Multilevel SEM Using the `lavaan` Package

2022 International Conference on Multilevel Analysis

Terrence D. Jorgensen

(Un)Structured Models for (Un)Clustered Data

Isn't Everything Regression?

The general(ized) linear model (GLM) assumes

1. predictors are measured without error
2. observations are independent

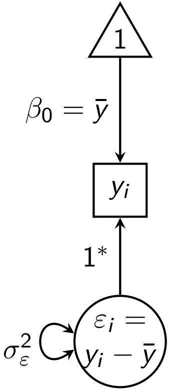
- other assumptions vary across links/distributions

Assumptions can be relaxed with further generalizations

1. Structural equation modeling (SEM) developed to accommodate measurement error
2. Multilevel modeling (MLM) developed to accommodate systematic violations of independence

Multilevel SEM (MLSEM or MSEM) is a unified generalization

Unclustered Univariate Data

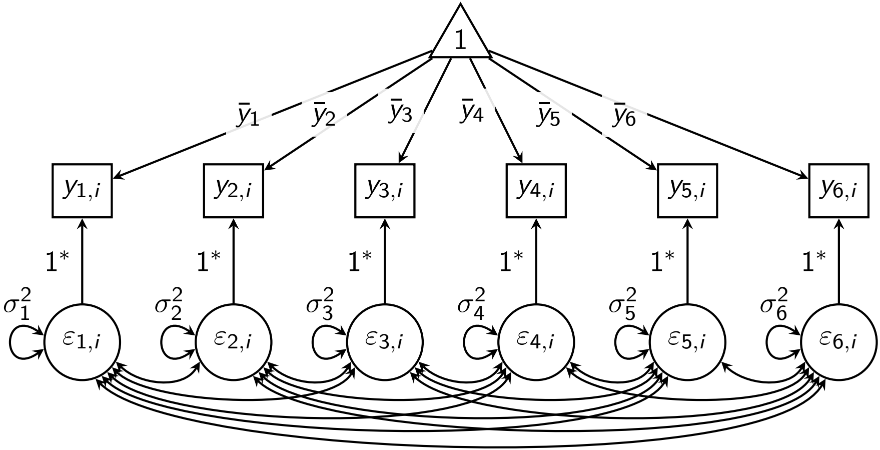


$$y_i = \bar{y} + (y_i - \bar{y})$$

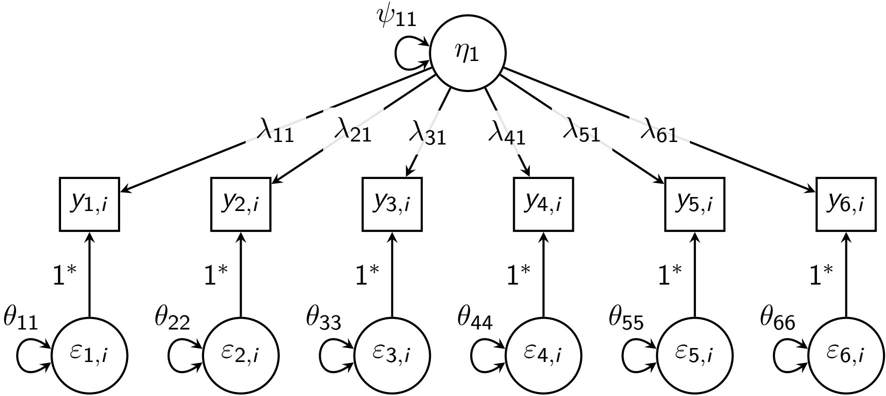
GLM: $y_{ij} = \beta_0 + \varepsilon_i$

$$\varepsilon_i \sim \mathcal{N}(0, \sigma_\varepsilon)$$

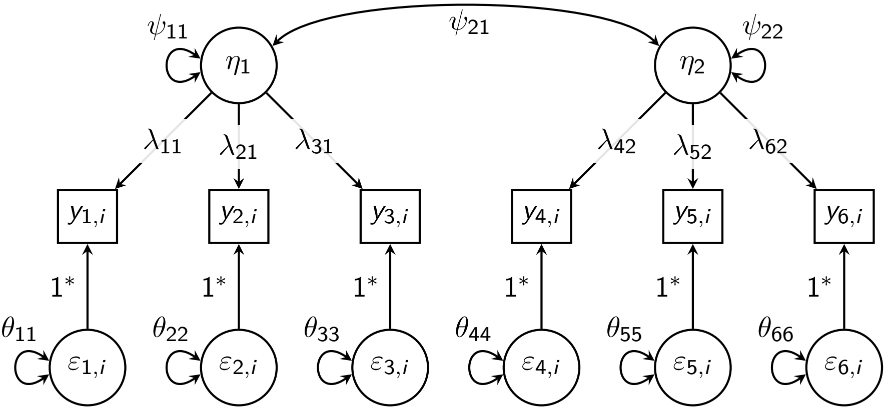
Unclustered Multivariate Data



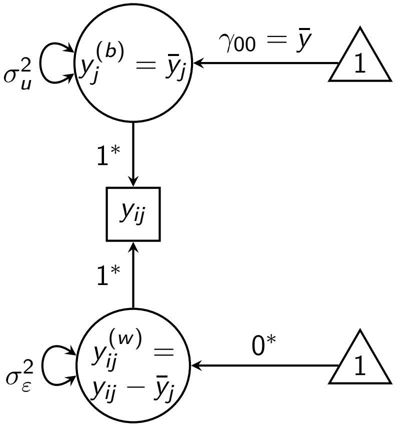
Unclustered Multivariate Data: 1-Factor Structure



Unclustered Multivariate Data: 2-Factor Structure



Clustered Univariate Data



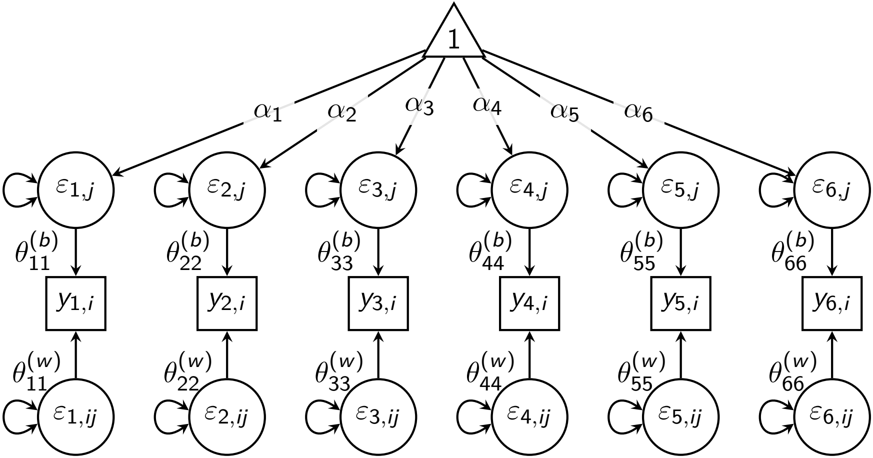
$$y_{ij} = \bar{y} + (\bar{y}_j - \bar{y}) + (y_{ij} - \bar{y}_j)$$

MLM: $y_{ij} = \gamma_{00} + u_{0j} + \varepsilon_{ij}$

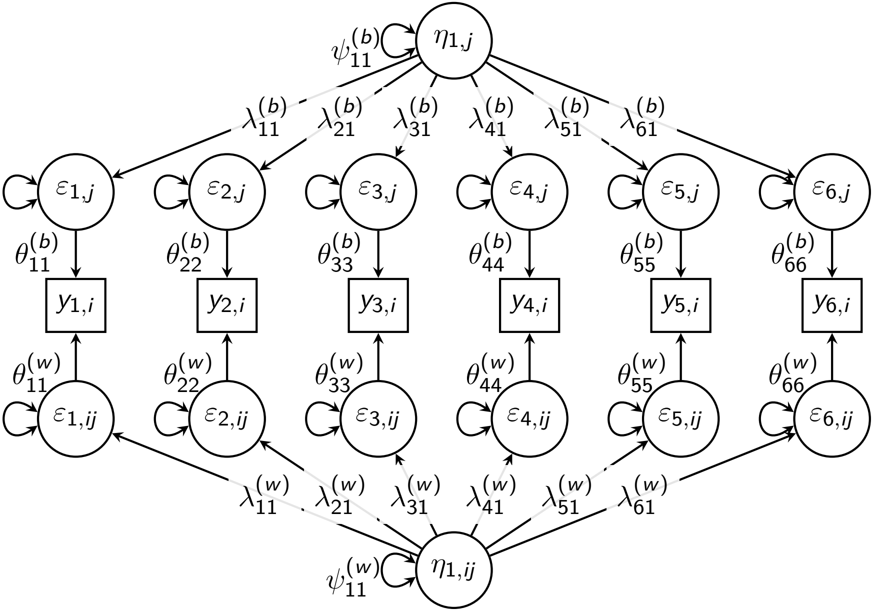
$$u_{0j} \sim \mathcal{N}(0, \sigma_u)$$

