55 0.25 0.31
## Int 0.55 1.00 0.27 0.34
## Beh 0.25 0.27 1.00 0.53
## PB  0.31 0.34 0.53 1.00
##
## $data$`48`
##      Cap  Int  Beh  PB
## Cap 1.000 0.644 0.486 0.590
## Int 0.644 1.000 0.378 0.485
## Beh 0.486 0.378 1.000 0.545
## PB  0.590 0.485 0.545 1.000
##
## $data$`49`
##      Cap  Int  Beh  PB
## Cap 1.00 0.52 0.17 0.49
## Int 0.52 1.00 0.22 0.49
## Beh 0.17 0.22 1.00 0.36
## PB  0.49 0.49 0.36 1.00
##
## $data$`51`
##      Cap  Int  Beh  PB
## Cap 1.00 0.64 0.37 0.44
## Int 0.64 1.00 0.59 0.70
## Beh 0.37 0.59 1.00 0.60
## PB  0.44 0.70 0.60 1.00
##
## $data$`52`
##      Cap  Int  Beh  PB
## Cap 1.00 0.29 0.26 0.24
## Int 0.29 1.00 0.40 0.41
## Beh 0.26 0.40 1.00 0.61
## PB  0.24 0.41 0.61 1.00
##
## $data$`53`
##      Cap  Int  Beh  PB
## Cap 1.00 0.70 0.42 0.46
## Int 0.70 1.00 0.42 0.63
## Beh 0.42 0.42 1.00 0.42
## PB  0.46 0.63 0.42 1.00
##
## $data$`54`
##      Cap  Int  Beh  PB
## Cap 1.00 0.80 0.51 0.32
## Int 0.80 1.00 0.72 0.47
## Beh 0.51 0.72 1.00 0.46
## PB  0.32 0.47 0.46 1.00
##
## $data$`73`
##      Cap  Int  Beh  PB
## Cap 1.00 0.15 0.53 -0.13
## Int 0.15 1.00 0.11 0.49
## Beh 0.53 0.11 1.00 0.05
## PB -0.13 0.49 0.05 1.00

```

```

##
## $data$`76`
##      Cap   Int   Beh   PB
## Cap 1.000 0.567 0.145 0.187
## Int 0.567 1.000 0.149 0.224
## Beh 0.145 0.149 1.000 0.199
## PB  0.187 0.224 0.199 1.000
##
## $data$`77`
##      Cap   Int   Beh   PB
## Cap 1.000 0.667 0.227 0.277
## Int 0.667 1.000 0.261 0.315
## Beh 0.227 0.261 1.000 0.517
## PB  0.277 0.315 0.517 1.000
##
## $data$`78`
##      Cap   Int   Beh   PB
## Cap 1.00 0.49 0.40 0.43
## Int 0.49 1.00 0.61 0.59
## Beh 0.40 0.61 1.00 0.37
## PB  0.43 0.59 0.37 1.00
##
## $data$`80`
##      Cap   Int   Beh   PB
## Cap 1.00 0.57 0.53 0.64
## Int 0.57 1.00 0.64 0.78
## Beh 0.53 0.64 1.00 0.81
## PB  0.64 0.78 0.81 1.00
##
## $data$`81`
##      Cap   Int   Beh   PB
## Cap 1.00 0.35 0.20 0.27
## Int 0.35 1.00 0.52 0.64
## Beh 0.20 0.52 1.00 0.45
## PB  0.27 0.64 0.45 1.00
##
##
## $n
## [1] 413 41 1403 133 523 596 85 365 620 743 109 79 273 95 103
## [16] 139 146 54 225 62
##
## $beh_freq_high
## [1] 1 1 1 1 1 1 1 1 0 0 1 1 1 0 1 1 1 1 1
##
## Show the no. of studies per correlation
pattern.n(df3$data, show.na = FALSE)

