Preregistration: SEMi-Complete by Design: A Monte Carlo simulation to assess Measurement Invariance in Moderated Nonlinear Factor Analysis and SEM Trees

Leonie Hagitte[1,2], Andreas M. Brandmaier[2]

December 4, 2025

Version: 1.1
Last updated: 2024-11-18
Template based on:
Siepe, B. S., Bartoš, F., Morris, T. P., Boulesteix, A.-L., Heck, D. W., & Pawel, S. (2024). Simulation Studies for Methodological Research in Psychology: A Standardized Template for Planning, Preregistration, and Reporting. *Psychological Methods*. https://doi.org/10.1037/met0000695, https://doi.org/10.31234/osf.io/ufgy6

[1]Center for Lifespan Psychology, Max Planck Institute for Human Development, Berlin, Germany; The International Max Planck Research School on the Life Course, Berlin, Germany

[2]Department of Psychology, MSB Medical School Berlin, Berlin, Germany

1 General Information

1.1 What is the title of the project?

SEMi-Complete by Design: A Monte Carlo simulation to assess Measurement Invariance in Moderated Nonlinear Factor Analysis and SEM-Trees.

1.2 Who are the current and future project contributors?

Leonie Hagitte, Andreas M. Brandmaier

1.3 Provide a description of the project.

Ensuring the validity of psychological assessments is crucial, yet differential item functioning (DIF) can threaten measurement invariance (MI) when test items function differently across groups (**Bauer2020**). Recent calls for improved DIF detection methods emphasize the need for more advanced statistical approaches (**Lee2024**). Moderated nonlinear factor analysis (MNLFA) is a recent approach for assessing MI via parameter moderation within a single-group confirmatory factor analysis framework. MLNFA evaluates MI across multiple continuous and categorical covariates, and accounts for heteroskedasticity by modeling factor and residual variances as functions of these covariates. While MNLFA offers continuous moderation of several parameters of SEMs (e.g.: factor loadings, covariances etc.), it requires a priori specification of covariates and their functional relationships (**Bauer2017**; **Kolbe2024**). In contrast, structural equation modeling (SEM) trees and forests are data-driven, non-parametric methods that use recursive partitioning to identify latent subgroups in which model parameters differ, without assuming specific functional forms or predefined covariate effects. These approaches allow for nonlinear moderation of factor loadings and can reveal complex interaction effects, enabling the exploratory detection of DIF (**Brandmaier2016**; **Brandmaier2013**). In this study, we conducted a Monte Carlo simulation to compare the performance of MNLFA and SEM trees and forests in detecting DIF and assessing MI under varying conditions. Specifically, we evaluate their effectiveness in identifying non-invariance and detecting relevant covariates. Our findings will inform best practices for selecting statistical techniques to test MI in psychological assessment.

1.4 Did any of the contributors already conduct related simulation studies on this specific question?

Neither of the authors has conducted simulation studies that are of immediate relevance to the current project before. However, Andreas Brandmaier was involved in conducting simulation studies on SEM trees/ SEM forests as well as SEM modeling in general (**Buchberger2024**; **SilvaDaz2025**; **Arnold2021**).

2 Aims

2.1 What is the aim of the simulation study?

The aim of this simulation study is to evaluate different methods (i.e. MNLFA and SEM trees/ SEM forests) regarding their accuracy, power, and false-positive rate in detecting MI across different factor models (ADEMP category 'hypothesis testing').

3 Data-Generating Mechanism

3.1 How will the parameters for the data-generating mechanism (DGM) be specified?

The data will be generated parametrically. Different population structural equation models (SEM) with latent variables and continuous indicators will be simulated in two different simulation studies:

3.1.1 Study 1

Single-Factor Model with Moderated Parameters
We specify a confirmatory factor analysis model (model 1) with a single latent factor η and four observed indicators $\mathbf{x} = (x_1, \ldots, x_4)^\top$. The influence of a continuous moderator is introduced into selected parameters via predefined transformation functions applied to an underlying raw moderator variable. The population model is:

$$\mathbf{x} = \boldsymbol{\nu}(Z) + \boldsymbol{\Lambda}_x(Z) \cdot \eta + \boldsymbol{\varepsilon}(Z)$$

where:

- $\boldsymbol{\nu}(Z)$ is the vector of intercepts,
- $\boldsymbol{\Lambda}_x(Z)$ is the 4×1 factor loading matrix,
- $\boldsymbol{\varepsilon}(Z) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Theta}_\varepsilon(Z))$ is the residual vector,
- $\eta \sim \mathcal{N}(\mu_\eta(Z), \sigma^2_\eta(Z))$ is the latent factor,
- $Z = h(M)$ is the effective moderator entering the model equations, obtained by transforming a raw continuous covariate M.

Moderator Transformation
Let $M \sim \mathcal{N}(0, 1)$ denote a raw continuous covariate. The effective moderator is defined as

$$Z = h(M),$$

where $h(\cdot)$ is chosen from the following deterministic transformation functions. Each transformation type is treated as a distinct condition in the simulation design:

1. **Linear:**
$$h(M) = M.$$
In this case, the effective moderator is unbounded and symmetric around zero.

2. **Sigmoid:**
$$h(M) = 2 \cdot \text{logit}^{-1}(\text{slope} \cdot M) - 1,$$
with a fixed slope parameter (e.g., `sigmoid_slope_1` in the implementation). This transformation maps the real line to the open interval $(-1, 1)$ and yields a bounded moderator symmetric around zero.