$$\varepsilon_{ij} \sim \mathcal{N}(0, \sigma_\varepsilon)$$

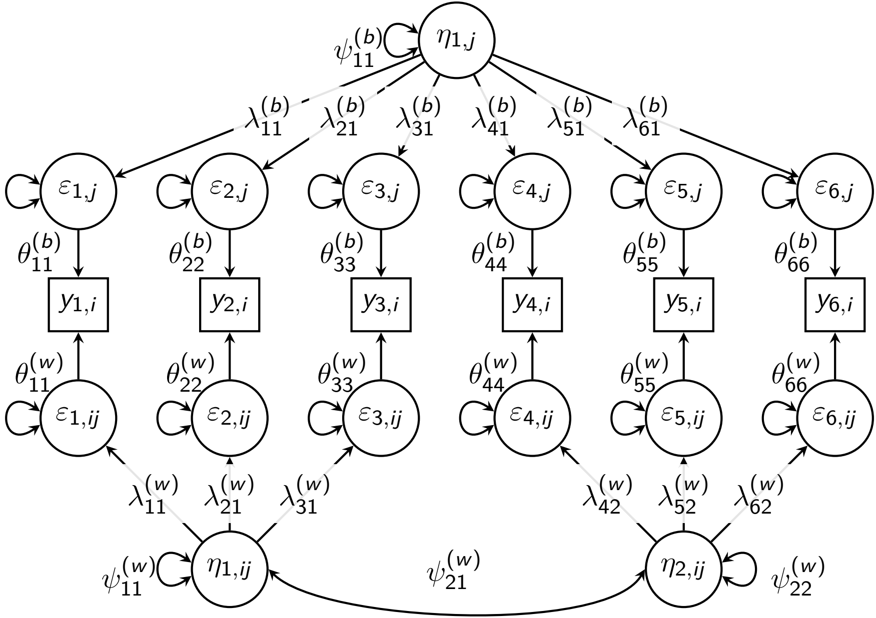
Clustered Multivariate Data



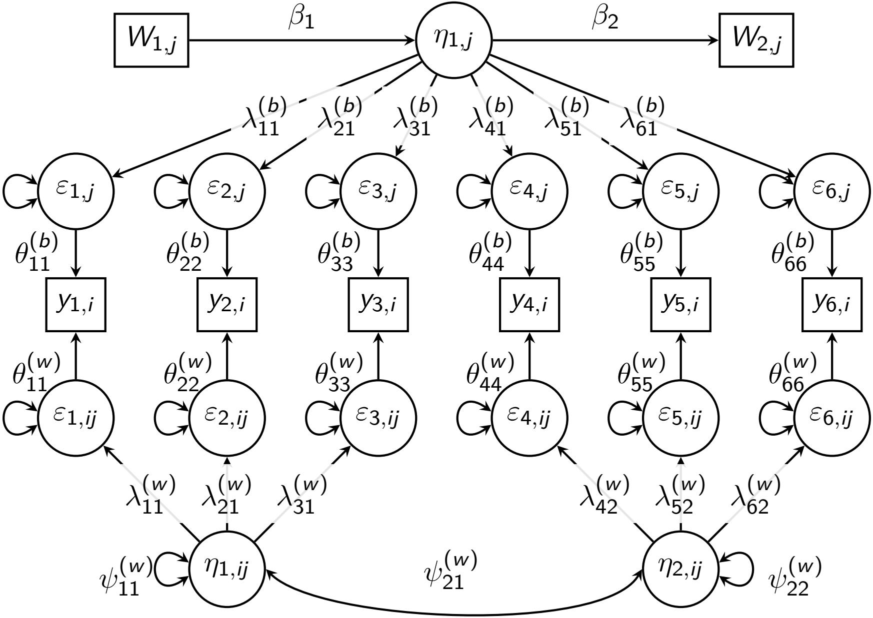
Clustered Multivariate Data: 1-Factor Structure



Clustered Multivariate Data: Different Structures



Clustered Multivariate Data: Level-2 Variables



Outline

- Translate familiar MLMs to MLSEM
  - intercept-only model, calculate ICC
  - add a predictor
- Conflated vs. Decomposed Effects
  - manifest approach (MLM)
  - latent approach (MLSEM)
  - contextual effects
- Special Details
  - level-2 predictors
  - estimation options (FIML, EM, marginal)
- Multilevel Path Models
  - multilevel mediation
  - decomposing indirect effects
  - Evaluating Global vs. Level-Specific Fit
- Multilevel Factor Models
  - Configural and Shared Constructs
  - Cluster Invariance
  - Reliability of Latent Partitions or Composites

Brief Note About lavaan's Model Syntax

`lavaan` does not use `?formula` objects, like most modeling functions in R, because an SEM includes several formulas.

The `?model.syntax` help page describes several operators:

- `~` specifies regression slopes (like a `formula` object)
- `~~` specifies (co)variances (double-headed arrow)
- `=~` specifies factor loadings to define latent variables (on left)

`lavaan` syntax for multigroup SEM can be specify either:

- with a vector of parameters (1 per group), e.g., label the slope `y~x` in 2 groups: `y ~ c(b1, b2)*x`
- in "blocks" of syntax per group, similar to M*plus*:

```
model <- ' ## equivalent to y ~ c(b1, b2)*x
  group: 1
    y ~ b1*x
  group: 2         # or level: 1 (or 2) for MLSEM
    y ~ b2*x  '
```

Multilevel Regression for Clustered Data

Import Example Data

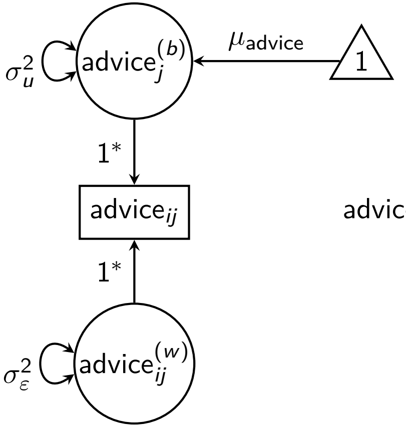
Available from the companion site for the textbook *Multilevel analysis: Techniques and applications* (Hox, Moerbeek, & van de Schoot, 2018). Right-click this link to download the file.

```
Galo <- read.table("Galo.dat",
                   col.names = c("school","sex","galo","advice",
                                 "feduc","meduc","focc","denom"))
Galo[Galo == 999] <- NA
## dummy codes for school type (ref. group = nondenominational)
Galo$Protestant <- ifelse(Galo$denom == 1, yes = 1, no = 0)
Galo$Catholic   <- ifelse(Galo$denom == 3, yes = 1, no = 0)
```

- father's occupational status (`focc`), father's education (`feduc`), and mother's education (`meduc`) used as SES indicators
- GALO school-achievement scores mediate the effect of SES on teacher's `advice` about the student's level of secondary schooling

But let's begin simply by modeling `advice`

Compare MLM and MLSEM for These Data



$$\text{advice}_{ij} = \mu_{\text{advice}} + u_{0j} + \varepsilon_{ij}$$

Intercept-Only Model with `nmle::lme()`

```
(mlm0 <- lme(fixed = advice ~ 1, random = ~ 1 | school,
             data = Galo, method = "ML", na.action = na.exclude))
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: Galo
##   Log-likelihood: -2661.849
##   Fixed: advice ~ 1
## (Intercept)
##    3.070933
##
## Random effects:
##  Formula: ~1 | school
##         (Intercept) Residual
## StdDev:   0.5167046 1.304665
##
## Number of Observations: 1552
## Number of Groups: 58
```

```
vc0 <- as.numeric(VarCorr(mlm0)[,"Variance"])
vc0[1] / sum(vc0) ## ICC
```

```
## [1] 0.1355843
```

Intercept-Only Model with `lme4::lmer()`

```
(mlm0 <- lmer(advice ~ 1 + (1 | school),
              data = Galo, REML = FALSE))
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: advice ~ 1 + (1 | school)
##    Data: Galo
##       AIC       BIC    logLik  deviance  df.resid
##  5329.698  5345.740 -2661.849  5323.698      1549
## Random effects:
##  Groups   Name        Std.Dev.
##  school   (Intercept) 0.5167
##  Residual             1.3047
## Number of obs: 1552, groups:  school, 58
## Fixed Effects:
## (Intercept)
##       3.071
```

```
vc0 <- as.data.frame(VarCorr(mlm0))$vcov
vc0[1] / sum(vc0) ## ICC
```

```
## [1] 0.1355843
```