##      Cap Int Beh PB
## Cap 20 20 20 20
## Int 20 20 20 20
## Beh 20 20 20 20
## PB 20 20 20 20
##
## Show the total sample sizes per correlation
pattern.n(df3$data, df3$n)

```

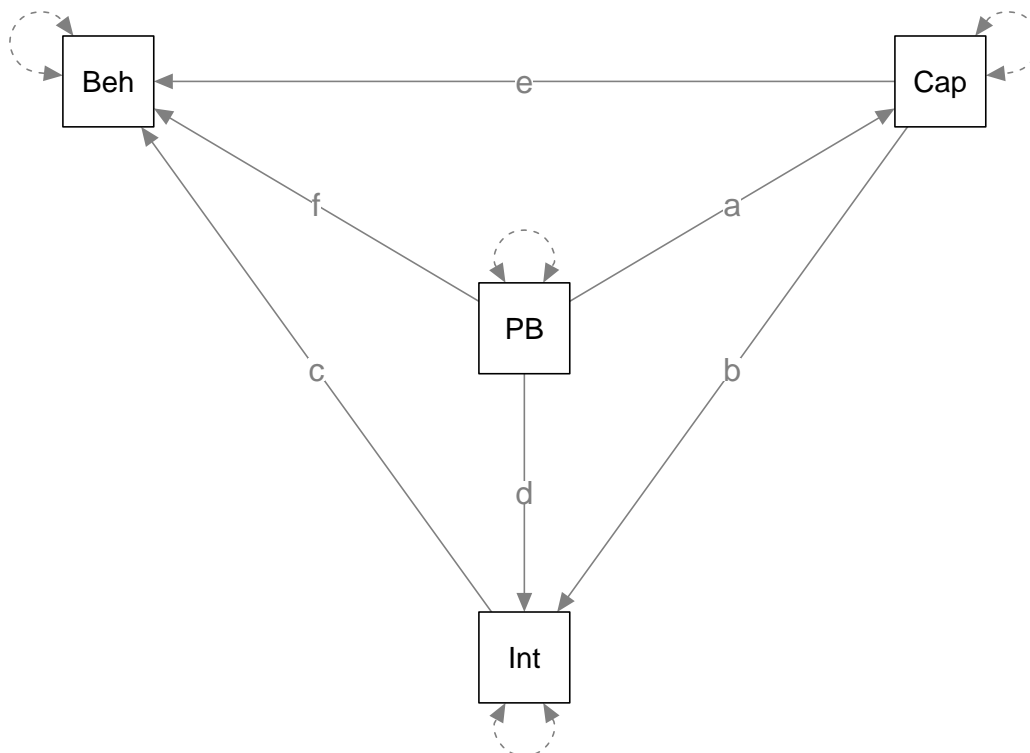
```
##      Cap  Int  Beh  PB
## Cap 6207 6207 6207 6207
## Int 6207 6207 6207 6207
## Beh 6207 6207 6207 6207
## PB   6207 6207 6207 6207
```

Meta-analyzing the indirect and direct effects

Calculation of indirect and direct effects

```
## Calculate indirect and direct effects as effect sizes in each study
## PB -> Cap -> Int -> Beh
model3 <- "Cap ~ a*PB
          Int ~ b*Cap + d*PB
          Beh ~ e*Cap + c*Int + f*PB
          PB ~~ 1*PB
          ## Define indirect and direct effects
          Ind_CapInt := a*b*c
          Ind_Cap := a*e
          Ind_Int := d*c
          Dir_PB := f"

## Display the proposed model
plot(model3, layout="circle")
```



```
## Estimate the indirect and direct effects
IE.df3 <- mapply(function(x, y) {calEffSizes(model=model3, n=y, Cov=x)},
                 df3$data, df3$n, SIMPLIFY = FALSE)

## Rename the variances and covariances of the effect sizes from Cov1 to Cov10 for ease of reference
```

```
IE.df3 <- t(sapply(IE.df3,
  function(x) { acov <- vech(x$VCOV)
    names(acov) <- paste0("Cov", 1:10)
    c(x$ES, acov)} ))

