

# Dynamic Structural Equation Models

Tihomir Asparouhov, Bengt Muthén and Ellen Hamaker  
Part 4

April 10, 2017

- DSEM model
- Estimation
- DIC
- Simulation Studies
- Unevenly spaced and individual-specific times of observations

Almost everything in this talk is from the paper "Dynamic Structural Equation Models" to be on our web site with the release of V8.

- Merge "time series", "structural equation", "multilevel" and "TVEM(time varying effect modeling)" concepts in a generalized modeling framework in Mplus V8
- $Y_{it}$ ,  $\eta_{it}$  and  $X_{it}$  - are the observed dependent variables, latent factors and predictors for individual  $i$  at time  $t$
- Four distinct sources of correlation in such observed data:
  - correlation due to individual specific effects (multilevel)
  - correlation due to proximity of observations (time series)
  - correlation between different variables (SEM)
  - correlation due to the same stage of evolution (TVEM)
- DSEM finds these correlations

- Main decomposition equation

$$Y_{it} = Y_{1,it} + Y_{2,i} + Y_{3,t}$$

- $Y_{2,i}$ ,  $Y_{3,t}$  are the "individual" and "time" specific contribution. These are latent variables.  $Y_{1,it}$  is the residual.
- Includes three separate models:
  - single level DSEM: type=general, N=1,  $Y_{2,i}$ ,  $Y_{3,t}$  are removed
  - twolevel level DSEM: type=twolevel,  $Y_{3,t}$  is removed
  - cross-classified DSEM: type=cross, full version
- We describe the cross-classified DSEM as it is the most general model, however ....

- Cross-classified DSEM requires that the time scale is aligned for all individuals - not every data set is applicable, ex. observational studies. Time  $t$  specific random effect apply for all individuals so time  $t$  has to mean the same thing, ex second grade.
- The two-level DSEM much simpler formulation
- The two-level DSEM is the most common and introductory model for applications
- The two-level DSEM can be estimated with less data, fewer requirements for size of  $N$  and  $T$  as compared to cross-classified DSEM, for example unbalanced designs
- The two-level DSEM easier to estimate as compared to cross-classified DSEM: much fewer number of random effects
- Mplus 8 speed for two-level DSEM always acceptable, Mplus 8 speed for cross-classified DSEM: depends on the model, some models acceptable, models with random variances or random autoregressive parameters can be very slow
- Single level model - one individual modeled separately

- The within level model includes latent variables and observed variables from the previous  $L$  (lag) periods

$$Y_{1,it} = v_1 + \sum_{l=0}^L \Lambda_{1,l} \eta_{1,i,t-l} + \sum_{l=0}^L R_l Y_{1,i,t-l} + \sum_{l=0}^L K_{1,l} X_{1,i,t-l} + \varepsilon_{1,it}$$

$$\eta_{1,it} = \alpha_1 + \sum_{l=0}^L B_{1,l} \eta_{1,i,t-l} + \sum_{l=0}^L Q_l Y_{1,i,t-l} + \sum_{l=0}^L \Gamma_{1,l} X_{1,i,t-l} + \xi_{1,it}$$

- Note that all predictors are centered i.e.  $Y_{1,i,t-l}$  is not  $Y_{i,t-l}$  (covariates X are optional)

- The usual structural equations at level 2 and 3.

$$Y_{2,i} = \nu_2 + \Lambda_2 \eta_{2,i} + \varepsilon_{2,i}$$

$$\eta_{2,i} = \alpha_2 + B_2 \eta_{2,i} + \Gamma_2 x_{2,i} + \xi_{2,i}$$

$$Y_{3,t} = \nu_3 + \Lambda_3 \eta_{3,t} + \varepsilon_{3,t}$$

$$\eta_{3,t} = \alpha_3 + B_3 \eta_{3,t} + \Gamma_3 x_t + \xi_{3,t}$$

- These include not just between parts of  $Y_{it}$  but also observed between level variables

- Random parameters on within level
  - intercepts
  - slopes
  - loadings
  - auto-regressive parameters
  - variances - new V8 feature available for DSEM and non-DSEM
  - random covariance? Only via random factor variances
- We have not found an easy to interpret, random covariance model, that is based on normally distributed random effects which can be used in linear equations as predictors or to be predicted by other variables

- Every within level random parameter  $s$  has an individual specific part  $s_{2,i}$  and time specific part  $s_{3,t}$

$$s = s_{2,i} + s_{3,t}$$

- $s_{2,i}$ ,  $s_{3,t}$  are normally distributed random effects which are a part of the between level latent variable vectors  $\eta_{2,i}$  and  $\eta_{3,t}$

- Random variances are special

$$s = \text{Exp}(s_{2,i} + s_{3,t})$$

- This way we always keep these positive

- The general model on the within level can now also be written with indices  $i$  and  $t$  for all the possible random parameters

$$Y_{1,it} = \nu_1 + \sum_{l=0}^L \Lambda_{1,lit} \eta_{1,i,t-l} + \sum_{l=0}^L R_{lit} Y_{1,i,t-l} + \sum_{l=0}^L K_{1,lit} X_{1,i,t-l} + \varepsilon_{1,it}$$

$$\eta_{1,it} = \alpha_{1,it} + \sum_{l=0}^L B_{1,lit} \eta_{1,i,t-l} + \sum_{l=0}^L Q_{lit} Y_{1,i,t-l} + \sum_{l=0}^L \Gamma_{1,lit} X_{1,i,t-l} + \xi_{1,it}$$

- The above model assumes normality
- Ordered polytomous and Binary dependent variables using the underlying  $Y^*$  approach
- Missing data: MAR likelihood based treatment via MCMC estimation. If there is autocorrelation in the data the missing data will be imputed from the neighbouring observations rather than from the average for the person! Note that standard econometrics methodology even for single level models does not include missing data. Even for single level data with missing observations this is new.

