STRUCTURAL EQUATION MODELING
2025, VOL. 32, NO. 4
ISSN: 1070-5511 | E-ISSN: 1532-8007

TEACHER'S CORNER

OPEN ACCESS

modsem: An R Package for Estimating Latent Interactions and Quadratic Effects

Kjell S. Slupphaug[a], Mehmet Mehmetoglu[a] and Matthias Mittner[a,b]

[a]NTNU – Norwegian University of Science and Technology; [b]UiT – The Arctic University of Norway

ABSTRACT

Estimating interaction and quadratic effects in structural equation models (SEMs) is complex. Latent product term (LPT) models, designed for interaction effects, also accommodate quadratic effects. After 30 years of research, no consensus has emerged on the best modeling methods in SEMs, partly due to efforts to simplify approaches for accessibility. LPT models include product indicator (PI) and distribution analytic (DA) approaches. While PI methods are simpler, they sacrifice accuracy and can be error prone. DA methods, like latent moderated structural equations (LMS) and Quasi maximum likelihood (QML), are more accurate but computationally intensive. We introduce modsem, an R package that simplifies these estimations and enhances accessibility.

KEYWORDS

Interaction effects; latent interaction; LMS; moderation; R; SEM

1. Introduction

The estimation of interaction and quadratic effects in Structural Equation Models (SEMs) has been an active research topic for many years. One approach for estimating these types of models in an SEM framework is using latent product term models (LPT). LPT models were originally developed for estimating interaction effects, i.e., product terms between latent variables (Kenny & Judd, 1984), but are also capable of estimating quadratic effects (Umbach et al., 2017), since a quadratic effect implies a product term of a variable with itself.

Although it has been over 30 years since the first LPT models for SEMs were presented, there is still a disagreement in the literature about the best way to estimate such models (Klein & Moosbrugger, 2000; Marsh et al., 2004, 2013; Schumacker & Marcoulides, 1998). While the scarcity of research on the topic is one main reason for this continuing debate, there has also been a parallel development in terms of the priorities that the researchers have emphasized. On one side, emphasizing the pragmatic implementation of such models, methodological researchers have focused on simplifying existing approaches to make them more accessible to applied researchers. On the other hand, more theoretically oriented researchers have focused on developing more theoretically sound (but also generally more computationally expensive and less accessible) approaches.

As a consequence, interaction effects are less frequently estimated when compared to linear models, even though they are often theorized to exist (Steinmetz et al., 2011). In summary, there is a lack of freely available statistical tools for estimating interaction effects in SEMs, which makes the approaches less accessible to applied researchers. The absence of such tools is also a significant barrier for studies comparing different approaches to estimating interaction effects in different contexts.

To bridge the existing gap between theoretically hypothesized and statistical models, there is a need for free user-friendly statistical tools implementing the most recent and theoretically sound approaches for estimating interaction effects in SEMs, that can be applied even without a deep understanding of the underlying statistical theory.

1.1. Different Approaches to Estimating LTP Models in SEMs

Over the years, a wide range of different approaches to estimating interaction effects in SEMs have been recommended in the literature. These can be broadly divided into two categories:

1. Product Indicator (PI) Approaches
2. Distribution Analytic (DA) Approaches.

In general, the PI approaches were intended to be performed manually by the user and were estimated using traditional estimation approaches based on the sample covariance matrix. These methods rely on implementing a set of constraints on products of indicators in the larger SEM. Early implementations of that idea were particularly complicated and error prone to specify (Jöreskog & Yan, 1996; Kenny & Judd, 1984; Schumacker & Marcoulides, 1998). This led to a continuing trend of simplifying the approach to achieve models that are more easily implemented without sacrificing model accuracy (Algina & Moulder, 2001; Lin et al., 2010; Little et al., 2006; Martin & Marsh, 1999).

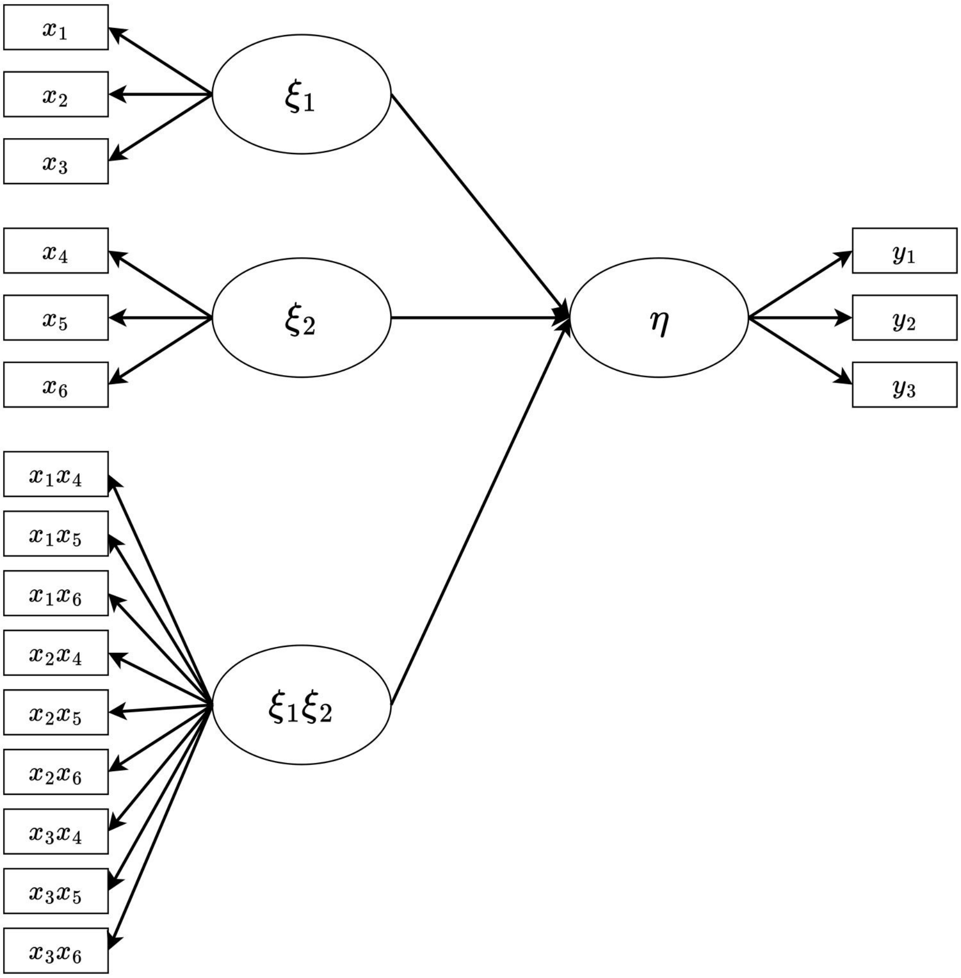

Figure 1. Elementary interaction model.

On the other hand, the parallel development of DA approaches, such as the *Latent Moderated Structural Equations* (LMS) (Klein & Moosbrugger, 2000) approach and the *Quasi Maximum Likelihood* (QML) (Klein & Muthén, 2007) approach, applied a different logic based on estimating the interaction effects from the distributional characteristics of the indicators for the endogenous variables.

1.2. Product Indicator (PI) Approaches

1.2.1. The Constrained Approach

The first latent variable approach to estimating interaction effects between latent variables was presented by Kenny and Judd (1984) and Marsh et al. (2013). In their paper, they proposed a solution for estimating the interaction effect of latent variables ξ_1 and ξ_2 onto an observed variable y. Such that

$$y = \gamma_1\xi_1 + \gamma_2\xi_2 + \gamma_3\xi_1\xi_2 + \zeta \quad (1)$$

Jaccard and Wan (1995) later replaced the observed variable y with a latent variable η, making it a fully latent variable approach. This model is now widely referred to as the *elementary interaction model* (EIM). The EIM is illustrated in Figure 1[1].

