

Longitudinal Analysis via Multilevel and Single-Level Structural Equation Modeling (M-SEM and SEM)

- Topics:
 - Univariate MLM as Multilevel SEM
 - Univariate MLM as Single-Level SEM
 - Pros and cons of moving from a univariate MLM to the M-SEM and SEM estimation frameworks

Longitudinal Analysis via MLM

- Most of the example models we've seen so far (in PSQF 6271 and this semester) have been estimated using software for traditional **univariate multilevel models** (MLM)
 - e.g., using SPSS/SAS/STATA MIXED; using R LME or LMER
 - Here, "**univariate**" means that only one longitudinal outcome can be predicted via fixed and random effects in each model (i.e., one long-format column as the single y_{ti} in the model)
 - Still technically multivariate outcomes given multiple outcomes over time per person, but it's "univariate" from a longitudinal/multilevel perspective
 - You can trick univariate MLM software into predicting multiple longitudinal outcomes at the same time by stacking them into a single column (see Hoffman 2015 ch. 9), but then relations between outcomes can only be specified as covariances (and not as directed fixed slopes or "paths")
- Of the limitations of univariate MLM software, perhaps the biggest is that it requires partitioning of variance in level-1 predictors to be done using observed variables (stay tuned)
 - This is why it's important that you learn longitudinal models another way!

You need a **multivariate** longitudinal model:

- To allow one longitudinal outcome's **variance components** (person-specific random effects and time-specific residuals) to be used as a **predictor of something else**
 - e.g., to examine individual differences in change as a predictor variable
 - e.g., to examine a time-varying outcome's relations with time-varying predictors that also show individual differences in change over time
- To allow cases with **missing predictors** remain in the model
 - i.e., to bring predictors into the joint likelihood (so the model will try to re-create their means, variances, and covariances, just like it does for those of an outcome) under an assumption of missing at random (MAR)
- To model **change over time in latent variables** (as measured by multiple observed variables at each occasion)
 - Requires at least some **overlap** of observed measures across occasions!

2 Options for Estimating “Truly” Multivariate Longitudinal Models

1. “Multilevel Structural Equation Modeling” (**M-SEM**)
 - Without latent variable measurement models, this is “multivariate MLM”!
 - i.e., to ME, just as “path models” that only use observed variables are not SEM (but they are estimated using software for “structural equation modeling”)
 - Uses **long-format** data; options vary by software package, such as:
 - [R Lavaan](#): only general outcomes; no *generalized* models or random slopes (other packages, like [Bayesian BRMS](#) or [OpenMx](#) have more options)
 - [STATA SEM/GSEM](#): allows general(ized) models and random slopes
 - [Mplus BETWEEN/WITHIN](#): allows general(ized) models and random slopes
 - Mplus has “[dynamic SEM](#)” that estimates additional types of parameters, too
2. (Single-level) “Structural Equation Modeling” (**SEM**)
 - Uses **wide-format** data, most software requires balanced time
 - Otherwise need to use “definition variables” (e.g., Mplus TSCORES)

Understanding M-SEMs in Mplus

- Syntax specifies the “**WITHIN**” and “**BETWEEN**” levels explicitly; results are given separately by level as well
- Parameters are labeled differently based on whether variables are unpredicted or predicted (also true in single-level SEMs)
 - **Unpredicted** variables (that have been brought into the likelihood as “outcomes”) have “**means**” and “**variances**” (i.e., as in empty models)
 - **Predicted** variables have “**intercepts**” and “**residual variances**” instead (in which “residual” means “leftover” and can occur at any level)
- Level-2 fixed slopes are always in the BETWEEN output
- Level-1 fixed slopes can show up in either level’s output based on how you write the syntax, but they will always show up in the BETWEEN output if they have random slopes with them
 - What the random slope multiplies varies by estimator ([Hoffman, 2019](#))!

M-SEM: Unconditional Random Linear Time

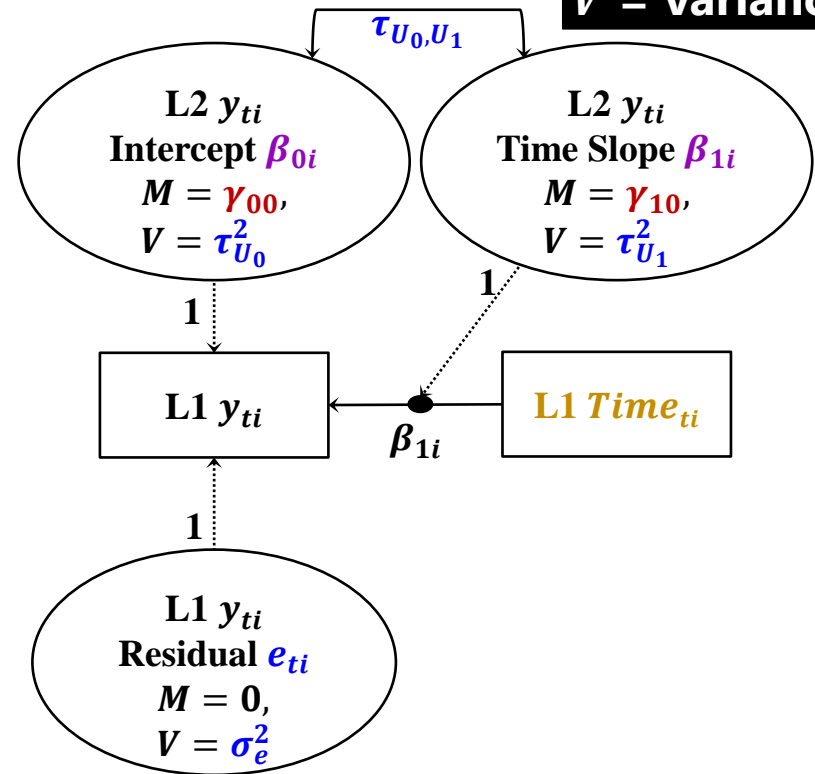
Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}(\text{Time}_{ti}) + e_{ti}$

Level 2: $\beta_{0i} = \gamma_{00} + U_{0i}$
 $\beta_{1i} = \gamma_{10} + U_{1i}$

Composite:

$$y_{ti} = (\gamma_{00} + U_{0i}) + (\gamma_{10} + U_{1i})(\text{Time}_{ti}) + e_{ti}$$

M = mean
V = variance



• The left figure shows the same model in an **M-SEM** diagram

- Boxes = observed variables
- Ovals = “latent” variables (which includes the β_i level-2 individual effects and level-1 e_{ti} residual, which is why some call this “M-SEM” always)
- Directed arrows = “paths” = slopes; undirected arrows = covariances
- **Key idea: latent variables are created by (and stay within) the model**

M-SEM: Unconditional Random Linear Time

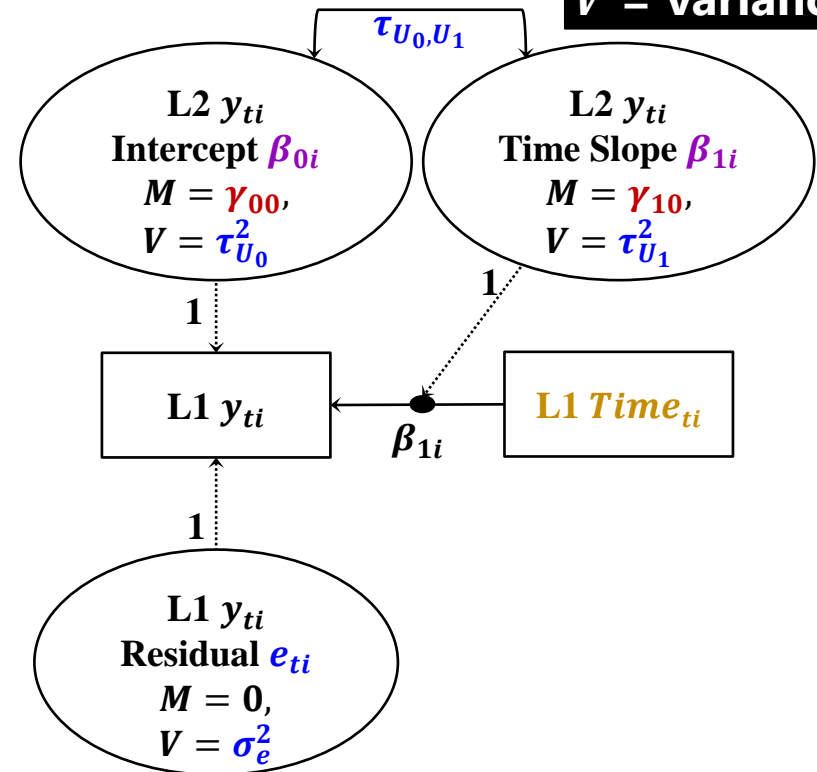
Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}(\text{Time}_{ti}) + e_{ti}$