Intercept-Only Model with lavaan

```
mod0 <- ' ## Specify model for Level-1 component(s)
  level: within              # or level: 1
    advice ~~ var_W*advice   # label Level-1 variance

## Specify model for Level-2 component(s)
  level: between             # or level: 2
    advice ~~ var_B*advice   # label Level-2 variance
    advice ~     mu*1        # label Level-2 mean

## User-defined parameter: ICC
  icc := var_B / (var_W + var_B)
'
fit0 <- lavaan(mod0, data = Galo, cluster = "school")
lavInspect(fit0, "icc") ## ICC
```

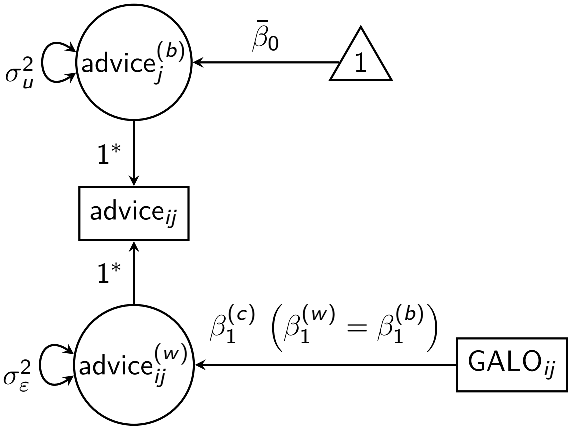
```
## advice
##  0.136
```

```
parameterEstimates(fit0, output = "pretty")
```

Intercept-Only Model with lavaan

```
##
##
## Level 1 []:
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     advice  (vr_W)    1.702    0.062   27.344    0.000    1.580    1.824
##
##
## Level 2 []:
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     advice    (mu)    3.071    0.076   40.331    0.000    2.922    3.220
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     advice  (vr_B)    0.267    0.062    4.329    0.000    0.146    0.388
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     icc               0.136    0.028    4.919    0.000    0.082    0.190
```

Multilevel Regression: Level-1 Exogenous Predictor



$$\text{advice}_{ij} = \beta_{0j} + \beta_1 \text{GALO}_{ij} + \varepsilon_{ij}$$

MLM: Conflated Effects

```
mlm1 <- lmer(advice ~ galo + (1 | school), data = Galo, REML = FALSE)
summary(mlm1)
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: advice ~ galo + (1 | school)
##    Data: Galo
##
##      AIC      BIC   logLik deviance df.resid
##   3624.5   3645.9  -1808.3   3616.5     1548
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.2798 -0.6991  0.1380  0.7080  3.7490
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  school   (Intercept) 0.0353   0.1879
##  Residual             0.5810   0.7622
## Number of obs: 1552, groups:  school, 58
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept) -5.947547   0.163762  -36.32
## galo         0.088542   0.001575   56.21
##
## Correlation of Fixed Effects:
```

MLSEM: Conflated Effects

```
mod1 <- ' ## Specify model for Level-1 component(s)
  level: 1
    advice ~ b_conf*galo  # label Level-1 slope

## Specify model for Level-2 component(s)
  level: 2
    advice ~~ advice
    advice ~ 1
'
fit1 <- sem(mod1, data = Galo, cluster = "school")
summary(fit1, rsquare = TRUE, nd = 4)
```

- Analogous to listing `galo` a within-level in M*plus* and omitting it from the `%BETWEEN%`-level model

```
VARIABLE: ...
    WITHIN = galo;
MODEL:
    %WITHIN%: advice ON galo;
```

MLSEM: Conflated Effects

```
##
##
## Level 1 []:
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   advice ~
##     galo    (b_cn)   0.0885   0.0016  56.1787   0.0000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .advice           0.5810   0.0212  27.3405   0.0000
##
## R-Square:
##                    Estimate
##     advice           0.6985
##
##
## Level 2 []:
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .advice          -5.9475   0.1638 -36.3062   0.0000
##
## Variances:
```

MLM: Pseudo-R^2 for Conflated Effects

Compare `lavaan`'s Level-1 R^2 to the **level-specific** pseudo-R^2 calculable from MLM, as the decrease in Model 1's variance components from the (intercept-only) Model 0

$$\text{Pseudo-}R^2_{\text{Level-1}} = \frac{\sigma^2_{\varepsilon(0)} - \sigma^2_{\varepsilon(1)}}{\sigma^2_{\varepsilon(0)}}$$

$$\text{Pseudo-}R^2_{\text{Level-2}} = \frac{\sigma^2_{u(0)} - \sigma^2_{u(1)}}{\sigma^2_{u(0)}}$$

```
vc1 <- as.data.frame(VarCorr(mlm1))$vcov
setNames( (vc0 - vc1) / vc0 ,
         nm = c("between","within"))
```

```
##   between    within
## 0.8677815 0.6586865
```

Modeling Covariance Structures at Each Level

Conflated vs. Decomposed Effects

Estimating a single slope for x_{ij} *conflates* its within- and between-level effects

- **ecological/atomistic fallacies**: assumes the same causal process at individual and group levels
- **difficult interpretation**: 1-unit increase on x_{ij} could compare two subjects from (with) same cluster (mean), different cluster means, or both

Level-1 predictors (x_{ij}) can be partitioned into Level-specific components, similar to the outcome (y_{ij})

- Between-cluster $\bar{y}_j$ variance can only be explained by between-level component $\bar{x}_j$
- Within-cluster $(y_{ij} - \bar{y}_j)$ variance can only be explained by within-level component $(x_{ij} - \bar{x}_j)$

$$y_{ij} = \beta_{0j} + \beta_1^{(b)} \bar{x}_j + \beta_1^{(w)} (x_{ij} - \bar{x}_j) + \varepsilon_{ij}$$

Contextual Effect

Why decompose a Level-1 predictor's effect?

- investigate causal process at each level of analysis
  - design intervention for school vs. individual?
- estimate the *context effect*
  - difference between x's between- and within-level effect $(\beta_\text{B} - \beta_\text{W})$
  - e.g., the effect of being in a school with 1-unit higher average GALO scores ($\bar{x}_j$) on a teacher's `advice`, given an individual's own GALO score (x_{ij})

$$
\begin{aligned}
\widehat{\text{advice}} &= \beta^{(\text{w})}(x_{ij} - \bar{x}_j) + \beta^{(\text{b})}\bar{x}_j \\
&= \beta^{(\text{w})}x_{ij} - \beta^{(\text{w})}\bar{x}_j + \beta^{(\text{b})}\bar{x}_j \\
&= \beta^{(\text{w})}x_{ij} + (\beta^{(\text{b})} - \beta^{(\text{w})})\bar{x}_j
\end{aligned} \tag{1}
$$

Manifest vs. Latent Decomposition

Common factors are latent variables

- scale items (e.g.) are observed indicators
- reliable composite (e.g., scale mean per subject) as proxy

Cluster means are also latent variables

- casewise observations are the indicators
- sample estimates are rarely reliable (often small N)

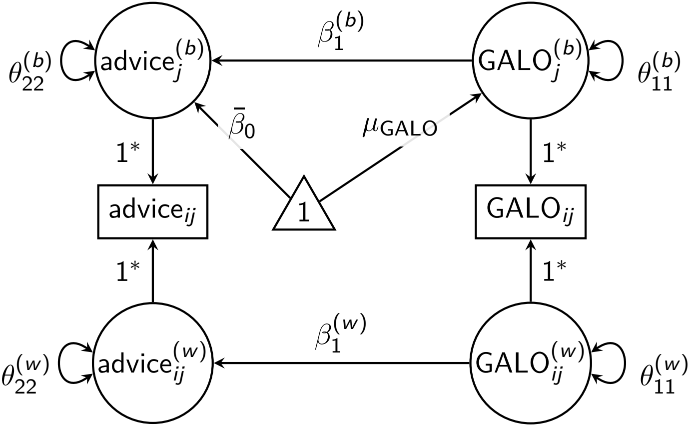
To cluster-mean-center a manifest variable, calculate sample estimates of cluster means (cluster composite)

- analogous to calculating scale mean as proxy for common factor

MLSEM can treat both common factors and cluster means as latent

- avoid potential attenuation of unreliable composites
- "doubly latent" approach (Lüdtke et al., 2011)