3. **Quadratic:**
$$h(M) = M^2.$$
Here, the effective moderator takes values on $[0, \infty)$ and is therefore non-negative and right-skewed, capturing nonlinear but strictly non-negative moderation patterns.

4. **Noise (non-informative):**
$$h(M) = 0.$$
In this condition, the moderator has no effect on any model parameter and serves as a non-informative (noise) moderator.

Factor Loadings

The baseline factor loadings were fixed to a discrete value of $\lambda_{xi} = 0.7$ for all items exept item 4, which was used for scaling. Variation in loadings across simulation conditions arises exclusively through the presence or absence of moderation, which is treated as a fully crossed factor in the grid design.

Moderated items:

At the population level, the baseline (unmoderated) measurement model is specified with a single latent factor and four indicators. The factor loading matrix is constructed such that the first three indicators share a common loading value λ, whereas the fourth indicator is fixed to 1.0 to set the scale of the latent factor:

$$\mathbf{\Lambda}_x = \begin{bmatrix} \lambda \\ \lambda \\ \lambda \\ 1.0 \end{bmatrix}.$$

The latent factor variance is fixed at a prespecified value $\psi_\eta = \psi_{\text{var}}$, and baseline residual variances are derived analytically from a single target reliability level applied uniformly across all indicators. For each item,

$$\theta_i = \frac{\lambda_i^2 \, \psi_\eta}{\text{Reliability}} - \lambda_i^2 \, \psi_\eta,$$

yielding a diagonal residual covariance matrix

$$\mathbf{\Theta}_\varepsilon = \text{diag}(\theta_1, \theta_2, \theta_3, \theta_4).$$

This baseline specification sets the measurement scale (via the reference loading), ensures positive residual variances for all indicators, and serves as the starting point upon which moderation effects are introduced in subsequent conditions.
For each condition involving moderation, the loading of item i is defined as a deterministic function of the moderator Z:

$$\lambda_{xi}(Z) = \lambda_{xi} + \delta_{\lambda_i} Z, \quad Z \in [-1, 1].$$

The parameter δ_{λ_i} represents the strength of the moderation effect and is specified using a discrete set of admissible values chosen to ensure that the moderated loading remains within a psychometrically plausible interval:

$$\lambda_{xi}(Z) \in [0.3,\ 1.0].$$

Given the fixed baseline loading $\lambda_{xi} = 0.7$, this constraint yields an admissible range

$$\delta_{\lambda_i} \in [-0.4,\ 0.3].$$

To comply with the grid-search framework, δ_{λ_i} is not sampled but instead evaluated at the following discrete levels:

$$\delta_{\lambda_i} \in \{-0.4,\ -0.2,\ 0,\ 0.2,\ 0.3\}.$$

These values span the full admissible range while maintaining deterministic, reproducible simulation conditions.

For moderator transformations beyond the linear case (quadratic or sigmoid), the functional form of $\lambda_{xi}(Z)$ is adapted accordingly, and the same discrete set of admissible moderation strengths is applied. This yields fully crossed combinations of (i) type of moderator transformation, (ii) number of moderated items, and (iii) moderation strength.

Factor loading matrix:

$$\mathbf{\Lambda}_x(Z) = \begin{bmatrix} \lambda_1(Z) \\ \lambda_2(Z) \\ \lambda_3 \\ \lambda_4 \end{bmatrix}$$

Here, $\lambda_1(Z)$ and $\lambda_2(Z)$ are either moderated or unmoderated depending on the corresponding grid condition, whereas λ_3 and λ_4 remain fixed across all conditions. The structure of $\mathbf{\Lambda}_x(Z)$ is therefore determined jointly by the moderation-design factors and the functional form of Z.

Moderation effects are applied at the case level by transforming an individual-level raw moderator M into an effective moderator $Z = h(M)$ according to the transformation functions described above. For each generated observation, the raw moderator is drawn independently as

$$M \sim \mathcal{N}(0, 1),$$

and the effective moderator is computed as $Z = h(M)$.
The resulting distributions of Z differ systematically across transformation types:

- **Linear:** $Z = M$, yielding an unbounded moderator symmetric around zero.
- **Sigmoid:** $Z = 2 \cdot \text{logit}^{-1}(\text{slope} \cdot M) - 1$, yielding a bounded moderator in $(-1, 1)$.
- **Quadratic:** $Z = M^2$, yielding a non-negative moderator with right-skewed distribution concentrated near zero.
- **Noise:** $Z = 0$, yielding a non-informative moderator.

For each observation, the effective moderator Z is then inserted into the parameter functions of the measurement model. Two moderation conditions are implemented:

1. **Full moderation (Model 1.1):** All factor loadings and all residual variances vary with Z:
$$\lambda_{xi}(Z) = \lambda_{xi} + \delta_\lambda Z, \qquad \theta_i(Z) = \theta_{i0}\bigl(1 + \delta_\theta Z\bigr),$$
where θ_{i0} denotes the baseline residual variance derived from the reliability grid (see below). The multiplicative scaling ensures $\theta_i(Z) > 0$ for all cases, provided $1 + \delta_\theta Z > 0$.

2. **Partial moderation (Model 1.2):** Only the loadings of items 1 and 2 vary with the moderator:
$$\lambda_{x1}(Z) = \lambda_{x1} + \delta_\lambda Z, \qquad \lambda_{x2}(Z) = \lambda_{x2} + \delta_\lambda Z,$$
whereas all residual variances remain fixed at their baseline values θ_{i0}.

