```
> ##Model 2: PSY --> MOV;PAIN --> PSY; PAIN-->MOV


>
> ## Load the library
> library(metaSEM)
>
> ## Create the matrix ##
>
> Studiestest10 <-
+
+ structure(list(data = structure(list(`Trost2012` = structure(c(1,-
0.38,-0.11,-0.38,1,0.14,-0.11,0.14,1), .Dim = c(3L, 3L), .Dimnames =
list(c("Movement",
+
+ "Psychological_factor", "Pain_intensity"), c("Movement",
+
+ "Psychological_factor", "Pain_intensity"))), `Thomas2008A` =
structure(c(1,-0.54,-0.24,-0.54,1,0.28,-0.24,0.28,1), .Dim = c(3L, 3L),
.Dimnames = list(c("Movement",
+
+ "Psychological_factor", "Pain_intensity"), c("Movement",
+
+ "Psychological_factor", "Pain_intensity"))), `Thomas2008B` =
structure(c(1,-0.41,-0.24,-0.41,1,0.28,-0.24,0.28,1), .Dim = c(3L, 3L),
.Dimnames = list(c("Movement",
+
+ "Psychological_factor", "Pain_intensity"), c("Movement",
+
+ "Psychological_factor", "Pain_intensity"))), `Thomas2008C` =
structure(c(1,-0.35,-0.24,-0.35,1,0.28,-0.24,0.28,1), .Dim = c(3L, 3L),
.Dimnames = list(c("Movement",
+
+ "Psychological_factor", "Pain_intensity"), c("Movement",
+
+ "Psychological_factor", "Pain_intensity"))), `Watson1997A` =
structure(c(1,-0.28,-0.2,-0.28,1,0.28,-0.2,0.28,1), .Dim = c(3L, 3L),
.Dimnames = list(c("Movement",
+
+ "Psychological_factor", "Pain_intensity"), c("Movement",
+
+ "Psychological_factor", "Pain_intensity"))), `Watson1997B` =
structure(c(1,-0.27,-0.2,-0.27,1,0.28,-0.2,0.28,1), .Dim = c(3L, 3L),
.Dimnames = list(c("Movement",
+
+ "Psychological_factor", "Pain_intensity"), c("Movement",
+
+ "Psychological_factor", "Pain_intensity"))), `Alschuler2009` =
structure(c(1,-0.45,-0.11,-0.45,1,0.28,-0.11,0.28,1), .Dim = c(3L, 3L),
.Dimnames = list(c("Movement",
+
+ "Psychological_factor", "Pain_intensity"), c("Movement",
+
+ "Psychological_factor", "Pain_intensity"))), `Grotle2004A` =
structure(c(1,-0.08,-0.19,-0.08,1,0.13,-0.19,0.13,1), .Dim = c(3L, 3L),
.Dimnames = list(c("Movement",
+
```

```
+ "Psychological_factor", "Pain_intensity"), c("Movement",
+
+ "Psychological_factor", "Pain_intensity"))), `Grotle2004B` =
structure(c(1,-0.11,-0.23,-0.11,1,0.21,-0.23,0.21,1), .Dim = c(3L, 3L),
.Dimnames = list(c("Movement",
+
+ "Psychological_factor", "Pain_intensity"), c("Movement",
+
+ "Psychological_factor", "Pain_intensity"))), `Kim2017` =
structure(c(1,-0.03,-0.02,-0.03,1,0.25,-0.02,0.25,1), .Dim = c(3L, 3L),
.Dimnames = list(c("Movement",
+
+ "Psychological_factor", "Pain_intensity"), c("Movement",
+
+ "Psychological_factor", "Pain_intensity")))), .Names = c("Trost2012",
"Thomas2008A", "Thomas2008B", "Thomas2008C", "Watson1997A",
"Watson1997B", "Alschuler2009", "Grotle2004A", "Grotle2004B",
"Kim2017")), n = c(51, 36, 36, 36, 36, 36, 76, 123, 233, 30),
Questionnaire = c("PASS", "PASS", "TSK", "PCS", "FABQ", "self_efficacy",
"TSK", "FABQ", "FABQ", "FABQ")), .Names = c("data", "n",
+
+ "Questionnaire"))
>
> ## show the studies ##
>
> head(Studiestest10$data)
$`Trost2012`
                    Movement Psychological_factor Pain_intensity
Movement                1.00                -0.38          -0.11
Psychological_factor   -0.38                 1.00           0.14
Pain_intensity         -0.11                 0.14           1.00

$Thomas2008A
                    Movement Psychological_factor Pain_intensity
Movement                1.00                -0.54          -0.24
Psychological_factor   -0.54                 1.00           0.28
Pain_intensity         -0.24                 0.28           1.00

$Thomas2008B
                    Movement Psychological_factor Pain_intensity
Movement                1.00                -0.41          -0.24
Psychological_factor   -0.41                 1.00           0.28
Pain_intensity         -0.24                 0.28           1.00

$Thomas2008C
                    Movement Psychological_factor Pain_intensity
Movement                1.00                -0.35          -0.24
Psychological_factor   -0.35                 1.00           0.28
Pain_intensity         -0.24                 0.28           1.00

$Watson1997A
                    Movement Psychological_factor Pain_intensity
Movement                1.00                -0.28          -0.20
Psychological_factor   -0.28                 1.00           0.28
Pain_intensity         -0.20                 0.28           1.00

$Watson1997B
                    Movement Psychological_factor Pain_intensity
```

```
Movement                 1.00                -0.27          -0.20
Psychological_factor    -0.27                 1.00           0.28
Pain_intensity          -0.20                 0.28           1.00

>
> ## Display the sample sizes
> Studiestest10$n
