

```

require(OpenMx)

## Loading required package: OpenMx

## To take full advantage of multiple cores, use:
##   mxOption(key='Number of Threads', value=parallel::detectCores()) #now
##   Sys.setenv(OMP_NUM_THREADS=parallel::detectCores()) #before library(OpenMx)
library(metaSEM)

## "SLSQP" is set as the default optimizer in OpenMx.
## mxOption(NULL, "Gradient algorithm") is set at "central".
## mxOption(NULL, "Optimality tolerance") is set at "6.3e-14".
## mxOption(NULL, "Gradient iterations") is set at "2".

# Use more than one cores
mxOption(NULL, 'Number of Threads', parallel::detectCores()-2)

# Load data
R1 <- matrix(c(1.00, 0.41, 0.11, 0.31, 0.20, 0.53,
              0.41, 1.00, 0.19, 0.01, 0.35, 0.55,
              0.11, 0.19, 1.00, 0.18, 0.28, 0.09,
              0.31, 0.01, 0.18, 1.00, 0.01, 0.55,
              0.20, 0.35, 0.28, 0.01, 1.00, 0.51,
              0.53, 0.55, 0.09, 0.55, 0.51, 1.00), nrow=6, ncol=6, byrow = TRUE,
            dimnames = list(c("LC","DA","MT","PK","PE","PS"),
                          c("LC","DA","MT","PK","PE","PS")))
R2 <- matrix(c(1.000, -0.045, -0.030, -0.021, 0.008, 0.090,
              -0.045, 1.000, 0.323, 0.504, 0.435, 0.520,
              -0.030, 0.323, 1.000, 0.351, 0.326, 0.409,
              -0.021, 0.504, 0.351, 1.000, 0.342, 0.536,
              0.008, 0.435, 0.326, 0.342, 1.000, 0.588,
              0.090, 0.520, 0.409, 0.536, 0.588, 1.000), nrow=6, ncol=6, byrow = TRUE,
            dimnames = list(c("LC","DA","MT","PK","PE","PS"),
                          c("LC","DA","MT","PK","PE","PS")))
R3 <- matrix(c(NA, NA, NA, NA, NA, NA,
              NA, 1.000, 0.580, -0.228, 0.866, 0.351,
              NA, 0.580, 1.000, -0.028, 0.395, 0.818,
              NA, -0.228, -0.028, 1.000, -0.015, 0.183,
              NA, 0.866, 0.395, -0.015, 1.000, 0.026,
              NA, 0.351, 0.818, 0.183, 0.026, 1.000), nrow=6, ncol=6, byrow = TRUE,
            dimnames = list(c("LC","DA","MT","PK","PE","PS"),
                          c("LC","DA","MT","PK","PE","PS")))
R4 <- matrix(c(NA, NA, NA, NA, NA, NA,
              NA, 1.000, 0.552, -0.098, 0.629, 0.741,
              NA, 0.552, 1.000, -0.365, 0.387, 0.780,
              NA, -0.098, -0.365, 1.000, 0.318, -0.468,
              NA, 0.629, 0.387, 0.318, 1.000, 0.663,
              NA, 0.741, 0.780, -0.468, 0.663, 1.000), nrow=6, ncol=6, byrow = TRUE,
            dimnames = list(c("LC","DA","MT","PK","PE","PS"),
                          c("LC","DA","MT","PK","PE","PS")))
R5 <- matrix(c(1.000, 0.410, 0.236, -0.166, 0.409, 0.368,
              0.410, 1.000, 0.501, -0.220, 0.637, 0.615,
              0.236, 0.501, 1.000, -0.071, 0.514, 0.574,

```

```

-0.166, -0.220, -0.071, 1.000, -0.004, -0.090,
0.409, 0.637, 0.514, -0.004, 1.000, 0.653,
0.368, 0.615, 0.574, -0.090, 0.653, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
c("LC", "DA", "MT", "PK", "PE", "PS")))
R6 <- matrix(c(NA, NA, NA, NA, NA, NA,
NA, 1.00, -0.02, 0.12, 0.28, 0.13,
NA, -0.02, 1.00, 0.30, 0.32, 0.38,
NA, 0.12, 0.30, 1.00, 0.26, 0.18,
NA, 0.28, 0.32, 0.26, 1.00, 0.54,
NA, 0.13, 0.38, 0.18, 0.54, 1.00), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
c("LC", "DA", "MT", "PK", "PE", "PS")))
R7 <- matrix(c(1.000, -0.081, -0.347, -0.258, -0.176, -0.023,
-0.081, 1.000, 0.156, 0.423, 0.223, 0.575,
-0.347, 0.156, 1.000, 0.267, 0.407, -0.092,
-0.258, 0.423, 0.267, 1.000, 0.362, 0.390,
-0.176, 0.223, 0.407, 0.362, 1.000, 0.317,
-0.023, 0.575, -0.092, 0.390, 0.317, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
c("LC", "DA", "MT", "PK", "PE", "PS")))
R8 <- matrix(c(1.000, 0.582, -0.731, -0.539, 0.501, 0.723,
0.582, 1.000, -0.317, -0.455, 0.451, 0.448,
-0.731, -0.317, 1.000, 0.344, -0.484, -0.634,
-0.539, -0.455, 0.344, 1.000, -0.661, -0.536,
0.501, 0.451, -0.484, -0.661, 1.000, 0.678,
0.723, 0.448, -0.634, -0.536, 0.678, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
c("LC", "DA", "MT", "PK", "PE", "PS")))
R9 <- matrix(c(1.000, 0.290, -0.167, 0.333, 0.432, -0.062,
0.290, 1.000, 0.513, 0.467, 0.610, 0.055,
-0.167, 0.513, 1.000, 0.195, 0.245, -0.125,
0.333, 0.467, 0.195, 1.000, 0.397, -0.186,
0.432, 0.610, 0.245, 0.397, 1.000, 0.200,
-0.062, 0.055, -0.125, -0.186, 0.200, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
c("LC", "DA", "MT", "PK", "PE", "PS")))
R10 <- matrix(c(1.00, 0.34, 0.36, 0.14, 0.40, 0.53,
0.34, 1.00, 0.19, 0.17, 0.24, 0.29,
0.36, 0.19, 1.00, 0.00, 0.29, 0.53,
0.14, 0.17, 0.00, 1.00, 0.03, 0.07,
0.40, 0.24, 0.29, 0.03, 1.00, 0.46,
0.53, 0.29, 0.53, 0.07, 0.46, 1.00), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
c("LC", "DA", "MT", "PK", "PE", "PS")))
R11 <- matrix(c(NA, NA, NA, NA, NA, NA,
NA, 1.000, -0.334, -0.645, -0.078, 0.000,
NA, -0.334, 1.000, 0.345, -0.680, 0.009,
NA, -0.645, 0.345, 1.000, -0.002, 0.252,
NA, -0.078, -0.68, -0.002, 1, 0.309,
NA, 0.000, 0.009, 0.252, 0.309, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
c("LC", "DA", "MT", "PK", "PE", "PS")))

