Visualizing and Interpreting Multi-Group Confirmatory Factor Analysis

Erin M. Buchanan[1]

[1] Harrisburg University of Science and Technology

Author Note

Thank you to K.D. Valentine and Chelsea Parlett-Pelleriti for feedback on some ugly graphs.

The authors made the following contributions. Erin M. Buchanan: Conceptualization, Writing - Original Draft Preparation, Writing - Review & Editing.

Correspondence concerning this article should be addressed to Erin M. Buchanan, 326 Market St., Harrisburg, PA, USA. E-mail: ebuchanan@harrisburgu.edu

Abstract

Latent variable modeling as a lens for psychometric theory is a popular tool for social scientists to examine measurement of constructs (Beaujean, 2014). Journals such as *Assessment* regularly publish articles supporting new or previously established measures of latent constructs (e.g., depression, anxiety) wherein a measurement model is established for the scale in question. These measurement models designate the relationship between the measured, observed variables, and the underlying construct, with the assumption that these relations hold in many samples. Confirmatory factor analysis can be used to investigate the replicability and generalizability of the measurement model in new samples, while multi-group confirmatory factor analysis is used to examine the measurement model across groups within samples (Brown, 2015). With the rise of the replication crisis and "psychology's renaissance" (Nelson, Simmons, & Simonsohn, 2018), interest in divergence in measurement has increased, often focused on small parameter differences within the latent model. While the statistical procedure for examining measurement invariance is moderately well established, it is clear that the toolkit for inspecting these results is lacking. This manuscript will outline ways to visualize potential non-invariance, to supplement large numbers of tables that often overwhelm a reader within these published reports. Further, given these visualizations, readers will learn how to interpret the impact and size of the proposed non-invariance in models. While it is tempting to suggest that problems with replication and generalizability are simply issues with measurement, it is crucial to remember that all models have variability and error, even those models estimating the differences between item functioning, such as multi-group confirmatory factor analysis. This manuscript will help provide a framework for researchers interested in registered reports in this area.

Keywords: multigroup confirmatory factor analysis, measurement invariance, visualization, effect size

Visualizing and Interpreting Multi-Group Confirmatory Factor Analysis

Psychological assessments play a critical role in our ability to measure and analyze constructs to support theories and experimental hypotheses. Defining and creating assessments to validly and reliability measure constructs is often difficult because phenomenon, such as anxiety, are often not directly observable. Instead, we use surveys and questionnaires to indirectly assess the underlying construct (DeVellis & Thorpe, 2022). Latent variable modeling (i.e., structural equation modeling) is a popular tool for the validation of developed survey instruments to verify scale dimensionalality, structure, and model fit. A simple search for scale development reveals thousands of articles in psychology that examine new and previously published work, thus, illustrating the interest in both measurement and the use of validation techniques. Unfortunately, except in specialty journals, much of the validity evidence and/or development for measures used in empirical studies is not reported within the journal article Weidman, Steckler, & Tracy (2017). Without this information, it is difficult to interpret individual study conclusions, as validity information allows for judgment of usefulness of the measured values (Flake & Fried, 2020). Further, the current focus on replication Zwaan, Etz, Lucas, & Donnellan (2018), reproducibility (Nelson et al., 2018), and the credibility of our results (Vazire, Schiavone, & Bottesini, 2022) has demonstrated questionable measurement practices - decisions that researchers make like survey selection and scoring that impact the results of the study (Flake & Fried, 2020). Transparent reporting of the use and creation of scales can improve both interpretation and reproducibility when using surveys developed to measure latent constructs (Shadish, Cook, & Campbell, 2001).

A secondary concern for developed measures is the potential for differential responding and assessment within target populations. For example, Trent et al. (2013) examined for potential variability in the Revised Child Anxiety and Depression Scale in White and Black youths (Chorpita, Yim, Moffitt, Umemoto, & Francis, 2000). They found that the scale mostly functioned the same for both White and Black individuals but differences in averages on individual items could potentially affect the scoring and interpretation of the scale results. This comparison of sub-populations is the test of measurement invariance (Meredith, 1993). Invariance or equivalence implies that the scale operates in the same fashion for each sub-group, and thus, differences in the final latent variable scores can interpretated as differences in populations. Non-invariance suggests that individuals respond or interpret items differently, and thus, differences in scores may represent different scores on the latent variable in the population or differences in measurement. Non-invariant measurement may lead to misleading results when making group comparisons, and assessing invariance has become a popular technique in scale development (Van De Schoot, Schmidt, De Beuckelaer, Lek, & Zondervan-Zwijnenburg, 2015).

Measurement invariance is typically analyzed using confirmatory factor analysis, specifically, multi-group confirmatory factor analysis (MGCFA) or less often, with item response theory Tay, Meade, & Cao (2015). First, the model is examined with the factor structure proposed for the latent and observed variables, and then often these models are assessed for each group separately. The two models are then combined together into one nested CFA in order to determine configural invariance Kline (2016). Configural invariance tests if the proposed factor structure is the same between groups. In this model, all other estimated parameters are allowed to vary between groups. The general approach is to use this model as a baseline for starting a sequential analysis of further restrictions between group parameters (i.e., more restrictive with each step). However, models without configural invariance can occur and often point to misspecification for the observed and latent variables within one group (i.e., cross loadings of items onto other latent variables or correlated error terms for one group only).

Next, the estimated parameter between each observed variable and its latent variable are constrained to be equal between groups for metric invariance. For example, item 1's factor loading must be equal to item 1's factor loading for each group. This test examines if the items represent the same relationship to the latent variable, or if specific items have weaker or stronger relationships in specific groups. Finding non-invariance at this stage generally points to items that have different functioning or interpretation for one group. At the third model, the item intercepts (i.e., item averages) are restricted across groups for scalar invariance. Scalar non-invariance would indicate that items have the same strength of relationship with their latent variable, just one group has a higher overall average on that item. Last (although sometimes not used), we may consider constraining error variances for each observed variable to be equal across groups for strict invariance. Strict non-invariance can occur when one group has a higher range of values on the observed variable, thus showing a larger variance. For example, if using a Likert scale, one group may use the full 1 to 7 range (creating a flatter distribution and larger variance), while the other group shows a ceiling effect of only using 5 to 7.

These concepts have been explored and implemented for the last fifty years Sörbom (1978) and implemented in the most popular structural equation modeling programs Boker et al. (2011). Byrne, Shavelson, and Muthén (1989) extended the ideas of multi-group testing by suggesting partial invariance (followed by Meredith, 1993). Partial invariance occurs when non-invariance is found but can be attributed to only a few parameter estimate differences between groups (i.e., items 1 and 2 have different factor loadings but all others are the same). This testing provided an advantage to understand where the potential non-invariance may occur for further study and interpretation guidelines. To determine when non-invariance and partial invariance occurred, each model is sequentially compared to the previous model using some form of a difference test. Traditionally, since models were nested, a chi-square difference test was used Cheung & Rensvold (2002); however, given the known issues with chi-square (Thompson & Daniel, 1996), people have favored empirical cutoffs for differences in fit indices. As the field pushes back against favoring cutoff criteria and rules of thumb Putnick & Bornstein (2016), an effect size measure for translating “how much” non-invariance was developed d_{MACS} (Nye & Drasgow, 2011). This effect size examines the differences in observed variables between the two groups for both the factor loading and the item intercept; thus, any differences in either or both will increase the effect size for non-invariance (Stark et al., 2006).

With d_{MACS} and measurement invariance testing, researchers can begin to quantify how and where their construct measurement may vary between groups. However, given the large number of studies that show non-invariance, it is clear that equivalence can be hard to meet. It is difficult to know if non-invariance occurs because of random sampling error, true population differences, or differences in replication and reproducibility of the construct in a new sample. Further, it is important to remember that the parameter estimates that we are testing are just that - estimates. All the parameter estimates have measures of standard error to indicate that they are more than likely variable with a new sample or population. Given that this information is generally ignored during the examination of measurement invariance, it may be that we are claiming that many scales are non-invariant, when in reality, the differences between loadings or item intercepts are small and unimportant. d_{MACS} provides the opportunity to begin to think about the smallest effect size of interest or the smallest meaningful effect size Anvari & Lakens (2021). As mentioned, d_{MACS} has only really been explored for a combined intercept and loadings, and while useful, does not necessarily allow a researcher to pinpoint specific issues within an observed variable. The purpose of this manuscript is provide readers with a framework for visualization of differences in loadings, intercepts, and variances for each item, and the impact of those differences on the distribution of the latent mean. No known visualization techniques have been proposed for measurement invariance. By creating panel visualizations, we can supplement a researchers ability to judge the strength of the non-invariance differences and effect size for each item. Coupled with other indicators (i.e., fit indice differences, d_{MACS}), we can move toward a better understanding of how much measurement non-invariance is meaningful.

By the end of this manuscript, readers will:

1. Be able to create visualizations for common steps to multi-group confirmatory factor analysis.
2. Be able to interpret the impact and size of potential non-invariance on measurement.
3. Understand the impact of measurement variability on replication and generalizability.

Method

Design and Analysis

Data was simulated using the `simulateData` function in the *R* package `lavaan` (Rosseel, 2012) assuming multivariate normality using a μ of 0 and σ of 1 for the data. This function allows you to write `lavaan` syntax for your model with estimated values to generate data for observed variables. The data included two groups of individuals ("Group 1", "Group 2") for a multi-group confirmatory factor analysis (n_{group} = 250, N = 500). The latent variables were assumed to be continuous normal. The model consisted of five observed items predicted by one latent variable (`lv =~ q1 + q2 + q3 + q4 + q5`); however, the demonstration in this manuscript extends to multiple latent variables and other combinations of observed variables. Each item was assumed to be related to the latent variable with loadings approximately equal to .40 to .80, except when cases of non-invariance on the loadings was assumed.

The Brown (2015) steps of testing measurement invariance are demonstrated in this manuscript for illustration purposes, but in line with Stark et al. (2006) suggestions, the visualizations show the impact of loadings and intercepts together. The configural model was analyzed nesting both groups into the same CFA model requiring that both groups show the same model structure, but all other parameters are free to vary between groups. The metric model constrained the factor loadings of each group to be equal within the model. The scalar model then constrained the item intercepts (i.e., item mean) to be equal across groups. Finally, the strict model constrained the item variances (i.e., error variances) to be equal for each item across groups. These models are normally tested sequentially, and a convenience function `mgcfa` is provided in the supplemental documents for this manuscript. Fit indices for the steps for multi-group models are presented in the appendix for comparison of cutoff rules of thumb (Cheung & Rensvold, 2002) to effect sizes and visualizations presented in this manuscript. Fit indices include Akaike Information Criterion (AIC, Akaike, 1998), Bayesian Information Criterion (BIC, Schwarz, 1978), Comparative Fit Index (CFI, Bentler, 1990), Tucker Lewis Index (TLI, Tucker & Lewis, 1973), root mean squared error of approximation RMSEA (Steiger, 1990), and standardized root mean squar residual (SRMR, Bentler, 1995).

