visualizemi: Visualization, Effect Size, and Replication of Measurement Invariance for Registered Reports

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.

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 measures of latent constructs wherein a measurement model is established. 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 et al., 2018), interest in divergence in measurement has increased, often focused on small parameter differences within the latent model. This manuscript presents `visualizemi`, an *R* package that provides functionality to calculate multigroup models, partial invariance, visualizations for (non)-invariance, effect sizes for models and parameters, and potential replication rates compared to random models. Readers will learn how to interpret the impact and size of the proposed non-invariance in models with a focus on potential replication and how to plan for registered reports.

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

visualizemi: Visualization, Effect Size, and Replication of Measurement Invariance for Registered Reports

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 dimensionality, 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 (Barry et al., 2014; Weidman et al., 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 (Makel et al., 2012; Makel & Plucker, 2014; Zwaan et al., 2018), reproducibility (Nelson et al., 2018), and the credibility of our results (Vazire et al., 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 et al., 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 et al., 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 interpreted 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 et al., 2015).

Measurement invariance has been explored and implemented for the last fifty years (Jöreskog, 1971; Sörbom, 1978) and implemented in the most popular structural equation modeling programs (Boker et al., 2011; Jöreskog & Sörbom, 2001; Rosseel, 2012). Byrne et al. (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. As the field pushes back against favoring cutoff criteria and rules of thumb (Marsh et al., 2004; 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. Yet, 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. The field of psychology is increasingly interested in pre-registration (i.e., registering plans for a study before data collection, Nosek et al., 2018) and the promotion of transparency in study design, implementation, and analysis (Mayo-Wilson et al., 2021), in addition to supporting replication studies (Zwaan et al., 2018). Registered (replication) reports provide an advantageous avenue for the pre-registration of measurement tests, as they allow a researcher the ability to have their study accepted in principle, regardless of the results of a test of construct validity, reliability, or measurement invariance (Hobson, 2019; Nosek & Lakens, 2014). However, there are few tools that can provide effect size measures for models, individual parameters, or visualization for researchers to plan for future studies. d_{MACS} provides the opportunity to begin to think about the smallest effect size of interest or the smallest meaningful effect size for measurement invariance and replication (Anvari & Lakens, 2021; i.e., two studies with overlapping confidence intervals "replicate," even if the test of measurement invariance does not, Lakens, 2017). 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.

Therefore, purpose of this manuscript is describe an *R* package, `visualizemi`, that provides functionality to calculate multi-group confirmatory factor analysis, partial invariance tests, visualizations of the size of non-invariance, and potential effect sizes for overall models and individual parameters. 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. The proposed effect sizes demonstrate the likelihood of replication with a

similar sample as compared to a randomly assigned group model, thus, illustrating what type of measurement one might expect to find, and how different that is from random chance. Within this technique, the individual parameter effect sizes can calculated: both the group differences within a model as compared to random and the likelihood of a parameter replication compared to random groups. Coupled with other indicators (i.e., fit indices differences, d_{MACS}), we can move toward a better understanding of how much measurement non-invariance is meaningful. This tutorial and package will help researchers plan future studies and aid in the ability to estimate a smallest effect of interest for measurement invariance studies, rather than relying on fit indices and rules of thumb alone.

By the end of this tutorial manuscript, readers will:

1. Learn how to use *visualizemi* to analyze multi-group confirmatory factor analysis, examine partial invariance, and create visualizations of parameters.
2. Learn how to estimate the potential replication of multi-group models and their parameters using bootstrapping compared to a random group model.
3. Be able to calculate and interpret effect sizes for model and parameter replication, as well as parameter group differences.
4. Understand the impact of measurement variability on replication and generalizability.

The tutorial will start with simulated data based on known effect sizes using d_{MACS} and demonstrating the package functions for 1) running the multigroup analysis, 2) running a partial invariance analysis, 3) plotting the partial invariance, 4) estimating replication and effect sizes at the model level, and 5) estimating replication and effect sizes at the parameter level. Last, data from Aiena et al. (2014) examining the measurement invariance of the RS-14 (Wagnild, 2009) will be used to demonstrate the application of the package on real data. The *visualizemi* package vignette includes an additional tutorial walk through.

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 (see supplemental for examples). 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 package functions do not require this assumption). 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 simulated.

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. A convenience function `mgcfa` is used for these steps or other measurement invariance test orders and combinations. 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 square 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 et al. (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 (Dueber, 2023; Nye & Drasgow, 2011). Only one item in each model was manipulated from the invariant model to create the non-invariant models. Given the data was simulated with a z-score scaling, the loading values were simulated at .30 points apart (given d_{MACS} suggestions of .2, .4, .7), the intercepts at .25 points apart, and the residuals at .25 points apart. To plan a simulation for your own study, these values can be used to simulate small, medium, and large non-invariance effects by first converting data into z-score.

Package Code Examples

The complete code for this manuscript can be found at https://osf.io/wev5f/. This tutorial was registered at https://osf.io/vwf4d, and the example provided at the end of the manuscript was added after that registration. The *R* package and replication/effect sizes was added after the original manuscript submission.

Multi-group CFA Caculation

First, we would create our model code in *lavaan* syntax (Rosseel, 2012). The `lv` latent variable predicts the five measured variables, which are present as columns in our `df.invariant` data set. The package does generally require raw data for bootstrapping purposes, and an example of how to simulate data from models and covariance/correlations tables sometimes provided in manuscripts (rather than the raw data) is provided in the supplemental documentation.

lavaan automatically sets the mean (i.e., the intercept) for latent variables to zero. If we wish to visualize the impact of the changes in parameter estimates across groups on the latent means, we need to allow the latent mean estimation with `lv ~ 1`. However,

adding this estimation into our model will create a non-identified model. To solve this problem, you can set one of the intercepts of another variable to a value to scale the model. Here we will set the scale of the model by using `q1 ~ 0*1`, thus, scaling the expected means to zero. With simulation, this step is easy to know which variable to pick - we set the intercept on the variable we know did not show differences. In real data, you may wish to run the model steps *without* setting this option, examine the results of a configural or separate models, and then add the option for the values most similar. Additionally, you could complete partial invariance steps to determine which value appears most consistent to fix the estimate.

```
# create lavaan model
model.overall <- "
# overall one-factor model
lv =~ q1 + q2 + q3 + q4 + q5
# set the intercept (mean) of q1 to zero
q1 ~ 0*1
# allow the lv intercept to be freely estimated
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
```

The `mgcfa` function is designed to flexibly allow you to leverage `lavaan`'s package functions to calculate multiple measurement steps at once. You would include:

1) the model syntax in the `model` argument
2) the dataframe in the `data` argument of our function
3) the name of the grouping variable in quotes for `group`
4) and the equality constraints you would like to impose in order in `group.equal`
5) `...` any other *lavaan* arguments you would like to use such as `meanstructure` or `estimator`.

Note: you can also use `sample.cov`, `sample.mean`, `sample.nobs` in this step for estimation of multi-group models, but simulated dataframes are needed for bootstrapping replication estimates.

```
# run our mgcfa function to run all models
results.invariant <-
  # name of the saved model syntax
  mgcfa(model = model.overall,
        # name of the dataframe
        data = df.invariant,
        # name of the grouping variable
        group = "group",
        # equality constraints to impose in order
        group.equal = c("loadings", "intercepts", "residuals"),
        # other options to send to lavaan cfa function
        meanstructure = T)

# what is saved for you
names(results.invariant)
```

```
## [1] "model_coef"        "model_fit"         "model_overall"
## [4] "group_models"      "model_configural"  "invariance_models"
```

The following output is saved:

1) `model_coef`: The parameter estimates for each model with the model step included in a *model* column. This set of coefficients can be used for other functions. This dataframe is created with *broom*'s `tidy()` function if you wish to recreate this table without running the `mgcfa()` function (Robinson et al., 2023).

```
results.invariant$model_coef[1:10 , ]
```

```
## # A tibble: 10 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 =~ " =~      1         0         NA     NA         0.803   0.616   0.616
##  2 "lv =~ " =~      0.655     0.0880     7.44   9.77e-14  0.526   0.493   0.493
##  3 "lv =~ " =~      0.640     0.0895     7.15   8.83e-13  0.514   0.463   0.463
##  4 "lv =~ " =~      0.277     0.0749     3.69   2.24e- 4  0.222   0.209   0.209
##  5 "lv =~ " =~      0.955     0.117      8.13   4.44e-16  0.766   0.656   0.656
##  6 "q1 ~1 " ~1      0         0         NA     NA         0       0       0
##  7 "lv ~1 " ~1     -0.0305    0.0582    -0.524  6.00e- 1 -0.0380 -0.0380 -0.0380
##  8 "q1 ~~ " ~~      1.05      0.0995    10.6    0         1.05    0.620   0.620
##  9 "q2 ~~ " ~~      0.860     0.0653    13.2    0         0.860   0.757   0.757
## 10 "q3 ~~ " ~~      0.966     0.0711    13.6    0         0.966   0.785   0.785
## # i 4 more variables: model <chr>, block <int>, group <int>, label <chr>
```

2) `model_fit`: The model fit indices from `fitmeasures()` to review for overall model fit and invariance judgments. The name of the model is included in a *model* column.

```
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 0.998 7516. 7580. 1      0.650    15 0               0      0.00616 1.04
## 2 0.948 3766. 3819. 0.976  7.79     15 0.0473          0.108  0.0312  0.953
## 3 0.974 3768. 3820. 1      4.48     15 0               0.0831 0.0210  1.01
## 4 0.961 7533. 7660. 0.991 12.3      30 0.0301          0.0785 0.0261  0.982
## 5 0.965 7528. 7638. 0.994 15.4      26 0.0200          0.0660 0.0330  0.992
## 6 0.969 7522. 7615. 1     17.3      22 0               0.0542 0.0352  1.00
## # i 8 more variables: converged <lgl>, estimator <chr>, ngroups <int>,
## #   missing_method <chr>, nobs <int>, norig <int>, nexcluded <int>, model <chr>
```

3) `model_overall`: A saved *lavaan* fitted model of all groups together without any equality constraints or grouping variables. These objects can be used with any function that normally takes a saved model: `parameterEstimates()`, `modificationIndices()`, `semPlot::semPaths()`, and so on (Epskamp, 2022).
4) `group_models`: A list of saved fitted models for each group separately.
5) `model_configural`: A saved fitted model for the configural model that nests together each group into one model with no other constraints.
6) `invariance_models`: A list of saved fitted models that consecutively adds `group.equal` constraints.

Visualization of Invariance

Package Function

The results from the `model_coef` table can then be used directly in `plot_mi()`. 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 (Robinson et al., 2023). 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-axis 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. This value also constrains the latent mean diagram to help zoom in on group differences because the scale of latent means is usually centered over zero. You can use this parameter to zoom out to a more traditional histogram using `c(-2, 2)`.
5) `y_limits`: What do you want the y-axis 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) `conf.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.

9) `plot_groups`: If you include more than two groups in a multi-group model, the automatic assumption is that you want the first two groups for this visualization. If not, include the names of the groups here to plot.

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


names(invariant.plot)
```

```
## [1] "complete"  "intercept" "mean"      "variance"
```

The output from this function are several *ggplot2* objects that can be edited or saved directly using *ggplot2* functionality (Wickham, 2016).

1) `complete`: 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. All plots are made with *ggplot2* and *cowplot* (Wilke, 2020).

2) `intercept`: Only the left hand side of the complete plot designed to represent intercepts and factor loadings. Factor loadings represent the slope of the regression equation for the latent variable predicting the scores on the observed variable ($\hat{Y} \sim b_0 + b_1 X + \epsilon$). 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 0.21. 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.

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.

3) `mean`: The right hand side of the complete plot graphing the latent variable means and density from the data. 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 lines indicate the means of the latent variables from the simulated dataset. Group labels are represented in the figure caption on the bottom. Group 1 is usually the group that is alphabetically first in the data set or whichever group is the first that appears when using the `levels()` command.

4) `variance`: A split geom violin plot indicating the variance distribution of the plotted item. 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: 1.22). 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. The plots are included separately so they can be arranged in a different fashion if desired.

Simulated Results

The d_{MACS} value for item 4 in the invariant model was 0.16, 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 ($d_{MACS} = 0.27$) difference in loadings, while Figure 3 displays the results from the medium ($d_{MACS} = 0.53$) difference in loadings, and Figure 4 shows the large ($d_{MACS} = 0.68$) 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 data set 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 in this model are estimated as equal 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 et al., 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.

The latent means in Figure 4 do appear to show differences, albeit visually small. The latent means diagram shows the impact of any group differences that aren't constrained, and this image shows the configural model (as the metric model would force them to be equal). In the simulated model, the *only* manipulated parameter is question 4's loading. In real models, the differences may be larger due to other variation found in the parameter estimates. Therefore, once you discover items you believe would make a model "partially" invariant, you may wish to estimate that model and graph the item again using the partially invariant model to see only the effect of the non-invariant items. Additionally, consider that we set the scaling of the model to 0. The estimate for the lv mean in the large loading model was group 1: 0.00, and group 2: -0.06, which results in 0.06 difference in group means. The practical implications of this difference will depend on the research and interpretations of the researcher.

For intercepts, the small (Figure 5), medium (Figure 6), and large (Figure 7) depictions represent d_{MACS} values of 0.26, 0.47, and 0.70, respectively. Intercept differences can be clearly seen represented by the spacing out of the intercept locations (and thus, the overall line as well). While the changes in intercept do not appear to change the latent means, the caveat to this simulation is that only item four was manipulated. An example is provided below that demonstrates large changes in latent means.

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 available for the residuals, our models showed 0.19, 0.19, and 0.16, respectively. These differences in values are variable due to the random generation of data sets 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.

Partial Invariance Calculation

Package Function

The results of the simulated models are presented in the appendix, demonstrating that each simulated dataset shows partial invariance if item four is allowed to vary between groups. The function takes the following arguments:

1) `saved_model`: The saved *lavaan* model with the equality constraints at the level of measurement invariance you would like to examine for partial invariance.

2) `data`: The dataframe where the model was estimated.

3) `model`: The model syntax for the overall model.

4) `group`: The grouping variable column in the dataframe.

5) `group.equal`: The equality constraints including in your original multi-group tests.

6) `partial_step`: The level of partial invariance you wish to test.

```
partial.invariant <-
  partial_mi(
    # saved model output with constraints
    saved_model = results.invariant$invariance_models$model.residuals,
    # dataframe from model
    data = df.invariant,
    # model syntax
    model = model.overall,
    # group column name
    group = "group",
    # group equality constraints from your mgcfa
    group.equal = c("loadings", "intercepts", "residuals"),
    # which step you want to examine for partial invariance
    partial_step = "residuals"
    )


names(partial.invariant)
```

```
## [1] "models"    "fit_table"
```

In this function, each parameter with the appropriate *lavaan* syntax is relaxed individually (i.e., `~1` for intercepts, `~~` for residuals, etc.). The fitted models are saved in the `models` output, and the `fit_table` output includes all fit indices for each model to investigate potential areas of partial invariance based on the researcher's desired criterion.

Note: the `partial_step` function is used to determine which types of `op` or operators to freely estimate between groups. If one chooses residuals, you will also freely estimate the residual for the latent variable or any other residuals found in the model. These items may be ignored if they were not meant to be included.

```
head(partial.invariant$fit_table %>%
       dplyr::select(free.parameter, cfi, rmsea))
```

```
## # A tibble: 6 x 3
##   free.parameter cfi        rmsea
##   <chr>          <lvn.vctr> <lvn.vctr>
## 1 q1 ~~ q1       0.9902679  0.02108648
## 2 q2 ~~ q2       0.9868905  0.02447336
## 3 q3 ~~ q3       0.9958241  0.01381266
## 4 q4 ~~ q4       1.0000000  0.00000000
## 5 q5 ~~ q5       0.9868088  0.02454944
## 6 lv ~~ lv       0.9906154  0.02025143
```

Replication and Effect Size: Model

Package Function

The `bootstrap_rr` function in *visualizemi* was designed to estimate the likely replication of overall model invariance with the assumption that the data used for the estimation represents the larger population. The following arguments are used:

1) `saved_configural`: a saved fitted model at the configural level with no equality constraints. This model should include all other *lavaan* settings you would like to use, such as `estimator` or `ordered`.

```
boot.model.invariant <-
  bootstrap_rr(
```

```
    # saved configural model
    saved_configural = results.invariant$model_configural,
    # dataframe
    data = df.invariant,
    # model syntax
    model = model.overall,
    # group variable column in dataframe
    group = "group",
    # number of bootstraps
    nboot = 1000,
    # which fit index you would like to use
    invariance_index = "cfi",
    # what is your criterion for that fit index
    invariance_rule = .01,
    # what equality constraints are you testing
    group.equal = c("loadings", "intercepts", "residuals")
)
```

The data included in this function will be sampled, with replacement, at the same size as the current dataset, and the included invariance equality constraints are estimated. Each step will be compared to the previous step using the invariance index and comparison rule entered. The output is a dataframe of the proportion of non-invariant bootstraps from the real data and the same bootstrapped dataset with the group labels randomly assigned. The effect size comparison of proportions, h, for non-invariant comparisons:

$$h_{nmi} = 2 \times (asin\sqrt{p_{data}} - asin\sqrt{p_{random}})$$

The alternative, h_{mi}, for effect size of measurement invariance replication would simply be the inverse sign of h_{nmi} and is also included in the table. Two additional columns h_{nmi_p}

and h_{nmi_p} represent the h values divided by the upper bound of h (i.e., π), to help with interpretation of the effect size (thus, bounding h to -1 to 1).

Simulated Results

Figure 11 portrays the h_{nmi_p} values by simulated non-invariance, strength of non-invariance, and type of equality constraint. This image represents 100 simulations of data by 1000 bootstrapped runs (averaged) to explore the expected pattern of results. The bars are arranged to show what a researcher might inspect when thinking about replication possibilities and their effect sizes (i.e., only three bars for each equality constraint would be calculated).

In the data that was simulated to be invariant between groups, effect sizes are still non-zero (loadings $h_{nmi_p} = 0.28$, intercepts = $h_{nmi_p} = 0.06$, $h_{nmi_p} = 0.00$). This result mirrors the effects found in the literature - that often, many models fail to show invariance, and potentially not because measurement is poor but because of natural random variation in parameter estimates. This result also indicates the need to be able to identify if specific parameters are driving the differences, which is shown in the next section.

Next, Figure 11 demonstrates the patterns one might find for small, medium, and large effects at each type of invariance when data is simulated with *one* difference. For loadings, the pattern shows a larger effect for loadings with zero or negative effect sizes for other effect sizes. The intercept simulations show non-zero effect sizes in the loadings and intercepts, likely for the same reasons d_{MACS} is interpreted as a combined effect size. When intercepts are changed, loadings may naturally shift with those means. Last, the residual results present an unexpected pattern, wherein the effect is primarily seen in the loadings, rather than the residuals step. However, when distributions of error variance are different, one may expect that those effects are pushed toward the loadings as well (as values can vary more, thus potentially weakening the relationship between observed and latent variable).

An example of interpretation on real data is given in a later section. From a research study, only one effect size for each equality constraint would be calculated. The interpretation will often be up to the researcher's smallest effect of interest, and this simulation gives some guidance that the values should not be interpreted with traditional rules of thumb. The pattern of effects is potentially the most useful information: 1) positive effects on the loadings with negative or very close to zero effects on the other parameters may indicate a non-replication in loadings, 2) equal effects on loadings and intercepts with smaller or negative effects may indicate intercepts may be an issue, and 3) residuals may be determined by the same pattern as loadings but with a smaller ratio of loadings to residuals effect (i.e., loadings h_{nmi} / residuals h_{nmi}. The "size" could be determined by the ratio of effect sizes for each constraint. Of course, this represents one simulation study, and results from many studies in a meta-analysis would be fruitful for future work.

Replication and Effect Size: Parameters

Package Function

After examining the overall model potential replication effect size, the individual parameters within a model can be bootstrapped for partial invariance to with that parameter relaxed (overall partial model statistics) and the difference in group parameter estimates (parameter effect size). This function uses arguments seen in other functions, so they will not be repeated here. The general setup consists of using the model you think could be partially invariant in the `saved_model` argument and the fit index for comparison for the model with less constraints in `invariance_compare`. This example examines the loadings in the invariant model, so `saved_model` uses the `mgcfa` output for equality constraints present on the loadings and compares that model to the configural model with no equality constraints on the loadings. The `partial_step` argument will be used to determine which operation syntax (i.e. `=~` for loadings) to relax for modeling.

```
boot.partial.invariant <-
      bootstrap_partial(
        # saved model you want to examine the partial loadings for
        saved_model = results.invariant$invariance_models$model.loadings,
        # the dataset
        data = df.invariant,
        # the model
        model = model.overall,
        # the group variable in the dataset
        group = "group",
        # number of bootstraps
        nboot = 1000,
        # which fit index you would like to use to determine partial invariance
        invariance_index = "cfi",
        # what is the invariance rule
        invariance_rule = .01,
        # what are we comparing the saved model fit index to
        invariance_compare = fitmeasures(results.invariant$model_configural, "cfi"),
        # what step are we using for invariance
        partial_step = "loadings",
        # what equality constraints should be imposed
        group.equal = c("loadings")
      )
```

```
names(boot.partial.invariant)
```

```
## [1] "invariance_plot"        "effect_invariance_plot" "density_plot"
## [4] "boot_DF"                "boot_summary"           "boot_effects"
```

The saved output includes several dataframes and plots. The first is the `boot_DF`

which the summary of each run in a dataframe for plotting or summarization. This dataframe includes the estimate for each paramter (`term`) separated by group and type (`boot_1`, `boot_2` are the bootstrapped estimates for group 1 and group 2, while the same `random` columns indicate the randomly assigned groups). The fit index used to determine invariance is included for bootstrapped and random estimates, and then the differences between groups and if they were "invariant" or not given the researcher supplied rule.

```
head(boot.partial.invariant$boot_DF)
```

```
##       term    boot_1     boot_2  random_1  random_2  boot_fit random_fit
## 1 lv =~ q1 0.4548783 0.49928877 0.4627486 0.4651391 0.9296990  1.0000000
## 2 lv =~ q2 0.3599017 0.56241016 0.4100874 0.4980215 0.9441125  1.0000000
## 3 lv =~ q3 0.4254283 0.33640233 0.4274329 0.3422124 0.9377130  1.0000000
## 4 lv =~ q4 0.3930716 0.03320619 0.1380833 0.2628802 0.9750274  1.0000000
## 5 lv =~ q5 0.7306414 0.73512673 0.7093891 0.7532471 0.9266587  1.0000000
## 6 lv =~ q1 0.5537083 0.57086815 0.5732166 0.5475714 0.8958929  0.9814658
##   boot_difference random_difference boot_index_difference
## 1    -0.044410454      -0.002390463                 FALSE
## 2    -0.202508484      -0.087934027                 FALSE
## 3     0.089025927       0.085220565                 FALSE
## 4     0.359865463      -0.124796846                 FALSE
## 5    -0.004485377      -0.043857947                 FALSE
## 6    -0.017159815       0.025645271                 FALSE
##   random_index_difference
## 1                    TRUE
## 2                    TRUE
## 3                    TRUE
## 4                    TRUE
```

```
## 5                    TRUE
## 6                    TRUE
```

Next, the `boot_summary` includes a summarized form of the bootstrapped results from separated by bootstrapping versus random and invariant/non-invariant. The d_s for between groups Cohen's d is shown below, and the non-central confidence interval is included. Effect sizes are only calculated when the number of bootstrapped estimates is at least 10% of the data - therefore, you would not receive effect sizes with almost no bootstrapped runs. This dataframe should be used to determine which parameter may be different and at what size between groups in a replication of the study.

```
boot.partial.invariant$boot_summary %>%
  dplyr::select(term, d_boot, d_random)
```

```
## # A tibble: 10 x 4
## # Groups:   term, invariant [10]
##    invariant term       d_boot d_random
##    <lgl>     <chr>       <dbl>    <dbl>
##  1 FALSE     lv =~ q1 -0.0299   0.0583
##  2 TRUE      lv =~ q1  0.0337   0.0116
##  3 FALSE     lv =~ q2 -0.0326   0.0933
##  4 TRUE      lv =~ q2  0.146    0.0309
##  5 FALSE     lv =~ q3 -0.0463   0.113
##  6 TRUE      lv =~ q3 -0.148    0.0743
##  7 FALSE     lv =~ q4  0.00785 -0.0668
##  8 TRUE      lv =~ q4 -0.0157   0.0389
##  9 FALSE     lv =~ q5 -0.00129 -0.169
## 10 TRUE      lv =~ q5  0.122   -0.00853
```

The `boot_effects` table creates a summary similar to the overall model replication table based on the proportion of runs that were considered invariant versus not for each parameter. Note that the effects match the overall results, such that simulated invariant data appears to still show the likelihood that loadings may not replicate in a similar dataset.

```
boot.partial.invariant$boot_effects
```

```
## # A tibble: 5 x 7
##   term     non_invariant random_non_invariant h_nmi  h_mi h_nmi_p h_mi_p
##   <chr>            <dbl>                <dbl> <dbl> <dbl>   <dbl>  <dbl>
## 1 lv =~ q1         0.853                0.236  1.34 -1.34   0.427 -0.427
## 2 lv =~ q2         0.858                0.237  1.35 -1.35   0.430 -0.430
## 3 lv =~ q3         0.851                0.23   1.35 -1.35   0.429 -0.429
## 4 lv =~ q4         0.84                 0.229  1.32 -1.32   0.420 -0.420
## 5 lv =~ q5         0.819                0.237  1.25 -1.25   0.397 -0.397
```

Plots of the results from dataframes can be found within the `bootstrap_partial()` function. Figure 12 shows the difference between parameters for groups in the bootstrapped and randomly assigned group runs. Figure 13 shows the density plot of the estimates for each group organized by bootstrapped and randomly assigned groups and the invariance decision for each bootstrapped run. Last, Figure 14 indicates the d_s value between groups with an indication of the number of data points in each estimate (i.e., dot size). These visualizations should allow a researcher to understand the likelihood of replication for each parameter, as well as the potential size of the differences. Therefore, one could indicate a specific smallest effect size of interest, rather than a invariance cut-off rule of thumb when planning a replication or registered report.

Simulated Results

Figure 15 shows the effect size differences within large loadings simulations. The results demonstrate that most of the loadings were considered non-invariant in the bootstrapped models (while holdings all others equal). This result is partially due to simulating very good data, so small changes in loadings results in a drop in fit for our chosen invariance index. However, we can use this graph to show that question four shows a possible effect size ranging from -0.07 to 0.13. The h_{nmi_p} value for question four was 0.27, representing about a quarter of a possible total effect. Last, the density plot in Figure 16 shows the separation of the two different groups loadings in item four, thus, illustrating group differences in the findings for their loadings. Each of the other combination of plots can be found in the supplemental materials.

An Example Analysis

Aiena et al. (2014) examined the RS-14 (Wagnild, 2009) exploring the factor structure of the Resiliency Scale in a clinical sample receiving treatment services and a college student sample. Measurement invariance was calculated for differences separately for these samples for gender and race finding a partially invariant models with a few item intercepts or residuals that differed between groups. Aiena et al. (2014) did not compare the clinical to the student sample for measurement invariance, and it is reasonable to expect potential differences in these two populations. This example will demonstrate the procedure for researchers who wish to use partial invariance steps and how to interpret real, messy data.