In this model, there are two exogenous latent variables ξ_1 and ξ_2, including their interaction term $\xi_1\xi_2$, and a single endogenous latent variable η, with a disturbance variable (i.e., residual) ζ such that:

$$\eta = \gamma_1\xi_1 + \gamma_2\xi_2 + \gamma_3\xi_1\xi_2 + \zeta. \quad (2)$$

In the approach by Kenny and Judd (1984) and Jaccard and Wan (1995), the *product indicators* for $\xi_1\xi_2$ were computed as all unique products between the indicators for ξ_1 and ξ_2. To make the model mathematically correct, Kenny and Judd (1984) placed a large number of constraints (many of which are nonlinear) in both the measurement and structural model. As a consequence, implementing and estimating the model was computationally demanding, tedious, and in general quite error prone.

Notably, the approach failed to take into account the mean structure of the measurement and structural model. For example, it is assumed that all of the latent exogenous variables have mean zero. This then implies that the covariance between ξ_1 and ξ_2 is assumed to be zero since $\mathbb{E}[\xi_1\xi_2] = \sigma(\xi_1, \xi_2)$ if $\mathbb{E}[\xi_1] = \mathbb{E}[\xi_2] = 0$. Thus the model would only be correctly specified if $\sigma(\xi_1, \xi_2) = 0$. If not, $\mathbb{E}[\xi_1\xi_2]$ is non-zero – violating the assumption that all latent exogenous variables variables have zero mean.

Jöreskog and Yan (1996) proposed a revised model where the mean structure for both the observed and latent variables

[1] The model originally only had two indicators per latent variable.

was taken into account. Thus, the structural model from Equation (2) was revised to include an intercept for η such that:

$$\eta = \alpha + \gamma_1\xi_1 + \gamma_2\xi_2 + \gamma_3\xi_1\xi_2 + \zeta. \tag{3}$$

This approach led to a substantial improvement (Marsh et al., 2013) over the original model by Kenny and Judd (1984). However, the model would often not successfully converge on a solution.

Shortly after, Algina and Moulder (2001) developed a simplified version of the model in which they centered the indicators before computing the indicator products. Setting the means of indicators to zero before computing the product indicators allowed them to remove the terms in the constraints that included the means of the indicators. This was a substantial improvement to the model because it reduced the number of constraints and made the model both more robustly estimated and easier to implement.

This change to Equation (3) corresponds to a general linear model where ξ_1 and ξ_2 are centered before computing the product term, such that:

$$\begin{aligned}\eta = {} & (\alpha - \gamma_1\mathbb{E}[\xi_1] - \gamma_2\mathbb{E}[\xi_2] + \gamma_3\mathbb{E}[\xi_1]\mathbb{E}[\xi_2]) \\ & + (\gamma_1 - \gamma_3\mathbb{E}[\xi_1])\xi_1 + (\gamma_2 - \gamma_3\mathbb{E}[\xi_2])\xi_2 + \gamma_3\xi_1\xi_2 + \zeta\end{aligned} \tag{4}$$

Rearranging the equation in this way allows us to see how the parameter estimates are affected by the centering of the indicators [where $\hat{\alpha}$ is the estimate for α in the model by Algina and Moulder (2001), etc.]:

$$\begin{aligned}\hat{\alpha} & \rightarrow (\alpha - \gamma_1\mathbb{E}[\xi_1] - \gamma_2\mathbb{E}[\xi_2] + \gamma_3\mathbb{E}[\xi_1]\mathbb{E}[\xi_2]) \\ \hat{\gamma}_1 & \rightarrow (\gamma_1 - \gamma_3\mathbb{E}[\xi_1]) \\ \hat{\gamma}_2 & \rightarrow (\gamma_2 - \gamma_3\mathbb{E}[\xi_2]) \\ \hat{\gamma}_3 & \rightarrow \gamma_3\end{aligned} \tag{5}$$

Thus, the coefficient for the interaction term, γ_3, remains unchanged by centering, while only the intercept and the simple main effects are affected. In this sense, both Jöreskog and Yang's, and Algina and Moulder's model should yield the same estimates for the interaction effect, but may provide different estimates for the simple main effects, depending on the mean structure of ξ_1 and ξ_2

While both the Jöreskog and Yang model and Algina and Moulder's simplified version are known as the constrained approach, it is generally agreed that Algina and Moulder's model is the better version of the two (Marsh et al., 2004). This is partly due to the fact that the Algina and Moulder model has less complicated constraints, but also because centering the indicators reduces multicollinearity between the product indicators and the other indicators in the model. This is because the expected covariance between a product of two normally distributed variables (independent of their correlation) and the variables in the product is expected to be zero, i.e.,

$$\begin{aligned}& \sigma(\xi_1, (\xi_1 - \mathbb{E}[\xi_1])(\xi_2 - \mathbb{E}[\xi_2])) \\ & \quad = \sigma(\xi_2, (\xi_1 - \mathbb{E}[\xi_1])(\xi_2 - \mathbb{E}[\xi_2])) = 0.\end{aligned} \tag{6}$$

1.2.2. The Unconstrained Approach

One of the main drawbacks of the constrained approach is the difficulty in implementing a complex set of constraints, which is often prohibitive for researchers who want to examine interaction effects. Martin and Marsh (1999) took a drastic approach to that problem and removed almost all of the constraints from the constrained approach. Similarly to Algina and Moulder (2001), Martin and Marsh (1999) centered their indicators before computing product indicators. They then removed or changed almost all of the constraints in the model proposed by Jöreskog and Yan (1996), except for constraining the mean of the latent product term to be equal to the covariance among the latent variables in the interaction, such that:

$$\mathbb{E}[\xi_1\xi_2] = \phi_{21} \tag{7}$$

where ϕ_{21} is the covariance between ξ_1 and ξ_2.

In addition, they changed the constraints on the residual covariances such that the residual covariances between the product indicators with a common indicator are freely estimated. Surprisingly, this model performed quite well despite removing almost all of the constraints from the constrained approach. It performed comparably to the constrained approach in cases where normality assumptions were met and even outperformed the constrained approach in cases where the normality assumptions were violated (Marsh et al., 2004).

1.2.3. The Residual Centering Approach

The residual centering approach (RCA) was introduced by Little et al. (2006). Residual centering is a statistical technique used to remove shared variance from a set of variables while allowing each variable to retain its unique variance (Lance, 1988). Assuming that the indicators are normally distributed (and centered), the shared variance between a product indicator and the indicators from which it is computed should be zero (Lin et al., 2010).

However, in cases where the indicators are not normally distributed, the shared variance between the product indicator and the indicators from which it is computed is not necessarily zero, even if the indicators are centered beforehand. Given that the standard approach to creating product indicators is to compute the product between all unique pairs of indicators, a large number of product indicators ($N_1 \times N_2$) is used that can potentially lead to a large amount of shared variance between the product indicators and the indicators from which they are computed (Little et al., 2006). The residual centering method would therefore be particularly effective in removing multicollinearity caused by the product indicators, in cases where the indicators from which they were computed are non-normally distributed (Geldhof et al., 2013).

The method works by computing indicator products with uncentered indicators from the latent variables, which are regressed on all of the indicators in the latent variables in the product term, thereby removing the shared variance between the indicators in the product term and its corresponding indicators. Since the residuals are centered at zero, it also allows for the removal of the mean structure from the model (i.e., removing $\mathbb{E}[\xi_1\xi_2] = \phi_{21}$).

While the RCA does not seem to improve performance in relatively simple cases (Lin et al., 2010), Geldhof et al. (2013) presented a set of cases where the approach should be seen as the favored alternative to more traditional approaches. These cases include higher-order interactions (e.g., interactions between three latent variables) and estimating interaction effects where there is a high covariance between the product indicators and the indicators from which they were computed, where multicollinearity might cause convergence issues in more traditional approaches.

That said, some researchers have raised concerns about the method. For example, Lin et al. (2010) showed that in cases where the indicators are skewed, there can be a mismatch between residual centering on the indicator and the latent level, introducing a non-random bias.