```

```

R12 <- matrix(c(NA, NA, NA, NA, NA, NA,
  NA, 1.000, 0.139, 0.271, 0.413, 0.567,
  NA, 0.139, 1.000, 0.585, -0.393, -0.159,
  NA, 0.271, 0.585, 1.000, -0.288, 0.200,
  NA, 0.413, -0.393, -0.288, 1.000, 0.696,
  NA, 0.567, -0.159, 0.200, 0.696, 1.000), nrow=6, ncol=6, byrow = TRUE,
  dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
    c("LC", "DA", "MT", "PK", "PE", "PS")))
R13 <- matrix(c(1.00, -0.06, -0.11, -0.24, 0.06, -0.04,
  -0.06, 1.00, 0.07, -0.11, 0.32, 0.01,
  -0.11, 0.07, 1.00, -0.29, 0.13, 0.25,
  -0.24, -0.11, -0.29, 1.00, -0.07, -0.22,
  0.06, 0.32, 0.13, -0.07, 1.00, 0.41,
  -0.04, 0.01, 0.25, -0.22, 0.41, 1.00), nrow=6, ncol=6, byrow = TRUE,
  dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
    c("LC", "DA", "MT", "PK", "PE", "PS")))
R14 <- matrix(c(1.00, 0.30, -0.22, 0.33, 0.07, 0.28,
  0.30, 1.00, -0.06, 0.12, 0.20, 0.41,
  -0.22, -0.06, 1.00, -0.33, -0.20, 0.09,
  0.33, 0.12, -0.33, 1.00, 0.03, 0.08,
  0.07, 0.20, -0.20, 0.03, 1.00, 0.41,
  0.28, 0.41, 0.09, 0.08, 0.41, 1.00), nrow=6, ncol=6, byrow = TRUE,
  dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
    c("LC", "DA", "MT", "PK", "PE", "PS")))
R15 <- matrix(c(1.000, -0.104, -0.361, -0.500, -0.072, -0.040,
  -0.104, 1.000, -0.048, 0.340, 0.731, 0.593,
  -0.361, -0.048, 1.000, 0.355, -0.061, -0.049,
  -0.500, 0.340, 0.355, 1.000, 0.275, 0.188,
  -0.072, 0.731, -0.061, 0.275, 1.000, 0.706,
  -0.040, 0.593, -0.049, 0.188, 0.706, 1.000), nrow=6, ncol=6, byrow = TRUE,
  dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
    c("LC", "DA", "MT", "PK", "PE", "PS")))
R16 <- matrix(c(1.00, 0.18, 0.13, 0.12, 0.26, 0.24,
  0.18, 1.00, 0.10, 0.10, 0.31, 0.25,
  0.13, 0.10, 1.00, 0.28, 0.21, -0.18,
  0.12, 0.10, 0.28, 1.00, 0.10, 0.10,
  0.26, 0.31, 0.21, 0.10, 1.00, 0.32,
  0.24, 0.25, -0.18, 0.10, 0.32, 1.00), nrow=6, ncol=6, byrow = TRUE,
  dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
    c("LC", "DA", "MT", "PK", "PE", "PS")))
R17 <- matrix(c(1.00, 0.37, -0.09, 0.23, 0.32, 0.33,
  0.37, 1.00, -0.36, 0.21, 0.50, 0.64,
  -0.09, -0.36, 1.00, -0.01, -0.43, -0.54,
  0.23, 0.21, -0.01, 1.00, 0.22, 0.19,
  0.32, 0.50, -0.43, 0.22, 1.00, 0.58,
  0.33, 0.64, -0.54, 0.19, 0.58, 1.00), nrow=6, ncol=6, byrow = TRUE,
  dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
    c("LC", "DA", "MT", "PK", "PE", "PS")))
R18 <- matrix(c(1.000, 0.188, -0.137, 0.288, 0.231, 0.063,
  0.188, 1.000, -0.536, 0.190, 0.472, 0.642,
  -0.137, -0.536, 1.000, -0.076, -0.482, -0.689,
  0.288, 0.190, -0.076, 1.000, 0.121, 0.125,
  0.231, 0.472, -0.482, 0.121, 1.000, 0.533,

