psych

Article

A Cautionary Note Regarding Multilevel Factor Score Estimates from Lavaan

Steffen Zitzmann

Hector Research Institute of Education Sciences and Psychology, University of Tübingen,
72072 Tübingen, Germany; steffen.zitzmann@uni-tuebingen.de

Abstract: To compute factor score estimates, lavaan version 0.6–12 offers the function lavPredict( ) that can not only be applied in single-level modeling but also in multilevel modeling, where characteristics of higher-level units such as working environments or team leaders are often assessed by ratings of employees. Surprisingly, the function provides results that deviate from the expected ones. Specifically, whereas the function yields correct EAP estimates of higher-level factors, the ML estimates are counterintuitive and possibly incorrect. Moreover, the function does not provide the expected standard errors. I illustrate these issues using an example from organizational research where team leaders are evaluated by their employees, and I discuss these issues from a measurement perspective.

Keywords: multilevel modeling; factor score; maximum likelihood estimate; expected a posteriori estimate

1. Introduction

It goes without saying that most concepts in psychology cannot directly be observed but can only be inferred from answered items of questionnaires or from other instruments. One way of obtaining individual scores on these dimensions is estimating them based on statistical models. This approach may thus be called model-based (for a discussion of this and other approaches that do not necessarily rely on statistical models, see [1–3]). There are different methods for obtaining factor score estimates from a model, with the two best known being the Bartlett or Maximum Likelihood (ML) methods [4] and the regression or Bayesian methods [5,6]. It is interesting to note that the obtained factor score estimates can differ greatly between these methods; the more they differ, the lower the reliability is.

A context in which these methods may be applied is diagnostics, where factor score estimates are used to assess an unobserved characteristic. Another context is secondary data analysis, where these estimates are used in further analyses (e.g., in regression analyses) to infer population characteristics such as the relationship between two variables of interest. However, it has been emphasized that in order to yield unbiased regression coefficients, corrections must be applied that take measurement error into account and correct the coefficients accordingly [7]. Various approaches for correcting the coefficients have been suggested in the literature (e.g., [8–10]) and are currently being further developed (e.g., [11–17]).

Types of Factor Score Estimate

In the following, it is assumed that the model is unidimensional (i.e., there is only one factor). By assuming unidimensionality, the factor score estimates can easily be defined, avoiding complicated notation. In the definitions of the factor score estimates, no specific measurement model is assumed, but later in the Motivating Example section, a measurement model will be used in order to illustrate my point.

The definitions follow from the general assumption that an unbiased estimate is equal to the population parameter plus an error and an additional assumption of normality.

check for updates

Citation: Zitzmann, S. A Cautionary Note Regarding Multilevel Factor Score Estimates from Lavaan. *Psych* **2023**, *5*, 38–49. https://doi.org/10.3390/psych5010004

Academic Editor: Okan Bulut

Received: 23 November 2022
Revised: 29 December 2022
Accepted: 3 January 2023
Published: 9 January 2023

Specifically, the ML estimate is defined as a random variable that is centered around the true value T and that follows a normal distribution. Formally, this can be expressed as:

$$\mathrm{MLE} \sim \mathrm{N}\left(T, \sigma^2\right) \tag{1}$$

The standard error of the ML estimate is:

$$SE_{\mathrm{MLE}} = \sigma \tag{2}$$

It can be argued that Equations (1) and (2) represent "a paradigm of the concept of measurement, namely: that a measurement equals the true quantity measured plus an orthogonal error of measurement" ([18], p. 513). Admittedly, normality is not necessarily assumed by this "paradigm". However, this additional assumption is made in order to facilitate the computation of the Expected A Posteriori (EAP) estimate.

Further assuming that the true value is normally distributed around zero (or in Bayesian terminology, assuming a normal prior for the true value with zero mean), the EAP estimate is simply:

$$\mathrm{EAP} = \omega \cdot \mathrm{MLE} \tag{3}$$

where ω is the reliability of the ML estimate. The standard error of the EAP estimate, which is sometimes also referred to as the measurement error, can be expressed as a function of the standard error of the ML estimate:

$$SE_{\mathrm{EAP}} = \omega \cdot SE_{\mathrm{MLE}} \tag{4}$$

See Appendix A for more details about how these equations were obtained (see also, [18–22]).

To easily compute the factor score estimates in application contexts, the software lavaan [23] could be used, which is a package for latent variable modeling in R [24]. For this purpose, lavaan version 0.6–12 offers the function lavPredict( ). However, during the course of my research, I noticed that in multilevel modeling, where characteristics of higher-level units are often assessed by ratings of lower-level units, the function's results deviated from what had been expected. For example, working environments or team leaders may be assessed by the ratings of employees. Whereas the function yielded correct EAP estimates of higher-level factors, the ML estimates deviated from the expected ones. In addition, the function did not provide the expected standard errors. Because users should be aware of these issues, they will be briefly illustrated next.