## Show the first few studies
head(IE.df3)
```

	Ind_CapInt	Ind_Cap	Ind_Int	Dir_PB	Cov1	Cov2
## 3	0.071885136	0.0843125034	0.08997364	0.4628287	2.607644e-04	-1.760859e-04
## 20	0.007398725	-0.0241413532	0.19498075	0.5897619	4.200800e-04	-1.340406e-03
## 29	0.046706843	0.1109320176	0.15686866	0.3154925	3.453848e-05	-1.035705e-05
## 30	0.036920098	-0.0035879815	0.14675358	0.4709143	2.251086e-04	-8.246915e-05
## 33	0.018842373	-0.0005248949	0.28657736	0.4511052	2.116888e-05	-5.401043e-06
## 34	0.091772694	0.0005940434	0.15946469	0.5281686	1.996415e-04	-1.418000e-04

	Cov3	Cov4	Cov5	Cov6	Cov7
## 3	2.205859e-04	-2.546141e-04	7.887380e-04	-2.546141e-04	-2.986314e-04
## 20	9.615378e-05	-1.027291e-04	4.383998e-03	-1.027291e-04	8.353152e-05
## 29	6.081536e-05	-7.006819e-05	1.699606e-04	-7.006819e-05	-5.407892e-05
## 30	2.349596e-04	-2.905685e-04	4.706487e-04	-2.905685e-04	-1.060688e-04
## 33	7.062148e-05	-7.886719e-05	6.817815e-05	-7.886719e-05	1.588054e-05
## 34	2.130739e-04	-2.466634e-04	4.793815e-04	-2.466634e-04	-9.076113e-05

	Cov8	Cov9	Cov10
## 3	0.0004161066	-0.0003186828	0.001902638
## 20	0.0047310537	-0.0027072504	0.008616020
## 29	0.0002709037	-0.0002353296	0.000712635
## 30	0.0017336374	-0.0011549800	0.005282320
## 33	0.0013017396	-0.0011995066	0.001980436
## 34	0.0005194685	-0.0004286035	0.001341391

Meta-analysis of indirect and direct effects

```
## Random-effects model with independent random effects
IE3 <- meta(y=IE.df3[, c("Ind_CapInt", "Ind_Cap", "Ind_Int", "Dir_PB")],
  v=IE.df3[, paste0("Cov", 1:10)],
  RE.constraints = Diag(paste0("0.01*Tau2_", 1:4, "_", 1:4)))
summary(IE3)
```

```
##
## Call:
## meta(y = IE.df3[, c("Ind_CapInt", "Ind_Cap", "Ind_Int", "Dir_PB")],
##       v = IE.df3[, paste0("Cov", 1:10)], RE.constraints = Diag(paste0("0.01*Tau2_",
##       1:4, "_", 1:4)))
##
## 95% confidence intervals: z statistic approximation (robust=FALSE)
## Coefficients:
```

	Estimate	Std.Error	lbound	ubound	z value	Pr(> z)
## Intercept1	2.4290e-02	5.7142e-03	1.3090e-02	3.5489e-02	4.2507	2.131e-05 ***
## Intercept2	2.7997e-02	1.0805e-02	6.8197e-03	4.9174e-02	2.5911	0.009566 **
## Intercept3	8.6211e-02	1.8510e-02	4.9931e-02	1.2249e-01	4.6574	3.202e-06 ***
## Intercept4	4.1141e-01	3.8419e-02	3.3611e-01	4.8671e-01	10.7084	< 2.2e-16 ***
## Tau2_1_1	4.4162e-04	1.9985e-04	4.9931e-05	8.3332e-04	2.2098	0.027118 *
## Tau2_2_2	1.5262e-03	7.0018e-04	1.5388e-04	2.8985e-03	2.1797	0.029276 *
## Tau2_3_3	5.2568e-03	2.1140e-03	1.1134e-03	9.4001e-03	2.4866	0.012895 *


```
## Tau2_4_4    2.3981e-02 9.6504e-03 5.0661e-03 4.2895e-02  2.4849  0.012958 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Q statistic on the homogeneity of effect sizes: 750.4992
## Degrees of freedom of the Q statistic: 76
## P value of the Q statistic: 0
##
## Heterogeneity indices (based on the estimated Tau2):
##                                Estimate
## Intercept1: I2 (Q statistic)    0.8653
## Intercept2: I2 (Q statistic)    0.7551
## Intercept3: I2 (Q statistic)    0.9305
## Intercept4: I2 (Q statistic)    0.8825
##
## Number of studies (or clusters): 20
## Number of observed statistics: 80
## Number of estimated parameters: 8
## Degrees of freedom: 72
## -2 log likelihood: -201.1973
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values may indicate problems.)
```

TSSEM

Stage 1 analysis

```
## Random-effects model
random1 <- tssem1(df3$data, df3$n)
summary(random1)