Level 2: $\beta_{0i} = \gamma_{00} + U_{0i}$
 $\beta_{1i} = \gamma_{10} + U_{1i}$

Composite:

$$y_{ti} = (\gamma_{00} + U_{0i}) + (\gamma_{10} + U_{1i})(\text{Time}_{ti}) + e_{ti}$$

M = mean
V = variance



Mplus MODEL code
(ON = regression,
WITH = covariance):

```

%WITHIN%
  y;                                ! Level-1 residual variance (in R)
  Lin | y ON time;                  ! Create betali placeholder named Lin
%BETWEEN%
  [y Lin];                          ! Fixed effects (for intercept and linear time)
  y Lin;                             ! Level-2 random effect variances (in G)
  y WITH Lin;                        ! Level-2 random effect covariance (in G)

```

M-SEM: Unconditional Random Quadratic Time

M = mean
 V = variance

Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}(\text{Time}_{ti}) + \beta_{2i}(\text{Time}_{ti}^2) + e_{ti}$

Level 2:

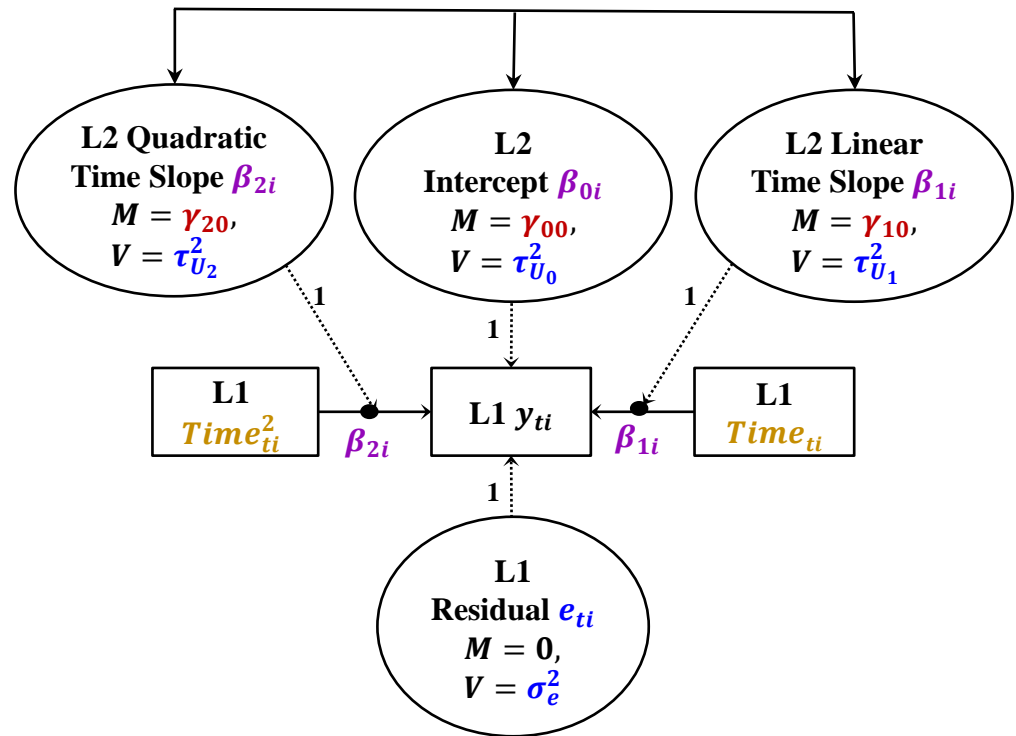
$$\beta_{0i} = \gamma_{00} + U_{0i}$$

$$\beta_{1i} = \gamma_{10} + U_{1i}$$

$$\beta_{2i} = \gamma_{20} + U_{2i}$$

Composite:

$$y_{ti} = (\gamma_{00} + U_{0i}) + (\gamma_{10} + U_{1i})(\text{Time}_{ti}) + (\gamma_{20} + U_{2i})(\text{Time}_{ti}^2) + e_{ti}$$



Arrows connecting β_i latent variable ovals represent all possible pairwise covariances (used to de-clutter)

M-SEM: Unconditional Random Quadratic Time

Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}(\text{Time}_{ti}) + \beta_{2i}(\text{Time}_{ti}^2) + e_{ti}$

M = mean
 V = variance

Level 2:

$$\beta_{0i} = \gamma_{00} + U_{0i}$$

$$\beta_{1i} = \gamma_{10} + U_{1i}$$

$$\beta_{2i} = \gamma_{20} + U_{2i}$$

Mplus MODEL code
 (ON = regression,
 WITH = covariance):

%WITHIN%

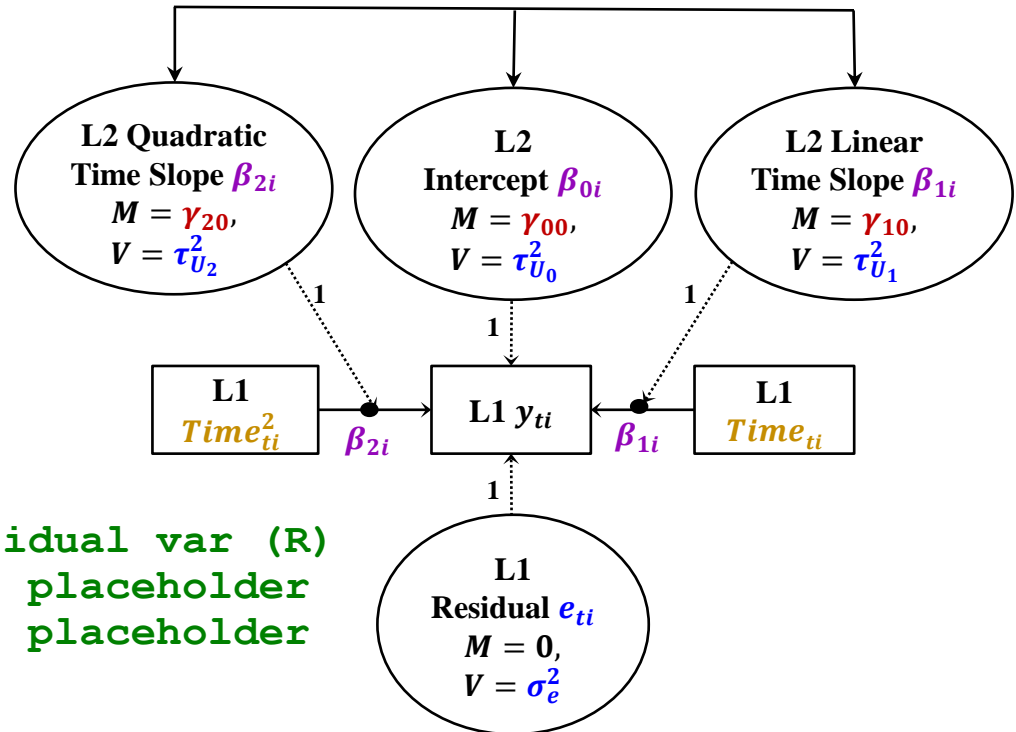
```

y;                ! L1 residual var (R)
Lin | y ON time;  ! beta1i placeholder
Qua | y ON timesq; ! beta2i placeholder
    
```

%BETWEEN%

```

[y Lin Qua];      ! Fixed effects (intercept, linear, quad)
y Lin Qua;        ! Level-2 random effect variances (G)
y Lin Qua WITH Y Lin Qua; ! Level-2 random effect covariances (G)
    
```