The data was then simulated to represent invariance across all model steps, small, medium, and large invariance using d_{MACS} estimated sizes from Nye, Bradburn, Olenick, Bialko, and Drasgow (2019). While d_{MACS} is used primarily for an effect size of the (non)-invariance for intercepts and loadings together, a similar approach was taken for the estimation of small, medium, and large effects on the residuals. The effect size is presented for all models, calculated from the *dmacs* package Nye & Drasgow (2011). Only one item in each model was manipulated from the invariant model to create the non-invariant models.

Results

Code Examples

The complete code for this manuscript can be found at https://osf.io/wev5f/, and the function code for the convenience function for multi-group models and plots is found in the appendix. First, we would create our model code in `lavaan` syntax. The `lv` latent variable predicts the five measured variables, which are present as columns in our `df.invariant` dataset. You would include the dataframe in the `data` argument of our function, the name of the grouping variable in quotes for `group`, and the `lavaan` model syntax in the `model` argument. The `mgcfa` function code runs an overall model with all data, regardless of group, each group separately on the model, then the steps described above: configural, metric, scalar, and strict invariance.

```
# create lavaan model
model.overall <- "lv =~ q1 + q2 + q3 + q4 + q5
q1 ~ 0*1
lv ~ 1"
# look at the data
head(df.invariant)
```

```
##           q1          q2          q3         q4         q5   group
## 1 -0.8903542 -0.81707530  0.06137292 -1.3236407 -1.7916418 Group 1
## 2  1.1054521 -0.03540948 -0.81299606  1.0028340 -0.1909127 Group 1
## 3  1.4555852  1.54083484  1.59084213 -0.3345967 -0.6865496 Group 1
## 4 -1.8745187 -1.27880245 -2.53565792 -1.0024193 -1.6253249 Group 1
## 5 -0.4449517 -0.17782974  1.05507079 -1.2615705  1.7536428 Group 1
## 6  0.2278813  0.71348845  1.63251893  0.6449847 -1.0055700 Group 1
```

```
# run our mgcfa function to run all models
results.invariant <- mgcfa(data = df.invariant, #dataframe
                           group = "group",
                           model = model.overall)
# what is saved for you
names(results.invariant)
```

```
## [1] "model_coef"       "model_fit"        "model.overall"    "model.group1"
## [5] "model.group2"     "model.configural" "model.metric"     "model.scalar"
## [9] "model.strict"
```

The results are saved as a list and include the following:

1) `model_coef`: a tidy dataframe with *all* model's coefficients saved from the `lavaan` outputs.

```
head(results.invariant$model_coef)
```

```
## # A tibble: 6 x 13
##   term       op    estimate std.error statistic   p.value std.lv std.all std.nox
##   <chr>      <chr>    <dbl>     <dbl>     <dbl>     <dbl>  <dbl>   <dbl>   <dbl>
## 1 "lv =~ q1" =~       1         0           NA    NA         0.780   0.598   0.598
## 2 "lv =~ q2" =~       0.564     0.0864       6.52  6.99e-11  0.440   0.435   0.435
## 3 "lv =~ q3" =~       0.748     0.105        7.12  1.09e-12  0.583   0.505   0.505
## 4 "lv =~ q4" =~       0.338     0.0804       4.20  2.62e- 5  0.264   0.250   0.250
## 5 "lv =~ q5" =~       0.904     0.120        7.52  5.48e-14  0.705   0.613   0.613
## 6 "q1 ~1 "   ~1       0         0           NA    NA         0       0       0
## # i 4 more variables: model <chr>, block <int>, group <int>, label <chr>
```

2) `model_fit`: a tidy dataframe with *all* model's fit indices saved from the `lavaan` outputs.

```
head(results.invariant$model_fit)
```

```
## # A tibble: 6 x 18
##    agfi   AIC   BIC   cfi chisq  npar  rmsea rmsea.conf.high   srmr   tli
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>           <dbl>  <dbl> <dbl>
## 1 0.979 7516. 7579. 0.994  6.37    15 0.0234          0.0697 0.0211 0.988
## 2 0.948 3766. 3819. 0.976  7.79    15 0.0473          0.108  0.0312 0.953
## 3 0.952 3762. 3815. 0.980  7.25    15 0.0424          0.104  0.0322 0.960
## 4 0.950 7528. 7654. 0.978 15.0     30 0.0449          0.0886 0.0317 0.956
## 5 0.942 7529. 7639. 0.954 24.7     26 0.0554          0.0905 0.0476 0.934
## 6 0.952 7523. 7616. 0.964 26.2     22 0.0428          0.0760 0.0488 0.960
```

```
## # i 8 more variables: converged <lgl>, estimator <chr>, ngroups <int>,
## #   missing_method <chr>, nobs <int>, norig <int>, nexcluded <int>, model <chr>
```

3) Saved `lavaan` fitted objects that you can use the `summary()`, `parameterEstimates()`, `fitIndices()`, etc. on. Overall model indicates the model without grouping variables testing all data on the proposed model structure. This model is then tested separately for each group (`model.group1`, `model.group2`). The final models follow the Brown (2015) naming convention for sequential steps for testing MGCFA for measurement invariance (`model.configural`, `model.metric`, `model.scalar`, `model.strict`).

The results from the `model_coef` table can then be used directly in the suggested plotting function. The plot outputs will be described below. First, here are the arguments for the function:

1) `data_coef`: A tidy dataframe of the parameter estimates from the models. This function assumes you have used `broom::tidy()` on the saved model from `lavaan` and added a column called "model" with the name of the model step. This function will only run for models that have used the grouping function (i.e., configural, metric, scalar, and strict or other combinations/steps you wish to examine).

2) `model_step`: Which model do you want to plot? You should match this name to the one you want to extract from your model column in the `data_coef`.

3) `item_name`: Which observed variable from your model syntax do you want to plot? Please list this variable name exactly how it appears in the model.

4) `x_limits`: What do you want the x-axits limits to be for your invariance plot? The default option is to assume the latent variable is standardized, and therefore, -1 to 1 is recommended. Use only two numbers, a lower and upper limit.

5) `y_limits`: What do you want the y-axits limits to be for your invariance plot? Given that the latent variable is used to predict the observed values in the data, you could use the minimum and maximum values found in the data. If that range is large, consider reducing this value to be able to visualize the results (i.e., otherwise it may be too zoomed out to judge group differences). Use only two numbers, a lower and upper limit.

6) `ci_level`: What confidence limit do you want to plot? Use 1 - α.

7) `model_results`: In this argument, include the saved `lavaan` output for the model listed in the `model_step` argument.

8) `lv_name`: Include the name of the latent variable, exactly how it is listed in your `lavaan` syntax. You should plot the latent variable that the `item_name` is linked to. If you have items that load onto multiple latent variables, you will need to make multiple plots.

```
plot_mgcfa(
  data_coef = results.invariant$model_coef, # output from model_coef
  model_step = "configural", # which model do you want to plot
  item_name = "q4", # name of observed item
  x_limits = c(-1,1), # latent variable limits to graph
  y_limits = c(min(df.invariant$q4), max(df.invariant$q4)), # Y min and max in data
  ci_level = .95, # what ci do you want
  model_results = results.invariant$model.configural, # what model results do you want
  lv_name = "lv" # which latent variable do you want
)
```

Visualization of Invariance

The output from this model can be found in Figure 1. On the left hand side, the item invariance is plotted, and on the right hand side, the latent mean distributions for the two groups are plotted. In the item invariance sub-plot, the visualization includes all three components traditionally seen in MGCFA testing steps: loadings, intercepts, and residuals. Each visualization element was designed to match the traditional visualization for that type of output. All parameter estimates are plotted on the unstandardized estimates and their confidence interval based on the standard error of the estimate.

Loadings. Factor loadings represent the slope of the regression equation for the latent variable predicting the scores on the observed variable ($\hat{Y} \sim b_0 + b_1X + \epsilon$). Therefore, the latent variable is shown on the x-axis using standardized values (i.e., z-scores) where -1 indicates one standard deviation below the mean for the latent variable, 0 indicates the mean for the latent variable and so on. The y-axis indicates the observed variable scores, and here, the plot includes the entire range of the scale of the data for item four. The coefficient (b_1) for group 1 was 0.40, while the coefficient for group 2 was 0.34. The ribbon bands around the plotted slopes indicate the confidence interval for that estimate. In this plot, while the coefficients for each group are not literally equal, the overlapping and parallel slope bands indicate they are not different practically.

Intercepts. The item intercepts (b_0) are plotted on the middle line where they would cross the y-axis at a latent variable score of zero. These are represented by a dot with a set of confidence error bars around the point. The intercept for group 1 was 0.07, while the coefficient for group 2 was 0.03. In this invariant depiction, the overlap in the intercepts is clear, indicating they are not different. You can use `y_limits` to zoom in on the graph if these are too small to be distinguishable.

Residuals. Residuals are trickier to plot, as they are the left over error when predicting the observed variables ϵ. It is tempting to plot this value as the confidence band around the slope, however, that defeats the purpose of understanding that the slopes are estimated separately from the residuals, and both have an associated variability around their parameter estimate. Therefore, residuals are represented in the inset picture at the bottom right of the item invariance plot. The black bars represent the estimated residual for each group (group 1: 0.91, group 2: 1.16). The distributions are plotted to represent the normal spread of values using the standard error of the residuals. The violin plot allows for direct comparison of those residuals and their potential distributions. Note that the placement has nothing to do with the x or y-axis and is designed to always show in the same location, regardless of size/value.

Latent Means. The overall impact of differences on the latent means can be found in the right hand visualization. The latent means are calculated by using the `predict` function and then plotted as overlapping histograms. The vertical colored lines represent the mean for each group, and the spread of the distribution can be examined using the density coloring. Finally, group labels are represented in the figure caption on the bottom. Group 1 is usually the group that is alphabetically first in the dataset or whichever group is the first that appears when using the `levels()` command.