##
## Call:
## meta(y = ES, v = acovR, RE.constraints = Diag(paste0(RE.startvalues,
##      "*Tau2_", 1:no.es, "_", 1:no.es)), RE.lbound = RE.lbound,
##      I2 = I2, model.name = model.name, suppressWarnings = TRUE,
##      silent = silent, run = run)
##
## 95% confidence intervals: z statistic approximation (robust=FALSE)
## Coefficients:
##              Estimate Std.Error    lbound    ubound z value  Pr(>|z|)
## Intercept1 0.5493203 0.0324270 0.4857646 0.6128761 16.9402 < 2.2e-16 ***
## Intercept2 0.3646155 0.0384858 0.2891847 0.4400463  9.4740 < 2.2e-16 ***
## Intercept3 0.3833905 0.0370397 0.3107939 0.4559870 10.3508 < 2.2e-16 ***
## Intercept4 0.4911759 0.0411916 0.4104419 0.5719099 11.9242 < 2.2e-16 ***
## Intercept5 0.5612951 0.0335067 0.4956233 0.6269670 16.7517 < 2.2e-16 ***
## Intercept6 0.5560411 0.0422023 0.4733262 0.6387560 13.1756 < 2.2e-16 ***
## Tau2_1_1    0.0179174 0.0067110 0.0047640 0.0310708  2.6698 0.007589 **
## Tau2_2_2    0.0248312 0.0096502 0.0059171 0.0437453  2.5731 0.010079 *
## Tau2_3_3    0.0230433 0.0084870 0.0064092 0.0396774  2.7151 0.006625 **
## Tau2_4_4    0.0301790 0.0104189 0.0097583 0.0505996  2.8966 0.003773 **
## Tau2_5_5    0.0196359 0.0068293 0.0062507 0.0330211  2.8752 0.004037 **
## Tau2_6_6    0.0325412 0.0113591 0.0102778 0.0548045  2.8648 0.004173 **
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Q statistic on the homogeneity of effect sizes: 1425.321
## Degrees of freedom of the Q statistic: 114
## P value of the Q statistic: 0
##
## Heterogeneity indices (based on the estimated Tau2):
##
##               Estimate
## Intercept1: I2 (Q statistic)  0.9221
## Intercept2: I2 (Q statistic)  0.9133
## Intercept3: I2 (Q statistic)  0.9152
## Intercept4: I2 (Q statistic)  0.9420
## Intercept5: I2 (Q statistic)  0.9339
## Intercept6: I2 (Q statistic)  0.9564
##
## Number of studies (or clusters): 20
## Number of observed statistics: 120
## Number of estimated parameters: 12
## Degrees of freedom: 108
## -2 log likelihood: -80.71658
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values may indicate problems.)

## Average correlation matrix under a random-effects model
averageR <- vec2symMat(coef(random1, select="fixed"), diag = FALSE)
dimnames(averageR) <- list(obs.vars3, obs.vars3)
averageR

##           Cap      Int      Beh      PB
## Cap 1.0000000 0.5493203 0.3646155 0.3833905
## Int 0.5493203 1.0000000 0.4911759 0.5612951
## Beh 0.3646155 0.4911759 1.0000000 0.5560411
## PB  0.3833905 0.5612951 0.5560411 1.0000000

## Heterogeneity variances of the random-effects
coef(random1, select="random")

##   Tau2_1_1  Tau2_2_2  Tau2_3_3  Tau2_4_4  Tau2_5_5  Tau2_6_6
## 0.01791742 0.02483117 0.02304332 0.03017896 0.01963589 0.03254116
```

Stage 2 analysis

```
RAM3 <- lavaan2RAM(model3, obs.variables=obs.vars3)
RAM3

## $A
##      Cap      Int      Beh PB
## Cap "0"    "0"    "0"  "0*a"
## Int "0*b"  "0"    "0"  "0*d"
## Beh "0*e"  "0*c"  "0"  "0*f"
## PB  "0"    "0"    "0"  "0"
##
## $S
##      Cap      Int      Beh      PB
## Cap "0*CapWITHCap" "0"      "0"      "0"
## Int "0"            "0*IntWITHInt" "0"      "0"
```

```

## Beh "0"          "0"          "0*BehWITHBeh" "0"
## PB  "0"          "0"          "0"          "1"
##
## $F
##   Cap Int Beh PB
## Cap   1   0   0   0
## Int   0   1   0   0
## Beh   0   0   1   0
## PB    0   0   0   1
##
## $M
##   Cap Int Beh PB
## 1    0   0   0   0

tssem.fit <- tssem2(random1, RAM=RAM3, intervals.type = "LB",
                    mx.algebras = list(Ind_CapInt=mxAlgebra(a*b*c, name="Ind_CapInt"),
                                       Ind_Cap=mxAlgebra(a*e, name="Ind_Cap"),
                                       Ind_Int=mxAlgebra(d*c, name="Ind_Int"),
                                       Dir_PB=mxAlgebra(f, name="Dir_PB")))