```
# load the data, it is called DF
load("manu_data/RS14.Rdata")


# build the one-factor model
model.rs <- "RS =~ RS1+RS2+RS3+RS4+RS5+RS6+RS7+RS8+RS9+RS10+RS11+RS12+RS13+RS14"
# run the multi-group CFA
```

```
results.rs <- mgcfa(
  model = model.rs,
  data = DF,
  group = "sample",
  group.equal = c("loadings", "intercepts", "residuals")
  )
```

```
# how to get results in table
results.rs$model_fit %>%
        dplyr::select(model, AIC, BIC, cfi, tli, rmsea, srmr)
```

Table 1 indicates the results after running the one-factor model. There are several guidelines for assessing assessing a degradation in model fit (Cao & Liang, 2022; Cheung & Rensvold, 2002; Counsell et al., 2020; Jin, 2020; Putnick & Bornstein, 2016) but for the purposes of this illustration ΔCFI $>$ .01 will be used. Table 1 indicates that fit was degraded when the constraint on equal item intercepts was added. The code below provides an example of testing each item individually by relaxing the constraints and recalculating the CFI. If these Items bring the CFI value back up to ΔCFI $<=$ .01 from the metric model, then the model would be considering partially invariant at the scalar level. It seems unlikely that the residuals will show invariance, if partial scalar invariance can be found, as the drop in fit is quite large.

```
partial.rs <-
  partial_mi(
    saved_model = results.rs$invariance_models$model.intercepts,
    data = DF,
    model = model.rs,
    group = "sample",
    # be sure to do only up to the step you are interested in
    group.equal = c("loadings", "intercepts"),
```

```
    partial_step = "intercepts")


partial.rs$fit_table %>%
  dplyr::select(free.parameter, cfi)
```

```
## # A tibble: 15 x 2
##    free.parameter cfi
##    <chr>          <lvn.vctr>
##  1 "RS1 ~1 "      0.9116914
##  2 "RS2 ~1 "      0.9129976
##  3 "RS3 ~1 "      0.9117235
##  4 "RS4 ~1 "      0.9111212
##  5 "RS5 ~1 "      0.9126742
##  6 "RS6 ~1 "      0.9133618
##  7 "RS7 ~1 "      0.9139287
##  8 "RS8 ~1 "      0.9111397
##  9 "RS9 ~1 "      0.9119702
## 10 "RS10 ~1 "     0.9118309
## 11 "RS11 ~1 "     0.9110574
## 12 "RS12 ~1 "     0.9112309
## 13 "RS13 ~1 "     0.9112367
## 14 "RS14 ~1 "     0.9112015
## 15 "RS ~1 "       0.9108805
```

The output indicates that RS6 and RS7 are potential items that could be relaxed to improve model fit and create a partial scalar invariant model (i.e., by picking the largest CFI values). The code below show to check the addition of these items, which are added one at a time. You use the `group.partial` open to "relax" or freely estimate that

parameter for each group separately.

```
# run the partially invariant model with group.partial
partial.rs.1 <- mgcfa(model = model.rs,
                      data = DF,
                      group = "sample",
                      group.equal = c("loadings", "intercepts"),
                      group.partial = c("RS7~1"),
                      meanstructure = TRUE)


# examine the loadings
partial.rs.1$model_coef %>%
  filter(term == "RS7 ~1 ") %>%
  filter(model == "intercepts") %>%
  dplyr::select(term, group, estimate, std.error)
```

```
## # A tibble: 2 x 4
##   term      group estimate std.error
##   <chr>     <int>    <dbl>     <dbl>
## 1 "RS7 ~1 "     1     4.95    0.0580
## 2 "RS7 ~1 "     2     4.49    0.0529
```

```
# examine the fit indices
partial.rs.1$model_fit %>%
  filter(model == "intercepts") %>%
  dplyr::select(AIC, BIC, cfi, tli, rmsea, srmr)
```

```
## # A tibble: 1 x 6
##       AIC     BIC   cfi   tli rmsea   srmr
##     <dbl>   <dbl> <dbl> <dbl> <dbl>  <dbl>
```

```
## 1 122454. 122804. 0.914 0.912 0.102 0.0502
```

```
# effect size model
lavaan_dmacs(partial.rs.1$invariance_models$model.intercepts, "Clinical")$DMACS[7]
```

```
##       RS7
## 0.282302
```

By examining our estimates, we can see that item seven on the RS-14 is estimated at nearly 5 points for the clinical sample, while the student sample has a lower mean around 4.5 points. Generally, students show higher means on the items of the RS14, but when all loadings and other intercepts are constrained to be equal, and this one item is relaxed, this pattern flips so that clinical groups show higher item intercepts. Given the scale is a 1-7 Likert type scale, .5 a point represents a potentially sizable change on the scale. Item seven covers perseverance after hardship, and all items can be found in the user manual for the scale at www.resiliencecenter.com. The effect size from d_{MACS} suggests a small to medium effect, 0.28. In this next code section, we repeat this process for the RS6, as the CFI for our model with only RS7 does not achieve the levels of partial invariance for our ΔCFI criterion (i.e., <= .01 downward change in fit: metric CFI = .925, partial scalar CFI = .914). See Figure 17 for the difference between item intercepts and latent means.

```
# add the second intercept
partial.rs.2 <- mgcfa(model = model.rs,
                      data = DF,
                      group = "sample",
                      group.equal = c("loadings", "intercepts"),
                      group.partial = c("RS7~1", "RS6~1"),
                      meanstructure = TRUE)


# examine the loadings
```

```
partial.rs.2$model_coef %>%
  filter(term == "RS6 ~1 ") %>%
  filter(model == "intercepts") %>%
  dplyr::select(term, group, estimate, std.error)
```

```
## # A tibble: 2 x 4
##   term      group estimate std.error
##   <chr>     <int>    <dbl>     <dbl>
## 1 "RS6 ~1 "     1     5.00    0.0605
## 2 "RS6 ~1 "     2     4.54    0.0533
```

```
# examine the fit indices
partial.rs.2$model_fit %>%
  filter(model == "intercepts") %>%
  dplyr::select(AIC, BIC, cfi, tli, rmsea, srmr)
```

```
## # A tibble: 1 x 6
##       AIC     BIC   cfi   tli rmsea   srmr
##     <dbl>   <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1 122363. 122719. 0.917 0.915 0.100 0.0488
```

```
# effect size model
lavaan_dmacs(partial.rs.2$invariance_models$model.intercepts, "Clinical")$DMACS[6]
```

```
##       RS6
## 0.2796334
```

Again, we see about a half-point difference between our clinical and student samples for item 6, which is about drive to achieve. The CFI for this model does meet the requirements for partial invariance, .917. The effect size is approximately the same at

649 $d_{MACS} = 0.28$. See Figure 18 shows the difference between item intercepts and latent
650 means.

```
# plot the image for RS7
plot_mi(
  data_coef = partial.rs.2$model_coef,
  model_step = "intercepts",
  item_name = "RS7",
  x_limits = c(-1,1),
  y_limits = c(min(DF$RS7), max(DF$RS7)),
  conf.level = .95,
  model_results = partial.rs.2$invariance_models$model.intercepts,
  # which latent variable do you want
  lv_name = "RS"
)$complete
```

```
# plot the image for RS6
plot_mi(
  data_coef = partial.rs.2$model_coef,
  model_step = "intercepts",
  item_name = "RS6",
  x_limits = c(-1,1),
  y_limits = c(min(DF$RS6), max(DF$RS6)),
  conf.level = .95,
  model_results = partial.rs.2$invariance_models$model.intercepts,
  # which latent variable do you want
  lv_name = "RS"
)$complete
```

Next, we would examine our replication potential for this model. Given our current results, we may not expect our intercepts to replicate. Given the order of `group.equal`, the boot function will select the first non-invariant step in the calculation of the effect size for potential replication. In our output, we do not see a loadings effect size, and this result occurs when *none* of the bootstrapped or random results are non-invariant. Therefore, we would expect the loadings to replicate (and the effect size would be 0 difference between bootstrapped and random, both showing invariance). The intercepts show a large (i.e., close to the max possible value) non-invariant effect, and therefore, we should not expect this model to show invariance in a replication.

```
boot.model.rs <-
  bootstrap_rr(
    saved_configural = results.rs$model_configural,
    data = DF,
    model = model.rs,
    group = "sample",
    nboot = 1000,
    invariance_index = "cfi",
    invariance_rule = .01,
    group.equal = c("loadings", "intercepts", "residuals")
)
```

```
boot.model.rs
```

```
## # A tibble: 2 x 7
##   model      non_invariant random_non_invariant  h_nmi    h_mi h_nmi_p  h_mi_p
##   <chr>              <dbl>                <dbl>  <dbl>   <dbl>   <dbl>   <dbl>
## 1 intercepts         0.998                    0 3.05   -3.05    0.972  -0.972
## 2 residuals          0.002                    0 0.0895 -0.0895  0.0285 -0.0285
```

Next, we would examine the strength of the effects of replication on each parameter at the intercept level. By examining 2, it is clear that most of the item means are unlikely to replicate, even though two particular items can be used to create partial invariance. Figures 19 and 20 display the three plots provided in the `bootstrap_partial()` function. In general, we should expect $M_D = 0.23$ when items are invariant and $M_D = 0.26$ when items are not invariant. The effect size of non-invariant items ranges from 0.43 to 0.62.

The density plot shown at the bottom of Figure 19 illustrates the likely reasons for the differences found in the top plots. It appears that many items show a bimodal distribution within group 1 (Clinical Sample) and when items are invariant, the intercept averages to the same intercept as group 2 (Student Sample). In non-invariant estimates, the same bimodal distributions are found, but they are more extreme than the student samples, and therefore, item show different averages due to the presence of two separate means of data. Further, some items also appear to show two separate student item averages within the data. This result suggests that it would be fruitful to understand a potential predictor of these differences or other confounding variable that separates these samples, creating differences in item averages.

In summary, if one were planning a replication, the prediction would be that item intercepts would likely not replicate, with a large effect size (i.e., it is easy to judge h_{nmi_p} close to the max of one as large). While this study found partial invariance by relaxing constraints on two individual items, bootstrapped partial invariance indicates that any item could potentially be problematic with an effect size averaging $d \sim 0.50$ difference in means. While d_{MACS} values represented a "small" effect based on previous publications, this effect may be muted by examining both loadings and intercepts. The results here suggest that the effect is driven by intercepts. The overall average score on items is high: $M_M = 5.04$ ($M_{SD} = 1.72$). Given the mean standard deviation, a $d \sim 0.50$ represents 0.86 or nearly one whole point on the scale. A researcher could decide that at least $d = 0.33$ or at least a third of a

standard deviation would be an important change and set that as their smallest effect size of interest for invariance. Further, a newly planned study should investigate what variables may predict when and why samples separate into bimodal representations for item means.