```

```

        0.063, 0.642, -0.689, 0.125, 0.533, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R19 <- matrix(c(1.000, 0.213, 0.388, 0.583, -0.018, 0.162,
                0.213, 1.000, 0.275, 0.261, 0.302, 0.565,
                0.388, 0.275, 1.000, 0.475, -0.159, 0.042,
                0.583, 0.261, 0.475, 1.000, -0.059, 0.131,
                -0.018, 0.302, -0.159, -0.059, 1.000, 0.468,
                0.162, 0.565, 0.042, 0.131, 0.468, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R20 <- matrix(c(NA, NA, NA, NA, NA, NA,
                NA, 1.000, -0.288, 0.041, 0.232, 0.483,
                NA, -0.288, 1.000, 0.260, -0.188, -0.158,
                NA, 0.041, 0.260, 1.000, 0.085, 0.044,
                NA, 0.232, -0.188, 0.085, 1.000, 0.379,
                NA, 0.483, -0.158, 0.044, 0.379, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R21 <- matrix(c(1.00, 0.22, 0.21, 0.22, 0.17, 0.22,
                0.22, 1.00, -0.76, 0.10, 0.55, 0.67,
                0.21, -0.76, 1.00, 0.26, -0.15, -0.11,
                0.22, 0.10, 0.26, 1.00, 0.07, 0.00,
                0.17, 0.55, -0.15, 0.07, 1.00, 0.66,
                0.22, 0.67, -0.11, 0.00, 0.66, 1.00), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R22 <- matrix(c(1.00, -0.06, -0.24, -0.05, -0.23, 0.09,
                -0.06, 1.00, -0.10, 0.03, 0.31, 0.51,
                -0.24, -0.10, 1.00, 0.23, 0.15, 0.15,
                -0.05, 0.03, 0.23, 1.00, 0.30, 0.11,
                -0.23, 0.31, 0.15, 0.30, 1.00, 0.42,
                0.09, 0.51, 0.15, 0.11, 0.42, 1.00), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R23 <- matrix(c(1.000, 0.212, 0.256, 0.349, -0.165, 0.276,
                0.212, 1.000, 0.533, 0.217, 0.311, 0.376,
                0.256, 0.533, 1.000, 0.383, 0.033, 0.207,
                0.349, 0.217, 0.383, 1.000, 0.052, 0.619,
                -0.165, 0.311, 0.033, 0.052, 1.000, 0.361,
                0.276, 0.376, 0.207, 0.619, 0.361, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R24 <- matrix(c(1.000, 0.186, -0.203, -0.104, 0.178, 0.336,
                0.186, 1.000, -0.158, 0.363, 0.492, 0.588,
                -0.203, -0.158, 1.000, 0.105, -0.211, -0.292,
                -0.104, 0.363, 0.105, 1.000, 0.075, 0.273,
                0.178, 0.492, -0.211, 0.075, 1.000, 0.634,
                0.336, 0.588, -0.292, 0.273, 0.634, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R25 <- matrix(c(NA, NA, NA, NA, NA, NA,
                NA, 1.000, 0.191, -0.073, 0.277, 0.566,

```

```

NA, 0.191, 1.000, -0.092, 0.261, 0.252,
NA, -0.073, -0.092, 1.000, 0.006, 0.086,
NA, 0.277, 0.261, 0.006, 1.000, 0.525,
NA, 0.566, 0.252, 0.086, 0.525, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R26 <- matrix(c(1.000, 0.189, 0.301, 0.504, 0.114, 0.203,
                0.189, 1.000, 0.040, -0.010, 0.077, 0.268,
                0.301, 0.040, 1.000, 0.289, -0.067, -0.158,
                0.504, -0.010, 0.289, 1.000, 0.193, 0.136,
                0.114, 0.077, -0.067, 0.193, 1.000, 0.477,
                0.203, 0.268, -0.158, 0.136, 0.477, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R27 <- matrix(c(NA, NA, NA, NA, NA, NA,
                NA, 1.000, 0.396, 0.011, -0.056, 0.052,
                NA, 0.396, 1.000, 0.214, 0.580, 0.764,
                NA, 0.011, 0.214, 1.000, -0.306, -0.038,
                NA, -0.056, 0.580, -0.306, 1.000, 0.614,
                NA, 0.052, 0.764, -0.038, 0.614, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R28 <- matrix(c(NA, NA, NA, NA, NA, NA,
                NA, 1.000, 0.271, 0.671, 0.239, -0.211,
                NA, 0.271, 1.000, 0.307, 0.585, 0.255,
                NA, 0.671, 0.307, 1.000, -0.191, -0.339,
                NA, 0.239, 0.585, -0.191, 1.000, 0.455,
                NA, -0.211, 0.255, -0.339, 0.455, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R29 <- matrix(c(1.000, 0.059, -0.299, -0.116, 0.098, 0.174,
                0.059, 1.000, 0.064, 0.192, 0.282, 0.324,
                -0.299, 0.064, 1.000, 0.146, 0.218, 0.437,
                -0.116, 0.192, 0.146, 1.000, 0.209, 0.162,
                0.098, 0.282, 0.218, 0.209, 1.000, 0.357,
                0.174, 0.324, 0.437, 0.162, 0.357, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R30 <- matrix(c(1.000, 0.521, 0.173, 0.448, 0.526, 0.407,
                0.521, 1.000, 0.659, 0.657, 0.820, 0.556,
                0.173, 0.659, 1.000, 0.559, 0.542, 0.597,
                0.448, 0.657, 0.559, 1.000, 0.455, 0.559,
                0.526, 0.820, 0.542, 0.455, 1.000, 0.601,
                0.407, 0.556, 0.597, 0.559, 0.601, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                c("LC", "DA", "MT", "PK", "PE", "PS")))
R31 <- matrix(c(1.000, 0.344, 0.365, 0.549, -0.032, 0.165,
                0.344, 1.000, 0.052, 0.340, 0.207, 0.514,
                0.365, 0.052, 1.000, 0.346, -0.202, -0.207,
                0.549, 0.340, 0.346, 1.000, 0.095, 0.175,
                -0.032, 0.207, -0.202, 0.095, 1.000, 0.279,
                0.165, 0.514, -0.207, 0.175, 0.279, 1.000), nrow=6, ncol=6, byrow = TRUE,
dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),

