**Robust Evidence for the Simple View of Second Language Reading:**

**Secondary Meta-Analysis of Jeon and Yamashita (2022)**

**Supplementary Material 3: Statistical Outputs**

## R Packages

library(meta)

library(metafor)

library(effectsize)

library(metaSEM)

library(openxlsx)
library(lavaan)

library(psych)

options(digits = 6)

# Replication

## Data Frame

dataset <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 1) # read sheet No.1
head(dataset)

## ES Calculation

dL2D <- na.omit(dataset[, c(1, 2, 3)]) # L2 Decoding
L2D <- metacor(cor = L2R\_L2D,
 n = N,
 studlab = Study,
 data = dL2D,
 sm = "COR",
 fixed = FALSE,
 random = TRUE,
 method.tau = "REML",
 hakn = TRUE)

dL2V <- na.omit(dataset[, c(1, 2, 4)]) # L2 Vocabulary Knowledge
L2V <- metacor(cor = L2R\_L2V,
 n = N,
 studlab = Study,
 data = dL2V,
 sm = "COR",
 fixed = FALSE,
 random = TRUE,
 method.tau = "REML",
 hakn = TRUE)

dL2G <- na.omit(dataset[, c(1, 2, 5)]) # L2 Grammar Knowledge
L2G <- metacor(cor = L2R\_L2G,
 n = N,
 studlab = Study,
 data = dL2G,
 sm = "COR",
 fixed = FALSE,
 random = TRUE,
 method.tau = "REML",
 hakn = TRUE)

dL1R <- na.omit(dataset[, c(1, 2, 6)]) # L1 Reading Comprehension
L1R <- metacor(cor = L2R\_L1R,
 n = N,
 studlab = Study,
 data = dL1R,
 sm = "COR",
 fixed = FALSE,
 random = TRUE,
 method.tau = "REML",
 hakn = TRUE)

dL2P <- na.omit(dataset[, c(1, 2, 7)]) # L2 Phonological Awareness
L2P <- metacor(cor = L2R\_L2P,
 n = N,
 studlab = Study,
 data = dL2P,
 sm = "COR",
 fixed = FALSE,
 random = TRUE,
 method.tau = "REML",
 hakn = TRUE)

dL2O <- na.omit(dataset[, c(1, 2, 8)]) # L2 Orthographic Knowledge
L2O <- metacor(cor = L2R\_L2O,
 n = N,
 studlab = Study,
 data = dL2O,
 sm = "COR",
 fixed = FALSE,
 random = TRUE,
 method.tau = "REML",
 hakn = TRUE)

dL2M <- na.omit(dataset[, c(1, 2, 9)]) # L2 Morphological Knowledge
L2M <- metacor(cor = L2R\_L2M,
 n = N,
 studlab = Study,
 data = dL2M,
 sm = "COR",
 fixed = FALSE,
 random = TRUE,
 method.tau = "REML",
 hakn = TRUE)

dL2L <- na.omit(dataset[, c(1, 2, 10)]) # L2 Listening Comprehension
L2L <- metacor(cor = L2R\_L2L,
 n = N,
 studlab = Study,
 data = dL2L,
 sm = "COR",
 fixed = FALSE,
 random = TRUE,
 method.tau = "REML",
 hakn = TRUE)

dWM <- na.omit(dataset[, c(1, 2, 11)]) # Working Memory
WM <- metacor(cor = L2R\_WM,
 n = N,
 studlab = Study,
 data = dWM,
 sm = "COR",
 fixed = FALSE,
 random = TRUE,
 method.tau = "REML",
 hakn = TRUE)

dMet <- na.omit(dataset[, c(1, 2, 12)]) # Metacognition
Met <- metacor(cor = L2R\_Meta,
 n = N,
 studlab = Study,
 data = dMet,
 sm = "COR",
 fixed = FALSE,
 random = TRUE,
 method.tau = "REML",
 hakn = TRUE)