1.2.4. The Double Centering Approach

The double mean centering approach (DCA) was presented by Lin et al. (2010), and was a further simplification of the Unconstrained Approach (UCA), taking inspiration from the removal of the mean structure in the RCA (Lin et al., 2010). In this approach, the indicators are centered before and after computing the product indicators. The idea behind this is to remove the mean structure from the model completely and to eliminate potential collinearities when the indicators are non-normal (Lin et al., 2010).

As discussed above, Lin et al. (2010) showed that the RCA led to a non-random bias in the estimates of the simple main effects in the model, a situation which the DCA avoids. Therefore, while the two approaches both yield the same estimate for the interaction effect, they do not necessarily yield the same estimates for the simple main effects.

It can be shown (through some tedious calculations) that the transformations of the simple main effects (and intercept) are equivalent in both the DCA and the RCA (Lin et al., 2010), when the indicators used in the product term are normally distributed. Thus, the approaches are equivalent in cases where the indicators are normally distributed. In cases where the indicators are skewed, Lin et al. (2010) demonstrated that the DCA does not have the same inconsistency regarding the main effects as the RCA, and should therefore be seen as the preferred alternative to the RCA.

1.3. The Distribution Analytic (DA) Approaches

In traditional maximum likelihood algorithms for SEM estimation that are based on the sample covariance matrix, it is usually assumed that the underlying latent variables are normally distributed. Klein and Moosbrugger (2000) however, pointed out that this assumption is violated in interaction models. Even in the case where ξ_1 and ξ_2 are normally distributed, their product $\xi_1\xi_2$ is not—implying that η is non-normal as well since it includes the product term as a linear term. This leads to product indicator approaches underestimating standard errors (Schumacker & Marcoulides, 1998), an effect that is particularly pronounced for small sample sizes.

Addressing this issue, Klein and Moosbrugger (2000) developed the Latent Moderated Structural Equations (LMS) approach, which explicitly takes into account the non-normal distribution of the product term and $\boldsymbol{\eta}$. Klein and Moosbrugger (2000) only presented the equations for the cases with a single endogenous variable, where the model was given in matrix notation by:

$$\boldsymbol{\eta} = \boldsymbol{\alpha} + \Gamma\boldsymbol{\xi} + \boldsymbol{\xi}'\Omega\boldsymbol{\xi} + \boldsymbol{\zeta} \tag{8}$$

where $\boldsymbol{\xi}$ is a random vector of latent exogenous variables, and Ω is a matrix of coefficients for the interaction terms. The LMS approach models the joint distribution of the indicator vectors $(\boldsymbol{x}, \boldsymbol{y})$ as a finite mixture of normal densities. This is achieved by decomposing $\boldsymbol{\xi}$ into two subvectors: $\boldsymbol{\xi} = \boldsymbol{A}\boldsymbol{z}, \boldsymbol{z} = (\boldsymbol{z}_1, \boldsymbol{z}_2)$, where $\boldsymbol{z}_1$ represents latent exogenous variables with a non-linear effect on $\boldsymbol{\eta}$, $\boldsymbol{z_2}$ represents variables with a linear effect, and $\boldsymbol{A}$ is the Cholesky decomposition of the covariance matrix (Φ) between the latent exogenous variables. The Cholesky decomposition of $\boldsymbol{A}$ allows us to model $\boldsymbol{z}$ as a combination of p independent random variables (where p is the number of latent exogenous variables), which are transformed into correlated random variances by $\boldsymbol{A}$.

The method proceeds by substituting a finite set of values for $\boldsymbol{z}_1$ drawn from a discrete approximation of the normal distribution, using Gaussian quadrature. At each point of $\boldsymbol{z}_1$, the log likelihood of the model is computed by evaluating the likelihood of each observation of $(\boldsymbol{x}, \boldsymbol{y})$ given the implied mean vector and covariance matrix, for the values of $\boldsymbol{z}_1$. The probability of observing each value of $\boldsymbol{z}_1$ is then used as a mixing weight for generating the likelihood of each observation, across all values of $\boldsymbol{z}_1$. The estimation procedure is performed using the EM algorithm, where the expectation step is executed by computing the likelihood of each observation, given the current values of the estimated parameters. These probabilities are then used to weight the importance of each observation in the maximization step, where the parameters are optimized to maximize the likelihood of the model. These steps are repeated until the log likelihood of the model converges onto a predefined criterion (Klein & Moosbrugger, 2000). The LMS approach was shown in the same paper to yield unbiased parameter estimates and standard errors.

Subsequently, Klein and Muthén (2007) presented the Quasi Maximum Likelihood (QML) approach, which took a slightly different approach, transforming the joint indicator vector such that only the first component of the vector (where $\lambda_1 = 1$) was non-normal. The QML approach then estimates the likelihood of the non-normal indicator by using an approximation of the non-normal distribution of y_1, while the rest of the (transformed) indicators are modeled as normally distributed variables. Klein and Muthén (2007) showed that the QML approach yielded virtually the same estimates as the LMS approach. When all the normality assumptions of the LMS approach were met, it performed slightly worse, and it performed slightly better when those assumptions were violated. The main advantage of the QML approach, however, is that it severely reduces the computational complexity of the algorithm. The computational cost is particularly problematic in the LMS approach when there are a large number of exogenous variables with non-linear effects, as the number of nodes in the mixture model increases exponentially with the number of exogenous variables (Klein & Moosbrugger, 2000; Klein & Muthén, 2007).

While the LMS and QML approaches have shown promising results (Klein & Moosbrugger, 2000; Klein & Muthén, 2007), they have not been widely adopted by researchers. This is likely due to the fact that the LMS approach has only been implemented in `Mplus`, and the QML approach was available only in a custom program (Klein & Muthén, 2007), which does not seem to be available any longer. One of the main reasons for the absence of user-friendly implementations of these procedures is likely due to the fact that the set of equations for both approaches was never actually published in full. Rather, only a simplified version focusing on a single endogenous variable was published. For example, the `nlsem` package in `R` (Umbach et al., 2017) only allows for the estimation of interaction effects in models with a single endogenous variable due to the absence of reliable documentation of the full set of equations (personal communication with the author of `nlsem`).

In `modsem`, we have implemented extensions of both the LMS and QML approaches, which allow for the estimation of models with multiple endogenous variables and interaction effects with endogenous and exogenous latent variables. The equations for the extended LMS approach are based on the equations given in Jin et al. (2020) and (Wall & Amemiya, 2007). In their paper, Jin et al. (2020) presented an extended version of Equation (8), where they split the equation into a system of recursive equations. We did not adapt all of the equations from Jin et al. (2020) into `modsem`, choosing a subset that is more easily adapted into the LMS and QML approaches. In particular, we adapted the equations that allow the estimation of models with multiple endogenous variables, and interaction effects between endogenous and exogenous variables. To allow for interaction effects between endogenous variables, we used a simpler approach based on general recommendations in Wall and Amemiya (2007), which applies a simplified version of the equations for linear SEM, based on Mulaik (2010).

Concretely, let k denote the number of endogenous variables, and n denote the number of exogenous variables in the model, such that:

$$\boldsymbol{\eta} = \boldsymbol{\alpha} + \boldsymbol{B}\boldsymbol{\eta} + \Gamma\xi + (\boldsymbol{I_k} \otimes \xi)'\Omega\xi + (\boldsymbol{I_k} \otimes \xi)'\Xi\boldsymbol{\eta} + \zeta \tag{9}$$

where $\boldsymbol{B}$ is a $k \times k$ matrix of coefficients relating endogenous variables to each other, and Ω and Ξ are partitioned matrices containing submatrices with interaction coefficients for each random variable in $\boldsymbol{\eta}$ (Jin et al., 2020). In particular, Ω is a $kn \times n$ matrix of coefficients for interaction effects between exogenous variables, such that:

$$\Omega = \begin{pmatrix} \Omega_1 \\ \Omega_2 \\ \vdots \\ \Omega_k \end{pmatrix} \tag{10}$$

and Ξ is a $kn \times k$ matrix of coefficients for interaction effects between exogenous and endogenous variables, such that:

$$\Xi = \begin{pmatrix} \Xi_1 \\ \Xi_2 \\ \vdots \\ \Xi_k \end{pmatrix} \tag{11}$$

where $\boldsymbol{I}_k$ is the identity matrix of size k and $A \otimes B$ denotes the Kronecker product between A and B.