```
boot.partial.rs <-
  bootstrap_partial(
    saved_model = results.rs$invariance_models$model.intercepts,
    data = DF,
    model = model.rs,
    group = "sample",
    nboot = 1000,
    invariance_index = "cfi",
    invariance_rule = .01,
    invariance_compare = fitmeasures(results.rs$invariance_models$model.loadings, "cfi")
    partial_step = "intercepts",
    group.equal = c("loadings", "intercepts")
)
```

Discussion

In this tutorial, we examined how to use multiple tools to examine measurement invariance and its potential replication. Model fit comparisons and statistics can be paired with the proposed effect size measures, and a visualization to examine individual items and the overall latent mean scores. The impact of potential replication was estimated on the overall model and the individual parameters. Using real data, the effect of two non-invariant item intercepts was examined and visualized. This tutorial manuscript has provided a concrete way to plan for pre-registration and/or registered reports. Researchers could simulate results based on published or previously collected data to determine the likelihood and size of potential replication. They could plan and pre-register a smallest effect of interest. For example, we may determine that an h_{nmi_p} value above .20 represents

an important level of non-invariance for our model overall, while $h_{nmi_p} > .30$ for any individual parameter warrant caution against invariance for groups. Others have begun to discuss the importance of focusing on effects in the scale of the data and their practical importance (Anvari & Lakens, 2021; Cumming, 2012).

From the example, our interpretation may be that the difference between group's latent means is large, as a 0.72 change on a 7 point scale is approximately 10% more resiliency for students when compared to the clinical sample. Practically, 10% in resiliency for an area of the United States (Mississippi) often hit with natural disasters (hurricanes, tornadoes, floods) and high levels of poverty would be very important. Even the smaller difference of .5 point on each individual item could translate into increases in resiliency, and these results may elucidate avenues for further exploration into areas of focus within resiliency, given the items.

What do the results of a study on measurement invariance with these results tell us about replication, generalizability, and validity overall? If a researcher decides their effects are large, they should likely caution against suggesting that these scores are directly comparable without weighting or other adjustment. Let's consider a scenario wherein the change metric between models picked (i.e., ΔCFI, ΔRMSEA) indicates a "significant" change in model fit. However, if both the effect size and a visual inspection of the invariance indicates a small difference, we may decide to lessen the practical importance of those results, much like "just significant" p-values with small effect sizes are treated now. Given that the goal of measurement invariance is to compare *estimates*, we should expect some differences across samples due to the nature of sampling and estimation. It may be that many of the published models presented represent these effects - small variations between groups due to sampling error or other small crud - but do not represent a fundamental problem with the measurement or generalizability of the results.

730 **References**

731 Aiena, B. J., Baczwaski, B. J., Schulenberg, S. E., & Buchanan, E. M. (2014). Measuring
732 Resilience With the RS–14: A Tale of Two Samples. *Journal of Personality*
733 *Assessment*, *97*(3), 291–300. https://doi.org/10.1080/00223891.2014.951445
734 Akaike, H. (1998). *Information theory and an extension of the maximum likelihood*
735 *principle* (E. Parzen, K. Tanabe, & G. Kitagawa, Eds.; pp. 199–213). Springer New
736 York. http://link.springer.com/10.1007/978-1-4612-1694-0_15
737 Anvari, F., & Lakens, D. (2021). Using anchor-based methods to determine the smallest
738 effect size of interest. *Journal of Experimental Social Psychology*, *96*, 104159.
739 https://doi.org/10.1016/j.jesp.2021.104159
740 Barry, A. E., Chaney, B., Piazza-Gardner, A. K., & Chavarria, E. A. (2014). Validity and
741 Reliability Reporting Practices in the Field of Health Education and Behavior: A
742 Review of Seven Journals. *Health Education & Behavior*, *41*(1), 12–18.
743 https://doi.org/10.1177/1090198113483139
744 Beaujean, A. A. (2014). *Latent variable modeling using r: A step by step guide*.
745 Routledge/Taylor & Francis Group.
746 Bentler, P. M. (1990). Comparative fit indexes in structural models. *Psychological Bulletin*,
747 *107*(2), 238–246. https://doi.org/10.1037/0033-2909.107.2.238
748 Bentler, P. M. (1995). *EQS structural equations program manual*.
749 Boker, S., Neale, M., Maes, H., Wilde, M., Spiegel, M., Brick, T., Spies, J., Estabrook, R.,
750 Kenny, S., Bates, T., Mehta, P., & Fox, J. (2011). OpenMx: An Open Source Extended
751 Structural Equation Modeling Framework. *Psychometrika*, *76*(2), 306–317.
752 https://doi.org/10.1007/s11336-010-9200-6
753 Brown, T. A. (2015). *Confirmatory factor analysis for applied research* (Second edition).
754 The Guilford Press.
755 Byrne, B. M., Shavelson, R. J., & Muthén, B. (1989). Testing for the equivalence of factor
756 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

Cao, C., & Liang, X. (2022). The impact of model size on the sensitivity of fit measures in measurement invariance testing. *Structural Equation Modeling: A Multidisciplinary Journal*, *29*(5), 744–754. https://doi.org/10.1080/10705511.2022.2056893

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.

Counsell, A., Cribbie, R. A., & Flora, D. B. (2020). Evaluating equivalence testing methods for measurement invariance. *Multivariate Behavioral Research*, *55*(2), 312–328. https://doi.org/10.1080/00273171.2019.1633617

Cumming, G. (2012). *Understanding the new statistics: Effect sizes, confidence intervals, and meta-analysis*. Routledge.

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

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

Epskamp, S. (2022). *semPlot: Path diagrams and visual analysis of various SEM packages' output*. https://CRAN.R-project.org/package=semPlot

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

Hobson, H. (2019). Registered reports are an ally to early career researchers. *Nature*

Human Behaviour, *3*(10), 1010–1010. https://doi.org/10.1038/s41562-019-0701-8

Jin, Y. (2020). A note on the cutoff values of alternative fit indices to evaluate measurement invariance for ESEM models. *International Journal of Behavioral Development*, *44*(2), 166–174. https://doi.org/10.1177/0165025419866911