# Table 1
t1 <- data.frame(k = c(L2D$k.study, L2P$k.study, L2O$k.study, L2M$k.study, L2V$k.study,
 L2G$k.study, L1R$k.study, L2L$k.study, WM$k.study, Met$k.study),
 ES = c(L2D$TE.random, L2P$TE.random, L2O$TE.random, L2M$TE.random, L2V$TE.random,
 L2G$TE.random, L1R$TE.random, L2L$TE.random, WM$TE.random, Met$TE.random),
 CI\_Lower = c(L2D$lower.random, L2P$lower.random, L2O$lower.random, L2M$lower.random, L2V$lower.random,
 L2G$lower.random, L1R$lower.random, L2L$lower.random, WM$lower.random, Met$lower.random),
 CI\_upper = c(L2D$upper.random, L2P$upper.random, L2O$upper.random, L2M$upper.random, L2V$upper.random,
 L2G$upper.random, L1R$upper.random, L2L$upper.random, WM$upper.random, Met$upper.random))
rownames(t1) <- c("L2D", "L2P", "L2O", "L2M", "L2V", "L2G", "L1R", "L2L", "WM", "Meta")
print(t1)

## k ES CI\_Lower CI\_upper
## L2D 37 0.571699 0.502751 0.640648
## L2P 18 0.475642 0.373064 0.578220
## L2O 6 0.566343 0.326957 0.805728
## L2M 10 0.527974 0.408676 0.647273
## L2V 58 0.574154 0.525308 0.623000
## L2G 33 0.596792 0.512247 0.681337
## L1R 38 0.388494 0.314981 0.462006
## L2L 20 0.619393 0.526499 0.712288
## WM 20 0.310927 0.201837 0.420018
## Meta 10 0.314751 0.101326 0.528176

## ES Comparisons

esL2D <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 2, cols = c(1, 2))
esL2V <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 2, cols = c(3, 4))
esL2G <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 2, cols = c(5, 6))
esL1R <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 2, cols = c(7, 8))
esL2P <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 2, cols = c(9, 10))
esL2O <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 2, cols = c(11, 12))
esL2M <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 2, cols = c(13, 14))
esL2L <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 2, cols = c(15, 16))
esWM <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 2, cols = c(17, 18))
esMeta <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 2, cols = c(19, 20))

esL2D.r <- t.test(ES ~ L2D, data = esL2D, paired = F)
esL2V.r <- t.test(ES ~ L2V, data = esL2V, paired = F)
esL2G.r <- t.test(ES ~ L2G, data = esL2G, paired = F)
esL1R.r <- t.test(ES ~ L1R, data = esL1R, paired = F)
esL2P.r <- t.test(ES ~ L2P, data = esL2P, paired = F)
esL2O.r <- t.test(ES ~ L2O, data = esL2O, paired = F)
esL2M.r <- t.test(ES ~ L2M, data = esL2M, paired = F)
esL2L.r <- t.test(ES ~ L2L, data = esL2L, paired = F)
esWM.r <- t.test(ES ~ WM, data = esWM, paired = F)
esMeta.r <- t.test(ES ~ Meta, data = esMeta, paired = F)

# Table 1 (Continue)
t1 <- data.frame(t = c(esL2D.r$statistic, esL2P.r$statistic, esL2O.r$statistic, esL2M.r$statistic, esL2V.r$statistic,
 esL2G.r$statistic, esL1R.r$statistic, esL2L.r$statistic, esWM.r$statistic, esMeta.r$statistic),
 df = c(esL2D.r$parameter, esL2P.r$parameter, esL2O.r$parameter, esL2M.r$parameter, esL2V.r$parameter,
 esL2G.r$parameter, esL1R.r$parameter, esL2L.r$parameter, esWM.r$parameter, esMeta.r$parameter),
 p = c(esL2D.r$p.value, esL2P.r$p.value, esL2O.r$p.value, esL2M.r$p.value, esL2V.r$p.value,
 esL2G.r$p.value, esL1R.r$p.value, esL2L.r$p.value, esWM.r$p.value, esMeta.r$p.value),
 d = c(cohens\_d(ES ~ L2D, data = esL2D)$Cohens\_d, cohens\_d(ES ~ L2P, data = esL2P)$Cohens\_d,

cohens\_d(ES ~ L2O, data = esL2O)$Cohens\_d, cohens\_d(ES ~ L2M, data = esL2M)$Cohens\_d,

cohens\_d(ES ~ L2V, data = esL2V)$Cohens\_d, cohens\_d(ES ~ L2G, data = esL2G)$Cohens\_d,

cohens\_d(ES ~ L1R, data = esL1R)$Cohens\_d, cohens\_d(ES ~ L2L, data = esL2L)$Cohens\_d,

cohens\_d(ES ~ WM, data = esWM)$Cohens\_d, cohens\_d(ES ~ Meta, data = esMeta)$Cohens\_d))
rownames(t1) <- c("L2D", "L2P", "L2O", "L2M", "L2V", "L2G", "L1R", "L2L", "WM", "Meta")
print(t1, digits = 2)