Multilevel Regression: Decomposed Effects



$$\text{advice}_{ij} = \beta_{0j} + \beta_1^{(b)} \mu_j^{\text{GALO}} + \beta_1^{(w)} \left(\text{GALO}_{ij} - \mu_j^{\text{GALO}} \right) + \varepsilon_{ij}$$

MLM: "Manifest Covariate" Approach (Lüdtke et al., 2008)

```r
## calculate cluster means for GALO
galoMs <- aggregate(galo ~ school, data = Galo, FUN = mean)
## rename variable
names(galoMs)[names(galoMs) == "galo"] <- "galo_B"
## merge new variable into data
Galo.c <- merge(Galo, galoMs, by = "school")
## cluster-mean-center each student's score
Galo.c$galo_W <- Galo.c$galo - Galo.c$galo_B
    ## or use ?misty::center
## fit multilevel manifest-covariate (MMC) model
mmc <- lmer(advice ~ galo_W + galo_B + (1 | school),
            data = Galo.c, REML = FALSE)
## calculate contextual effect
mmcCoefs <- fixef(mmc)
mmcCoefs["galo_B"] - mmcCoefs["galo_W"]
```

```
##      galo_B
## 0.002417381
```

MLM: "Manifest Covariate" Approach (Lüdtke et al., 2008)

```
summary(mmc)
```

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: advice ~ galo_W + galo_B + (1 | school)
##    Data: Galo.c
##
##      AIC      BIC   logLik deviance df.resid
##   3626.4   3653.1  -1808.2   3616.4     1547
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.2830 -0.7032  0.1382  0.7078  3.7475
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  school   (Intercept) 0.03504  0.1872
##  Residual             0.58100  0.7622
## Number of obs: 1552, groups:  school, 58
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept) -6.173882   0.562322  -10.98
## galo_W       0.088345   0.001644   53.73
## galo_B       0.090762   0.005506   16.48
##
## Correlation of Fixed Effects:
```

MLSEM: “Latent Covariate” Approach (Lüdtke et al., 2008)

```
mod2 <- ' ## Specify model for Level-1 component(s)
  level: 1
    advice ~ b1*galo  # label Level-1 slope

## Specify model for Level-2 component(s)
  level: 2
    advice ~ b2*galo  # label Level-2 slope
    advice ~ 1

## User-defined parameter: contextual effect
  contextual := b2 - b1
'
fit2 <- sem(mod2, data = Galo, cluster = "school")
summary(fit2)
```

MLSEM: Within-School Results

```
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   advice ~
##     galo      (b1)   0.0883   0.0016  53.7019   0.0000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .advice           0.5810   0.0213  27.3393   0.0000
##
## R-Square:
##                    Estimate
##     advice           0.6587
```

Same Level-1 (pseudo-)R^2:

```
setNames((vc0 - vc1) / vc0, nm = c("between","within"))
```

```
##   between    within
## 0.8677815 0.6586865
```

MLSEM: Between-School Results

MLM's Level-2 pseudo-R^2 attenuated by unreliability of $\bar{x}_j$

```
##
## Regressions:
##                      Estimate   Std.Err   z-value   P(>|z|)
##   advice ~
##     galo      (b2)     0.0919    0.0066   13.9992    0.0000
##
## Intercepts:
##                      Estimate   Std.Err   z-value   P(>|z|)
##    .advice            -6.2901    0.6711   -9.3727    0.0000
##
## Variances:
##                      Estimate   Std.Err   z-value   P(>|z|)
##    .advice             0.0349    0.0107    3.2541    0.0011
##
## R-Square:
##                      Estimate
##     advice             0.8735
##
## Defined Parameters:
##                      Estimate   Std.Err   z-value   P(>|z|)
##     contextual         0.0036    0.0068    0.5261    0.5988
```

Exercise 1

Fit a multilevel regression model in which GALO scores are predicted by the parental education variables (`meduc` and `feduc`) at each level of analysis

- Label all slopes in the syntax
- Define (`:=`) contextual effect per parent
  - The difference in slopes across levels
- You can also define (`:=`) the difference between each parent's effect
  - i.e., mother vs. father education, separately at each level
  - This is **not** a contextual effect, or even a multilevel question, but a potentially interesting comparison in these data

Or, if you brought your own data, you can fit a regression model you are interested in. I will be available for questions.

Level-2 Predictors

Level-2 Predictors

Predictors that only vary between clusters can only explain between-cluster variance of a Level-1 outcome

- within a cluster, everyone has the same cluster mean of the outcome
- Level-2 predictor changes subject scores by affecting their cluster

In our example data, the type of school is represented by 2 dummy codes for Catholic ($N - 245$) and Protestant ($N = 234$) schools

- $N = 1080$ nondenominational schools are the reference category
- controlling for additional (correlated) predictors affects interpretation

Coefficients for between- and within-level components of a Level-1 predictor are still comparable across levels (i.e., contextual effect)

Control for School Type

```
mod.L2 <- ' level: 1
    advice ~ b1g*galo + b1f*feduc + b1m*meduc

  level: 2
    advice ~ b2g*galo + b2f*feduc + b2m*meduc +
               b2c*Catholic + b2p*Protestant

## contextual effects
  context_galo := b2g - b1g
  context_mom  := b2m - b1m
  context_dad  := b2f - b1f
## compare parameters
  M.v.F_w      := b1m - b1f
  M.v.F_b      := b2m - b2f
  Prot.v.Cath  := b2p - b2c
'
fit.L2 <- sem(mod.L2, data = Galo, cluster = "school")
summary(fit.L2, rsq = TRUE)
```

Level-1 Results

```
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   advice ~
##     galo     (b1g)    0.086    0.002   48.301    0.000
##     feduc    (b1f)    0.015    0.012    1.245    0.213
##     meduc    (b1m)    0.041    0.013    3.213    0.001
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .advice            0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .advice            0.580    0.022   26.498    0.000
##
## R-Square:
##                    Estimate
##     advice            0.661
```

Level-2 Results

```
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   advice ~
##     galo     (b2g)    0.034    0.028    1.212    0.226
##     feduc    (b2f)    1.076    0.668    1.609    0.108
##     meduc    (b2m)   -1.129    0.799   -1.413    0.158
##     Catholic (b2c)    0.141    0.110    1.277    0.202
##     Protstnt (b2p)   -0.030    0.125   -0.241    0.810
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .advice           -1.347    2.554   -0.527    0.598
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .advice           -0.002    0.021   -0.094    0.925
##
## R-Square:
##                    Estimate
##     advice               NA
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|)
##     context_galo     -0.052    0.028   -1.821    0.069
##     context_mom      -1.171    0.801   -1.462    0.144
##     context_dad       1.061    0.670    1.584    0.113
##     M.v.F_w           0.026    0.022    1.231    0.218
##     M.v.F_b          -2.205    1.467   -1.503    0.133
##     Prot.v.Cath      -0.171    0.199   -0.857    0.391
```

Estimation Options

Full-Information Maximum Likelihood Estimation

FIML is available for incomplete data

- data assumed missing at random (MAR), conditional on observed data in the model
- each subject's likelihood calculated for subset of observed variables
- FIML not available with MLM

```
fiml <- sem(mod2, data = Galo, cluster = "school", missing = "FIML")
summary(fiml)
```

```
## lavaan 0.6-11 ended normally after 32 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                         5
##
##   Number of observations                          1559
##   Number of clusters [school]                       58
##   Number of missing patterns -- level 1              2
##
## Model Test User Model:
```

Expectation–Maximization Algorithm

M*plus* uses an accelerated EM (EMA) algorithm by default

- switch to quasi-Newton if EM gets stuck (likelihood changes slowly)

`lavaan`'s default is quasi-Newton, but EM is available

- **not** fast for large models
- see control options on the tutorial page

```
em <- sem(mod2, data = Galo, cluster = "school",
          # missing = "FIML", # fails to converge with FIML
          optim.method = "em")
summary(em)
```

```
## lavaan 0.6-11 ended normally after 23 iterations
##
##   Estimator                                         ML
##   Optimization method                               EM
##   Number of model parameters                         5
##
##                                                   Used       Total
##   Number of observations                          1552        1559
##   Number of clusters [school]                       58
```

Marginal Models in `lavaan`

Using the `cluster=` argument with a single-level model triggers cluster-robust *SE*s and test statistics