The case-wise parameter matrices $\mathbf{\Lambda}_x(Z)$ and $\mathbf{\Theta}_\varepsilon(Z)$ are therefore obtained by evaluating these functions for each individual subject. Given the magnitudes of the moderation coefficients used in the design, these transformations keep all loadings and residual variances within admissible psychometric ranges across all values of Z, ensuring numerically stable and well-defined population covariance structures.

Residual Variances

Item reliabilities were varied deterministically across simulation conditions using the fixed grid
$$\text{Reliability}_i \in \{0.60,\ 0.70,\ 0.80,\ 0.95\}.$$

For each indicator, a baseline residual variance θ_{i0} is obtained analytically from the standard reliability expression,

$$\text{Reliability}_i = \frac{\lambda_{xi}^2}{\lambda_{xi}^2 + \theta_{i0}} \quad \Longrightarrow \quad \theta_{i0} = \frac{\lambda_{xi}^2(1 - \text{Reliability}_i)}{\text{Reliability}_i}.$$

These baseline values define the unmoderated diagonal residual covariance matrix. Residual-variance moderation is then introduced multiplicatively as

$$\theta_i(Z) = \theta_{i0}\bigl(1 + \delta_\theta Z\bigr),$$

where δ_θ denotes the residual-variance moderation coefficient. For items not subject to moderation, $\delta_\theta = 0$ and thus $\theta_i(Z) = \theta_{i0}$.

This specification combines (a) a reliability-based baseline θ_{i0} with (b) multiplicative moderation, yielding residual variances that remain positive and interpretable across all grid conditions.

Intercepts
Intercept moderation is manipulated via two fully crossed factors: (i) the pattern of moderated items and (ii) the magnitude of the moderation effect. For each indicator i, the intercept is modeled as

$$\nu_i(Z) = \nu_i + \delta_{\nu_i} Z,$$

where ν_i denotes the baseline intercept and δ_{ν_i} controls the strength and direction of intercept moderation. The pattern of moderated items is varied across three levels:

1. **No intercept moderation:** $\delta_{\nu_i} = 0$ for all $i = 1, \dots, 4$, such that
$$\nu_i(Z) = \nu_i \quad \text{for all } i.$$
2. **Partial intercept moderation (items 1 and 2):**
$$\delta_{\nu_i} \in \{-1.0,\ -0.5,\ 0.0,\ 0.5,\ 1.0\} \quad \text{for } i \in \{1,2\}, \qquad \delta_{\nu_i} = 0 \quad \text{for } i \in \{3,4\},$$
such that only $\nu_1(Z)$ and $\nu_2(Z)$ vary with Z.
3. **Full intercept moderation (all items):**
$$\delta_{\nu_i} \in \{-1.0,\ -0.5,\ 0.0,\ 0.5,\ 1.0\} \quad \text{for } i = 1, \dots, 4,$$
such that the intercepts of all four indicators vary with Z.

Baseline intercepts are varied deterministically across simulation conditions using the fixed grid

$$\nu_i \in \mathcal{G}_\nu = \{-1,\ 0,\ 1\},$$

providing low, medium, and high intercept levels centered around zero. The simulation design thus yields fully crossed combinations of (a) intercept moderation pattern (none, partial, full), (b) moderation strength levels δ_{ν_i}, and (c) baseline intercept levels ν_i.

Latent Distribution
Latent mean moderation was specified deterministically as:

$$\mu_\eta(Z) = \alpha + \delta_\eta Z,$$

with both parameters selected from discrete sets:

$$\alpha \in \{0\}, \qquad \delta_\eta \in \{-1.0,\ -0.5,\ 0,\ 0.5,\ 1.0\}.$$

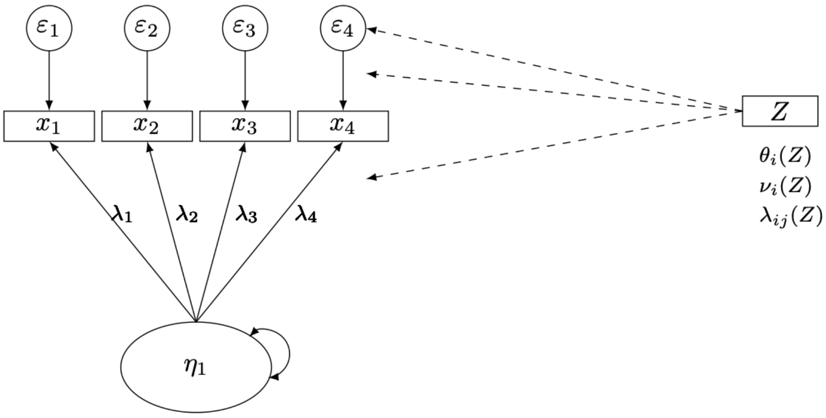
Latent variance moderation was modeled using the log-linear form:

$$\sigma^2_\eta(Z) = \exp(\beta_0 + \beta_1 Z),$$

where the parameters are varied over a fully specified grid rather than sampled from continuous distributions:

$$\beta_0 \in \{-0.5,\ 0,\ 0.5\}, \qquad \beta_1 \in \{-1.0,\ -0.5,\ 0,\ 0.5,\ 1.0\}.$$

Analytical Model Study 1



The analytical model will also have one latent factor η_1. With four manifest indicators $x_1, ..., x_4$, their factor loadings $\lambda_1, ..., \lambda_4$ and their respective residuals $\epsilon_1, ..., \epsilon_4$. In the analytical model the residuals will be assumed and modeled to be uncorrelated. Furthermore, the model includes a moderator variable Z, which can moderate the residuals, intercepts and factor loadings from none, one or two of the manifest variables. The moderator variable Z will be modeled as linear or quadratic.