## t df p d
## L2D -0.289 59 0.7738 -0.0717
## L2P -2.241 37 0.0311 -0.7168
## L2O -0.380 10 0.7116 -0.2196
## L2M -2.277 20 0.0341 -0.9422
## L2V -3.319 95 0.0013 -0.6450
## L2G -1.025 48 0.3104 -0.2752
## L1R -2.104 68 0.0391 -0.4985
## L2L -1.742 34 0.0905 -0.5509
## WM -0.964 37 0.3411 -0.3087
## Meta 0.011 19 0.9914 0.0047

## Forest Plots (see Supplemental Material 5)

# L2 Decoding
pL2D <- forest(L2D,
 slab = paste(dataset$Study),
 xlab = "Effect size [r]",
 mlab = "",
 pch = 18,
 cex = 0.5,
 sortvar = L2D$TE)

# L2 Phonological Awareness
pL2P <- forest(L2P,
 slab = paste(dataset$Study),
 xlab = "Effect size [r]",
 mlab = "",
 pch = 18,
 cex = 0.5,
 sortvar = L2P$cor)

# L2 Orthographic Knowledge
pL2O <- forest(L2O,
 slab = paste(dataset$Study),
 xlab = "Effect size [r]",
 mlab = "",
 pch = 18,
 cex = 0.5,
 sortvar = L2O$cor)

# L2 Morphological Knowledge
pL2M <- forest(L2M,
 slab = paste(dataset$Study),
 xlab = "Effect size [r]",
 mlab = "",
 pch = 18,
 cex = 0.5,
 sortvar = L2M$cor)

# L2 Vocabulary Knowledge
pL2V <- forest(L2V,
 slab = paste(dataset$Study),
 xlab = "Effect size [r]",
 mlab = "",
 pch = 18,
 cex = 0.5,
 sortvar = L2V$cor)

# L2 Grammar Knowledge
pL2G <- forest(L2G,
 slab = paste(dataset$Study),
 xlab = "Effect size [r]",
 mlab = "",
 pch = 18,
 cex = 0.5,
 sortvar = L2G$cor)

# L1 Reading Comprehension
pL1R <- forest(L1R,
 slab = paste(dataset$Study),
 xlab = "Effect size [r]",
 mlab = "",
 pch = 18,
 cex = 0.5,
 sortvar = L1R$cor)

# L2 Listening Comprehension
pL2L <- forest(L2L,
 slab = paste(dataset$Study),
 xlab = "Effect size [r]",
 mlab = "",
 pch = 18,
 cex = 0.5,
 sortvar = L2L$cor)

# Working Memory
pWM <- forest(WM,
 slab = paste(dataset$Study),
 xlab = "Effect size [r]",
 mlab = "",
 pch = 18,
 cex = 0.5,
 sortvar = WM$cor)

# Metacognition
pMet <- forest(Met,
 slab = paste(dataset$Study),
 xlab = "Effect size [r]",
 mlab = "",
 pch = 18,
 cex = 0.5,
 sortvar = Met$cor)

# Secondary Analysis

## MASEM

dataset2 <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 3) # read sheet No.3
head(dataset2)

# make list of correlation matrices (cordat), NA on diagonal
nvar2 <- 28
varnames2 <- c("L2R", "L2DA", "L2DF", "L2V", "L2G", "L2P", "L2M", "L2L")
labels2 <- list(varnames2, varnames2)
cordat2 <- list()
for (i in 1:nrow(dataset2)){
 cordat2[[i]] <- vec2symMat(as.matrix(dataset2[i, 3:30]), diag = FALSE)
 dimnames(cordat2[[i]]) <- labels2
}
head(cordat2)

# Table 3
pattern.na(cordat2, show.na = FALSE)

## L2R L2DA L2DF L2V L2G L2P L2M L2L
## L2R 78 30 19 60 33 20 11 21
## L2DA 30 78 9 23 6 16 9 8
## L2DF 19 9 78 12 6 5 3 4
## L2V 60 23 12 78 29 12 9 16
## L2G 33 6 6 29 78 3 3 12
## L2P 20 16 5 12 3 78 5 3
## L2M 11 9 3 9 3 5 78 3
## L2L 21 8 4 16 12 3 3 78