- At time  $t = 1, \dots, L$  the DSEM model uses predictors with negative time indices such as  $\eta_{i,t=0}$ ,  $\eta_{i,t=-1}$ ,  $Y_{1,i,t=0}$ ,  $Y_{1,i,t=-1}$ ,  $X_{i,t=0}$ ,  $X_{i,t=-1}$ . We treat these as auxiliary parameters with their own prior.
- If sequences are long such as  $T > 50$  the prior does not affect the results. For smaller time-series the priors may have minor effect.
- Mplus implements 2 options
- A. Mplus default: automatic priors, in the first 100 burnin MCMC iterations we update the priors from the sample statistics of  $\eta_{it}$ ,  $Y_{1,it}$ , or  $X_{i,t}$ , then we discard those 100 MCMC iteration, and retain the constructed priors. Works quite well even for small  $T$ .

- B. Specify a normal prior for these auxiliary parameters in model prior. Difficult to use in practice especially when variables are not standardized.

```
model:  
  f by y@1 (&1); f*0.6;  
  y on f&1*0.4 y&1*0.5 y&2*0.2;  
  y@0.01;  
  
model prior: f~N(0,0.6); y~N(0,1);
```

- Non-recursive models  $Y_1$  on  $Y_2$ ,  $Y_2$  on  $Y_1$
- $R_0$  and  $Q_0$  can not be random
- $\Lambda_{1,l}$ ,  $B_{1,l}$  and random variances, can be random but can not include a time specific random effect
- For categorical variables the lagged variables  $Y_{i,t-l}^*$  are not a part of the model. For categorical variables time series models can be built only through latent variables measured by the categorical variable or other continuous dependent or independent variables.

- MCMC with Gibbs sampler. All latent variables, missing values, initial conditions, random effects and model parameters, i.e., all unknown quantities are placed in one of 13 blocks:
  - B1:  $Y_{2,i}$
  - B2: All random slopes  $s_{2,i}$
  - B3:  $Y_{3,t}$
  - B4: All random slopes  $s_{3,t}$
  - B5: Other latent variables  $\eta_{2,i}$  and  $\eta_{3,t}$
  - B6: Latent variables  $\eta_{1,it}$ , including initial conditions where  $t \leq 0$
  - B7: Missing variables  $Y_{it}$
  - B8: Initial conditions  $Y_{1,it}$  and  $X_{1,it}$  for  $t \leq 0$
  - B9: Threshold parameters for all categorical variables  $\theta_3$
  - B10: Underlying variables  $Y_{it}^*$  for all categorical variables
  - B11: Non-random intercepts, slope and loadings parameters  $\theta_1$
  - B12: Non-random variance, covariance and correlation parameters  $\theta_2$
  - B13: Random variance parameters

- Determine each block conditional distribution, given all other blocks and the data
- Update (generate new values for) each block from that conditional distribution
- Repeat cycling between the blocks until convergence and use the generated values as the posterior distribution, point estimates, SE
- Mplus mini-max strategy for block formation: minimize the number of block while keeping conditional distributions explicit, i.e., maximizing the blocks. Each block is further split into the sub-blocks that are conditionally independent and update these separately. Strategy for most efficient computation and mixing. Blocks 3,6,7 sequentially updated.
- Bayes estimation inheritance: DSEM algorithm is an extension of Mplus 7.4, i.e., not developed from scratch.
- As an illustration we provide the conditional distribution for block 1

$$Y'_{1,it} = Y_{it} - Y_{3,t}$$

$$\begin{aligned}
 Y'_{1,it} - Y_{2,i} &= (I - R_0)^{-1} \mathbf{v}_1 + \sum_{l=0}^L (I - R_0)^{-1} \Lambda_{1,l} \eta_{1,i,t-l} + \\
 \sum_{l=1}^L (I - R_0)^{-1} R_l (Y'_{1,i,t-l} - Y_{2,i}) &+ \sum_{l=0}^L (I - R_0)^{-1} K_{1,l} X_{1,i,t-l} + (I - R_0)^{-1} \boldsymbol{\varepsilon}_{1,it}
 \end{aligned} \tag{1}$$

$$\begin{aligned}
 Y'_{1,it} - (I - \sum_{l=1}^L (I - R_0)^{-1} R_l) Y_{2,i} &= (I - R_0)^{-1} \mathbf{v}_1 + \sum_{l=0}^L (I - R_0)^{-1} \Lambda_{1,l} \eta_{1,i,t-l} + \\
 \sum_{l=1}^L (I - R_0)^{-1} R_l Y'_{1,i,t-l} &+ \sum_{l=0}^L (I - R_0)^{-1} K_{1,l} X_{1,i,t-l} + (I - R_0)^{-1} \boldsymbol{\varepsilon}_{1,it}
 \end{aligned} \tag{2}$$

$$F(Y_{2,i}) = \sum_t L(Y'_{1,it} | *) + L(Y_{2,i} | *)$$

- $F$  is a quadratic function of  $Y_{2,i}$  - because we split  $s_{2,i}$  into a separate block, which is specific for DSEM

$$Y_{2,i} \sim N(F''^{-1}F'(0), F''^{-1})$$

- Summary for the rest blocks can be found in the paper.

- DIC can be used to compare DSEM models. Implemented for models with all continuous dependent variable (no categorical).

$$D(\theta) = -2 \log(p(Y|\theta))$$

$$p_D = \bar{D} - D(\bar{\theta})$$

$$DIC = D(\bar{\theta}) + 2p_D$$

- Despite the clear definition with the above formulas, there is substantial variation in what DIC actually is. The source of the variation is the definition of  $\theta$ , and if it includes the latent variables or not.
- Different definitions of DIC are not comparable. You can compare only if they are using the same likelihood  $[Y|\theta]$
- DIC most likely can not be used to compare models if the two models use different  $\theta$

- In DSEM the following are used in the  $\theta$  vector in addition to all model parameters
  - $Y_{2,i}$  and all random effects  $s_{2,i}$
  - $Y_{3,t}$  and all random effects  $s_{3,t}$
  - Initial conditions
  - Latent variables  $\eta_{1,it}$  if their lagged variables are used in the model
  - Missing variables  $Y_{it}$  if their lagged variable is used in the model
- To compare two models with DIC all you need to verify is that  $\theta$  between the two models is "the same". Random effect with zero variance is OK.
- This list makes easy the computation of  $[Y|\theta]$
- $p_D$  - estimated number of parameters should generally be near the size of the vector  $\theta$ , i.e., should be near the count of the above list
- In DSEM  $p_D$  is large and needs extra long MCMC sequence for stable estimate
- ARMA(1,1) model not comparable to AR(1) with DIC for V8.