3.1.2 Study 2

Two Factors with Moderator Effects We specify a confirmatory factor analysis model with two latent factors, η_1 and η_2, and seven observed indicators $\mathbf{x} = (x_1, \ldots, x_4)^\top$ and $\mathbf{y} = (y_1, \ldots, y_3)^\top$. Along their respective residuals $\epsilon_1, ..., \epsilon_4$ and $\delta_1, ..., \delta_3$. The first factor loads onto indicators x_1 to x_4, the second onto y_1 to y_3. A continuous moderator variable Z introduces conditional heterogeneity into selected measurement parameters (i.e. residuals, intercepts, loadings and latent factor covariance) using one of three functional forms (linear, quadratic or sigmoid).

$$\mathbf{x} = \boldsymbol{\nu}_x(Z) + \boldsymbol{\Lambda}_x(Z)\,\eta_1 + \boldsymbol{\varepsilon}(Z),$$
$$\mathbf{y} = \boldsymbol{\nu}_y(Z) + \boldsymbol{\Lambda}_y(Z)\,\eta_2 + \boldsymbol{\delta}(Z),$$

where:

- $\boldsymbol{\nu}_x(Z)$ and $\boldsymbol{\nu}_y(Z)$ are the 4×1 and 3×1 vectors of intercepts for $\mathbf{x}$ and $\mathbf{y}$, respectively,
- $\boldsymbol{\Lambda}_x(Z)$ is the 4×1 loading vector for η_1, and $\boldsymbol{\Lambda}_y(Z)$ is the 3×1 loading vector for η_2,
- $\boldsymbol{\varepsilon}(Z) \sim \mathcal{N}\big(\mathbf{0}, \boldsymbol{\Theta}_x(Z)\big)$ and $\boldsymbol{\delta}(Z) \sim \mathcal{N}\big(\mathbf{0}, \boldsymbol{\Theta}_y(Z)\big)$ are the residual vectors for $\mathbf{x}$ and $\mathbf{y}$,
- $\boldsymbol{\eta} = (\eta_1, \eta_2)^\top \sim \mathcal{N}\big(\boldsymbol{\mu}_\eta(Z), \boldsymbol{\Phi}_\eta(Z)\big)$ is the latent factor vector,
- $Z = h(M)$ is the effective moderator entering the model equations, obtained by transforming a raw continuous covariate M.

Moderator Transformation

A raw continuous covariate

$$M \sim \mathcal{N}(0, 1)$$

is transformed deterministically into an effective moderator

$$Z = h(M),$$

where $h(\cdot)$ is chosen from a finite set of transformation functions. Each transformation type is treated as a separate level in the simulation design and is fully crossed with the remaining design factors.

The following moderator transformations are implemented:

1. **Linear:**
$$h(M) = M.$$
This yields an unbounded moderator symmetric around zero.

2. **Quadratic:**
$$h(M) = M^2.$$
The resulting moderator is non-negative and right-skewed, with most probability mass concentrated near zero.

3. **Sigmoid:**
$$h(M) = 2 \cdot \text{logit}^{-1}(\delta M) - 1,$$
where δ is a fixed slope parameter. This transformation maps $\mathbb{R}$ to $(-1, 1)$ and generates a bounded moderator symmetric around zero.

Factor Loadings

For each latent factor, a single reference indicator loading is fixed to 1.0 to set the measurement scale, and all remaining baseline loadings for that factor are fixed at 0.7. For the two-factor model in Study 2, this yields

$$\mathbf{\Lambda}_x = \begin{bmatrix} 0.7 \\ 0.7 \\ 0.7 \\ 1.0 \end{bmatrix}, \qquad \mathbf{\Lambda}_y = \begin{bmatrix} 0.7 \\ 0.7 \\ 1.0 \end{bmatrix},$$

with x_4 and y_3 serving as reference indicators.

Moderation of factor loadings is introduced only for the first two x-indicators. For each moderated indicator $i \in \{1, 2\}$,

$$\lambda_{xi}(Z) = \lambda_{xi} + \delta_{\lambda_i} Z.$$

To ensure that moderated loadings remain within an admissible interval,

$$\lambda_{xi}(Z) \in [0.3,\ 1.0],$$

the moderation coefficients are restricted to

$$\delta_{\lambda_i} \in [-0.4,\, 0.3],$$

and evaluated on the discrete grid

$$\delta_{\lambda_i} \in \{-0.4,\, -0.2,\, 0,\, 0.2,\, 0.3\}.$$

All remaining loadings remain fixed across conditions:

$$\lambda_{x3}(Z) = 0.7, \quad \lambda_{x4}(Z) = 1.0, \quad \lambda_{yj}(Z) = 0.7 \quad (j = 1, 2), \qquad \lambda_{y3}(Z) = 1.0.$$

Thus, the loading matrices take the form

$$\boldsymbol{\Lambda}_x(Z) = \begin{bmatrix} \lambda_{x1}(Z) \\ \lambda_{x2}(Z) \\ 0.7 \\ 1.0 \end{bmatrix}, \qquad \boldsymbol{\Lambda}_y(Z) = \begin{bmatrix} 0.7 \\ 0.7 \\ 1.0 \end{bmatrix},$$

yielding fully crossed combinations of (i) moderator transformation type $Z = h(M)$, (ii) moderated vs. unmoderated indicators, and (iii) moderation strength δ_{λ_i}.