### MASEM Stage 1

# Create Pooled Correlations
S1 <- tssem1(Cov = cordat2, n = dataset2$N, method = "REM", RE.type = "Diag", acov = "weighted")
S1 <- rerun(S1, autofixtau2 = TRUE)

# Extract the fixed coefficients (correlations)
REs1 <- coef(S1, "fixed")
REs1.mat <- vec2symMat(REs1, diag = FALSE)

# Table 3
dimnames(REs1.mat)[[1]] <- c("L2R", "L2DA", "L2DF", "L2V", "L2G", "L2P", "L2M", "L2L")
dimnames(REs1.mat)[[2]] <- c("L2R", "L2DA", "L2DF", "L2V", "L2G", "L2P", "L2M", "L2L")
round(REs1.mat, 2)

## L2R L2DA L2DF L2V L2G L2P L2M L2L
## L2R 1.00 0.56 0.46 0.51 0.52 0.46 0.49 0.56
## L2DA 0.56 1.00 0.63 0.49 0.56 0.53 0.56 0.42
## L2DF 0.46 0.63 1.00 0.36 0.35 0.42 0.34 0.30
## L2V 0.51 0.49 0.36 1.00 0.53 0.43 0.53 0.52
## L2G 0.52 0.56 0.35 0.53 1.00 0.39 0.58 0.48
## L2P 0.46 0.53 0.42 0.43 0.39 1.00 0.46 0.37
## L2M 0.49 0.56 0.34 0.53 0.58 0.46 1.00 0.52
## L2L 0.56 0.42 0.30 0.52 0.48 0.37 0.52 1.00

### MASEM Stage 2

## Specify the model
# Model A
ModelA <-
 'Comprehension =~ L2L + L2V + L2G
 Decoding =~ L2DA + L2DF
 Meta =~ L2P + L2M
 L2R ~ Comprehension + Decoding
 Comprehension ~~ Meta
 Decoding ~ Meta
 Comprehension ~~ Decoding'
RAM1 <- lavaan2RAM(ModelA, obs.variables = c("L2R", "L2DA", "L2DF", "L2V", "L2G", "L2P", "L2M", "L2L"),
 A.notation = "on", S.notation = "with")

# Model B
ModelB <-
 'Comprehension =~ L2L + L2V + L2G
 Decoding =~ L2DA + L2DF
 Meta =~ L2P + L2M
 L2R ~ Comprehension + Decoding + Meta
 Comprehension ~~ Meta
 Decoding ~ Meta
 Comprehension ~~ Decoding'
RAM2 <- lavaan2RAM(ModelB, obs.variables = c("L2R", "L2DA", "L2DF", "L2V", "L2G", "L2P", "L2M", "L2L"),
 A.notation = "on", S.notation = "with")

# Model C
ModelC <- # Lee et al.'s model
 'Comprehension =~ L2L + L2V + L2G
 Decoding =~ L2DA + L2DF
 Meta =~ L2P + L2M
 L2R ~ Comprehension + Decoding
 Comprehension ~~ 0 \* Meta
 Decoding ~ 0 \* Meta
 Comprehension ~~ Decoding'
RAM3 <- lavaan2RAM(ModelC, obs.variables = c("L2R", "L2DA", "L2DF", "L2V", "L2G", "L2P", "L2M", "L2L"),
 A.notation = "on", S.notation = "with")

## Model fitting
REA <- tssem2(S1,
 RAM = RAM1,
 intervals = "z",
 diag.constraints = FALSE,
 model.name = "Model A")

REA <- rerun(REA, autofixtau2 = TRUE)

REB <- tssem2(S1,
 RAM = RAM2,
 intervals = "z",
 diag.constraints = FALSE,
 model.name = "Model B")

REB <- rerun(REB, autofixtau2 = TRUE, extraTries = 100)

REC <- tssem2(S1,
 RAM = RAM3,
 intervals = "z",
 diag.constraints = FALSE,
 model.name = "Model C")
REC <- rerun(REC, autofixtau2 = TRUE)

## Model Comparison
anova(REB, REA)

## base comparison ep minus2LL df AIC diffLL diffdf p

## 1 Model B <NA> 13 19.6398 -13 45.6398 NA NA NA

## 2 Model B Model A 12 22.1492 -12 46.1492 2.5094 1 0.113169

anova(REB, REC)

