Hierarchical Bayesian Dynamic Structural Equation Models: A Tutorial in Stan

FGME - 11 September, 2023

Jean-Paul Snijder[a], Valentin Pratz[a], and Anna-Lena Schubert[b]
[a]University of Heidelberg, [b]University of Mainz

1 Dynamic Structural Equation Modeling (DSEM)

Introduction

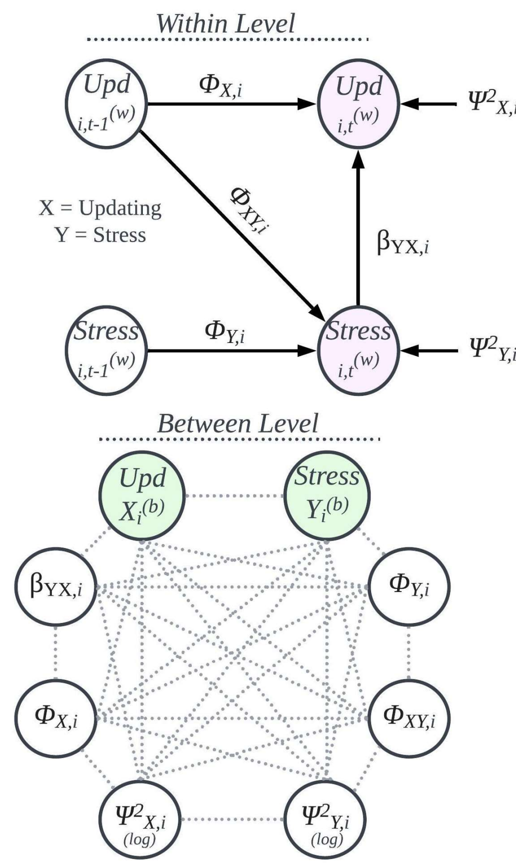
- Technological advancements are increasing availability of Intensive Longitudinal Data (ILD) from:
  - Experience Sampling Methods (ESM, EMA, AA)
  - Electro-EncephaloGram (EEG)
  - Wearables
- ILD are densely spaced repeated measures data collected from large samples
- Need for models that allow to examine dynamic changes over time
- Computational models are developed and adapted to match this growing demand

4

Dynamic Structural Equation Modeling

Combines: [1]

- **Time-series modeling**
  - *allows lagged relationships*
- **Multilevel modeling**
  - *allows modeling of nested data structures*
- **Structural equation modeling**
  - *allows latent variable/path analysis*



1. Hamaker et al. (2023)

DSEM in Mplus

Pros

- Widely used
- online user and program support
- Considered user-friendly
- Low computational time
  - Gibbs sampler with conjugate priors (Normal ⇔ Inverse Wishart)

Cons

- Not fully customizable
- currently doesn't support some model extensions and specifications
- limited prior options and access to sampler settings
  - *i.e., no LKJ distribution*
- Limited options for missing data
- License costs money

Stan

Pros

- Free
- Fully customizable
- Open Code & Reproducible Science
- Online community support
- Hamiltonian Monte Carlo
  - Efficient general-purpose MCMC sampler

Cons

- Programming can pose a barrier
  - *Fully code-based*
  - *No GUI*
- Higher computational time
  - *but reasonable (minutes to hours)*
  - *not optimized for a specific model family*

7

DSEM software alternatives

- `dlsem` in R:
  - Uses frequentist inference
- `ctsem` in R:
  - Slow for full Bayesian estimation
  - Oriented towards continuous time systems
    - *but discrete can be used*
  - Less user-friendly
  - No latent classes and limited non-continuous measurement models
- `JAGS`