2. Motivating Example

Suppose leadership behavior, which is a characteristic of a team leader, is assessed by their employees' perceptions Y by first asking employees to rate their team leader and then averaging the ratings across the employees in the team (e.g., [25]). This mean is assumed to *reflect* the employees' shared perception and is thus conceptualized as a *latent* rather than a manifest variable (see [26] for a detailed discussion). This latent mean varies *between* teams, whereas the employee-specific deviations from this mean vary *within* a team. In M*plus* notation, the small letters b and w indicate between and within, respectively. Thus, Y can be decomposed into a latent between component Y^b and a within component Y^w that is centered around Y^b (i.e., it has a mean of zero; [27]; see also [26]). Formally, for an employee i in a team j:

$$Y_{ij} = Y_j^b + Y_{ij}^w \tag{5}$$

Note that this equation is of the type "observed indicator = latent factor + measurement error", and this is why it can also be considered a measurement model for the employees' shared perception and thus for leadership behavior. More specifically, it can be considered a parallel model because the employees are exchangeable raters that act as parallel indicators of leadership behavior (see [28–30]); that is, they have equal loadings and equal error variances.

2.1. Maximum Likelihood Estimate

Given the parallel model, the ML estimate of a team leader j's behavior is simply the average across their employees' ratings:

$$\text{MLE}_j = \frac{\sum_{i=1}^{n} Y_{ij}}{n} \tag{6}$$

and the standard error of the ML estimate is obtained by analogy to the simple normal model as:

$$SE_{\text{MLE}} = \sqrt{\frac{\text{var}(Y^w)}{n}} \tag{7}$$

where $\text{var}(Y^w)$ is the variance of Y^w (see [31]).

To obtain an expression for the reliability of the ML estimate, it is instructive to note that perceptions of the employees from the same team should be more similar than perceptions of employees from different teams. This similarity can be assessed by the Intraclass Correlation, or ICC:

$$\text{ICC} = \frac{\text{var}\left(Y^b\right)}{\text{var}\left(Y^b\right) + \text{var}(Y^w)} \tag{8}$$

Thus, it is very intuitive to consider the ICC as one determining factor of the reliability: the larger the ICC is (i.e., the more similar the perceptions are), the more reliable is the ML estimate. In addition, note that other employees would be equally suited as raters of the team leader if they were their employees. That is, the employees in a team are only a sample from a much larger population of raters. Therefore, the number of raters in a team n can be considered another determining factor of the reliability: the larger n, the higher the reliability tends to be (e.g., [26]). In the literature on multilevel modeling (e.g., [32–34]), the reliability is thus often expressed as:

$$\text{rel} = \frac{n \cdot \text{ICC}}{1 + (n-1) \cdot \text{ICC}} \tag{9}$$

It reflects the extent to which the differences in the ML estimates between team leaders can be explained by true differences in leadership behavior. For an equivalent expression obtained from generalizability theory [35], see the formula for Design B in [31].

2.2. Expected a Posteriori Estimate

Using the reliability, the EAP estimate is:

$$\text{EAP}_j = \frac{n \cdot \text{ICC}}{1 + (n-1) \cdot \text{ICC}} \cdot \frac{\sum_{i=1}^{n} Y_{ij}}{n} \tag{10}$$

and the standard error is:

$$SE_{\text{EAP}} = \frac{n \cdot \text{ICC}}{1 + (n-1) \cdot \text{ICC}} \cdot \sqrt{\frac{\text{var}(Y^w)}{n}} \tag{11}$$

See Appendix B for the derivation.

In the next section, the factor score estimates obtained from lavaan will be investigated with the help of an artificial dataset and compared with "custom-built" estimates.

3. Factor Score Estimates from Lavaan

3.1. Data and Method

Suppose 1000 employees from 100 teams evaluated their team leader by indicating their level of agreement with whether their team leader is able to set goals and to support the team in achieving these goals on a five-point scale from −2 (disagree) to 2 (fully agree). Artificial data were generated according to the model in Equation (5), adapting the procedure of [36]. As the number of employees surveyed in a team was 10, and the ICC was assumed to be 0.10, the reliability was 0.53. Using these values, each team leader's true score was simulated, as well as their employees' ratings. More specifically, each team leader's true score was drawn from a normal distribution with a mean of 0.0 and a variance of 0.1. Their 10 employees' ratings were drawn from a normal distribution with a mean equal to the team leader's true value and a variance of 0.9.

