Conducting Multilevel Confirmatory Factor Analysis Using R

Francis L. Huang
University of Missouri

Abstract

Clustered data are a common occurrence in the social and behavioral sciences and pose a challenge when analyzing data using confirmatory factor analysis (CFA). In addition to potentially compromising point estimates and standard errors, factor structures may also differ between levels of analysis when using nested data. However, multilevel CFA (MCFA) can address these concerns and although the procedures for performing MCFA have been proposed over a decade ago, the practice has seen little use in applied psychometric research. This article presents a step-by-step procedure for conducting a MCFA with R using the **lavaan** package. The dataset and complete R syntax, as well as a function for generating the required matrices, are provided.

Keywords: multilevel confirmatory factor analysis, nested data structures, **lavaan**.

1. Introduction

The analyses of nested data is fairly common in social and behavioral research where naturally occurring clustered data structures (e.g., students within schools, patients within hospitals) are found. Ignoring the clustered nature of the data violates the well-known assumption of observation independence (Cohen, Cohen, West, and Aiken 2003). In a regression framework, researchers will often use multilevel modeling (Raudenbush and Bryk 2002) or some other alternative technique (Huang 2016) to account for the clustered nature of the data. However, in a factor analytic framework, nested data structures are often ignored despite warnings that "the application of covariance models to multilevel data without accounting for the dependencies among observations is a potentially dangerous practice" (Julian 2001, p. 342).

Although the procedures for performing multilevel confirmatory factor analyses (MCFA) were outlined over a decade ago (Hox 2002; Muthen 1994), the practice has seen infrequent use in applied research (Byrne, 2012). Konold, Cornell, Huang, Meyer, Lacey, Nekvasil, Heilbrun, and Shukla (2014) suggested several reasons why this may be the case: 1) limited number of software packages capable of automatically running such analyses; 2) estimation and convergence issues; or 3) a failure to recognize the nested data structure when present. Heck and Thomas (2008) indicated only a few years ago that getting the software to estimate multilevel factor analytic models were "programming nightmares for even simple within- and between-group factor models" (p. 114).

In this article, we discuss the relevance of MCFA and outline the steps for performing a MCFA using the freely available R software with the **lavaan** (latent variable analysis; Rosseel

2012) package. Though several books have documented how to perform factor analysis using R (e.g., Beaujean 2014; Finch and French 2015), procedures for conducting a MCFA are not readily available and as of yet are not built-in **lavaan**. Results are then compared to MCFA conducted using Mplus.

1.1. The need for multilevel CFA

Properly accounting for the clustered nature of the data is not merely a technical issue. Not accounting for clustering in factor analysis can result in biased parameter estimates, misestimated standard errors, and a distorted view of model fit (Julian 2001; Kaplan and Elliott 1997; Muthen and Satorra 1995). Although several techniques can partially account for the clustered nature of the data by adjusting standard errors (e.g., demeaning the data, using the `type = complex` option in Mplus), these procedures assume that factor structures at the individual and group levels are the same. If factor structures are the same at both levels, factor structures are referred to as invariant (Schweig 2013), homologous (Chen, Bliese, and Mathieu 2005), or isomorphic (Kozlowski and Klein 2000). Unfortunately, the assumption of factor model invariance may often be violated in practice (Zyphur, Kaplan, and Christian 2008). Although invariance is often used in the comparison of factor models across different groups, we use the term invariance in this article to refer to differences in the factor structures at the between- and within-levels of analysis.

In many situations, individual level data are collected and aggregated to form group-level scales (Chan 1998). Two common examples from educational research include the measurement of school climate and the student ratings of teacher effectiveness. In both instances, both the individual- and group-level composites are meaningful though the group-level aggregates are of particular interest and the basis of policy relevant decisions. Constructs themselves may have different interpretations based on the level analysis (Bliese 2000; Roux 2004) and some constructs may have meaning at the individual level (e.g., personality traits), the group level (e.g., racial diversity), or both (e.g., individual feelings of safety vs. a school safety scale). In such cases, factor analytic techniques are frequently used to provide a basis for combining individual item responses to form the scales of interest.

However, studies have shown that nested data may have factor structures that differ by level of analysis and thus may result in erroneously formed composites (D'Haenens, Van Damme, and Onghena 2010; Dyer, Hanges, and Hall 2005; Huang, Cornell, and Konold 2014). For example, a review of school climate measures has shown that of the dozen instruments investigated, none were analyzed using MCFA and often only used traditional single-level CFA (Ramelow, Currie, and Felder-Puig 2015) despite school climate being a property of the school and not of any single individual reporter (Griffith 1997; van Horn 2003). Group-level composites formed on the basis of factor structures derived from single-level CFA may result in misleading conclusions (Schweig 2013). Drawing incorrect conclusions about the relationship of variables between groups based on individual-level data has been referred to as an atomistic fallacy (Roux 2002).

1.2. Decomposing the within- and between-group covariance matrices

Traditionally, clustered data have been analyzed using CFA by focusing on either the lowest level of measurement (i.e., scores from individuals) or aggregating scores to the higher level of measurement (i.e., group averaged scores) and then using single-level analysis (Heck 2001).

However, using single-level analysis for the analysis of multilevel data may not be optimal and is associated with a set of analytic and interpretation difficulties (see Byrne 2012).

Compared to single-level CFA, MCFA allows researchers to consider both levels of data simultaneously. More specifically, MCFA involves partitioning the total population covariance matrix, $\mathbf{\Sigma_T}$, into a within-covariance matrix, $\mathbf{\Sigma_W}$, and a between-covariance matrix, $\mathbf{\Sigma_B}$, to estimate both within- and between-cluster effects. The two variance components are orthogonal and additive which means that the relationship among variables between groups do not have to be the same (but they could be) as the relationship that exists within groups.

Using sample data, the total (or overall) covariance matrix, $\mathbf{S_T}$, can also be decomposed into $\mathbf{S_B}$ and $\mathbf{S_W}$ matrices. However, running a MCFA using the $\mathbf{S_B}$ and $\mathbf{S_W}$ matrices to estimate both $\mathbf{\Sigma_W}$ and $\mathbf{\Sigma_B}$ is not as straightforward (Hox 2002). Instead, two sample covariance matrices need to be defined: $\mathbf{S_{PW}}$, the pooled within covariance matrix and $\mathbf{S_B}$, the between group covariance matrix.

The $\mathbf{S_{PW}}$ matrix is an unbiased estimate of the population within groups covariance matrix, $\mathbf{\Sigma_W}$ (Muthen 1994). The pooled within covariance matrix is calculated by:

$$\mathbf{S_{PW}} = (n - G)^{-1} \sum_{g=1}^{G} \sum_{i=1}^{n_g} (\mathbf{y_{ig}} - \mathbf{\overline{y}_g})(\mathbf{y_{ig}} - \mathbf{\overline{y}_g})'$$

where n is the total sample size, G is the number of groups, $\mathbf{y_{ig}}$ is the score of observation i nested in group g and $\overline{y}_g$ is the cluster specific mean in group g. $\mathbf{S_{PW}}$ is also equivalent to the covariance matrix of individual deviation scores from the group means with the exception that the denominator is $n-G$ instead of $n-1$. Factor analyzing the $\mathbf{S_{PW}}$ matrix is straightforward and does not present any modeling challenges. A simple way to generate the $\mathbf{S_{PW}}$ can be done by group-mean centering all the variables of interest, generating a covariance matrix using the centered variables, multiplying the covariance matrix by $n - 1$, and then dividing the product by $n - G$.