## base comparison ep minus2LL df AIC diffLL diffdf p

## 1 Model B <NA> 13 19.6398 -13 45.6398 NA NA NA

## 2 Model B Model C 10 1185.9485 -10 1205.9485 1166.31 3 1.49625e-252

anova(REA, REC)

## base comparison ep minus2LL df AIC diffLL diffdf p
## 1 Model A <NA> 12 22.1492 -12 46.1492 NA NA NA
## 2 Model A Model C 10 1185.9485 -10 1205.9485 1163.8 2 1.92396e-253

## Model Statistics
summary(REA)

##
## 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,
## mxModel.Args = mxModel.Args, subset.variables = subset.variables,
## 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|)
## DecodingonMeta 0.86447 0.05058 0.76534 0.96359 17.092 < 2e-16\*\*\*
## L2DAonDecoding 0.92904 0.03032 0.86961 0.98848 30.637 < 2e-16\*\*\*
## L2DFonDecoding 0.67669 0.02620 0.62534 0.72804 25.829 < 2e-16\*\*\*
## L2GonComprehension 0.75659 0.02319 0.71113 0.80205 32.620 < 2e-16\*\*\*
## L2LonComprehension 0.68908 0.02668 0.63680 0.74137 25.830 < 2e-16\*\*\*
## L2MonMeta 0.73679 0.03384 0.67047 0.80310 21.776 < 2e-16\*\*\*
## L2PonMeta 0.65714 0.03787 0.58292 0.73136 17.353 < 2e-16\*\*\*
## L2RonComprehension 0.58290 0.07755 0.43090 0.73490 7.516 5.64e-14\*\*\*
## L2RonDecoding 0.17810 0.08347 0.01450 0.34170 2.134 0.0329\*
## L2VonComprehension 0.71598 0.02285 0.67120 0.76076 31.339 < 2e-16\*\*\*
## ComprehensionwithDecoding -0.11084 0.07270 -0.25333 0.03165 -1.525 0.1274
## ComprehensionwithMeta 0.98930 0.04703 0.89712 1.08148 21.035 < 2e-16\*\*\*
## ---
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## Goodness-of-fit indices:
## Value
## Sample size 12062.000
## Chi-square of target model 22.149
## DF of target model 16.000
## p value of target model 0.138
## Number of constraints imposed on "Smatrix" 0.000
## DF manually adjusted 0.000
## Chi-square of independence model 3347.084
## DF of independence model 28.000
## RMSEA 0.006
## RMSEA lower 95% CI 0.000
## RMSEA upper 95% CI 0.011
## SRMR 0.043
## TLI 0.997
## CFI 0.998
## AIC -9.851
## BIC -128.216
## OpenMx status1: 0 ("0" or "1": The optimization is considered fine.
## Other values indicate problems.)

## Figure 3
plot(REA)

### Publication Bias (see Supplemental Material 4)

# Correlations
cor <- data.frame(dataset$Datapoints, dataset$N, dataset$IF)
colnames(cor) <- c("Datapoints", "N", "IF")
cor.test(cor$Datapoints, cor$IF)

##

## Pearson's product-moment correlation
##
## data: cor$Datapoints and cor$IF
## t = -0.8192, df = 77, p-value = 0.415
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.307743 0.130842
## sample estimates:
## cor
## -0.0929574

cor.test(cor$N, cor$IF)

##
## Pearson's product-moment correlation
##
## data: cor$N and cor$IF
## t = 0.7837, df = 93, p-value = 0.435
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.122547 0.278001
## sample estimates:
## cor
## 0.0809961

# Egger's Test
x <- coef(S1)
se <- sqrt(sqrt(diag(vcov(S1))))
CSE <- data.frame(Correl = x, SE = se)

PB <- rma.uni(yi = CSE[1:28, 1], sei = CSE[1:28, 2], method = "REML")
regtest(PB)

##
## Regression Test for Funnel Plot Asymmetry
##
## Model: mixed-effects meta-regression model
## Predictor: standard error
##
## Test for Funnel Plot Asymmetry: z = -0.8899, p = 0.3735
## Limit Estimate (as sei -> 0): b = 0.6831 (CI: 0.2484, 1.1179)

funnel(PB)