The (grand-mean) centered data can be downloaded at https://figshare.com/articles/dataset/Example_data_of_A_Cautionary_Note_Regarding_Multilevel_Factor_Score_Estimates_from_Lavaan/21613872. Each line contains the rating of a single employee. Column g, the grouping variable, indicates which team leader is evaluated. After having downloaded them, they can be read into R for analysis.

The function lavPredict( ) in lavaan version 0.6–12 [23] is a general way for obtaining factor score estimates from (almost) arbitrary models with latent variables. However, before the factor score estimates can be computed, the model needs to be specified as:

```
mlm <- '
      level: 1
            Yw =~ a*Y
      level: 2
            Yb =~ a*Y
'
```

Lavaan applies the within–between framework of M*plus*, which decomposes variables into within and between components. Accordingly, the syntax has two parts: the within part and the between part. The within part of the syntax contains the definition of a lower-level factor, which is indicated by the observed variable. Its loading is fixed at 1 to identify the metric of the latent variable, which is lavaan's default strategy. As there is only one item, the error variance is defaulted to 0. In the between part, the higher-level factor, which represents leadership behavior, is defined analogously. However, the indicator's loading was constrained to be equal to the corresponding loading at the lower level, which is often done in multilevel modeling (e.g., [37–40]). If not explicitly specified, lavaan would default the loading to 1, which—for this specific model—would lead to the same numerical results. A summary of the output from lavaan is shown in Appendix C.

3.2. Results

Factor score estimates are obtained from the model by the lavPredict( ) function, which allows users to choose between the two discussed types of factor score estimates. ML estimates are obtained by specifying the option method = "Bartlett", whereas EAP estimates are obtained by method = "regression". Note that the option level = 2 has to be specified because the factor score estimates of interest are those from the higher-level factor. Normally, applying the attributes( ) function to the output objects would yield their standard errors. However, here, both standard errors are NA. This issue will be elaborated further in the Discussion section.

The obtained factor score estimates look as follows:

```
              MLE SE_MLE         EAP SE_EAP
 [1,]  0.01350424     NA  0.01350424     NA
 [2,]  0.32330489     NA  0.32330489     NA
 [3,] -0.17971389     NA -0.17971389     NA
 [4,]  0.03329527     NA  0.03329527     NA
 [5,] -0.13949739     NA -0.13949739     NA
 [6,] -0.63425187     NA -0.63425187     NA
 [7,] -0.21836302     NA -0.21836302     NA
 [8,]  0.51805248     NA  0.51805248     NA
 [9,]  0.05040566     NA  0.05040566     NA
[10,] -0.35429797     NA -0.35429797     NA
    .
    .
    .
```

From this table's columns MLE and EAP, it can be seen that the ML and EAP estimates that the lavPredict( ) function provided are numerically equal, meaning that they cannot be distinguished from one another. This is surprising given that from the definition of the EAP estimate in Equation (3), it immediately follows that the ML estimate should generally be larger (in absolute value) than the EAP estimate by a factor of 1/reliability. One would expect the equivalence only if the reliability is perfect; that is, if it is equal to one. However, computing the reliability by Equation (9) using the model results from lavaan yielded a value of 0.60, which is much smaller than 1.00. Hence, one or both of the factor score estimates from lavaan must deviate from the expected ones. In order to determine which one deviates, "custom-built" estimates are computed according to Equations (6) and (10), using the model results from lavaan (i.e., the estimates of $\mathrm{var}(Y^w)$ and $\mathrm{var}(Y^b)$) as plug-in estimates:

```
               MLE    SE_MLE          EAP     SE_EAP
 [1,]   0.02267735 0.301777   0.01350424 0.1797065
 [2,]   0.54291830 0.301777   0.32330489 0.1797065
 [3,]  -0.30178930 0.301777  -0.17971389 0.1797065
 [4,]   0.05591196 0.301777   0.03329527 0.1797065
 [5,]  -0.23425468 0.301777  -0.13949739 0.1797065
 [6,]  -1.06508424 0.301777  -0.63425187 0.1797065
 [7,]  -0.36669188 0.301777  -0.21836302 0.1797065
 [8,]   0.86995334 0.301777   0.51805248 0.1797065
 [9,]   0.08464504 0.301777   0.05040566 0.1797065
[10,]  -0.59496425 0.301777  -0.35429797 0.1797065
     .
     .
     .
```

As can be seen from this table's columns MLE and EAP, the ML estimate is larger than the EAP estimate, which is in accordance with the expectation. Comparing the table's column MLE with the corresponding column in the table with the results from lavaan, it becomes evident that the ML estimate provided by lavaan deviates from the "custom-built" one. However, whereas the ML estimates differ, the EAP estimate from lavaan is equal to the "custom-built" estimate, indicating that lavaan provides the correct EAP estimate. Thus, it is the ML estimate from lavaan that is possibly incorrect. The findings can be reproduced by the R code in Appendix D.