The sample between-group covariance matrix $\mathbf{S_B}$ can be calculated using:

$$\mathbf{S_B} = (G - 1)^{-1} \sum_{g=1}^{G} n_g (\mathbf{y_g} - \mathbf{\overline{y}})(\mathbf{y_g} - \mathbf{\overline{y}})'$$

where $\mathbf{\overline{y}}$ represents the overall grand mean. Similarly, $\mathbf{S_B}$ can be computed by generating a covariance matrix using the deviation scores of the repeating group means from the overall grand mean, multiplying the matrix by $n - 1$ to compute the sums of squares, and then dividing again by $G-1$. Unfortunately, $\mathbf{S_B}$ is a biased estimator of $\mathbf{\Sigma_B}$ and actually estimates a combination of both $\mathbf{\Sigma_W}$ and $\mathbf{\Sigma_B}$ such that $\mathbf{S_B} = \mathbf{\Sigma_W} + c.\mathbf{\Sigma_B}$ where $c.$ represents the average cluster size (Muthen 1994). For unbalanced cases (which is most often the case), $c.$ is computed as:

$$c. = [n^2 - \sum_{g=1}^{G} n_g^2][n(G-1)]^{-1}$$

and in many instances, $c.$ will be approximately n/G. As a result, $\mathbf{\Sigma_B}$ can be roughly estimated by $c.^{-1}(\mathbf{S_B} - \mathbf{S_W})$. The expected value then of $\mathbf{\Sigma_B}$ is comprised of one unit of within-group variance and $c.$ units of between-group variance.

Performing the between-group portion of the CFA model, when not done automatically using software such as Mplus, requires the use of an unconventional, manual multigroup CFA analysis wherein the sample within and between matrices are used simultaneously with a specific set of constraints. Although articles and book chapters have illustrated how to conduct the analyses using software such as EQS, LISREL (Stapleton 2006) and Mplus (Dyer *et al.* 2005), no article shows how to perform this procedure using freely available software using R which is the focus of this manuscript.

2. The process for performing a MCFA

As Muthen (1994) noted that MCFA may not always converge, methodologists have recommended step-by-step procedures to allow researchers to carefully build their models and debug issues that may arise. Two popular and similar procedures were proposed by Muthen (1994) and Hox (2002). Of the two, Muthen's method is most commonly used though Hox's steps have been said to be the most straightforward (Selig, Card, and Little 2008). Both procedures require the $\mathbf{S_{PW}}$ and $\mathbf{S_B}$ matrices and the estimated *c.* scaling factor.[1] We describe the procedures outlined by Hox (2002) though keeping in mind that the same setup can also be performed following Muthen's (1994) steps.

2.1. Conducting a MCFA using clustered data

To illustrate the procedures used in performing a MCFA, we use the R software environment (R Core Team 2016) with the **lavaan** (Rosseel 2012) package installed. Several freely available tutorials on using **lavaan** are available (Rosseel 2016).[2] We provide a function, `mcfa.input()`, that can be used to generate all the necessary matrices used in the analyses based on the raw data. The function can be loaded into R by using the statement:

```
R> source('http://faculty.missouri.edu/huangf/data/mcfa/mcfa.R')
```

We will analyze a random subset of data from a school climate dataset where 3,894 teachers from 254 schools provided their perceptions of student engagement (Huang and Cornell 2015a). Six questions (variables x1 to x6) asked teachers about their perceptions of student engagement (see Table 1 for the questions and descriptive statistics) and response options used a six point scale (1 = strongly disagree, 2 = disagree, 3 = somewhat disagree, 4 = somewhat agree, 5 = agree, 6 = strongly agree).

The procedures outlined use the covariance matrices as the inputs for the analyses. Prior published studies (Huang, Cornell, Konold, Meyer, Lacey, Nekvasil, Heilbrun, and Shukla 2015) suggest the presence of two factors at level one (i.e., cognitive and affective engagement) and one factor at level two (i.e., an overall school-level factor of general engagement). Often, simpler factor structures are found at the higher level (Dedrick and Greenbaum 2011; Dyer *et al.* 2005; Huang *et al.* 2015).

Using the `mcfa.input()` function provided, the $\mathbf{S_B}$ and $\mathbf{S_{PW}}$ were generated along with the *c.* scaling factor which was 15.31, close to the average cluster size of 15.33. To use the function,

[1] Hox also provides a free DOS program, Split2.exe, available at http://joophox.net/papers/papers.htm to generate correlation matrices and also computes the scaling factor.

[2] An online tutorial is available at http://lavaan.ugent.be/tutorial/index.html

Variable	Item	M	SD	*Skew*	*ICC*
x1	Students generally like this school.	4.78	0.88	-1.08	0.20
x2	Students are proud to be at this school.	4.61	0.99	-0.84	0.25
x3	Students finish their homework at this school.	3.54	1.17	-0.39	0.15
x4	Students hate going to school. (reverse coded)	4.38	1.05	-0.67	0.11
x5	Getting good grades is very important to most students here.	4.24	1.10	-0.55	0.25
x6	Most students want to learn as much as they can at this school.	3.95	1.11	-0.54	0.13

Table 1: School engagement survey questions (n = 3,894 students in 254 schools).

the user must provide the name of the grouping variable as well as the dataset containing only the grouping variable and the variables of interest. Users must read in the dataset and then use the function with their dataset. For example, a csv (comma separated values) file is read into an object called `raw` and then the function is applied to the dataset specifying that `sid` (the name of the school id variable in the dataset) is the clustering variable (must be within quotes). All the output is stored into a new object `x` which can be used for the various input elements needed for the subsequent analysis.

```
R> raw <- read.csv("http://faculty.missouri.edu/huangf/data/mcfa/raw.csv")
R> x <- mcfa.input("sid", raw)
```

Using a structure function on the `x` object, `str(x)`, will display the contents of `x` and the names of the list objects within. To access the data stored within `x`, users can use the `$` notation in R to directly refer to the data. For example: for the $\mathbf{S_{PW}}$ matrix, users can enter `x$pw.cov`; for total sample size, users can enter `x$n`; for the number of groups, users can enter `x$G`.

Step 1: The level one model

The first step in Hox's (2002) procedure is to conduct a factor analysis only using the $\mathbf{S_{PW}}$ matrix, ignoring $\mathbf{S_B}$. The effective sample size for the analysis is $n - G$ or 3,640. If an adequate fit is not found, there is little point in proceeding and researchers should revisit their theory behind their CFA.