# Model fit evaluation based on comparing sample and model estimated statistics

- Assuming stationarity of the DSEM model we compute subject specific model estimated mean, variances, autocorrelations of lag  $L$ . These can be compared to their sample counterparts.
- Caution about non-stationary models where trend is in the model: de-trending the data first, the Mplus residual output does not apply directly. This applies only to the residual output in Mplus and not to model results.
- Model fit evaluation using MSE and correlation between sample v.s. model estimated. These are in Mplus plot utilities. For example, means.

$$R = Cor(\mu_i, \overline{Y_{i*}}) \quad (3)$$

$$MSE = \sum_{i=1}^N (\mu_i - \overline{Y_{i*}})^2 / N. \quad (4)$$

# Time-series model estimated means, variance, correlations using Yule-Walker

$$Z_t = \mu + \sum_{l=1}^L A_l Z_{t-l} + \zeta$$

$$\Sigma = \text{Var}(\zeta)$$

$$E(Z_t) = \left( I - \sum_{l=1}^L A_l \right)^{-1} \mu$$

$$\Gamma_j = \text{Cov}(Z_t, Z_{t-j})$$

$$\begin{bmatrix} \Gamma_0 & \Gamma_1^T & \Gamma_2^T & \dots & \Gamma_L^T \\ \Gamma_1 & \Gamma_0 & \Gamma_1^T & \dots & \Gamma_{L-1}^T \\ \Gamma_2 & \Gamma_1 & \Gamma_0 & \dots & \Gamma_{L-2}^T \\ \dots & \dots & \dots & \dots & \dots \\ \Gamma_L & \Gamma_{L-1} & \Gamma_{L-2} & \dots & \Gamma_0 \end{bmatrix} \begin{bmatrix} I \\ -A_1^T \\ -A_2^T \\ \dots \\ -A_L^T \end{bmatrix} = \begin{bmatrix} \Sigma \\ 0 \\ 0 \\ \dots \\ 0 \end{bmatrix}$$

- Simulation example using two-level random autoregressive AR(1)  $\mu = 0, r = 0.3, \sigma_{11} = \sigma_w = 3, \sigma_{22} = 0.01, \sigma_{12} = 0$
- Mplus latent centering

$$Y_{it} = \mu_i + \varepsilon_{it}$$

$$\varepsilon_{it} = r_i \varepsilon_{i,t-1} + \xi_{it}.$$

- Observed centering

$$Y_{it} = \mu_i + \varepsilon_{it}$$

$$\varepsilon_{it} = r_i (Y_{i,t-1} - \overline{Y_{i*}}) + \xi_{it}$$

- Uncentered

$$Y_{it} = \mu_i + \varepsilon_{it}$$

$$\varepsilon_{it} = r_i Y_{i,t-1} + \xi_{it}$$

- Nickell, S. (1981). Biases in dynamic models with fixed effects. *Econometrica: Journal of the Econometric Society*, 1417-1426.
- Hamaker E.L. and Grasman R.P.P.P. (2015) To center or not to center? Investigating inertia with a multilevel autoregressive model. *Front. Psychol.*, 5, 1492.
- Ludtke, O., Marsh, H.W., Robitzsch, A., Trautwein, U., Asparouhov, T., & Muthén, B. (2008). The multilevel latent covariate model: A new, more reliable approach to group-level effects in contextual studies. *Psychological Methods*, 13, 203-29.
- Asparouhov, T. & Muthén, B. (2006). Constructing covariates in multilevel regression. *Mplus Web Notes: No. 11*.

- Ludtke bias is for two-level models, involves 2 different variables, and the bias is on the between

$$\frac{(\beta_w - \beta_b)\psi_w}{T\psi_b + \psi_w}$$

- Nickell bias is for DSEM, involves 1 variable, and the bias is on the within

$$-\frac{1+r}{T-1}$$

- Both stem from not accounting for the error in the sample mean estimate of the mean
- Both disappear when cluster sample size  $T$  increases
- Both can appear in parallel in the same example

- Note that observed centering or uncentered do not exist in case there is missing data - listwise deletion is never ok
- Simulation example for various  $N$  and  $T$
- Hamaker and Grasman (2015) show that the uncentered method eliminates Nickell's bias. It does create other bias however, ex for  $\sigma_{11}$
- Hamaker and Grasman (2015) show that using the true mean to center still creates bias

Table: Nickell's bias for  $r=0.3$

T	N	Latent centering	Observed centering	Nickell's formula
10	100	0.025	-0.140	-0.144
20	50	0.006	-0.070	-0.068
30	30	0.008	-0.042	-0.045
50	50	0.000	-0.029	-0.027
100	100	-0.001	-0.014	-0.013

Nickell's formula is very accurate. Latent centering eliminates Nickell's bias.

Table: Nickell's bias for  $r=0.3$

T	N	Latent centering	Observed centering	Nickell's formula
10	100	0.025	-0.140	-0.144
20	50	0.006	-0.070	-0.068
30	30	0.008	-0.042	-0.045
50	50	0.000	-0.029	-0.027
100	100	-0.001	-0.014	-0.013

Nickell's formula is very accurate. Latent centering eliminates Nickell's bias.

Table: Bias for  $Var(\mu_i) = 3$

T	N	latent centering	Uncentered
10	100	-0.015	-1.637
20	50	0.217	-1.483
30	30	0.645	-1.256
50	50	0.378	-1.361
100	100	0.096	-1.508

For latent centering bias on  $Var(\mu_i)$  as  $N$  increases (or with using weakly informative priors). For the uncentered method in will not disappear even asymptotically as the fundamentals of the model are wrong.