Let

$$\boldsymbol{C^{-1}} = \left(\boldsymbol{I_k} - \Gamma - \boldsymbol{B} - (\boldsymbol{I_k} \otimes \xi)'\Xi\right)^{-1} \tag{12}$$

such that $\boldsymbol{\eta}$ in Equation (9) can be expressed in reduced form as:

$$\boldsymbol{\eta} = \boldsymbol{C^{-1}}\left(\boldsymbol{\alpha} + \Gamma\xi + (\boldsymbol{I_k} \otimes \xi)'\Omega\xi + \zeta\right). \tag{13}$$

This equation allows for interaction effects between both exogenous variables and between exogenous and endogenous variables. It does, however, not allow for interaction effects between endogenous variables—since the equation could not easily be written in reduced form (Wall & Amemiya, 2007). One alternative is to split the model into a system of recursive equations (i.e., one equation can be substituted into the next) (Wall & Amemiya, 2007). Similarly to Jin et al. (2020), we choose to split the model into two equations. However, while Jin et al. (2020) split the model into a system of recursive equations where both equations contain interaction terms, we opted to split the model into two equations (i.e., two submodels), where one equation contains the interaction terms and the other does not (since this is more easily adapted into the LMS and QML approaches). To achieve this, we added a second equation for ξ in Equation (13), such that:

$$\boldsymbol{\psi} = \boldsymbol{B}\boldsymbol{\psi} + \Gamma\phi + \zeta_{\psi} \tag{14}$$

Here $\boldsymbol{\psi}$ represents the endogenous variables of the submodel, while ϕ represents the exogenous variables. Where ξ in Equation (13) is replaced with $(\boldsymbol{\psi}, \phi)$ (Jin et al., 2020; Mulaik, 2010; Wall & Amemiya, 2007). The covariance matrix of the exogenous variables (Φ) used in the QML approach, along with the Cholesky decomposition of Φ ($\boldsymbol{A}$), can be replaced with the model-implied Φ and $\boldsymbol{A}$ from Equation (14). This replacement allows for the estimation of interaction effects between endogenous variables as well, provided they can be expressed in a linear model.

In `modsem`, this is achieved through an optional argument (`cov.syntax`) in the `modsem` function, which permits the specification of an additional structural model defining the relationship between the exogenous variables in the model. Thus, if the interaction effect between two endogenous variables is to be estimated, where at least one of the endogenous variables can be expressed as a linear function of the exogenous variables, the model can be split into two submodels. In the first submodel, the interaction terms are included, and one of the endogenous variables is treated as an exogenous variable. In the second submodel, the functional relationship between the exogenous variables and the endogenous variables is then specified. An example can be found in the publicly available vignette.[2]

These extended equations for the LMS and QML approaches are currently an experimental feature in modsem. Initial tests have shown promising results when tested on individual models, but further work evaluating this method is needed.

1.4. Commercial Software for Estimating Interaction Effects in SEMs

Since the introduction of interaction effects in SEMs, it has become possible to estimate them using various statistical software packages. For example, the Kenny and Judd model was estimated using COSAN, while the Jöreskog and Yang, and the Aligna and Moulder models were estimated using LISREL. See Schumacker (2002) for a guide on using different tools (LISREL, SIMPLIS, and PRELIS2) from the LISREL software package to estimate interaction effects in SEMs.

A key feature for estimating interaction effects using product indicator (PI) approaches (in particular, different versions of the CA) is the ability to specify and estimate models with non-linear constraints. Estimating models with non-linear constraints is currently possible in various software packages (e.g., LISREL, lavaan, Mplus, and others). Most of these packages, however, require users to manually specify how the interaction effects are to be estimated, such as defining product indicators and specifying constraints.

One notable exception is Mplus, which implemented a version of the Latent Moderated Structural Equations (LMS) approach, starting with version 3.0 (Muthén & Muthén, 2004). The LMS approach does not require users to specify product indicators or model constraints, making it easier to specify interaction effects, especially in complex models. Moreover, Mplus also allows the estimation of SEMs with non-linear constraints, meaning it is possible to estimate models using the CA. However, researchers still need to manually define product indicators and constraints in those cases.

Although B. Muthén presented the QML approach alongside A. Klein (Klein & Muthén, 2007), it does not seem that the QML approach has been implemented in Mplus, which still uses the LMS approach (Muthén & Muthén, 2017).

While Mplus is highly flexible and user-friendly, it is commercial software, which limits its accessibility to some researchers. Furthermore, because the source code for Mplus is not publicly available, it cannot modified, or extended by other researchers. These limitations affect the transparency of the software and, consequently, its overall utility for the research community.

1.5. Packages in R

Currently, there are only two packages in R that assist with investigating interaction effects between latent variables in SEMs: semTools (Contributors, 2016) and nlsem (Umbach et al., 2017).

1.5.1 semTools

The semTools package implements a specialized function that allows users to create product indicators, which can then be used in lavaan. semTools can compute product indicators suitable for all of the mentioned PI approaches (see Section 1.2). While helpful, the usefulness of this function is somewhat limited as it does not have the ability to read the lavaan syntax for a model, automatically create appropriate product indicators, or generate the necessary constraints for the model. This is a problem, even for the simpler approaches (e.g., the DCA), which still require the user to specify the residual covariances between the product indicators manually. This becomes particularly problematic when the estimation of interaction effects using more complicated approaches is desirable, where the constraints can become quite complicated and tedious to specify.

1.5.2. Nlsem

The nlsem package allows users to estimate interaction effects using both the LMS and QML. While nlsem has some support for converting a model specified using the lavaan syntax to a model object suitable for the nlsem package, it requires several extra steps from the user. For example, the user needs to create a matrix that stores the data in a pre-specified and rigid way (e.g., the columns must be sorted in a specific order), as well as specifying the starting values for the parameters in the model. By default, the mean structure of the latent variables (ξ_1, ξ_2, η_1) is estimated freely, where the intercept for the first indicator for each latent variable is set to zero. This is not the default assumption made in the LMS and QML approach (Klein & Moosbrugger, 2000; Klein & Muthén, 2007), where (alongside other constraints) it is assumed that ξ_1 and ξ_2 have zero mean. The nlsem package is implemented purely in R (as opposed to a low-level language like C or C++), which in turn makes the estimation process quite slow. Lastly, and more importantly, the nlsem package only allows the user to estimate interaction effects for models with a single endogenous variable, making it unsuitable for a large set of models.

1.6. The Need for Better and Freely Available Tools

While it is possible to estimate interaction effects in SEMs using various tools, it can often be a tedious and error-prone task. One exception is Mplus, having implemented a

[2] https://modsem.org/articles/interaction_two_etas.html

version of the LMS approach. While it is quite easy to estimate models with interaction effects in Mplus, if is also closed-source and quite expensive – making it less accessible to many researchers. Here, we present modsem, an open-source and freely available R package, which implements multiple approaches for estimating latent interaction effects. modsem uses an extended version of the lavaan syntax (Rosseel, 2012), which many R users are already familiar with. Since modsem relies on lavaan for model specification and fitting (in case of the PI approaches), the package is easy to learn and compatible with many different types of structural equation models. Furthermore, modsem does not only support latent × latent interactions, but also latent × observed and observed × observed interactions. modsem automatically detects these interactions and handles them appropriately.modsem supports the product indicator-based approaches and also implements the LMS and QML approaches. The product indicator-based approaches are estimated using lavaan (Rosseel, 2012), whilst the LMS and QML approaches are estimated using a custom implementation. To make the estimation process more efficient, the LMS and QML were partially implemented in C++. Unlike the nlsem package, the QML and LMS approaches in modsem are available for models with multiple endogenous variables, where interaction effects can be estimated between both endogenous and exogenous latent variables.