Residual Variances
Item-specific reliabilities are varied deterministically across simulation conditions using the fixed grid

$$\text{Reliability}_i \in \{0.60,\, 0.70,\, 0.80,\, 0.95\}.$$

Baseline residual variances are derived analytically from the reliability identity

$$\text{Reliability}_i = \frac{\lambda_{xi}^2}{\lambda_{xi}^2 + \theta_{i0}} \quad \Longrightarrow \quad \theta_{i0} = \frac{\lambda_{xi}^2(1 - \text{Reliability}_i)}{\text{Reliability}_i}.$$

Baseline factor loadings in Study 2 follow the fixed measurement structure:

$$\lambda_{x1} = \lambda_{x2} = \lambda_{x3} = 0.7, \qquad \lambda_{x4} = 1.0,$$

$$\lambda_{y1} = \lambda_{y2} = 0.7, \qquad \lambda_{y3} = 1.0.$$

These values are used to compute the corresponding baseline residual variances $\theta_{10}, \ldots, \theta_{70}$. Residual-variance moderation is then introduced multiplicatively as

$$\theta_i(Z) = \theta_{i0}\left(1 + \delta_\theta Z\right),$$

where δ_θ is the residual-variance moderation coefficient. Indicators that are not subject to residual moderation use $\delta_\theta = 0$ and therefore retain $\theta_i(Z) = \theta_{i0}$ for all Z.
Since residual-variance moderation operates solely through the multiplicative term above, variation in factor loadings does not affect residual variances. Indicators with $\delta_\theta = 0$ therefore retain their baseline values θ_{i0} for all Z.

Latent Variable Distribution

Latent means are held constant at zero for both latent factors:

$$\boldsymbol{\mu}_\eta(Z) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Latent variances are allowed to vary with the effective moderator Z through the log-linear specification

$$\sigma^2_{\eta_j}(Z) = \exp(\beta_{0j} + \beta_{1j} Z)\,, \qquad j = 1, 2,$$

ensuring positivity for all variance values.
The parameters governing baseline variance and moderation strength are selected from discrete grids:

$$\beta_{0j} \in \{-0.5,\, 0,\, 0.5\}, \qquad \beta_{1j} \in \{-0.5,\, 0,\, 0.5\}.$$

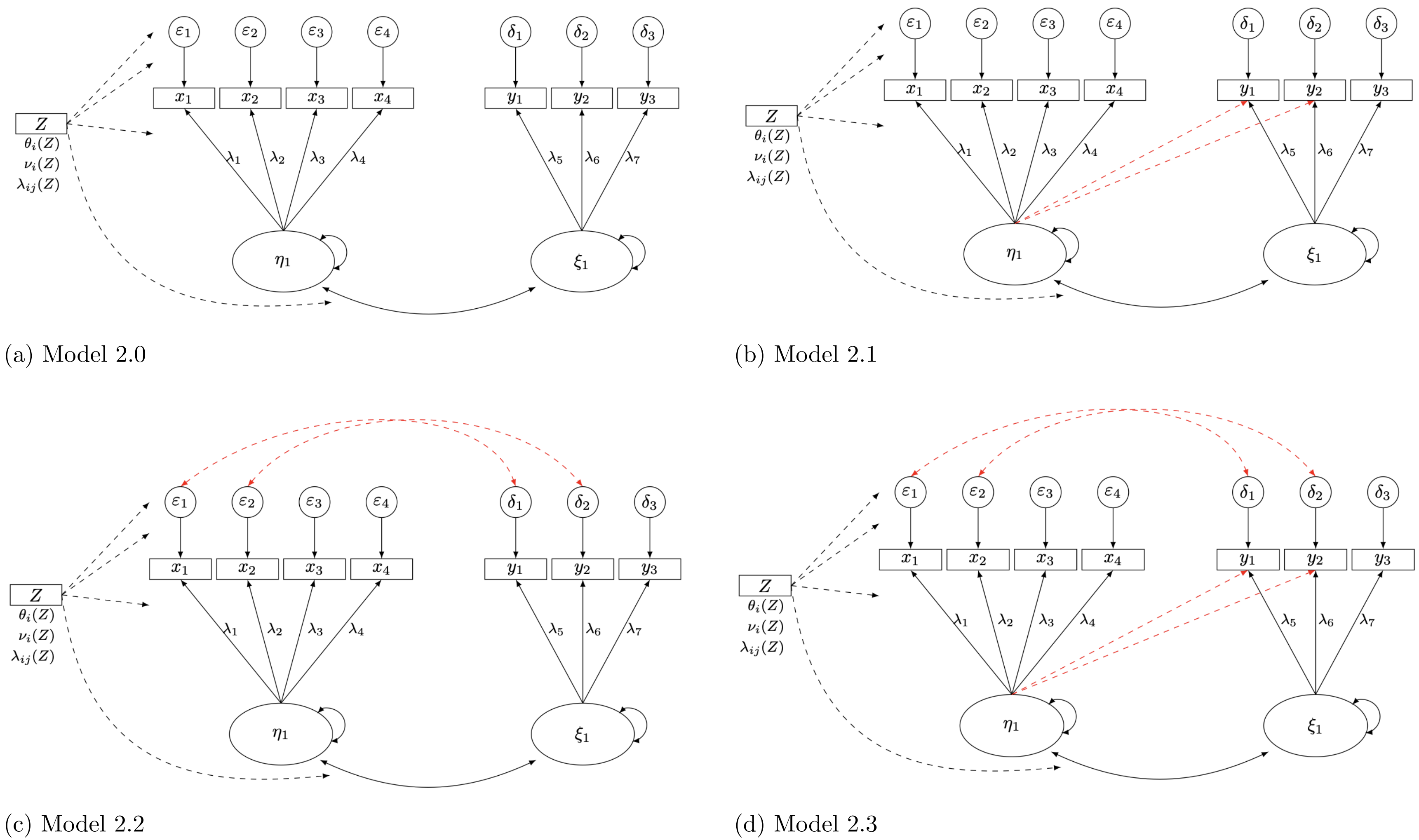
The latent covariance between η_1 and η_2 is fixed across all conditions:

$$\phi_{12}(Z) = 0.4,$$

yielding the latent covariance matrix

$$\boldsymbol{\Phi}_\eta(Z) = \begin{bmatrix} \sigma^2_{\eta_1}(Z) & 0.4 \\ 0.4 & \sigma^2_{\eta_2}(Z) \end{bmatrix}.$$

Analytical Models Study 2



(a) Model 2.0

(b) Model 2.1

(c) Model 2.2

(d) Model 2.3

Figure 1: We specify a confirmatory factor analysis model with two latent factors, η_1 and ξ_1, and seven observed indicators, $\mathbf{x} = (x_1, \ldots, x_4)^\top$ and $\mathbf{y} = (y_1, y_2, y_3)^\top$, along with their respective residuals, $\varepsilon_1, \ldots, \varepsilon_4$ and $\delta_1, \delta_2, \delta_3$. The first factor, η_1, loads onto indicators x_1 to x_4, while the second factor, ξ_1, loads onto indicators y_1 to y_3. A continuous moderator variable Z introduces conditional heterogeneity into selected measurement parameters, including residuals, intercepts, factor loadings, and latent factor covariance, using one of two functional forms (linear or quadratic). Red dashed arrows indicate model misspecifications.

3.2 What will be the different factors of the data-generating mechanism?

The simulation design includes several manipulated factors that govern the data-generating mechanism (DGM). These factors are varied across simulation conditions and apply consistently to both studies, unless otherwise noted.

Core DGM Factors (Applied in Both Studies)

The following factors are systematically varied in both studies using a grid search design:

- **Functional form of moderation:** The moderator variable Z is defined as a transformation of $X \in [-1, 1]$, evaluated systematically using three functional forms:
  - Linear: $Z = X$
  - Quadratic: $Z = X^2$
  - Sigmoid: $Z = \frac{1}{1+\exp(-\delta(X-c))}$

  For the sigmoid function, parameters are fixed to:

  $$c = 0, \quad \delta \in \{5, 2.5, 1\}$$

  representing steep, moderate, and relatively flat sigmoid curves, respectively.

- **Moderated parameters:** The parameters subject to moderation by Z are systematically specified across grid conditions. For each condition, moderation is applied to:
  - None
  - One parameter (e.g., λ_{x1} or ν_1)
  - Two parameters (e.g., $\lambda_{x1}, \lambda_{x2}$)

  This replaces random selection with explicit condition specification for clear, reproducible simulation configurations.

- **Latent variance moderation:** The latent variance is modeled as a log-linear function of Z:

  $$\sigma^2_\eta(Z) = \exp(\beta_0 + \beta_1 Z)$$

  where:

  $$\beta_0 \in \{-0.5, 0, 0.5\}, \quad \beta_1 \in \{-0.5, 0, 0.5\}$$

- **Model misspecification of moderator form:** The DGM defines Z using one of the three functional forms above, while the analytical model assumes either a *linear* or *quadratic* moderator effect.

Study-Specific Factors

Study 1. Focuses on functional form misspecification and data-related conditions:

- **Analysis model:** Assumes either a linear or quadratic moderator.
- **Sample sizes:** $N \in \{300, 500, 700, 1000\}$
- **Indicator reliabilities:** Low (.6), moderate (.7), moderate to high (.8), or high (.95).
- **No structural misspecification:** The factor structure and residual terms follow the specified model.

Study 2. Builds on Study 1 and introduces additional model misspecification in the factor structure:

- **Same DGM and moderator form variation as in Study 1.**
- **Analysis model:** Assumes either a linear or quadratic form for the moderator.
- **Structural model misspecifications:** (between-subjects factor)
  1. **No misspecification:** Factor structure matches the analysis model.
  2. **Cross-loadings:** Items y_5, y_6 additionally load on a second latent factor:
  $$\lambda_5^{(CL)}, \lambda_6^{(CL)} \in \{0.3, 0.4\}$$
  3. **Correlated residuals:** Residual covariances are added between:
  $$\mathrm{Cov}(\varepsilon_1, \varepsilon_5), \quad \mathrm{Cov}(\varepsilon_2, \varepsilon_6) \in \{0.2, 0.3\}$$
  4. **Combined:** Both cross-loadings and residual correlations are present.

3.3 If there is more than one factor: How will the factor levels be combined and how many simulation conditions will this create?

We employ a full grid search, in which parameter values are sampled from pre-specified discrete values. A total of 6000 simulation replications should ideally be conducted. This number was chosen to ensure that the confidence intervals for key estimated proportions (e.g., power or Type I error rates near $p = 0.8$) achieve an approximate width of ±1%. This is based on the calculation:

$$\mathrm{SE} = 2 \times \sqrt{\frac{p(1-p)}{6000}},$$

which yields a standard error consistent with the desired precision for evaluating simulation outcomes within this study. To ensure computational feasability, we propose an iterative approach whereby initial pilot simulations $n = 100$ are conducted to amongst others, estimate the needed simulation time. These estimates then inform a more principled determination of the total number of replications, targeting a predefined precision threshold.