- Jongerling J, Laurenceau J.P., Hamaker E. (2015). A Multilevel AR(1) Model: Allowing for Inter-Individual Differences in Trait-Scores, Inertia, and Innovation Variance. *Multivariate Behav Res.* 50(3), 334-349.
- In this paper it is shown that if subject specific variances are ignored - the structural parameters can be slightly biased. This does not happen in regular two-level models.

$$Y_{it} = \mu_i + \varepsilon_{it}$$

$$\varepsilon_{it} = r_i \varepsilon_{i,t-1} + \xi_{it}$$

$$v_i = \text{Log}(\text{Var}(\xi_{it}))$$

- The bias depends on how high the correlation is between  $r_i$  and  $v_i$

# Subject specific variance -results

**Table:** Comparing the estimation with random variance and without random variance (invariant variance): Bias(coverage)

parameter	$Cov(r_i, v_i)$	random variance	invariant variance
$E(r_i)$	high	.001(.97)	.040(.35)
$E(r_i)$	medium	.001(.98)	.028(.65)
$E(r_i)$	low	.001(.97)	.017(.83)
$E(r_i)$	none	.001(.96)	.007(.92)
$Var(r_i)$	high	.001(.97)	-.012(.47)
$Var(r_i)$	medium	.001(.93)	-.007(.78)
$Var(r_i)$	low	.001(.93)	-.004(.88)
$Var(r_i)$	none	.001(.94)	-.001(.91)

# Subject specific variance - results

More detailed method for evaluation of model estimation

$$Cor(\hat{r}_i, r_i)$$

$$SMSE = \sqrt{(1/N) \sum_i (\hat{r}_i - r_i)^2}$$

	$Cov(r_i, v_i)$	random variance	invariant variance
SMSE	high	.255	.346
SMSE	medium	.293	.329
SMSE	low	.300	.316
SMSE	none	.300	.310
correlation	high	.96	.87
correlation	medium	.92	.89
correlation	low	.91	.90
correlation	none	.91	.90

## Subject specific variance - conclusions

- Looking at the parameter estimates alone may not be enough when comparing estimation methods. Distortion of structural parameters due to ignoring subject specific variance is not simple shift in the autoregressive parameter. Error is actually doubled when looking at the random effects SMSE.
- Even in standard two-level models, using cluster specific variance is important if we use SMSE as a criterion
- Subject specific variance extracts more information from the data, yields more accurate estimation
- More simulations are needed to evaluate this issue in multivariate setting - study the effect of subject specific covariance. I was not able to find further problems on that front.

- The ARMA(1,1) model

$$Y_t = \mu + \beta Y_{t-1} + \varepsilon_t + \theta \varepsilon_{t-1}$$

$$\sigma = \text{Var}(\varepsilon_t)$$

- The measurement error AR(1) = MEAR(1)

$$Y_t = \mu + f_t + \xi_t$$

$$f_t = \beta f_{t-1} + \varepsilon_t$$

$$\sigma_1 = \text{Var}(\xi_t), \sigma_2 = \text{Var}(\varepsilon_t)$$

- The two models are equivalent

$$\sigma_1 = -\frac{\theta \sigma}{\beta}$$

$$\sigma_2 = (1 + \theta^2) \sigma + \frac{(1 + \beta^2) \theta \sigma}{\beta}$$

- ... as long as  $\sigma_1 > 0$  and  $\sigma_2 > 0$
- The MEAR(1) shows how traditional SEM logic doesn't hold for DSEM: one indicator factor model is perfectly identified
- Two reasons to prefer MEAR(1) v.s. AR(1,1)
  - More efficient Mplus estimation
  - Easier to interpret - SEM like flavor
- Two reasons to prefer MEAR(1)/AR(1,1) v.s. AR(1)
  - AR(1) exponential decay of autocorrelation is not realistic
  - ARMA(1,1) is a two-parameter fit for the autocorrelation function , v.s., AR(1) which is one parameter
- Easy to test MEAR(1)/AR(1,1) v.s. AR(1) using significance of parameter.

# AR(1) v.s. ARMA(1,1) autocorrelation decay

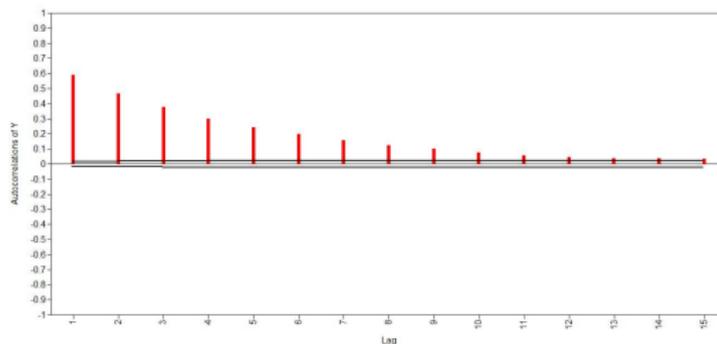
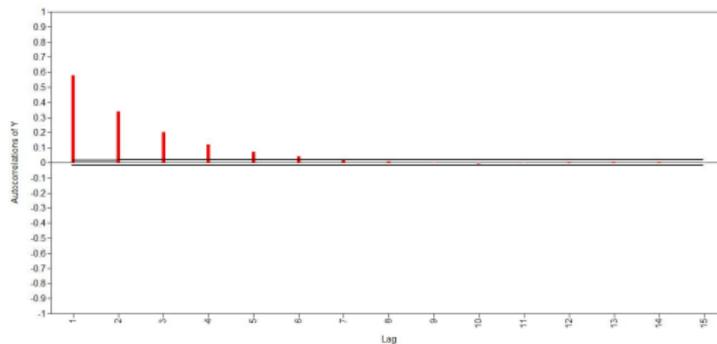


Table: Bias(coverage) MEAR(1) / ARMA(1,1), N=1

parameter	True value	$T = 100$	$T = 200$	$T = 300$	$T = 500$
$\mu$	0	-.09(.82)	-.01(.89)	-.04(.85)	-.02(.87)
$\beta$	.8	-.07(.96)	-.04(.92)	-.03(.87)	-.01(.95)
$\sigma_1$	1	-.10(.97)	-.09(.94)	-.08(.88)	-.04(.90)
$\sigma_2$	1	.25(.95)	.17(.92)	.14(.91)	.08(.90)