2. The modsem Package

To install modsem, the latest stable release can be directly downloaded from the Comprehensive R Archive Network (CRAN), or the latest development version can be installed from GitHub as shown below.

```
# Install from CRAN
install.packages("modsem")

# Install from GitHub
install.packages("devtools")
devtools::install_github("kss2k/modsem",
build_vignettes = TRUE)
```

In order to estimate a model with modsem, it must first be specified using a modified version of the lavaan syntax (see the lavaan website[3]), where the ':' operator can be used to denote an interaction (similarly to the syntax used in the lm() command). An SEM can be specified in lav–aan through a simple set of equations, which are stored in a string (i.e., a character vector with a single element). Strings in R can be expressed using either single or double quotes, but it is considered good practice to use single quotes for lavaan models, as double quotes can be used in the lavaan syntax. The outer/measurement model is specified through the '=~' operator, where the left-hand side (LHS) denotes the latent variable, and the right-hand side (RHS) denotes the corresponding indicators. To make the model more readable, the '+' operator can be used to denote that multiple variables are part of the same expression.

To specify the inner/structural model, the ' ' operator is used, where the LHS denotes the dependent variable, and the RHS denotes the independent variables. In the following example, we add a new latent variable 'Y' to the model and specify that it is a function of 'X':

```
syntax < - '
  # Outer Model
  X =~ x1 + x2 + x3
  Y =~ y1 + y2 + y3
  # Inner model
  Y ~ X
'
```

Similarly to the '= ' operator, the '+' operator can be used to add extra variables to the expression. Adding a new latent variable 'Z' to the model, we can see an example where 'Y' is a function of both 'X' and 'Z'

```
syntax < - '
  # Outer Model
  X =~ x1 + x2 + x3
  Y =~ y1 + y2 + y3
  Z =~ z1 + z2 + z3
  # Inner model
  Y ~ X + Z
'
```

On top of this, there are multiple other operators (and combinations of operators) which can be used to generate more complicated models, such as the ' ' operator, which is used for specifying the (co-)variances of the exogenous variables in the model, as well as residual (co-)variances for the endogenous and observed variables in the model. Furthermore, the '*' operator can be used to define labels for the parameters in the model, which can be used to generate simple constraints. For more complicated constraints, labels can be used in conjunction with mathematical expressions specified using the '==', ' < ', ' > ' operators. A full explanation of these features is beyond this introduction, but can be found in the lavaan documentation on CRAN[4] or on the lavaan website. One of the goals of modsem is that the user should be able to specify their model in as simple a manner as possible and thus avoid a lot of these more complicated features.

In modsem, models can be specified using the modsem() function, where the syntax is specified as a string using the lavaan syntax. The only difference is that an interaction effect can be added using the ':' operator. Thus we can modify our last example, where there now is an interaction effect between 'X' and 'Z' (denoted by 'X:Z') on 'Y'

```
syntax < - '
  # Outer Model
  X =~ x1 + x2 + x3
```

[3] https://lavaan.ugent.be/

[4] https://cran.r-project.org/web/packages/lavaan/index.html

```
  Y =~ y1 + y2 + y3
  Z =~ z1 + z2 + z3
  # Inner model
  Y ~ X + Z + X:Z
`
```

To estimate the model, you can use the modsem() function, where the first argument is the syntax, the second argument is the data, and the third is the method. Thus the user only has to change the method argument if they wish to change the approach used to estimate the model. In addition to the mentioned approaches, modsem can also estimate the model via Mplus, given that the user has a version of Mplus installed on their computer.

```
# using default (double centering) approach
modsem(syntax, data = data)
# constrained approach
modsem(syntax, data = data, method = “ca”)
# unconstrained approach
modsem(syntax, data = data, method = “uca”)
# double centering approach
modsem(syntax, data = data, method = “dblcent”)
# residual centering approach
modsem(syntax, data = data, method = “rca”)
# LMS approach
modsem(syntax, data = data, method = “lms”)
# QML approach
modsem(syntax, data = data, method = “qml”)
# Mplus
modsem(syntax, data = data, method = “mplus”)
```

Here, we present an example using the QML approach on a simulated dataset (oneInt) included in the modsem package. Using the summary() function, we can print a summary of the models. For models estimated using one of the PI approaches, the lavaan summary will be returned (since they are estimated using lavaan). For the LMS and QML approaches, modsem will return a similar output inspired by the summary() function from lavaan. If standardized = TRUE is specified in the summary() function, the package will print the standardized estimates instead of the unstandardized estimates.

```
syntax < - `
  # Outer Model
  X =~ x1 + x2 + x3
  Y =~ y1 + y2 + y3
  Z =~ z1 + z2 + z3
  # Inner model
  Y ~ X + Z + X:Z
`

est_qml < - modsem(syntax, data = oneInt,
method = “qml”)
summary(est_qml, standardized = TRUE)
```

```
modsem (version 1.0.3):
  Estimator                                   QML
  Optimization method                      NLMINB
  Number of observations                     2000
  Number of iterations                        116
  Loglikelihood                         -17496.22
  Akaike (AIC)                           35054.43
  Bayesian (BIC)                         35228.06

Fit Measures for H0:
  Loglikelihood                            -17832
  Akaike (AIC)                           35723.75
  Bayesian (BIC)                         35891.78
  Chi-square                                17.52
  Degrees of Freedom (Chi square)              24
  P value (Chi square)                      0.826
  RMSEA                                     0.000

Comparative fit to H0 (no interaction
effect)
  Loglikelihood change                     335.66
  Difference test (D)                      671.32
  Degrees of freedom (D)                        1
  P value (D)                               0.000

R-Squared:
  Y                                         0.607

R-Squared Null-Model (H0):
  Y                                         0.395

R-Squared Change:
  Y                                         0.211

Parameter Estimates:
  Coefficients                       standardized
  Information                            observed
  Standard errors                        standard

Latent Variables:
     Estimate Std.Error   z.value   P(>|z|)
X =~
  x1    0.928
  ×2    0.892     0.014     63.96     0.000
  ×3    0.913     0.013     67.80     0.000
Z =~
  z1    0.927
  z2    0.899     0.014     65.12     0.000
  z3    0.913     0.013     67.62     0.000
Y =~
  y1    0.968
  y2    0.953     0.009    107.57     0.000
  y3    0.960     0.009    112.55     0.000
Regressions:
     Estimate Std.Error   z.value   P(>|z|)
Y ~
  X     0.432     0.021     20.94     0.000
  Z     0.369     0.019     18.96     0.000
  X:Z   0.460     0.018     25.45     0.000
Covariances:
     Estimate Std.Error   z.value   P(>|z|)
X ~~
  Z     0.200     0.024      8.24     0.000
  Variances:
     Estimate Std.Error   z.value   P(>|z|)
x1      0.138     0.008     18.14     0.000
×2      0.204     0.009     23.19     0.000
×3      0.167     0.008     20.82     0.000
z1      0.140     0.008     18.34     0.000
z2      0.192     0.009     22.62     0.000
z3      0.167     0.008     20.71     0.000
y1      0.062     0.003     17.98     0.000
y2      0.092     0.004     22.67     0.000
y3      0.078     0.004     20.71     0.000
×       1.000     0.037     26.99     0.000
Z       1.000     0.037     26.95     0.000
Y       0.393     0.016     24.87     0.000
```

`modsem` also offers some post-estimation tools. For example, you can use the `plot_interaction()` function to visualize the interaction effects (see Figure 2). It is similar to the `margins` and `marginsplot` commands in `STATA` (StataCorp, 2023):

```
plot_interaction(x = “X”, z = “Z”, y = “Y”,
                 xz = “X:Z”,
  vals_z = c(-1, −0.7), model = est_qml)
```

In the `modsem` package, you can visualize the interaction effects using the `plot_interaction()` function. The parameters for this function are defined as follows:

- `x`: The name of the variable on the x-axis.
- `z`: The name of the moderating variable.
- `y`: The name of the variable on the y-axis.
- `xz`: The name of the interaction term.
- `vals_z`: The values of `z` at which to plot the effect of `x` on `y`.
- `model`: A model fitted using `modsem`.