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

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

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

Mayo-Wilson, E., Grant, S., Supplee, L., Kianersi, S., Amin, A., DeHaven, A., & Mellor, D. (2021). Evaluating implementation of the transparency and openness promotion (TOP) guidelines: The TRUST process for rating journal policies, procedures, and practices. *Research Integrity and Peer Review*, *6*(1), 9. https://doi.org/10.1186/s41073-021-00112-8

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

Nosek, B. A., Ebersole, C. R., DeHaven, A. C., & Mellor, D. T. (2018). The preregistration revolution. *Proceedings of the National Academy of Sciences of the United States of America*, *115*(11), 2600–2606. https://doi.org/10.1073/pnas.1708274114

Nosek, B. A., & Lakens, D. (2014). Registered Reports: A Method to Increase the Credibility of Published Results. *Social Psychology*, *45*(3), 137–141. https://doi.org/10.1027/1864-9335/a000192

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

Robinson, D., Hayes, A., Couch [aut, S., cre, Software, P., PBC, Patil, I., Chiu, D., Gomez, M., Demeshev, B., Menne, D., Nutter, B., Johnston, L., Bolker, B., Briatte, F., Arnold, J., Gabry, J., Selzer, L., Simpson, G., … Reinhart, A. (2023). *Broom: Convert statistical objects into tidy tibbles*. https://CRAN.R-project.org/package=broom

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. 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*. 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

Trent, L. R., Buchanan, E., Ebesutani, C., Ale, C. M., Heiden, L., Hight, T. L., Damon, J. D., & 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*. 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

Wagnild, G. (2009). A review of the resilience scale. *Journal of Nursing Measurement*, *17*(2), 105–113. https://doi.org/10.1891/1061-3749.17.2.105

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

Wickham, H. (2016). *ggplot2: Elegant graphics for data analysis*. Springer-Verlag New York. https://ggplot2.tidyverse.org

Wilke, C. O. (2020). *Cowplot: Streamlined plot theme and plot annotations for 'ggplot2'*. https://CRAN.R-project.org/package=cowplot

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

Table 1

Model Fit for RS-14 Example

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
Overall	126,722.491	126,888.707	0.934	0.923	0.094	0.036
Group Clinical	52,961.421	53,099.720	0.919	0.904	0.090	0.044
Group Student	69,100.985	69,254.310	0.928	0.915	0.108	0.035
Configural	122,118.406	122,617.055	0.926	0.912	0.102	0.036
loadings	122,144.532	122,566.010	0.925	0.918	0.098	0.043
intercepts	122,544.109	122,888.415	0.911	0.910	0.103	0.052
residuals	126,466.241	126,727.438	0.780	0.793	0.156	0.086

Table 2

Boot Partial Effects Results for RS-14 Intercepts

term	non_invariant	random_non_invariant	h_nmi	h_mi	h_nmi_p	h_mi_p
RS ~1	0.991	0.007	2.784	-2.784	0.886	-0.886
RS1 ~1	0.989	0.007	2.764	-2.764	0.880	-0.880
RS10 ~1	0.988	0.007	2.755	-2.755	0.877	-0.877
RS11 ~1	0.991	0.007	2.784	-2.784	0.886	-0.886
RS12 ~1	0.991	0.007	2.784	-2.784	0.886	-0.886
RS13 ~1	0.991	0.007	2.784	-2.784	0.886	-0.886
RS14 ~1	0.990	0.007	2.774	-2.774	0.883	-0.883
RS2 ~1	0.985	0.007	2.728	-2.728	0.869	-0.869
RS3 ~1	0.988	0.007	2.755	-2.755	0.877	-0.877
RS4 ~1	0.990	0.007	2.774	-2.774	0.883	-0.883
RS5 ~1	0.984	0.007	2.720	-2.720	0.866	-0.866
RS6 ~1	0.979	0.007	2.683	-2.683	0.854	-0.854
RS7 ~1	0.974	0.007	2.650	-2.650	0.844	-0.844
RS8 ~1	0.991	0.007	2.784	-2.784	0.886	-0.886
RS9 ~1	0.987	0.007	2.746	-2.746	0.874	-0.874

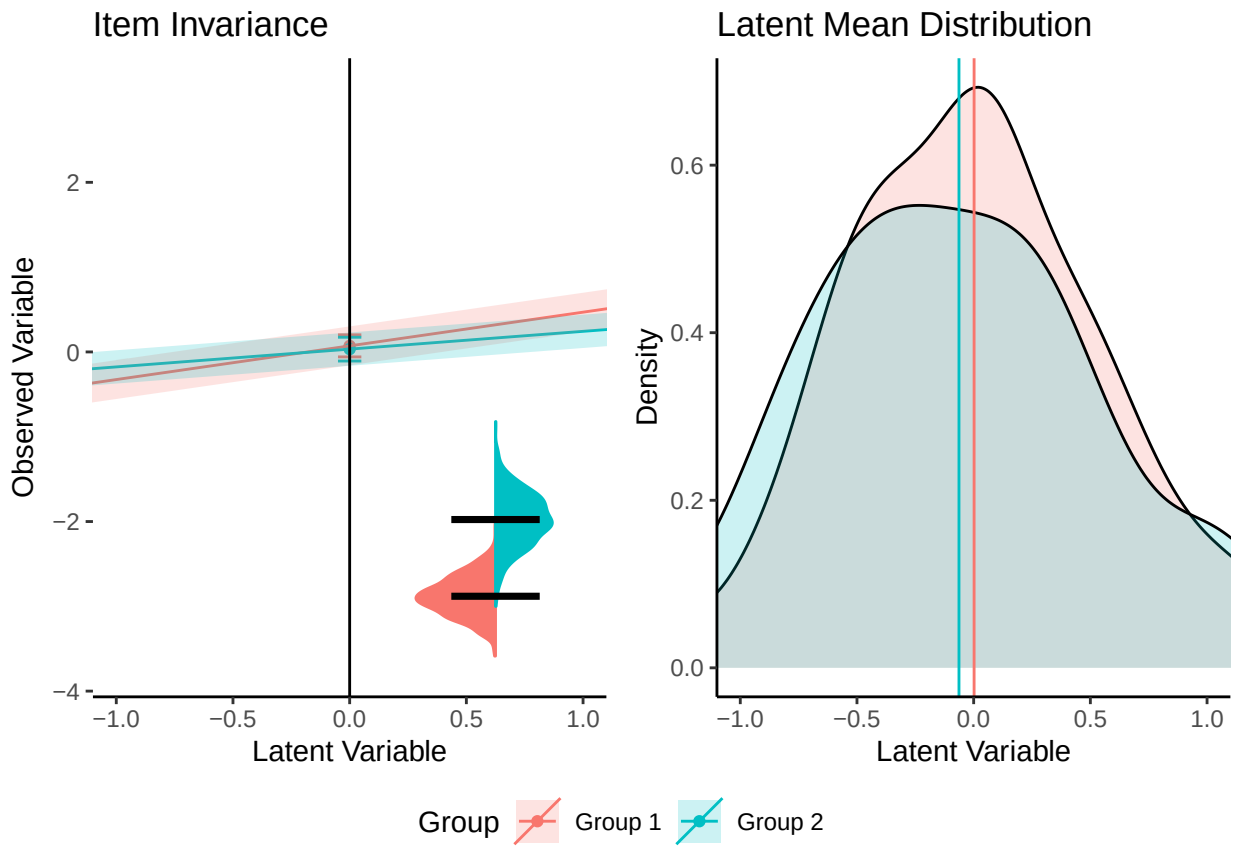


Figure 1

Invariant Model Visualization

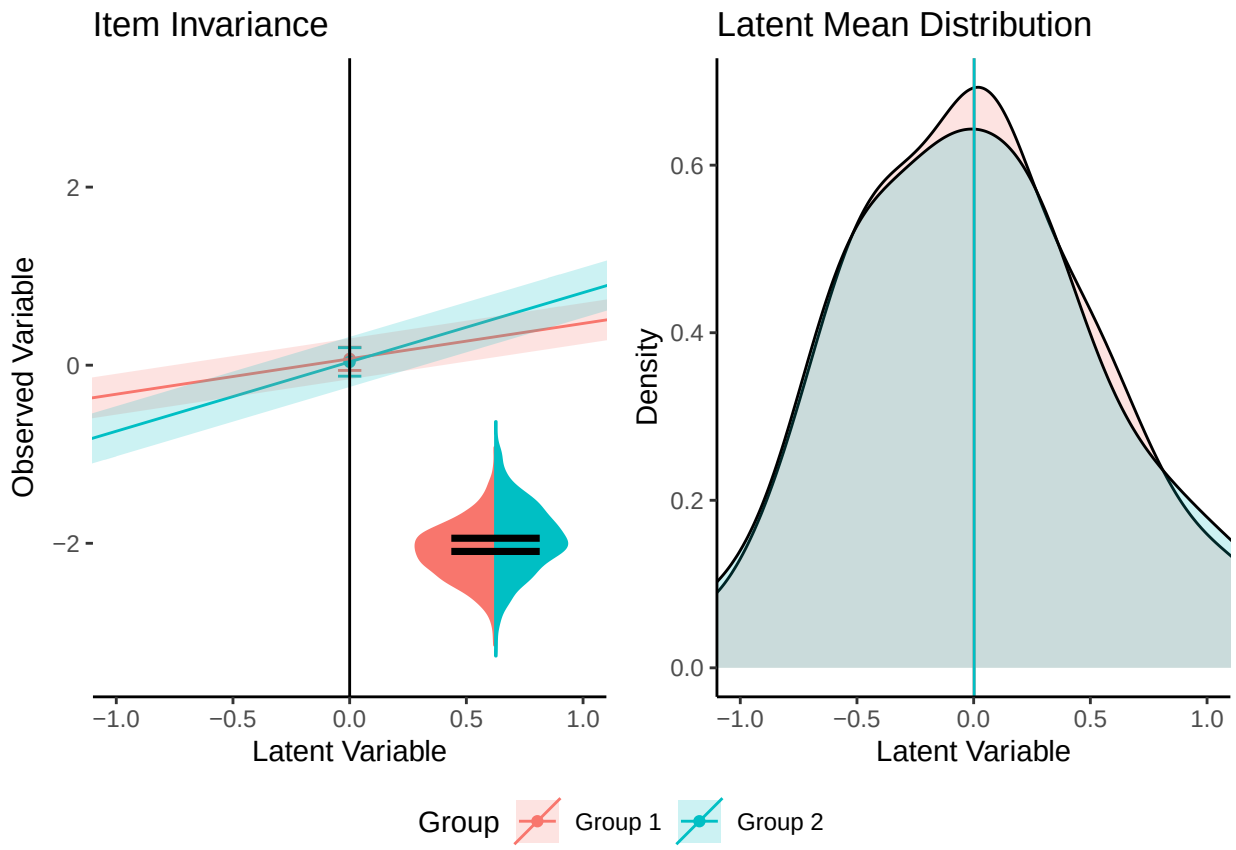


Figure 2

Small Loadings Model Visualization

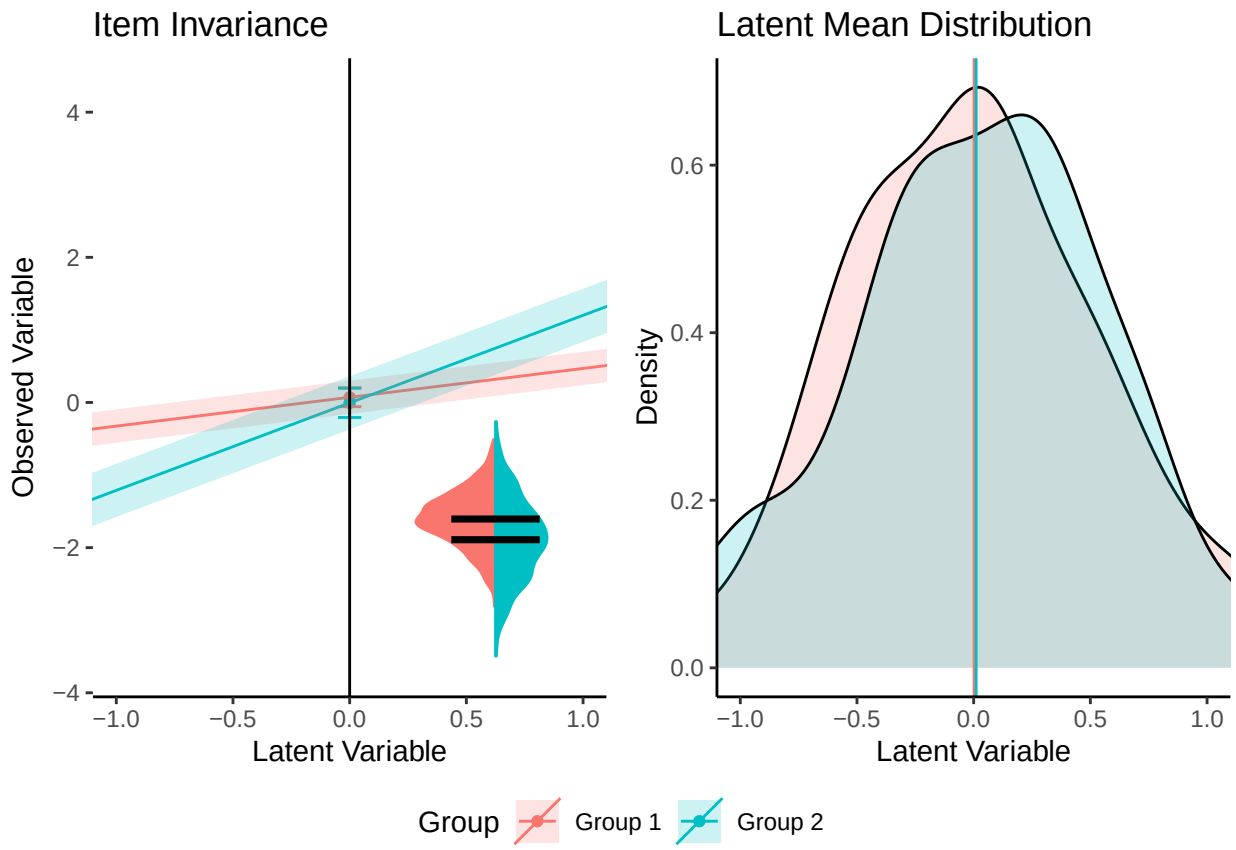


Figure 3

Medium Loadings Model Visualization

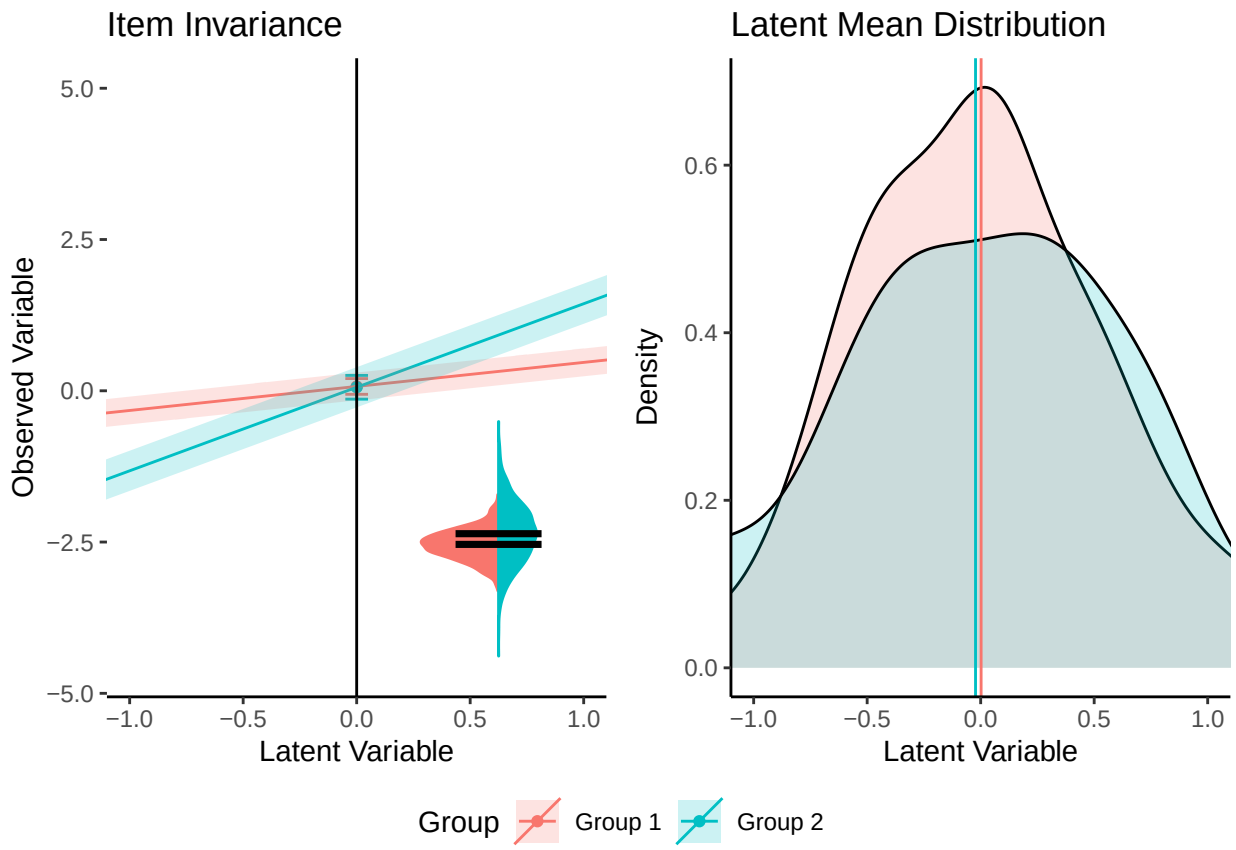


Figure 4

Large Loadings Model Visualization

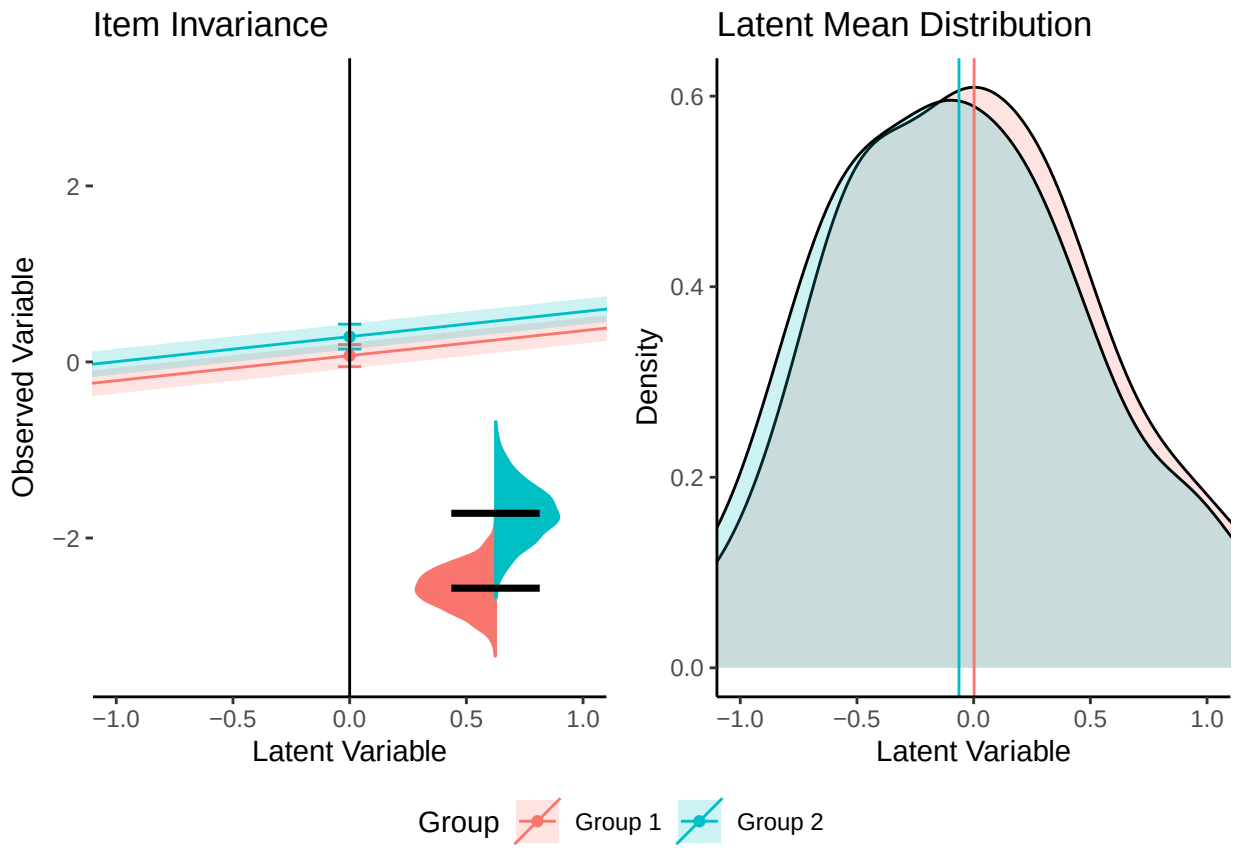


Figure 5

Small Intercepts Model Visualization

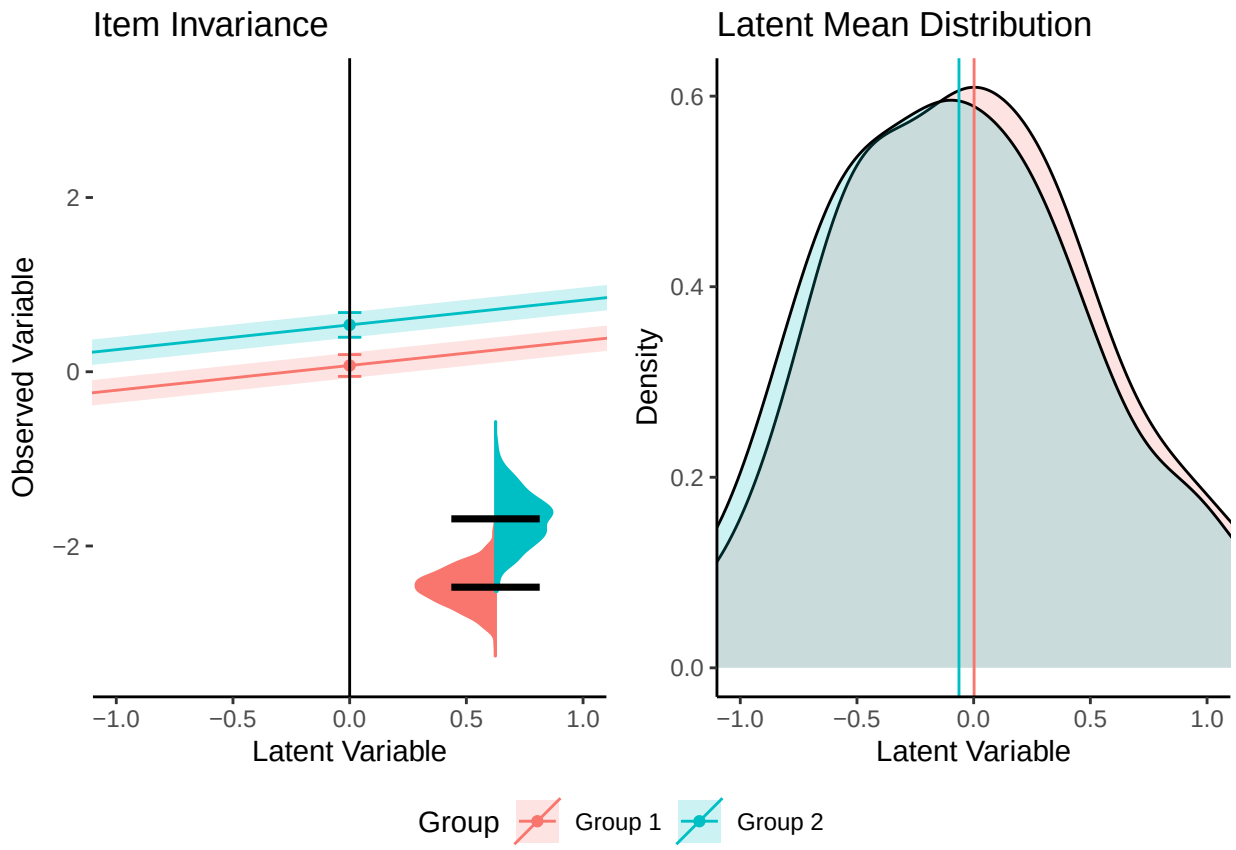


Figure 6

Medium Intercepts Model Visualization

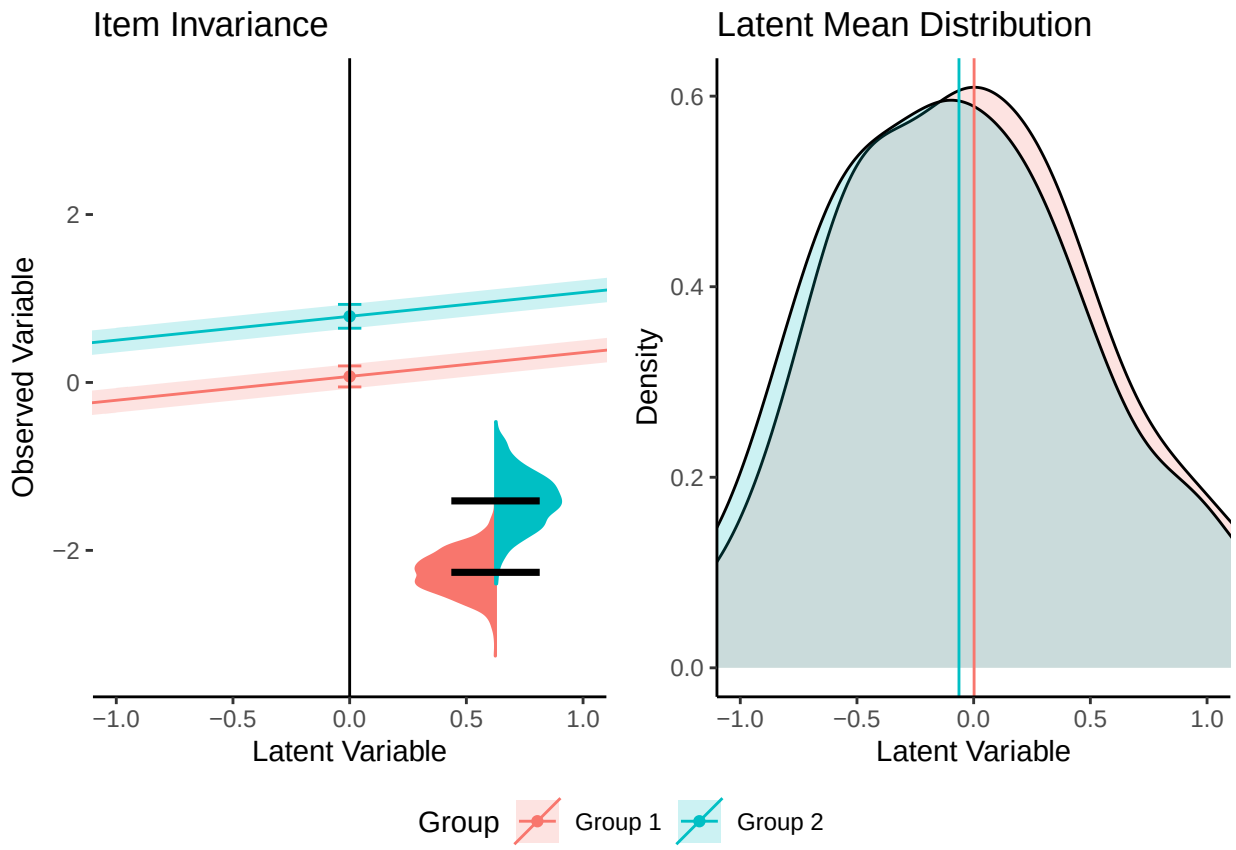


Figure 7

Large Intercepts Model Visualization

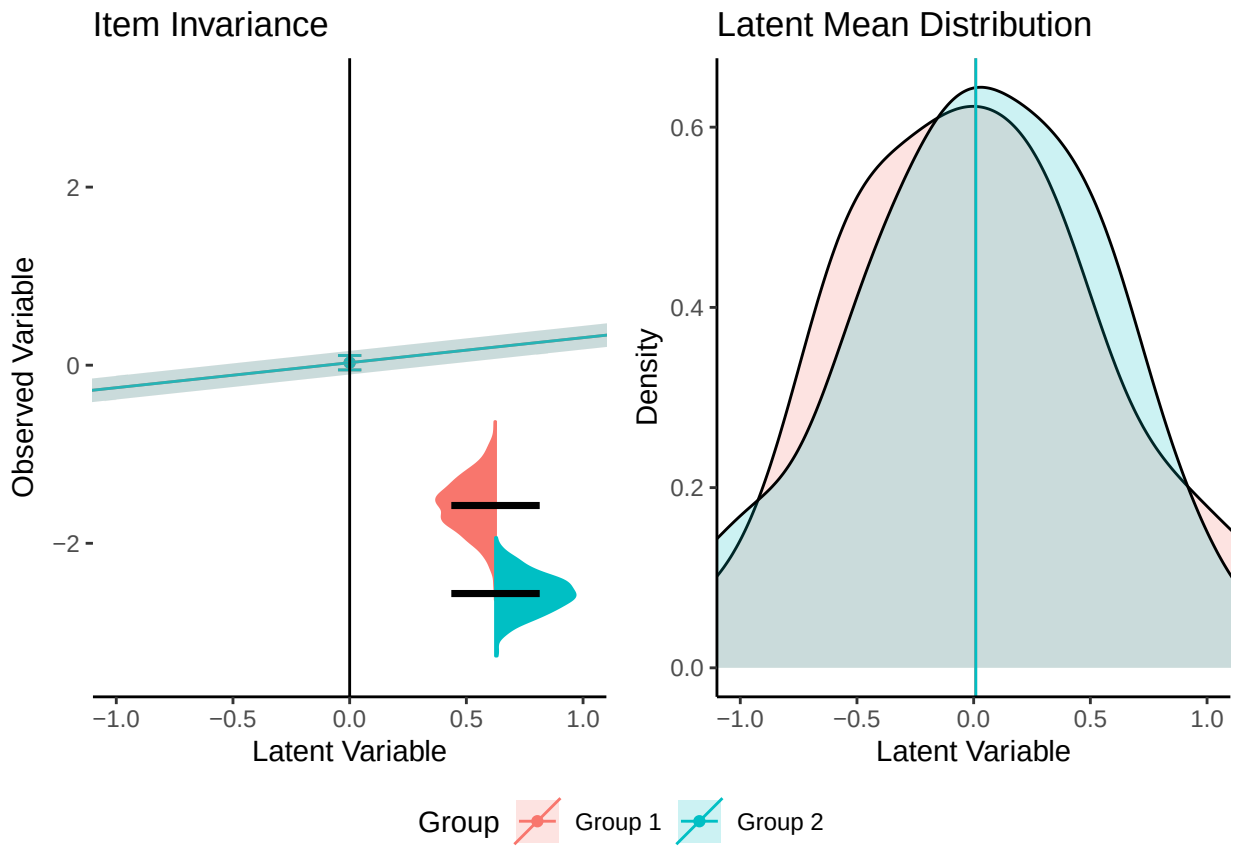


Figure 8

Small Residuals Model Visualization

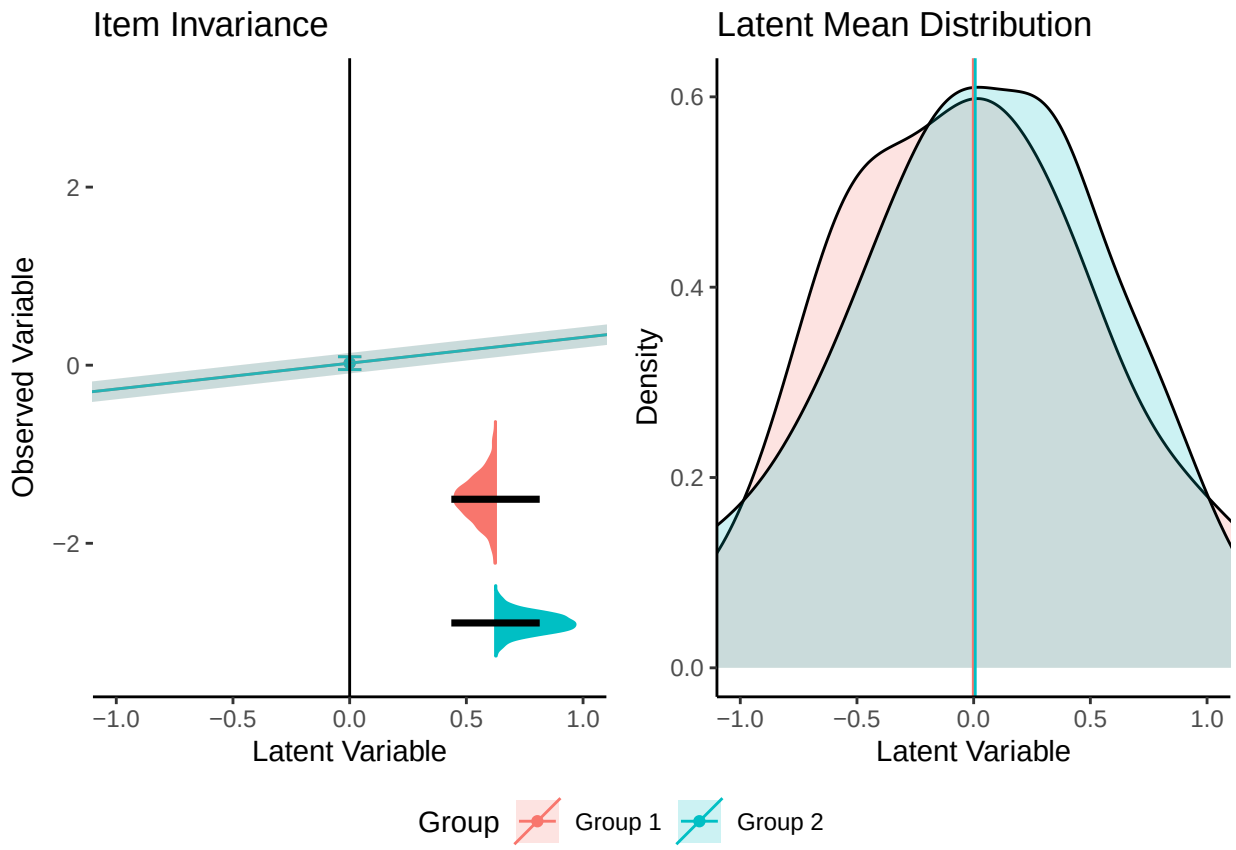


Figure 9

Medium Residuals Model Visualization

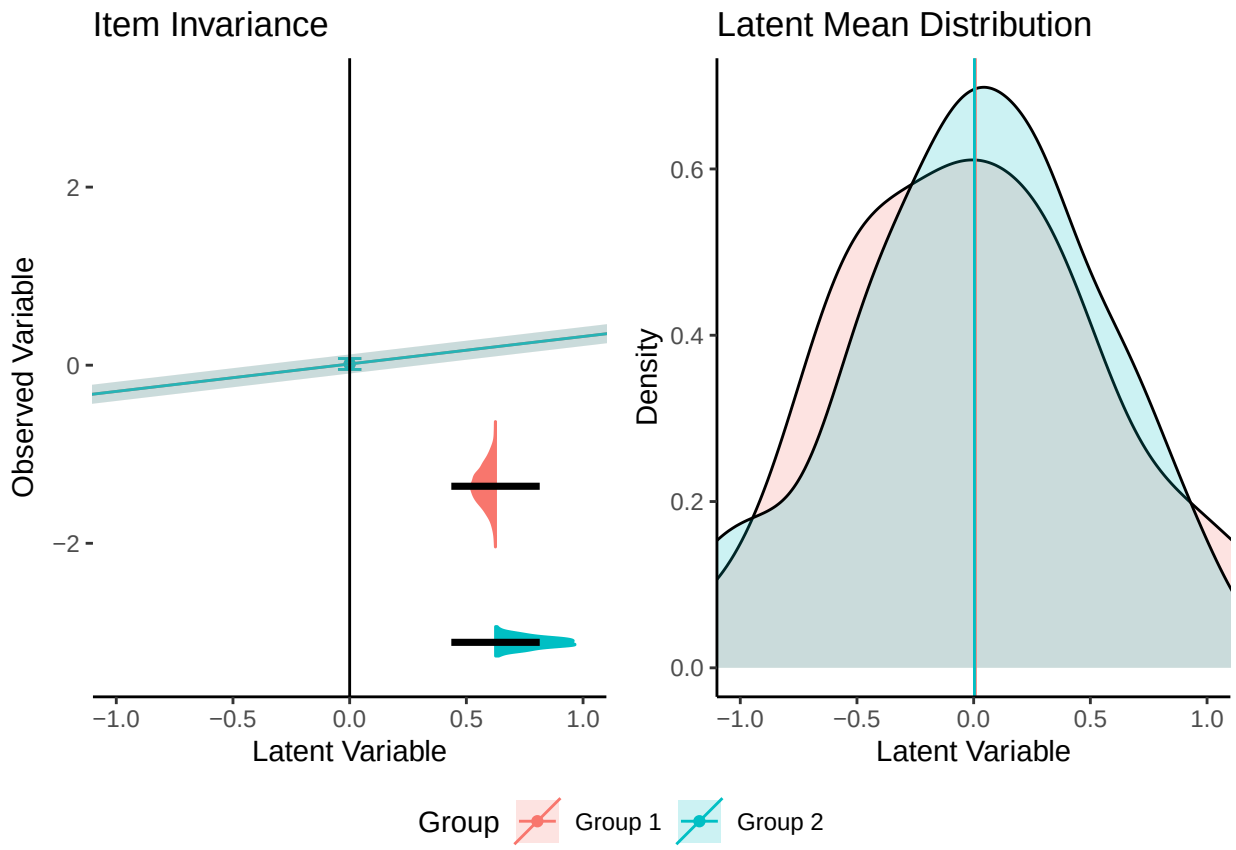


Figure 10

Large Residuals Model Visualization

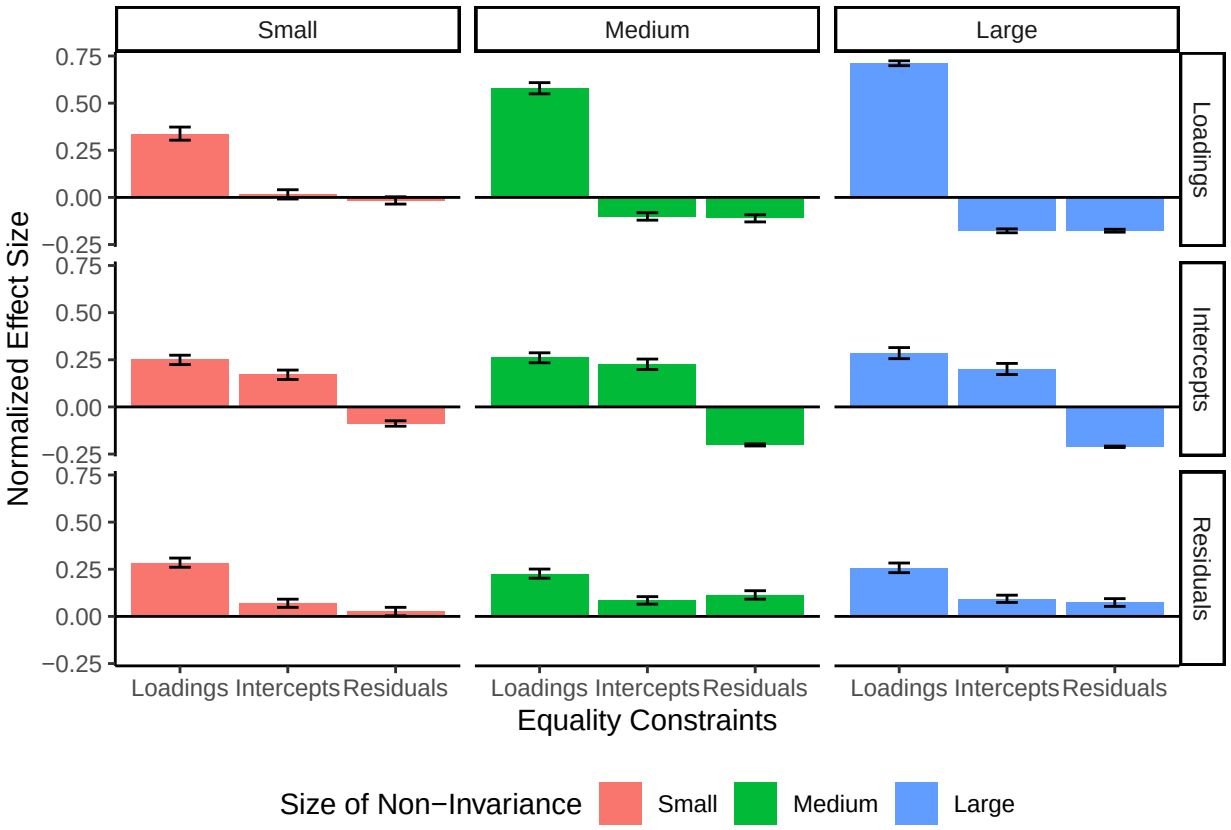


Figure 11

Visualization of the effect size of bootstrapped replication proportions on simulated data. Each panel indicates the simulated data type, colors represent the differences in the strength of the non-invariance, and the bars on the x-axis represent the effect size for the equality constraint.

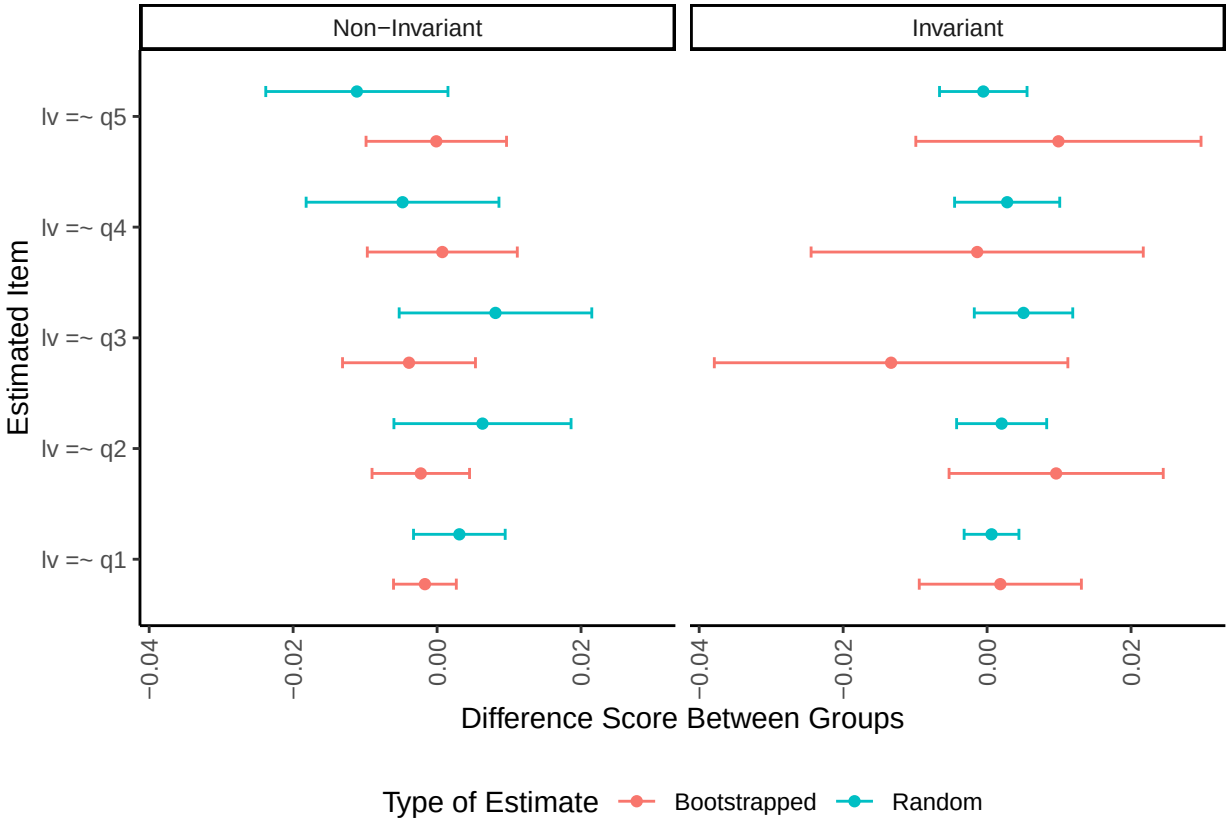


Figure 12

Visualization of the difference score between groups by parameter for invariant and non-invariant bootstrapped and randomly assigned group data.

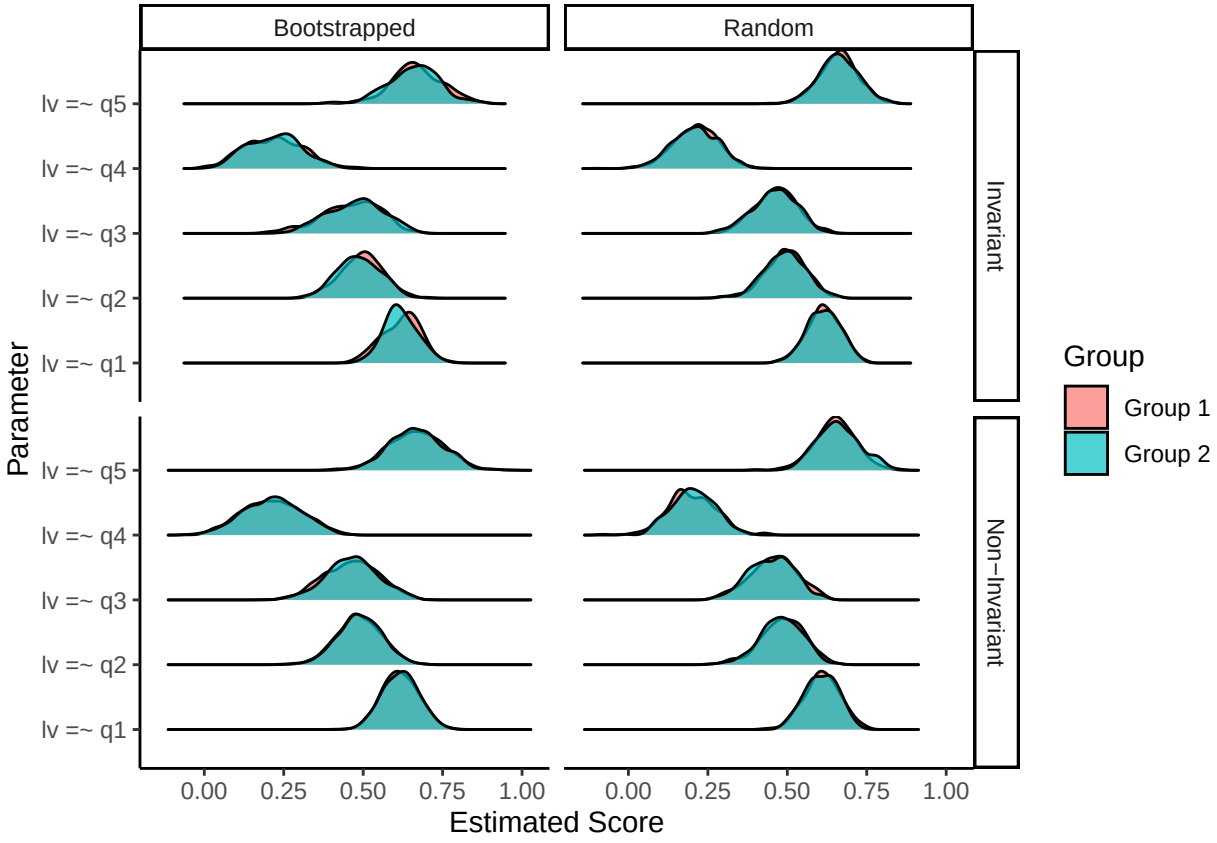


Figure 13

Visualization of the number of estimates for each group by bootstrapped and randomly assigned group runs by their invariance decision.

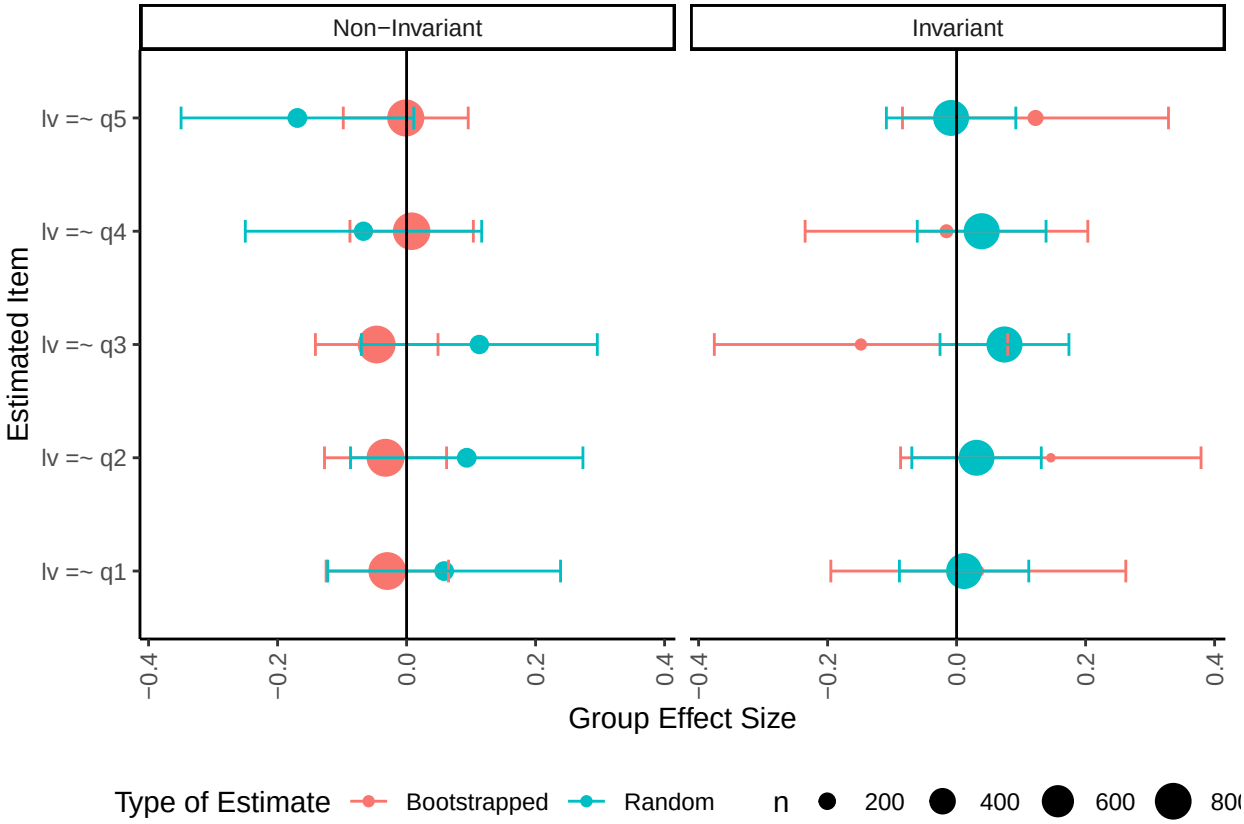


Figure 14

Visualization of effect size between groups by parameter for invariant and non-invariant bootstrapped and randomly assigned group data. The size of the dots indicate the number of data points for that estimate.

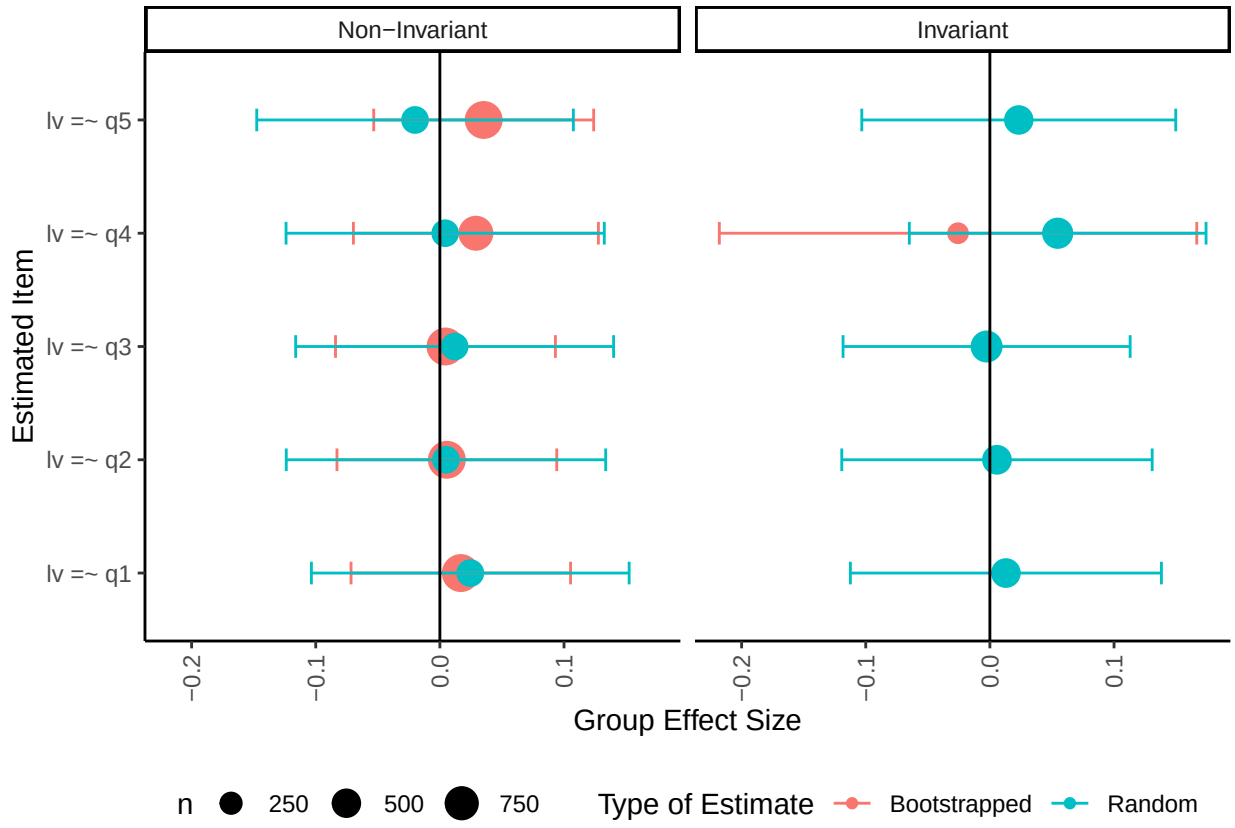


Figure 15

Bootstrapped and Random Group effect size differences in loadings for the Large Loading difference simulation. The size of the point reprensents the number of data points included in that calculation.

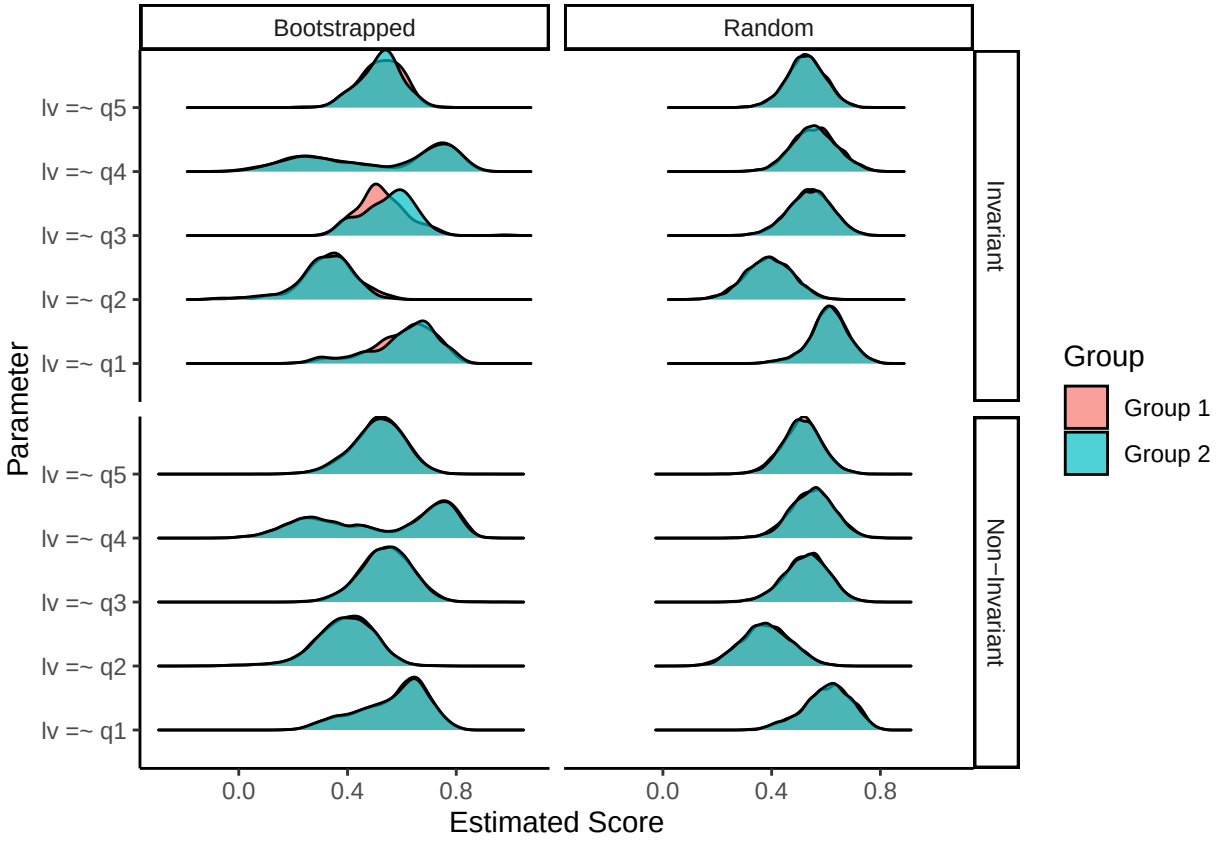


Figure 16

Bootstrapped and Random density plots for invariant and non-invariant bootstrapped partial effects examining only large loadings.

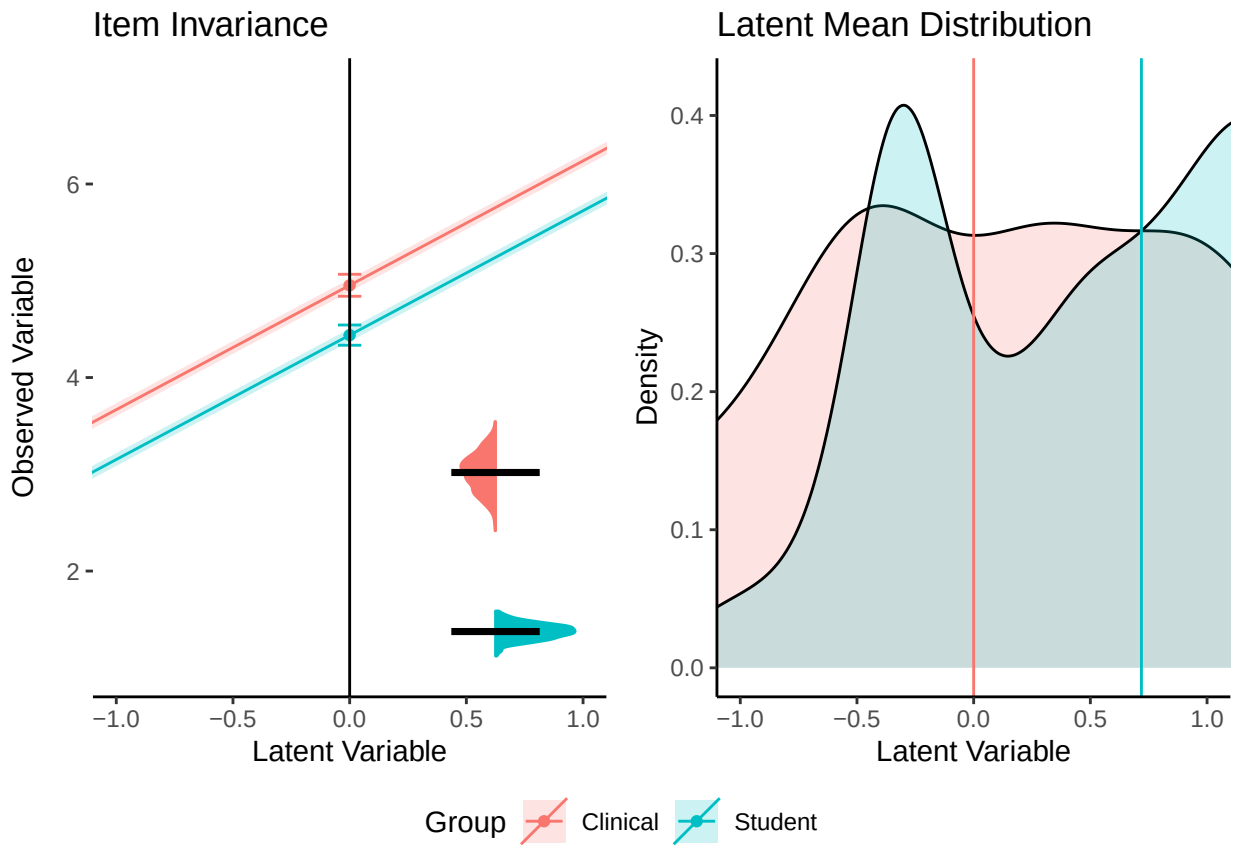


Figure 17

RS7 Item Invariance Visualization

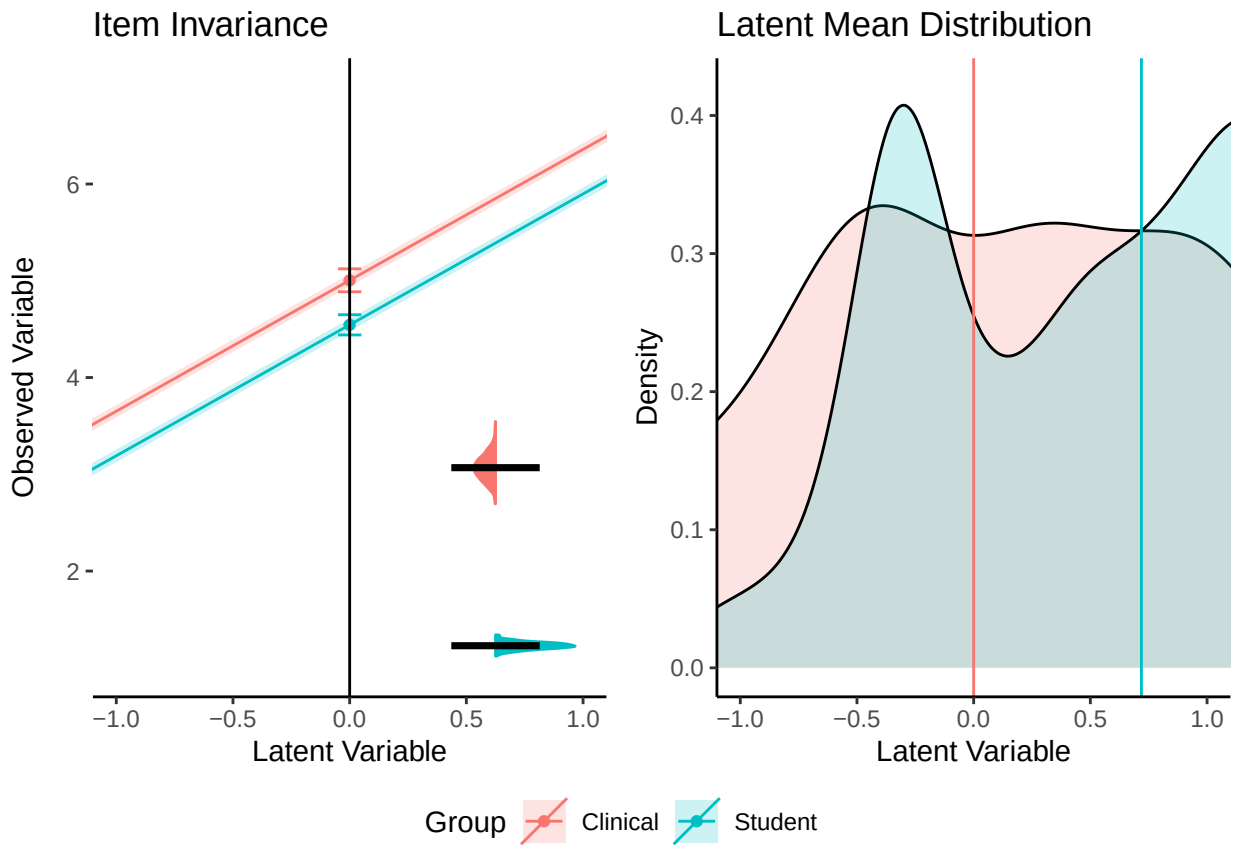


Figure 18

RS6 Item Invariance Visualization

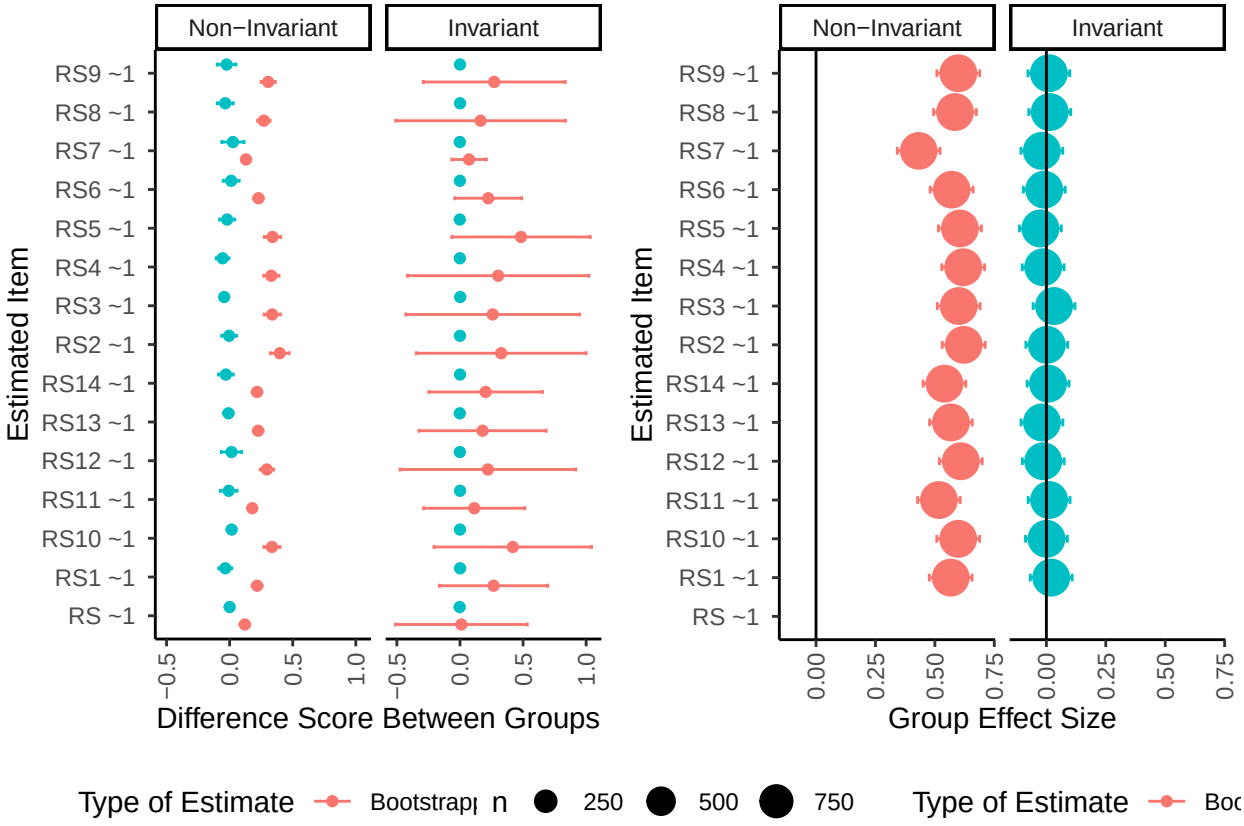


Figure 19

RS14 scale invariance for item intercepts. The left panel indicates the raw score difference between groups and items, while the right panel indicates the effect size for group differences based on invariance.

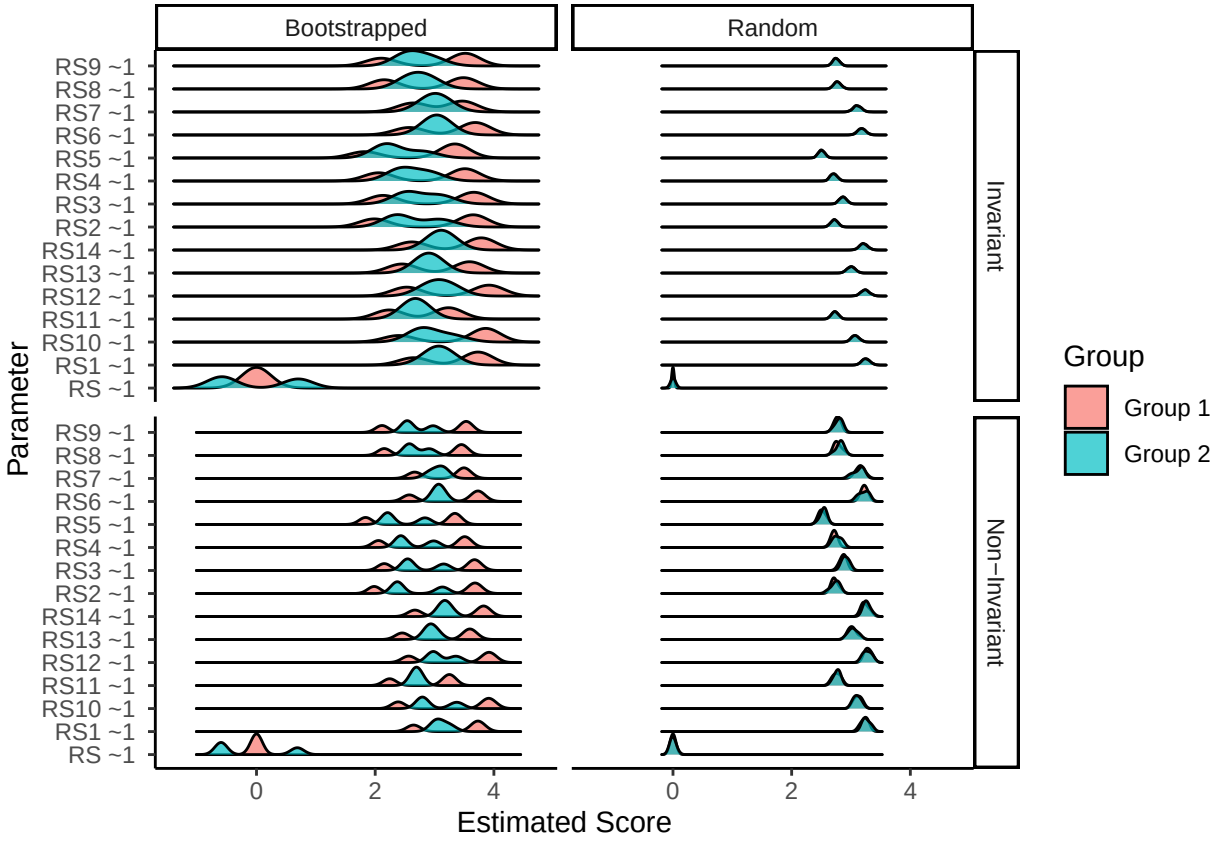


Figure 20

RS14 scale invariance density plots, illustrating invariant versus non-invariant bootstrapped and random runs for each parameter.

Simulating from models

Here's an example of how to simulate directly from a `lavaan` model:

```
# first build your model

# this example is separate for each group

model.invariant.g1 <- "

# loadings

lv =~ .8*q1 + .4*q2 + .6*q3 + .3*q4 + .6*q5

# set the residual for invariance on q4

q4 ~~ 1*q4

# set the intercept for invariance on q4
```

```
q4 ~ 0*1

# set the intercept to zero for df purposes

q1 ~ 0*1

# allow the latent mean to be estimated

lv ~ 1"

model.invariant.g2 <- "lv =~ .77*q1 + .43*q2 + .58*q3 + .3*q4 + .61*q5

q4 ~~ 1*q4

q4 ~ 0*1

q1 ~ 0*1

lv ~ 1"


# simulate data invariant separately for each group

df.invariant <- bind_rows(

  # lavaan function

  simulateData(

    # model with estimates

    model = model.invariant.g1,

    # how many data points

    sample.nobs = 250,

    # mean structure for mgcfa models

    meanstructure = T,

    # model type

    model.type = "cfa",

    # set seed for reproducibility

    seed = 1234) %>%

    # add a group label to the data

    mutate(group = "Group 1"),

  simulateData(

    model = model.invariant.g2,
```

```
    sample.nobs = 250,
    meanstructure = T,
    model.type = "cfa",
    seed = 1234) %>%
    mutate(group = "Group 2")
)
```

Simulating from matrices

Here's an example of how to simulate using `MASS` and covariance or correlation matrices.

```
library(MASS)


# covariance matrix
university.cov <- lav_matrix_lower2full(
    c(169.00,
      73.710, 182.2500,
      73.229, 88.4250, 171.6100,
      63.375, 72.5625, 127.7250, 156.2500,
      42.120, 67.4325, 122.0265, 123.1875, 182.2500,
      57.226, 63.2610, 117.1926, 154.4250, 138.0240, 201.6400,
      30.875, 32.0625, 60.9805, 62.9375, 76.9500, 79.5910, 90.2500,
      36.075, 38.9610, 61.0722, 58.2750, 65.9340, 70.9290, 81.1965, 123.2100,
      18.096, 21.1410, 26.2131, 39.1500, 44.6310, 46.9452, 48.7635, 56.0106, 75.6900))


# give it names
rownames(university.cov) <-
    colnames(university.cov) <-
    c("class", "social", "learn", "chronic", "physical", "sex",
      "depression", "anxiety", "stress")


# means - you need standard deviation if you only have a correlation matrix
university.means <- c(3.4, 4.3, 3.7, 3.2, 4.5, 1.2, 4.0, 3.5, 4.2)