4. Discussion

Lavaan version 0.6–12 offers the lavPredict( ) function, which could be applied to obtain factor score estimates, even in multilevel modeling, where characteristics of higher-level units are often assessed by ratings of lower-level units. With the help of an example from organizational research, it has been shown that the function provides results that deviate from the expected ones, particularly the ML estimate.

One explanation for the finding that the ML estimate was numerically equal to the EAP estimate is that Yves Rosseel, the programmer and mastermind behind lavaan, has made a mistake. Although this scenario certainly lies within the realms of possibility, it appears not very likely. Another possible, more likely explanation is that the reliability is assumed to be perfect (i.e., no measurement error), and Yves Rosseel himself brought this up after he had read a preprint version of this article (Y. Rosseel, personal communication, 10 December 2022). Notice that the employees in a team were assumed to be exchangeable raters from a larger population of raters, resulting in an imperfect reliability. This assumption can, however, be debated, particularly when one does not want to generalize to a team leader's true value that would be obtained if all raters from the population of raters evaluated the team leader. For example, rather than in the team leader's behavior towards employees in general, one may be interested in how the team leader behaves in their specific team. In this case, the formula in Equation (9) does not apply, and the reliability is indeed perfect (i.e., it is equal to 1), yielding:

$$\mathrm{EAP}_j = 1 \cdot \frac{\sum_{i=1}^{n} Y_{ij}}{n} = \mathrm{MLE}_j \tag{12}$$

It is interesting to ask whether this can also explain why lavPredict( ) does not provide standard errors, neither for the ML nor the EAP estimates?

By expressing the standard error of the ML estimate as a function of the reliability and assuming a perfect reliability, the standard errors of the ML and EAP estimates become:

$$SE_{\mathrm{MLE}} = \sqrt{\frac{1 - \frac{n \cdot \mathrm{ICC}}{1+(n-1) \cdot \mathrm{ICC}}}{\frac{n \cdot \mathrm{ICC}}{1+(n-1) \cdot \mathrm{ICC}}} \cdot \mathrm{var}\left(Y^B\right)} = \sqrt{\frac{1-1}{1} \cdot \mathrm{var}\left(Y^B\right)} = 0 \tag{13}$$

and

$$SE_{\mathrm{EAP}} = 1 \cdot 0 = 0 \tag{14}$$

Hence, even if the aim is *not* generalization across raters (i.e., a perfect reliability), standard errors will not be absent! Rather than NA, the result of the lavPredict( ) function should be zero for each of the standard errors in this case.

My example model is a simple two-level model with only one item. However, more realistic models have more than one item. Thus, it is interesting to ask whether findings would be similar in multilevel Confirmatory Factor Analysis (CFA) models. To address this question, artificial data were generated according to a simple extension of the example model, namely, a two-level CFA model with 12 items, assuming that the items are parallel indicators (i.e., they have equal loadings and equal error variances; see [17,41]). The findings from this model can be summarized as follows. First, as in the simple model, the lavPredict( ) function yielded the correct EAP estimate of the higher-level factor as verified by the custom-built EAP estimate. In the more complex model, the ML estimate differed from the EAP estimate, and it stood to reason that the ML estimate is correct. However, although in this model the reliability was smaller than one, it still deviated from the expected result, and so did the ML estimate. The deviation between the ML estimate and the custom-built ML estimate could again be explained by the fact that the function took unreliability due to surveying only a limited number of raters not into account, whereas it correctly took unreliability due to the items into account. Thus, only in the special case where one does not want to generalize across raters are the ML estimate and its standard error correct! Moreover, it was found that only in this case, the standard error of the EAP estimate was close to the expected one. However, it was slightly larger because the function provides the prediction error instead of the measurement error (for an in-depth discussion of these types of standard error, see [18]).

Conclusions

To conclude, the function lavPredict( ) yields correct EAP estimates of higher-level factors, but given my assumptions, the ML estimates are possibly incorrect. Fortunately, as EAP estimates are essentially Bayes estimates, they can be more accurate than ML estimates (i.e., they exhibit a smaller mean-squared error; e.g., [42–44]). Therefore, EAP estimates may be the better choice in diagnostics (e.g., [45,46]) and secondary data analysis (e.g., [9]) anyway. However, lavaan does not provide the expected standard errors, which users should, as a minimum, keep in mind when using them.

Funding: This research received no external funding.