Graphing Effect Size. The d_{MACS} value for item 4 in the invariant model was 0.06, representing a nil or unimportant difference in this manuscript. It is important to note that while Nye et al. (2019) suggests specific sizes for small, medium, and large, each researcher should determine for themselves what effects represent. Figure 2 displays the results from the small ($d_{MACS} = 0.12$) difference in loadings, while Figure 3 displays the results from the medium ($d_{MACS} = 0.43$) difference in loadings, and Figure 4 shows the large ($d_{MACS} = 0.63$) differences. When investigating the slope values, we can clearly see the change in the loading for the second group (the only manipulated variable, although random dataset generation may also change intercepts and residuals slightly). At the medium effect size, we see that the confidence bands do not overlap (at the edges), and at the large effect size, we can see a clear separation of two lines. Note that the intercepts are still fixed so the loading representation will not literally separate, but the steepness of the lines is the indicator of the difference between the slopes. You can imagine these lines are interpreted like a simple slopes analysis for interactions in regression (Cohen, Cohen, West, & Aiken, 2003). When simple slopes for interactions are plotted, if they are parallel, there is no interaction, and if they cross, then there is an interaction. Here, we can use this same logic. If they are parallel, there is likely invariance (they are the same), and the further from parallel they become, the larger the effect size for the differences between group loadings.

For intercepts, the small (Figure 5), medium (Figure 6), and large (Figure 7) depictions represent d_{MACS} values of 0.29, 0.52, and 0.76, respectively. Intercept differences can be clearly seen represented by the spacing out of the intercept locations (and thus, the overall line as well). Note how little the intercept differences appear to influence the latent variable means and distributions.

Last, the effect of the residuals is plotted in small (Figure 8), medium (Figure 9), and large (Figure 10) formats. While d_{MACS} values are not technically avaliable for the residuals, our models showed 0.20, 0.14, and 0.11, respectively. These differences in values are variable due to the random generation of datasets for each measurement invariance manipulation. At first glance, the differences in the small chart may seem large, because the black lines are not touching, but notice that the distributions overlap, indicating a likely small difference. The medium and large differences better illustrate differences in residuals across groups. Further, the impact of the residuals on the shape of the latent mean distribution can also been seen (and unintentionally, in the first figures as well due to random variation). The impact is due to the standard error of the residuals, as smaller standard errors represent lepokurtic distributions (taller), and larger standard errors represent platykurtic distributions (flatter). The effect size difference of the residuals does not appear to change the effects in the latent means.

Discussion

Conclusions:

- framework for submitted/interpreting reports

References

Akaike, H. (1998). *Information theory and an extension of the maximum likelihood principle* (E. Parzen, K. Tanabe, & G. Kitagawa, Eds.). New York, NY: Springer New York. Retrieved from http://link.springer.com/10.1007/978-1-4612-1694-0_15

Anvari, F., & Lakens, D. (2021). Using anchor-based methods to determine the smallest effect size of interest. *Journal of Experimental Social Psychology*, *96*, 104159. https://doi.org/10.1016/j.jesp.2021.104159

Barry, A. E., Chaney, B., Piazza-Gardner, A. K., & Chavarria, E. A. (2014). Validity and Reliability Reporting Practices in the Field of Health Education and Behavior: A Review of Seven Journals. *Health Education & Behavior*, *41*(1), 12–18. https://doi.org/10.1177/1090198113483139

Beaujean, A. A. (2014). *Latent variable modeling using r: A step by step guide*. New York: Routledge/Taylor & Francis Group.

Bentler, P. M. (1990). Comparative fit indexes in structural models. *Psychological Bulletin*, *107*(2), 238–246. https://doi.org/10.1037/0033-2909.107.2.238

Bentler, P. M. (1995). *EQS structural equations program manual*. Encino, CA.

Boker, S., Neale, M., Maes, H., Wilde, M., Spiegel, M., Brick, T., … Fox, J. (2011). OpenMx: An Open Source Extended Structural Equation Modeling Framework. *Psychometrika*, *76*(2), 306–317. https://doi.org/10.1007/s11336-010-9200-6

Brown, T. A. (2015). *Confirmatory factor analysis for applied research* (Second edition). New York ; London: The Guilford Press.

Byrne, B. M. (2001). Structural Equation Modeling With AMOS, EQS, and LISREL: Comparative Approaches to Testing for the Factorial Validity of a Measuring Instrument. *International Journal of Testing*, *1*(1), 55–86. https://doi.org/10.1207/S15327574IJT0101_4

Byrne, B. M., Shavelson, R. J., & Muthén, B. (1989). Testing for the equivalence of factor covariance and mean structures: The issue of partial measurement invariance.

Psychological Bulletin, *105*(3), 456–466. https://doi.org/10.1037/0033-2909.105.3.456

Cheung, G. W., & Rensvold, R. B. (2002). Evaluating Goodness-of-Fit Indexes for Testing Measurement Invariance. *Structural Equation Modeling: A Multidisciplinary Journal*, *9*(2), 233–255. https://doi.org/10.1207/s15328007sem0902_5

Chorpita, B. F., Yim, L., Moffitt, C., Umemoto, L. A., & Francis, S. E. (2000). Assessment of symptoms of DSM-IV anxiety and depression in children: a revised child anxiety and depression scale. *Behaviour Research and Therapy*, *38*(8), 835–855. https://doi.org/10.1016/S0005-7967(99)00130-8

Cohen, J., Cohen, P., West, S. G., & Aiken, L. (2003). *Applied multiple regression / correlation analysis for the behavioral sciences* (3rd ed.). Lawrence Erlbaum Associates.

DeVellis, R. F., & Thorpe, C. T. (2022). *Scale development: Theory and applications* (Fifth edition). Thousand Oaks, California: SAGE Publications, Inc.

Dueber, D. (2023). *Dmacs*. Retrieved from https://github.com/ddueber/dmacs

Flake, J. K., & Fried, E. I. (2020). Measurement Schmeasurement: Questionable Measurement Practices and How to Avoid Them. *Advances in Methods and Practices in Psychological Science*, *3*(4), 456–465. https://doi.org/10.1177/2515245920952393

Jöreskog, K. G. (1971). Simultaneous factor analysis in several populations. *Psychometrika*, *36*(4), 409–426. https://doi.org/10.1007/BF02291366

Jöreskog, Karl G., & Sörbom, D. (2001). *LISREL 8: user's reference guide* (2. ed., updated to LISREL 8). Lincolnwood, Ill: SSI Scientific Software Internat.

Kline, R. B. (2016). *Principles and practice of structural equation modeling* (Fourth edition). New York: The Guilford Press.

Lakens, D. (2017). Equivalence Tests. *Social Psychological and Personality Science*, *8*(4), 355–362. https://doi.org/10.1177/1948550617697177

Makel, M. C., & Plucker, J. A. (2014). Facts Are More Important Than Novelty: Replication in the Education Sciences. *Educational Researcher*, *43*(6), 304–316. https://doi.org/10.3102/0013189X14545513

Makel, M. C., Plucker, J. A., & Hegarty, B. (2012). Replications in Psychology Research: How Often Do They Really Occur? *Perspectives on Psychological Science*, *7*(6), 537–542. https://doi.org/10.1177/1745691612460688

Marsh, H. W., Hau, K.-T., & Wen, Z. (2004). In search of golden rules: Comment on hypothesis-testing approaches to setting cutoff values for fit indexes and dangers in overgeneralizing hu and bentler's (1999) findings. *Structural Equation Modeling: A Multidisciplinary Journal*, *11*(3), 320–341. https://doi.org/10.1207/s15328007sem1103_2

Meade, A. W., Johnson, E. C., & Braddy, P. W. (2008). Power and sensitivity of alternative fit indices in tests of measurement invariance. *Journal of Applied Psychology*, *93*(3), 568–592. https://doi.org/10.1037/0021-9010.93.3.568

Meredith, W. (1993). Measurement invariance, factor analysis and factorial invariance. *Psychometrika*, *58*(4), 525–543. https://doi.org/10.1007/BF02294825

Nelson, L. D., Simmons, J., & Simonsohn, U. (2018). Psychology's renaissance. *Annual Review of Psychology*, *69*(1), 511–534. https://doi.org/10.1146/annurev-psych-122216-011836

Nye, C. D., Bradburn, J., Olenick, J., Bialko, C., & Drasgow, F. (2019). How Big Are My Effects? Examining the Magnitude of Effect Sizes in Studies of Measurement Equivalence. *Organizational Research Methods*, *22*(3), 678–709. https://doi.org/10.1177/1094428118761122

Nye, C. D., & Drasgow, F. (2011). Effect size indices for analyses of measurement equivalence: Understanding the practical importance of differences between groups. *Journal of Applied Psychology*, *96*(5), 966–980. https://doi.org/10.1037/a0022955

Putnick, D. L., & Bornstein, M. H. (2016). Measurement invariance conventions and reporting: The state of the art and future directions for psychological research. *Developmental Review*, *41*, 71–90. https://doi.org/10.1016/j.dr.2016.06.004

Rosseel, Y. (2012). Lavaan: An r package for structural equation modeling. *Journal of*

Statistical Software, *48*(1), 1–36. https://doi.org/10.18637/jss.v048.i02

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, *6*(2), 461–464. Retrieved from https://www.jstor.org/stable/2958889

Shadish, W. R., Cook, T. D., & Campbell, D. T. (2001). *Experimental and quasi-experimental designs for generalized causal inference*. Boston: Houghton Mifflin.