```
fitCR <- sem('advice ~ galo', data = Galo, cluster = "school")
```

```
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   advice ~
##     galo              0.089    0.003   34.267    0.000
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .advice           -5.989    0.269  -22.225    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .advice            0.616    0.025   24.652    0.000
##
## R-Square:
##                    Estimate
##     advice            0.688
```

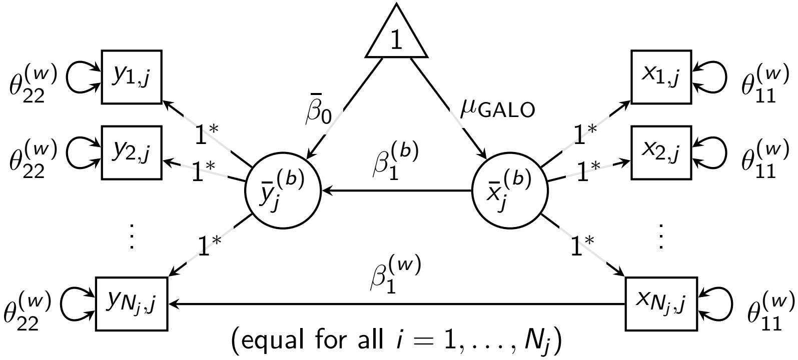
Comparable to Marginal Regression Model

Generalized estimating equation (GEE) with exchangeable correlation structure (compound symmetry)

```
library(geeM)
summary(geem(advice ~ galo, data = Galo,
             id = school, corstr = "exchangeable"))
```

```
##              Estimates Model SE Robust SE   wald p
## (Intercept)  -5.94800 0.163900  0.261000 -22.79 0
## galo          0.08854 0.001576  0.002516  35.19 0
##
##  Estimated Correlation Parameter:  0.05719
##  Correlation Structure:  exchangeable
##  Est. Scale Parameter:  0.6171
##
##  Number of GEE iterations: 3
##  Number of Clusters:  58    Maximum Cluster Size:  46
##  Number of observations with nonzero weight:  1552
```

Wide-Format Approach



- subjects are indicators (in columns) of cluster means
- common factor = random intercept (Mehta & Neale, 2005)
- option for **many small** clusters, infeasible with many variables
- also for categorical data (Barendse & Rosseel, 2020)

Multilevel Path Models

Multilevel Path Models

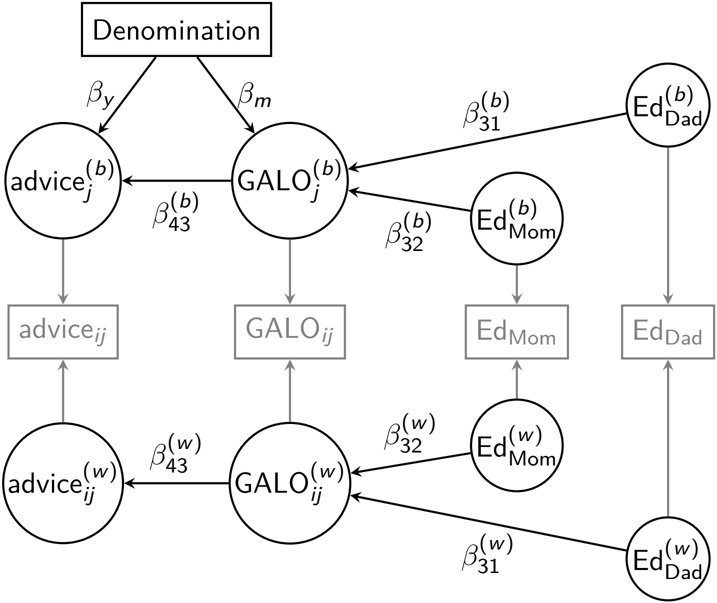
Indirect effects can also be conflated or decomposed

- When predictor(s), mediator(s), or ultimate outcome(s) only vary at Level 2, indirect effect is **only** defined at Level 2
- Preacher et al. (2010) provide taxonomy, discuss implications and possibilities with MLM vs. MLSEM

The following example adds both parents' education levels as predictors of the mediator GALO

- The "*a*" paths could be interesting to compare between parents and across levels
- The "*b*" path had no discernible contextual effect, but the indirect effect could be cause the "*a*" paths had contextual effects

Multilevel Mediation Model



Multilevel Mediation in lavaan

```
mod.med <- ' level: 1
    advice ~ b1*galo            #+ c1f*feduc + c1m*meduc
      galo ~ a1f*feduc + a1m*meduc

  level: 2
    advice ~ b2*galo +            #c2f*feduc + c2m*meduc +
               b2c.y*Catholic + b2p.y*Protestant
      galo ~ a2f*feduc + a2m*meduc +
               b2c.m*Catholic + b2p.m*Protestant

## Indirect effects
  ind_w_mom     := a1m*b1
  ind_w_dad     := a1f*b1
  ind_b_mom     := a2m*b2
  ind_b_dad     := a2f*b2
## contextual effects
  context_galo  := b2 - b1
  context_mom   := a2m - a1m
  context_dad   := a2f - a1f
  context_ind_m := ind_b_mom - ind_w_mom
  context_ind_f := ind_b_dad - ind_w_dad
'
fit.med <- sem(mod.med, data = Galo, cluster = "school",
               missing = "FIML", fixed.x = FALSE) # incomplete exogenous
summary(fit.med, rsq = TRUE, fit = TRUE)
```

Level-1 Results

```
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   advice ~
##     galo      (b1)    0.088    0.002   53.473    0.000
##   galo ~
##     feduc    (a1f)    1.126    0.175    6.428    0.000
##     meduc    (a1m)    0.920    0.190    4.848    0.000
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   feduc ~~
##     meduc             2.300    0.124   18.529    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .advice            0.581    0.021   27.339    0.000
##    .galo            131.880    4.845   27.219    0.000
##     feduc             4.453    0.168   26.567    0.000
##     meduc             3.743    0.139   26.904    0.000
##
## R-Square:
##                    Estimate
##     advice            0.659
##     galo              0.093
```

Level-2 Results, Indirect & Contextual Effects

```
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   advice ~
##     galo      (b2)    0.101    0.007   15.230    0.000
##     Cathlc (b2c.y)    0.032    0.084    0.381    0.703
##     Prtstn (b2p.y)    0.253    0.085    2.961    0.003
##   galo ~
##     feduc    (a2f)    8.151    7.230    1.127    0.260
##     meduc    (a2m)   -6.806    9.829   -0.692    0.489
##     Cathlc (b2c.m)    0.815    1.846    0.442    0.659
##     Prtstn (b2p.m)   -4.115    1.551   -2.653    0.008
##
## R-Square:
##                    Estimate
##     advice            0.910
##     galo              0.840
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|)
##     ind_w_mom         0.081    0.017    4.829    0.000
##     ind_w_dad         0.099    0.016    6.380    0.000
##     ind_b_mom        -0.688    0.995   -0.692    0.489
##     ind_b_dad         0.824    0.733    1.124    0.261
##     context_galo      0.013    0.007    1.898    0.058
##     context_mom      -7.726    9.843   -0.785    0.432
##     context_dad       7.026    7.242    0.970    0.332
##     context_ind_m    -0.769    0.996   -0.772    0.440
##     context_ind_f     0.725    0.734    0.988    0.323
```

Evaluating Global Fit

```
##
## Model Test User Model:
##
##   Test statistic                                 5.374
##   Degrees of freedom                                 4
##   P-value                                        0.251
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.999
##   Tucker-Lewis Index (TLI)                       0.997
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.015
##   90 Percent confidence interval - lower         0.000
##   90 Percent confidence interval - upper         0.043
##
## Standardized Root Mean Square Residual (corr metric):
##
##   SRMR (within covariance matrix)                0.028
##   SRMR (between covariance matrix)               0.023
```

Evaluating Level-Specific Fit

Global fit (dominated by Level-1 N) conflates how well the

- Level-1 model reproduces within-cluster (co)variances
- Level-2 model reproduces between-cluster (means & co)variances

Ryu & West (2009) recommended evaluating each level separately, by saturating the other level

- "partially saturated models" approach
- Add direct effects (1 level at a time) to saturate each model

Also fit partially saturated **null** models

- to calculate incremental fit indices (e.g., CFI & TLI)
- independence model should only constrain orthogonality for *endogenous* variables
- *exogenous* predictors should freely covary

Saturated Between, Constrained Within

```
mod1.sat2 <- ' level: 1
    advice ~ galo
      galo ~ feduc + meduc
  level: 2
    advice ~ galo + feduc + meduc + Catholic + Protestant
      galo ~ feduc + meduc + Catholic + Protestant
'
fit1.sat2 <- sem(mod1.sat2, data = Galo, cluster = "school",
                 missing = "FIML", fixed.x = FALSE)
```

Saturated Between, Null Model Within