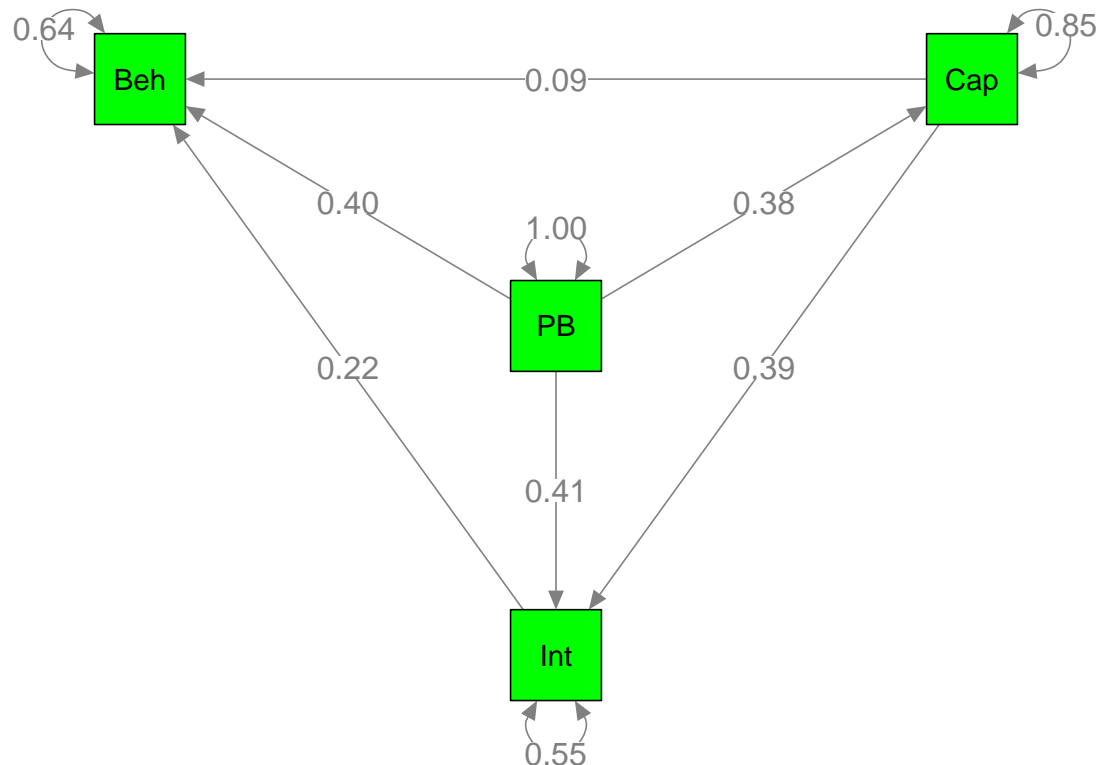
summary(tssem.fit)

##
## Call:
## wls(Cov = pooledS, aCov = aCov, n = tssem1.obj$total.n, RAM = RAM,
##      Amatrix = Amatrix, Smatrix = Smatrix, Fmatrix = Fmatrix,
##      diag.constraints = diag.constraints, cor.analysis = cor.analysis,
##      intervals.type = intervals.type, mx.algebras = mx.algebras,
##      model.name = model.name, suppressWarnings = suppressWarnings,
##      silent = silent, run = run)
##
## 95% confidence intervals: Likelihood-based statistic
## Coefficients:
##      Estimate Std. Error    lbound    ubound z value Pr(>|z|)
## e  0.092868      NA -0.037863  0.219469      NA      NA
## c  0.216138      NA  0.041998  0.380501      NA      NA
## f  0.399119      NA  0.263434  0.534512      NA      NA
## a  0.383390      NA  0.310794  0.455987      NA      NA
## b  0.391700      NA  0.308615  0.471684      NA      NA
## d  0.411121      NA  0.326527  0.492921      NA      NA
##
## mxAlgebras objects (and their 95% likelihood-based CIs):
##              lbound Estimate    ubound
## Ind_CapInt[1,1] 0.006112218 0.03245834 0.06294880
## Ind_Cap[1,1]    -0.015366406 0.03560474 0.08251943
## Ind_Int[1,1]    0.018159937 0.08885880 0.15923180
## Dir_PB[1,1]     0.263433762 0.39911924 0.53451200
##
## Goodness-of-fit indices:
##
##                               Value
## Sample size                   6207.00
## Chi-square of target model      0.00
## DF of target model              0.00
## p value of target model         0.00
## Number of constraints imposed on "Smatrix" 0.00
## DF manually adjusted            0.00

```