M-SEM: Conditional Random Quadratic Time

Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}(\text{Time}_{ti}) + \beta_{2i}(\text{Time}_{ti}^2) + e_{ti}$

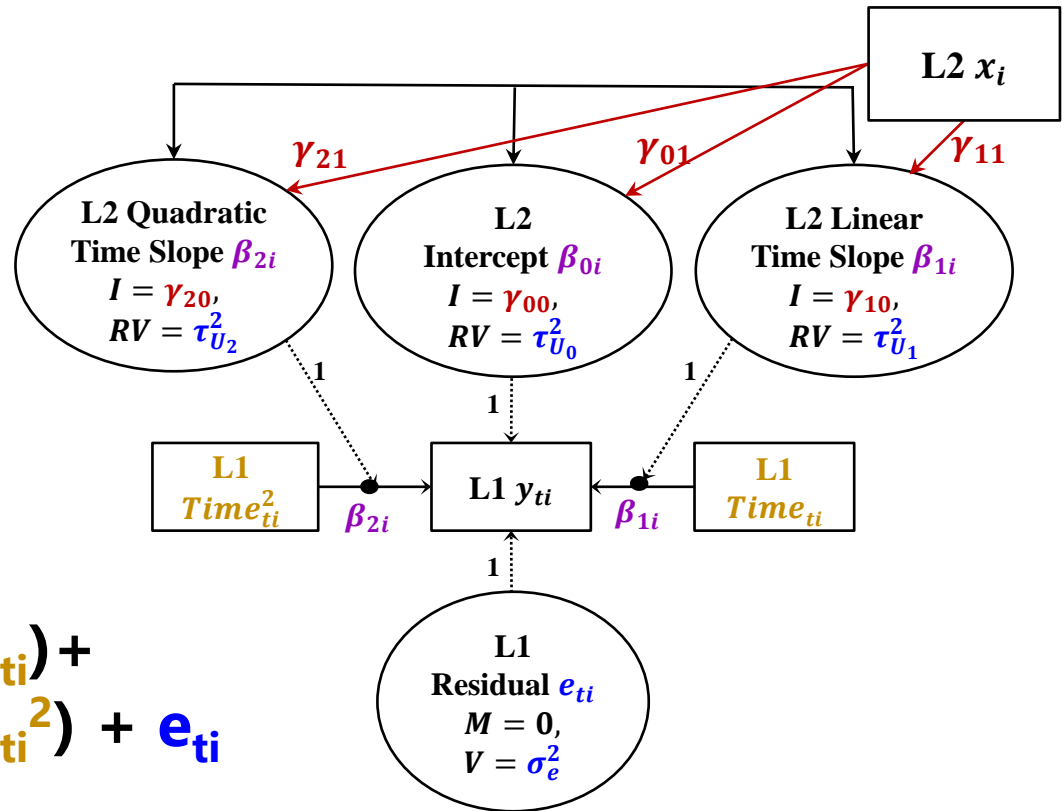
I = intercept
RV = residual
variance

Level 2:

$$\beta_{0i} = \gamma_{00} + \gamma_{01}(\text{L2}x_i) + U_{0i}$$

$$\beta_{1i} = \gamma_{10} + \gamma_{11}(\text{L2}x_i) + U_{1i}$$

$$\beta_{2i} = \gamma_{20} + \gamma_{21}(\text{L2}x_i) + U_{2i}$$



Composite:

$$y_{ti} = (\gamma_{00} + \gamma_{01} \text{L2}x_i + U_{0i}) + (\gamma_{10} + \gamma_{11} \text{L2}x_i + U_{1i})(\text{Time}_{ti}) + (\gamma_{20} + \gamma_{21} \text{L2}x_i + U_{2i})(\text{Time}_{ti}^2) + e_{ti}$$

L2 x_i is a time-invariant (observed) predictor, so its slopes must point to the β_i latent variables

M-SEM: Conditional Random Quadratic Time

Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}(\text{Time}_{ti}) + \beta_{2i}(\text{Time}_{ti}^2) + e_{ti}$

I = intercept
RV = residual variance

Level 2:

$$\beta_{0i} = \gamma_{00} + \gamma_{01}(\text{L2}x_i) + U_{0i}$$

$$\beta_{1i} = \gamma_{10} + \gamma_{11}(\text{L2}x_i) + U_{1i}$$

$$\beta_{2i} = \gamma_{20} + \gamma_{21}(\text{L2}x_i) + U_{2i}$$

Mplus MODEL code
 (ON = regression,
 WITH = covariance):

`%WITHIN%`

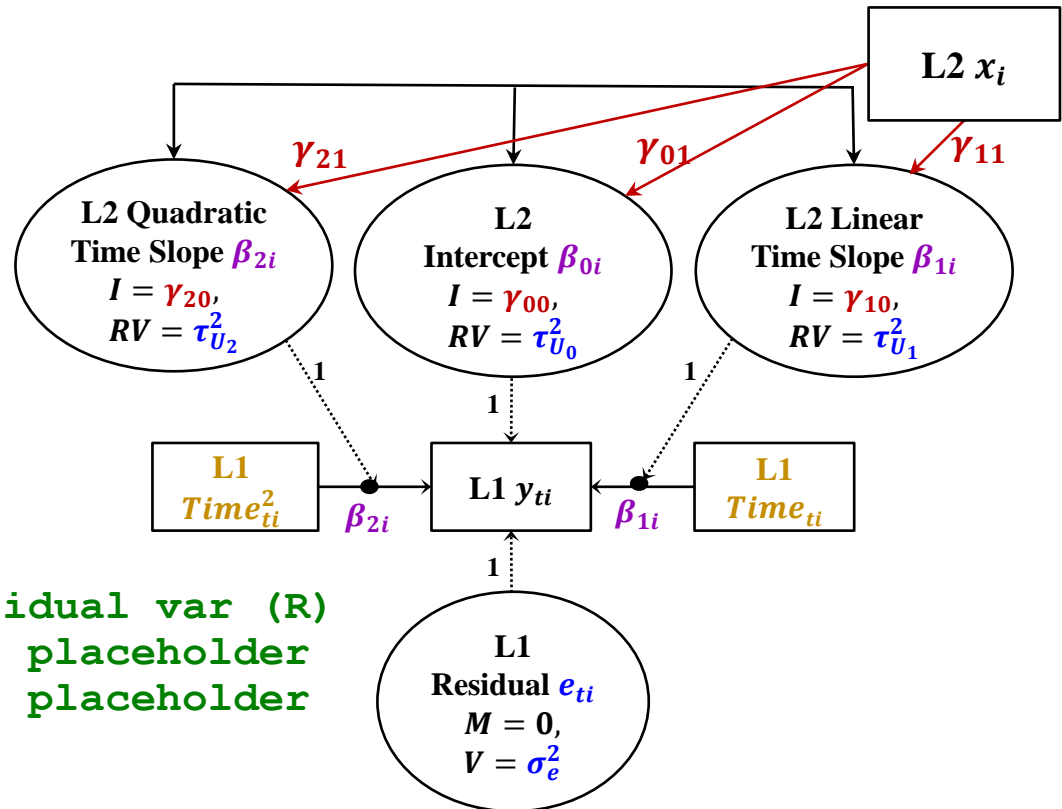
```

y;           ! L1 residual var (R)
Lin | y ON time;   ! beta1i placeholder
Qua | y ON timesq; ! beta2i placeholder
    
```

`%BETWEEN%`

```

[y Lin Qua];           ! Fixed effects (intercept, linear, quad)
y Lin Qua;             ! Level-2 random effect variances (G)
y Lin Qua WITH Y Lin Qua; ! Level-2 random effect covariances (G)
y Lin Qua ON L2x;      ! Level-2 predictor of random effects
    
```



M-SEM: Random Effects as Predictors Instead!

Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}(\text{Time}_{ti}) + \beta_{2i}(\text{Time}_{ti}^2) + e_{ti}$

Level 2:

$$\beta_{0i} = \gamma_{00} + U_{0i}$$

$$\beta_{1i} = \gamma_{10} + U_{1i}$$

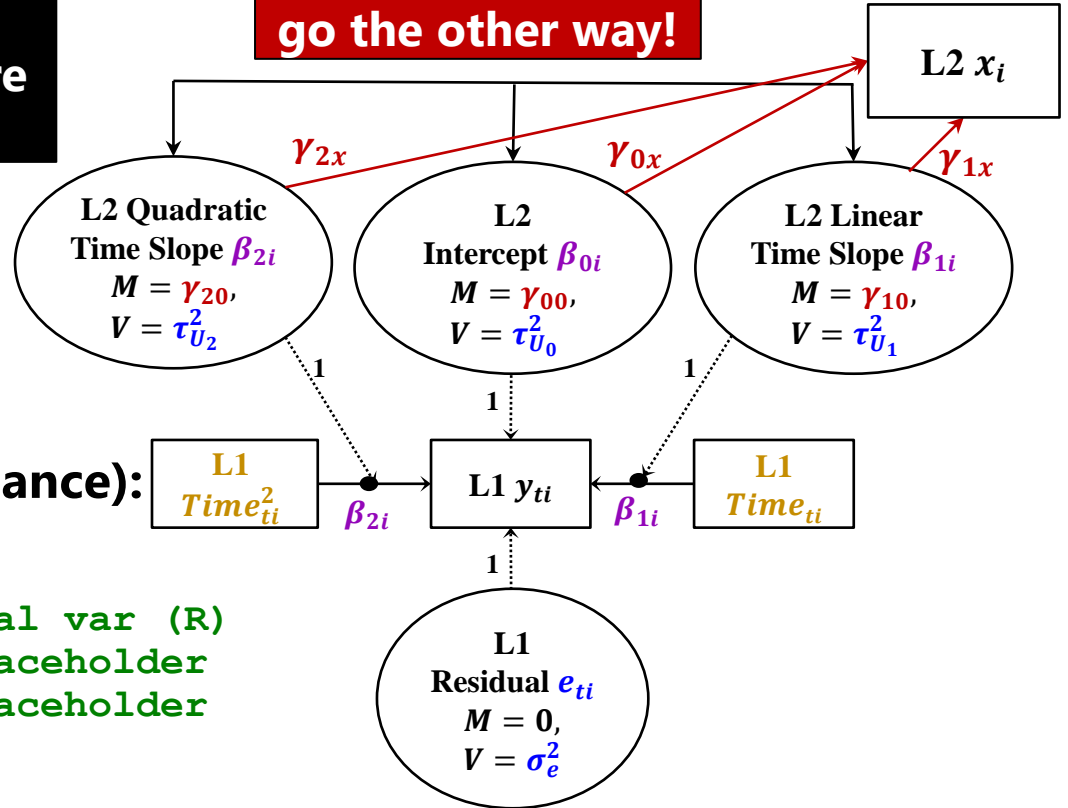
$$\beta_{2i} = \gamma_{20} + U_{2i}$$

Notation scheme would need to change, but here is an attempt:

$$\text{L2}x_i = \gamma_{\text{int}x} + \gamma_{0x}(\beta_{0i}) + \gamma_{1x}(\beta_{1i}) + \gamma_{2x}(\beta_{2i}) + U_{xi}$$

M = intercept
V = variance

Red arrows now go the other way!



Mplus MODEL code (ON = regression, WITH = covariance):

```

%WITHIN%
y;                ! L1 residual var (R)
Lin | y ON time;  ! beta1i placeholder
Qua | y ON timesq; ! beta2i placeholder

%BETWEEN%
[y Lin Qua];     ! Fixed effects (intercept, linear, quad)
y Lin Qua;       ! Level-2 random effect variances (G)
y Lin Qua WITH y Lin Qua; ! Level-2 random effect covariances (G)
L2x ON y Lin Qua; ! Level-2 random effects predicting L2x
[L2x]; L2x;      ! Level-2 variable intercept and res var
    
```

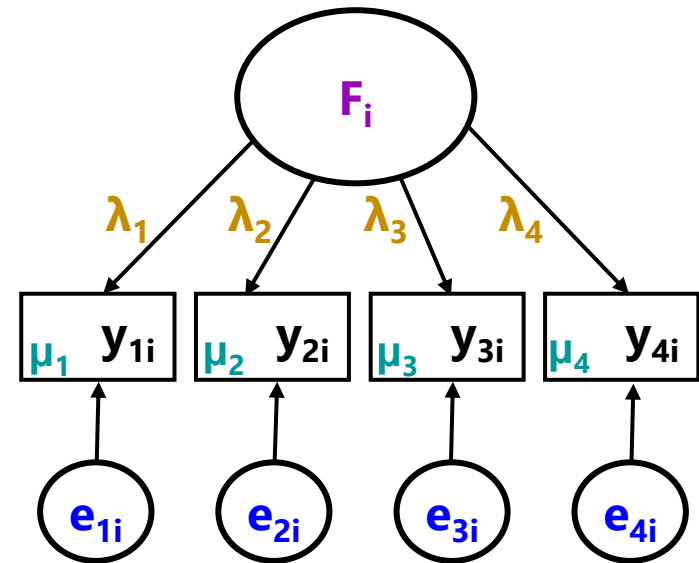
Longitudinal Models using SEM

- Same model will look different when depicted and estimated as (single-level) **SEMs using wide data (= one box per occasion)**
 - Why? Because SEMs were designed for latent variable measurement models, we must hack them a bit to use them for modeling longitudinal change
- For example, a measurement model for **4 outcomes** (“indicators”):

$$\begin{aligned} \mathbf{y}_{ti} &= \boldsymbol{\mu}_t + \boldsymbol{\lambda}_t(\mathbf{F}_i) + \mathbf{e}_{ti} \\ \mathbf{y}_{1i} &= \boldsymbol{\mu}_1 + \boldsymbol{\lambda}_1(\mathbf{F}_i) + \mathbf{e}_{1i} \\ \mathbf{y}_{2i} &= \boldsymbol{\mu}_2 + \boldsymbol{\lambda}_2(\mathbf{F}_i) + \mathbf{e}_{2i} \\ \mathbf{y}_{3i} &= \boldsymbol{\mu}_3 + \boldsymbol{\lambda}_3(\mathbf{F}_i) + \mathbf{e}_{3i} \\ \mathbf{y}_{4i} &= \boldsymbol{\mu}_4 + \boldsymbol{\lambda}_4(\mathbf{F}_i) + \mathbf{e}_{4i} \end{aligned}$$

- “**Measurement Model**”
for each indicator y_i
 - $\boldsymbol{\mu}_t$ = intercept (expected outcome)
 - $\boldsymbol{\lambda}_t$ = factor loading (slope)
 - \mathbf{e}_{ti} = residual (discrepancy)

- “**Structural Model**”: mean and variance for each factor \mathbf{F}_i (and covariances between factors in multidimensional factor models)



SEM: Unconditional Random Linear Time

Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}(\text{Time}_{ti}) + e_{ti}$

Level 2: $\beta_{0i} = \gamma_{00} + U_{0i}$

$\beta_{1i} = \gamma_{10} + U_{1i}$

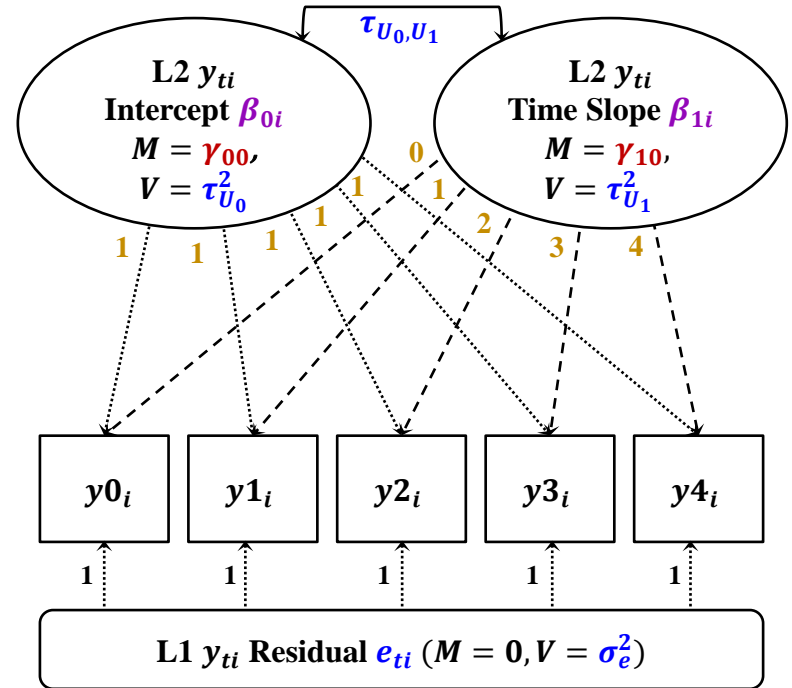
- **SEM** diagram for **5 occasions** and **2 factors** (= Int and Slope)