```
mod.null.sat2 <- ' level: 1
## endogenous variances
  advice ~~ advice
    galo ~~ galo
## exogenous (co)variances
   feduc ~~ feduc + meduc
   meduc ~~ meduc
level: 2
     advice ~~ advice + galo + feduc + meduc + Catholic + Protestant
       galo ~~ galo + feduc + meduc + Catholic + Protestant
      feduc ~~ feduc + meduc + Catholic + Protestant
      meduc ~~ meduc + Catholic + Protestant
   Catholic ~~ Catholic + Protestant
 Protestant ~~ Protestant
## all intercepts at Level 2
  advice + galo + feduc + meduc + Catholic + Protestant ~ 1
'
fit.null.sat2 <- lavaan(mod.null.sat2, data = Galo, cluster = "school",
                        missing = "FIML", fixed.x = FALSE)
## FAILS to converge, avoid CFI/TLI at Level 1
```

Saturated Between, Test/Evaluate Within

```r
fitMeasures(fit1.sat2, output = "pretty")
```

```
##
## Model Test User Model:
##
##   Test statistic                                14.429
##   Degrees of freedom                                 2
##   P-value                                        0.001
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.063
##   90 Percent confidence interval - lower         0.035
##   90 Percent confidence interval - upper         0.095
##
## Standardized Root Mean Square Residual (corr metric):
##
##   SRMR (within covariance matrix)                0.026
##   SRMR (between covariance matrix)               0.025
```

Saturated Within, Constrained Between

```
mod2.sat1 <- ' level: 1
    advice ~ galo + feduc + meduc
      galo ~ feduc + meduc
  level: 2
    advice ~ galo + Catholic + Protestant
      galo ~ feduc + meduc + Catholic + Protestant
'
fit2.sat1 <- sem(mod2.sat1, data = Galo, cluster = "school",
                  missing = "FIML", fixed.x = FALSE)
```

Saturated Within, Null Model Between

```
mod.null.sat1 <- ' level: 1
     advice ~~ advice + galo  + feduc + meduc
       galo ~~ galo   + feduc + meduc
      feduc ~~ feduc  + meduc
      meduc ~~ meduc
level: 2
  ## endogenous variances
    advice ~~ advice
      galo ~~ galo
  ## exogenous (co)variances
      feduc ~~ feduc + meduc + Catholic + Protestant
      meduc ~~ meduc + Catholic + Protestant
   Catholic ~~ Catholic + Protestant
 Protestant ~~ Protestant
## all intercepts at Level 2
  advice + galo + feduc + meduc + Catholic + Protestant ~ 1
'
fit.null.sat1 <- lavaan(mod.null.sat1, data = Galo, cluster = "school",
                        missing = "FIML", fixed.x = FALSE)
```

Saturated Within, Test/Evaluate Between

```r
fitMeasures(fit2.sat1, baseline.model = fit.null.sat1, output = "pretty")
```

```
##
## Model Test User Model:
##
##   Test statistic                                 0.000
##   Degrees of freedom                                 2
##   P-value                                        1.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    1.000
##   Tucker-Lewis Index (TLI)                       1.066
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.000
##   90 Percent confidence interval - lower         0.000
##   90 Percent confidence interval - upper         0.000
##
## Standardized Root Mean Square Residual (corr metric):
##
##   SRMR (within covariance matrix)                0.003
##   SRMR (between covariance matrix)               0.021
```

Exercise 2

These data contain responses to 3 variables rated by 126 teachers about their 2990 6th-grade students, and 1 variable that are students' self-reports. They were original obtained from the Study of Life Transitions (MSALT, 1983–1985).

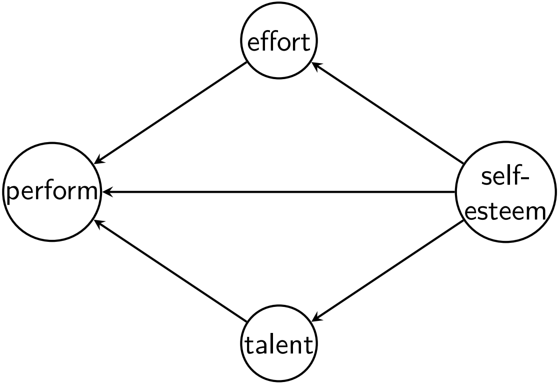
```
msalt <- read.table("msalt-med.dat", header = TRUE)
msalt[1:3,]
```

```
##   teacherID effort talent perform  gse
## 1         1      7      6       5 -0.2
## 2         1      6      4       3  0.0
## 3         1      5      3       3 -0.8
```

Specify and fit a parallel mediation model in which the effect of students' self-reported global self-esteem (`gse`) on their teacher-rated mathematics performance (`perform`) is partially mediated by their `talent` and `effort` (also rated by teachers). This model is analogous to Preacher et al.'s (2010) Example 1.

Exercise 2

Here is the model to fit **at each level of analysis**



Multilevel (Confirmatory) Factor Analysis

Multilevel (Confirmatory) Factor Analysis

Multiple indicators of common factors can also be measured in a multilevel setting

- can consider item responses as cross-classified in persons and items (e.g., Generalizability Theory designs)

Indicators can be measured by Level-2 units

- e.g., teachers/managers (self-)report info (about group)

When indicators are measured by Level-1 units (students/coworkers), what is the nature of the construct?

- **Configural** constructs are latent aggregates of Level-1 common factors
  - analogous to partitioning observed Level-1 variables
- **Shared** constructs are characteristics of Level-2 units
  - Level-1 units are “raters” of their group’s characteristic

Multilevel CFA: "Enjoy Math" Factor

MSALT data include 3 indicators of how much students enjoy math

- "I find working on math assignments very ..." (boring — interesting)
- "How much do you like doing math?" (a little — a lot)
- "Compared to other subjects, how good are you at math?" (much worse — much better)

Unconstrained 3-indicator CFA is saturated at both levels

```
math <- read.table("msalt-math.dat", header = TRUE)
mod.config <- ' level: 1
  enjoy_W =~ likemath + howmuch + ranksubj
level: 2
  enjoy_B =~ likemath + howmuch + ranksubj
'
fit.config <- cfa(mod.config, data = math, std.lv = TRUE,
                  cluster = "teacherID")
summary(fit.config, std = TRUE, fit = TRUE)
```

```
## lavaan 0.6-11 ended normally after 51 iterations
```

CFA Results: Level 1

```
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.all
##   enjoy_W =~
##     likemath          1.403    0.030   47.181    0.000     0.810
##     howmuch           1.772    0.032   55.087    0.000     0.932
##     ranksubj          0.905    0.028   32.429    0.000     0.569
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.all
##    .likemath          1.031    0.051   20.401    0.000     0.344
##    .howmuch           0.474    0.070    6.780    0.000     0.131
##    .ranksubj          1.707    0.047   36.362    0.000     0.676
##     enjoy_W           1.000                                1.000
##
## R-Square:
##                    Estimate
##     likemath          0.656
##     howmuch           0.869
##     ranksubj          0.324
```

CFA Results: Level 2

```
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.all
##   enjoy_B =~
##     likemath          0.417    0.047    8.887    0.000    0.986
##     howmuch           0.406    0.047    8.563    0.000    1.004
##     ranksubj          0.243    0.041    5.899    0.000    0.830
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.all
##    .likemath          4.673    0.049   94.465    0.000   11.049
##    .howmuch           4.833    0.050   96.259    0.000   11.959
##    .ranksubj          4.953    0.039  126.690    0.000   16.905
##     enjoy_B           0.000                               0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.all
##    .likemath          0.005    0.015    0.341    0.733    0.029
##    .howmuch          -0.001    0.013   -0.099    0.921   -0.008
##    .ranksubj          0.027    0.013    2.027    0.043    0.310
##     enjoy_B           1.000                               1.000
##
## R-Square:
##                    Estimate
##     likemath          0.971
```

Configural Construct

Is `enjoy_B` the aggregate (cluster mean) analog of `enjoy_W`?

- Level-2 component of subject-level factor scores = classroom-average levels of enjoying math
- Level-1 component of subject-level factor scores = student deviations from classroom average

Calculating the proportion of factor variance at Level 2 (the ICC) requires invariance of factor loadings (Λ) across clusters

- implies invariant Λ across levels (Jak et al., 2013, 2017)

Violation (**DIF**) implies λ_j varies across clusters

- **random slope**, which `lavaan` cannot currently estimate
- current convention: compare metric invariance to less restricted configural model with unequal loadings across levels
  - does not actually represent H_A of cluster DIF
  - within-level loadings still invariant across clusters

Metric Invariance Across Clusters

Label the loadings identically across levels

- No need to fix **both** factor variances = 1
- identify by fixing only 1 level's variance, OR
- estimate both, but constrain their sum = 1