Data Availability Statement: The data can be downloaded at https://figshare.com/articles/dataset/Example_data_of_A_Cautionary_Note_Regarding_Multilevel_Factor_Score_Estimates_from_Lavaan/21613872.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Derivation of Equations (1)–(4)

To derive the equations, the general assumption that an (unbiased) estimate is equal to the population parameter plus an error is used. Accordingly, the ML estimate can be expressed as:

$$\text{MLE} = T + E \tag{A1}$$

where—applying McDonald's analogy of the ML estimate to a measure—T is the true value and E is the measurement error. This error has a variance of σ^2. Assuming normality of the error in order to facilitate computation, the equation can be written as:

$$\text{MLE} \sim \text{N}\left(T, \sigma^2\right) \tag{A2}$$

that is, Equation (1) in the main body of the text. The standard error of the ML estimate is simply the square root of the error variance:

$$SE_{\text{MLE}} = \sigma \tag{A3}$$

that is, Equation (2) in the text.

If the additional assumption that the true score is normally distributed around zero is made, which is equivalent to selecting a normal prior for the true value with zero mean and variance τ^2:

$$T \sim \text{N}\left(0, \tau^2\right) \tag{A4}$$

and this prior is combined with the (normal) likelihood in Equation (A2), the posterior is obtained as:

$$T \sim \text{N}\left(\frac{\tau^2}{\tau^2 + \sigma^2} \cdot \text{MLE}, \frac{\tau^2}{\tau^2 + \sigma^2} \cdot \sigma^2\right) \tag{A5}$$

The mean of this distribution is the EAP estimate:

$$\text{EAP} = \frac{\tau^2}{\tau^2 + \sigma^2} \cdot \text{MLE} \tag{A6}$$

Note that $\tau^2/\left(\tau^2 + \sigma^2\right)$ is the ratio of true score variance to observed variance; that is, the reliability of the ML estimate. Using ω as an abbreviation for this reliability, the equation becomes:

$$\text{EAP} = \omega \cdot \text{MLE} \tag{A7}$$

that is, Equation (3) in the text. The standard error of the EAP estimate or, more precisely, its measurement error, is yielded by taking the square root of its variance and inserting Equations (A7) and (A3):

$$SE_{\text{EAP}} = \sqrt{\text{var}(\text{EAP})} = \sqrt{\text{var}(\omega \cdot \text{MLE})} = \sqrt{\omega^2 \cdot \sigma^2} = \omega \cdot SE_{\text{MLE}} \tag{A8}$$

that is, Equation (4) in the text.

Appendix B. Derivation of Equations (10) and (11)

The measurement model for a team leader j's behavior reads:

$$Y_{ij} = Y_j^b + Y_{ij}^w \tag{A9}$$

Assuming normality of the within component, the equation can also be expressed as:

$$Y_{ij} = \text{N}\left(Y_j^b, \text{var}(Y^w)\right) \tag{A10}$$

The team-specific likelihood is:

$$\frac{\sum_{i=1}^{n} Y_{ij}}{n} \sim \mathrm{N}\left(Y_j^b, \frac{\mathrm{var}(Y^w)}{n}\right) \tag{A11}$$

If the additional assumption that the latent between component is normally distributed around zero is made (equivalent to selecting a normal prior with a mean of zero):

$$Y_j^b \sim \mathrm{N}\left(0, \mathrm{var}\left(Y^b\right)\right) \tag{A12}$$

and this prior is combined with Equation (A11), the following posterior is obtained:

$$Y_j^b \sim \mathrm{N}\left(\frac{\mathrm{var}(Y^b)}{\mathrm{var}(Y^b) + \mathrm{var}(Y^w)/n} \cdot \frac{\sum_{i=1}^{n} Y_{ij}}{n}, \frac{\mathrm{var}(Y^b)}{\mathrm{var}(Y^b) + \mathrm{var}(Y^w)/n} \cdot \frac{\mathrm{var}(Y^w)}{n}\right) \tag{A13}$$

Thus, the team leader's EAP estimate (i.e., the mean of the posterior) is:

$$\mathrm{EAP}_j = \frac{\mathrm{var}(Y^b)}{\mathrm{var}(Y^b) + \mathrm{var}(Y^w)/n} \cdot \frac{\sum_{i=1}^{n} Y_{ij}}{n} \tag{A14}$$

Expressing $\mathrm{var}(Y^b)$ as a function of the ICC and simplifying, the EAP estimate becomes:

$$\mathrm{EAP}_j = \frac{n \cdot \mathrm{ICC}}{1 + (n-1) \cdot \mathrm{ICC}} \cdot \frac{\sum_{i=1}^{n} Y_{ij}}{n} \tag{A15}$$

that is, Equation (10) in the text. From this equation, the standard error of the EAP estimate is yielded by computing the square root of its variance:

$$SE_{\mathrm{EAP}} = \sqrt{\mathrm{var}\left(\frac{n \cdot \mathrm{ICC}}{1 + (n-1) \cdot \mathrm{ICC}} \cdot \frac{\sum_{i=1}^{n} Y_{ij}}{n}\right)} = \frac{n \cdot \mathrm{ICC}}{1 + (n-1) \cdot \mathrm{ICC}} \cdot \sqrt{\frac{\mathrm{var}(Y^w)}{n}} \tag{A16}$$

that is, Equation (11) in the text.