We conducted a test using both a basic one- and two-factor model. Performing a CFA in **lavaan** involves three steps: 1) specifying the model, 2) fitting the model, and 3) viewing the summary statistics. The models are specified using the syntax provided and are fit using the `cfa()` function in **lavaan**. In **lavaan** model syntax, the operator "`=~`" is short for "measured by" and is equivalent to the `by` statement in Mplus. To define a one factor model, where the factor is named `f1`, using the six manifest variables, the model specification would read:

```
R> onefactor <- 'f1 =~ x1 + x2 + x3 + x4 + x5 + x6'
```

The `onefactor` object is referred to as a model syntax object. To define the two factor model (see Figure 1), where the first factor is affective engagement and the second factor is cognitive engagement, the model specification would be:

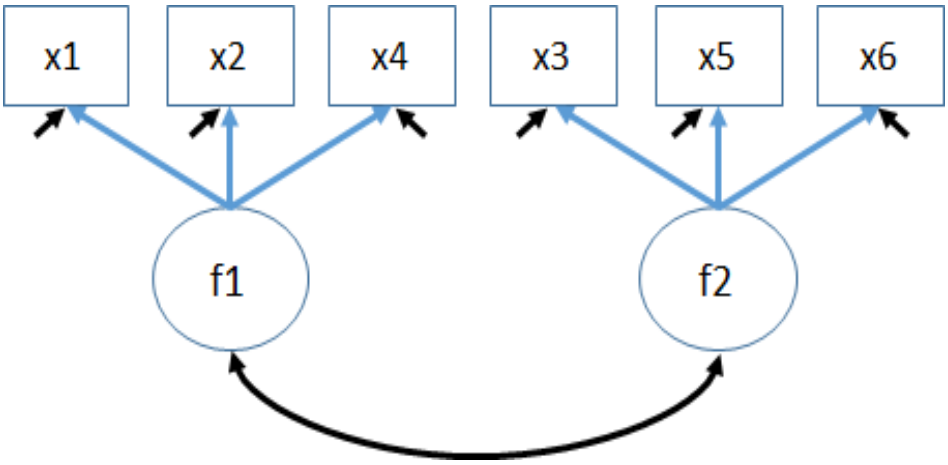

Figure 1: Single-level two factor model.

```
R> twofactor <- 'f1 =~ x1 + x2 + x4; f2= ~x3 + x5 + x6'
```

Automatically, the factor loading of the first indicator of a latent variable is fixed to 1 to set the scale of the factor, the same as the default option in Mplus. Residual variances are automatically added as well and all exogenous latent variables are correlated by default. As we will see shortly, we will have to override the default options to properly specify a multilevel factor model.

Once model object has been defined, a model is fit using the `cfa()` function where the first argument is the model object containing the model definition specified by the researcher. The second argument indicates the source and type of the data to be analyzed (`sample.cov = x$pw.cov`) and the third argument indicates the effective number of observations (`sample.nobs = x$n-x$G`). In other words, the model is being fit using the pooled within group covariance matrix as input and the effective sample size is 3,640. To fit the onefactor model example and save the output into another object (e.g., `results1`), the syntax would read:

```
R> results1 <- cfa(onefactor, sample.cov = x$pw.cov, sample.nobs = x$n - x$G)
```

After the model has been fit, the `summary()` function can provide the measures of model fit (`fit.measures = T`) and the factor loadings as is commonly seen in other latent variable modeling programs. We also request for standardized loadings using the `standardized = T` argument. The syntax would read:

```
R> summary(results1, fit.measures = T, standardized = T)
```

As we have 21 pieces of unique data (i.e., 6 variances and 15 covariances) and for a one factor model, we are estimating 5 factor loadings, 6 residual variances, and 1 factor variance; 9 degrees of freedom are left (i.e., 21-12). For a two factor model, 13 parameters are estimated leaving 8 degrees of freedom. As expected based on prior research, the one factor model did not fit the data well, $\chi^2(9) = 1{,}971.60$, RMSEA = .245, CFI = .790, TLI = .650, SRMR = .081, but the two factor model had a good fit, $\chi^2(8) = 53.95$, RMSEA = .040, CFI = .995,

TLI = .991, SRMR = .021. If all the researcher wanted was a level-one CFA model that had unbiased estimates as a result of clustering, the researchers could stop and interpret results appropriately. Group mean centered variables, as a result of demeaning the data, have been stripped of group-level effects. So far, not much is different from a standard CFA model with the exception that $\mathbf{S_{PW}}$ is used instead of $\mathbf{S_T}$ and the number of observations is $n - G$.

Step 2: The null model

For step 2, a null model is specified where both the $\mathbf{S_{PW}}$ and $\mathbf{S_B}$ matrices are used in a multigroup setup using the factor structure defined at step 1 on both matrices with all equality constraints set to be equal. We do not really have two groups but the multigroup setup will be used to analyze both the 'within group' and the 'between group' matrices simultaneously.

In **lavaan**, multiple input covariance matrices and the sample sizes for each are stored in a list object:

```
R> combined.cov <- list(within = x$pw.cov, between = x$b.cov)
R> combined.n <- list(within = x$n - x$G, between = x$G)
```

The first object in the list refers to group one and the second object refers to group two and we create two new objects (i.e., `combined.cov` and `combined.n`) that contain the two covariance matrices (i.e., $\mathbf{S_{PW}}$ and $\mathbf{S_B}$) and the sample size for each ($n - G$ and G, respectively).

Next, a model imposing the equality constraints must be specified. In this step, the model specification expands quite a bit. In **lavaan**, the equality constrains are imposed for the particular variable by indicating `c(a,a)*variable` where `c()` is the concatenate function, `a` is a label assigned by the user to indicate that loading `a` for group one is set to be equal for loading `a` in group two. The same label names instruct **lavaan** to use the same estimates between groups or in other words, specify equality constraints. To specify equal factor loadings for both factors for the within and between models, we indicate: `f1 =~ x1 + c(a,a)*x2 + c(b,b)*x4; f2 =~ x3 + c(c,c)*x5 + c(d,d)*x6`. The loadings for x1 and x3 are automatically set to 1 so do not need to be specified.

```
R> nullmodel <- '
+   f1 =~ x1 + c(a,a)*x2 + c(b,b)*x4
+   f2 =~ x3 + c(c,c)*x5 + c(d,d)*x6
+   x1 ~~ c(e,e)*x1
+   x2 ~~ c(f,f)*x2
+   x3 ~~ c(g,g)*x3
+   x4 ~~ c(h,h)*x4
+   x5 ~~ c(i,i)*x5
+   x6 ~~ c(j,j)*x6
+   f1 ~~ c(k,k)*f1
+   f2 ~~ c(l,l)*f2
+   f1 ~~ c(m,m)*f2
+ '
R> results3 <- cfa(nullmodel, sample.cov = combined.cov,
+  sample.nobs = combined.n)
R> summary(results3, fit.measures = T, standardized = T)
```

In addition to the factor loadings, the variance and covariance for each of the variables and latent factors must also set to be equal across both groups. To specify the variance/covariance of a variable or factor, the operator `~~` is used such that `x1 ~~ c(e,e)*x1` indicates that the variance for x1 will be held constant between groups. Factor variance and the covariance between the two factors must also be constrained to be equal. The statement `f1 ~~ c(m,m)*f2` indicates that the covariance between the two latent variables (f1 and f2) are constrained to be equal between models. The syntax has gotten a bit longer (see appendix) though much of this could be done through careful copying and pasting. The label names (e.g., `e` and `m`) can be set to any other labels (beginning with a letter) though must be the same to specify holding those constant between groups (or in this case, between levels).