Sörbom, D. (1978). An alternative to the methodology for analysis of covariance. *Psychometrika*, *43*(3), 381–396. https://doi.org/10.1007/BF02293647

Stark, S., Chernyshenko, O. S., & Drasgow, F. (2006). Detecting differential item functioning with confirmatory factor analysis and item response theory: Toward a unified strategy. *Journal of Applied Psychology*, *91*(6), 1292–1306. https://doi.org/10.1037/0021-9010.91.6.1292

Steiger, J. H. (1990). Structural model evaluation and modification: An interval estimation approach. *Multivariate Behavioral Research*, *25*(2), 173–180. https://doi.org/10.1207/s15327906mbr2502_4

Tay, L., Meade, A. W., & Cao, M. (2015). An Overview and Practical Guide to IRT Measurement Equivalence Analysis. *Organizational Research Methods*, *18*(1), 3–46. https://doi.org/10.1177/1094428114553062

Thompson, B., & Daniel, L. G. (1996). Factor Analytic Evidence for the Construct Validity of Scores: A Historical Overview and Some Guidelines. *Educational and Psychological Measurement*, *56*(2), 197–208. https://doi.org/10.1177/0013164496056002001

Trent, L. R., Buchanan, E., Ebesutani, C., Ale, C. M., Heiden, L., Hight, T. L., … Young, J. (2013). A measurement invariance examination of the revised child anxiety and depression scale in a southern sample: Differential item functioning between african american and caucasian youth. *Assessment*, *20*(2), 175–187. https://doi.org/10.1177/1073191112450907

Tucker, L. R., & Lewis, C. (1973). A reliability coefficient for maximum likelihood factor analysis. *Psychometrika*, *38*(1), 1–10. https://doi.org/10.1007/BF02291170

Van De Schoot, R., Schmidt, P., De Beuckelaer, A., Lek, K., & Zondervan-Zwijnenburg, M. (2015). Editorial: Measurement invariance. *Frontiers in Psychology*, *6*. Retrieved from https://www.frontiersin.org/articles/10.3389/fpsyg.2015.01064

Vazire, S., Schiavone, S. R., & Bottesini, J. G. (2022). Credibility Beyond Replicability: Improving the Four Validities in Psychological Science. *Current Directions in Psychological Science*, *31*(2), 162–168. https://doi.org/10.1177/09637214211067779

Weidman, A. C., Steckler, C. M., & Tracy, J. L. (2017). The jingle and jangle of emotion assessment: Imprecise measurement, casual scale usage, and conceptual fuzziness in emotion research. *Emotion*, *17*(2), 267–295. https://doi.org/10.1037/emo0000226

Zwaan, R. A., Etz, A., Lucas, R. E., & Donnellan, M. B. (2018). Making replication mainstream. *Behavioral and Brain Sciences*, *41*, e120. https://doi.org/10.1017/S0140525X17001972

Appendix

Model Fit Statistics

Model fit statistics are provided for each of the ten model combinations (invariant, three sizes for each ladings, intercepts, and residuals). These tables could be used to examine the traditional change in fit statistics cutoff rules of thumb (Cheung & Rensvold, 2002), such as Δ CFI or Δ RMSEA, to the visualizations presented in the manuscript.

MGCFA Convenience Function

Please note that any partial invariance is not automatically included in this function. This function returns a list with all model summaries, the model coefficients in a tidy dataframe, and the model fit statistics in a tidy dataframe. You will need the libraries listed below for this function to work properly.

```
library(lavaan)
library(dplyr)
library(broom)
# CFA function
mgcfa <- function(data, group, model){

  group_names <- unique(data[ , group])

  model.overall <- cfa(model = model, data = data)
  model.group1 <- cfa(model = model,
                      data = subset(data, group == group_names[1]))
  model.group2 <- cfa(model = model,
                      data = subset(data, group == group_names[2]))
  model.configural <- cfa(model = model, data = data,
                          group = group, meanstructure = T)
```

```
model.metric <- cfa(model = model, data = data,
                    group = group, meanstructure = T,
                    group.equal = "loadings")
model.scalar <- cfa(model = model, data = data,
                    group = group, meanstructure = T,
                    group.equal = c("loadings", "intercepts"))
model.strict <- cfa(model = model, data = data,
                    group = group, meanstructure = T,
                    group.equal =  c("loadings", "intercepts", "residuals"))

model_coef <- bind_rows(
  tidy(model.overall, conf.level = .95) %>%
    mutate(model = "overall"),
  tidy(model.group1, conf.level = .95) %>%
    mutate(model = "group1"),
  tidy(model.group2, conf.level = .95) %>%
    mutate(model = "group2"),
  tidy(model.configural, conf.level = .95) %>%
    mutate(model = "configural"),
  tidy(model.metric, conf.level = .95) %>%
    mutate(model = "metric"),
  tidy(model.scalar, conf.level = .95) %>%
    mutate(model = "scalar"),
  tidy(model.strict, conf.level = .95) %>%
    mutate(model = "strict")
)

model_fit <- bind_rows(
  glance(model.overall) %>% mutate(model = "overall"),
```

```
    glance(model.group1) %>% mutate(model = "group1"),
    glance(model.group2) %>% mutate(model = "group2"),
    glance(model.configural) %>% mutate(model = "configural"),
    glance(model.metric) %>% mutate(model = "metric"),
    glance(model.scalar) %>% mutate(model = "scalar"),
    glance(model.strict) %>% mutate(model = "strict")
    )

  return(list(
    "model_coef" = model_coef,
    "model_fit" = model_fit,
    "model.overall" = model.overall,
    "model.group1" = model.group1,
    "model.group2" = model.group2,
    "model.configural" = model.configural,
    "model.metric" = model.metric,
    "model.scalar" = model.scalar,
    "model.strict" = model.strict
  ))

}
```

Measurement Invariance Plot Function

This function creates the plots shown in the manuscript. You will need the libraries listed for this function to work. Plots may be modified to rearrange for those who are familiar with `ggplot2`. Please note that the function assumes you will use the outputs from the previous `mgcfa` function or a tidy dataframe that includes the coefficients from the model with a column `model` that indicates which step of the MGCFA you are wanting to plot. If you have more than two groups, you should first filter the dataframe model coefficient outputs to only include to the two groups you want to compare. This code does not plot more than two groups (although, it could be modified for this, but the assumption here is that you only have two, as this is how you would normally proceed in a MGCFA using pairwise comparisons to find where the invariance occurs).

```
library(dplyr)
library(ggplot2)
library(cowplot)
library(lavaan)
# devtools::install_github("psyteachr/introdataviz")
library(introdataviz)
# Plot MI MGCFA
plot_mgcfa <- function(data_coef, # output from model_coef
                       model_step, # which model
                       item_name, # name of observed item
                       x_limits = c(-1,1), # LV limits to graph
                       y_limits, # Y min and max in data
                       ci_level, # what ci do you want
                       model_results, # what model results do you want
                       lv_name # which latent is the observed variable on
                       ){

  # calculate cutoff
  cutoff <- qt(p = (1-ci_level)/2,
               df = sum(unlist(model_results@Data@nobs)),
               lower.tail = F)

  # first get the data
```

```
graph.data <- data_coef %>% # put in tidy coefficients
filter(model == model_step) %>% # pick a model
filter(grepl(item_name, term)) %>%  # pick a question
mutate(group = factor(group, levels = names(table(data_coef$group)),
                      labels = c("Group 1", "Group 2")))

# make ribbon data y = slope*x + intercept for ci for slopes
ribbondata <- bind_rows(
  data.frame(
  x = seq(from = x_limits[1] - 1,
          to = x_limits[2] + 1,
          by = .05),
  group = unique(graph.data$group)[1]
) %>%
  mutate(ymin = (graph.data %>% filter(op == "=~") %>%
             slice_head() %>% pull(estimate) * x) -
           (cutoff*graph.data %>% filter(op == "=~") %>%
              slice_head() %>% pull(std.error)) +
           graph.data %>% filter(op == "~1") %>%
             slice_head() %>% pull(estimate),
         ymax = (graph.data %>% filter(op == "=~") %>%
             slice_head() %>% pull(estimate) * x) +
           (cutoff*graph.data %>% filter(op == "=~") %>%
              slice_head() %>% pull(std.error)) +
           graph.data %>% filter(op == "~1") %>%
             slice_head() %>% pull(estimate)),
  data.frame(
    x = seq(from = x_limits[1] - 1,
            to = x_limits[2] + 1,
```

```
             by = .05),
      group = unique(graph.data$group)[2]
    ) %>%
      mutate(ymin = (graph.data %>% filter(op == "=~") %>%
                       slice_tail() %>% pull(estimate) * x) -
                 (cutoff*graph.data %>% filter(op == "=~") %>%
                          slice_tail() %>% pull(std.error)) +
                 graph.data %>% filter(op == "~1") %>%
                       slice_tail() %>% pull(estimate),
             ymax = (graph.data %>% filter(op == "=~") %>%
                       slice_tail() %>% pull(estimate) * x) +
                 (cutoff*graph.data %>% filter(op == "=~") %>%
                          slice_tail() %>% pull(std.error)) +
                 graph.data %>% filter(op == "~1") %>%
                       slice_tail() %>% pull(estimate))
  )

  # make point data to draw on the intercepts
  pointdata <- data.frame(
  x = c(0,0),
  y = graph.data %>% filter(op == "~1") %>% pull(estimate),
  group = graph.data %>% filter(op == "~1") %>% pull(group),
  ymin = graph.data %>% filter(op == "~1") %>% pull(estimate) -
    cutoff * graph.data %>% filter(op == "~1") %>% pull(std.error),
  ymax = graph.data %>% filter(op == "~1") %>% pull(estimate) +
    cutoff * graph.data %>% filter(op == "~1") %>% pull(std.error)
  )

  # make the line data to draw on the slopes
```

```
linedata <- data.frame(
slope = graph.data %>% filter(op == "=~") %>% pull(estimate),
intercept = graph.data %>% filter(op == "~1") %>% pull(estimate),
group = graph.data %>% filter(op == "~1") %>% pull(group)
)

# make the distributions for the residuals
violindata <- data.frame(
y = c(rnorm(n = 1000,
          mean = graph.data %>% filter(op == "~~") %>%
            slice_head() %>% pull(estimate),
          sd = graph.data %>% filter(op == "~~") %>%
            slice_head() %>% pull(std.error)),
      rnorm(n = 1000,
          mean = graph.data %>% filter(op == "~~") %>%
            slice_tail() %>% pull(estimate),
          sd = graph.data %>% filter(op == "~~") %>%
            slice_tail() %>% pull(std.error))),
group = c(rep(graph.data %>% filter(op == "~~") %>%
            slice_head() %>% pull(group), 1000),
          rep(graph.data %>% filter(op == "~~") %>%
            slice_tail() %>% pull(group), 1000)),
x = 1
)

# make the latent mean data for right panel
latent_means <- lavPredict(model_results,
                             type = "lv",
                             label = TRUE,
```

```
                                assemble = TRUE,
                                append.data = TRUE)

latent_means$lv <- latent_means[ , lv_name]


# make a plot of the variance
variance_plot <-
ggplot(violindata, aes(x = 1, y = y, color = group, fill = group)) +
geom_split_violin() +
theme_void() +
theme(legend.position = "none") +
stat_summary(fun = "mean",
             geom = "crossbar",
             width = 0.5,
             colour = "black")

# make the plot with intercepts and slopes
intercept_plot <-
ggplot() +
# basic set up
theme_classic() +
xlab("Latent Variable") +
ylab("Observed Variable") +
coord_cartesian(xlim = x_limits, ylim = y_limits) +
# plot the intercepts
geom_point(data = pointdata,
           aes(x = x, y = y, color = group),
           inherit.aes = FALSE) +
geom_errorbar(data = pointdata,
```

```
                aes(x = x, ymin = ymin, ymax = ymax, color = group),
                inherit.aes = FALSE, width = .10) +
# plot the slopes
geom_abline(data = linedata,
              aes(slope = slope, intercept = intercept, color = group)) +
geom_ribbon(data = ribbondata,
              aes(x = x, ymin = ymin, ymax = ymax, fill = group),
              inherit.aes = FALSE, alpha = .2) +
scale_color_discrete(name = "Group") +
scale_fill_discrete(name = "Group") +
geom_vline(xintercept = 0) +
theme(axis.line.y = element_blank())

# make the latent means plot
mean_plot <- ggplot(latent_means, aes(x = lv, fill = group)) +
  geom_density(alpha = .2) +
  theme_classic() +
  xlab("Latent Variable") +
  ylab("Density") +
  geom_vline(data = latent_means %>%
               group_by(group) %>%
               summarize(mean = mean(lv)),
             aes(xintercept = mean, color = group)) +
  theme(legend.position = "none") +
  coord_cartesian(xlim = x_limits)

y_range = abs(y_limits[2] - y_limits[1])

# line up the two plots
```

```
  prow <- plot_grid(
    intercept_plot +
      ggtitle("Item Invariance") +
      theme(legend.position = "none") +
      annotation_custom(ggplotGrob(variance_plot),
                        xmin = .25, xmax = 1,
                        ymin = y_limits[1],
                        ymax = y_limits[2]-y_range/1.8),
    mean_plot +
      ggtitle("Latent Mean Distribution") +
      theme(legend.position = "none"),
    align = 'vh',
    hjust = -1,
    nrow = 1
  )

  # get the lengend
  legend_b <- get_legend(
    intercept_plot +
      guides(color = guide_legend(nrow = 1)) +
      theme(legend.position = "bottom")
  )

  # send out the plot
  plot_grid(prow, legend_b, ncol = 1, rel_heights = c(1, .1))

}
```