Appendix C. Lavaan Output

```
lavaan 0.6-12 ended normally after 14 iterations
  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                         3
  Number of observations                          1000
  Number of clusters [g]                           100

Model Test User Model:
  Test statistic                                 0.000
  Degrees of freedom                                 0

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Observed
  Observed information based on                Hessian


Level 1 [within]:

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  Yw =~
```

```
    Y           (a)    1.000

Intercepts:
                    Estimate  Std.Err  z-value  P(>|z|)
   .Y                  0.000
    Yw                 0.000

Variances:
                    Estimate  Std.Err  z-value  P(>|z|)
   .Y                  0.000
    Yw                 0.911    0.043   21.213    0.000


Level 2 [g]:

Latent Variables:
                    Estimate  Std.Err  z-value  P(>|z|)
  Yb =~
    Y           (a)    1.000

Intercepts:
                    Estimate  Std.Err  z-value  P(>|z|)
   .Y                 -0.000    0.047   -0.000    1.000
    Yb                 0.000

Variances:
                    Estimate  Std.Err  z-value  P(>|z|)
   .Y                  0.000
    Yb                 0.134    0.032    4.173    0.000
```

Appendix D. R Code

The following R code can be used to reproduce the findings from the Factor Score Estimates from Lavaan section.

```
# Set working directory
wd <- file.path( "C:/MyFolder" )
setwd( wd )


# Read example data

exampleData <- read.table( "exampleData.txt", header = T, sep = "\t"
   )

J <- length( unique( exampleData$g ) )
nn <- NA for ( i in 1:J ) {
      nn[ i ] <- length( which( exampleData$g == 1 ) ) }
n <- mean( nn )


# Specify and run the example model in lavaan

# install.packages( "lavaan" )
library( lavaan )

mlm <- '
      level: 1
            Yw =~ a*Y
      level: 2
```

```
             Yb =~ a*Y
'

fit <- sem( mlm, data = exampleData, cluster = "g" )


# Obtain factor scores from lavaan

lavMles <- lavPredict( fit, level = 2, method = "Bartlett", se = "
    standard" )

lavEaps <- lavPredict( fit, level = 2, method = "regression", se = "
    standard" )

lavMles.se <- attributes( lavMles )$se[[1]]
lavMles <- cbind( lavMles, rep( lavMles.se[1], J ) )
colnames( lavMles ) <- c( "MLE", "SE_MLE" )

lavEaps.se <- attributes( lavEaps )$se[[1]]
lavEaps <- cbind( lavEaps, rep( lavEaps.se[1], J ) )
colnames( lavEaps ) <- c( "EAP", "SE_EAP" )

lavFscores <- cbind( lavMles, lavEaps )
print( fscores[ 1:10,  ] )

# Obtain "custom-built" factor scores

mles <- rep( NA, J )
for ( i in 1:J ){
      mles[ i ] <- mean( as.numeric( dat$Y[ which( dat$g == i ) ] ) )
}

params <- parameterEstimates(fit)
var.Yb <- params[ which( params$lhs == "Yb" & params$op == "~~" &
    params$rhs == "Yb" & params$level == 2 ), 7 ]
var.Yw <- params[ which( params$lhs == "Yw" & params$op == "~~" &
    params$rhs == "Yw" & params$level == 1 ), 7 ]
rel <- var.Yb/(var.Yb + var.Yw/n)
eaps <- rel*mles

mle.se <- sqrt( var.Yw/n )
mles <- cbind( mles, rep( mle.se, J ) ) colnames( mles ) <- c( "MLE",
     "SE_MLE" )

eap.se <- rel*mle.se
eaps <- cbind( eaps, rep( eap.se, J ) )
colnames( eaps ) <- c( "EAP", "SE_EAP" )

fscores <- cbind( mles, eaps )
print ( fscores[ 1:10,  ] )
```