- $T \geq 200$  recommended for small bias and acceptable coverage levels
- For two-level models smaller  $T$  are acceptable as long as not all four parameters are subject specific - typically  $\sigma_1$  and  $\sigma_2$  will not be subject specific

# ARMA(2,1) simulation study

```
title: ARMA(2,1)

montecarlo:
  names = y;
  nobs = 500;
  nreps = 100;
  lagvar=y(2);

analysis: estimator=bayes;

model montecarlo:
  f by y@1 (&1); f*0.6;
  y on f&1*0.4 y&1*0.5 y&2*0.2;
  y@0.00000001;

model:
  f by y@1 (&1); f*0.6;
  y on f&1*0.4 y&1*0.5 y&2*0.2;
  y@0.01;
```

$$Y_t = \mu + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \varepsilon_t + \beta_3 \varepsilon_{t-1}$$

# ARMA(2,1) simulation study results

## MODEL RESULTS

		Population	ESTIMATES Average	Std. Dev.	S. E. Average	M. S. E.	95% Cover	¶ Sig Coeff
F	BY							
Y		1.000	1.0000	0.0000	0.0000	0.0000	1.000	0.000
Y	ON							
F&1		0.400	0.3875	0.2143	0.2429	0.0456	0.960	0.293
Y	ON							
Y&1		0.500	0.5178	0.2079	0.2373	0.0431	0.949	0.859
Y&2		0.200	0.1765	0.1754	0.1933	0.0310	0.960	0.111
Intercepts								
Y		0.000	-0.0016	0.0540	0.0466	0.0029	0.869	0.131
Variances								
F		0.600	0.5809	0.0373	0.0392	0.0017	0.939	1.000
Residual Variances								
Y		0.010	0.0100	0.0000	0.0000	0.0000	1.000	0.000

- Since the underlying variable of a categorical variable is not available for lag modeling MEAR(1) is a good fit and allows time series modeling with a single categorical variable
- For binary variable

$$P(Y_{it} = 1) = \Phi(\mu_i + f_{it})$$

$$f_{it} = \beta f_{i,t-1} + \xi_{it}$$

$$\mu_i \sim N(\mu, \sigma_b), \sigma_2 = \sigma_w = \text{Var}(\xi_{it}), \sigma_1 = 1$$

- For ordered polytomous variable

$$P(Y_{it} = j) = \Phi(\tau_{j+1} - \mu_i - f_{it}) - \Phi(\tau_j - \mu_i - f_{it})$$

$$f_{it} = \beta f_{i,t-1} + \xi_{it}$$

$$\mu_i \sim N(0, \sigma_b), \sigma_2 = \sigma_w = \text{Var}(\xi_{it}), \sigma_1 = 1$$

# Two-level ARMA(1,1)/MEAR(1) for binary

**Table:** Two-level ARMA(1,1)/MEAR(1) with binary variable, N=100, T=300

parameter	True value	Estimate(Coverage)
$\mu$	0	0.00 (.95)
$\beta$	.5	0.50(.78)
$\sigma_w$	1	1.01(.71)
$\sigma_b$	0.5	0.52(.94)

# Two-level ARMA(1,1)/MEAR(1) for ordered polytomous

**Table:** Two-level ARMA(1,1)/MEAR(1) with ordered polytomous, N=100, T=100

parameter	True value	Estimate(Coverage)
$\tau_1$	-3	-3.06 (.87)
$\tau_2$	-1	-1.02 (.81)
$\tau_3$	0	-0.01 (.79)
$\tau_4$	1	1.01 (.75)
$\tau_5$	3	3.05 (.81)
$\beta$	.5	0.50(.93)
$\sigma_w$	1	1.09(.83)
$\sigma_b$	0.5	0.54(.94)

# How to add a covariate in AR(1,1) and AR(1) models

- The same applies for AR(1) and AR(1,1)/MEAR(1). Three ways to do it
- Direct model

$$Y_t = \mu + f_t + \beta_1 X_t + \xi_t$$
$$f_t = \phi f_{t-1} + \varepsilon_t.$$

- Indirect model

$$Y_t = \mu + f_t + \xi_t$$
$$f_t = \phi f_{t-1} + \beta_2 X_t + \varepsilon_t.$$

- Full model

$$Y_t = \mu + f_t + \beta_1 X_t + \xi_t$$
$$f_t = \phi f_{t-1} + \beta_2 X_t + \varepsilon_t.$$

# How to add a covariate in AR(1,1) and AR(1) models

- Consider the fundamental difference between the models by what it implies for  $E(Y_t|X)$
- Direct model - no effect beyond the last value of  $X$

$$E(Y_t|X) = \mu + \beta_1 X_t.$$

- Indirect model - accumulation effect of  $X$  with diminishing effects

$$E(Y_t|X) = \mu + \beta_2(X_t + \phi X_{t-1} + \phi^2 X_{t-2} + \phi^3 X_{t-3} + \dots).$$

- Full model - direct and accumulated effect

$$E(Y_t|X) = \mu + \beta_1 X_t + \beta_2(X_t + \phi X_{t-1} + \phi^2 X_{t-2} + \phi^3 X_{t-3} + \dots).$$

# How to add a covariate in AR(1,1) and AR(1) models. The special case of $X_t = t$ , linear growth model

- Hamaker, E.L. (2005) Conditions for the equivalence of the autoregressive latent trajectory model and a latent growth curve model with autoregressive disturbances. *Sociological Methods and Research*, 33, 3, 404 - 418.
- It is shown in this paper that the direct and the indirect models are equivalent, i.e., the full model is not identified