Table 1

Model Fit for Invariant Model

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
overall	7,515.72	7,578.94	0.99	0.99	0.02	0.02
group1	3,765.75	3,818.57	0.98	0.95	0.05	0.03
group2	3,761.95	3,814.77	0.98	0.96	0.04	0.03
configural	7,527.70	7,654.14	0.98	0.96	0.04	0.03
metric	7,529.39	7,638.97	0.95	0.93	0.06	0.05
scalar	7,522.90	7,615.62	0.96	0.96	0.04	0.05
strict	7,519.51	7,591.16	0.96	0.96	0.04	0.06

Table 2

Model Fit for Small Differences in Loadings

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
overall	7,537.67	7,600.89	0.98	0.96	0.04	0.02
group1	3,765.75	3,818.57	0.98	0.95	0.05	0.03
group2	3,777.83	3,830.66	0.98	0.96	0.05	0.03
configural	7,543.58	7,670.02	0.98	0.95	0.05	0.03
metric	7,548.90	7,658.48	0.94	0.92	0.07	0.06
scalar	7,541.81	7,634.53	0.95	0.95	0.05	0.06
strict	7,541.66	7,613.31	0.93	0.94	0.05	0.07

Table 3

Model Fit for Medium Differences in Loadings

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
overall	7,554.55	7,617.77	0.97	0.94	0.05	0.03
group1	3,765.75	3,818.57	0.98	0.95	0.05	0.03
group2	3,784.92	3,837.74	1.00	1.00	0.02	0.02
configural	7,550.67	7,677.11	0.99	0.98	0.04	0.03
metric	7,562.71	7,672.29	0.93	0.89	0.07	0.06
scalar	7,556.86	7,649.58	0.93	0.93	0.06	0.06
strict	7,558.05	7,629.70	0.91	0.92	0.06	0.08

Table 4

Model Fit for Large Differences in Loadings

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
overall	7,662.99	7,726.21	0.98	0.97	0.04	0.02
group1	3,765.75	3,818.57	0.98	0.95	0.05	0.03
group2	3,857.21	3,910.03	0.97	0.94	0.08	0.03
configural	7,622.96	7,749.40	0.97	0.94	0.06	0.03
metric	7,659.19	7,768.77	0.85	0.79	0.12	0.08
scalar	7,652.60	7,745.32	0.86	0.85	0.10	0.09
strict	7,660.63	7,732.27	0.82	0.85	0.10	0.12

Table 5

Model Fit for Small Differences in Intercepts

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
overall	7,519.69	7,582.91	1.00	0.99	0.02	0.02
group1	3,765.75	3,818.57	0.98	0.95	0.05	0.03
group2	3,770.41	3,823.23	0.93	0.86	0.08	0.04
configural	7,536.16	7,662.60	0.95	0.91	0.07	0.04
metric	7,531.36	7,640.94	0.96	0.94	0.05	0.04
scalar	7,531.34	7,624.06	0.94	0.93	0.06	0.05
strict	7,523.54	7,595.18	0.95	0.96	0.04	0.05

Table 6

Model Fit for Medium Differences in Intercepts

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
overall	7,542.77	7,605.99	1.00	1.00	0.01	0.02
group1	3,765.75	3,818.57	0.98	0.95	0.05	0.03
group2	3,770.41	3,823.23	0.93	0.86	0.08	0.04
configural	7,536.16	7,662.60	0.95	0.91	0.07	0.04
metric	7,531.36	7,640.94	0.96	0.94	0.05	0.04
scalar	7,554.20	7,646.92	0.85	0.83	0.09	0.07
strict	7,546.38	7,618.03	0.86	0.88	0.08	0.07

Table 7

Model Fit for Large Differences in Intercepts

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
overall	7,579.17	7,642.39	1.00	1.00	0.00	0.02
group1	3,765.75	3,818.57	0.98	0.95	0.05	0.03
group2	3,770.41	3,823.23	0.93	0.86	0.08	0.04
configural	7,536.16	7,662.60	0.95	0.91	0.07	0.04
metric	7,531.36	7,640.94	0.96	0.94	0.05	0.04
scalar	7,590.29	7,683.01	0.70	0.66	0.13	0.10
strict	7,582.47	7,654.12	0.71	0.75	0.11	0.10

Table 8

Model Fit for Small Differences in Residuals

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
overall	7,449.49	7,512.71	1.00	1.01	0.00	0.01
group1	3,765.75	3,818.57	0.98	0.95	0.05	0.03
group2	3,693.32	3,746.14	1.00	1.01	0.00	0.02
configural	7,459.07	7,585.51	0.99	0.98	0.03	0.03
metric	7,461.41	7,570.99	0.97	0.95	0.05	0.05
scalar	7,455.85	7,548.58	0.97	0.97	0.04	0.05
strict	7,453.48	7,525.12	0.96	0.97	0.04	0.05

Table 9

Model Fit for Medium Differences in Residuals

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
overall	7,378.57	7,441.79	1.00	1.00	0.00	0.02
group1	3,765.75	3,818.57	0.98	0.95	0.05	0.03
group2	3,597.77	3,650.60	1.00	1.03	0.00	0.02
configural	7,363.52	7,489.96	1.00	0.99	0.02	0.02
metric	7,366.63	7,476.21	0.97	0.96	0.05	0.05
scalar	7,360.15	7,452.87	0.98	0.98	0.03	0.05
strict	7,382.53	7,454.18	0.88	0.90	0.08	0.07

Table 10

Model Fit for Large Differences in Residuals

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
overall	7,294.21	7,357.43	1.00	1.01	0.00	0.01
group1	3,765.75	3,818.57	0.98	0.95	0.05	0.03
group2	3,453.47	3,506.29	0.95	0.90	0.07	0.03
configural	7,219.22	7,345.66	0.96	0.92	0.06	0.03
metric	7,216.38	7,325.96	0.96	0.94	0.05	0.04
scalar	7,210.65	7,303.37	0.96	0.96	0.04	0.05
strict	7,297.89	7,369.54	0.59	0.65	0.13	0.18