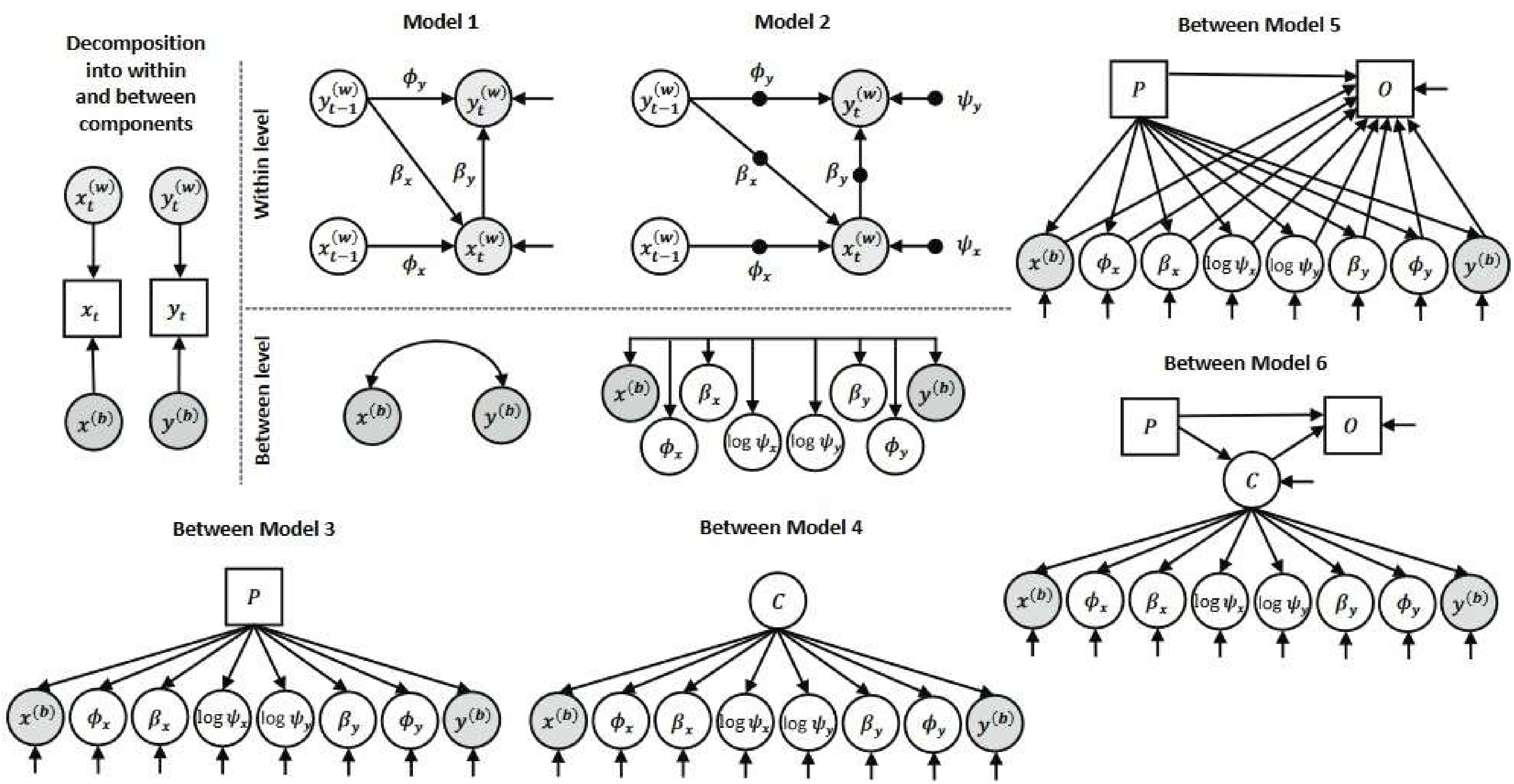
Our project

- Stan tutorial using DSEM framework as example
  1. Introducing DSEM
  2. Improving the accessibility to Stan

- **6 model archetypes**[1]
  1. Bivariate, Single Case
  2. **Bivariate, Multilevel**
  3. Model 2 + predictor variable
  4. Model 2 + latent variable
  5. Model 3 + outcome variable
  6. Model 4 + mediation

1. Hamaker et al. (2023)

Our project



Taken from [1] *slightly altered for fit*

1. Hamaker et al. (2023)

Our project

- Stan tutorial using DSEM framework as example
  1. Introducing DSEM
  2. Improving the accessibility to Stan

- **6 model archetypes**[1]
  1. Bivariate, Single Case
  2. **Bivariate, Multilevel**
  3. Model 2 + predictor variable
  4. Model 2 + latent variable
  5. Model 3 + outcome variable
  6. Model 4 + mediation

- **For each archetype in Stan:**
  1. *Simple:* tutorial model
  2. *Reparam:* reparameterized model
  3. *Full:* missing data model

1. Hamaker et al. (2023)

2 DSEM

13

M2: Two Variable, Multilevel Model

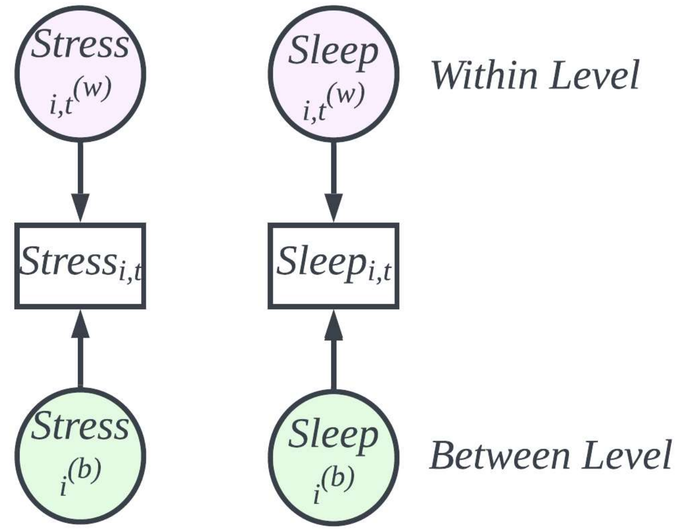
- Model 2: two variables + multilevel
  - Stress
  - Sleep