This function enables researchers to visualize how the interaction between variables influences the dependent variable, making it easier to interpret the results of the model.

3. Application to Real-Life Datasets

3.1. Application 1 (Interaction Effect): Theory of Planned Behaviour

The Theory of Planned Behaviour (TPB) is a social psychological theory used to predict human behaviour. The theory posits that *behaviour* is a function of *intention*, which in turn is a function of *attitude*, *subjective norm* (SN), and *perceived behavioural control* (PBC) (Ajzen, 1991). According to Ajzen (1991), a "*behavioral intention can find expression in behavior only if the behavior in question is under volitional control.*" In other words, Ajzen (1991) specifies an interaction effect where an individual's PBC will moderate the effect of their *intentions* on their *behaviour*, indicating that a high PBC will increase the effect of *intentions* on *behaviour*. The model is visualized in Figure 3

The data is taken from a UK sample, used in a replication study by Hagger et al. (2023). The data was collected at two different timepoints, with each latent variable having between 3 and 6 indicators, each specified as a 7-point Likert scale. For simplicity, four indicators were selected for each latent variable here (ignoring the differences in timepoint), resulting in each latent variable having 4 indicators with high communalities (≥ 0.7). The `lavaan` syntax for the model is as follows:

```
  model < - `
  # Outer Model (Based on Hagger et al.,
2007)
```

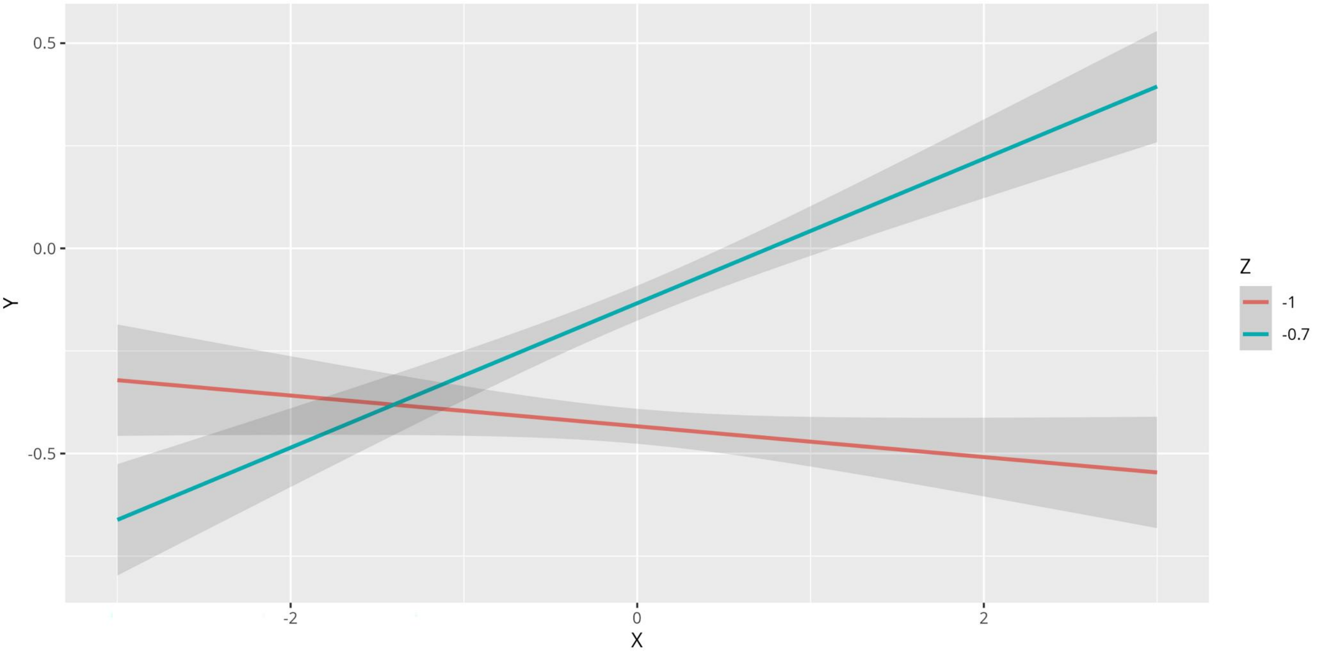

Figure 2. Plot of the marginal effect of *X* on *Y* given *Z*, created using the `plot_interaction()` function.

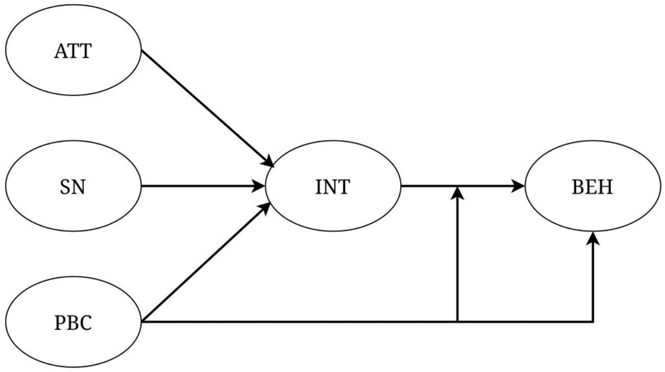

Figure 3. Theory of Planned behaviour.

```
  ATT =~ att3 + att2 + att1 + att4
  SN =~ sn4 + sn2 + sn3 + sn1
  PBC =~ pbc2 + pbc1 + pbc3 + pbc4
  INT =~ int2 + int1 + int3 + int4
  BEH =~ beh3 + beh2 + beh1 + beh4

  # Inner Model (Based on Steinmetz et al.,
2011)
  # Causal Relationships
  INT ~ ATT + SN + PBC
  BEH ~ INT + PBC
  BEH ~ INT:PBC
  `
```

The models were estimated using the CA, UCA, RCA, DCA, LMS, and QML approaches. For comparison, we also estimated a model using `Mplus` (called through `modsem`). The code used to estimate the models is as follows:

```
est_ca <- modsem(model, data = TPB_UK, method
= “ca”)
est_uca <- modsem(model, data = TPB_UK, method
= “uca”)
est_dca <- modsem(model, data = TPB_UK, method
= “dblcent”)
est_rca <- modsem(model, data = TPB_UK, method
= “rca”)
est_lms <- modsem(model, data = TPB_UK, method
= “lms”, nodes = 100) # for increased precision
est_qml <- modsem(model, data = TPB_UK, method
= “qml”)
est_mplus <- modsem(model, data = TPB_UK, method
= “mplus”)
```

The structural coefficients can be seen in Tables 1 and 2. In general, the estimates for the interaction effect were quite similar across the different approaches. The models using QML and CA yielded an interaction effect ($INT \cdot PBC$) around 0.13, LMS and `Mplus` around 0.14, and UCA and DCA around 0.15.

3.2. Application 2 (Quadratic Effect): Jordan Sample from PISA 2006 Survey

This example was obtained from Umbach et al. (2017), where the authors tested a model with two quadratic effects and a single interaction effect, using a dataset included in the `nlsem` package. The data originally stemmed from the large-scale assessment study Program for International Student Assessment (PISA) 2006 (Organisation for Economic Co-Operation and Development, 2009), which measured the reading, mathematics, and science proficiency of 15-year-old students.

Table 1. Estimates from the PI approaches using a UK sample for the TPB model.