# Moderator Analysis
dataset2\_hi <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 4) # read sheet No.4
dataset2\_li <- read.xlsx("SSLA\_MASEM\_L2\_Reading.xlsx", 5) # read sheet No.5

describe(dataset2\_hi$IF)

## vars n mean sd median trimmed mad min max range skew kurtosis se
## X1 1 39 4.79 0.47 4.9 4.75 0 4.1 6 1.9 0.73 1.14 0.08

describe(dataset2\_li$IF)

## vars n mean sd median trimmed mad min max range skew kurtosis se
## X1 1 40 2.27 0.92 2.5 2.34 0.74 0.27 3.6 3.33 -0.57 -0.36 0.14

cordat\_hi <- list()
for (i in 1:nrow(dataset2\_hi)){
 cordat\_hi[[i]] <- vec2symMat(as.matrix(dataset2\_hi[i, 3:30]), diag = FALSE)
 dimnames(cordat\_hi[[i]]) <- labels2
}
head(cordat\_hi)

cordat\_li <- list()
for (i in 1:nrow(dataset2\_li)){
 cordat\_li[[i]] <- vec2symMat(as.matrix(dataset2\_li[i, 3:30]), diag = FALSE)
 dimnames(cordat\_li[[i]]) <- labels2
}
head(cordat\_li)

S1hi <- tssem1(Cov = cordat\_hi, n = dataset2\_hi$N, method = "REM", RE.type = "Diag", acov = "weighted")
S1hi <- rerun(S1hi, autofixtau2 = TRUE)

S1li <- tssem1(Cov = cordat\_li, n = dataset2\_li$N, method = "REM", RE.type = "Diag", acov = "weighted")
S1li <- rerun(S1li, autofixtau2 = TRUE)

REs\_hi <- coef(S1hi, "fixed")
REs\_hi.mat <- vec2symMat(REs\_hi, diag = FALSE)
dimnames(REs\_hi.mat)[[1]] <- c("L2R", "L2DA", "L2DF", "L2V", "L2G", "L2P", "L2M", "L2L")
dimnames(REs\_hi.mat)[[2]] <- c("L2R", "L2DA", "L2DF", "L2V", "L2G", "L2P", "L2M", "L2L")
round(REs\_hi.mat, 2)

## L2R L2DA L2DF L2V L2G L2P L2M L2L
## L2R 1.00 0.51 0.57 0.63 0.37 0.56 0.59 0.56
## L2DA 0.51 1.00 0.69 0.18 0.35 0.46 0.58 0.34
## L2DF 0.57 0.69 1.00 0.53 0.54 0.50 0.38 0.41
## L2V 0.63 0.18 0.53 1.00 0.25 0.59 0.12 0.49
## L2G 0.37 0.35 0.54 0.25 1.00 0.56 0.65 0.59
## L2P 0.56 0.46 0.50 0.59 0.56 1.00 0.29 0.28
## L2M 0.59 0.58 0.38 0.12 0.65 0.29 1.00 0.28
## L2L 0.56 0.34 0.41 0.49 0.59 0.28 0.28 1.00

REs\_li <- coef(S1li, "fixed")
REs\_li.mat <- vec2symMat(REs\_li, diag = FALSE)
dimnames(REs\_li.mat)[[1]] <- c("L2R", "L2DA", "L2DF", "L2V", "L2G", "L2P", "L2M", "L2L")
dimnames(REs\_li.mat)[[2]] <- c("L2R", "L2DA", "L2DF", "L2V", "L2G", "L2P", "L2M", "L2L")
round(REs\_li.mat, 2)

## L2R L2DA L2DF L2V L2G L2P L2M L2L
## L2R 1.00 0.59 0.54 0.49 0.36 0.42 0.48 0.50
## L2DA 0.59 1.00 0.49 0.42 0.12 0.51 0.56 0.14
## L2DF 0.54 0.49 1.00 0.54 0.43 0.54 0.46 0.33
## L2V 0.49 0.42 0.54 1.00 0.18 0.59 0.22 0.41
## L2G 0.36 0.12 0.43 0.18 1.00 0.50 0.54 0.53
## L2P 0.42 0.51 0.54 0.59 0.50 1.00 0.31 0.00
## L2M 0.48 0.56 0.46 0.22 0.54 0.31 1.00 0.17
## L2L 0.50 0.14 0.33 0.41 0.53 0.00 0.17 1.00