M2: Two Variable, Multilevel Model

- Model 2: two variables + multilevel
  - Stress
  - Sleep

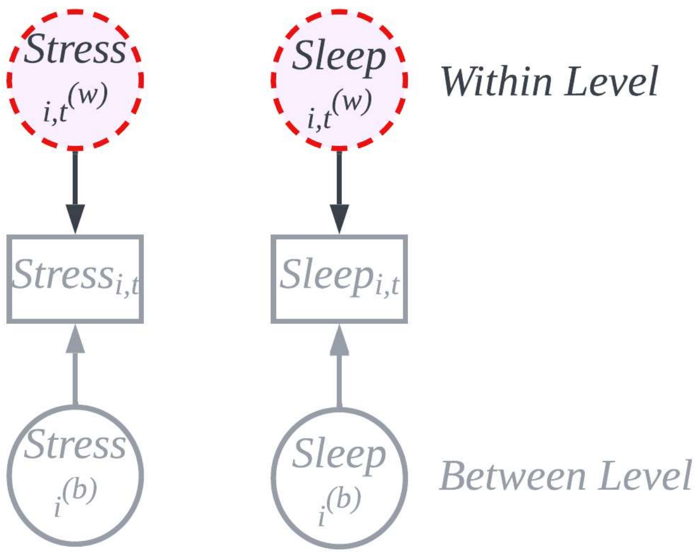
- Within- & between-person decomposition
  - *Between: time-**insensitive** mean of subject*
  - *Within: time-**sensitive** deviation from that mean*

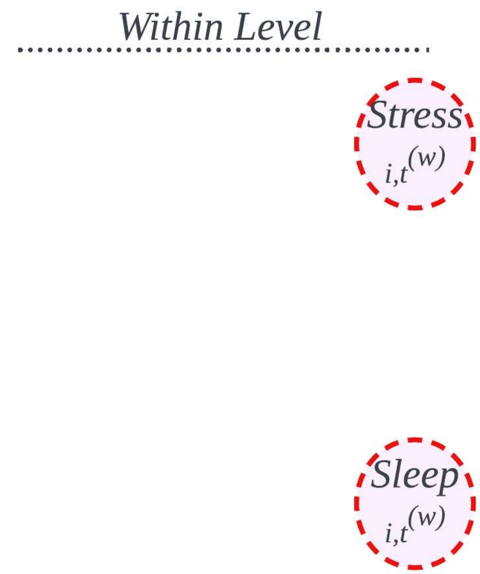
- Allows for specifying time-*dynamics* in within-person model



M2: Within-person Model I

- The decomposed within-person variables are the start of the within-person model $\longrightarrow$

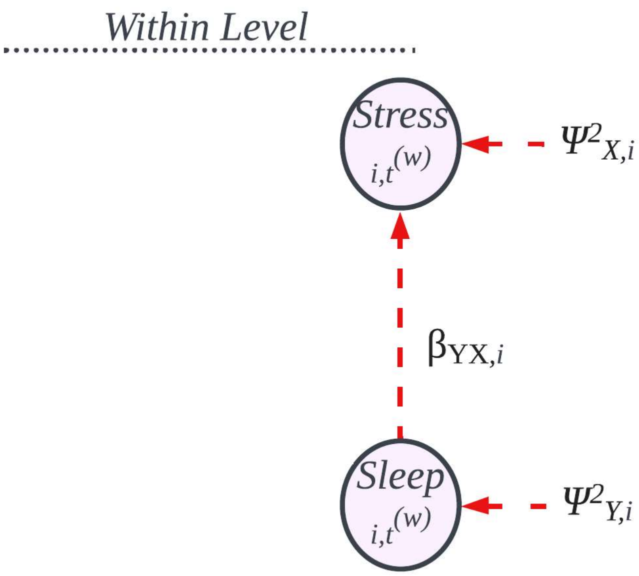




M2: Within-person Model II

Relationships & Parameters:

- Regression:
  - β_{YX} = Stress_t regressed on Sleep_t

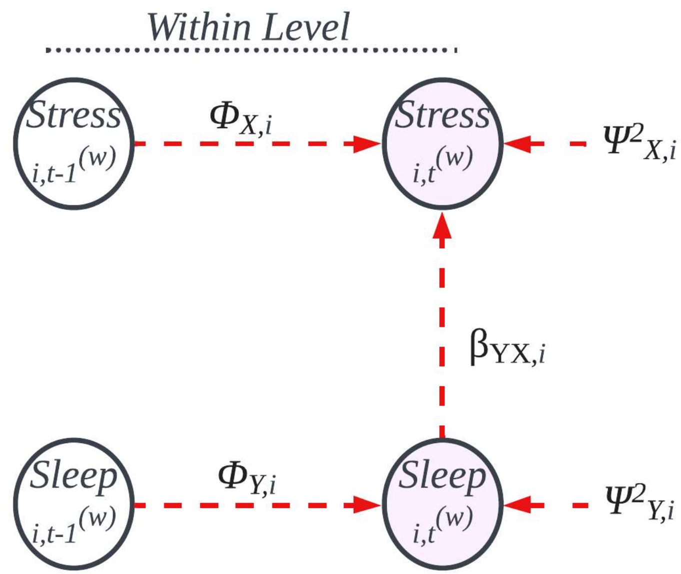


17

M2: Within-person Model II

Relationships & Parameters:

- Regression:
  - β_{YX} = Stress_t regressed on Sleep_t
- Time Dynamic Regressions:
  - $\Phi_{X,i}$ = auto-regressive parameter Stress
  - $\Phi_{Y,i}$ = auto-regressive parameter Sleep
    - $Stress_{i,t\text{-}1}^{(w)}$ and $Sleep_{i,t\text{-}1}^{(w)}$ are lag(1) variables
    - E.g., *if* $_t$ = *observation 9* $\Rightarrow$ $_{t\text{-}1}$ = *observation 8 …*

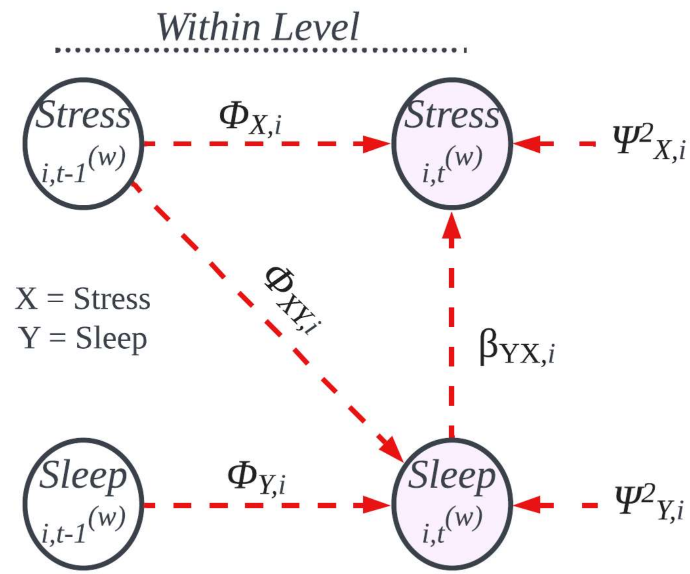


18

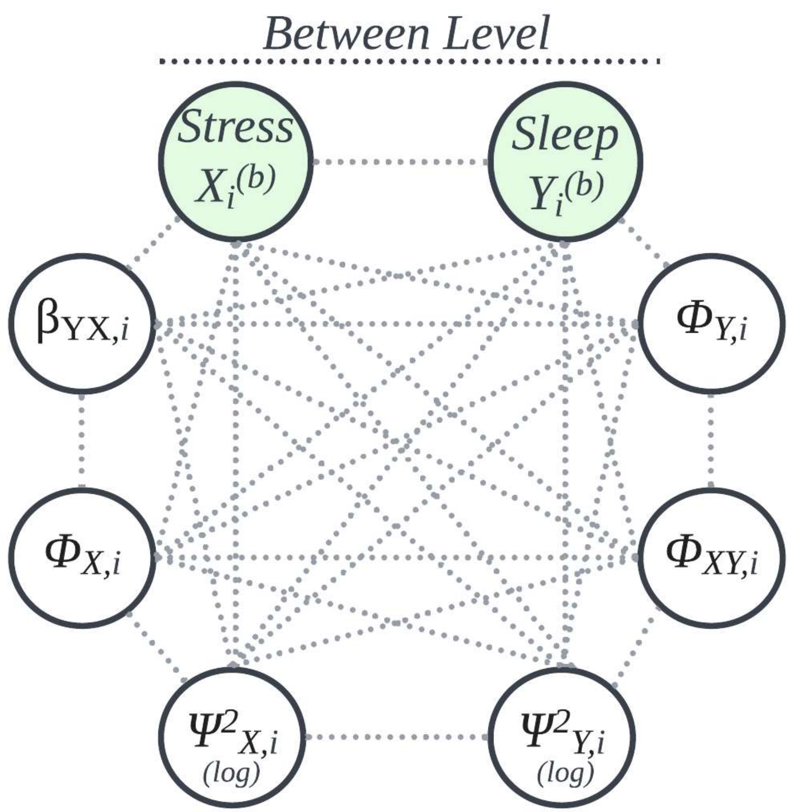
M2: Within-person Model II

Relationships & Parameters:

- Regression:
  - β_{YX} = Stress$_t$ regressed on Sleep$_t$
- Time Dynamic Regressions:
  - $\Phi_{X,i}$ = auto-regressive parameter Stress
  - $\Phi_{Y,i}$ = auto-regressive parameter Sleep
  - $\Phi_{XY,i}$ = cross-regressive parameter Sleep$_{i,t}$ onto Stress$_{i,t-1}$
- Residual variances:
  - $\Psi^2{}_{X,i}$ and $\Psi^2{}_{Y,i}$

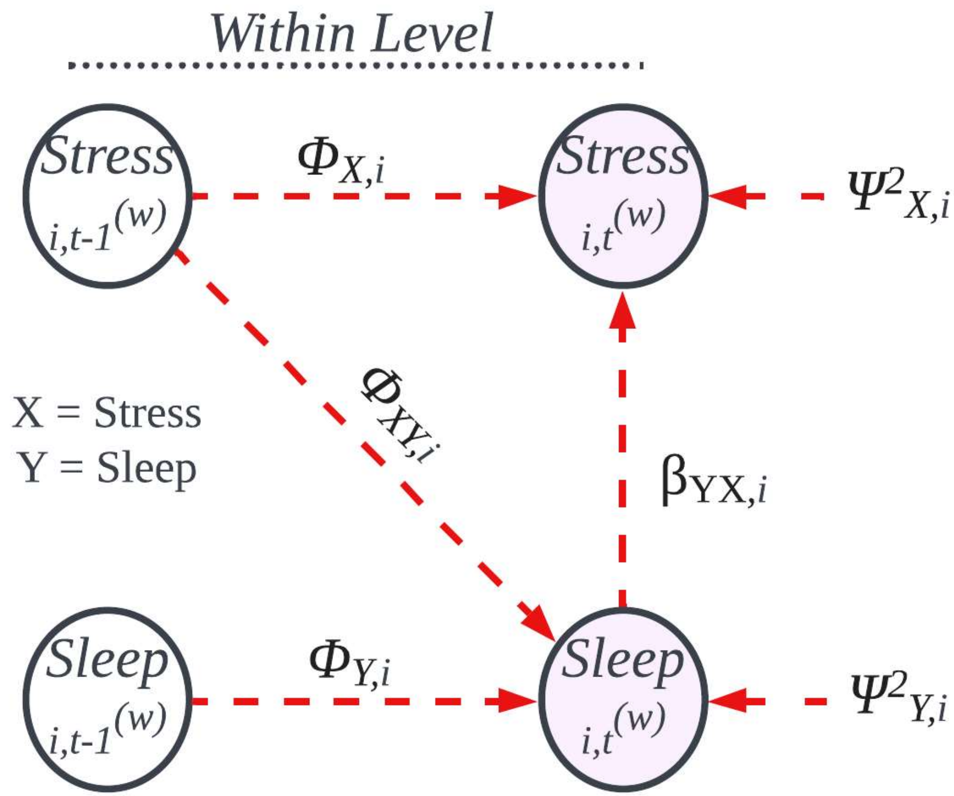


M2: Between-person Model



3 Stan

Within-level model I



$$Stress_{i,t}^{(w)} = \mathcal{N}([\Phi_{X,i}][Stress_{i,t-1}^{(w)}] + [\beta_{YX,i}][Sleep_{i,t}^{(w)}], \Psi_{X,i}^2)$$

$$Sleep_{i,t}^{(w)} = \mathcal{N}([\Phi_{Y,i}][Sleep_{i,t-1}^{(w)}] + [\Phi_{XY,i}][Stress_{i,t-1}^{(w)}], \Psi_{Y,i}^2)$$

23

Within-level model II

```
Stress_t ~ normal(phi_X * Stress_t_1 + beta_YX * Sleep_t, psi_X);
Sleep_t ~ normal(phi_Y * Sleep_t_1 + phi_XY * Stress_t_1, psi_Y);
```