References

1. Edelsbrunner, P.A. A model and its fit lie in the eye of the beholder: Long live the sum score. *Front. Psychol.* **2022**, *13*, 1–5. [CrossRef] [PubMed]
2. Robitzsch, A.; Lüdtke, O. Some thoughts on analytical choices in the scaling model for test scores in international large-scale assessment studies. *Meas. Instruments Soc. Sci.* **2022**, *4*, 9. [CrossRef]
3. Widaman, K.F.; Revelle, W. Thinking thrice about sum scores, and then some more about measurement and analysis. *Behav. Res. Methods* **2022**, *Advance Online Publication*. [CrossRef]
4. Bartlett, M.S. The statistical conception of mental factors. *Br. J. Psychol. Gen. Sect.* **1937**, *28*, 97–104. [CrossRef]

5. Thomson, G.H. The meaning of 'i' in the estimate of 'g'. *Br. J. Psychol. Gen. Sect.* **1934**, *25*, 92–99. [CrossRef]
6. Thurstone, L.L. *The Vectors of Mind*; University of Chicago Press: Chicago, IL, USA, 1935.
7. Skrondal, A.; Laake, P. Regression among factor scores. *Psychometrika* **2001**, *4*, 563–576. [CrossRef]
8. Croon, M.A. Using predicted latent scores in general latent structure models. In *Latent Variable and Latent Structure Modeling*; Marcoulides, G., Moustaki, I., Eds.; Lawrence Erlbaum: Mahwah, NJ, USA, 2002; pp. 195–223.
9. Croon, M.A.; van Veldhoven, M.J.P.M. Predicting group-level outcome variables from variables measured at the individual level: A latent variable multilevel model. *Psychol. Methods* **2007**, *12*, 45–57. [CrossRef] [PubMed]
10. Grilli, L.; Rampichini, C. The role of sample cluster means in multilevel models. *Methodology* **2011**, *7*, 121–133. [CrossRef]
11. Devlieger, I.; Rosseel, Y. Factor score path analysis: An Alternative for SEM? *Methodology* **2017**, *13*, 31–38. [CrossRef]
12. Kelcey, B.; Cox, K.; Dong, N. Croon's bias-corrected factor score path analysis for small- to moderate-sample multilevel structural equation models. *Organ. Res. Methods* **2019**, *24*, 55–77. [CrossRef]
13. Devlieger, I.; Rosseel, Y. Multilevel factor score regression. *Multivar. Behav. Res.* **2020**, *55*, 600–624. [CrossRef]
14. Aydin, B.; Algina, J. Best linear unbiased prediction of latent means in three-level data. *J. Exp. Educ.* **2021**, *90*, 452–468. [CrossRef]
15. Zitzmann, S.; Helm, C. Multilevel analysis of mediation, moderation, and nonlinear effects in small samples, using expected a posteriori estimates of factor scores. *Struct. Equ. Model.* **2021**, *28*, 529–546. [CrossRef]
16. Rosseel, Y.; Loh, W.W. A structural after measurement (SAM) approach to structural equation modeling. *Psychol. Methods* **2022**, *accepted*.
17. Zitzmann, S.; Lohmann, J.F.; Krammer, G.; Helm, C.; Aydin, B.; Hecht, M. A Bayesian EAP-based nonlinear extension of Croon and van Veldhoven's model for analyzing data from micro-macro multilevel designs. *Mathematics* **2022**, *10*, 842. [CrossRef]
18. McDonald, R.P. Measuring latent quantities. *Psychometrika* **2011**, *76*, 511–536. [CrossRef] [PubMed]
19. Bolstad, W.M.; Curran, J.M. *Introduction to Bayesian Statistics*; Wiley: Hoboken, NJ, USA, 2017.
20. Mislevy, R.J. Randomization-based inference about latent variables from complex samples. *Psychometrika* **1991**, *56*, 177–196. [CrossRef]
21. Hoff, P.D. *A First Course in Bayesian Statistical Methods*; Springer Texts in Statistics; Springer: New York, NY, USA, 2009.
22. Lüdtke, O.; Robitzsch, A. Einführung in die Plausible-Values-Technik für die psychologische Forschung [An introduction to the plausible value technique for psychological research]. *Diagnostica* **2017**, *63*, 193–205. [CrossRef]
23. Rosseel, Y. lavaan: An R package for structural equation modeling. *J. Stat. Softw.* **2012**, *48*, 1–36. [CrossRef]
24. R Development Core Team. *R: A Language and Environment for Statistical Computing*; R Development Core Team: Vienna, Austria, 2016.
25. Lüdtke, O.; Robitzsch, A.; Trautwein, U.; Kunter, M. Assessing the impact of learning environments: How to use student ratings of classroom or school characteristics in multilevel modelling. *Educ. Psychol.* **2009**, *34*, 120–131. [CrossRef]
26. Lüdtke, O.; Marsh, H.W.; Robitzsch, A.; Trautwein, U.; Asparouhov, T.; Muthén, B.O. The multilevel latent covariate model: A new, more reliable approach to group-level effects in contextual studies. *Psychol. Methods* **2008**, *13*, 203–229. [CrossRef] [PubMed]
27. Asparouhov, T.; Muthén, B.O. Constructing Covariates in Multilevel Regression (Mplus Web Notes No. 11, Version 2). 2007. Available online: https://www.statmodel.com/download/webnotes/webnote11.pdf (accessed on 1 November 2022).
28. Mehta, P.D.; Neale, M.C. People are variables too: Multilevel structural equations modeling. *Psychol. Methods* **2005**, *10*, 259–284. [CrossRef]
29. Koch, T.; Schultze, M.; Holtmann, J.; Geiser, C.; Eid, M. A multimethod latent state-trait model for structurally different and interchangeable methods. *Psychometrika* **2017**, *82*, 17–47. [CrossRef] [PubMed]
30. Zitzmann, S.; List, M.; Lechner, C.; Hecht, M.; Krammer, G. Reporting factor score estimates of teaching quality based on student ratings back to teachers: Recommendations from Psychometrics. *Educ. Psychol. Meas.* 2022, *submitted*.
31. Schweig, J.D. Quantifying error in survey measures of school and classroom environments. *Appl. Meas. Educ.* **2014**, *27*, 133–157. [CrossRef]
32. Kane, M.T.; Brennan, R.L. The generalizability of class means. *Rev. Educ. Res.* **1977**, *47*, 267–292. [CrossRef]
33. Bliese, P.D. Within-group agreement, non-independence, and reliability: Implications for data aggregation and analysis. In *Multilevel Theory, Research, and Methods in Organizations: Foundation, Extensions, and New Directions*; Klein, K.J., Kozlowski, S.W., Eds.; Jossey-Bass: San Francisco, CA, USA, 2000; pp. 349–381.
34. Snijders, T.A.B.; Bosker, R.J. *Multilevel Analysis: An Introduction to Basic and Advanced Multilevel Modeling*, 2nd ed.; Sage: Los Angeles, CA, USA, 2012.
35. Brennan, R.L. *Generalizability Theory*; Springer: New York, NY, USA, 2001.
36. Zitzmann, S.; Lüdtke, O.; Robitzsch, A. A Bayesian approach to more stable estimates of group-level effects in contextual studies. *Multivar. Behav. Res.* **2015**, *50*, 688–705. [CrossRef]
37. Lüdtke, O.; Marsh, H.W.; Robitzsch, A.; Trautwein, U. A 2 × 2 taxonomy of multilevel latent contextual models: Accuracy-bias trade-offs in full and partial error correction models. *Psychol. Methods* **2011**, *16*, 444–467. [CrossRef] [PubMed]
38. Stapleton, L.M.; Yang, J.S.; Hancock, G.R. Construct meaning in multilevel settings. *J. Educ. Behav. Stat.* **2016**, *41*, 481–520. [CrossRef]
39. Zitzmann, S.; Lüdtke, O.; Robitzsch, A.; Marsh, H.W. A Bayesian approach for estimating multilevel latent contextual models. *Struct. Equ. Model.* **2016**, *23*, 661–679. [CrossRef]