# How to add a covariate in AR(1,1) and AR(1) models. The special case of $X_t = t$ , linear growth model

- The direct linear growth AR(1) model is

$$Y_t = \gamma_0 + \gamma_1 t + \xi_t$$

$$\xi_t = \phi \xi_{t-1} + \varepsilon_t$$

- The indirect linear growth AR(1) model is

$$Y_t = \beta_0 + \beta_1 t + \phi Y_{t-1} + \varepsilon_t$$

- The two models are algebraically equivalent and the full model is unidentified

$$\gamma_0 = \frac{\beta_0}{1 - \phi} - \frac{\phi \beta_1}{(1 - \phi)^2}$$

$$\gamma_1 = \frac{\beta_1}{1 - \phi}$$

# AR(1) quadratic growth model, $X_t = (t, t^2)$

- The direct quadratic growth AR(1) model is

$$Y_t = \gamma_0 + \gamma_1 t + \gamma_2 t^2 + \xi_t$$
$$\xi_t = \phi \xi_{t-1} + \varepsilon_t$$

- The indirect quadratic growth AR(1) model is

$$Y_t = \beta_0 + \beta_1 t + \beta_2 t^2 + \phi Y_{t-1} + \varepsilon_t$$

- The two models are algebraically equivalent and the full model is unidentified

$$\gamma_0 = \frac{\beta_0}{1-\phi} - \frac{\phi\beta_1}{(1-\phi)^2} + \frac{\beta_2\phi(1+\phi)}{(1-\phi)^3}$$
$$\gamma_1 = \frac{\beta_1}{1-\phi} - \frac{2\phi\beta_2}{(1-\phi)^2}$$
$$\gamma_2 = \frac{\beta_2}{1-\phi}$$

$$Y_{it} = \mu_i + f_t + \beta_1 X_{it} + \xi_{it}$$

$$f_{it} = \phi f_{i,t-1} + \beta_2 X_{it} + \varepsilon_{it}$$

$$\mu_i \sim N(\mu, \sigma_b)$$

- $\beta_1 = 0.3, \beta_2 = 0.4, \phi = 0.5, \mu = 0, \sigma_b = 0.7,$   
 $Var(\xi_{it}) = Var(\varepsilon_{it}) = 1$
- The covariate is generated using AR(1) process with  
 $Var(X_{it}) = 1$  and autocorrelation  $r_x = 0, 0.5, 0.8$
- We analyze the data using the full, direct and indirect models

Table: Two-level full ARMA(1,1) with covariate, N=200, T=100

parameter	$r_x$	True value	Estimate(Coverage)
$\beta_1$	0	.30	.30(.87)
$\beta_1$	0.5	.30	.30(.96)
$\beta_1$	0.8	.30	.31(.89)
$\beta_2$	0	.40	.40(.87)
$\beta_2$	0.5	.40	.40(.93)
$\beta_2$	0.8	.40	.40(.90)
$\phi$	0	.50	.50(.88)
$\phi$	0.5	.50	.50(.93)
$\phi$	0.8	.50	.50(.93)

No bias. Good coverage. Model is well identified.

**Table:** Two-level full ARMA(1,1) with covariate analyzed as direct, N=200, T=100

parameter	$r_x$	True value	Estimate(Coverage)
$\beta_1$	0	.70	.65(.00)
$\beta_1$	0.5	.70	.74(.07)
$\beta_1$	0.8	.70	.88(.00)
$\phi$	0	.50	.50(.92)
$\phi$	0.5	.50	.51(.85)
$\phi$	0.8	.50	.52(.83)

Both parameters are biased. Coverage is low. Bias depends on  $r_x$ .

**Table:** Two-level full ARMA(1,1) with covariate analyzed as indirect, N=200, T=100

parameter	$r_x$	True value	Estimate(Coverage)
$\beta_2$	0	.70	.69(.92)
$\beta_2$	0.5	.70	.67(.21)
$\beta_2$	0.8	.70	.65(.07)
$\phi$	0	.50	.36(.00)
$\phi$	0.5	.50	.38(.00)
$\phi$	0.8	.50	.41(.00)

Both parameters are biased. Coverage is low. Bias depends on  $r_x$ .

# Simulation Study - Twolevel DAFS Lag 3 model

```
montecarlo:
  names = y1-y5;
  NOBS = 10000;
  NREP = 100;
  NCSIZES = 1;
  CSIZES = 100(100);

ANALYSIS:  TYPE IS TWOLEVEL;
           estimator=bayes;
           biter=(500); proc=2;

model montecarlo:
  %within%
  f by y1-y5*1 (& 3);
  y1-y5*1;
  f@1;
  f on f&1*0.4 f&2*0.2 f&3*0.1;

  %between%
  fb by y1-y5*0.5; fb@1; y1-y5*0.2;
```

# Results - Twolevel DAFS Lag 3 model

F	BY							
Y1		1.000	0.9992	0.0134	0.0120	0.0002	0.900	1.000
Y2		1.000	1.0010	0.0127	0.0120	0.0002	0.950	1.000
Y3		1.000	1.0001	0.0138	0.0120	0.0002	0.910	1.000
Y4		1.000	0.9997	0.0128	0.0120	0.0002	0.930	1.000
Y5		1.000	0.9988	0.0134	0.0119	0.0002	0.910	1.000
F	ON							
F&1		0.400	0.4024	0.0121	0.0123	0.0002	0.980	1.000
F&2		0.200	0.1983	0.0143	0.0139	0.0002	0.950	1.000
F&3		0.100	0.0997	0.0138	0.0126	0.0002	0.910	1.000

# Simulation Study - Twolevel WNFS Lag 5 model

```
montecarlo:
  names = y1-y5;
  NOBS = 10000;
  NREP = 100;
  NCSIZES = 1;
  CSIZES = 100(100);

ANALYSIS:  TYPE IS TWOLEVEL;
           estimator=bayes;
           biter=(500); proc=2;

model montecarlo:
  %within%
  f by y1-y5*1 (& 5);
  y1-y5*1;
  f@1;
  y1-y5 on f&1*0.4 f&2*0.2 f&3*0.3 f&4*0.2 f&5*0.1;

  %between%
  fb by y1-y5*0.5; fb@1; y1-y5*0.2;
```