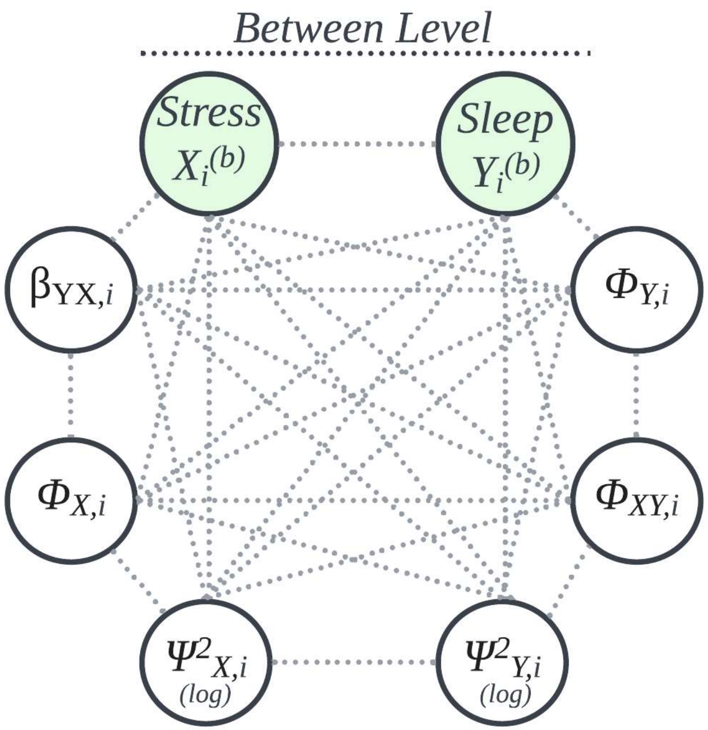
$$Stress^{(w)}_{i,t} = \mathcal{N}([\Phi_{X,i}][Stress^{(w)}_{i,t-1}] + [\beta_{YX,i}][Sleep^{(w)}_{i,t}], \Psi^2_{X,i})$$

$$Sleep^{(w)}_{i,t} = \mathcal{N}([\Phi_{Y,i}][Sleep^{(w)}_{i,t-1}] + [\Phi_{XY,i}][Stress^{(w)}_{i,t-1}], \Psi^2_{Y,i})$$

24

Between-level model I

- Using latent means and random intercepts/effects



$$
\begin{aligned}
X_i^{(b)} &= \gamma_1 + u_{i1} \\
Y_i^{(b)} &= \gamma_2 + u_{i2} \\
\Phi_{Xi} &= \gamma_3 + u_{i3} \\
\Phi_{Yi} &= \gamma_4 + u_{i4} \\
\Phi_{XYi} &= \gamma_5 + u_{i5} \\
\beta_{YXi} &= \gamma_6 + u_{i6} \\
\log \Psi^2_{Xi} &= \gamma_7 + u_{i7} \\
\log \Psi^2_{Yi} &= \gamma_8 + u_{i8}
\end{aligned}
$$

$$\boldsymbol{u} \sim \mathrm{MVNormal}(\mathbf{0}, \boldsymbol{\Omega})$$

Between-level model II

- Using latent means and random intercepts/effects

```
real mu_X = gamma[1] + u[i,1];
real mu_Y = gamma[2] + u[i,2];

real phi_X = gamma[3] + u[i,3];
real phi_Y = gamma[4] + u[i,4];
real phi_XY = gamma[5] + u[i,5];
real beta_YX = gamma[6] + u[i,6];

real psi_X = sqrt(exp(gamma[7] + u[i,7]));
real psi_Y = sqrt(exp(gamma[8] + u[i,8]));

u[i] ~ multi_normal(rep_vector(0, 8), Omega);
```

$$
\begin{aligned}
X_i^{(b)} &= \gamma_1 + u_{i1} \\
Y_i^{(b)} &= \gamma_2 + u_{i2} \\
\Phi_{Xi} &= \gamma_3 + u_{i3} \\
\Phi_{Yi} &= \gamma_4 + u_{i4} \\
\Phi_{XYi} &= \gamma_5 + u_{i5} \\
\beta_{YXi} &= \gamma_6 + u_{i6} \\
\log \Psi_{Xi}^2 &= \gamma_7 + u_{i7} \\
\log \Psi_{Yi}^2 &= \gamma_8 + u_{i8}
\end{aligned}
$$

$$\boldsymbol{u} \sim \text{MVNormal}(\boldsymbol{0}, \boldsymbol{\Omega})$$

26

Optimization: Reparameterization

- Improves convergence, can speed up sampling
- Classical example: $y \sim \mathcal{N}(\mu, \sigma^2) \Leftrightarrow y = \mu + \sigma \cdot \tilde{y} \text{ with } \tilde{y} \sim \mathcal{N}(0, 1)$

Handling missing data

- Missing data is unknown
- Parameters are unknown

$\longrightarrow$ treat missing data like parameters

- Preserves uncertainty (unlike mean imputation etc.)