# use mass function
DF <- mvrnorm(n = 200, mu = university.means, Sigma = university.cov)
```

```
head(DF)
```

```
##          class      social     learn   chronic    physical         sex
## [1,] 12.085294 26.83043663 13.895103 20.634374  16.7903295  30.5468832
## [2,]  3.939545 10.93624431  3.093025  8.316128  -7.2199148   9.3579991
## [3,] 15.659627  0.22811723  5.205657  5.224293  -1.4425219  -1.2710662
## [4,] 23.086133 12.43649966  1.891769 -5.913170 -19.4937389 -13.5922410
## [5,] 10.856492 23.30887194 17.124064 11.438840   2.4659294   1.7330709
## [6,] -4.328380  0.07907149 -1.000636 -1.654947  -0.7365838  -0.3958833
##      depression    anxiety      stress
## [1,]  14.174374   8.001766  13.6534279
## [2,]   3.940819  -6.598153  -7.9229552
## [3,]  -8.927508 -10.335571  -2.8100779
## [4,]  -0.149840   6.475669 -15.5300195
## [5,]  -6.492809 -12.524601   0.4924153
## [6,]  11.917631   3.433587   8.1456346
```

MGCFA 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.

[tbp]

Table 3

Model Fit for Invariant Model

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
Overall	7,516.454	7,579.673	1.000	1.036	0.000	0.006
Group Group 1	3,765.749	3,818.571	0.976	0.953	0.047	0.031
Group Group 2	3,767.599	3,820.421	1.000	1.008	0.000	0.021
Configural	7,533.348	7,659.786	0.991	0.982	0.030	0.026
loadings	7,528.476	7,638.056	0.994	0.992	0.020	0.033
intercepts	7,522.397	7,615.118	1.000	1.003	0.000	0.035
residuals	7,520.435	7,592.083	0.991	0.992	0.020	0.046

[tbp]

Table 4

Model Fit for Small Differences in Loadings

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
Overall	7,530.321	7,593.540	0.977	0.955	0.049	0.025
Group Group 1	3,765.749	3,818.571	0.976	0.953	0.047	0.031
Group Group 2	3,785.242	3,838.064	0.979	0.958	0.050	0.029
Configural	7,550.991	7,677.430	0.978	0.956	0.048	0.030
loadings	7,550.133	7,659.713	0.966	0.952	0.051	0.047
intercepts	7,542.675	7,635.397	0.979	0.977	0.035	0.047
residuals	7,534.091	7,605.739	0.993	0.994	0.019	0.054

Simulated Parital Invariance Results

[tbp]

Table 5

Model Fit for Medium Differences in Loadings

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
Overall	7,598.681	7,661.900	0.946	0.893	0.078	0.035
Group Group 1	3,765.749	3,818.571	0.976	0.953	0.047	0.031
Group Group 2	3,820.090	3,872.912	0.969	0.938	0.067	0.034
Configural	7,585.839	7,712.277	0.972	0.944	0.058	0.033
loadings	7,613.434	7,723.014	0.865	0.807	0.107	0.079
intercepts	7,606.648	7,699.370	0.874	0.860	0.091	0.079
residuals	7,600.057	7,671.705	0.880	0.895	0.079	0.091

[tbp]

Table 6

Model Fit for Large Differences in Loadings

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
Overall	7,684.982	7,748.201	0.987	0.973	0.040	0.023
Group Group 1	3,765.749	3,818.571	0.976	0.953	0.047	0.031
Group Group 2	3,857.143	3,909.965	0.992	0.984	0.037	0.029
Configural	7,622.892	7,749.330	0.986	0.972	0.042	0.030
loadings	7,674.188	7,783.767	0.817	0.738	0.131	0.095
intercepts	7,667.682	7,760.403	0.824	0.805	0.113	0.096
residuals	7,683.177	7,754.825	0.762	0.793	0.116	0.138

[tbp]

Table 7

Model Fit for Small Differences in Intercepts

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
Overall	7,520.471	7,583.690	1.000	1.035	0.000	0.007
Group Group 1	3,765.749	3,818.571	0.976	0.953	0.047	0.031
Group Group 2	3,767.599	3,820.421	1.000	1.008	0.000	0.021
Configural	7,533.348	7,659.786	0.991	0.982	0.030	0.026
loadings	7,528.476	7,638.056	0.994	0.992	0.020	0.033
intercepts	7,526.312	7,619.034	0.987	0.986	0.027	0.040
residuals	7,524.356	7,596.005	0.975	0.978	0.033	0.050

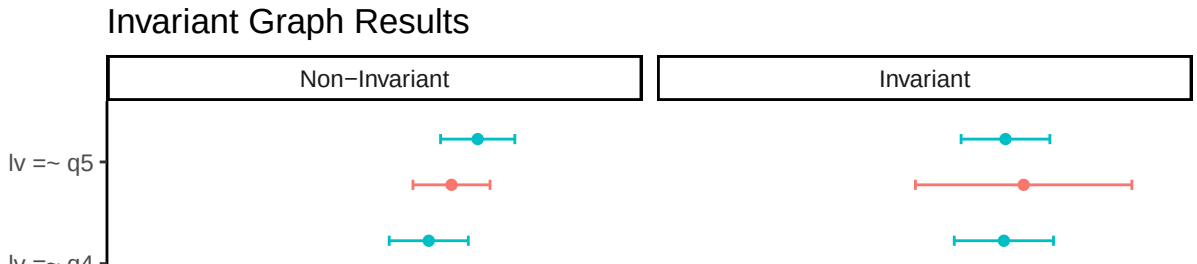
[tbp]

Table 8

Model Fit for Medium Differences in Intercepts

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
Overall	7,538.375	7,601.594	1.000	1.033	0.000	0.008
Group Group 1	3,765.749	3,818.571	0.976	0.953	0.047	0.031
Group Group 2	3,767.599	3,820.421	1.000	1.008	0.000	0.021
Configural	7,533.348	7,659.786	0.991	0.982	0.030	0.026
loadings	7,528.476	7,638.056	0.994	0.992	0.020	0.033
intercepts	7,544.002	7,636.724	0.917	0.907	0.068	0.059
residuals	7,542.064	7,613.712	0.905	0.917	0.065	0.067

Invariance Plots Difference Scores by Condition



[tbp]

Table 9

Model Fit for Large Differences in Intercepts

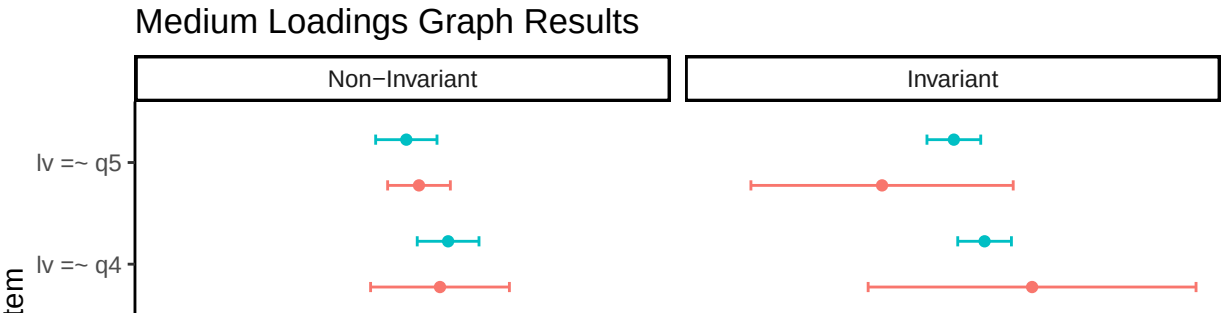
Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
Overall	7,568.748	7,631.967	1.000	1.032	0.000	0.008