For the null model, since we are using two matrices ($\mathbf{S_{PW}}$ and $\mathbf{S_B}$), we now have 42 pieces of unique data but are still estimating only 13 parameters instead of 26 since we have constrained them to be equal between models. As a result, 29 degrees of freedom (i.e., 42-13) are left. After specifying the model, the `cfa()` function is once again used but this time, the `combined.cov` and `combined.n` lists are used as inputs for the covariance matrix and the sample size indicator. As before, the `summary()` function is used to investigate model fit.

The resulting null model fit poorly, $\chi^2(29) = 1{,}237.56$, CFI = .890, TLI = .886, RMSEA = .146, SRMR = .229. The poor fit of the model indicates that there is between-group variance to be explained and if the null model had an acceptable fit, researchers could tentatively conclude that there appears to be no statistically significant group-level variance (Stapleton 2006).

In Muthen's (1994) steps, one step investigates how much variability is attributable to the group level which is indicated by the intraclass correlation where $\rho = (\sigma_B^2 + \sigma_W^2)^{-1}\sigma_B^2$ of each manifest variable. The within-group variance can be obtained from the diagonal of the $\mathbf{S_{PW}}$ matrix and the between-group variance can be obtained from the diagonal of the $c.1(\mathbf{S_B} - \mathbf{S_{PW}})$ matrix. The adjusted (or scaled) between group covariance matrix is automatically estimated using the `mcfa.input()` function and can be retrieved using `x$ab`. Alternatively, ICCs can be computed using an ANOVA framework where $[MSB - MSW]/[MSB + c.MSB]$. However, there is no real threshold as to what comprises a large ICC and even slight departures from zero can signify that the multilevel nature of the data should be accounted for (Julian 2001). The ICCs are also computed using the `mcfa.input()` function and ICCs can be retrieved by indicating `x$icc`. Reporting ICCs is standard practice when dealing with clustered data and for the current dataset, ICCs ranged from .11 to .25 (see Table 1).

Step 3: The independence model

For step 3, Hox (2002) proposed to estimate an independence model at the group level. As $\mathbf{S_B^*} = \mathbf{\Sigma_W} + \mathbf{c}.\mathbf{\Sigma_B}$, we begin estimating the between portion of the model by creating six new group-level 'factors' (see Figure 2). Step 3 now requires the use of the $c.$ scaling factor and each manifest variable variance is composed of one unit for $\mathbf{S_{PW}}$ and a portion of $\mathbf{S_B}$ variance. Each new group level factor has a specified loading of $\sqrt{c.}$ or 3.91 (the square root of 15.31) to its corresponding manifest variable (Muthen 1994). The square root of $c.$ is also automatically computed using the `mcfa.input()` and can be retrieved using `x$sqc`. Note, in the model syntax, modelers must manually specify the scaling factor (i.e., 3.91).

When specifying the model, additional syntax must be included to properly define the new factors that are only estimated for level 2 and not for the level 1 model. To define a new group

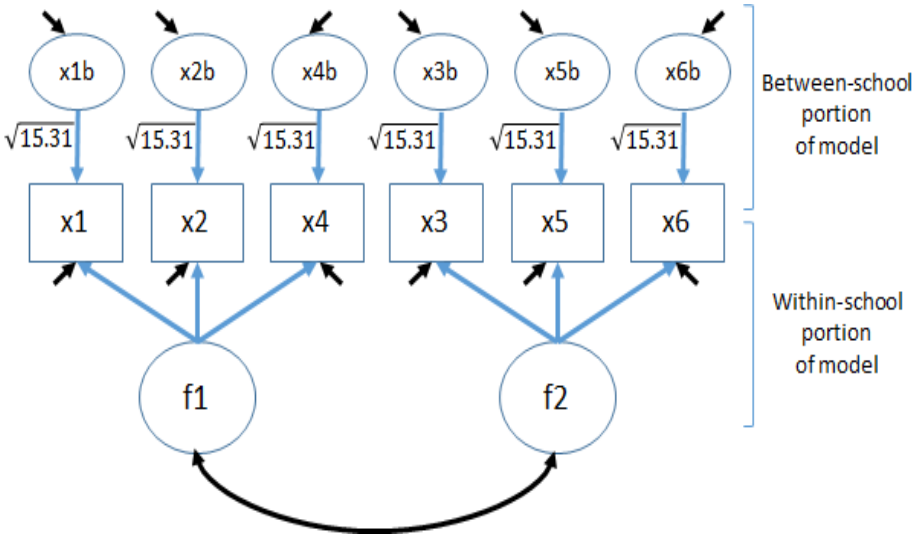

Figure 2: Two factor null model.

level factor, we use: `x1b =~ c(0,3.91)*x1` where `x1b` is the name of the level 2 factor (we use a 'b' so we remember it is a between level factor and to also differentiate it from the first level variable which is just x1) and `c(0,3.91)*x1` indicates that for the first model (which is the within model), `x1b` is not estimated as indicated by the 0 and for the second model (the between level model), the loading is fixed to 3.91 for x1. Since there are six manifest variables, we create six new latent variables, from x1b to x6b.

```
R> independence <- '
+    f1 =~ x1 + c(a,a)*x2 + c(b,b)*x4
+    f2 =~ x3 + c(c,c)*x5 + c(d,d)*x6
+    x1 ~~ c(e,e)*x1
+    x2 ~~ c(f,f)*x2
+    x3 ~~ c(g,g)*x3
+    x4 ~~ c(h,h)*x4
+    x5 ~~ c(i,i)*x5
+    x6 ~~ c(j,j)*x6
+    f1 ~~ c(k,k)*f1
+    f2 ~~ c(l,l)*f2
+    f1 ~~ c(m,m)*f2
+    x1b =~ c(0,3.91)*x1
+    x1b ~~ c(0,NA)*x1b
+    x2b =~ c(0,3.91)*x2
+    x2b ~~ c(0,NA)*x2b
+    x3b =~ c(0,3.91)*x3
+    x3b ~~ c(0,NA)*x3b
+    x4b =~ c(0,3.91)*x4
+    x4b ~~ c(0,NA)*x4b
+    x5b =~ c(0,3.91)*x5
+    x5b ~~ c(0,NA)*x5b
```

```
+   x6b =~ c(0,3.91)*x6
+   x6b ~~ c(0,NA)*x6b
+ '
R> results4 <- cfa(independence, sample.cov = combined.cov,
+ sample.nobs = combined.n, orthogonal = T)
R> summary(results4, fit.measures = T)
```

In addition to defining the latent level 2 variables, we must also estimate the variance for these latent variables using the following model statement: `x1b ~~ c(0,NA)*x1b`. Again, the "~~" indicates that this is a variance estimate and new to this is the use of `c(0,NA)*x1b` which specifies that the variance of the level 1 variable is not estimated as indicated again by the 0 and that the variance of the level 2 variable is estimated as indicated by the `NA`. This specification is important because if this is not explicitly specified, **lavaan** will attempt to estimate the variance for the latent factors in both groups even if the new `b` variables only exist for the between level model.