```
## Chi-square of independence model          903.82
## DF of independence model                  6.00
## RMSEA                                    0.00
## RMSEA lower 95% CI                      0.00
## RMSEA upper 95% CI                      0.00
## SRMR                                    0.00
## TLI                                    -Inf
## CFI                                    1.00
## AIC                                    0.00
## BIC                                    0.00
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values indicate problems.)
```

```
plot(tssem.fit, layout="circle", color="green")
```



```
sessionInfo()
```

```
## R version 4.0.3 (2020-10-10)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 20.04.1 LTS
##
## Matrix products: default
## BLAS: /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.9.0
## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.9.0
##
## locale:
##  [1] LC_CTYPE=en_SG.UTF-8      LC_NUMERIC=C
##  [3] LC_TIME=en_SG.UTF-8      LC_COLLATE=en_SG.UTF-8
##  [5] LC_MONETARY=en_SG.UTF-8  LC_MESSAGES=en_SG.UTF-8
##  [7] LC_PAPER=en_SG.UTF-8     LC_NAME=C
##  [9] LC_ADDRESS=C             LC_TELEPHONE=C
```

```

## [11] LC_MEASUREMENT=en_SG.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods   base
##
## other attached packages:
## [1] metaSEM_1.2.5 OpenMx_2.18.1
##
## loaded via a namespace (and not attached):
## [1] nlme_3.1-149      RColorBrewer_1.1-2 mi_1.0
## [4] tools_4.0.3      backports_1.1.10   R6_2.5.0
## [7] d3Network_0.5.2.1 rpart_4.1-15       Hmisc_4.4-1
## [10] colorspace_1.4-1 nnet_7.3-14        tidyselect_1.1.0
## [13] gridExtra_2.3     mnormt_2.0.2       compiler_4.0.3
## [16] fdrtool_1.2.15    qgraph_1.6.5       htmlTable_2.1.0
## [19] regsem_1.6.2      scales_1.1.1       checkmate_2.0.0
## [22] psych_2.0.9       mvtnorm_1.1-1      pbapply_1.4-3
## [25] sem_3.1-11        stringr_1.4.0      digest_0.6.27
## [28] pbivnorm_0.6.0    foreign_0.8-80     minqa_1.2.4
## [31] rmarkdown_2.5     base64enc_0.1-3    jpeg_0.1-8.1
## [34] pkgconfig_2.0.3   htmltools_0.5.0    lme4_1.1-23
## [37] lisrelToR_0.1.4   htmlwidgets_1.5.2  rlang_0.4.8
## [40] huge_1.3.4.1      rstudioapi_0.11    generics_0.0.2
## [43] gtools_3.8.2      dplyr_1.0.2        zip_2.1.1
## [46] magrittr_1.5      Formula_1.2-3      Matrix_1.2-18
## [49] Rcpp_1.0.5        munsell_0.5.0      abind_1.4-5
## [52] rockchalk_1.8.144 lifecycle_0.2.0    whisker_0.4
## [55] stringi_1.5.3     yaml_2.2.1         carData_3.0-4
## [58] MASS_7.3-53       plyr_1.8.6         matrixcalc_1.0-3
## [61] lavaan_0.6-7      grid_4.0.3         parallel_4.0.3
## [64] crayon_1.3.4      lattice_0.20-41    semPlot_1.1.2
## [67] kutils_1.70       splines_4.0.3      tmvnsim_1.0-2
## [70] knitr_1.30        pillar_1.4.6       igraph_1.2.6
## [73] rjson_0.2.20      boot_1.3-25        corpcor_1.6.9
## [76] BDgraph_2.63      reshape2_1.4.4     stats4_4.0.3
## [79] XML_3.99-0.5      glue_1.4.2         evaluate_0.14
## [82] latticeExtra_0.6-29 data.table_1.13.2  png_0.1-7
## [85] vctrs_0.3.4       nloptr_1.2.2.2     gtable_0.3.0
## [88] purrr_0.3.4       ggplot2_3.3.2      xfun_0.19
## [91] openxlsx_4.2.2    xtable_1.8-4       coda_0.19-4
## [94] Rsolnp_1.16       glasso_1.11        survival_3.2-7
## [97] truncnorm_1.0-8   tibble_3.0.4       arm_1.11-2
## [100] ellipse_0.4.2     cluster_2.1.0      statmod_1.4.35
## [103] ellipsis_0.3.1

```