# Results - Twolevel WNFS Lag 5 model

Y1	ON							
F&1		0.400	0.4012	0.0155	0.0160	0.0002	0.960	1.000
F&2		0.200	0.1989	0.0168	0.0161	0.0003	0.940	1.000
F&3		0.300	0.3021	0.0145	0.0165	0.0002	0.980	1.000
F&4		0.200	0.1998	0.0179	0.0167	0.0003	0.950	1.000
F&5		0.100	0.0985	0.0155	0.0164	0.0002	0.960	1.000

# Simulation Study - Twolevel DAFS-WNFS Combo Lag 1 model - ARMA(1,1) factor

```
montecarlo:
  names = y1-y5;
  NOBS = 10000;
  NREP = 100;
  NCSIZES = 1;
  CSIZES = 100(100);

ANALYSIS:  TYPE IS TWOLEVEL;
           estimator=bayes;
           biter=(500); proc=2;

model montecarlo:
  %within%
  f by y1-y5*1 (& 1);
  y1-y5*1;
  f@1;
  f on f&1*0.4;
  y1-y5 on f&1*0.6;

  %between%
  fb by y1-y5*0.5; fb@1; y1-y5*0.2;
```

# Results - Twolevel DAFS-WNFS Combo Lag 1 model - ARMA(1,1) factor

F	ON							
F&1		0.400	0.3992	0.0154	0.0139	0.0002	0.930	1.000
Y1	ON							
F&1		0.600	0.5980	0.0216	0.0218	0.0005	0.970	1.000
Y2	ON							
F&1		0.600	0.6020	0.0207	0.0220	0.0004	0.990	1.000
Y3	ON							
F&1		0.600	0.5999	0.0208	0.0219	0.0004	0.980	1.000
Y4	ON							
F&1		0.600	0.6006	0.0215	0.0219	0.0005	0.940	1.000
Y5	ON							
F&1		0.600	0.5985	0.0231	0.0219	0.0005	0.920	1.000

Note that this is counterintuitive from SEM perspective, but not from time series perspective. The model is essentially a factor analysis model with ARMA(1,1) factor

# Subject-specific times of observations

- The basic model assumes that observations are taken at equally spaced time.
- If times are subject-specific we slice the time grid in sufficiently refined grid and enter missing data for the times where observation is not taken.
- For example if several observations are taken during the day, and at different times for each individual, we slice the day in 24 hour periods and place the corresponding observations in the hour slots.
- Data from the next simulation looks like this for day 1 for individual 1 and 2.

# Subject-specific times of observations: subject 1 day 1

y1	y2	y3	y4	y5	T	ID	
999	999	999	999	999	999	1	1
999	999	999	999	999	999	2	1
999	999	999	999	999	999	3	1
999	999	999	999	999	999	4	1
999	999	999	999	999	999	5	1
999	999	999	999	999	999	6	1
999	999	999	999	999	999	7	1
999	999	999	999	999	999	8	1
999	999	999	999	999	999	9	1
999	999	999	999	999	999	10	1
999	999	999	999	999	999	11	1
999	999	999	999	999	999	12	1
5.026193	0.327383	1.017519	0.701296	-0.55917		13	1
999	999	999	999	999	999	14	1
1.628885	1.652829	2.324074	1.800932	4.013447		15	1
999	999	999	999	999	999	16	1
4.376545	1.652831	2.098822	6.188234	2.913506		17	1
1.534865	0.631455	-0.29779	2.798775	1.37025		18	1
0.359654	1.476764	-0.43374	0.348777	1.382437		19	1
999	999	999	999	999	999	20	1
999	999	999	999	999	999	21	1
999	999	999	999	999	999	22	1
999	999	999	999	999	999	23	1
999	999	999	999	999	999	24	1

# Subject-specific times of observations: subject 2 day 1

y1	y2	y3	y4	y5	T	ID
999	999	999	999	999	1	2
-4.10343	-2.63665	-0.71492	-1.18325	-2.38467	2	2
999	999	999	999	999	3	2
-1.49744	0.268575	-1.94601	-0.70893	-2.5558	4	2
999	999	999	999	999	5	2
999	999	999	999	999	6	2
999	999	999	999	999	7	2
999	999	999	999	999	8	2
0.158731	-0.49435	2.052861	2.387179	0.932945	9	2
999	999	999	999	999	10	2
999	999	999	999	999	11	2
999	999	999	999	999	12	2
999	999	999	999	999	13	2
999	999	999	999	999	14	2
999	999	999	999	999	15	2
999	999	999	999	999	16	2
999	999	999	999	999	17	2
999	999	999	999	999	18	2
-0.2193	-2.14029	0.050116	-0.42015	-1.00061	19	2
999	999	999	999	999	20	2
999	999	999	999	999	21	2
999	999	999	999	999	22	2
999	999	999	999	999	23	2
999	999	999	999	999	24	2

# Subject-specific times of observations - simulation study

```
montecarlo:
  names = y1-y5 u;
  NOBS = 30000;
  NREP = 100;
  NCSIZES = 1;
  CSIZES = 100(300);
  categorical=u;
  generate=u(1);
  within=u;
  missing=y1-y5;

model missing: [y1-y5@-15]; y1-y5 on u@30;

ANALYSIS: TYPE IS TWOLEVEL; estimator=bayes;
          biter=10000(500); proc=2;

model montecarlo:
  %within%
  [u$1*-0.83];
  f by y1-y5*1 (& 1);
  y1-y5*1;
  f@1;
  f on f&1*0.4;

  %between%
  fb by y1-y5*0.5; fb@1; y1-y5*0.2;
```

# Subject-specific times of observations - simulation study results

- 80% missing values, 20% present, in hourly scale that means 4 or 5 observations a day
- 99% convergence rate, 96 minutes computation for 100 replications, 1 min per replication

Within Level		Population	ESTIMATES Average	Std. Dev.	S. E. Average	M. S. E.	95% Cover	% Sig Coeff
F	BY							
	Y1	1.000	1.0015	0.0166	0.0180	0.0003	0.990	1.000
	Y2	1.000	1.0017	0.0172	0.0178	0.0003	0.949	1.000
	Y3	1.000	1.0001	0.0183	0.0179	0.0003	0.960	1.000
	Y4	1.000	1.0034	0.0157	0.0180	0.0003	0.970	1.000
	Y5	1.000	1.0010	0.0192	0.0180	0.0004	0.939	1.000
F	ON							
	F&1	0.400	0.3981	0.0234	0.0211	0.0005	0.889	1.000