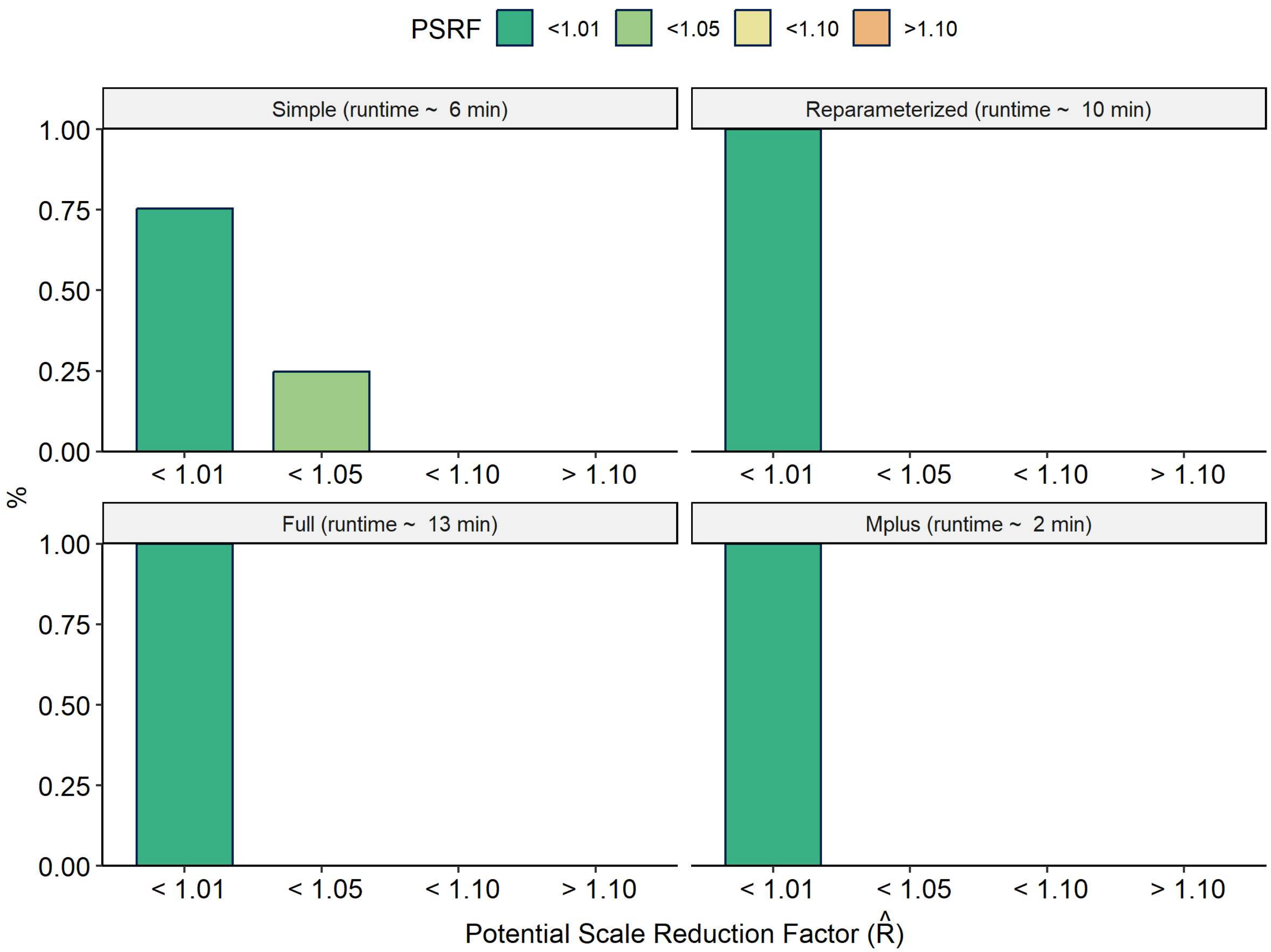
4 Simulation

Results with simulated data

- Model 2
- 100 subjects
- 100 observations
- relevant parameter ranges for sleep and stress
- for missing data model: 5% missingness
- no model misspecification
- Sampler:
  - 500 warmup/3500 sampling iterations
  - 4 chains, 16 cores