- Separate e_{ti} circles combined to denote variance constrained equal

- **From one factor to two:**

$$y_{ti} = \mu_t + \lambda_t(F_i) + e_{ti} \rightarrow y_{ti} = \mu_t + \lambda_{1t}(F1_i) + \lambda_{2t}(F2_i) + e_{ti}$$

- Set time-specific intercepts μ_t to 0 (to move their means up to factors)
 - Combine set of fixed and random effects into F_i factor; rename; reorder
 - Result = level-1 model: $y_{ti} = \beta_{0i}(\lambda_{1t} = 1) + \beta_{1i}(\lambda_{2t} = \text{Time}_{ti}) + e_{ti}$
 - Level-2 "fixed and random effects" \rightarrow "factor means and variances"



Mapping MLM and SEM Terminology

• **5-occasion "MLM":** $V = Z G Z^T + R$

Btw, **G** is sometimes labeled as **T** instead

$$V_i = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} \tau_{U_0}^2 & \tau_{U_{01}} \\ \tau_{U_{01}} & \tau_{U_1}^2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 & 4 \end{bmatrix} + \begin{bmatrix} \sigma_e^2 & 0 & 0 & 0 & 0 \\ 0 & \sigma_e^2 & 0 & 0 & 0 \\ 0 & 0 & \sigma_e^2 & 0 & 0 \\ 0 & 0 & 0 & \sigma_e^2 & 0 \\ 0 & 0 & 0 & 0 & \sigma_e^2 \end{bmatrix}$$

• **5-occasion "SEM":** $\Sigma = \Lambda \Phi \Lambda^T + \Psi$

Mean Vector = $\mu + \Lambda \gamma$

➤ See also lecture 4 of [my SEM class](#) (here, γ = factor means as fixed effects)

- $V = \Sigma$ = marginal predicted variances and covariances over time
- $Z = \Lambda$ = estimated loadings/fixed values for int and time predictors
- G or $T = \Phi(\mathbf{Phi})$ = level-2 factor/random variances and covariances
- $R = \Psi(\mathbf{Psi})$ = level-1 error/residual variances and covariances
- Slope factor may be replaced by time-specific intercepts (to control for mean differences over time without including individual change)
 - This would be a saturated means model instead (random intercept only)

SEM: Unconditional Random Linear Time

Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}(\text{Time}_{ti}) + e_{ti}$

Level 2: $\beta_{0i} = \gamma_{00} + U_{0i}$
 $\beta_{1i} = \gamma_{10} + U_{1i}$

Mplus MODEL code
 (ON = regression, WITH = covariance):

```
! Factor loadings fixed by @
Int BY y0@1 y1@1 y2@1 y3@1 y4@1;
Lin BY y0@0 y1@1 y2@2 y3@3 y4@4;

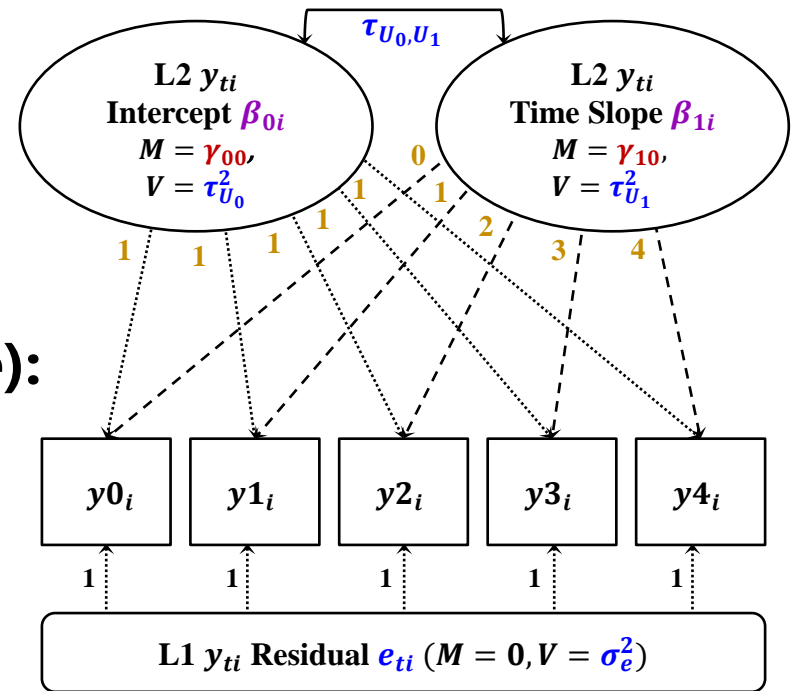
! Factor means = fixed effects
[Int Lin];

! Level-2 factor variances estimated (in G)
Int Lin;

! Level-2 factor covariance estimated (in G)
Int WITH Lin;

! Per-occasion intercepts fixed to 0
[y0@0 y1@0 y2@0 y3@0 y4@0];

! Level-1 residual variances held equal (in R)
y0 y1 y2 y3 y4 (ResVar);
```



SEM: Unconditional Random Quadratic Time

Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}(\text{Time}_{ti}) + \beta_{2i}(\text{Time}_{ti}^2) + e_{ti}$

M = mean
 V = variance

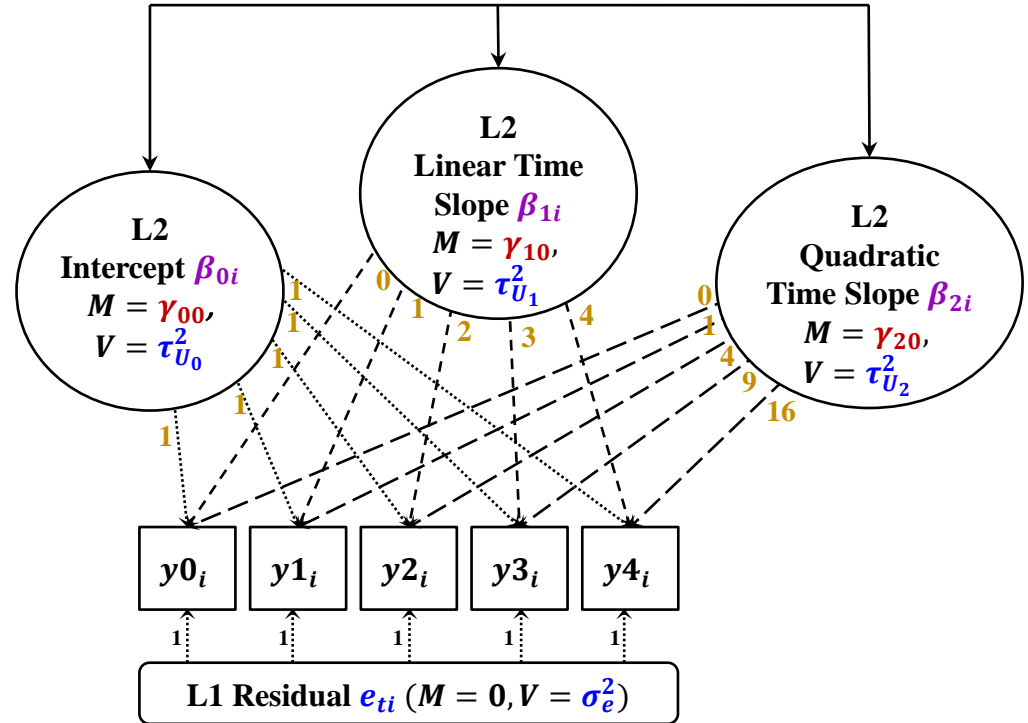
Level 2:

$$\beta_{0i} = \gamma_{00} + U_{0i}$$

$$\beta_{1i} = \gamma_{10} + U_{1i}$$

$$\beta_{2i} = \gamma_{20} + U_{2i}$$

Interpretation of β_i latent variables comes from the loading values!