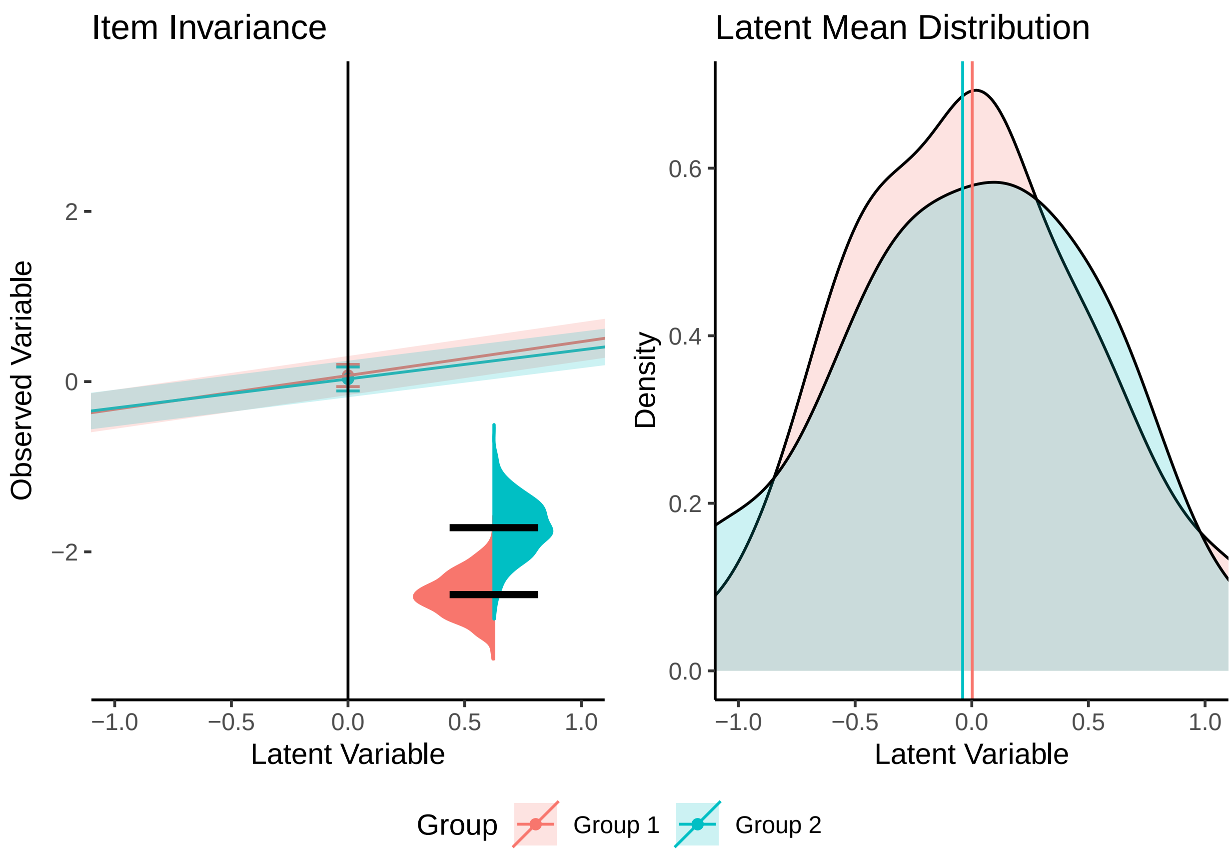

Figure 1. Invariant Model Visualization

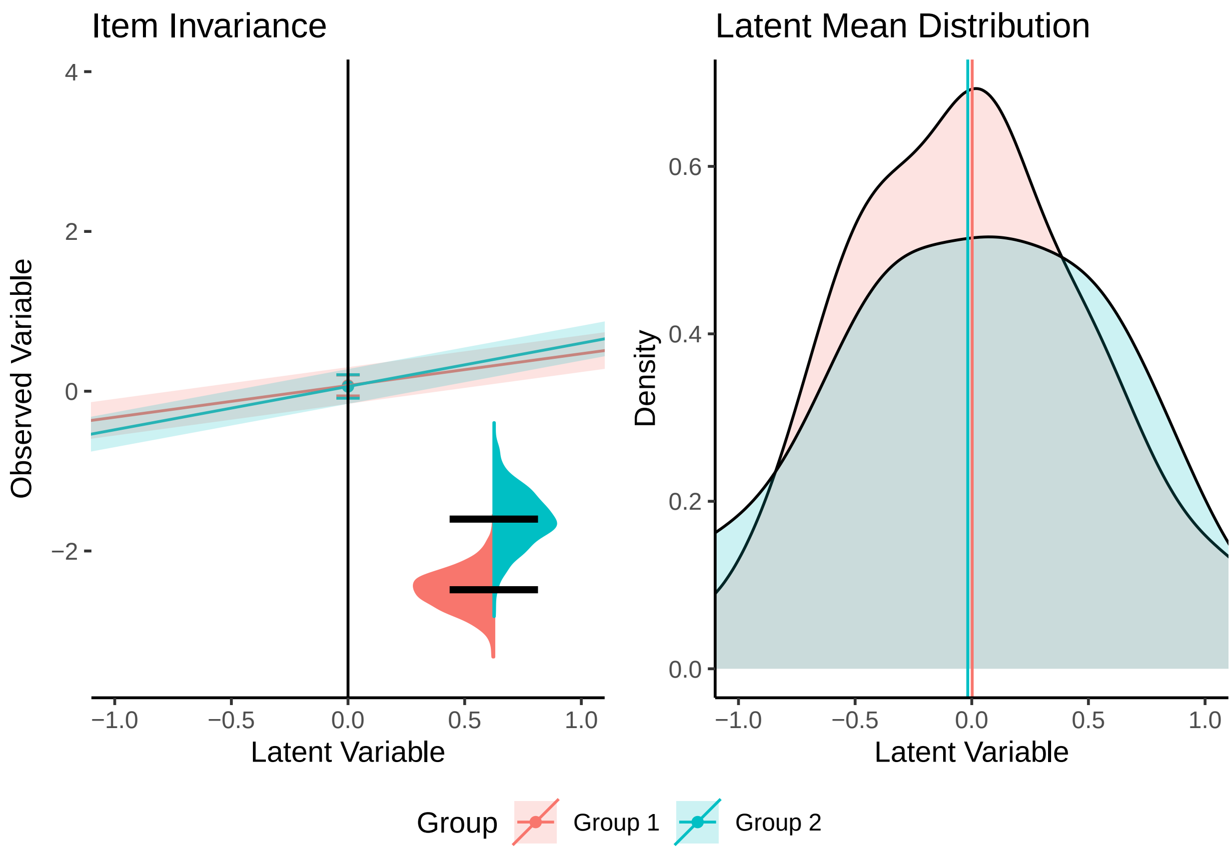

Figure 2. Small Loadings Model Visualization

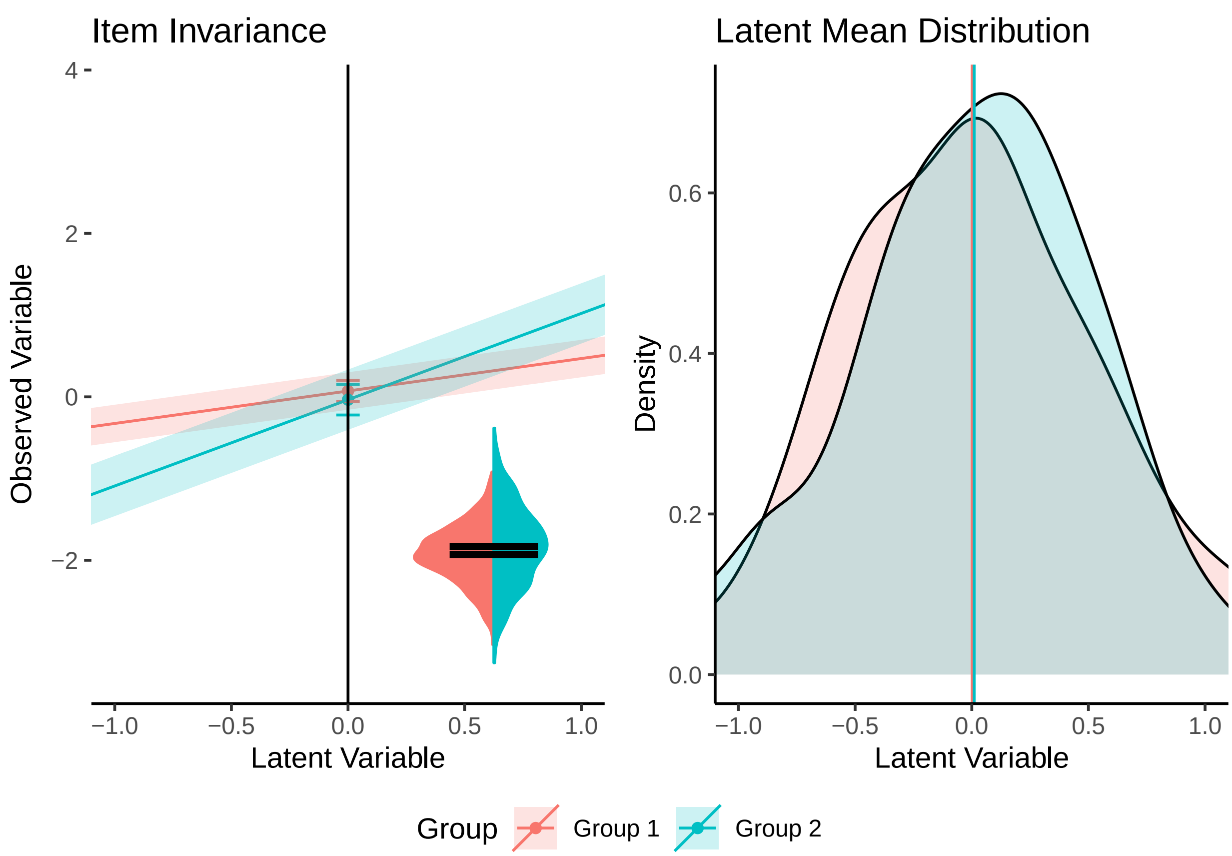

Figure 3. Medium Loadings Model Visualization

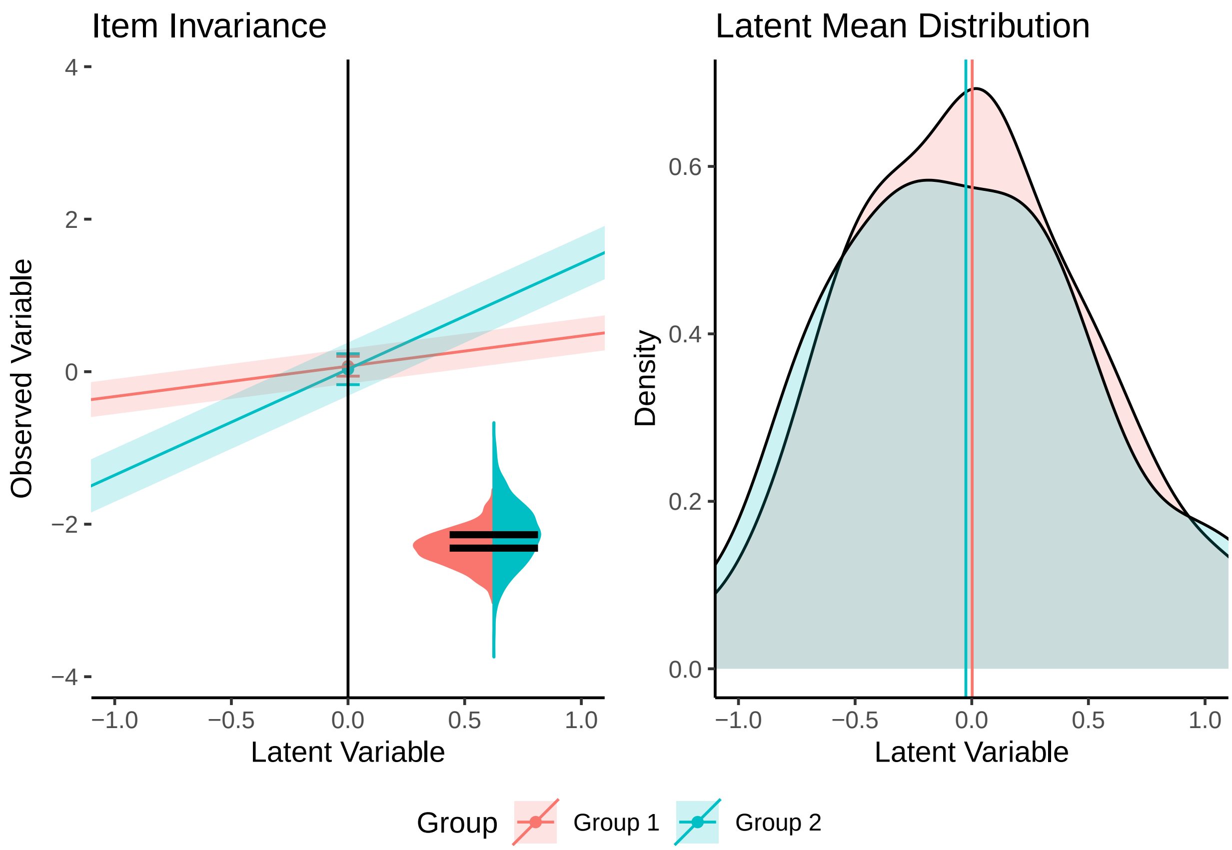