Approach	*Y*	*X*	Estimate	Std. error	*p* value
CA	**INT**	**ATT**	−0.093	0.029	.001
		SN	0.039	0.034	.245
		PBC	1.105	0.042	<.001
	BEH	**INT**	0.581	0.048	<.001
		PBC	0.437	0.054	<.001
		INT·PBC	0.132	0.006	<.001
UCA	**INT**	**ATT**	−0.056	0.027	.039
		SN	0.049	0.033	.133
		PBC	1.031	0.035	<.001
	BEH	**INT**	0.570	0.049	<.001
		PBC	0.415	0.053	<.001
		INT·PBC	0.147	0.009	<.001
DCA	**INT**	**ATT**	−0.068	0.028	.015
		SN	0.043	0.033	.197
		PBC	1.049	0.036	<.001
	BEH	**INT**	0.638	0.047	<.001
		PBC	0.350	0.051	<.001
		INT·PBC	0.150	0.011	<.001
RCA	**INT**	**ATT**	−0.053	0.028	.054
		SN	0.042	0.033	.205
		PBC	1.034	0.036	<.001
	BEH	**INT**	0.753	0.047	<.001
		PBC	0.241	0.050	<.001
		INT·PBC	0.153	0.011	<.001

Table 2. Estimates from the DA approaches using a UK sample for the TPB model.

Approach	*Y*	*X*	Estimate	Std. error	*p* value
LMS	**INT**	**ATT**	−0.060	0.027	.027
		SN	0.051	0.032	.112
		PBC	1.036	0.035	<.001
	BEH	**PBC**	0.404	0.050	<.001
		INT	0.601	0.046	<.001
		INT·PBC	0.140	0.005	<.001
QML	**INT**	**ATT**	−0.051	0.030	.081
		SN	0.047	0.033	.150
		PBC	1.027	0.036	<.001
	BEH	**PBC**	0.361	0.053	<.001
		INT	0.629	0.048	<.001
		INT·PBC	0.130	0.008	<.001
Mplus	**INT**	**ATT**	−0.053	0.031	.089
		SN	−0.065	0.024	.008
		PBC	1.090	0.036	<.001
	BEH	**PBC**	0.405	0.052	<.001
		INT	0.588	0.048	<.001
		INT·PBC	0.141	0.008	<.001

In their paper, Umbach et al. (2017) tested a model where career (CAREER) aspirations were predicted by enjoyment of science (ENJ) and academic self-concept in science (SC), where the relationship between ENJ and CAREER was moderated by SC. Furthermore, the model included the quadratic effects of both ENJ and SC on CAREER. The model can be seen in Figure 4.

The `lavaan` syntax for the model is as follows:

```
  model <-'
  # Measurement Model
```

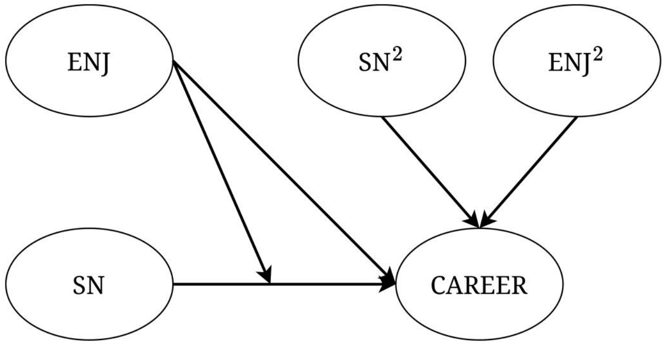

Figure 4. Model from Umbach et al. (2017).

```
  ENJ    =~ enjoy1 + enjoy2 + enjoy3 +
              enjoy4 + enjoy5
  CAREER =~ career1 + career2 + career3 +
              career4
  SC     =~ academic1 + academic2 + academic3+
             academic4 + academic5 + academic6
  # Structural Model
  CAREER ~ ENJ + SC + ENJ:ENJ + SC:SC + ENJ:SC
`
```

The models were estimated using the CA, UCA, DCA, RCA, LMS and QML approaches. As in the previous example, we also estimated a model using `Mplus`, for comparison. The code used to estimate the models is as follows:

```
est_dca <- modsem(model, data = jordan)
est_ca <- modsem(model, data = jordan, method
= "ca")
est_uca <- modsem(model, data = jordan, method
= "uca")
est_rca <- modsem(model, data = jordan, method
= "rca")
est_lms < - modsem(model, data = jordan, method
= "lms", nodes = 15) # actually 15 ^2
est_qml <- modsem(model, data = jordan, method
= "qml")
est_mplus <- modsem(model, data = jordan, method
= "mplus")
```

The structural coefficients can be seen in Tables 3 and 4. While the coefficients for the TPB model were relatively similar across the different approaches, the results for the PISA model gave drastically different outcomes depending on the approach used. In general, the estimates using LMS, QML, RCA, CA, and `Mplus` are quite similar, where both the quadratic effects (ENJ^2 and SC^2) and the interaction effect ($ENJ \cdot SC$) are relatively small. Notably, only the quadratic effect of ENJ was significant, when estimated using CA and RCA. The UCA and DCA approaches, however, yielded larger estimates for both the interaction effect and the quadratic effects, where the coefficients were multiple times larger than those estimated using the other approaches.

Table 3. Estimates from the PI approaches using a Jordan Sample from the PISA-2006 survey.

Approach	Y	X	Estimate	Std. error	p value
CA	**CAREER**	**ENJ**	0.537	0.019	<.001
		SC	0.420	0.022	<.001
		SC^2	−0.005	0.026	.839
		ENJ^2	0.054	0.016	.001
		ENJ·SC	−0.022	0.030	.466
UCA	**CAREER**	**ENJ**	0.674	0.033	<.001
		SC	0.444	0.030	<.001
		SC^2	0.046	0.022	.038
		ENJ^2	0.167	0.025	<.001
		ENJ·SC	−0.140	0.041	.001
RCA	**CAREER**	**ENJ**	0.521	0.019	<.001
		SC	0.464	0.023	<.001
		SC^2	0.005	0.022	.806
		ENJ^2	0.041	0.017	.018
		ENJ·SC	−0.017	0.038	.659
DCA	**CAREER**	**ENJ**	0.674	0.033	<.001
		SC	0.444	0.030	<.001
		SC^2	0.046	0.022	.038
		ENJ^2	0.167	0.025	<.001
		ENJ·SC	−0.140	0.041	.001

Table 4. Estimates from the DA approaches using a Jordan sample from the PISA-2006 survey.

Approach	Y	X	Estimate	Std. error	p value
LMS	**CAREER**	**ENJ**	0.525	0.019	<.001
		SC	0.464	0.023	<.001
		SC^2	0.002	0.034	.942
		ENJ^2	0.026	0.024	.278
		ENJ·SC	−0.049	0.047	.290
QML	**CAREER**	**ENJ**	0.523	0.020	<.001
		SC	0.467	0.024	<.001
		SC^2	−0.002	0.037	.958
		ENJ^2	0.026	0.022	.236
		ENJ·SC	−0.039	0.048	.415
Mplus	**CAREER**	**ENJ**	0.524	0.020	<.001
		SC	0.466	0.023	<.001
		SC^2	0.002	0.033	.957
		ENJ^2	0.026	0.021	.216
		ENJ·SC	−0.047	0.043	.273

It is beyond the scope of the current paper to discuss why the estimates differ so much between the different approaches. However, it is worth noting that such discrepancies become apparent only when using multiple approaches on the same dataset, something that has not been commonly done previously. As a consequence, the `modsem` package offers exciting new opportunities to study the effect of different estimation approaches in various contexts through large-scale simulation studies.