 [1]  51  36  36  36  36  36  76 123 233  30
>
> ## Variables used in the analysis
> var.names <- c("Mov", "Psy", "Pai")

> ### Stage 1 from Hagger (give the same results as version from Cheung)
> random1 <- tssem1(Studiestest10$data, Studiestest10$n, method="REM",
RE.type="Diag", acov="weighted")
> 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
Coefficients:
             Estimate   Std.Error      lbound      ubound  z value
Pr(>|z|)
Intercept1 -2.6959e-01  5.5461e-02 -3.7830e-01 -1.6089e-01  -4.8610
1.168e-06 ***
Intercept2 -1.9026e-01  1.7026e-02 -2.2363e-01 -1.5689e-01 -11.1744 <
2.2e-16 ***
Intercept3  2.1824e-01  3.4400e-02  1.5082e-01  2.8566e-01   6.3442
2.236e-10 ***
Tau2_1_1    1.3085e-02  1.0824e-02 -8.1289e-03  3.4299e-02   1.2089
0.2267
Tau2_2_2    1.0000e-10          NA          NA          NA       NA
NA
Tau2_3_3    1.0000e-10          NA          NA          NA       NA
NA
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Q statistic on the homogeneity of effect sizes: 25.07557
Degrees of freedom of the Q statistic: 27
P value of the Q statistic: 0.570227

Heterogeneity indices (based on the estimated Tau2):
                            Estimate
Intercept1: I2 (Q statistic)   0.4793
Intercept2: I2 (Q statistic)   0.0000
Intercept3: I2 (Q statistic)   0.0000

Number of studies (or clusters): 10
Number of observed statistics: 30
Number of estimated parameters: 6
Degrees of freedom: 24
-2 log likelihood: -47.77288
OpenMx status1: 5 ("0" or "1": The optimization is considered fine.
```

```
Other values may indicate problems.)
>
> ## Extract the fixed-effects estimates
> (est_fixed <- coef(random1, select="fixed"))
Intercept1 Intercept2 Intercept3
-0.2695942 -0.1902597  0.2182395
>
> ## Convert the estimated vector to a symmetrical matrix
> ## where the diagonals are fixed at 1 (for a correlation matrix)
> vec2symMat(est_fixed, diag=FALSE)
           [,1]       [,2]       [,3]
[1,]  1.0000000 -0.2695942 -0.1902597
[2,] -0.2695942  1.0000000  0.2182395
[3,] -0.1902597  0.2182395  1.0000000


> ## stage 2
> ## Regression coefficients --> why 0.2 (same as Hagger) ???
> A1 <- create.mxMatrix(c(0, "0.2*PsytoMov", "0.2*PaitoMov",
+ 0, 0, "0.2*PaitoPsy",
+ 0, 0, 0),
+ type="Full", byrow=TRUE, ncol=3, nrow=3,
+ as.mxMatrix=FALSE)
>
> ## This step is not necessary but it is useful for inspecting the
model.
> dimnames(A1)[[1]] <- dimnames(A1)[[2]] <- c("Mov","Psy","Pai")
> A1
    Mov Psy            Pai
Mov "0" "0.2*PsytoMov" "0.2*PaitoMov"
Psy "0" "0"            "0.2*PaitoPsy"
Pai "0" "0"            "0"
>
> ## Covariance matrix among the variables (CAUTION: NOT SURE FOR THIS
STEP --> why 0.2 ???)
> S1 <- create.mxMatrix(c("0.2*e_Mov",0,0,
+ 0, "0.2*e_Psy", 0,
+ 0, 0, "0.2*e_Pai"),
+ type="Full", byrow=TRUE, ncol=3, nrow=3, as.mxMatrix=FALSE)
>
> ## This step is not necessary but it is useful for inspecting the
model.
> dimnames(S1)[[1]] <- dimnames(S1)[[2]] <- c("Mov","Psy","Pai")
> S1
    Mov         Psy         Pai
Mov "0.2*e_Mov" "0"         "0"
Psy "0"         "0.2*e_Psy" "0"
Pai "0"         "0"         "0.2*e_Pai"
> ## Stage 2 analysis: different option are in the website from Cheung
and in its paper
> ## option 3 (Cheung/Hagger) --> works like option 1; but cannot have
the indirect effect
>
> random2 <- tssem2(random1, Amatrix=A1, Smatrix=S1,
diag.constraints=TRUE,
+ intervals.type="LB", mx.algebras( Ind=mxAlgebra(PaitoPsy*PaitoMov,
name="Ind"))
+
```

```
+
+ ## Refit again to have all the LBCI
+ random2 <- rerun(random2)
Erreur : unexpected symbol in:
"## Refit again to have all the LBCI
random2"
> summary(random2)

Call:
wls(Cov = pooledS, aCov = aCov, n = tssem1.obj$total.n, 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|)
PaitoMov -0.137996        NA -0.177185 -0.098776      NA       NA
PsytoMov -0.239478        NA -0.353150 -0.125946      NA       NA
PsytoPai  0.218240        NA  0.150818  0.285661      NA       NA

Goodness-of-fit indices:
                                           Value
Sample size                               693.00
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          141.58
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.)
>
> ## Display the A matrix
> mxEval(Amatrix, random2$mx.fit)
    Mov        Psy       Pai
Mov   0 -0.2394781 -0.137996
Psy   0  0.0000000  0.000000
Pai   0  0.2182395  0.000000
>
>
> ## Draw the output
> library(semPlot)
>
> ## Convert the model to semPlotModel object
```

```
> my.plot <- meta2semPlot(random2)
>
> ## Plot the parameter estimates
> semPaths(my.plot, whatLabels="est", nCharNodes=10, color="yellow",
+          layout="circle2", edge.label.cex = 0.8)
>
```