Although the group-level loadings are fixed for the new latent variables and do not consume any degrees of freedom, the variance for each of the latent variables must be estimated. As a result, we are left with 23 degrees of freedom. The six new latent factors are not allowed to covary at this step which is why this step is referred to as the independence model. For the `cfa()` function beginning in this step, an additional argument must be specified: `orthogonal=T`. Using this argument, only the factors we allow to covary will actually covary as all factors will be independent of each other. If this is not specified, the covariance of all the factors will be estimated which is the default for **lavaan** but is not appropriate for our model. Model fit once again is mixed though may be considered poor by a majority of fit indices, $\chi^2(23) = 815.61$, CFI = .928, TLI = .906, RMSEA = .133, SRMR = .176. If the independence model fit well, the conclusion would be that there is substantial group-level variance but there is no substantively interesting structural model (Hox 2002). If the independence model did not fit well, that suggests that there is some kind of structural model at the group level that should be modeled.

Step 4: The saturated model

For step 4, or testing the saturated model, the latent variables defined in step 3 are now allowed to covary with each other. Since there are 6 latent variables, a total of 15 covariance estimates will be calculated (i.e., $[k \times (k-1)]/2$ where k is the number of variables). Importantly, in this step, since the variables only exist at level 2, variance must be estimated only for the level 2 latent variables. This is specified by using the statement: `x5b ~~ c(0,NA)*x6b` where in this example, the covariance of the x5b and x6b latent variables are estimated only for level 2 which is indicated by an `NA`. The `NA` option indicates that the parameter will be estimated and again, the 0 indicates that the parameter will not be estimated (for level 1). A shortcut for specifying the covariance for multiple variables can be done using the form: `x4b ~~ c(0,NA)*x5b + c(0,NA)*x6b` where the covariance is estimated between x4b and x5b also for x4b and x6b. The rest of the R code merely specifies that all latent level 2 variables are correlated with each other and the `orthogonal = T` option must continue to be specified.

```
R> saturated <- '
+   f1 =~ x1+c(a,a)*x2 + c(b,b)*x4
```

```
+   f2 =~ x3+c(c,c)*x5 + c(d,d)*x6
+   x1 ~~ c(e,e)*x1
+   x2 ~~ c(f,f)*x2
+   x3 ~~ c(g,g)*x3
+   x4 ~~ c(h,h)*x4
+   x5 ~~ c(i,i)*x5
+   x6 ~~ c(j,j)*x6
+   f1 ~~ c(k,k)*f1
+   f2 ~~ c(l,l)*f2
+   f1 ~~ c(m,m)*f2
+
+   x1b =~ c(0,3.91)*x1
+   x1b ~~ c(0,NA)*x1b
+   x2b =~ c(0,3.91)*x2
+   x2b ~~ c(0,NA)*x2b
+   x3b =~ c(0,3.91)*x3
+   x3b ~~ c(0,NA)*x3b
+   x4b =~ c(0,3.91)*x4
+   x4b ~~ c(0,NA)*x4b
+   x5b =~ c(0,3.91)*x5
+   x5b ~~ c(0,NA)*x5b
+   x6b =~ c(0,3.91)*x6
+   x6b ~~ c(0,NA)*x6b
+
+   x1b ~~ c(0,NA)*x2b + c(0,NA)*x3b + c(0,NA)*x4b + c(0,NA)*x5b + c(0,NA)*x6b
+   x2b ~~ c(0,NA)*x3b + c(0,NA)*x4b + c(0,NA)*x5b + c(0,NA)*x6b
+   x3b ~~ c(0,NA)*x4b + c(0,NA)*x5b + c(0,NA)*x6b
+   x4b ~~ c(0,NA)*x5b + c(0,NA)*x6b
+   x5b ~~ c(0,NA)*x6b #fully saturated
+ '
R> results5 <- cfa(saturated, sample.cov = combined.cov,
+  sample.nobs = combined.n, orthogonal = T)
R> summary(results5, fit.measures = T, standardized = T)
```

The fit of the model in step 4 should be similar to the fit in step 1 as all degrees of freedom at the between level are used in a fully saturated model (df = 0). The variance/covariance estimates at level 2 viewed using the `summary()` function are similar to the adjusted between group variance/covariance matrix in `x$ab`. For step 4, the fit is: $\chi^2(8) = 53.95$, RMSEA = .054, CFI = .996, TLI = .984, SRMR = .020. If the fit is poor in step 4, this should be a signal that an error was made or that the model fit in step 1 was poor to begin with (Stapleton 2006). If the researcher is interested in modeling a relationship among the level 2 variables, the hypothesized relationships can be tested in the next step. Note that in Muthen's (1994) MCFA steps, the null, independence, and saturated models are not estimated.

Step 5: The hypothesized model

In step 5, the hypothesized level 2 measurement model or theoretical model is finally specified

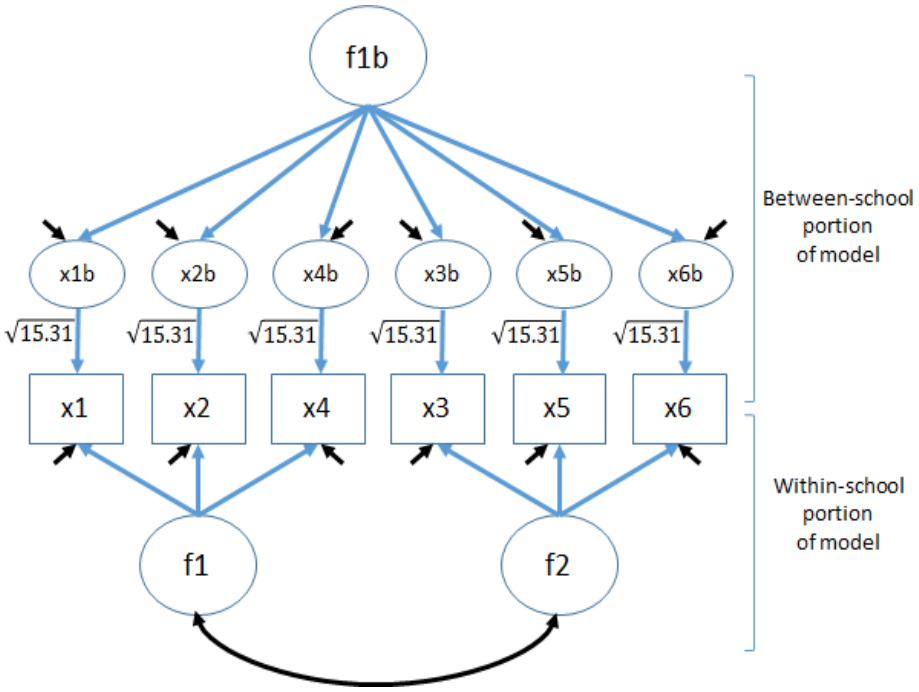

Figure 3: Hypothesized multilevel factor model with two factors at level one and one factor at level two.