Figure 4. Large Loadings Model Visualization

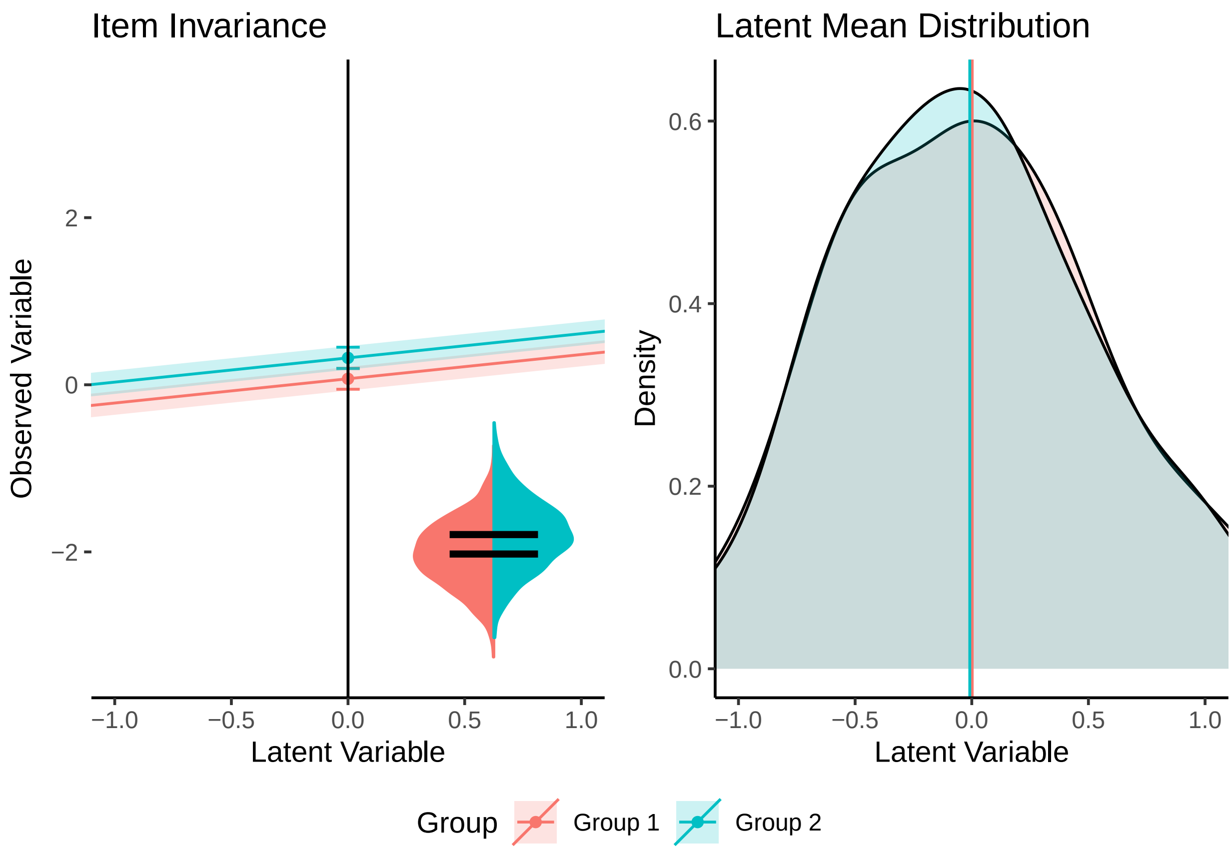

Figure 5. Small Intercepts Model Visualization

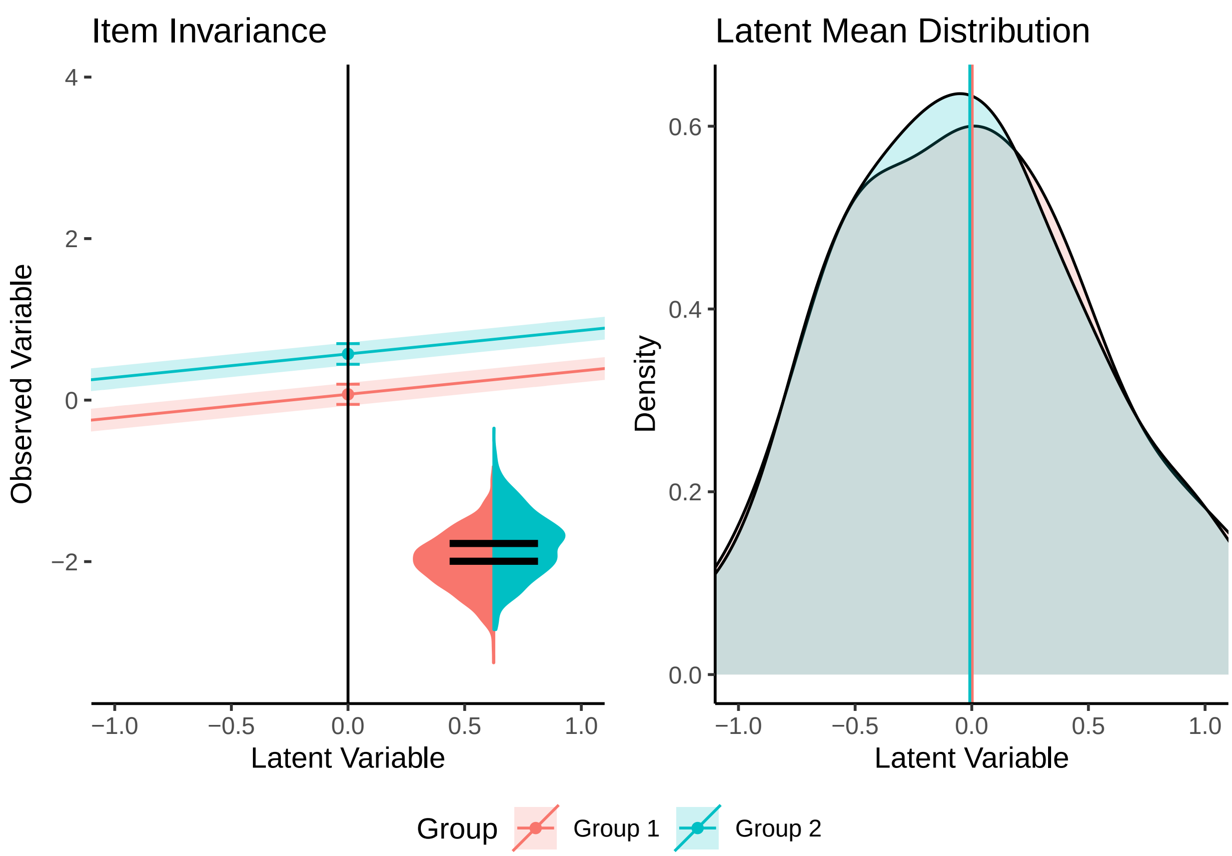

Figure 6. Medium Intercepts Model Visualization

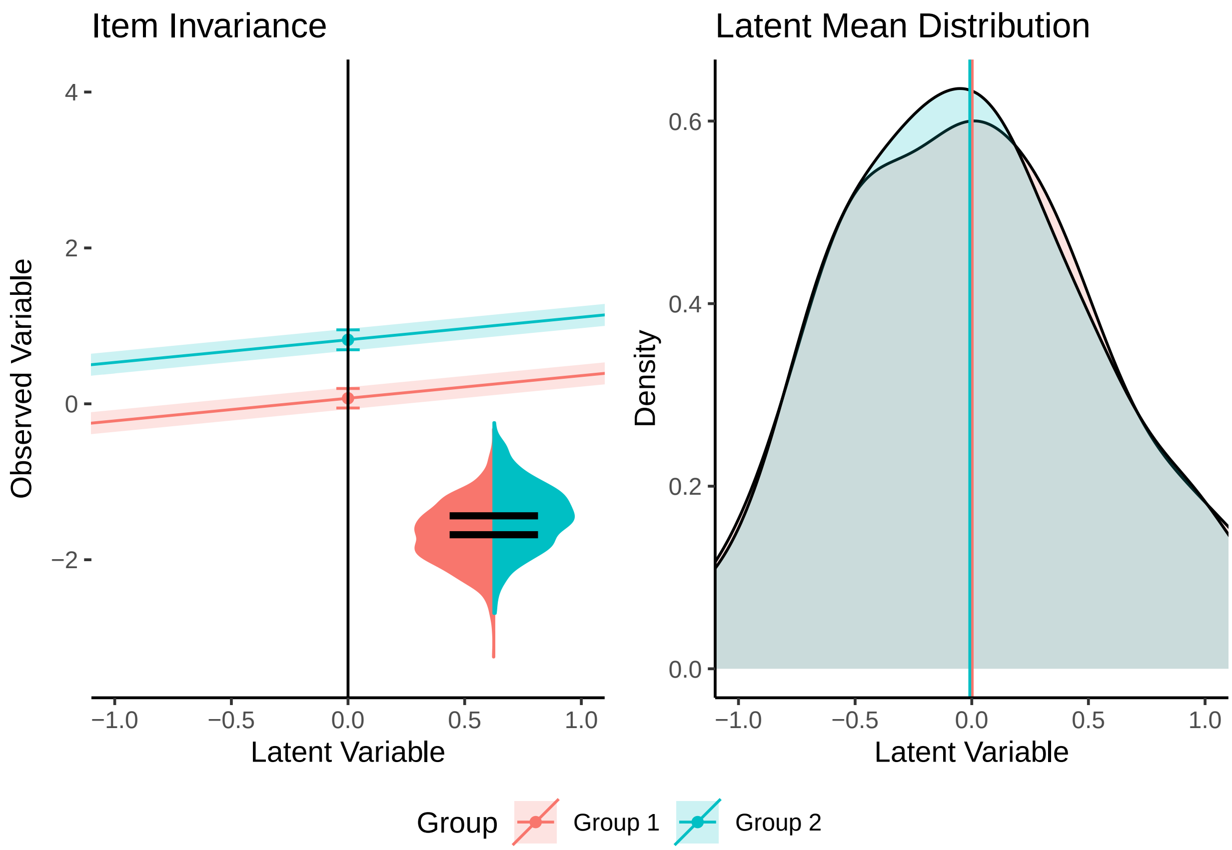