Arrows connecting β_i latent variable ovals represent all possible pairwise covariances (used to de-clutter)

SEM: Conditional Random Quadratic Time

Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}(\text{Time}_{ti}) + \beta_{2i}(\text{Time}_{ti}^2) + e_{ti}$

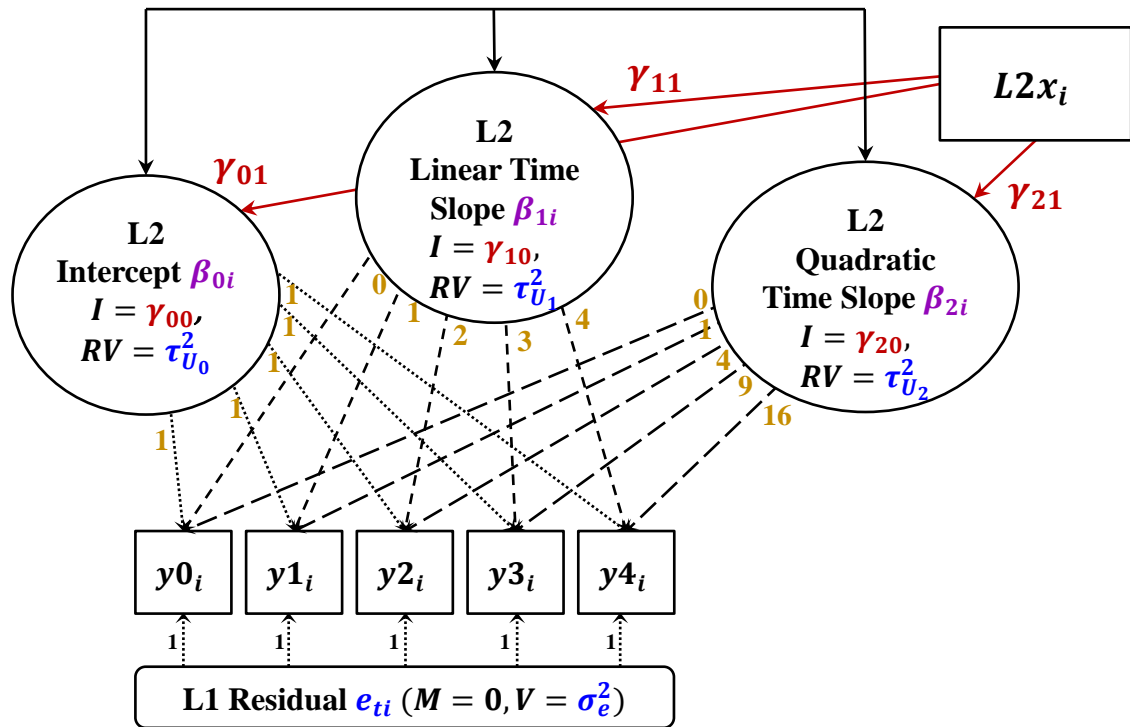
I = intercept
RV = residual variance

Level 2:

$\beta_{0i} = \gamma_{00} + \gamma_{01}L2x_i + U_{0i}$

$\beta_{1i} = \gamma_{10} + \gamma_{11}L2x_i + U_{1i}$

$\beta_{2i} = \gamma_{20} + \gamma_{21}L2x_i + U_{2i}$



Interpretation of β_i latent variables comes from the loading values!

Arrows connecting β_i latent variable ovals represent all possible pairwise residual covariances (used to de-clutter)

SEM: Random Effects as Predictors Instead!

Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}(\text{Time}_{ti}) + \beta_{2i}(\text{Time}_{ti}^2) + e_{ti}$

$M = \text{intercept}$
 $V = \text{variance}$

Level 2:

$$\beta_{0i} = \gamma_{00} + U_{0i}$$

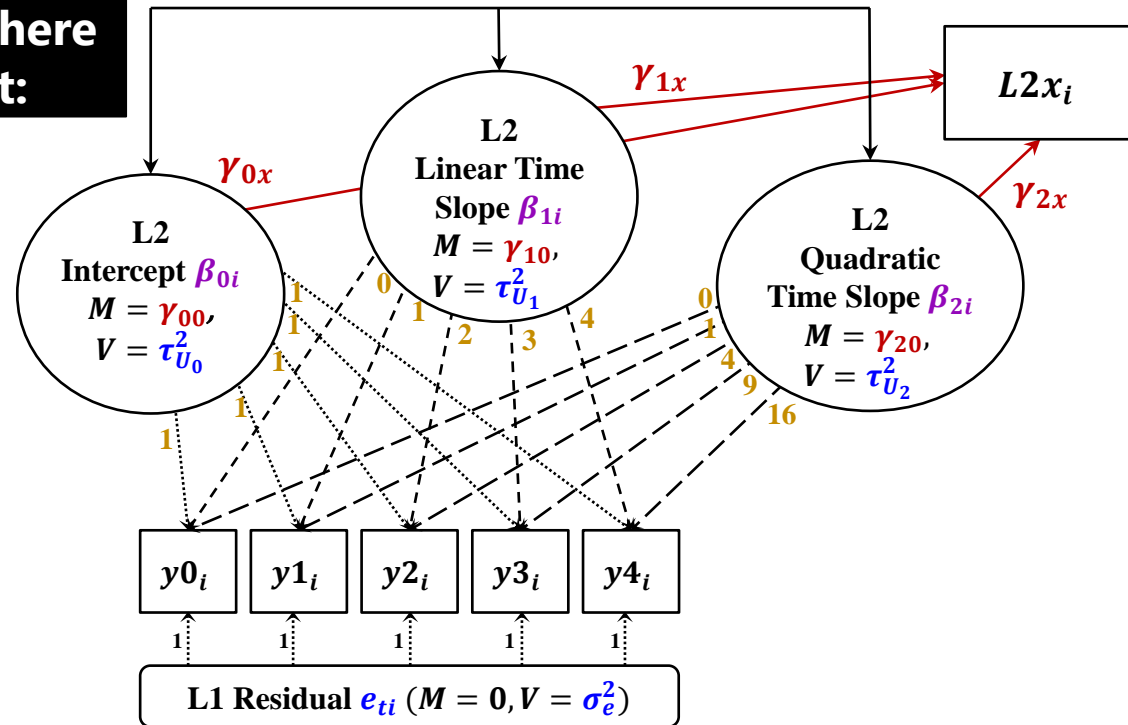
$$\beta_{1i} = \gamma_{10} + U_{1i}$$

$$\beta_{2i} = \gamma_{20} + U_{2i}$$

Notation scheme would need to change, but here is an attempt:

Red arrows now go the other way!

$$L2x_i = \gamma_{intx} + \gamma_{0x}(\beta_{0i}) + \gamma_{1x}(\beta_{1i}) + \gamma_{2x}(\beta_{2i}) + U_{xi}$$



Because there is nothing distinguishing time-specific from time-invariant variables, it can be a lot easier in SEM to accidentally create smushed effects that assume equal level-1 and level-2 fixed slopes (stay tuned!)