```
mod.metric <- ' level: 1
  enjoy =~ L1*likemath + L2*howmuch + L3*ranksubj
  enjoy ~~ var_W*enjoy
level: 2
  enjoy =~ L1*likemath + L2*howmuch + L3*ranksubj
  enjoy ~~ var_B*enjoy
  likemath + howmuch + ranksubj ~ 1
## Constrain sum of factor variances to 1
  var_B == 1 - var_W   # thus, var_B = ICC
'
fit.metric <- lavaan(mod.metric, data = math, auto.var = TRUE,
                     cluster = "teacherID")
summary(fit.metric, ci = TRUE, fit = TRUE,
        std = TRUE, rsq = TRUE)
```

Metric Invariance Results: Model Fit

Cannot evaluate χ^2-based fit with partially saturated models

- equality constraints across levels
- SRMR still available at each level

```
##
## Model Test User Model:
##
##   Test statistic                                5.248
##   Degrees of freedom                                2
##   P-value                                       0.073
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                   0.999
##   Tucker-Lewis Index (TLI)                      0.998
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                         0.022
##   90 Percent confidence interval - lower        0.000
##   90 Percent confidence interval - upper        0.047
##
## Standardized Root Mean Square Residual (corr metric):
##
##   SRMR (within covariance matrix)               0.002
##   SRMR (between covariance matrix)              0.057
```

Metric Invariance Results: Factor Loadings & Variances

```
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)  Std.all
##   enjoy =~
##     likemath  (L1)    1.460    0.030   48.088    0.000    0.817
##     howmuch   (L2)    1.809    0.032   56.682    0.000    0.926
##     ranksubj  (L3)    0.935    0.028   33.337    0.000    0.572

##
##
## Level 1 []:
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)  Std.all
##     enjoy   (vr_W)    0.944    0.011   85.866    0.000    1.000
##
##
## Level 2 []:
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)  Std.all
##     enjoy   (vr_B)    0.056    0.011    5.117    0.000    1.000
```

Scalar Invariance Across Clusters

Random intercepts are the Level-2 component of every Level-1 variable in a MLSEM

- indicator residuals, given the effects of common factors
- intercept invariance implies Level-2 residual variances = 0

```
mod.scalar <- ' level: 1
  enjoy =~ L1*likemath + L2*howmuch + L3*ranksubj
  enjoy ~~ var_W*enjoy
level: 2
  enjoy =~ L1*likemath + L2*howmuch + L3*ranksubj
  enjoy ~~ var_B*enjoy
  likemath + howmuch + ranksubj ~ 1
  ## SCALAR INVARIANCE: clusters intercepts constant
    likemath ~~ 0*likemath
    howmuch  ~~ 0*howmuch
    ranksubj ~~ 0*ranksubj
## Constrain sum of factor variances to 1
  var_B == 1 - var_W   # thus, var_B = ICC
'
fit.scalar <- lavaan(mod.scalar, data = math, auto.var = TRUE,
                     cluster = "teacherID")
```

Scalar Invariance Results: Test H_0

Can we reject H_0 using a likelihood ratio test (LRT)?

- e.g., using $\alpha = 1\%$

```
lavTestLRT(fit.scalar, fit.metric)
```

```
## Chi-Squared Difference Test
##
##            Df   AIC   BIC   Chisq Chisq diff Df diff Pr(>Chis
## fit.metric  2 34473 34552  5.2477
## fit.scalar  5 34484 34545 23.1352     17.887       3   0.0004
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
```

Which indicator(s) has/have significant variance of random intercepts?

- Inspect Wald z tests or CIs from metric model

Metric Invariance Results: Level-2 Residual Variances

```
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)  Std.all
##     likemath          0.031    0.011    2.714    0.007    0.206
##     howmuch          -0.027    0.012   -2.287    0.022   -0.169
##     ranksubj          0.035    0.014    2.481    0.013    0.417
##
## R-Square:
##                    Estimate
##     likemath          0.794
##     howmuch              NA
##     ranksubj          0.583
```

Luckily, scalar invariance across clusters is not usually necessary for hypotheses of interest

- e.g., we do not compare latent means across clusters

Still practical to understand why near-0 (even negative) residual variances are commonly found (Jak et al., 2021)

Exercise 3

Use `lavaan`'s simulated example data to fit a 2-factor CFA

- Level-1 indicators of Factor 1 (`y1`–`y3`)
- Level-1 indicators of Factor 2 (`y4`–`y6`)

```
data("Demo.twolevel", package = "lavaan")
```

Other variables in the data include:

- the `cluster` ID
- Level-1 covariates: `x1`–`x3`
- Level-2 covariates: `w1` & `w2`

Fit a CFA for configural constructs (i.e., equal loadings across levels), and compare it to a CFA without equality constraints

Reliability of Multilevel Measurements

Reliability of Multilevel Measurements

Multiple-item scales are commonly employed in multilevel settings

- common to ignore multilevel structure when reporting scale reliability (e.g., coefficient α)

Geldhof et al. (2014) recommend reporting separate reliability estimates for each level of analysis

- calculate α using saturated-model estimated covariance matrices at each level
- calculate model-based composite reliability (ω) with each level's estimated factor loadings and residual variances

Lai (2021) proposed reliability formulas for **observed** composites that correspond to construct meaning

Reliability of Shared (Level-2) Constructs

When group-level constructs are measured by subject-level responses, measurement error can be decomposed into different sources

- subject-level variance (individual perceptions of the cluster)
- item-specific variance

Generalizability theory (GT) provides a useful framework to calculate

- interrater reliability (IRR)
- scale reliability
- generalizability across both items and raters (subjects)

Classical GT models assume randomly parallel measurements

- loadings = 1, homogeneity of (residual/specific) variances
- variance decomposition in MLM (or SEM; Jorgensen, 2021)
- easily adapted for congeneric items (Vispoel et al., 2021)

Parallel-Indicators MLM as $i \times (p : c)$ GT Design