and tested (see Figure 3). In this step, the covariance structure specified in step 4 for the saturated model among all the level 2 variables will be removed and replaced with the hypothesized model with one overall general factor. We hypothesized that the level 2 factors are correlated with each other as a result of one overall general level 2 factor (i.e., `bf1` or between factor 1) which we define using the following statement: `bf1 =~ c(0,1)*x1b + c(0,NA)*x2b + c(0,NA)*x3b + c(0,NA)*x4b + c(0,NA)*x5b + c(0,NA)*x6b` in the model statement. Similar to the previous steps, the `c(0,1)` or `c(0,NA)` indicates that the factor is not defined for the first model estimated (as it should not be) or the within group model as indicated by the `0`. The `1` or the `NA` indicates that the loading is set to 1 (for the first variable) to set the scale for the factor or NA to indicate that the loading will be freely estimated.

```
R> level2.1factor <- '
+   f1 = ~x1 + c(a,a)*x2 + c(b,b)*x4
+   f2 = ~x3 + c(c,c)*x5 + c(d,d)*x6
+
+   x1 ~~ c(e,e)*x1
+   x2 ~~ c(f,f)*x2
+   x3 ~~ c(g,g)*x3
+   x4 ~~ c(h,h)*x4
+   x5 ~~ c(i,i)*x5
+   x6 ~~ c(j,j)*x6
+   f1 ~~ c(k,k)*f1
+   f2 ~~ c(l,l)*f2
+   f1 ~~ c(m,m)*f2
```

```
+
+   x1b =~ c(0,3.91)*x1
+   x1b ~~ c(0,NA)*x1b
+   x2b =~ c(0,3.91)*x2
+   x2b ~~ c(0,NA)*x2b
+   x3b =~ c(0,3.91)*x3
+   x3b ~~ c(0,NA)*x3b
+   x4b =~ c(0,3.91)*x4
+   x4b ~~ c(0,NA)*x4b
+   x5b =~ c(0,3.91)*x5
+   x5b ~~ c(0,NA)*x5b
+   x6b =~ c(0,3.91)*x6
+   x6b ~~ c(0,NA)*x6b
+
+   bf1 =~ c(0,1)*x1b + c(0,NA)*x2b + c(0,NA)*x3b + c(0,NA)*x4b +
+   c(0,NA)*x5b + c(0,NA)*x6b
+   bf1 ~~ c(0,NA)*bf1 + c(0,0)*f1 + c(0,0)*f2
+ '
R> results6 <- cfa(level2.1factor, sample.cov = combined.cov,
+  sample.nobs = combined.n, orthogonal = T)
R> summary(results6, fit.measures = T, standardized = T)
```

In addition to specifying the factor to be estimated at level 2, it is important to also specify `bf1 ~~ c(0,NA)*bf1 + c(0,0)*f1 + c(0,0)*f2` which indicates that the level 2 factor variance is estimated at level 2 and that the level 2 factor is not correlated with the two factors (f1 and f2) at level 1. The resulting overall model fit is acceptable, $\chi^2(17) = 142.47$, RMSEA = .062, CFI = .989, TLI = .980, SRMR = .024. For comparative purposes, the same model was also fit using Mplus resulting in the similar fit statistics, $\chi^2(17) = 140.12$, RMSEA = .043, CFI = .989, TLI = .980, $SRMR_W = .022$, $SRMR_B = .055$ (Mplus calculates SRMR at both the within and between levels). For the standardized loadings in **lavaan**, refer to the loadings under `Std.all` retrieved using the summary statement. The fit indices and the estimated factor loadings using both **lavaan** and Mplus are comparable and are shown in Table 2.

To illustrate how to model more than one factor at the group level, the two factor model at level 1 is also replicated at level 2 (see Figure 4). To specify a two factor model at level 2, we indicate in the model:

```
R> level2.2factors <- '
+   f1 =~ x1 + c(a,a)*x2 + c(b,b)*x4
+   f2 =~ x3 + c(c,c)*x5 + c(d,d)*x6
+
+   x1 ~~ c(e,e)*x1
+   x2 ~~ c(f,f)*x2
+   x3 ~~ c(g,g)*x3
+   x4 ~~ c(h,h)*x4
+   x5 ~~ c(i,i)*x5
+   x6 ~~ c(j,j)*x6
+   f1 ~~ c(k,k)*f1
```

```
+   f2 ~~ c(l,l)*f2
+   f1 ~~ c(m,m)*f2
+
+   x1b =~ c(0,3.91)*x1
+   x1b ~~ c(0,NA)*x1b
+   x2b =~ c(0,3.91)*x2
+   x2b ~~ c(0,NA)*x2b
+   x3b =~ c(0,3.91)*x3
+   x3b ~~ c(0,NA)*x3b
+   x4b =~ c(0,3.91)*x4
+   x4b ~~ c(0,NA)*x4b
+   x5b =~ c(0,3.91)*x5
+   x5b ~~ c(0,NA)*x5b
+   x6b =~ c(0,3.91)*x6
+   x6b ~~ c(0,NA)*x6b
+
+   bf1 =~ c(0,1)*x1b + c(0,NA)*x2b + c(0,NA)*x4b
+   bf2 =~ c(0,1)*x3b + c(0,NA)*x5b + c(0,NA)*x6b #second factor
+   bf1 ~~ c(0,NA)*bf1 + c(0,0)*f1 + c(0,0)*f2 + c(0,NA)*bf2
+   bf2 ~~ c(0,NA)*bf2 + c(0,0)*f1 + c(0,0)*f2
+ '
R> results7 <- cfa(level2.2factors, sample.cov = combined.cov,
+   sample.nobs = combined.n, orthogonal = T)
R> summary(results7, fit.measures = T, standardized = T)
```

We define the between factors using the corresponding latent variables at level 2 using the `c(0,1)*` option (for the first indicator variable) or the `c(0,NA)*` option. In addition, two additional lines of code are needed to estimate the covariance between the level 2 factors but also to specify that the two factors are not correlated with any of the other factors:

```
bf1 ~~ c(0,NA)*bf1 + c(0,0)*f1 + c(0,0)*f2 + c(0,NA)*bf2;
bf2 ~~ c(0,NA)*bf2 + c(0,0)*f1 + c(0,0)*f2
```

The resulting model also had a good fit, $\chi^2(16) = 85.67$, RMSEA = .047, CFI = .994, TLI = .988, SRMR = .021. Again, for comparative purposes, the same model was also fit using Mplus resulting in similar fit statistics, $\chi^2(16) = 86.93$, RMSEA = .034, CFI = .994, TLI = .988, SRMRW = .020, SRMRB = .026. However, given that we would prefer a more parsimonious model, that prior studies have suggested one overall factor at level 2, and the presence of a very high correlations of the between level factors indicating that the between level factors are almost identical (r = .88), our preference is for the simpler one-factor model at level 2.