```

```

                                c("LC", "DA", "MT", "PK", "PE", "PS"))
R32 <- matrix(c(1.000, 0.194, 0.291, 0.232, 0.359, 0.374,
               0.194, 1.000, 0.493, 0.405, 0.518, 0.474,
               0.291, 0.493, 1.000, 0.383, 0.511, 0.445,
               0.232, 0.405, 0.383, 1.000, 0.496, 0.402,
               0.359, 0.518, 0.511, 0.496, 1.000, 0.539,
               0.374, 0.474, 0.445, 0.402, 0.539, 1.000), nrow=6, ncol=6, byrow = TRUE,
             dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                             c("LC", "DA", "MT", "PK", "PE", "PS")))
R33 <- matrix(c(1.000, 0.283, 0.308, 0.412, 0.096, 0.304,
               0.283, 1.000, 0.040, 0.352, 0.347, 0.544,
               0.308, 0.040, 1.000, 0.258, -0.241, -0.065,
               0.412, 0.352, 0.258, 1.000, 0.168, 0.388,
               0.096, 0.347, -0.241, 0.168, 1.000, 0.502,
               0.304, 0.544, -0.065, 0.388, 0.502, 1.000), nrow=6, ncol=6, byrow = TRUE,
             dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                             c("LC", "DA", "MT", "PK", "PE", "PS")))
R34 <- matrix(c(NA, NA, NA, NA, NA, NA,
               NA, 1.000, 0.179, -0.050, 0.455, 0.224,
               NA, 0.179, 1.000, 0.052, 0.255, 0.206,
               NA, -0.050, 0.052, 1.000, 0.000, -0.042,
               NA, 0.455, 0.255, 0.000, 1.000, 0.306,
               NA, 0.224, 0.206, -0.042, 0.306, 1.000), nrow=6, ncol=6, byrow = TRUE,
             dimnames = list(c("LC", "DA", "MT", "PK", "PE", "PS"),
                             c("LC", "DA", "MT", "PK", "PE", "PS")))

cordat <- list(R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14,
              R15, R16, R17, R18, R19, R20, R21, R22, R23, R24, R25, R26,
              R27, R28, R29, R30, R31, R32, R33, R34)

nvar <- 6 # number of observed variables

# Use this code; Don't do this manually as doing so is error prone.
# put NA on diagonal if variable is missing (adapted from http://www.suzannejak.nl/Roorda_SES.R). The o
for (i in 1:length(cordat)){
  for (j in 1:nrow(cordat[[i]])){
    if (sum(is.na(cordat[[i]][j,]))==nvar-1)
      {cordat[[i]][j,j] <- NA}
  }}

# put NA on diagonal for variable with least present correlations (adapted from http://www.suzannejak.n
for (i in 1:length(cordat)){
  for (j in 1:nrow(cordat[[i]])){
    for (k in 1:nvar){
      if (is.na(cordat[[i]][j,k])==TRUE
          &is.na(cordat[[i]][j,j])!=TRUE
          &is.na(cordat[[i]][k,k])!=TRUE){

if(sum(is.na(cordat[[i]][j,])>sum(is.na(cordat[[i]][k,]))
{cordat[[i]][k,k] = NA}
if(sum(is.na(cordat[[i]][j,])<=sum(is.na(cordat[[i]][k,]))
{cordat[[i]][j,j] = NA}
}}}}

```

```
# Group 4 is not positive definite. Group 21 is not positive definite. The correlation matrices for the
```

```
# Research question 2
```

```
# Load the data, including column 1 (i.e., LC [listening comprehension])
```

```
cordat2 <- list(R1, R2, R3, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14,  
              R15, R16, R17, R18, R19, R20, R22, R23, R24, R25, R26,  
              R27, R28, R29, R30, R31, R32, R33, R34)
```

```
N2 <- c(386, 250, 6, 310, 88, 46, 22, 22, 36, 17, 15, 100, 113, 41, 96, 501,  
       127, 182, 117, 76, 17, 67, 141, 232, 10, 10, 100, 30, 263, 106, 217, 877)
```

```
stage1random2 <- tssem1(Cov = cordat2, n = N2, method = "REM", RE.type = "Diag")  
summary(stage1random2)
```

```
##
```

```
## 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	0.21450681	0.03116488	0.15342477	0.27558886	6.8830	5.862e-12
## Intercept2	0.01569689	0.05354316	-0.08924577	0.12063955	0.2932	0.7693973
## Intercept3	0.13365385	0.06053358	0.01501021	0.25229749	2.2079	0.0272492
## Intercept4	0.14979435	0.03222933	0.08662602	0.21296268	4.6478	3.356e-06
## Intercept5	0.24070535	0.02940325	0.18307605	0.29833466	8.1864	2.220e-16
## Intercept6	0.08948024	0.04439497	0.00246769	0.17649279	2.0155	0.0438472
## Intercept7	0.15700238	0.03784544	0.08282668	0.23117807	4.1485	3.346e-05
## Intercept8	0.36868548	0.02706352	0.31564196	0.42172900	13.6230	< 2.2e-16
## Intercept9	0.42250604	0.03236289	0.35907593	0.48593615	13.0553	< 2.2e-16
## Intercept10	0.17828489	0.03990859	0.10006550	0.25650428	4.4673	7.920e-06
## Intercept11	0.06055016	0.05169881	-0.04077765	0.16187797	1.1712	0.2415144
## Intercept12	0.07362111	0.05634813	-0.03681919	0.18406140	1.3065	0.1913688
## Intercept13	0.11719285	0.02703441	0.06420637	0.17017932	4.3350	1.458e-05
## Intercept14	0.16554242	0.03546830	0.09602582	0.23505901	4.6673	3.051e-06
## Intercept15	0.47477021	0.02142407	0.43277982	0.51676061	22.1606	< 2.2e-16
## Tau2_1_1	0.01230680	0.00556867	0.00139240	0.02322120	2.2100	0.0271047
## Tau2_2_2	0.05216843	0.01961893	0.01371602	0.09062083	2.6591	0.0078353
## Tau2_3_3	0.07091266	0.02475103	0.02240152	0.11942380	2.8650	0.0041696
## Tau2_4_4	0.01366655	0.00647274	0.00098022	0.02635288	2.1114	0.0347378
## Tau2_5_5	0.01033136	0.00508024	0.00037428	0.02028844	2.0336	0.0419881
## Tau2_6_6	0.04584753	0.01440355	0.01761709	0.07407798	3.1831	0.0014572
## Tau2_7_7	0.03057496	0.01122506	0.00857424	0.05257567	2.7238	0.0064533
## Tau2_8_8	0.01335317	0.00532333	0.00291963	0.02378670	2.5084	0.0121271
## Tau2_9_9	0.02238221	0.00783436	0.00702714	0.03773728	2.8569	0.0042776
## Tau2_10_10	0.03446514	0.01168017	0.01157242	0.05735787	2.9507	0.0031701
## Tau2_11_11	0.06712966	0.02046013	0.02702853	0.10723078	3.2810	0.0010344
## Tau2_12_12	0.08253905	0.02413923	0.03522702	0.12985107	3.4193	0.0006278
## Tau2_13_13	0.01122111	0.00541191	0.00061396	0.02182826	2.0734	0.0381341
## Tau2_14_14	0.02601382	0.00910347	0.00817135	0.04385628	2.8576	0.0042689
## Tau2_15_15	0.00698026	0.00291652	0.00126400	0.01269653	2.3934	0.0166950