Mplus: Random Quadratic SEM

Just showing **MODEL** part (as estimated using **wide** data)

!!! Random quadratic model + L2x as predictor

```
! Factor loadings fixed by @
  Int BY y0@1 y1@1 y2@1 y3@1 y4@1;
  Lin BY y0@0 y1@1 y2@2 y3@3 y4@4;
  Qua BY y0@0 y1@1 y2@4 y3@9 y4@16;

! Factor intercepts estimated = fixed effects
  [Int Lin Qua];

! Level-2 factor variances estimated (in G)
  Int Lin Qua;

! Level-2 factor covariances estimated (in G)
  Int Lin Qua WITH Int Lin Qua;

! Per-occasion intercepts fixed to 0
  [y0@0 y1@0 y2@0 y3@0 y4@0];

! Level-1 residual variances held equal (in R)
  y0 y1 y2 y3 y4 (ResVar);

! Fixed effects of L2x --> latent factors
  Int Lin Qua ON L2x;
```

!!! Random quadratic model + L2x as outcome

```
! Factor loadings fixed by @
  Int BY y0@1 y1@1 y2@1 y3@1 y4@1;
  Lin BY y0@0 y1@1 y2@2 y3@3 y4@4;
  Qua BY y0@0 y1@1 y2@4 y3@9 y4@16;

! Factor intercepts estimated = fixed effects
  [Int Lin Qua];

! Level-2 factor variances estimated (in G)
  Int Lin Qua;

! Level-2 factor covariances estimated (in G)
  Int Lin Qua WITH Int Lin Qua;

! Per-occasion intercepts fixed to 0
  [y0@0 y1@0 y2@0 y3@0 y4@0];

! Level-1 residual variances held equal (in R)
  y0 y1 y2 y3 y4 (ResVar);

! Fixed effects of latent factors --> L2x
  L2x ON Int Lin Qua;

! L2x intercept and residual variance
  [L2x]; L2x;
```

Note: There are Mplus syntax shortcuts for growth models I am not using: (1) to be explicit about what the model contains, (2) to not estimate separate residual variances

R lavaan: Random Quadratic SEM

```
RandQuadPSyntax = " # L2x as a predictor
# Factor loadings fixed by *
Int =~ 1*y0 + 1*y1 + 1*y2 + 1*y3 + 1*y4
Lin =~ 0*y0 + 1*y1 + 2*y2 + 3*y3 + 4*y4
Qua =~ 0*y0 + 1*y1 + 4*y2 + 9*y3 + 16*y4

# Factor intercepts estimated = fixed effects
Int ~ 1; Lin ~ 1; Qua ~ 1
# Level-2 factor variances estimated (in G)
Int ~~ Int; Lin ~~ Lin; Qua ~~ Qua
# Level-2 factor covariances estimated (in G)
Int ~~ Lin + Qua; Lin ~~ Qua

# Per-occasion intercepts fixed to 0
y0 ~ 0; y1 ~ 0; y2 ~ 0; y3 ~ 0; y4 ~ 0

! Level-1 residual variances held equal (in R)
y0 ~~ (ResVar)*y0; y1 ~~ (ResVar)*y1
y2 ~~ (ResVar)*y2; y3 ~~ (ResVar)*y3
y4 ~~ (ResVar)*y4

# Fixed effects of L2x --> latent factors
Int + Lin + Qua ~ L2x

"
RQPMModel = lavaan(data=ExampleWhatever,
  model=RandQuadPSyntax, estimator="ML",
  mimic="mplus"); summary(RQPMModel)
```

```
RandQuadOSyntax = " # L2x as an outcome
# Factor loadings fixed by *
Int =~ 1*y0 + 1*y1 + 1*y2 + 1*y3 + 1*y4
Lin =~ 0*y0 + 1*y1 + 2*y2 + 3*y3 + 4*y4
Qua =~ 0*y0 + 1*y1 + 4*y2 + 9*y3 + 16*y4

# Factor intercepts estimated = fixed effects
Int ~ 1; Lin ~ 1; Qua ~ 1
# Level-2 factor variances estimated (in G)
Int ~~ Int; Lin ~~ Lin; Qua ~~ Qua
# Level-2 factor covariances estimated (in G)
Int ~~ Lin + Qua; Lin ~~ Qua

# Per-occasion intercepts fixed to 0
y0 ~ 0; y1 ~ 0; y2 ~ 0; y3 ~ 0; y4 ~ 0

! Level-1 residual variances held equal (in R)
y0 ~~ (ResVar)*y0; y1 ~~ (ResVar)*y1
y2 ~~ (ResVar)*y2; y3 ~~ (ResVar)*y3
y4 ~~ (ResVar)*y4

# Fixed effects latent factors --> L2x
L2x ~ Int + Lin + Qua

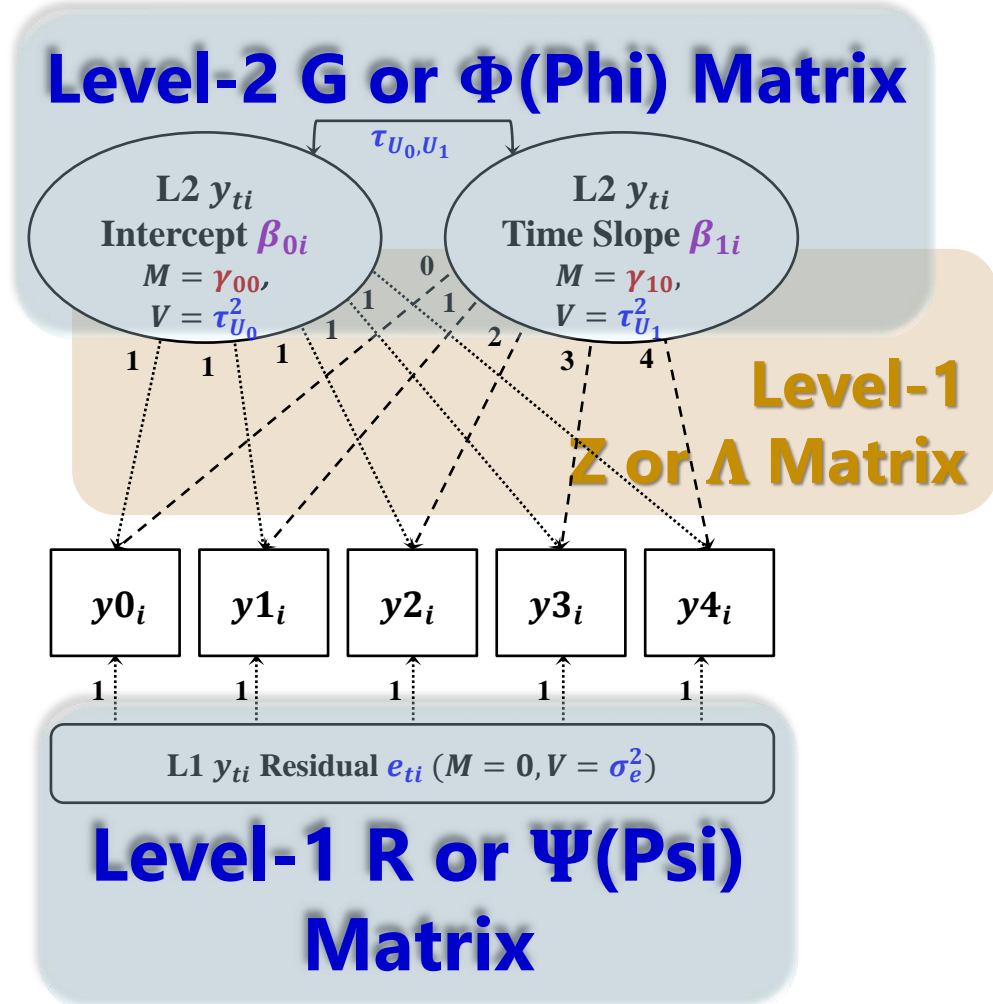
# L2x intercept and res var
L2x ~ 1; L2x ~~ L2x

"
RQOModel = lavaan(data=ExampleWhatever,
  model=RandQuadOSyntax, estimator="ML",
  mimic="mplus"); summary(RQOModel)
```

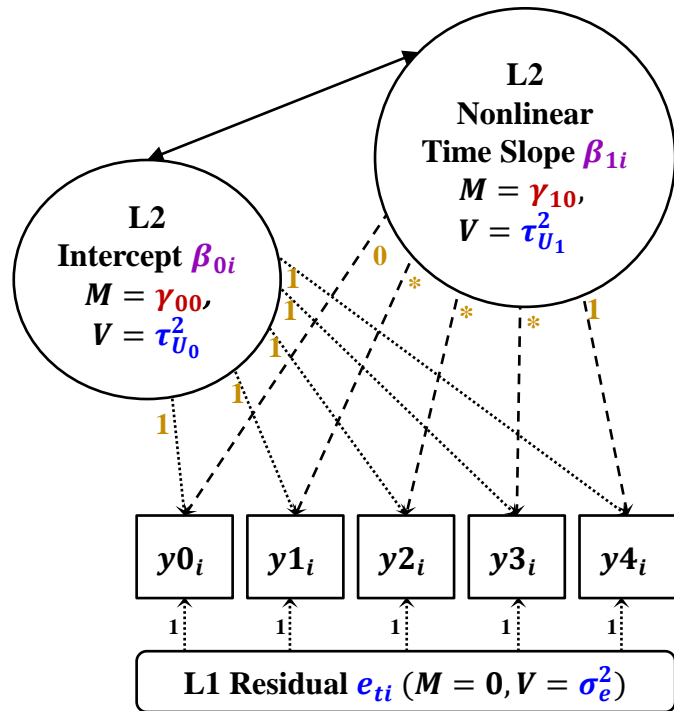
Note: There are lavaan syntax shortcuts for growth models I am not using: (1) to be explicit about what the model contains, (2) to not estimate separate residual variances