40. Zitzmann, S.; Weirich, S.; Hecht, M. Using the effective sample size as the stopping criterion in Markov chain Monte Carlo with the Bayes Module in M*plus*. *Psych* **2021**, *3*, 336–347. [CrossRef]
41. Zitzmann, S. A computationally more efficient and more accurate stepwise approach for correcting for sampling error and measurement error. *Multivar. Behav. Res.* **2018**, *53*, 612–632. [CrossRef] [PubMed]
42. Greenland, S. Principles of multilevel modelling. *Int. J. Epidemiol.* **2000**, *29*, 158–167. [CrossRef] [PubMed]
43. Zitzmann, S.; Lüdtke, O.; Robitzsch, A.; Hecht, M. On the performance of Bayesian approaches in small samples: A comment on Smid, McNeish, Miočević, and van de Schoot (2020). *Struct. Equ. Model.* **2021**, *28*, 40–50. [CrossRef]
44. Zitzmann, S.; Helm, C.; Hecht, M. Prior specification for more stable Bayesian estimation of multilevel latent variable models in small samples: A comparative investigation of two different approaches. *Front. Psychol.* **2021**, *11*, 611267. [CrossRef] [PubMed]
45. Lord, F.M.; Novick, M.R. *Statistical Theories of Mental Test Scores*; Addison-Wesley: Reading, MA, USA, 1968.
46. Zitzmann, S.; Bardach, L.; Horstmann, K.; Ziegler, M.; Hecht, M. Quantifying individual personality change more accurately by regression-based change scores. *Multivariate Behav. Res.* 2022, *submitted*.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.