2.2. Estimating reliability

After completing a CFA, researchers then explore scale reliability of the formed factor. Given the clustered nature of the data analyzed, reliabilities may also differ depending on the level of interest. Cronbach's alpha (1951) is a commonly used measure to estimate reliability, though not without its limitations (Streiner 2003). However, reliability measures estimated using a

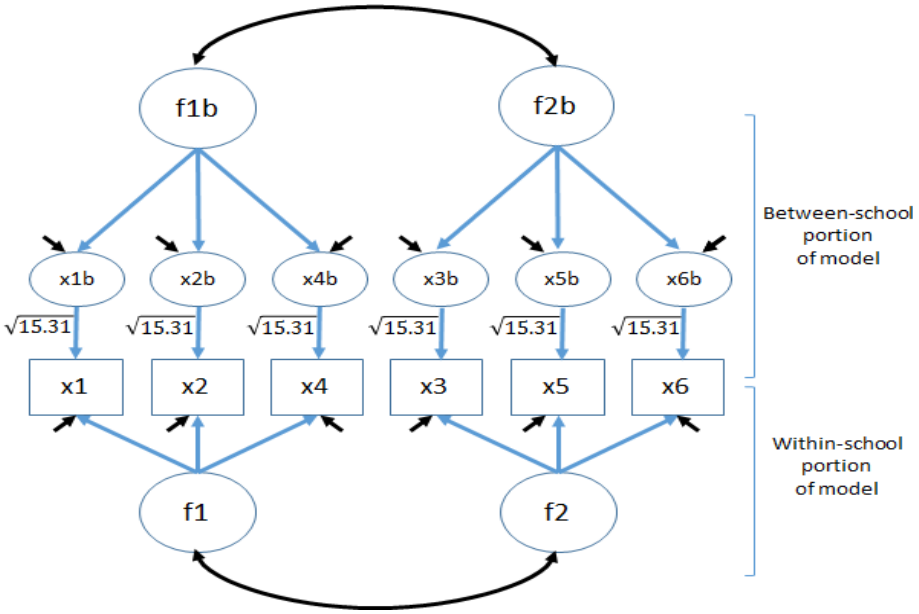

Figure 4: Multilevel factor model with two factors at levels one and two.

	Two Factors Within, One Factor Between			Two Factors Within, Two Factors Between			
	Within		Between	Within		Between	
Variable	F1	F2	F1	F1	F2	F1	F2
			lavaan				
x1	0.88		0.99	0.88		0.99	
x2	0.87		0.99	0.87		0.99	
x3		0.63	0.91		0.63		0.94
x4	0.45		0.87	0.45		0.87	
x5		0.82	0.90		0.82		0.94
x6		0.87	0.99		0.87		0.99
Cor(F1,F2)	0.63			0.65		0.88	
			Mplus				
x1	0.88		0.99	0.88		1.00	
x2	0.87		0.99	0.87		0.99	
x3		0.63	0.90		0.63		0.93
x4	0.45		0.91	0.45		0.91	
x5		0.82	0.90		0.82		0.94
x6		0.87	0.99		0.87		0.99
Cor(F1,F2)	0.63			0.65		0.88	

Table 2: Comparison of standardized factor loadings (n = 3,894) using **lavaan** and Mplus.

total covariance matrix will not reflect a scale's actual reliability not unless reliability is the same at each level (Geldhof, Preacher, and Zyphur 2014).

Based on Cronbach's (1951) equation 16, alpha is a function of the covariances (σ^2_{ij}), total variance (V_t), and the number of items in the scale (n) such that $\alpha = \frac{n^2\bar{\sigma}^2_{ij}}{V_t}$. The numerator is the product of the square of the number of items in the scale and the average of the unique covariance elements. The denominator is merely the sum of all the elements within the covariance matrix or summing together all the variances and two times the covariance elements. Extending alpha to a multilevel framework is straightforward and requires the use of the variance/covariance matrix estimated in a saturated model (i.e., step 4) or using the adjusted between level covariance matrix (i.e., `alpha(x$ab.cov)`).

Included as well in the syntax provided is an `alpha()` function which requires a covariance matrix as its input. To estimate multilevel alpha for the one factor model at level two, specify `alpha(x$ab.cov)` using the adjusted between group covariance matrix which results in an alpha of .97. In comparison, using the pooled within covariance matrix, `alpha(x$pw.cov)`, results in a level one alpha of .82 (note though the one factor model at level one did not fit well, this is shown for comparative purposes). Generally, higher level scales are often more reliable as a result of coming from multiple raters (Byrne 2012). Other multilevel reliability measures are available such as multilevel composite reliability ω (see Geldhof *et al.* 2014, for a comparison of features).

3. Conclusion

Although the importance of performing MCFA with clustered data has been discussed (Julian 2001; Muthen and Satorra 1995; Schweig 2013), the steps on how to perform the analyses have not been illustrated using R together with the **lavaan** package. We provide a function (i.e., `mcfa.input`) wherein all the necessary covariance matrices, the scaling factor, the sample size at both levels, and the ICCs are automatically computed from the raw data.

The manual modeling using the multigroup setup is unconventional though is required to properly estimate the hypothesized level 2 model. In addition, several instances require users to override the default options in **lavaan**. In this paper, we illustrate step-by-step how to conduct the analyses using the MCFA procedures outlined by Hox (2002) and others (Stapleton 2006) but within the R environment. Finally, we also show how to compute multilevel alpha as an estimate of scale reliability at the group level.

The assumption that the factor structures using nested data cannot be assumed: at times the factor structures may be the same at both levels (Konold *et al.* 2014), higher level factor structures may be simpler (Huang and Cornell 2015b), or the higher level factor structures may be totally different (Schweig 2013). In any case, using a MCFA, especially when the higher level factor structures are of interest goes beyond properly estimating standard errors or adjusting model fit indices but requires investigating the similarities or differences in factor structures at both levels simultaneously.

References

Beaujean AA (2014). *Latent Variable Modeling Using R: A Step-by-step Guide*. Routledge.

Bliese P (2000). "Within-group Agreement, Non-independence, and Reliability: Implications for Data Aggregation and Analysis." In K Bollen, J Long (eds.), *Multilevel theory, research, and methods in organizations: Foundations, extensions and new directions*, pp. 349–381. Jossey-Bass, San Francisco, CA.

Byrne B (2012). *Structural Equation Modeling with Mplus: Basic Concepts, Applications, and Programming*. Routledge, New York, NY.

Chan D (1998). "Functional Relations Among Constructs in the Same Content Domain at Different Levels of Analysis: A Typology of Composition Models." *Journal of Applied Psychology*, **83**(2), 234–246. doi:10.1037/0021-9010.83.2.234.

Chen G, Bliese PD, Mathieu JE (2005). "Conceptual Framework and Statistical Procedures for Delineating and Testing Multilevel Theories of Homology." *Organizational Research Methods*, **8**(4), 375–409. doi:10.1177/1094428105280056.

Cohen J, Cohen P, West SG, Aiken LS (2003). *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences*. Lawrence Erlbaum Associates, Mahwah, NJ. ISBN 978-1-134-80094-0.

Cronbach LJ (1951). "Coefficient Alpha and the Internal Structure of Tests." *Psychometrika*, **16**(3), 297–334. doi:10.1007/BF02310555.

Dedrick RF, Greenbaum PE (2011). "Multilevel Confirmatory Factor Analysis of a Scale Measuring Interagency Collaboration of Children's Mental Health Agencies." *Journal of Emotional and Behavioral Disorders*, **19**(1), 27–40. doi:10.1177/1063426610365879.

D'Haenens E, Van Damme J, Onghena P (2010). "Multilevel Exploratory Factor Analysis: Illustrating Its Surplus Value in Educational Effectiveness Research." *School Effectiveness and School Improvement*, **21**(2), 209–235. doi:10.1080/09243450903581218.