The factors that vary across simulation conditions include:

- The functional form of moderator effects (linear, quadratic, sigmoid)
- The type of moderated parameters (e.g., factor loadings, intercepts, residual variances)
- The degree of model misspecification (none, cross-loadings, correlated residuals, combined)
- Factor loadings (ranging from 0.7 to 0.9)
- Item reliabilities (ranging from 0.6 to 0.95)
- Moderator strength parameters
- Sample sizes per group (i.e. 300, 500, 700, 1000)

4 Estimands and Targets

4.1 What will be the estimands and/or targets of the simulation study?

4.2 What will be the estimands and/or targets of the simulation study?

The primary estimands in this simulation study pertain to the detection and accurate estimation of measurement non-invariance (MI) due to moderation effects on measurement parameters. Specifically, we assess whether the estimation methods compared can correctly identify the presence or absence of moderation in factor loadings, intercepts, and residual variances, reflecting the true data-generating structure. Consequently, key simulation outcomes include Type I error rates and statistical power associated with detecting such moderation effects, as well as bias and variability in the recovered parameter estimates.

Secondary estimands include model-level indicators of fit (e.g., RMSEA with confidence intervals, SRMR, CFI), which serve as auxiliary diagnostics for methods that yield such metrics (i.e., MNLFA). These are not applicable to tree-based approaches and are thus interpreted as method-specific targets rather than general evaluative criteria.

5 Methods

5.1 How many and which methods will be included and which quantities will be extracted?

We will compare the following methods:

1) **MNLFA** (Moderated Nonlinear Factor Analysis): MNLFA was initially introduced by **Bauer2009<empty citation>** within the framework of integrative data analysis and was subsequently extended into a general approach for assessing measurement invariance and moderation (**Bauer2017**). The fundamental principle of MNLFA is the parameterization of both measurement and structural model parameters as functions of one or more moderator variables. MNLFA accommodates both frequentist estimation methods, such as maximum likelihood (ML), and Bayesian techniques, including Markov Chain Monte Carlo (MCMC), thereby offering methodological flexibility aligned with specific analytical objectives (**Muench2024**).

   ```
   %code example?
   ```

2) **SEM-Trees**: Structural equation modeling (SEM) trees were originally introduced by **Brandmaier2013<empty citation>** as a way to systematically search for important covariates and their interactions in data, creating homogeneous subgroups. The method was refined later on by **Brandmaier2016**; **Arnold2021<empty citation>**. SEM trees and forests are data-driven, non-parametric methods that build upon the decision tree paradigm, thus using recursive partitioning to identify latent subgroups in which model parameters differ, without assuming specific functional forms or predefined covariate effects. These approaches allow for nonlinear moderation of factor loadings and can reveal complex interaction effects, enabling the exploratory detection of DIF.

   ```
   %code example?
   ```

For the parametric MNLFA, we will extract estimated moderated parameters (e.g., factor loadings, intercepts, residual variances), their associated standard errors, and two-sided p-values for testing the null hypothesis of no moderation effect. Additionally, model fit indices such as RMSEA (with confidence intervals), SRMR, and CFI will be recorded.
For the parameter estimates and fit indices, we will summarize their distributions across simulation replications by reporting key quantiles (2.5th, 50th, and 97.5th percentiles). The null hypothesis will be rejected if the associated p-value is less than the conventional significance level of 0.05.
In contrast, non-parametric Tree-based approaches do not yield direct parameter estimates with standard errors. There detection of MI/ non-invariance is operationalized via the presence of at least one split. This distinction ensures that each method is evaluated according to its own inferential framework, while maintaining comparability in terms of its ability to detect MI reliably.

6 Performance Measures

6.1 Which performance measures will be used?

For the MNLFA **exclusively**, we will evaluate the following performance measures:

- **Bias**: The average difference between the estimated parameter $\hat{\theta}$ and the true parameter θ, defined as
$$\widehat{\text{Bias}} = \frac{1}{n_{\text{sim}}} \sum_{i=1}^{n_{\text{sim}}} \hat{\theta}_i - \theta,$$
where n_{sim} is the number of simulation replications.

- **Absolute Bias**:
$$\widehat{\text{AbsBias}} = \frac{1}{n_{\text{sim}}} \sum_{i=1}^{n_{\text{sim}}} \left| \hat{\theta}_i - \theta \right|.$$

- **Relative Bias** (expressed as a proportion of the true parameter):
$$\widehat{\text{RelBias}} = \frac{\widehat{\text{Bias}}}{|\theta|} \quad \text{for } \theta \neq 0.$$

- **Root Mean Squared Error (RMSE)**:
$$\widehat{\text{RMSE}} = \sqrt{\frac{1}{n_{\text{sim}}} \sum_{i=1}^{n_{\text{sim}}} (\hat{\theta}_i - \theta)^2}.$$

- **Coverage Probability**: The proportion of confidence intervals that contain the true parameter θ.

- **Model Fit Indices** (e.g., RMSEA) will be reported if applicable, computed according to their standard definitions.