SEM for Change: Con and Pro

- **Con:** For **unbalanced time**, you need individual factor loadings for β_{1i} via “**definition variables**”
 - Mplus has **TSCORES** (wide data), but is still a problem for time-varying predictors (stay tuned)
 - Not a concern in M-SEM, in which “time” is a continuous predictor in your (long) data
- **Pro:** can estimate up to $n - 2$ factor loadings for β_{1i} in a “**latent basis**” model to capture data-specific nonlinear change
 - How much of an improvement over linear can be tested via constraints on estimated loadings



Estimated Nonlinearity via Latent Basis



Level 1:

$$y_{ti} = \beta_{0i}(1) + \beta_{1i}(\lambda_t \text{ or } \text{Time}_{ti}) + e_{ti}$$

Level 2:

$$\beta_{0i} = \gamma_{00} + U_{0i}$$

$$\beta_{1i} = \gamma_{10} + U_{1i}$$

- Latent basis models use **one latent slope factor** to capture nonlinearity
 - Can be helpful if slopes are predictors (no unique “linear vs quadratic” problem)
- Fix two factor loadings to constants, estimate the rest—for example:
 - 0 * * * 1 → loadings indicate proportion of total change at each occasion
 - 0 * * * 4 → loadings indicate amount of change at each occasion in time metric
 - Here, the intercept is at beginning, but it doesn't have to be

Btw, fixed loadings imply the same shape holds for everyone, but the loadings *could* be random instead—see [McNeish \(2020\)](#)

Mplus: Random Latent Basis SEM

Just showing **MODEL** part (as estimated using **wide** data)

!!! Random latent basis model + L2x as predictor !!! Random latent basis model + L2x as outcome

```
! Factor loadings fixed by @
Int BY y0@1 y1@1 y2@1 y3@1 y4@1;
Slp BY y0@0 y1* y2* y3* y4@1;
```

```
! Factor intercepts estimated = fixed effects
[Int Slp];
```

```
! Level-2 factor variances estimated (in G)
Int Slp;
```

```
! Level-2 factor covariance estimated (in G)
Int WITH Slp;
```

```
! Per-occasion intercepts fixed to 0
[y0@0 y1@0 y2@0 y3@0 y4@0];
```

```
! Level-1 residual variances held equal (in R)
y0 y1 y2 y3 y4 (ResVar);
```

```
! Fixed effects of L2x --> latent factors
Int Slp ON L2x;
```

```
! Factor loadings fixed by @
Int BY y0@1 y1@1 y2@1 y3@1 y4@1;
Slp BY y0@0 y1* y2* y3* y4@1;
```

```
! Factor intercepts estimated = fixed effects
[Int Slp];
```

```
! Level-2 factor variances estimated (in G)
Int Slp;
```

```
! Level-2 factor covariance estimated (in G)
Int WITH Slp;
```

```
! Per-occasion intercepts fixed to 0
[y0@0 y1@0 y2@0 y3@0 y4@0];
```

```
! Level-1 residual variances held equal (in R)
y0 y1 y2 y3 y4 (ResVar);
```

```
! Fixed effects of latent factors --> L2x
L2x ON Int Slp;
```

```
! L2x intercept and residual variance
[L2x]; L2x;
```

Note: There are Mplus syntax shortcuts for growth models I am not using: (1) to be explicit about what the model contains, (2) to not estimate separate residual variances

R lavaan: Random Latent Basis SEM

```
LatentBasisPSyntax = " # L2x as a predictor
# Factor loadings fixed by *
Int =~ 1*y0 + 1*y1 + 1*y2 + 1*y3 + 1*y4 + 1*y5
Slp =~ 0*y0 + y1 + y2 + y3 + y4 + 1*y5

# Factor intercepts estimated = fixed effects
Int ~ 1; Slp ~ 1
# Level-2 factor variances estimated (in G)
Int ~~ Int; Slp ~~ Slp
# Level-2 factor covariances estimated (in G)
Int ~~ Slp

# Per-occasion intercepts fixed to 0
y0 ~ 0; y1 ~ 0; y2 ~ 0; y3 ~ 0; y4 ~ 0

! Level-1 residual variances held equal (in R)
y0 ~~ (ResVar)*y0; y1 ~~ (ResVar)*y1
y2 ~~ (ResVar)*y2; y3 ~~ (ResVar)*y3
y4 ~~ (ResVar)*y4

# Fixed effects of L2x --> latent factors
Int + Slp ~ L2x

"
LBPMModel = lavaan(data=ExampleWhatever,
  model=LatentBasisPSyntax, estimator="ML",
  mimic="mplus"); summary(LBPMModel)
```

```
LatentBasisOSyntax = " # L2x as an outcome
# Factor loadings fixed by *
Int =~ 1*y0 + 1*y1 + 1*y2 + 1*y3 + 1*y4 + 1*y5
Slp =~ 0*y0 + y1 + y2 + y3 + y4 + 1*y5

# Factor intercepts estimated = fixed effects
Int ~ 1; Slp ~ 1
# Level-2 factor variances estimated (in G)
Int ~~ Int; Slp ~~ Slp
# Level-2 factor covariances estimated (in G)
Int ~~ Slp

# Per-occasion intercepts fixed to 0
y0 ~ 0; y1 ~ 0; y2 ~ 0; y3 ~ 0; y4 ~ 0

! Level-1 residual variances held equal (in R)
y0 ~~ (ResVar)*y0; y1 ~~ (ResVar)*y1
y2 ~~ (ResVar)*y2; y3 ~~ (ResVar)*y3
y4 ~~ (ResVar)*y4

# Fixed effects of latent factors --> L2x
L2x ~ Int + Slp

# L2x intercept and res var
L2x ~ 1; L2x ~~ L2x

"
LBOModel = lavaan(data=ExampleWhatever,
  model=LatentBasisOSyntax, estimator="ML",
  mimic="mplus"); summary(LBOModel)
```

Note: There are lavaan syntax shortcuts for growth models I am not using: (1) to be explicit about what the model contains, (2) to not estimate separate residual variances

(More) Pros and Cons

- **Multilevel/Mixed Models = Univariate MLM (where I start)**
 - Person dependency is captured primarily by **random effects** through “**levels**” in **long data**, so occasions can be unbalanced (time = predictor) and can add levels for more dependency
 - Only **univariate (general) MLMs** have **REML estimation** and **denominator DF** (DDF), which is needed in **small samples** for accurate random variances and corresponding SEs
 - **Random effects cannot be used as predictors; predictors cannot have missing data!**
- **Multilevel Structural Equation Models: M-SEMs**
 - Person dependency is captured primarily by **random effects** through “**levels**” in **long data**, so occasions can be unbalanced (time = predictor) and can add levels for more dependency
 - Given enough data, M-SEM *can* be better for time-varying predictors than univariate MLM
 - Can be easier to include random slopes for all level-1 predictors than single-level SEM
- **Single-Level Structural Equation Models: SEM**
 - Person dependency is captured by **latent variables** through **multivariate outcomes** in **wide, single-level data**, so univariate occasions are treated as observed boxes
 - **May not work for unbalanced data** or designs that need extra levels for dependency
 - Given balanced data, can easily assess absolute model fit (as in any SEM)
 - Can test longitudinal invariance (to examine change in latent factors, stay tuned!)
 - **Much easier to accidentally smush level-1 and level-2 effects together!**