```

##
## Intercept1 ***
## Intercept2
## Intercept3 *
## Intercept4 ***
## Intercept5 ***
## Intercept6 *
## Intercept7 ***
## Intercept8 ***
## Intercept9 ***
## Intercept10 ***
## Intercept11
## Intercept12
## Intercept13 ***
## Intercept14 ***
## Intercept15 ***
## Tau2_1_1 *
## Tau2_2_2 **
## Tau2_3_3 **
## Tau2_4_4 *
## Tau2_5_5 *
## Tau2_6_6 **
## Tau2_7_7 **
## Tau2_8_8 *
## Tau2_9_9 **
## Tau2_10_10 **
## Tau2_11_11 **
## Tau2_12_12 ***
## Tau2_13_13 *
## Tau2_14_14 **
## Tau2_15_15 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Q statistic on the homogeneity of effect sizes: 2531.069
## Degrees of freedom of the Q statistic: 420
## P value of the Q statistic: 0
##
## Heterogeneity indices (based on the estimated Tau2):
##
##           Estimate
## Intercept1: I2 (Q statistic)  0.6441
## Intercept2: I2 (Q statistic)  0.8783
## Intercept3: I2 (Q statistic)  0.9107
## Intercept4: I2 (Q statistic)  0.6591
## Intercept5: I2 (Q statistic)  0.6048
## Intercept6: I2 (Q statistic)  0.8648
## Intercept7: I2 (Q statistic)  0.8118
## Intercept8: I2 (Q statistic)  0.7170
## Intercept9: I2 (Q statistic)  0.8256
## Intercept10: I2 (Q statistic) 0.8311
## Intercept11: I2 (Q statistic) 0.9030
## Intercept12: I2 (Q statistic) 0.9194
## Intercept13: I2 (Q statistic) 0.6118
## Intercept14: I2 (Q statistic) 0.7902

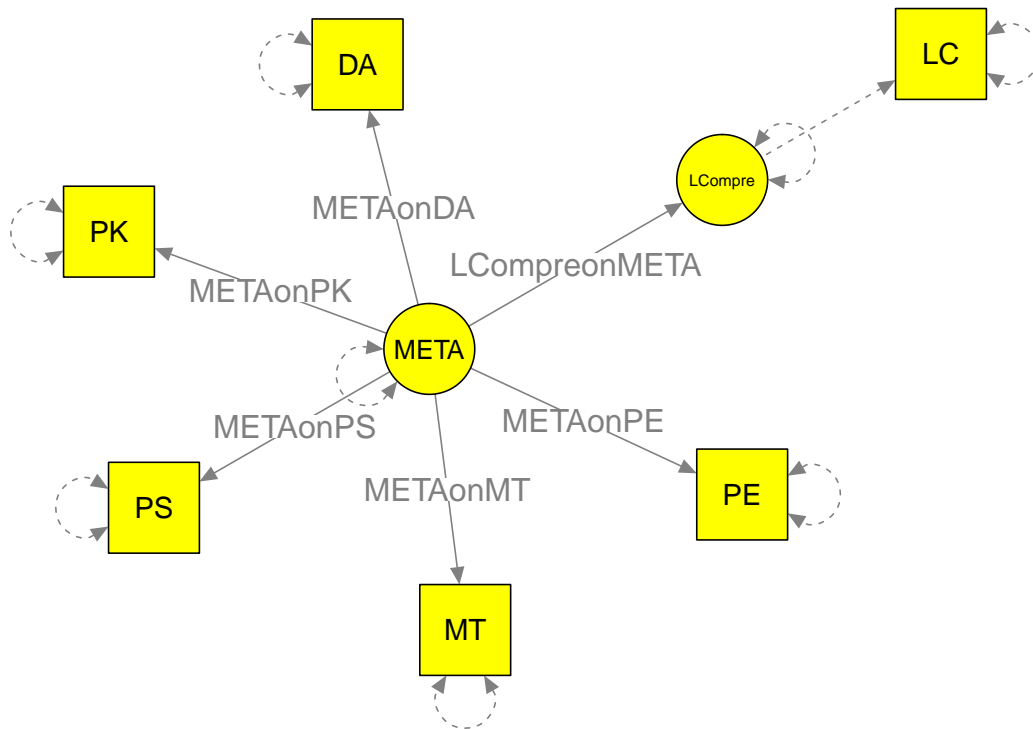
```



```
## Intercept15: I2 (Q statistic) 0.6077
##
## Number of studies (or clusters): 32
## Number of observed statistics: 435
## Number of estimated parameters: 30
## Degrees of freedom: 405
## -2 log likelihood: -90.55159
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values may indicate problems.)
```

Model with a latent variable

```
model3 <- '## Metacognitive strategy factor (META) is measured by DA, MT, PK, PE, and PS
  META =~ METAonDA*DA + METAonMT*MT + METAonPK*PK + METAonPE*PE + METAonPS*PS
  ## Listening comprehension is measured by LCompre (single-factor model)
  LCompre =~ 1*LC
  ## LC is regressed on META
  LCompre ~ LCompreonMETA*META
  LCompre ~~ 0*LCompre'
plot(model3, col="yellow", layout="spring")
```



```
RAM3 <- lavaan2RAM(model3, obs.variables=c("LC", "DA", "MT", "PK", "PE", "PS"))
RAM3
```

```
## $A
##      LC DA MT PK PE PS META      LCompre
## LC   "0" "0" "0" "0" "0" "0" "0"      "1"
## DA   "0" "0" "0" "0" "0" "0" "0*METAonDA" "0"
```

```

## MT      "0" "0" "0" "0" "0" "0" "0" "0*METAonMT"      "0"
## PK      "0" "0" "0" "0" "0" "0" "0" "0*METAonPK"      "0"
## PE      "0" "0" "0" "0" "0" "0" "0" "0*METAonPE"      "0"
## PS      "0" "0" "0" "0" "0" "0" "0" "0*METAonPS"      "0"
## META    "0" "0" "0" "0" "0" "0" "0" "0"                "0"
## LCompre "0" "0" "0" "0" "0" "0" "0" "0*LCompreonMETA" "0"
##
## $S
##      LC      DA      MT      PK      PE
## LC    "0*LCWITHLC" "0"      "0"      "0"      "0"
## DA    "0"          "0*DAWITHDA" "0"      "0"      "0"
## MT    "0"          "0"      "0*MTWITHMT" "0"      "0"
## PK    "0"          "0"      "0"      "0*PKWITHPK" "0"
## PE    "0"          "0"      "0"      "0"      "0*PEWITHPE"
## PS    "0"          "0"      "0"      "0"      "0"
## META  "0"          "0"      "0"      "0"      "0"
## LCompre "0"      "0"      "0"      "0"      "0"
##      PS      META LCompre
## LC    "0"      "0" "0"
## DA    "0"      "0" "0"
## MT    "0"      "0" "0"
## PK    "0"      "0" "0"
## PE    "0"      "0" "0"
## PS    "0*PSWITHPS" "0" "0"
## META  "0"      "1" "0"
## LCompre "0"      "0" "0"
##
## $F
##      LC DA MT PK PE PS META LCompre
## LC  1  0  0  0  0  0  0  0
## DA  0  1  0  0  0  0  0  0
## MT  0  0  1  0  0  0  0  0
## PK  0  0  0  1  0  0  0  0
## PE  0  0  0  0  1  0  0  0
## PS  0  0  0  0  0  1  0  0
##
## $M
##      LC DA MT PK PE PS META LCompre
## 1  0  0  0  0  0  0  0

```