Dyer NG, Hanges PJ, Hall RJ (2005). "Applying Multilevel Confirmatory Factor Analysis Techniques to the Study of Leadership." *The Leadership Quarterly*, **16**(1), 149–167. ISSN 10489843. doi:10.1016/j.leaqua.2004.09.009.

Finch WH, French BF (2015). *Latent Variable Modeling with R*. Routledge, New York. ISBN 978-0-415-83245-8.

Geldhof J, Preacher KJ, Zyphur MJ (2014). "Reliability Estimation in a Multilevel Confirmatory Factor Analysis Framework." *Psychological Methods*, **19**(1), 72–91. ISSN 1939-1463(Electronic);1082-989X(Print). doi:10.1037/a0032138.

Griffith J (1997). "Student and Parent Perceptions of School Social Environment: Are They Group Based?" *The Elementary School Journal*, **98**(2), 135–150. ISSN 0013-5984, 1554-8279. doi:10.1086/461888.

Heck RH (2001). "Multilevel Modeling with SEM." *New Developments and Techniques in Structural Equation Modeling*, pp. 89–127.

Heck RH, Thomas SL (2008). *An Introduction to Multilevel Modeling Techniques*. 2nd edition. Routledge, New York. ISBN 978-1-84169-756-7.

Hox JJ (2002). *Multilevel Analysis: Techniques and Applications*. Lawrence Erlbaum, Mahwah, NJ.

Huang F (2016). "Alternatives to Multilevel Modeling for the Analysis of Clustered Data." *Journal of Experimental Education*, **84**, 175–196. doi:10.1080/00220973.2014.952397.

Huang F, Cornell D, Konold T, Meyer P, Lacey A, Nekvasil E, Heilbrun A, Shukla K (2015). "Multilevel Factor Structure and Concurrent Validity of the Teacher Version of the Authoritative School Climate survey." *Journal of School Health*, **85**, 843–851. doi:10.1037/spq0000062.

Huang FL, Cornell DG (2015a). "Factor Structure of the High School Teacher Version of the Authoritative School Climate Survey." *Journal of Psychoeducational Assessment*. ISSN 0734-2829, 1557-5144. doi:10.1177/0734282915621439.

Huang FL, Cornell DG (2015b). "Using Multilevel Factor Analysis with Clustered Data: Investigating the Factor Structure of the Positive Values Scale." *Journal of Psychoeducational Assessment*, p. 0734282915570278. doi:10.1177/0734282915570278.

Huang FL, Cornell DG, Konold TR (2014). "Aggressive Attitudes in Middle Schools: A Factor Structure and Criterion Related Validity Study." *Assessment*. doi:10.1177/1073191114551016.

Julian MW (2001). "The Consequences of Ignoring Multilevel Data Structures in Nonhierarchical Covariance Modeling." *Structural Equation Modeling: A Multidisciplinary Journal*, **8**(3), 325–352. doi:10.1207/S15328007SEM0803_1.

Kaplan D, Elliott PR (1997). "A Didactic Example of Multilevel Structural Equation Modeling Applicable to the Study of Organizations." *Structural Equation Modeling: A Multidisciplinary Journal*, **4**(1), 1–24. doi:10.1080/10705519709540056.

Konold T, Cornell D, Huang F, Meyer P, Lacey A, Nekvasil E, Heilbrun A, Shukla K (2014). "Multilevel Multi-informant Structure of the Authoritative School Climate Survey." *School Psychology Quarterly*, **29**, 238–255. doi:10.1037/spq0000062.

Kozlowski SW, Klein KJ (2000). "A Multilevel Approach to Theory and Research in Organizations: Contextual, Temporal, and Emergent Processes." URL http://psycnet.apa.org/psycinfo/2000-16936-001.

Muthen BO (1994). "Multilevel Covariance Structure Analysis." *Sociological Methods & Research*, **22**(3), 376–398. doi:10.1177/0049124194022003006.

Muthen BO, Satorra A (1995). "Complex Sample Data in Structural Equation Modeling." *Sociological Methodology*, **25**, 267–316. doi:10.2307/271070.

R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

Ramelow D, Currie D, Felder-Puig R (2015). "The Assessment of School Climate: Review and Appraisal of Published Student-report Measures." *Journal of Psychoeducational Assessment*, **33**(8), 731–743. doi:10.1177/0734282915584852.

Raudenbush S, Bryk A (2002). *Hierarchical Linear Models: Applications and Data Analysis Methods*. 2nd edition. Sage, Thousand Oaks, CA.

Rosseel Y (2012). "**lavaan**: An R Package for Structural Equation Modeling." *Journal of Statistical Software*, **48**(2), 1–36. URL https://www.jstatsoft.org/htaccess.php?volume=48&type=i&issue=02&paper=true.

Rosseel Y (2016). "The **lavaan** Tutorial." *Technical report*, Ghent University, Belgium. URL http://lavaan.ugent.be/tutorial/tutorial.pdf.

Roux A (2002). "A Glossary for Multilevel Analysis." *Journal of Epidemiology and Community Health*, **56**(8), 588–594. doi:10.1136/jech.56.8.588.

Roux A (2004). "The Study of Group-level Factors in Epidemiology: Rethinking Variables, Study Designs, and Analytical Approaches." *Epidemiologic Reviews*, **26**(1), 104–111. doi:10.1093/epirev/mxh006.

Schweig J (2013). "Cross-Level Measurement Invariance in School and Classroom Environment Surveys: Implications for Policy and Practice." *Educational Evaluation and Policy Analysis*, **36**, 259–280. doi:10.3102/0162373713509880.

Selig JP, Card NA, Little TD (2008). "Latent Variable Structural Equation Modeling in Cross-cultural Research: Multigroup and Multilevel Approaches." In FJR van, DA van, YH Poortinga (eds.), *Multilevel Analysis of Individuals and Cultures*, pp. 93–119. Taylor & Francis Group/Lawrence Erlbaum Associates, New York, NY.

Stapleton L (2006). "Using Multilevel Structural Equation Modeling Techniques with Complex Sample Data." In G Hancock, R Mueller (eds.), *Structural Equation Modeling: A Second Course*, pp. 345–383. Information Age Publishing, Greenwich, CT.

Streiner DL (2003). "Starting at the Beginning: An Introduction to Coefficient Alpha and Internal Consistency." *Journal of Personality Assessment*, **80**(1), 99–103.

van Horn ML (2003). "Assessing the Unit of Measurement for School Climate Through Psychometric and Outcome Analyses of the School Climate Survey." *Educational and Psychological Measurement*, **63**(6), 1002–1019. doi:10.1177/0013164403251317.

Zyphur MJ, Kaplan SA, Christian MS (2008). "Assumptions of Cross-level Measurement and Structural Invariance in the Analysis of Multilevel data: Problems and Solutions." *Group Dynamics: Theory, Research, and Practice*, **12**(2), 127–140. doi:10.1037/1089-2699.12.2.127.

Affiliation:

Francis L. Huang, Ph.D.
Educational, School, and Counseling Psychology
University of Missouri
16 Hill Hall, Columbia, MO 65211
E-mail: huangf@missouri.edu

URL: http://faculty.missouri.edu/huangf/
Date: June 11, 2017 (updated)