Group Group 1	3,765.749	3,818.571	0.976	0.953	0.047	0.031
Group Group 2	3,767.599	3,820.421	1.000	1.008	0.000	0.021
Configural	7,533.348	7,659.786	0.991	0.982	0.030	0.026
loadings	7,528.476	7,638.056	0.994	0.992	0.020	0.033
intercepts	7,574.054	7,666.776	0.797	0.775	0.106	0.084
residuals	7,572.174	7,643.823	0.785	0.813	0.097	0.090

[tbp]

Table 10

Model Fit for Small Differences in Residuals

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
Overall	7,462.007	7,525.226	1.000	1.020	0.000	0.013
Group Group 1	3,765.749	3,818.571	0.976	0.953	0.047	0.031
Group Group 2	3,703.797	3,756.619	0.962	0.924	0.061	0.037
Configural	7,469.546	7,595.984	0.969	0.938	0.054	0.034
loadings	7,471.637	7,581.217	0.944	0.920	0.062	0.049
intercepts	7,465.722	7,558.443	0.952	0.946	0.051	0.051
residuals	7,465.986	7,537.635	0.930	0.939	0.054	0.065



[tbp]

Table 11

Model Fit for Medium Differences in Residuals

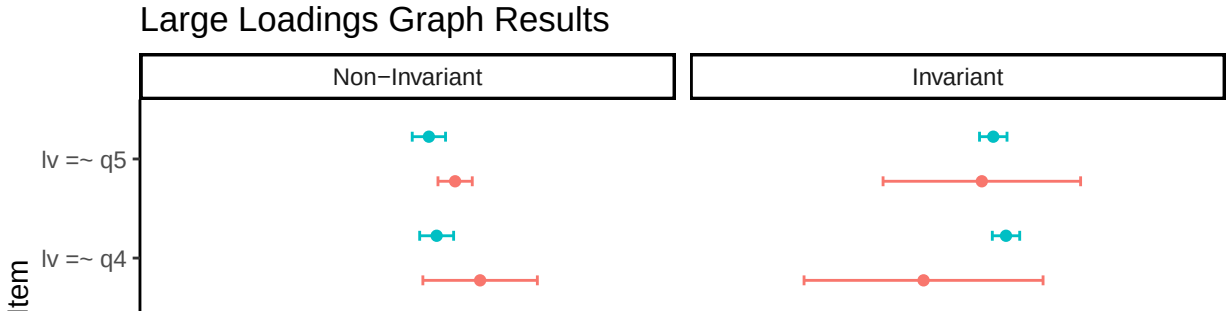
Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
Overall	7,382.013	7,445.232	0.997	0.995	0.016	0.018
Group Group 1	3,765.749	3,818.571	0.976	0.953	0.047	0.031
Group Group 2	3,602.905	3,655.727	1.000	1.013	0.000	0.023
Configural	7,368.654	7,495.092	0.992	0.983	0.028	0.027
loadings	7,364.904	7,474.483	0.990	0.986	0.025	0.036
intercepts	7,358.503	7,451.224	1.000	1.001	0.000	0.037
residuals	7,385.958	7,457.607	0.864	0.881	0.075	0.098

[tbp]

Table 12

Model Fit for Large Differences in Residuals

Model	AIC	BIC	CFI	TLI	RMSEA	SRMR
Overall	7,300.856	7,364.075	0.998	0.995	0.015	0.018
Group Group 1	3,765.749	3,818.571	0.976	0.953	0.047	0.031
Group Group 2	3,453.099	3,505.921	0.954	0.908	0.069	0.035
Configural	7,218.848	7,345.287	0.965	0.929	0.059	0.033
loadings	7,217.332	7,326.912	0.955	0.935	0.056	0.045
intercepts	7,211.566	7,304.287	0.962	0.958	0.046	0.047
residuals	7,304.566	7,376.215	0.562	0.619	0.137	0.189



[tbp]

Table 13

Fit Estimates for Partial Invariance Residuals on Invariant Data

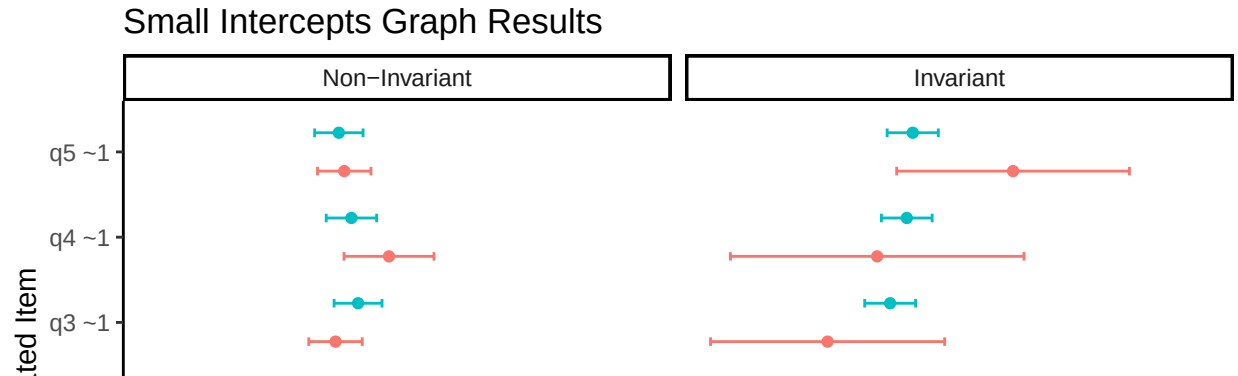
Estimated Parameter	CFI	RSMEA
q1 ~~ q1	0.990	0.021
q2 ~~ q2	0.987	0.024
q3 ~~ q3	0.996	0.014
q4 ~~ q4	1.000	0.000
q5 ~~ q5	0.987	0.025
lv ~~ lv	0.991	0.020

[tbp]

Table 14

Fit Estimates for Partial Invariance Loadings for Small Loading Data

Estimated Parameter	CFI	RSMEA
lv =~ q1	0.993	0.019
lv =~ q2	0.989	0.023
lv =~ q3	0.989	0.023
lv =~ q4	1.000	0.000
lv =~ q5	0.994	0.017



[tbp]

Table 15

Fit Estimates for Partial Invariance Loadings for Medium Loading Data

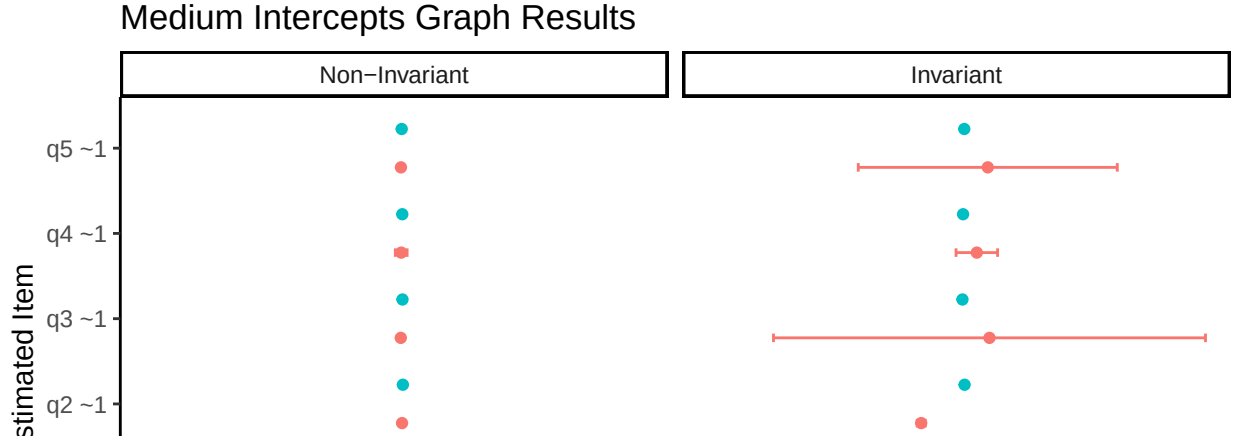
Estimated Parameter	CFI	RSMEA
lv =~ q1	0.880	0.079
lv =~ q2	0.898	0.074
lv =~ q3	0.878	0.081
lv =~ q4	0.962	0.045
lv =~ q5	0.907	0.071

[tbp]

Table 16

Fit Estimates for Partial Invariance Loadings for Large Loading Data

Estimated Parameter	CFI	RSMEA
lv =~ q1	0.762	0.116
lv =~ q2	0.770	0.117
lv =~ q3	0.762	0.119
lv =~ q4	0.971	0.041
lv =~ q5	0.842	0.097



[tbp]

Table 17

Fit Estimates for Partial Invariance Loadings for Small Intercept Data

Estimated Parameter	CFI	RSMEA
q1 ~1	0.975	0.033
lv ~1	0.975	0.033
q2 ~1	0.972	0.035
q3 ~1	0.972	0.036
q4 ~1	0.988	0.023
q5 ~1	0.971	0.036

[tbp]

Table 18

Fit Estimates for Partial Invariance Loadings for Medium Intercept Data

Estimated Parameter	CFI	RSMEA
q1 ~1	0.905	0.065
lv ~1	0.905	0.065
q2 ~1	0.901	0.067
q3 ~1	0.901	0.067
q4 ~1	0.988	0.023
q5 ~1	0.902	0.067

[tbp]

Table 19

Fit Estimates for Partial Invariance

Loadings for Large Intercept Data

Estimated Parameter	CFI	RSMEA
q1 ~1	0.785	0.097
lv ~1	0.785	0.097
q2 ~1	0.781	0.100
q3 ~1	0.781	0.100
q4 ~1	0.988	0.023
q5 ~1	0.784	0.099

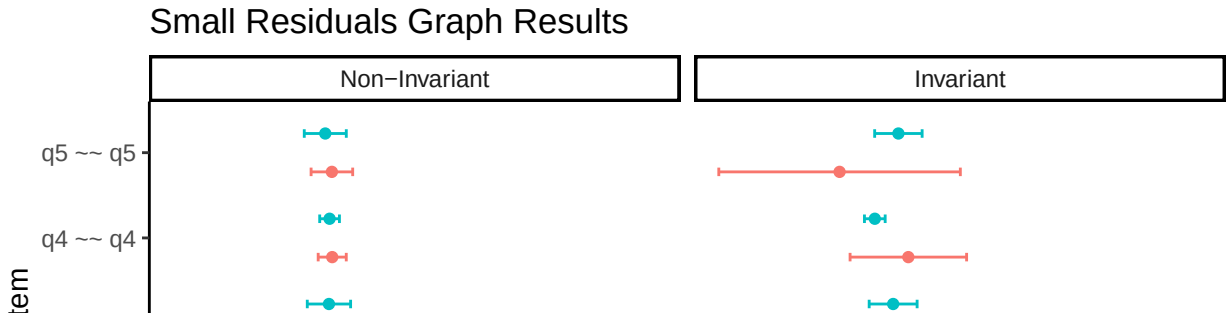
[tbp]

Table 20

Fit Estimates for Partial Invariance

Loadings for Small Residual Data

Estimated Parameter	CFI	RSMEA
q1 ~~ q1	0.928	0.056
q2 ~~ q2	0.936	0.053
q3 ~~ q3	0.926	0.057
q4 ~~ q4	0.955	0.044
q5 ~~ q5	0.926	0.057
lv ~~ lv	0.930	0.054



[tbp]

Table 21

Fit Estimates for Partial Invariance

Loadings for Medium Residual Data

Estimated Parameter	CFI	RSMEA
q1 ~~ q1	0.869	0.075
q2 ~~ q2	0.860	0.078
q3 ~~ q3	0.870	0.075
q4 ~~ q4	0.994	0.016
q5 ~~ q5	0.862	0.077
lv ~~ lv	0.864	0.075

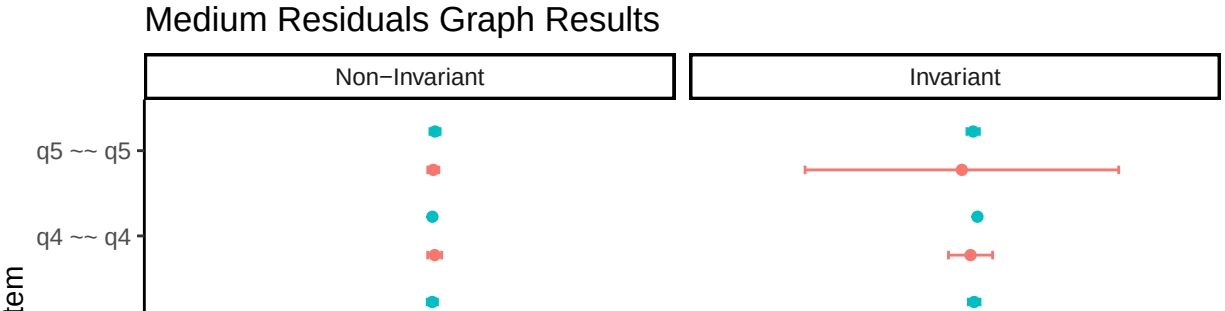
[tbp]

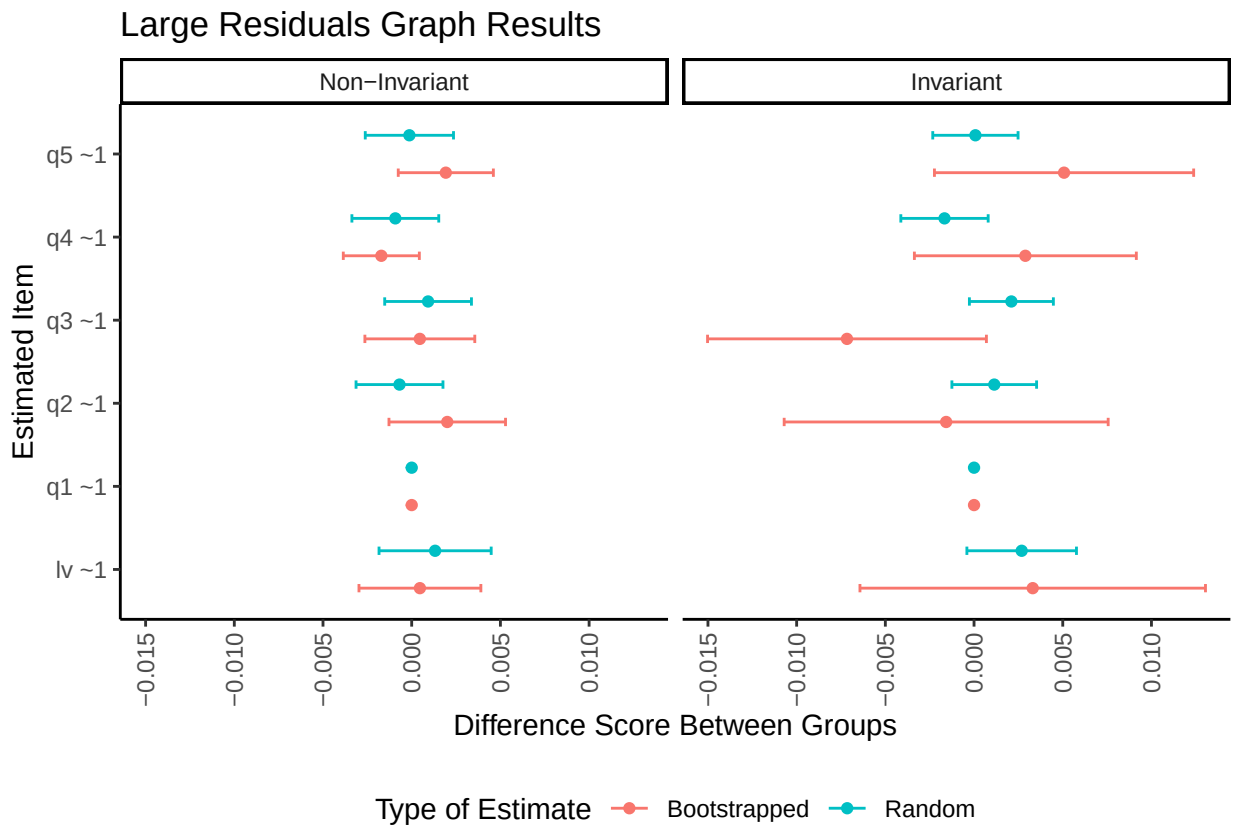
Table 22

Fit Estimates for Partial Invariance

Loadings for Large Residual Data

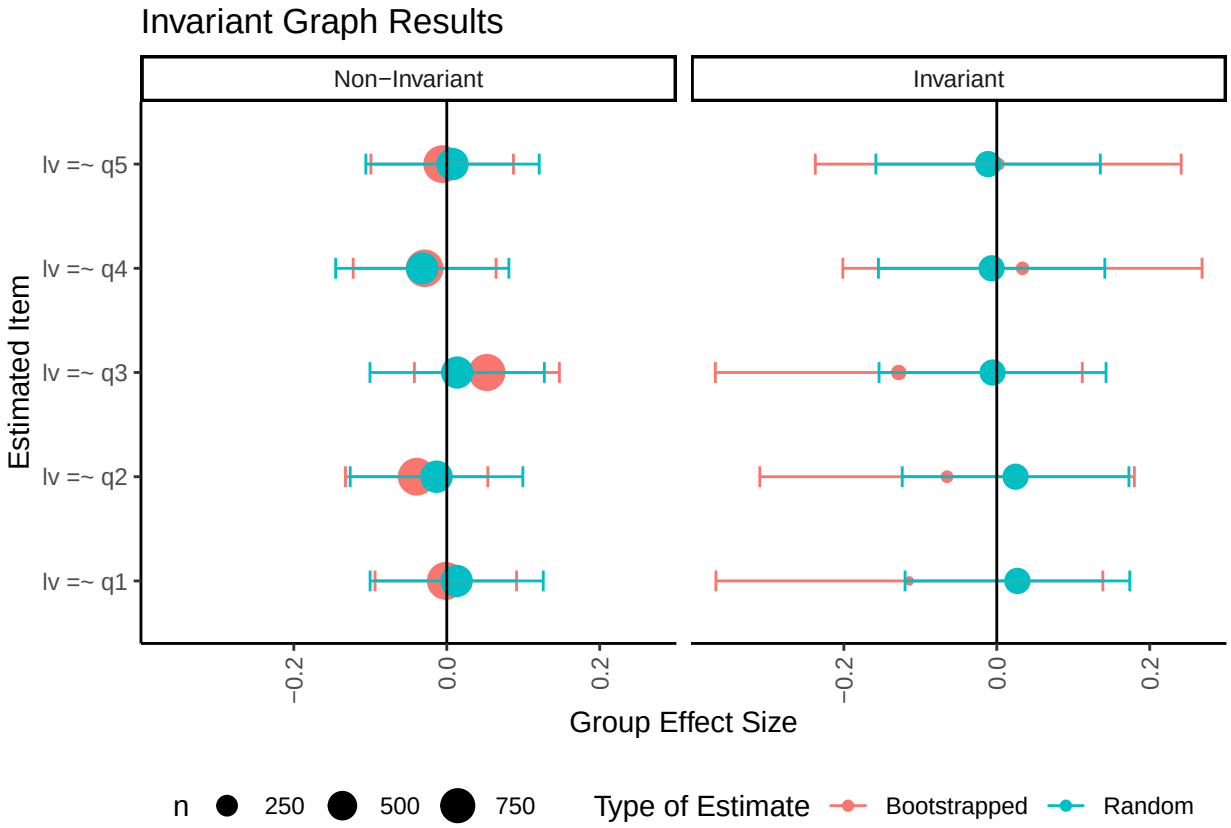
Estimated Parameter	CFI	RSMEA
q1 ~~ q1	0.558	0.140