```

stage2random_model3 <- tssem2(stage1random2, RAM=RAM3, diag.constraints=TRUE,
                              intervals.type="z")

```

```

## Warning in checkRAM(Amatrix = Amatrix, Smatrix = Smatrix, cor.analysis =
## cor.analysis): The variances of the independent variables in 'Smatrix' must be
## fixed at 1.
## Warning in checkRAM(Amatrix = Amatrix, Smatrix = Smatrix, cor.analysis =
## cor.analysis): The variances of the dependent variables in 'Smatrix' should be
## free.
summary(stage2random_model3)

```

```

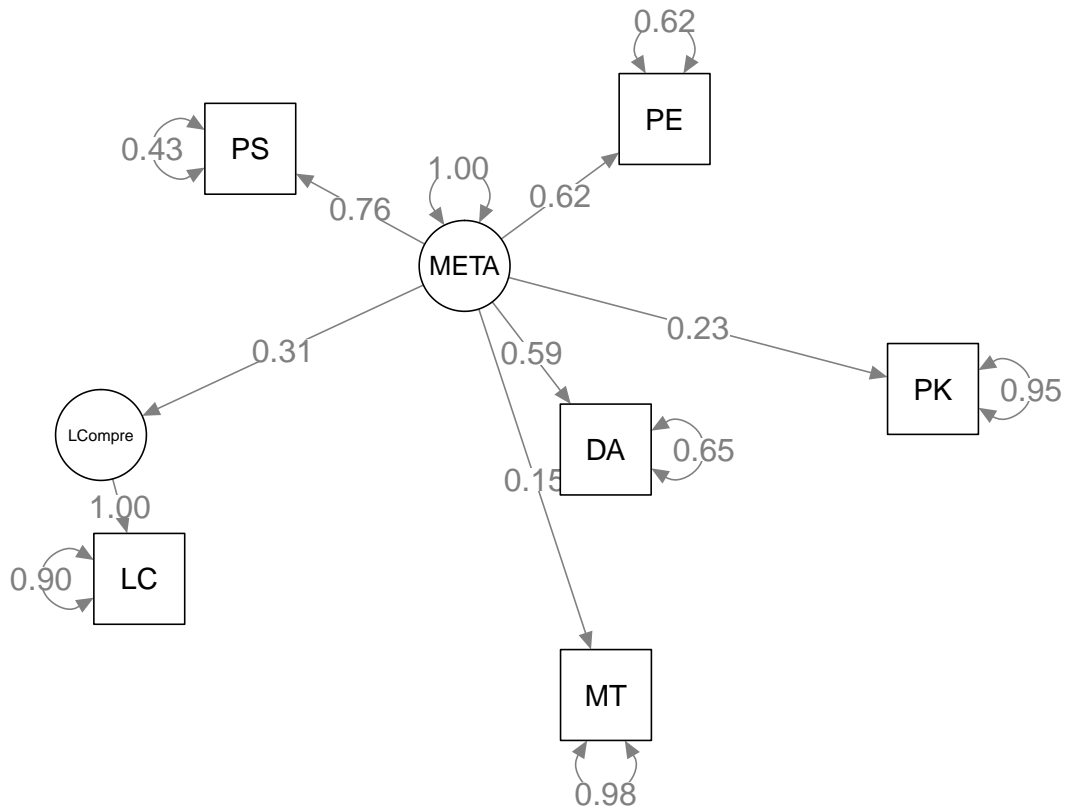
##
## Call:
## wls(Cov = pooledS, aCov = aCov, n = tssem1.obj$total.n, RAM = RAM,

```