```
lmer(DV ~ 1 + (1 | subject:cluster) + (1 | cluster) +
              (1 | item) + (1 | item:cluster))
```

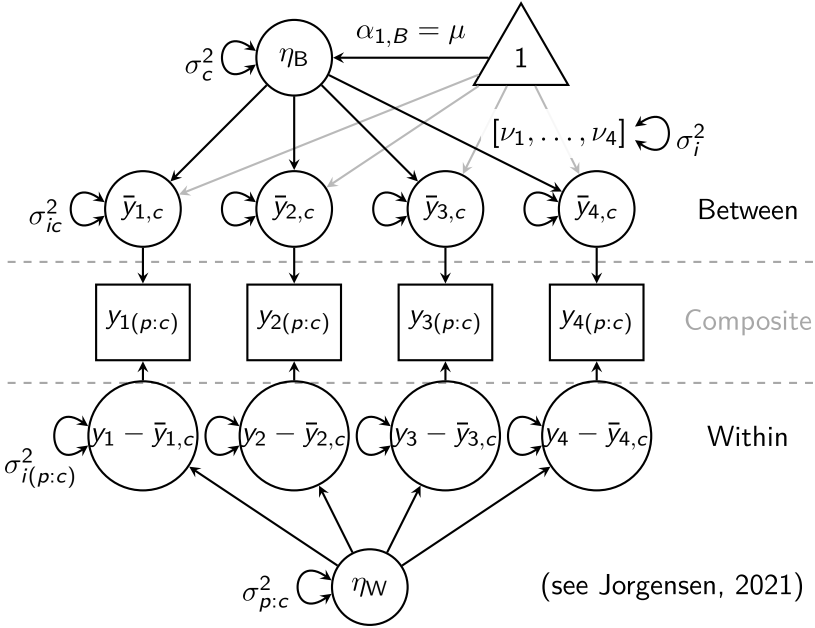
Parallel items make variance decomposition easier to see

- yields reliability estimates equivalent to α
- congeneric extension (in MLSEM) equivalent to ω

$$y_{i(p:c)} = \mu + \beta_c + \beta_i + \beta_{ic} + \beta_{p:c} + \beta_{i(p:c)}$$
$$\sigma^2_y = 0 + \sigma^2_c + \sigma^2_i + \sigma^2_{ic} + \sigma^2_{p:c} + \sigma^2_{i(p:c)}$$

- $\sigma^2_{p:c}$ = average rater differences (Level-1 factor)
- σ^2_c = average differences between clusters (Level-2 factor)
- σ^2_i = average differences between items
- σ^2_{ic} = item $\times$ cluster interaction (Level-2 residuals)
- $\sigma^2_{i(p:c)}$ = interaction + error (Level-1 residuals)

Parallel-Indicators MLSEM as $i \times (p : c)$ GT Design



(see Jorgensen, 2021)

Generalizability Coefficient for Level-2 Construct

$$\text{G-coef} = \frac{\sigma^2_c}{\sigma^2_c + \frac{\sigma^2_{ic} + \sigma^2_i}{N_i} + \frac{\sigma^2_{p:c}}{\hat{N}_{p:c}} + \frac{\sigma^2_{i(p:c)}}{\hat{N}_{p:c} \times N_i}}$$

G-coef's denominator includes all variance components *except* the main effect of items (σ^2_i)

- **Interpretation**: If we averaged all ($N_i \times N_{p:c}$) observations in cluster c, how reliably would that composite represent the construct?
- G-coef tells us how much regression/correlation coefficients would be attenuated by measurement error associated with raters **and** items

Congeneric G-coef (same interpretation)

$$\text{G-coef} = \frac{\psi^B \left(\sum_{i=1}^{N_i} \lambda_i \right)^2}{\left[\psi^B \left(\sum_{i=1}^{N_i} \lambda_i \right)^2 + \mathbf{1}' \Theta^B \mathbf{1} \right] + \frac{\psi^W \left(\sum_{i=1}^{N_i} \lambda_i \right)^2 + \mathbf{1}' \Theta^W \mathbf{1}}{\hat{N}_{p:c}}}$$

This formula generalizes to congeneric items (different loadings)

- λ_i = loading for item i (equal across levels!)
- ψ^W and ψ^B = Level-1 and -2 factor variances
- Θ^W and Θ^B = Level-1 and -2 residual covariance matrices, where $\mathbf{1}^{'}\Theta\mathbf{1}$ captures all level-specific error (co)variance
- $\hat{N}_{p:c}$ = (harmonic) mean cluster size

Same generalization applies to **all** consistency formulas

- equivalent to Lai's (2021) specification in Eq. 17 (ω^{b})

Scale-Composite Reliability (ignore clustering)

$$\alpha = \frac{\sigma^2_c + \frac{\sigma^2_{p:c}}{\hat{N}_{p:c}}}{\sigma^2_c + \frac{\sigma^2_{ic}}{N_i} + \frac{\sigma^2_{p:c}}{\hat{N}_{p:c}} + \frac{\sigma^2_{i(p:c)}}{\hat{N}_{p:c} \times N_i}}$$

Interpretation: If we averaged across N_i items for each person in cluster c, how reliably would *one person*'s composite score represent the cluster's level of the construct?

- Generalizability **only** across items
- Person-level variance becomes part of the numerator

Lai (2021, p. 94, Eq. 13) called the congeneric equivalent ω^{2L}

- the composite still has **both** sources of variance

Geldhof et al.'s (2014) Hypothetical Level-2 Reliability

$$\alpha^{\mathsf{B}} = \frac{\sigma_c^2 + \frac{\sigma_{p:c}^2}{\hat{N}_{p:c}}}{\sigma_c^2 + \frac{\sigma_{p:c}^2}{\hat{N}_{p:c}} + \frac{\sigma_{ic}^2}{N_i} + \frac{\sigma_{i(p:c)}^2}{\hat{N}_{p:c} \times N_i}}$$

Geldhof et al.'s (2014) Level-2 reliability ignores the fact that measurement error includes sampling error of cluster means!

- We cannot calculate a Level-2 composite from (latent) cluster means
- "observed"/estimated means have sampling/"rater" error
- captured by $\sigma_{p:c}^2$ and $\sigma_{i(p:c)}^2$

I recommend Lai's (2021) ω^{b}

- accounts for sampling/"rater" error

Level-1 Construct Only

$$\alpha^{\mathrm{W}} = \frac{\sigma_c^2 + \sigma_{p:c}^2}{\sigma_c^2 + \sigma_{p:c}^2 + \frac{\sigma_{ic}^2}{N_i} + \frac{\sigma_{i(p:c)}^2}{N_i}}$$

Geldhof et al. (2014) recommended the same formula for hypothetical reliability of a latent Level-1 composite

- still latent: individual deviations from latent cluster mean

Lai (2021) showed this formula does correspond to the actual reliability of a Level-1-only composite

- Level-2 variability removed by cluster-mean-centering indicators before summing/averaging across items
- Corresponds to fitting no hypothesized factor model at Level 2

Interrater Reliability of Shared Construct

The classic intraclass correlation coefficient (ICC) quantifies the proportion of variance explained by between-cluster differences

$$\text{ICC}_{\text{MLM}} = \text{ICC}(1) = \frac{\sigma_c^2}{\sigma_c^2 + \sigma_{p:c}^2} \text{ (single-rater)}$$

- Calculated using variance components estimated with an intercept-only MLM

Considering **subjects as raters** of their cluster-level construct, the same ICC quantifies IRR (McGraw & Wong, 1996)

- if you average across $N_{p:c}$ raters, scale Level-1 variance: $\text{ICC}(N_{p:c}) = \frac{\sigma_c^2}{\sigma_c^2 + \frac{\sigma_{p:c}^2}{N_{p:c}}}$

IRR of Latent Factor Scores vs. Composite

ICC calculable per indicator as its IRR, also for the factor scores

$$\text{IRR} = \frac{\psi^{\text{B}}}{\psi^{\text{B}} + \psi^{\text{W}}} = \frac{\sigma_c^2 + \frac{\sigma_{ic}^2}{N_i}}{\sigma_c^2 + \frac{\sigma_{ic}^2}{N_i} + \frac{\sigma_{p:c}^2}{\hat{N}_{p:c}} + \frac{\sigma_{i(p:c)}^2}{\hat{N}_{p:c} \times N_i}}$$

- ignores measurement error associated with items
- analogous to Geldhof et al.'s (2014) $\widetilde{\omega}^{\text{B}}$

Instead, calculate a single measure analogous to Lai's (2021) ω^{b}

$$\text{IRR} = \frac{\sigma_c^2 + \frac{\sigma_{ic}^2}{N_i}}{\sigma_c^2 + \frac{\sigma_{ic}^2}{N_i} + \frac{\sigma_{p:c}^2}{\hat{N}_{p:c}} + \frac{\sigma_{i(p:c)}^2}{\hat{N}_{p:c} \times N_i}}$$

Questions?

Whew! That was a lot of information.

Ready for some syntax examples?

- install and load `semTools`

```
library(semTools)
```

Level-Specific Reliability for `lavaan` Models

The `semTools` package includes a `compRelSEM()` function

- calculates α or ω within each “block”
  - each group
  - each level (using standard formulas, like Geldhof et al., 2014)

```
compRelSEM(fit.metric, tau.eq = TRUE) # alpha
```

```
##        level enjoy
## 1     within 0.810
## 2 teacherID 0.949
```

```
compRelSEM(fit.metric) # omega
```

```
##        level enjoy
## 1     within 0.840
## 2 teacherID 0.853
```

Lai's (2021) Reliability for Configural Constructs

If you tell `semTools::compRelSEM()` which factors are configural constructs, it instead calculates Lai's (2021) formula for α^{2L} or ω^{2L}

- also α^{W} or ω^{W} for within-level constructs
- warns when metric-invariance constraints are not detected

```
compRelSEM(fit.metric, config = "enjoy", tau.eq = TRUE) # alpha
```

```
## $config
## $config$enjoy
##   alpha_W  alpha_2L
## 0.8097092 0.8174180
```

```
compRelSEM(fit.metric, config = "enjoy") # omega
```

```
## $config
## $config$enjoy
##   omega_W  omega_2L
## 0.8401687 0.8408535
```

Lai's (2021) Reliability for Shared Constructs

If you tell `semTools::compRelSEM()` which factors are shared constructs, it instead calculates Lai's (2021) formula for α^{B} or ω^{B}

- also an overall IRR coefficient
- for overall scale reliability, can also list the shared factor in `config=`, but must fit configural model (Jak et al., 2021)

```
compRelSEM(fit.metric, shared = "enjoy", tau.eq = TRUE) # alpha
```

```
## $shared
## $shared$enjoy
##   alpha_B       IRR
## 0.5359491 0.5649516
```

```
compRelSEM(fit.metric, shared = "enjoy") # omega
```

```
## $shared
## $shared$enjoy
##   omega_B       IRR
## 0.4816287 0.5649516
```

Thank you for being here!

And good luck with fitting your MLSEMs!

- To facilitate learning and efficiency, please post your questions on the `lavaan` Google forum or on CrossValidated