For the SEM trees **as well as** for the MNLFA we will evaluate:

- **Type I Error Rate and Power**: The rejection rate of the null hypothesis at significance level $\alpha = 0.05$, calculated as
$$\widehat{\text{RRate}} = \frac{1}{n_{\text{sim}}} \sum_{i=1}^{n_{\text{sim}}} \mathbf{1}(p_i \leq 0.05),$$
where $\mathbf{1}(\cdot)$ is the indicator function.

Performance measures calculated for multiple parameters (e.g., factor loadings) will be aggregated by reporting the mean value across parameters.

6.2 How will Monte Carlo uncertainty of the estimated performance measures be calculated and reported?

We will quantify Monte Carlo uncertainty using Monte Carlo Standard Errors (MCSEs) for each performance measure, calculated as follows:

- The **Rejection Rate** refers to the proportion of simulated datasets in which a null hypothesis is rejected, commonly used to estimate empirical Type I error rates or statistical power.
  The **MCSE for the Rejection Rate** is calculated assuming a binomial distribution of rejections across n_{sim} simulation replications, which reflects the standard error of a proportion estimator:

$$\text{MCSE}_{\widehat{\text{RRate}}} = \sqrt{\frac{\widehat{\text{RRate}}(1-\widehat{\text{RRate}})}{n_{\text{sim}}}},$$

- For **Bias**, MCSE is calculated based on the sample variance of the specific parameter estimates across simulation replications:

$$\text{MCSE}_{\widehat{\text{Bias}}} = \frac{S_{\hat{\theta}}}{\sqrt{n_{\text{sim}}}},$$

- where $S_{\hat{\theta}} = \sqrt{\sum_{i=1}^{n_{\text{sim}}} \{\hat{\theta}_i - (\sum_{i=1}^{n_{\text{sim}}} \hat{\theta}_i/n_{\text{sim}})\}^2/(n_{\text{sim}}-1)}$ is the sample standard deviation of the effect estimates.

6.3 How many simulation repetitions will be used for each condition?

A total of 6000 simulation replications should ideally be conducted. This number was chosen to ensure that the confidence intervals for key estimated proportions (e.g., power or Type I error rates near $p = 0.8$) achieve an approximate width of ±1%. This is based on the calculation:

$$\text{SE} = 2 \times \sqrt{\frac{p(1-p)}{6000}},$$

which yields a standard error consistent with the desired precision for evaluating simulation outcomes within this study. To ensure computational feasability, we propose an iterative approach whereby initial pilot simulations $n = 100$ are conducted to amongst others, estimate the needed simulation time. These estimates then inform a more principled determination of the total number of replications, targeting a predefined precision threshold.

6.4 How will missing values due to non-convergence or other reasons be handled?

Non-convergence or other missing values may occur during model estimation. We plan to:

- Exclude only the specific simulation replicates that fail to converge for a given method while retaining data from other methods within the same replication.
- Report the proportion of non-converged cases per method and condition.
- If the proportion of non-convergence exceeds a pre-specified threshold (e.g., 5

No imputation of missing estimates will be performed.

6.5 How do you plan on interpreting the performance measures? (optional)

Explanation: It can be specified what a ‘relevant difference’ in performance, or what ‘acceptable’ and ‘unacceptable’ levels of performance might be to avoid post-hoc interpretation of performance. Furthermore, some researchers use regression models to analyze the results of simulations and compute effect sizes for different factors, or to assess the strength of evidence for the influence of a certain factor (**Skrondal2000**; **Chipman2022**). If such an approach will be used, please provide as many details as possible on the planned analyses.

Example

We define a type I error rate larger than 5% as non-acceptable performance. Amongst methods that exhibit acceptable performance regarding the type I error rate (within the MCSE), we consider a method X as performing better than a method Y in a certain simulation condition if the lower bound for the estimated power of method X ($\widehat{\text{Pow}} - \text{MCSE}$) is greater than the upper bound for the estimated power of method Y ($\widehat{\text{Pow}} + \text{MCSE}$).

Answer:

7 Other

7.1 Which statistical software/packages do you plan to use?

Explanation: Likely, not all software used can be prespecified before conducting the simulation. However, the main packages used for model fitting are usually known in advance and can be listed here, ideally with version numbers.

Example

We will use the following packages of `R` version 4.3.1 (**R2020**) in their most recent versions: The `mvtnorm` package (**Genz2009**) to generate data, the `lm()` function included in the `stats` package (**R2020**) to fit the different models, the `SimDesign` package (**Chalmers2020**) to set up and run the simulation study, and the `ggplot2` package (**Wickham2016**) to create visualizations.

Answer:

7.2 Which computational environment do you plan to use?

Explanation: Please specify the operating system and its version which you intend to use. If the study is performed on multiple machines or servers, provide information for each one of them, if possible.

Example

We will run the simulation study on a Windows 11 machine. The complete output of `sessionInfo()` will be saved and reported in the supplementary materials.

Answer:

7.3 Which other steps will you undertake to make simulation results reproducible? (optional)

Explanation: This can include sharing the code and full or intermediate results of the simulation in an open online repository. Additionally, this may include supplemental materials or interactive data visualizations, such as a shiny application.

Example

We will upload the fully reproducible simulation script and a data set containing all relevant estimates, standard errors, and p-values for each repetition of the simulation to OSF (https://osf.io/dfgvu/) and GitHub (https://github.com/bsiepe/SimPsychReview).

Answer:

7.4 Is there anything else you want to preregister? (optional)

Explanation: For example, the answer could include the most likely obstacles in the simulation design, and the plans to overcome them, or measures that increase the trust in the preregistration date (e.g., setting the seed based on a future event), as explained in the introduction of this template.

Example

No.

Answer:

References