```

##      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: z statistic approximation
## Coefficients:
##           Estimate Std.Error   lbound   ubound z value Pr(>|z|)
## METAonDA      0.591859  0.030949  0.531200  0.652518  19.1236 < 2.2e-16 ***
## LCompreonMETA 0.312846  0.030749  0.252578  0.373114  10.1740 < 2.2e-16 ***
## METAonMT       0.151865  0.045912  0.061880  0.241851   3.3078 0.0009405 ***
## METAonPE       0.616426  0.029417  0.558771  0.674082  20.9550 < 2.2e-16 ***
## METAonPK       0.232201  0.032000  0.169483  0.294919   7.2563 3.977e-13 ***
## METAonPS       0.755402  0.034211  0.688350  0.822453  22.0809 < 2.2e-16 ***
## DAWITHDA      0.649703  0.000000  0.649703  0.649703      Inf < 2.2e-16 ***
## LCWITHLC      0.902127  0.000000  0.902127  0.902127      Inf < 2.2e-16 ***
## MTWITHMT      0.976937  0.000000  0.976937  0.976937      Inf < 2.2e-16 ***
## PEWITHPE      0.620018  0.000000  0.620018  0.620018      Inf < 2.2e-16 ***
## PKWITHPK      0.946083  0.000000  0.946083  0.946083      Inf < 2.2e-16 ***
## PSWITHPS      0.429368  0.000000  0.429368  0.429368      Inf < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Goodness-of-fit indices:
##
##                               Value
## Sample size                    4621.0000
## Chi-square of target model      20.7596
## DF of target model              9.0000
## p value of target model         0.0138
## Number of constraints imposed on "Smatrix" 6.0000
## DF manually adjusted           0.0000
## Chi-square of independence model 835.5024
## DF of independence model       15.0000
## RMSEA                          0.0168
## RMSEA lower 95% CI             0.0072
## RMSEA upper 95% CI            0.0264
## SRMR                           0.0466
## TLI                            0.9761
## CFI                           0.9857
## AIC                           2.7596
## BIC                          -55.1857
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values indicate problems.)
plot(stage2random_model3, layout="spring")

```



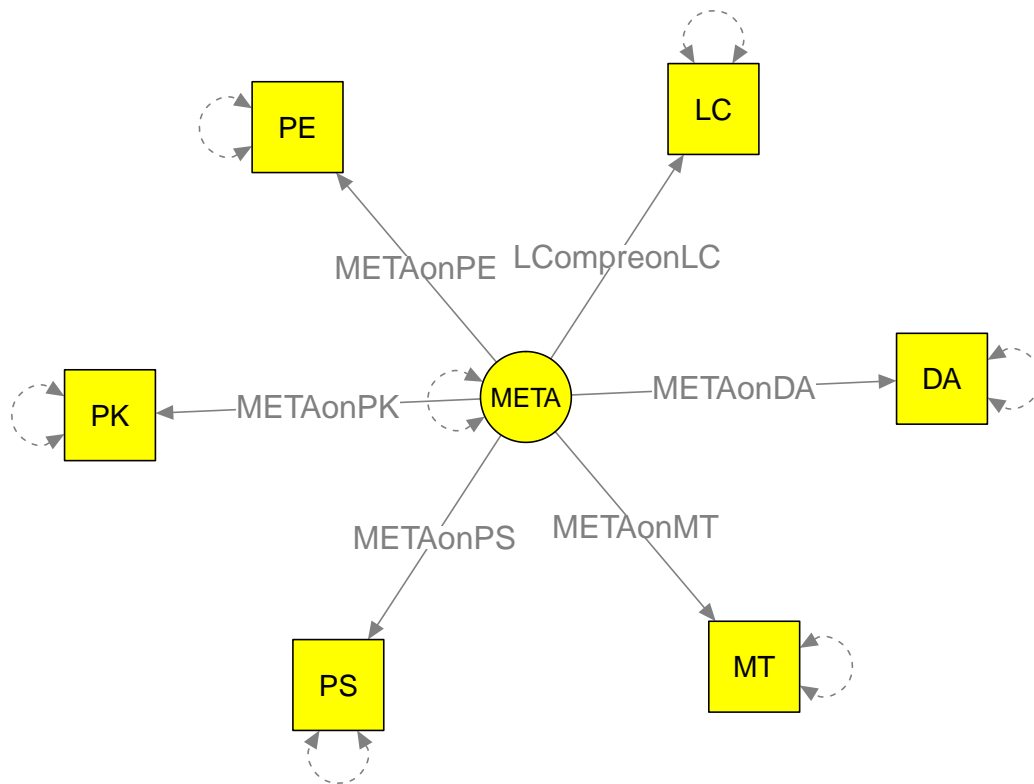
Model without any latent variable

```

model3a <- '## Metacognitive strategy factor (META) is measured by DA, MT, PK, PE, and PS
META =~ METAonDA*DA + METAonMT*MT + METAonPK*PK + METAonPE*PE +
METAonPS*PS + LCompreonLC*LC'
  
```

```

plot(model3a, col="yellow", layout="spring")
  
```



```
RAM3a <- lavaan2RAM(model3a, obs.variables=c("LC", "DA", "MT", "PK", "PE", "PS"))
RAM3a
```

```
## $A
##      LC  DA  MT  PK  PE  PS  META
## LC  "0" "0" "0" "0" "0" "0" "0*LCompreonLC"
## DA  "0" "0" "0" "0" "0" "0" "0*METAonDA"
## MT  "0" "0" "0" "0" "0" "0" "0*METAonMT"
## PK  "0" "0" "0" "0" "0" "0" "0*METAonPK"
## PE  "0" "0" "0" "0" "0" "0" "0*METAonPE"
## PS  "0" "0" "0" "0" "0" "0" "0*METAonPS"
## META "0" "0" "0" "0" "0" "0" "0"
##
## $S
##      LC      DA      MT      PK      PE
## LC  "0*LCWITHLC" "0"      "0"      "0"      "0"
## DA  "0"          "0*DAWITHDA" "0"      "0"      "0"
## MT  "0"          "0"          "0*MTWITHMT" "0"      "0"
## PK  "0"          "0"          "0"      "0*PKWITHPK" "0"
## PE  "0"          "0"          "0"      "0"      "0*PEWITHPE"
## PS  "0"          "0"          "0"      "0"      "0"
## META "0"          "0"          "0"      "0"      "0"
##      PS      META
## LC  "0"      "0"
## DA  "0"      "0"
## MT  "0"      "0"
## PK  "0"      "0"
```