4. Conclusion

Estimating interaction and quadratic effects in SEMs typically requires specialized software or complex model specifications that are prone to error and misspecification. In this paper, we introduce the R package `modsem`, designed to facilitate the estimation of models with interaction and quadratic effects using a modified version of the `lavaan` syntax, which is familiar to many R users. The package supports multiple estimation methods, including the LMS and QML approaches, alongside the more commonly used PI approaches. Previously, the LMS and QML approaches were limited to models with a single endogenous variable. To our knowledge, `Mplus` is the only program that allows for more complex models using a modified version of the LMS approach as presented by Klein and Moosbrugger (2000), and no program has yet implemented an extended version of the QML approach.

modsem addresses this gap by implementing extended versions of both the LMS and QML approaches as experimental features. While initial results are promising, further validation and refinement are necessary to ensure their robustness. Future work should focus on this validation process and the exploration of alternative methods for estimating interaction effects in SEMs. For instance, Bayesian approaches using Markov Chain Monte Carlo (MCMC) have shown promising results (Marsh et al., 2013). Additionally, recent developments by the lavaan team, such as the planned implementation of latent interaction estimation using a structural after measurement (SAM[5]) (Rosseel & Loh, 2021), indicate exciting future directions for the field.

[5] https://lavaan.ugent.be/about/donate.html

References

Ajzen, I. (1991, December) The theory of planned behavior. *Organizational Behavior and Human Decision Processes*, *50*, 179–211. https://www.sciencedirect.com/science/article/pii/074959789190020T https://doi.org/10.1016/0749-5978(91)90020-T

Algina, J., & Moulder, B. C. (2001). A note on estimating the Jöreskog–Yang model for latent variable interaction using LISREL 8.3. *Structural Equation Modeling: A Multidisciplinary Journal*, *8*, 40–52. https://doi.org/10.1207/S15328007SEM0801_3

Contributors, S. (2016). *semTools: Useful tools for structural equation modeling*. https://CRAN.R-project.org/package=semTools

Geldhof, G. J., Pornprasertmanit, S., Schoemann, A. M., & Little, T. D. (2013, February). Orthogonalizing through residual centering: Extended applications and caveats. *Educational and Psychological Measurement*, *73*, 27–46. https://doi.org/10.1177/0013164412445473

Hagger, M. S., Chatzisarantis, N. L. D., Barkoukis, V., Wang, J. C. K., Hein, V., Pihu, M., Soós, I., & Karsai, I. (2007). Cross-cultural generalizability of the theory of planned behavior among young people in a physical activity context. *Journal of Sport and Exercise Psychology*, *29*, 1–19.

Hagger, M. S., Hamilton, K., Liu, H., Ajzen, I., Bosnjak, M., & Schmidt, P. (2023, January). *Dataset for: Testing the replicability of the theory of planned behavior: a large-scale multi-sample registered replication study*. PsychArchives. https://psycharchives.org/en/item/758b296e-c5f6-4dee-aa3b-fbc25ae6be25

Jaccard, J., & Wan, C. K. (1995). Measurement error in the analysis of interaction effects between continuous predictors using multiple regression: Multiple indicator and structural equation approaches. *Psychological Bulletin*, *117*, 348–357. https://doi.org/10.1037/0033-2909.117.2.348

oreskog, K. G., & Yan, F. (1996). Nonlinear structural equation models: The Kenny–Judd model with interaction effects. In G. A. Marcoulides & R. E. Schumacker (Eds.), *Advanced Structural Equation Modeling* (pp. 57–88). Lawrence Erlbaum Associates, Inc.

Jin, S., Vegelius, J., & Yang-Wallentin, F. (2020). A marginal maximum J¨ likelihood approach for extended quadratic structural equation modeling with ordinal data. *Structural Equation Modeling: A Multidisciplinary Journal*, *27*, 864–873. (eprint: https://doi.org/10.1080/10705511.2020.1712552

Kenny, D. A., & Judd, C. M. (1984). Estimating the nonlinear and interactive effects of latent variables. *Psychological Bulletin*, *96*, 201–210. https://doi.org/10.1037/0033-2909.96.1.201

Klein, A., & Moosbrugger, H. (2000, December) Maximum likelihood estimation of latent interaction effects with the LMS method. *Psychometrika*, *65*, 457–474. Retrieved 20231105, from https://doi.org/10.1007/BF02296338

Klein, A., & Muthén, B. O. (2007). Quasi-maximum likelihood estimation of structural equation models with multiple interaction and quadratic effects. *Multivariate Behavioral Research*, *42*, 647–673. https://doi.org/10.1080/00273170701710205

Lance, C. E. (1988). Residual centering, exploratory and confirmatory moderator analysis, and decomposition of effects in path models containing interactions. *Applied Psychological Measurement*, *12*, 163–175. https://doi.org/10.1177/014662168801200205

Lin, G.-C., Wen, Z., Marsh, H. W., & Lin, H.-S. (2010, July). Structural equation models of latent interactions: clarification of orthogonalizing and double-mean-centering strategies. *Structural Equation Modeling: A Multidisciplinary Journal*, *17*, 374–391. https://doi.org/10.1080/10705511.2010.488999

Little, T. D., Bovaird, J. A., & Widaman, K. F. (2006, December). On the merits of orthogonalizing powered and product terms: Implications for modeling interactions among latent variables. *Structural Equation Modeling: A Multidisciplinary Journal*, *13*, 497–519. https://doi.org/10.1207/s15328007sem1304_1

Marsh, H. W., Wen, Z., & Hau, K.-T. (2004). Structural equation models of latent interactions: Evaluation of alternative estimation strategies and indicator construction. *Psychological Methods*, *9*, 275–300. https://doi.org/10.1037/1082-989X.9.3.275

Marsh, H. W., Wen, Z., Hau, K.-T., & Nagengast, B. (2013). Structural equation models of latent interaction and quadratic effects, 2nd ed. In G. R. Hancock & R. O. Mueller (Eds.), *Structural equation modeling a second course* (pp. 267–308). IAP Information Age Publishing.

Martin, A., & Marsh, H. (1999). Effect of the interaction of level and stability of self-esteem on a variety of school-related outcomes: Five alternative approaches to testing interactions between latent variables. *In Meeting of the Australian Association for Research in Education*.

Mulaik, S. (2010). *Linear causal modelling with structural equalities*. CRC.

Muthén, L. K., & Muthén, B. O. (2004). *Mplus user's guide (Version 3.0)*. 1008.

Muthén, L. K., & Muthén, B. O. (2017). *Mplus user's guide*, 8th ed.

Organisation for Economic Co-Operation and Development. (2009). *Pisa 2006: Science competencies for tomorrow's world (Technical Report)*. OECD.

Rosseel, Y. (2012). lavaan: An R package for structural equation modeling. *Journal of Statistical Software*, *48*, 1–36. https://doi.org/10.18637/jss.v048.i02

Rosseel, Y., & Loh, W. W. (2021). A structural after measurement (SAM) approach to SEM. Manuscript under review. https://osf.io/pekbm

Schumacker, R. E. (2002). Latent variable interaction modeling. *Structural Equation Modeling: A Multidisciplinary Journal*, *9*, 40–54. https://doi.org/10.1207/S15328007SEM0901_3

Schumacker, R. E., & Marcoulides, G. A. (1998). *Interaction and nonlinear effects in structural equation modeling*. Lawrence Erlbaum Associates Publishers.

StataCorp. (2023). *Stata statistical software: Release 18*. StataCorp LLC.

Steinmetz, H., Davidov, E., & Schmidt, P. (2011). Three approaches to estimate latent interaction effects: Intention and perceived behavioral control in the theory of planned behavior. In *Methodological innovations* (Vol 6, pp 95–110). SAGE Publications.

Umbach, N., Naumann, K., Brandt, H., & Kelava, A. (2017, May) Fitting nonlinear structural equation models in R with Package nlsem. *Journal of Statistical Software*, *77*, 1–20. https://www.jstatsoft.org/index.php/jss/article/view/v077i07 (Section: Articles) https://doi.org/10.18637/jss.v077.i07

Wall, M., & Amemiya, Y. (2007, December). Nonlinear structural equation modeling as a statistical method. *Handbook of Latent Variable and Related Models*, 321–343 (ISBN: 9780444520449)