# Subject-specific times of observations - summary of results

- Similar results for 85% missing and 90% missing, however, convergence is visibly slower for 90% missing values than for 80% missing values and non-convergence is more likely
- It appears that optimal time discretization would be about 80% to 85% missing values inserted to represent the distance between the observations
- Very likely any information contained in the unequal distances in the observations would be extracted well using the 80% to 85% missing values.
- It appears that this problem is solved with this simple setup and there is no need to develop Brownian motion continuous modeling.
- Mplus now implements a utility that will setup the missing data for you, given the precise times of observations

# Subject-specific times of observations algorithm

- $t_{\text{interval}} = t(0.1)$ ;
- Split the time axis in bins of size 0.1. Then place each observation in the correct bin
- Repeat these steps until each bin contains no more than 1 observation
  - find a bin with more than 1 observations
  - locate the nearest empty bin (look up or down)
  - move one of the extra observation to fill in the the empty bin but keep order of the observations so the extra observation bumps the rest of the observations towards the empty bin
- Fill in the remaining bins with missing values and set the time as  $T=1,2, \dots$  and  $T$  is the bin number.
- Other algorithms are possible. The algorithm discretizes the time scale and approximates the times of observations with times from the discretized grid.

## Simulated Example Results. Continuous time modeling

- Using one generated data set from the 80% missing simulation. Consider the results for the autocorrelation parameter under various analysis. True value=0.4. Note that AR meaning depends on the time interval between observations. Note that the true grid value that we know is sufficient is 1. Note that the model is almost the same as the empirical example.

Grid size	$\rho$	implied	$ \rho - \text{implied} $
0.5	0.633	-	-
1	0.398	0.401	0.003
1.5	0.303	0.254	0.049
2	0.229	0.161	0.068
ignore time	0.102	-	

- You can see in this controlled experiment that "implied - estimated" quantity clearly discovers the correct grid size

## Two-level AR(1) TVEM - simulated example

- General TVEM framework: multivariate, multilevel, time-series, continuous and categorical dependent variables.
- Consider the following example  $N=500$ ,  $T=50$

$$Y_{it} = \mu_t + Y_i + F_{it}$$

$$F_{it} = \rho F_{i,t-1} + \beta_t X_{it} + \varepsilon_{it}$$

$$\mu_t = f_1(t) = \log(t)$$

$$\beta_t = f_2(t) = a + bt + ct^2 = 0.001 \cdot t \cdot (50 - t)$$

- $f_1(t)$  and  $f_2(t)$  are arbitrary functions of  $t$  - no specific functional form is assumed in the model

# Two-level AR(1) TVEM - input file

```
variable:
    names are y x t c;
    within=x;
    cluster=c t;

data:file=new1.dat;

ANALYSIS: TYPE = CROSS random;
estimator=bayes; biter=(5000); process=2;

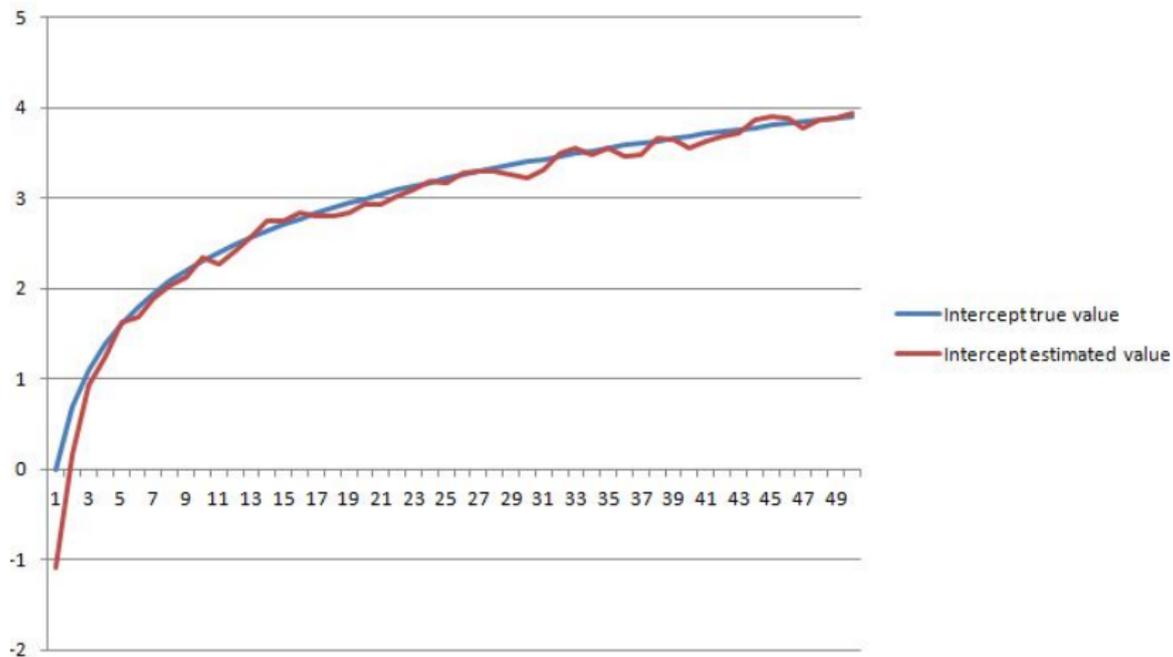
model:
    %within%
    y@0.05; eta by y@1 (&1);
    eta on eta&1*0.5; eta*1.2;
    s | eta on x;

    %between t%
    y; s; [s];

    %between c%
    y*0.5; [y*0]; s@0; [s@0];

savedata: file is TVEM.dat; save=FS(100);
```

# Two-level AR(1) TVEM - intercept results



# Two-level AR(1) TVEM - slope results