```

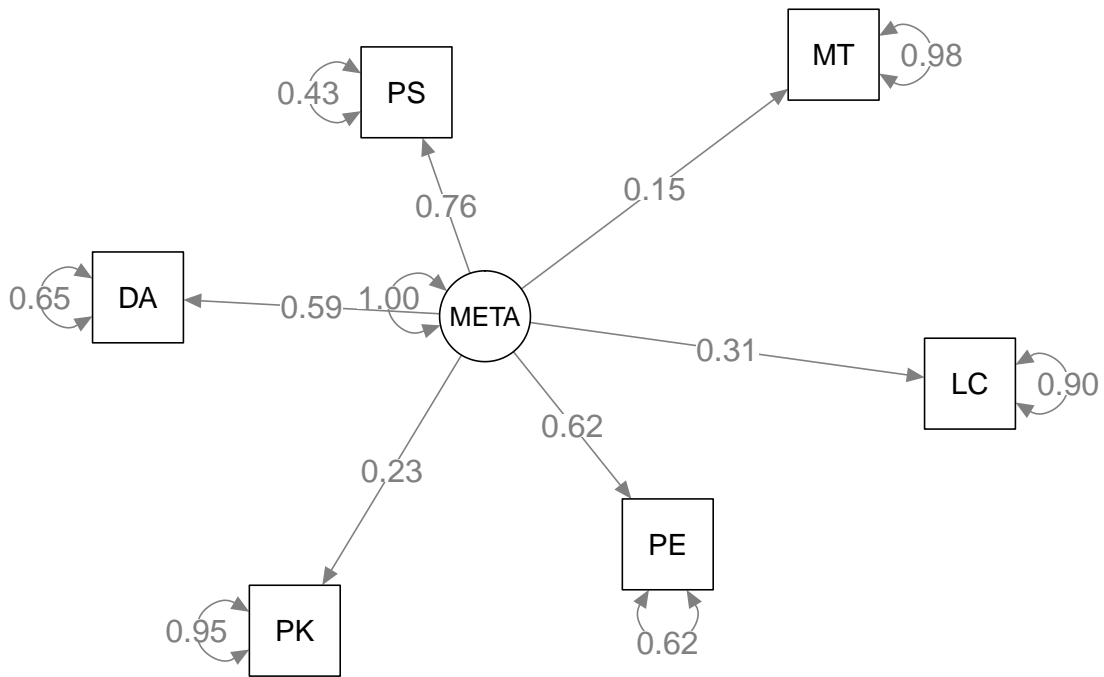
## PE "0" "0"
## PS "0*PSWITHPS" "0"
## META "0" "1"
##
## $F
## LC DA MT PK PE PS META
## LC 1 0 0 0 0 0 0
## DA 0 1 0 0 0 0 0
## MT 0 0 1 0 0 0 0
## PK 0 0 0 1 0 0 0
## PE 0 0 0 0 1 0 0
## PS 0 0 0 0 0 1 0
##
## $M
## LC DA MT PK PE PS META
## 1 0 0 0 0 0 0
stage2random_model3a <- tssem2(stage1random2, RAM=RAM3a, diag.constraints=FALSE,
                             intervals.type="z")
summary(stage2random_model3a)

##
## 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: z statistic approximation
## Coefficients:
## Estimate Std.Error lbound ubound z value Pr(>|z|)
## METAonDA 0.591859 0.030949 0.531200 0.652518 19.1236 < 2.2e-16 ***
## LCompreonLC 0.312846 0.030749 0.252578 0.373114 10.1740 < 2.2e-16 ***
## METAonMT 0.151865 0.045912 0.061880 0.241851 3.3078 0.0009405 ***
## METAonPE 0.616426 0.029417 0.558771 0.674082 20.9550 < 2.2e-16 ***
## METAonPK 0.232201 0.032000 0.169483 0.294919 7.2563 3.977e-13 ***
## METAonPS 0.755402 0.034211 0.688350 0.822453 22.0809 < 2.2e-16 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Goodness-of-fit indices:
## Value
## Sample size 4621.0000
## Chi-square of target model 20.7596
## DF of target model 9.0000
## p value of target model 0.0138
## Number of constraints imposed on "Smatrix" 0.0000
## DF manually adjusted 0.0000
## Chi-square of independence model 835.5024
## DF of independence model 15.0000
## RMSEA 0.0168
## RMSEA lower 95% CI 0.0072
## RMSEA upper 95% CI 0.0264

```

```
## SRMR                0.0466
## TLI                 0.9761
## CFI                 0.9857
## AIC                 2.7596
## BIC                 -55.1857
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values indicate problems.)
```

```
plot(stage2random_model3a, layout="spring")
```



```
sessionInfo()
```

```
## R version 4.0.3 (2020-10-10)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 20.04.2 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_1.2.5.1 OpenMx_2.19.6
##
## loaded via a namespace (and not attached):
## [1] nlme_3.1-152      RColorBrewer_1.1-2  mi_1.0
## [4] tools_4.0.3      backports_1.2.1    utf8_1.2.1
## [7] R6_2.5.0         rpart_4.1-15       Hmisc_4.5-0
## [10] DBI_1.1.1        colorspace_2.0-0   nnet_7.3-14
## [13] tidymodels_1.1.0  gridExtra_2.3      mnormt_2.0.2
## [16] compiler_4.0.3    fdrtool_1.2.16     qgraph_1.6.9
## [19] htmlTable_2.1.0   regsem_1.8.0       scales_1.1.1
## [22] checkmate_2.0.0   psych_2.0.12       mvtnorm_1.1-1
## [25] pbapply_1.4-3     sem_3.1-11         stringr_1.4.0
## [28] digest_0.6.27     pbivnorm_0.6.0     foreign_0.8-81
## [31] minqa_1.2.4       rmarkdown_2.8      base64enc_0.1-3
## [34] jpeg_0.1-8.1     pkgconfig_2.0.3    htmltools_0.5.1.1
## [37] lme4_1.1-26      lisrelToR_0.1.4    htmlwidgets_1.5.3
## [40] rlang_0.4.10     rstudioapi_0.13    generics_0.1.0
## [43] gtools_3.8.2     dplyr_1.0.5        zip_2.1.1
## [46] magrittr_2.0.1   Formula_1.2-4      Matrix_1.2-18
## [49] Rcpp_1.0.6       munsell_0.5.0      fansi_0.4.2
## [52] abind_1.4-5      rockchalk_1.8.144  lifecycle_1.0.0
## [55] stringi_1.5.3    yaml_2.2.1         carData_3.0-4
## [58] MASS_7.3-53      plyr_1.8.6         matrixcalc_1.0-4
## [61] lavaan_0.6-9     grid_4.0.3         parallel_4.0.3
## [64] crayon_1.4.1     lattice_0.20-41    semPlot_1.1.2
## [67] kutils_1.70      splines_4.0.3      tmvnsim_1.0-2
## [70] knitr_1.31       pillar_1.5.1       igraph_1.2.6
## [73] boot_1.3-25      corpcor_1.6.9     reshape2_1.4.4
## [76] stats4_4.0.3     XML_3.99-0.6       glue_1.4.2
## [79] evaluate_0.14    latticeExtra_0.6-29 data.table_1.14.0
## [82] RcppParallel_5.0.3 png_0.1-7          vctrs_0.3.6
## [85] nloptr_1.2.2.2    gtable_0.3.0      purrr_0.3.4
## [88] assertthat_0.2.1 ggplot2_3.3.3     xfun_0.23
## [91] openxlsx_4.2.3   xtable_1.8-4      Rsolnp_1.16
## [94] coda_0.19-4     glasso_1.11       survival_3.2-10
## [97] truncnorm_1.0-8  tibble_3.1.0      arm_1.11-2
## [100] ellipse_0.4.2    cluster_2.1.0     statmod_1.4.35
## [103] ellipsis_0.3.1

```