q2 ~~ q2	0.559	0.140
q3 ~~ q3	0.560	0.140
q4 ~~ q4	0.972	0.035
q5 ~~ q5	0.559	0.140
lv ~~ lv	0.562	0.137

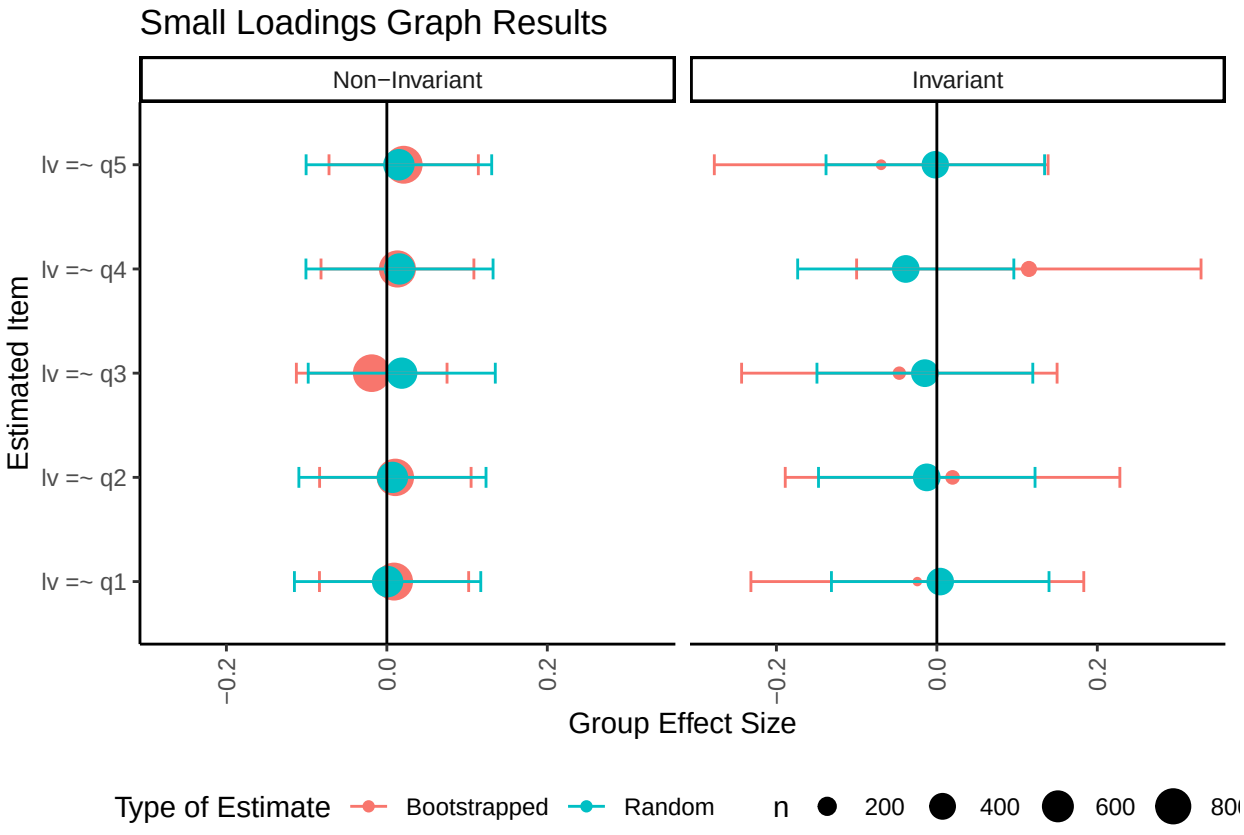


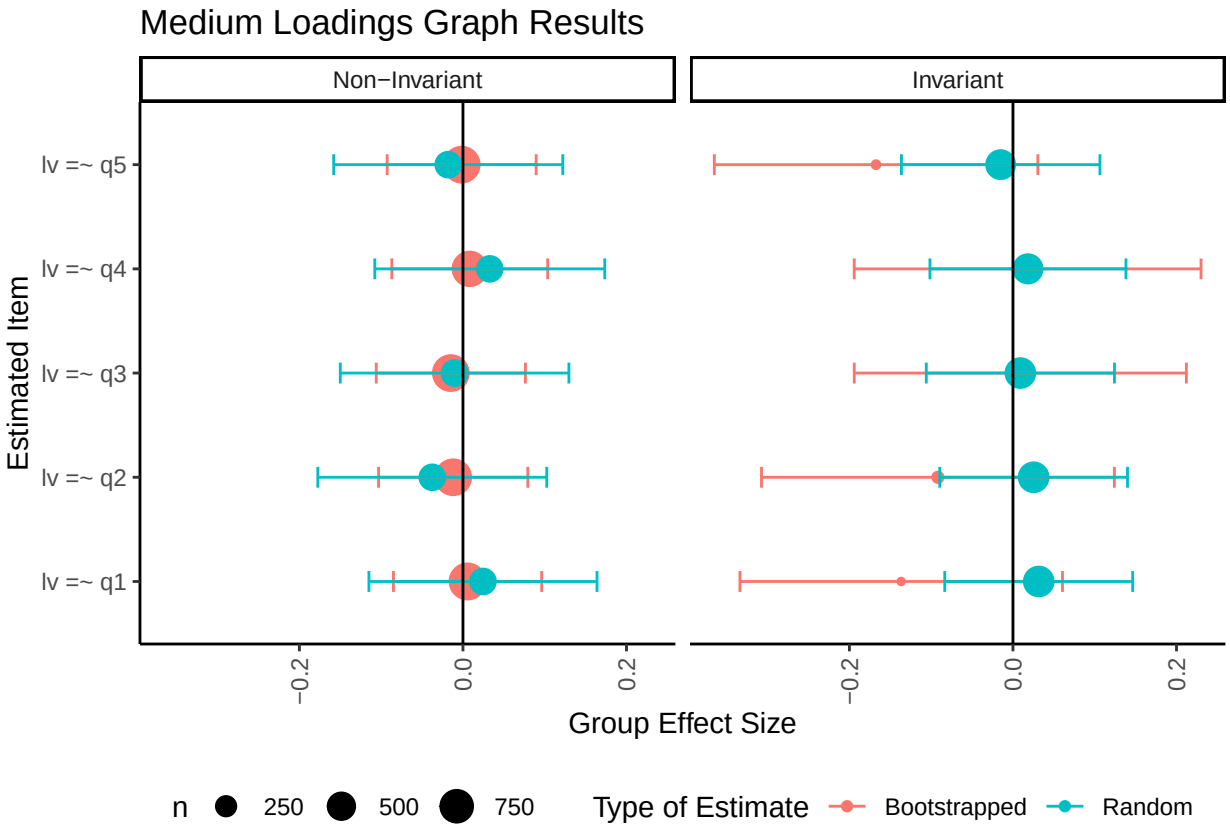

Large Residuals Graph Results
Non−Invariant
Invariant
Estimated Item
q5 ~1
q4 ~1
q3 ~1
q2 ~1
q1 ~1
Iv ~1
−0.015
−0.010
−0.005
0.000
0.005
0.010
Difference Score Between Groups
Type of Estimate
Bootstrapped
Random

910

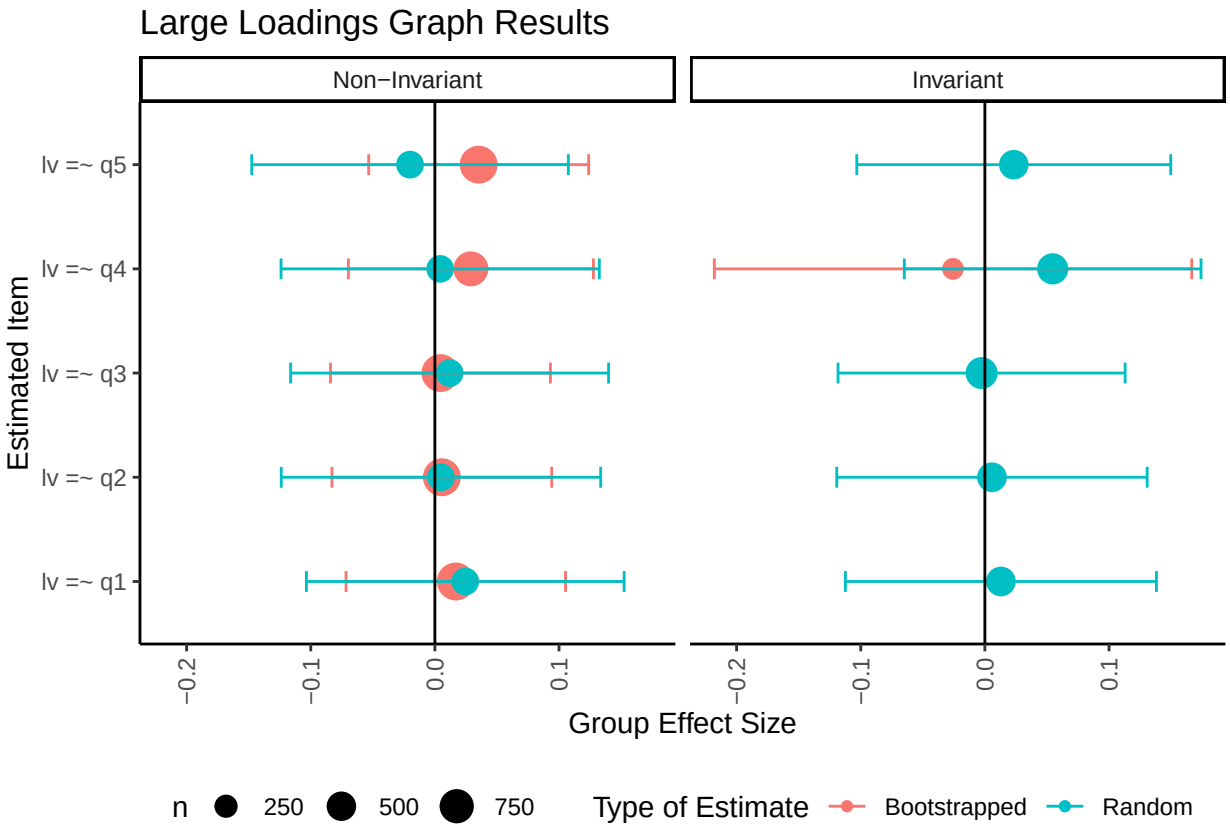
Invariance Plots Effect Sizes by Condition



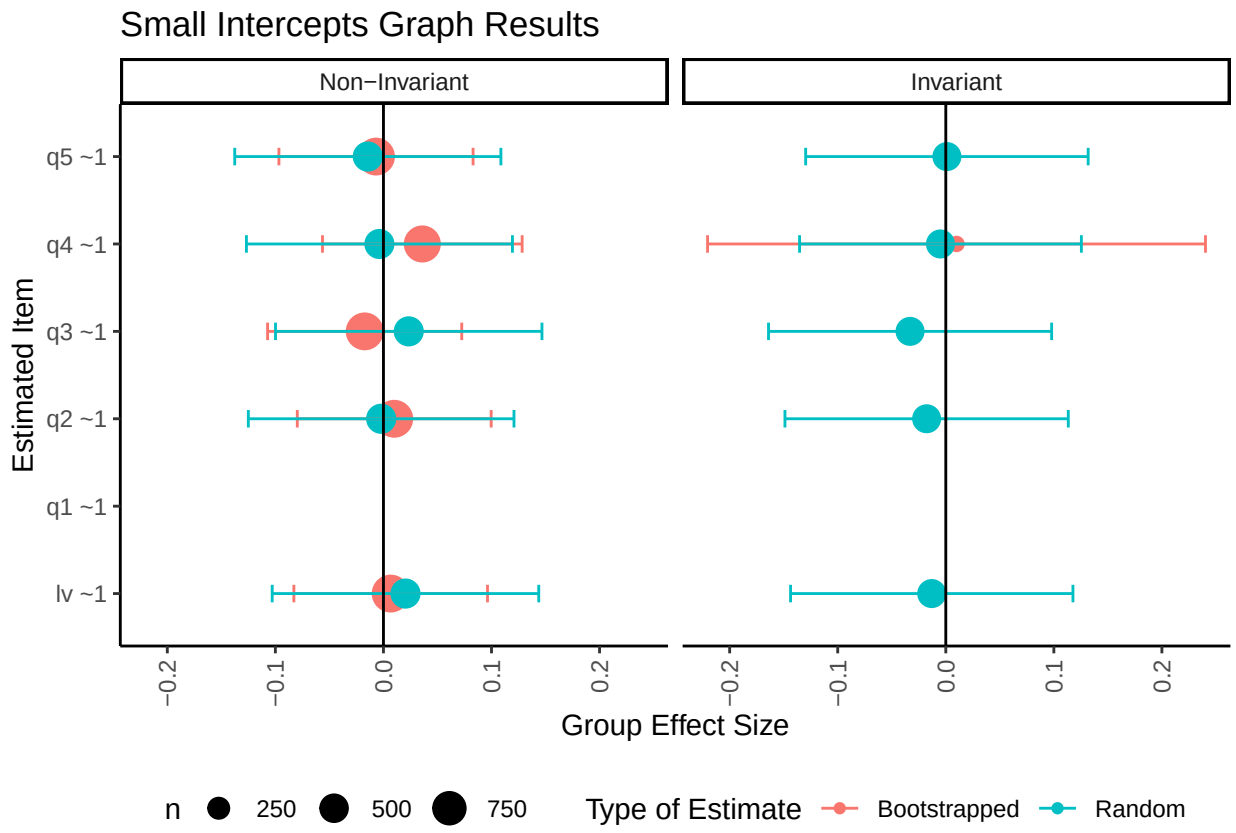



Medium Loadings Graph Results
Non−Invariant
Invariant
lv =~ q5
lv =~ q4
lv =~ q3
lv =~ q2
lv =~ q1
Estimated Item
−0.2
0.0
0.2
Group Effect Size
n
250
500
750
Type of Estimate
Bootstrapped
Random

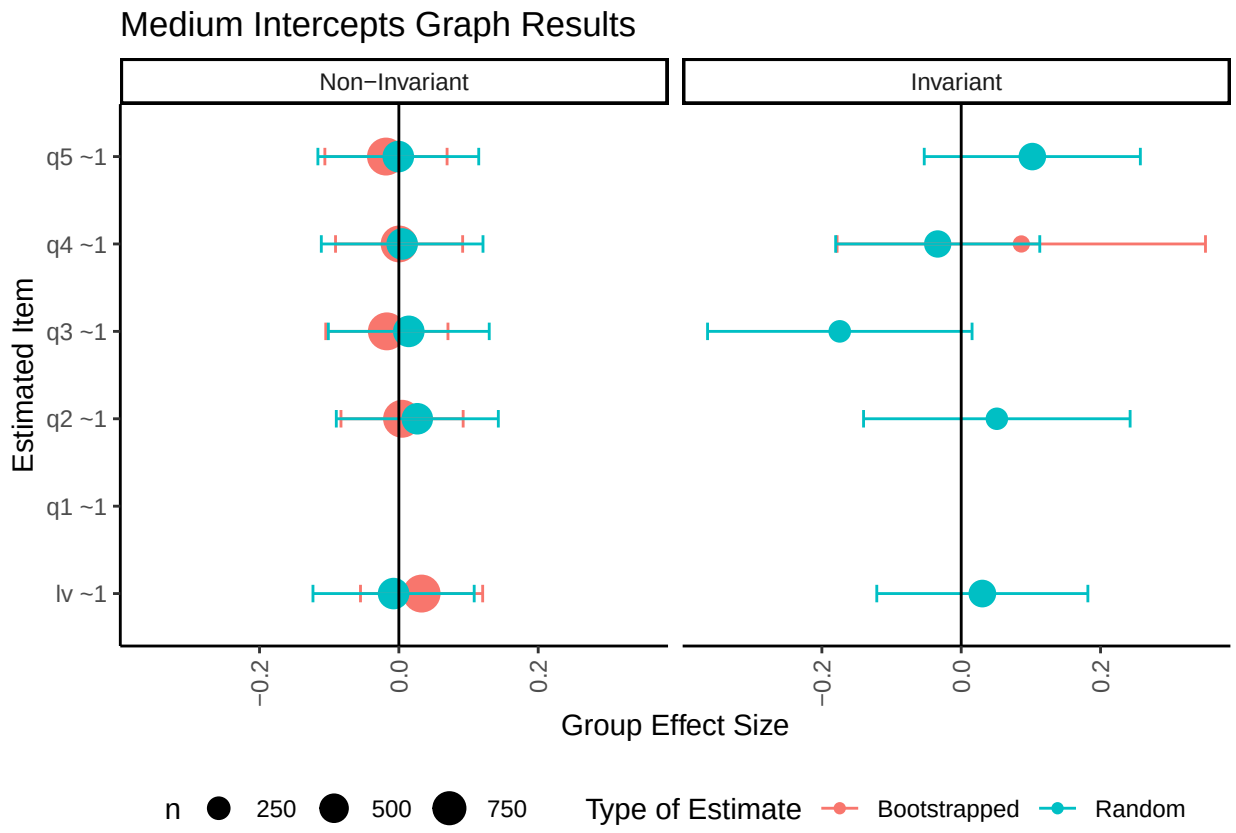
914


Large Loadings Graph Results
Non−Invariant
Invariant
lv =~ q5
lv =~ q4
lv =~ q3
lv =~ q2
lv =~ q1
Estimated Item
−0.2
−0.1
0.0
0.1
Group Effect Size
n
250
500
750
Type of Estimate
Bootstrapped
Random

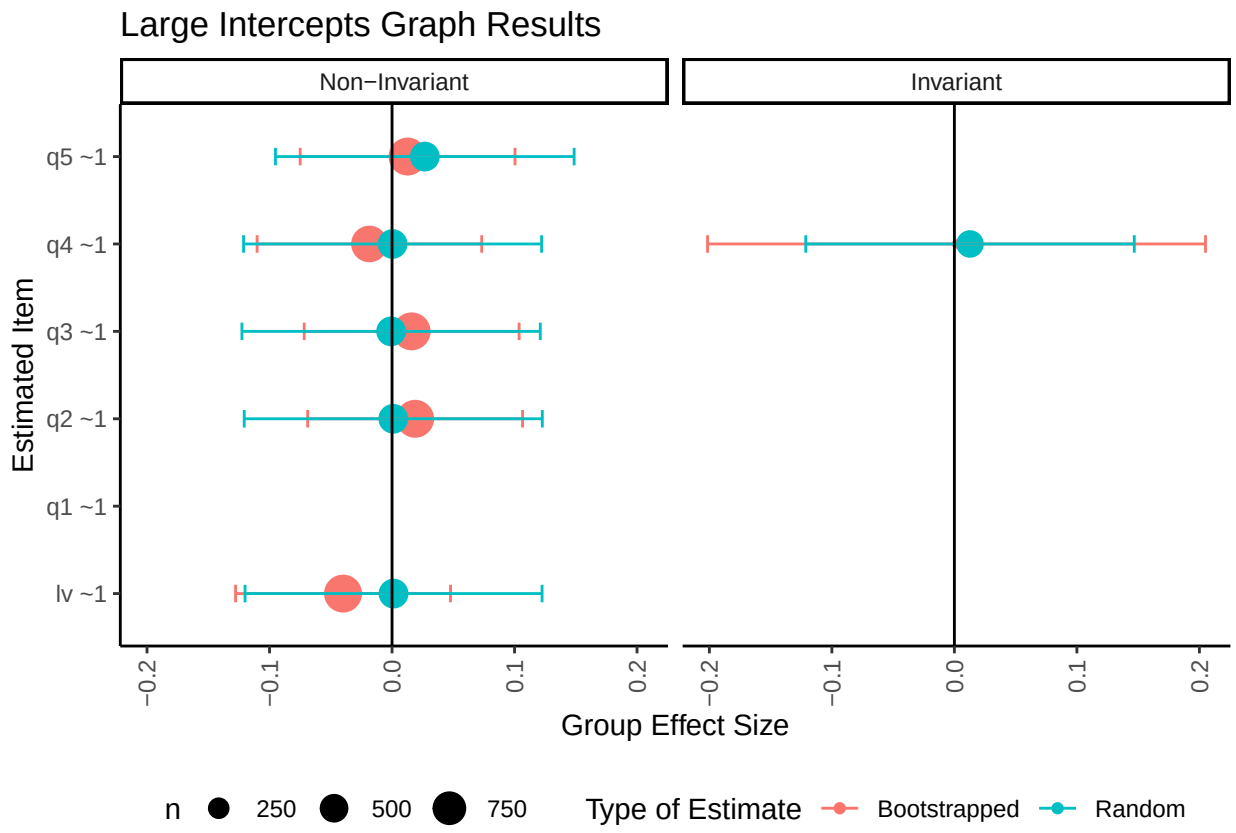
915

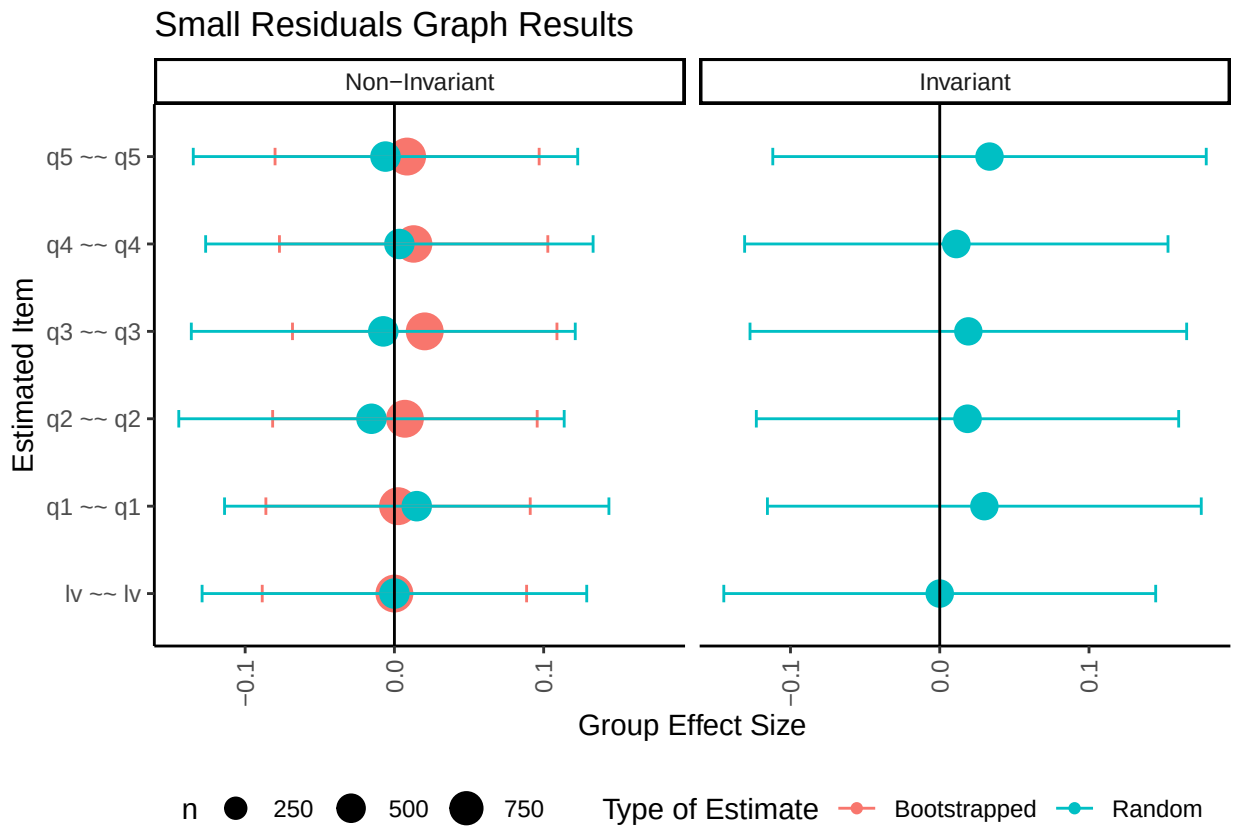

Small Intercepts Graph Results
Non−Invariant
Invariant
Estimated Item
q5 ~1
q4 ~1
q3 ~1
q2 ~1
q1 ~1
lv ~1
−0.2
−0.1
0.0
0.1
0.2
Group Effect Size
n
250
500
750
Type of Estimate
Bootstrapped
Random

916

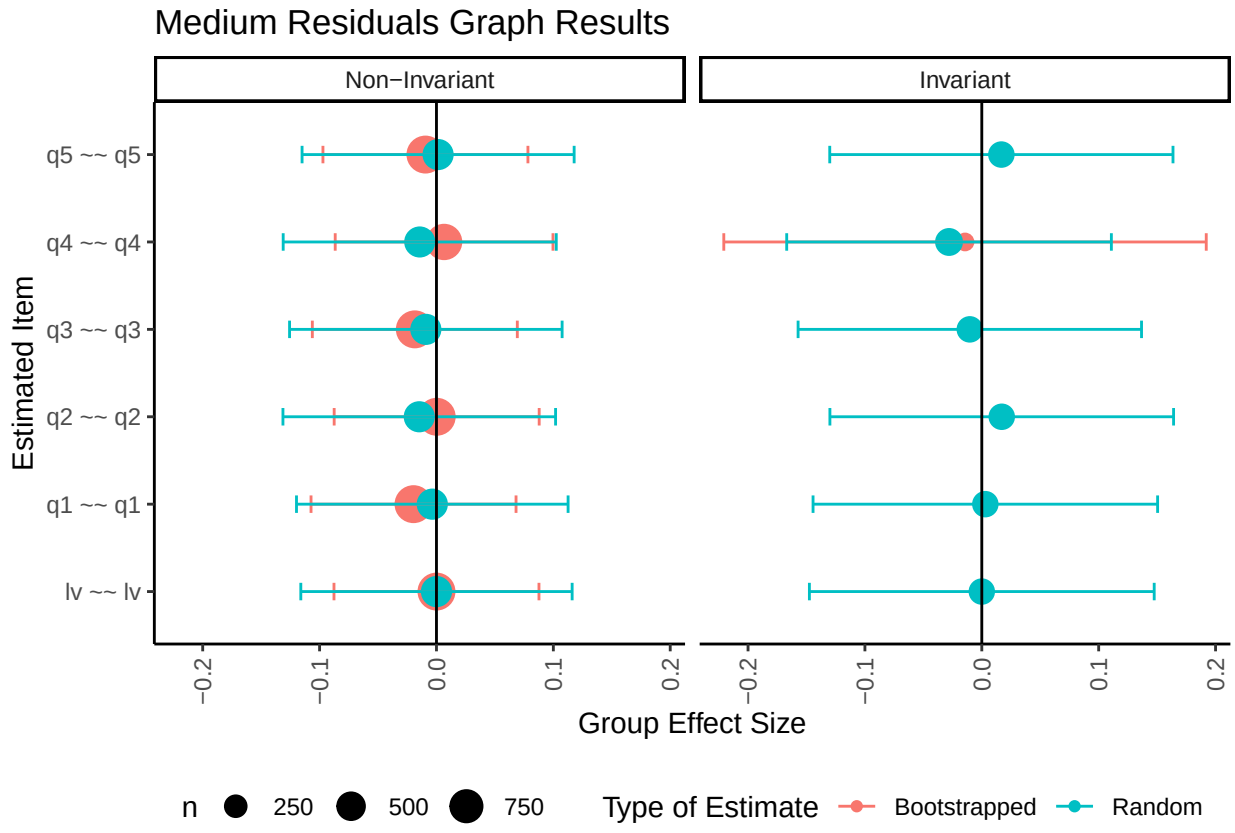

Medium Intercepts Graph Results
Non−Invariant
Invariant
Estimated Item
q5 ~1
q4 ~1
q3 ~1
q2 ~1
q1 ~1
lv ~1
−0.2
0.0
0.2
Group Effect Size
n
250
500
750
Type of Estimate
Bootstrapped
Random

917

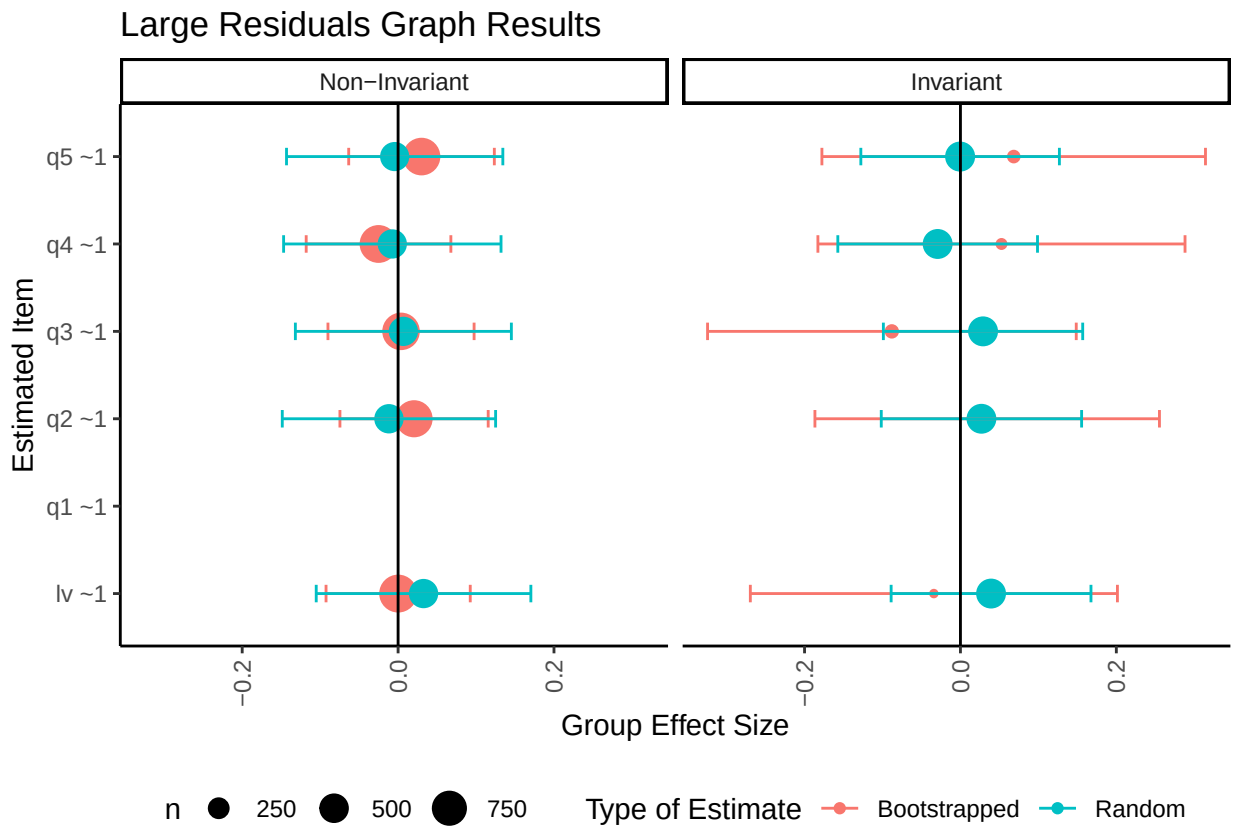

Large Intercepts Graph Results
Non−Invariant
Invariant
Estimated Item
q5 ~1
q4 ~1
q3 ~1
q2 ~1
q1 ~1
Iv ~1
−0.2
−0.1
0.0
0.1
0.2
Group Effect Size
n
250
500
750
Type of Estimate
Bootstrapped
Random


Small Residuals Graph Results
Non−Invariant
Invariant
Estimated Item
q5 ~~ q5
q4 ~~ q4
q3 ~~ q3
q2 ~~ q2
q1 ~~ q1
lv ~~ lv
−0.1
0.0
0.1
Group Effect Size
n
250
500
750
Type of Estimate
Bootstrapped
Random

919

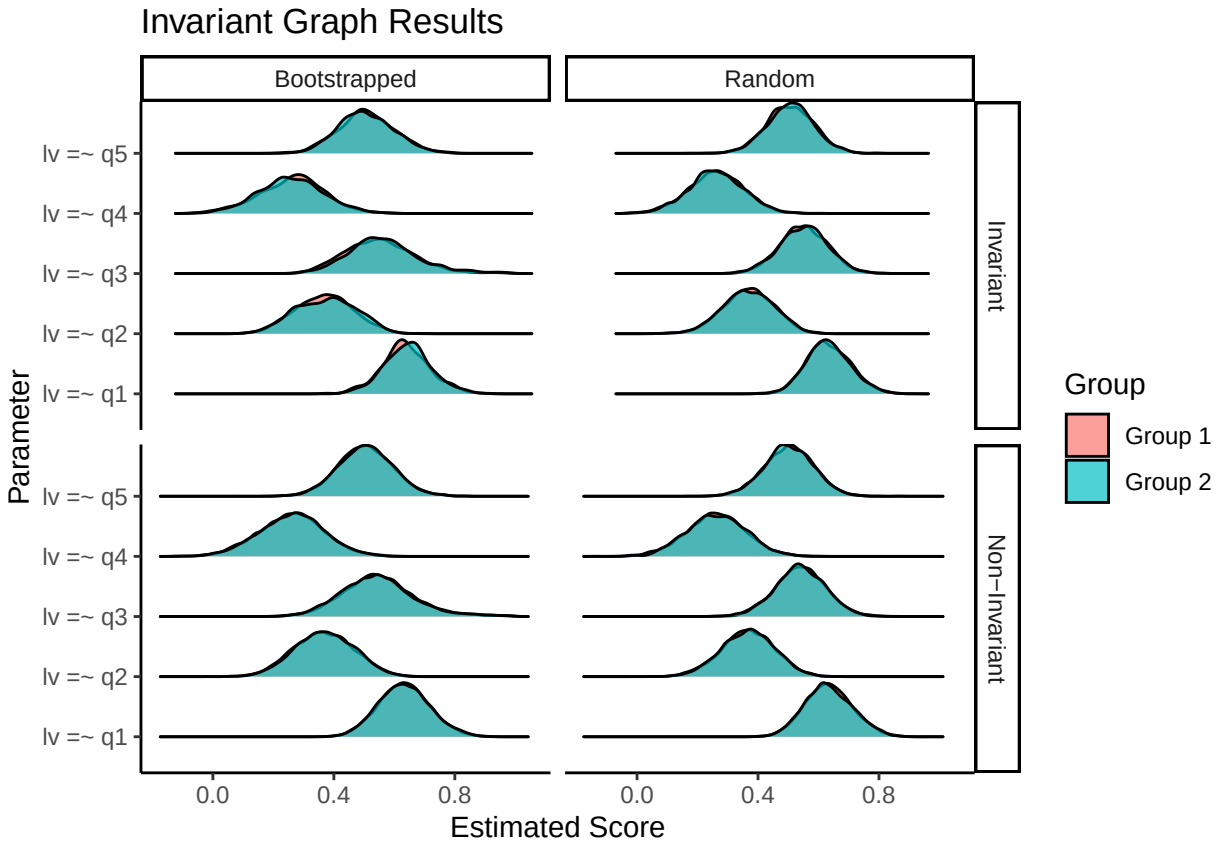

Medium Residuals Graph Results
Non−Invariant
Invariant
Estimated Item
q5 ~~ q5
q4 ~~ q4
q3 ~~ q3
q2 ~~ q2
q1 ~~ q1
lv ~~ lv
−0.2
−0.1
0.0
0.1
0.2
Group Effect Size
n
250
500
750
Type of Estimate
Bootstrapped
Random

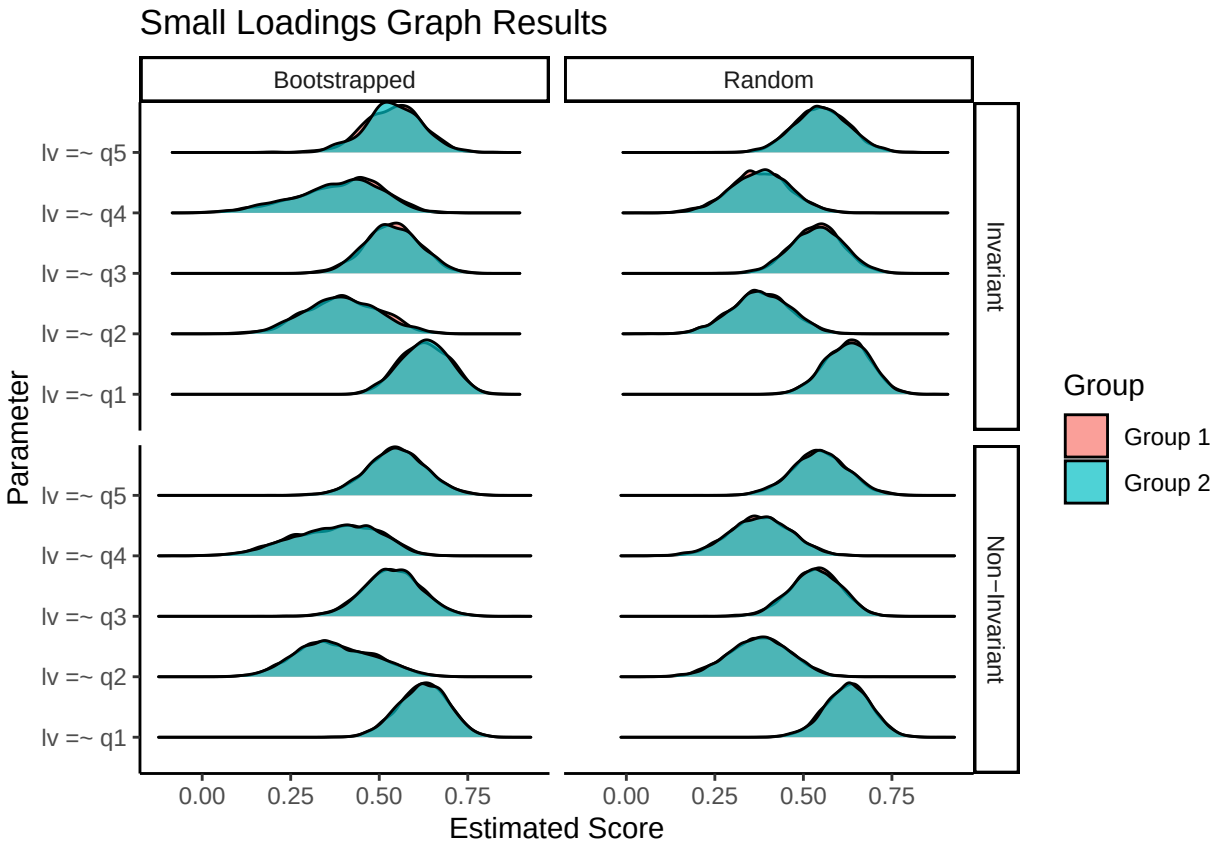
920

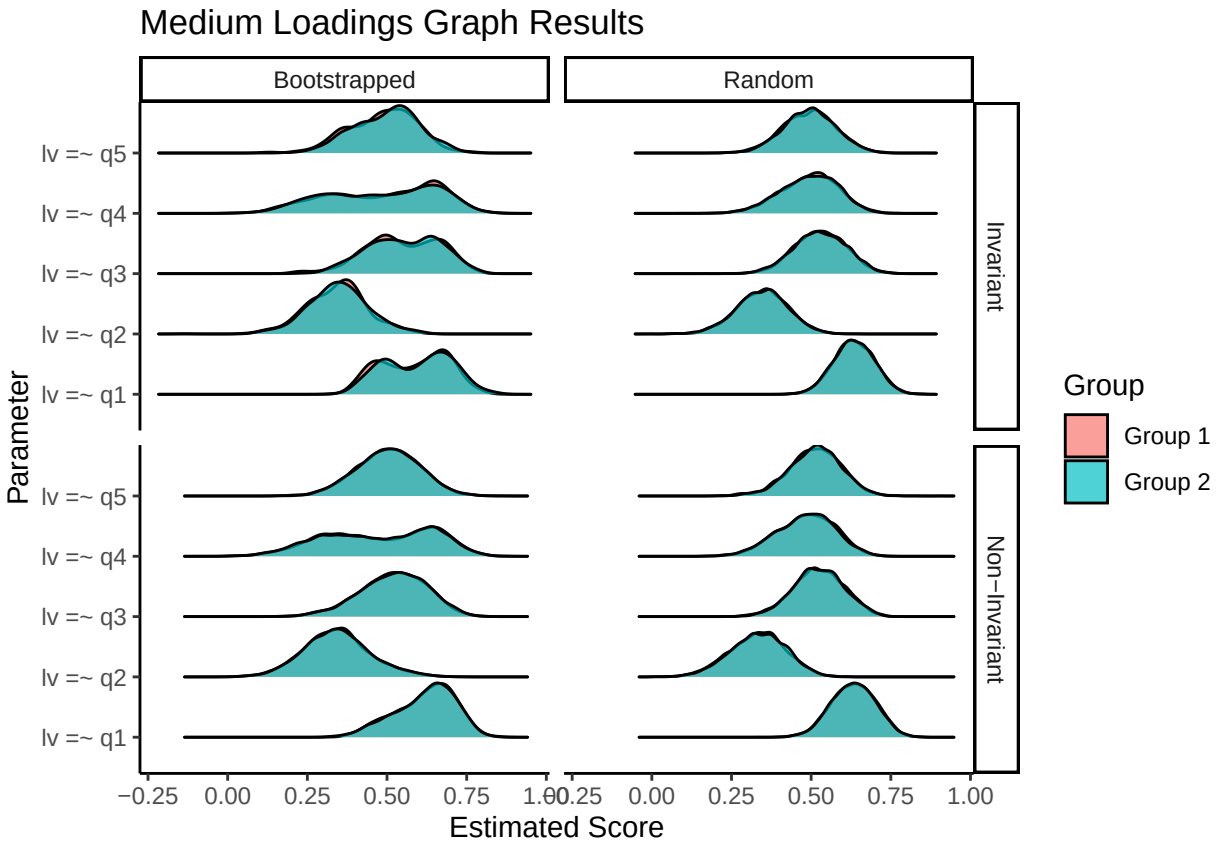

Large Residuals Graph Results
Non−Invariant
Invariant
Estimated Item
q5 ~1
q4 ~1
q3 ~1
q2 ~1
q1 ~1
Iv ~1
−0.2
0.0
0.2
Group Effect Size
n
250
500
750
Type of Estimate
Bootstrapped
Random

921

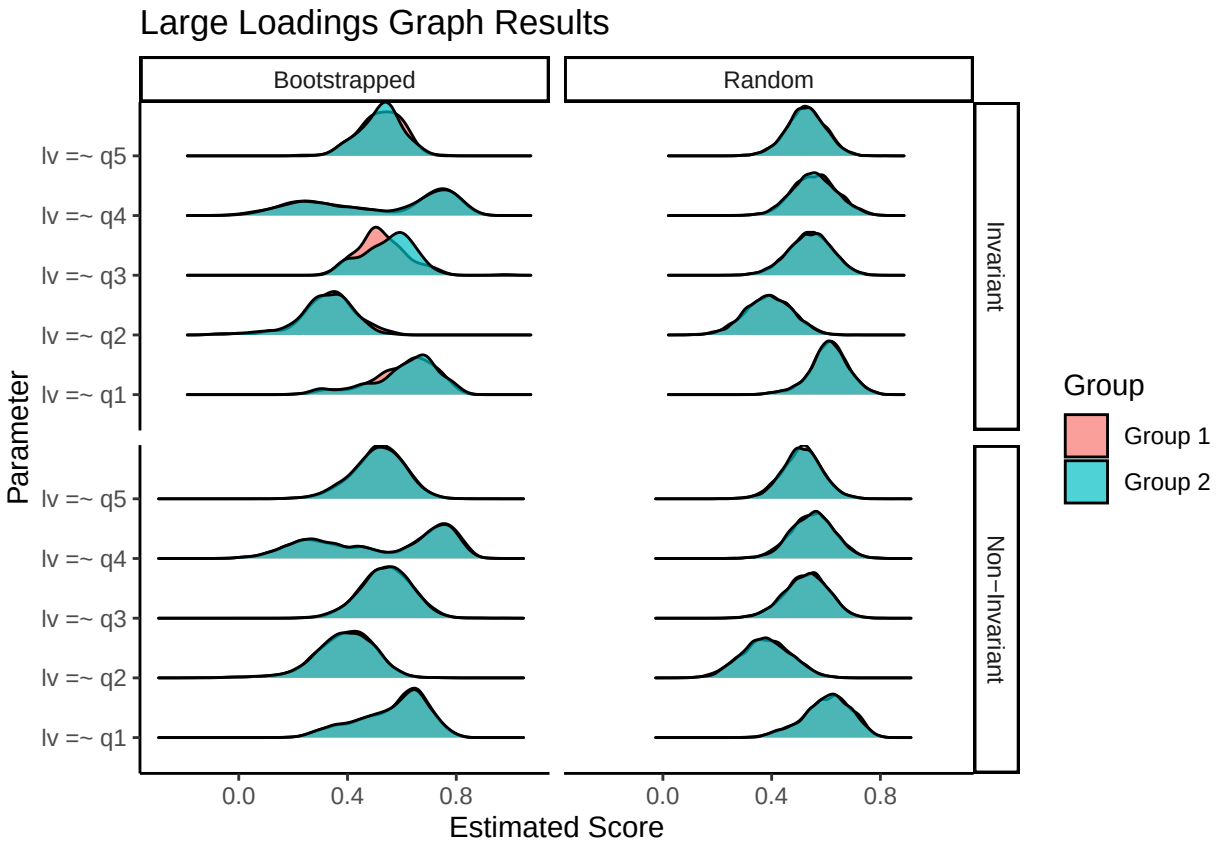
Density Plots by Condition



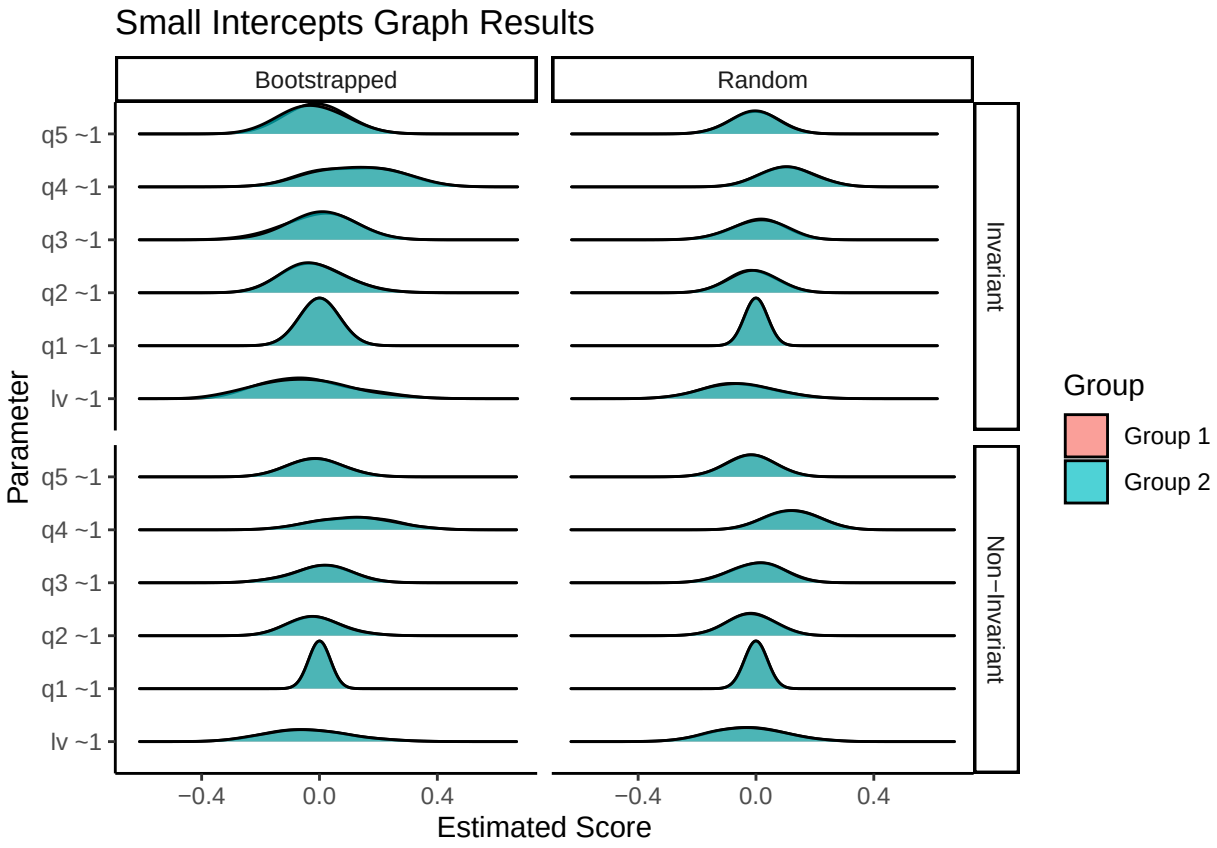



Medium Loadings Graph Results
Bootstrapped
Random
Invariant
Non−Invariant
Parameter
lv =~ q5
lv =~ q4
lv =~ q3
lv =~ q2
lv =~ q1
−0.25
0.00
0.25
0.50
0.75
1.00
Estimated Score
Group
Group 1
Group 2

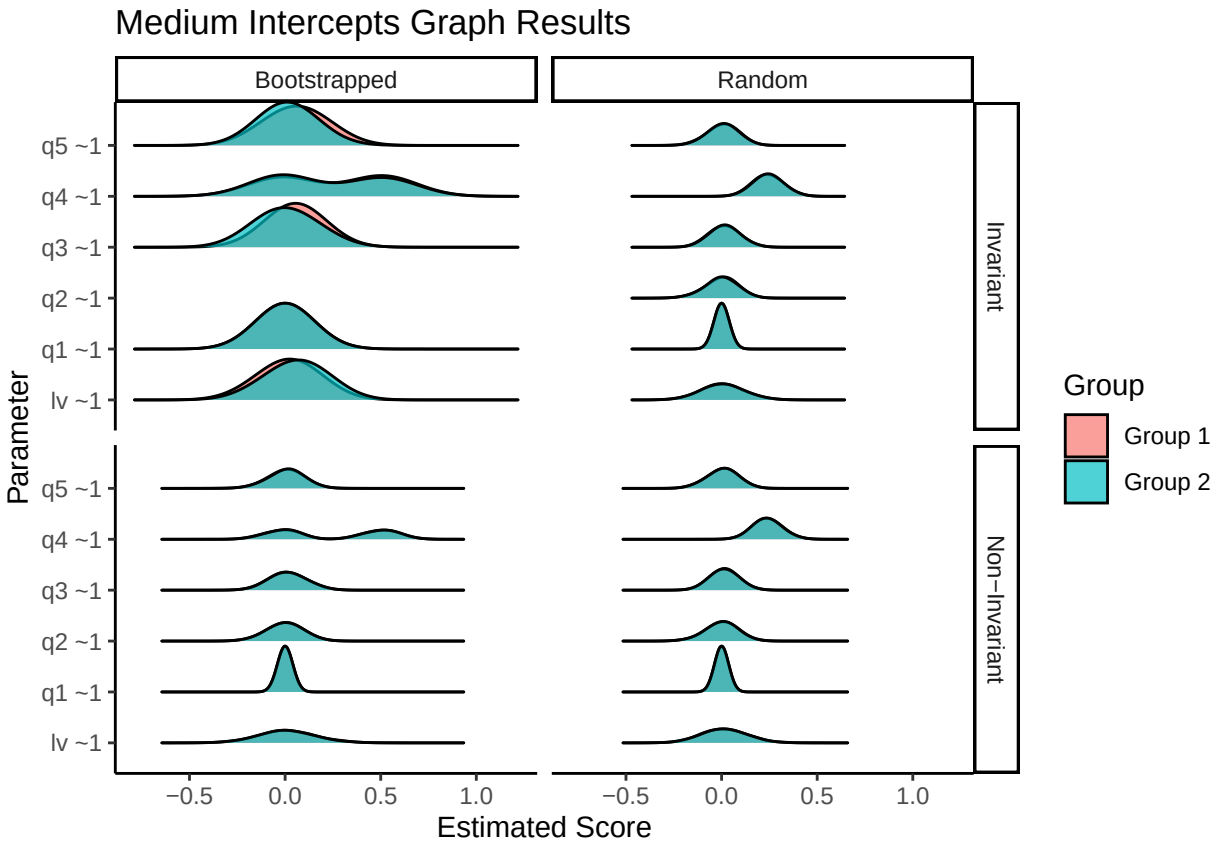
925


Large Loadings Graph Results
Bootstrapped
Random
Invariant
Non−Invariant
Parameter
lv =~ q5
lv =~ q4
lv =~ q3
lv =~ q2
lv =~ q1
0.0
0.4
0.8
Estimated Score
Group
Group 1
Group 2

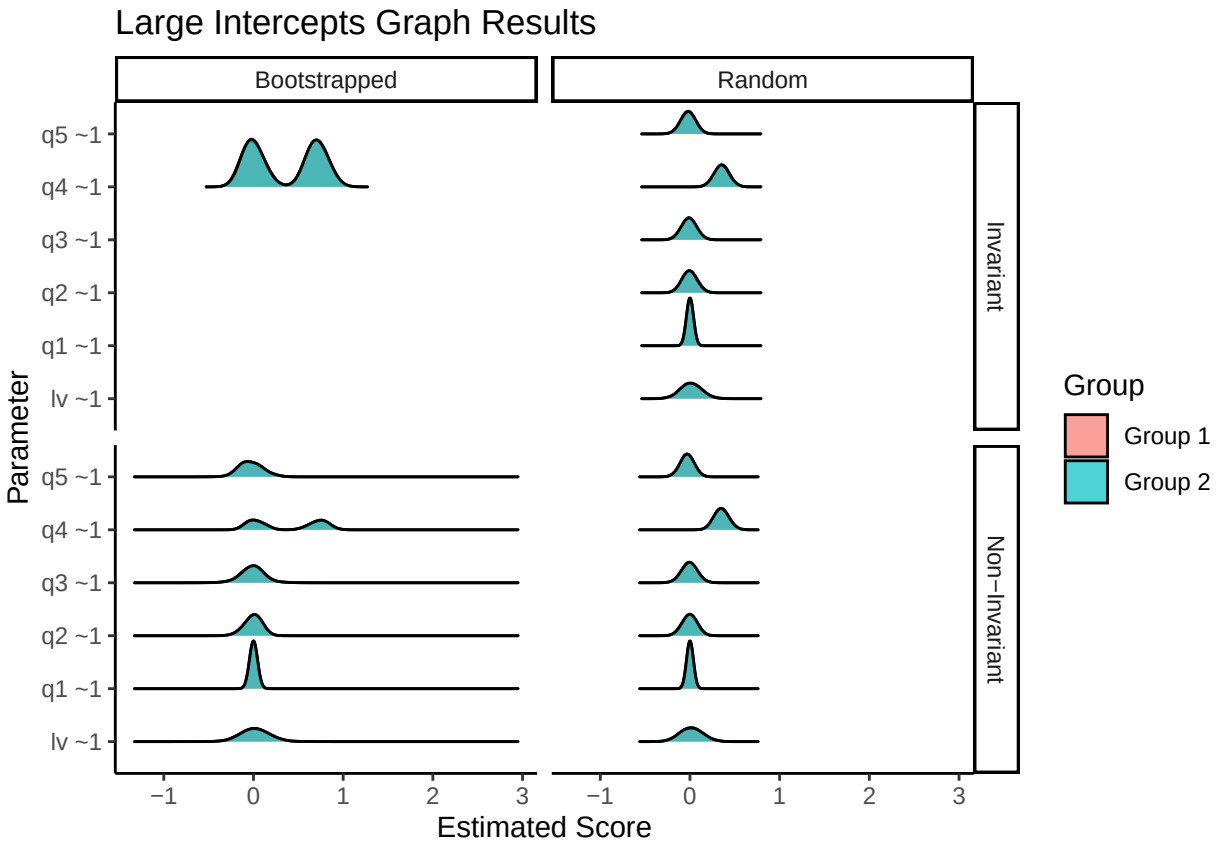
926


Small Intercepts Graph Results
Bootstrapped
Random
Invariant
Non−Invariant
Parameter
q5 ~1
q4 ~1
q3 ~1
q2 ~1
q1 ~1
lv ~1
−0.4
0.0
0.4
Estimated Score
Group
Group 1
Group 2

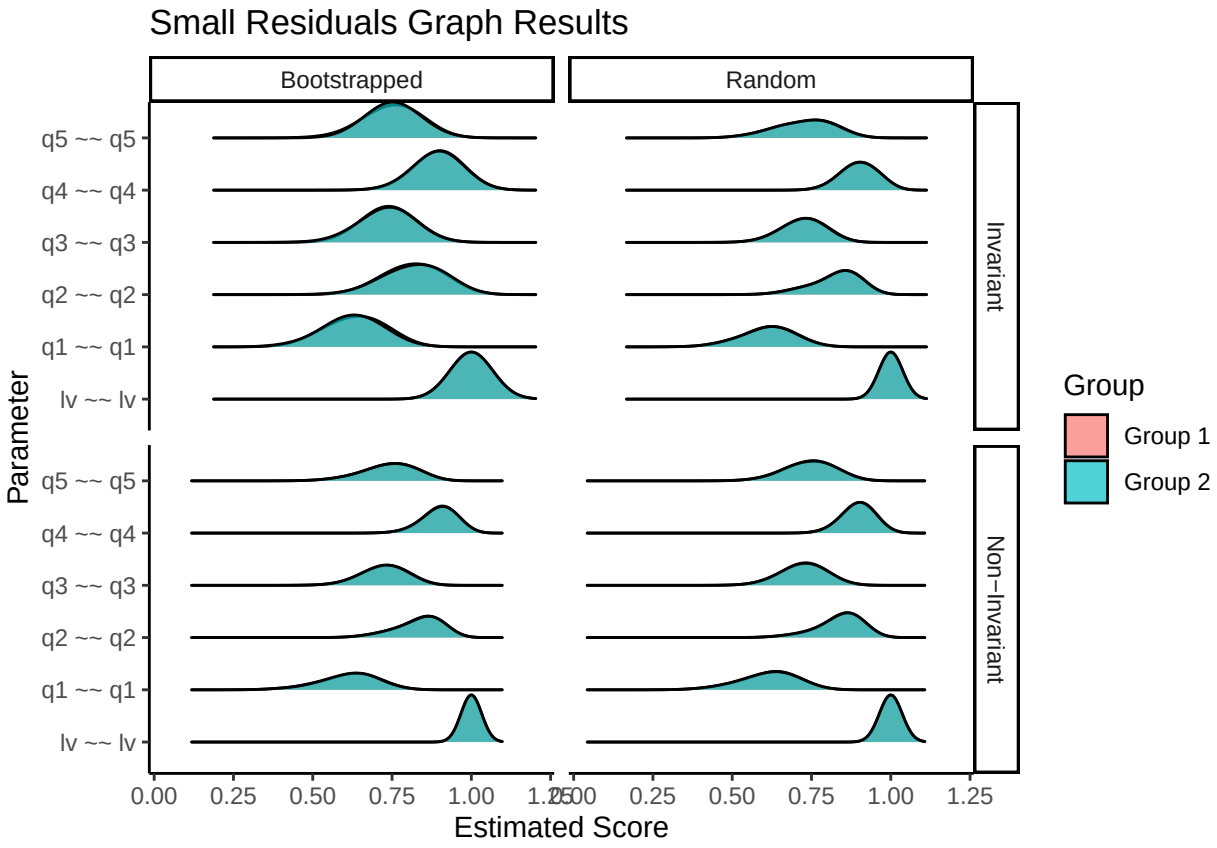
927


Medium Intercepts Graph Results
Bootstrapped
Random
Invariant
Non−Invariant
Parameter
q5 ~1
q4 ~1
q3 ~1
q2 ~1
q1 ~1
lv ~1
−0.5
0.0
0.5
1.0
Estimated Score
Group
Group 1
Group 2

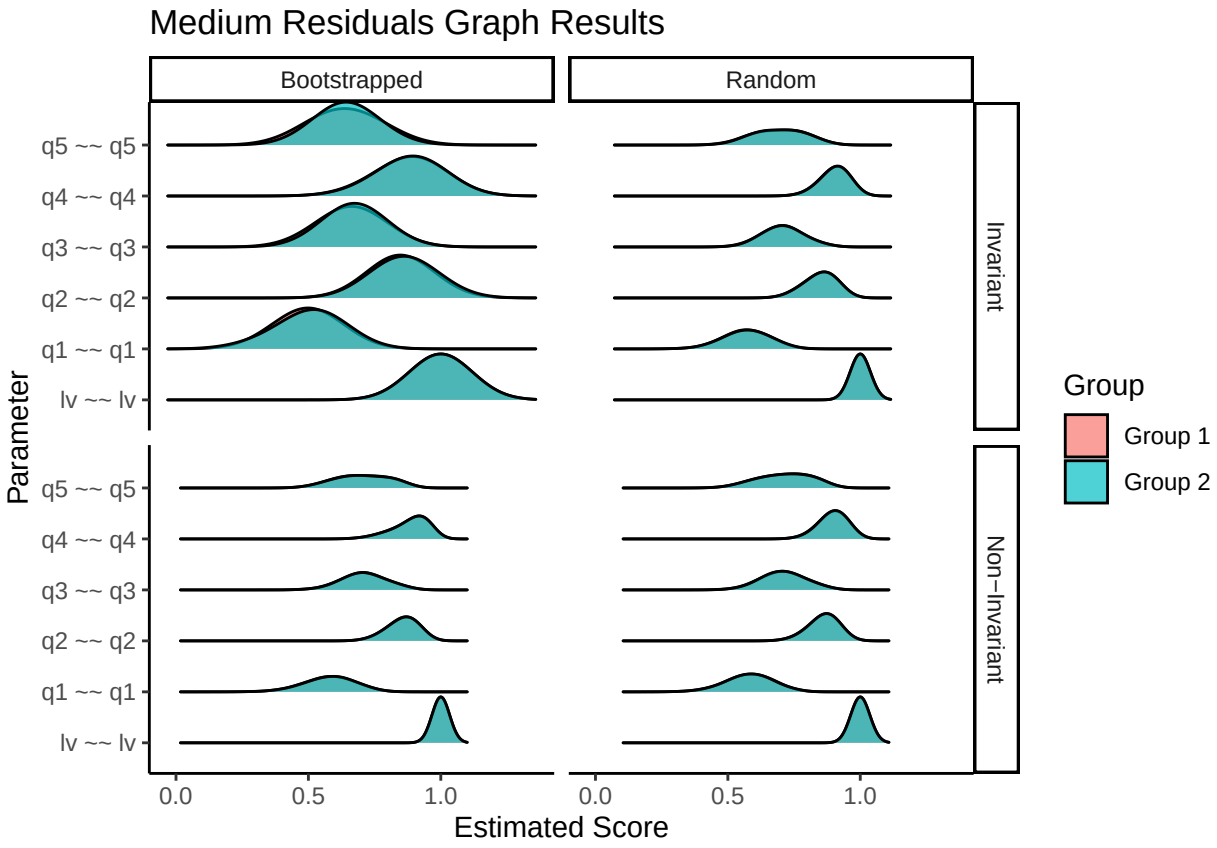
928


Large Intercepts Graph Results
Bootstrapped
Random
Invariant
Non−Invariant
Parameter
q5 ~1
q4 ~1
q3 ~1
q2 ~1
q1 ~1
Iv ~1
−1
0
1
2
3
Estimated Score
Group
Group 1
Group 2

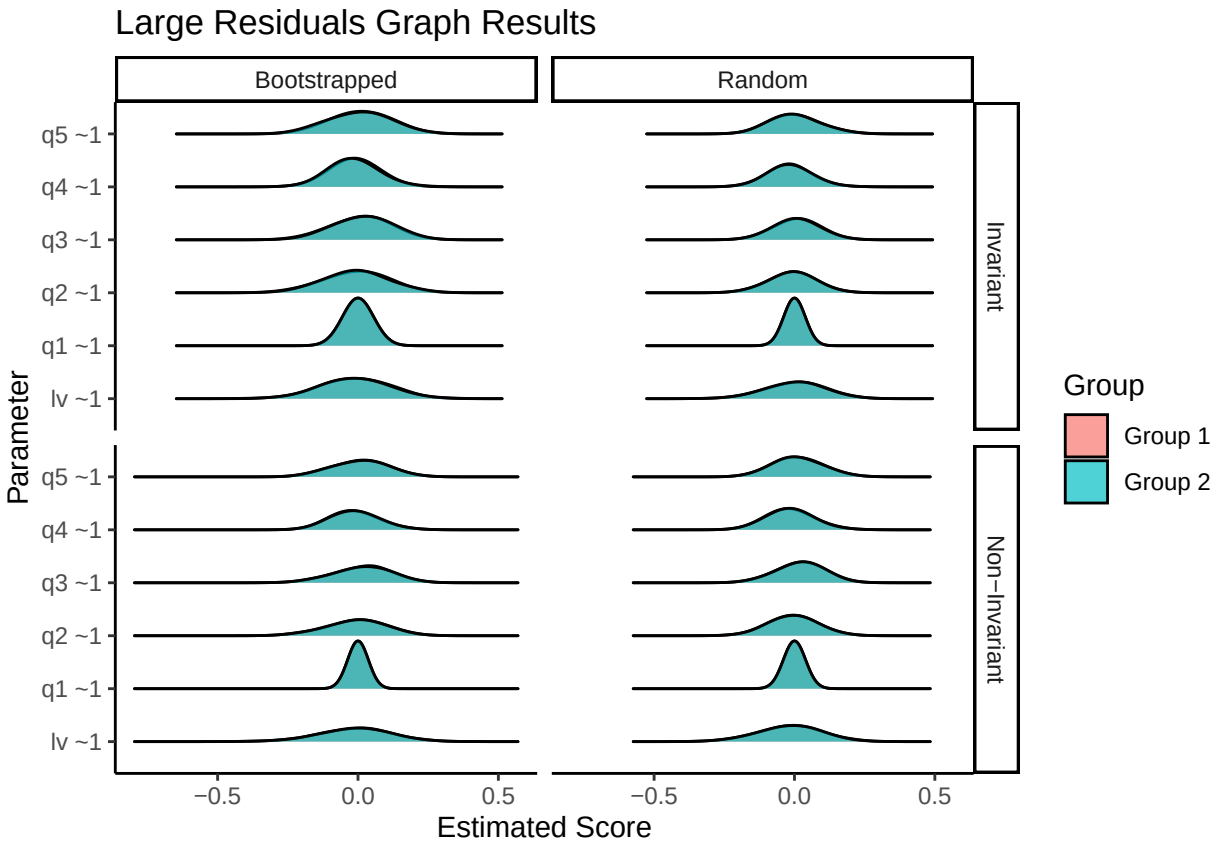
929


Small Residuals Graph Results
Bootstrapped
Random
Invariant
Non−Invariant
Parameter
q5 ~~ q5
q4 ~~ q4
q3 ~~ q3
q2 ~~ q2
q1 ~~ q1
lv ~~ lv
0.00
0.25
0.50
0.75
1.00
1.25
Estimated Score
Group
Group 1
Group 2

930


Medium Residuals Graph Results
Bootstrapped
Random
Invariant
Non−Invariant
Parameter
q5 ~~ q5
q4 ~~ q4
q3 ~~ q3
q2 ~~ q2
q1 ~~ q1
lv ~~ lv
0.0
0.5
1.0
Estimated Score
Group
Group 1
Group 2

931


Large Residuals Graph Results
Bootstrapped
Random
Invariant
Non−Invariant
Parameter
q5 ~1
q4 ~1
q3 ~1
q2 ~1
q1 ~1
lv ~1
−0.5
0.0
0.5
Estimated Score
Group
Group 1
Group 2

932