31

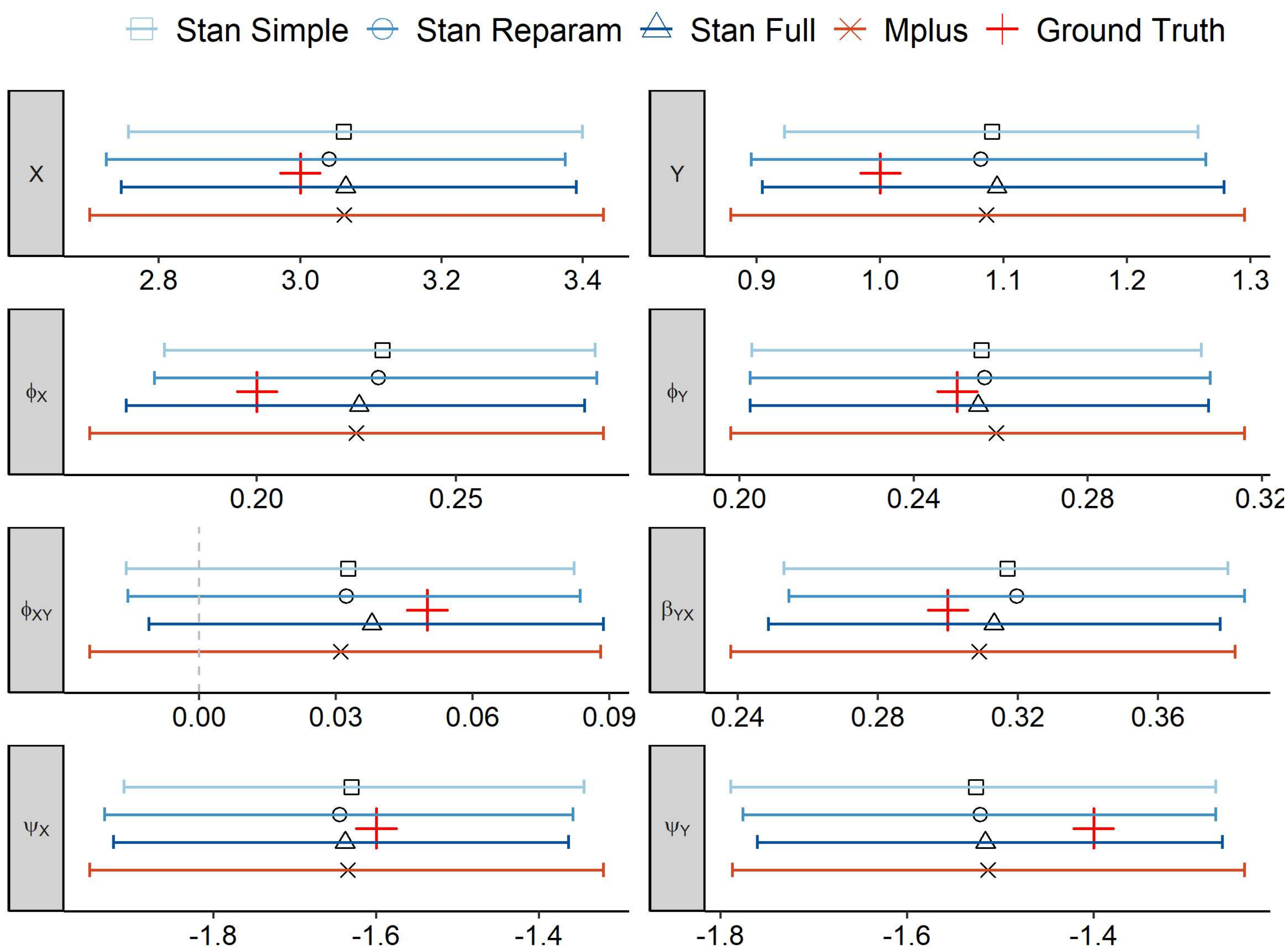
Model convergence



Model 2, simulated data convergence.

32

Parameter recovery



Model 2, simulated data. Errorbars: 95% CI.

5 Discussion

Future

- *current:* Simulations
  - relevant parameter ranges
  - prior calibration
- *near:* Standardized estimates
- *near:* Model implementation for cognitive behavioral tasks
- *far:* R Package with Stan as back-end

Thanks to

- **Ellen Hamaker** for instructional material on DSEM
- **Mauricio Garnier-Villarreal** (blavaan) for sharing his Stan knowledge online
- The **Stan community** for educational material on reparameterization and other tricks
- Valentin Pratz, student assistant

Thank you

Questions?

Github repo with presentation + reproducable model 2 example

References

Hamaker, E. L., Asparouhov, T., & Muthén, B. (2023). Dynamic Structural Equation Modeling as a Combination of Time Series Modeling, Multilevel Modeling, and Structural Equation Modeling. In R. H. Hoyle (Ed.), *Handbook of structural equation modeling* (2nd ed., pp. 576–597). New York: Guilford Press.

Schmiedek, F., Lövdén, M., & Lindenberger, U. (2010). Hundred Days of Cognitive Training Enhance Broad Cognitive Abilities in Adulthood: Findings from the COGITO Study. *Frontiers in Aging Neuroscience*, *2*, 27. https://doi.org/10.3389/fnagi.2010.00027