Figure 7. Large Intercepts Model Visualization

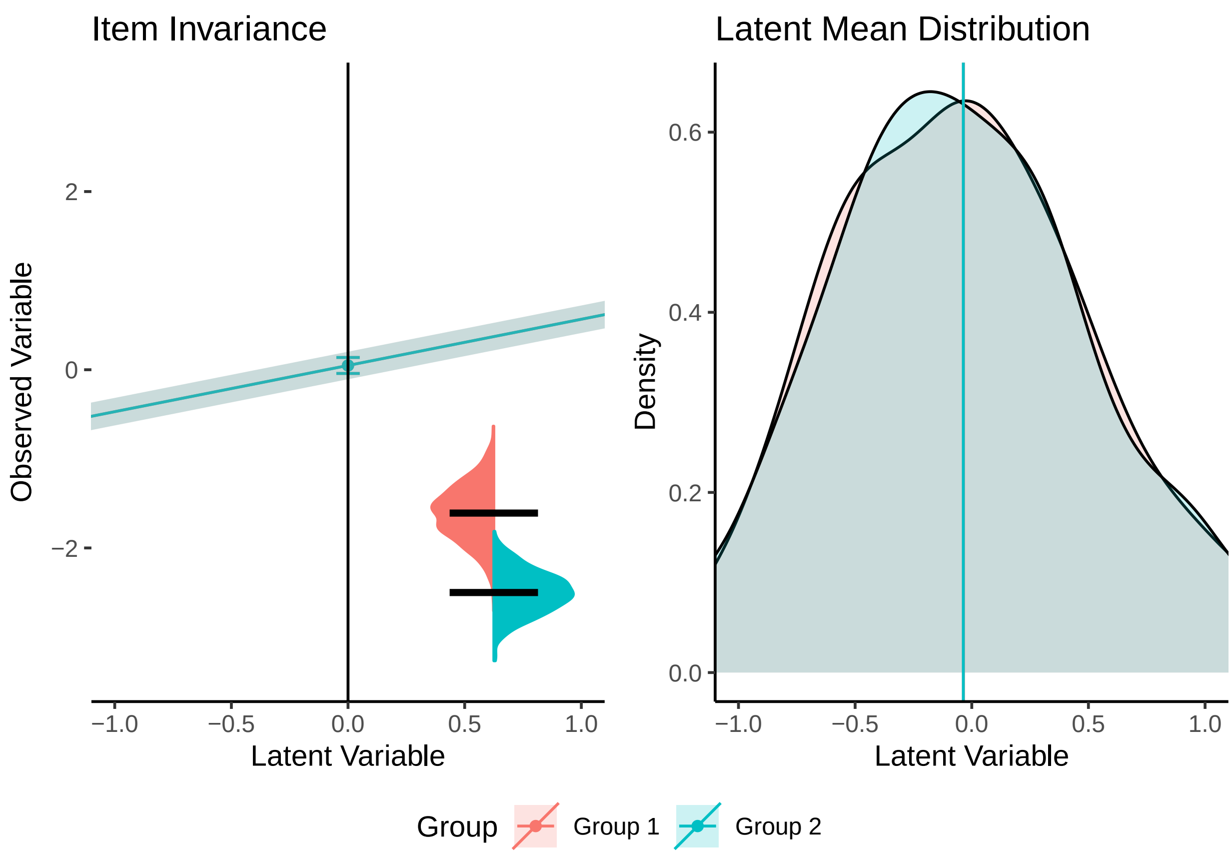

Figure 8. Small Residuals Model Visualization

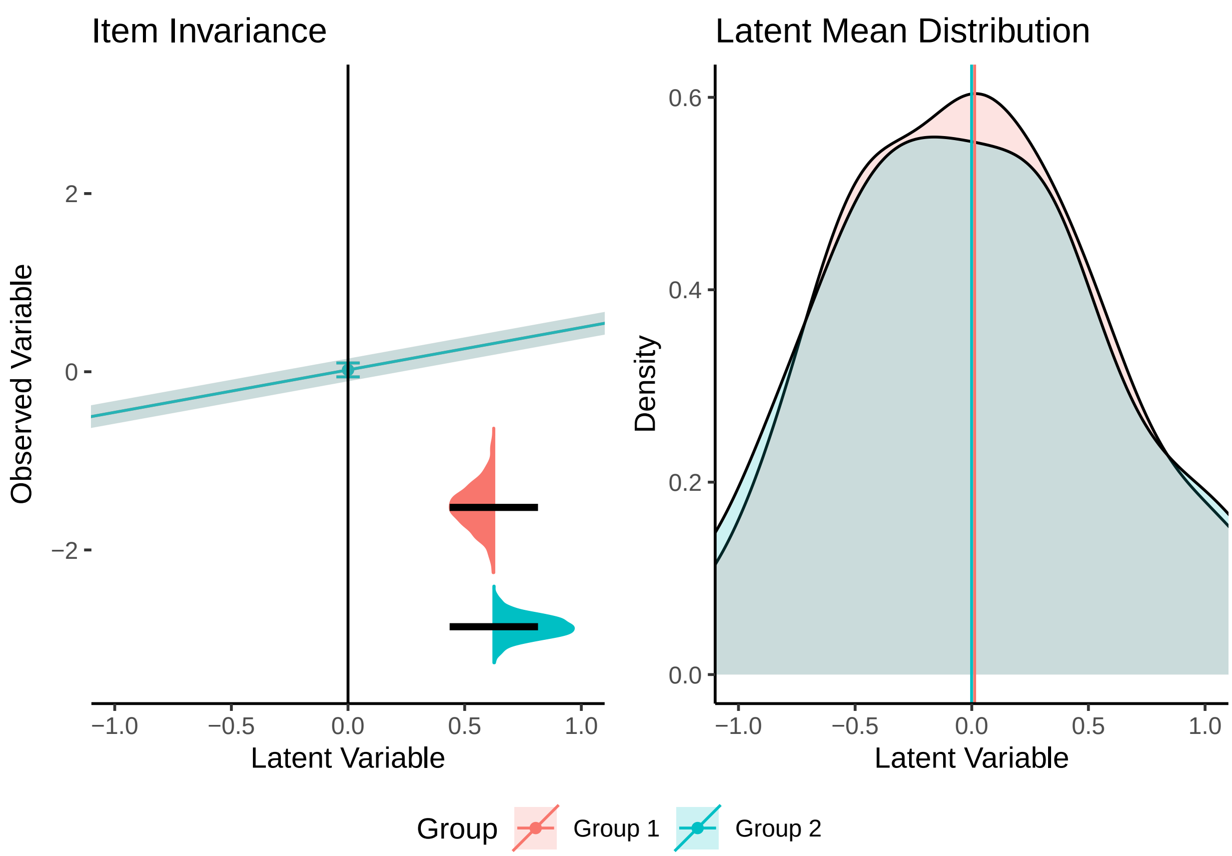

Figure 9. Medium Residuals Model Visualization

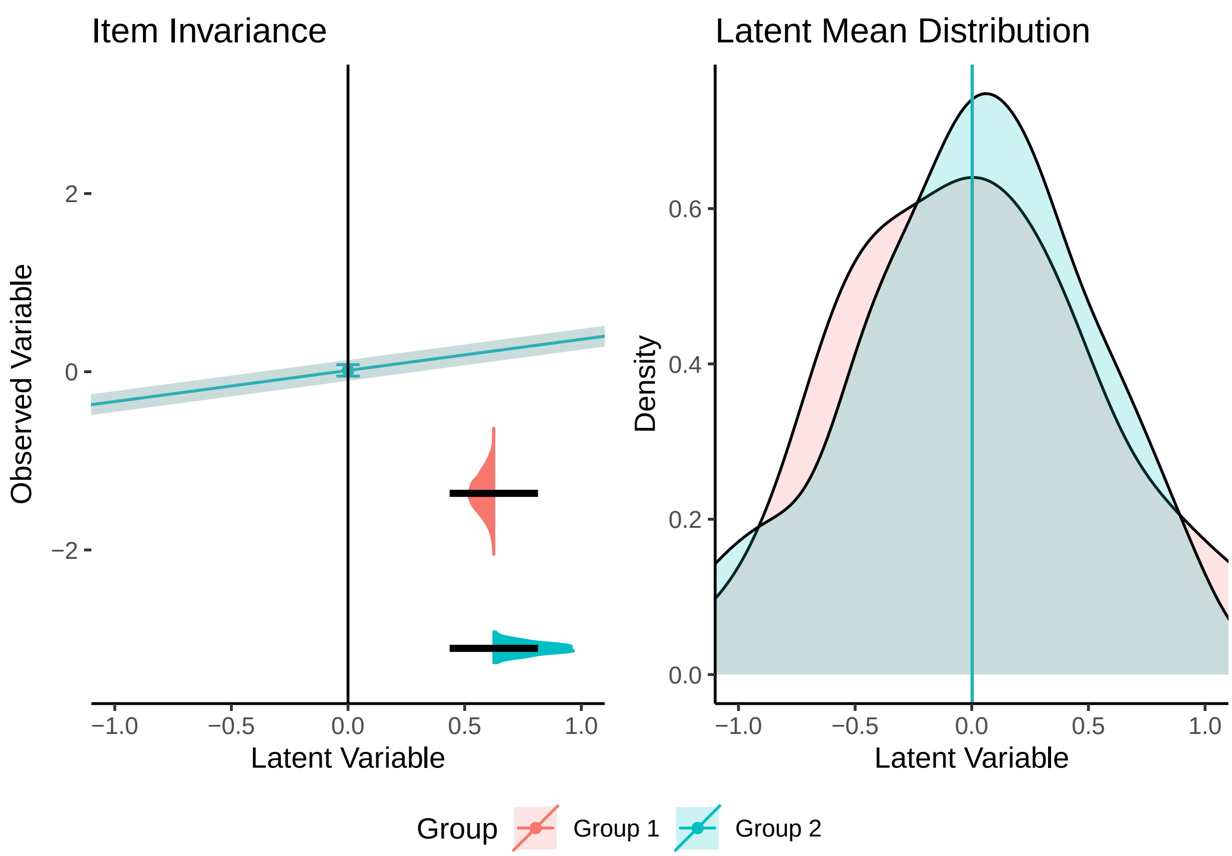

Figure 10. Large Residuals Model Visualization