Permutation Tests of Configural Invariance: Respecifying Poor-Fitting Measurement Models

Terrence D. Jorgensen
University of Amsterdam
Configural Invariance

• Comparing groups on latent parameters requires equivalence of (some) measurement parameters

• For parameters to have same interpretation across groups, each group's population model must have the same form (configuration)
  – Configural invariance must therefore be assumed before any more restrictive levels of invariance can be tested
Testing Configural Invariance

• Significant likelihood ratio test (LRT) is used as evidence against the omnibus $H_0$ of invariance
  – $\chi^2$ test of exact overall fit for configural invariance
• Because $\chi^2$ is sensitive to negligible differences when $N$ is large, some researchers prefer to use fit indices (e.g., CFI or RMSEA) to evaluate approximate fit using a rule-of-thumb cutoff
  – e.g., CFI > .95 or RMSEA < .06 (Hu & Bentler, 1999) are thought to indicate good fit
Limitations of the Omnibus Test

• Rule-of-thumb cutoffs have high Type I errors at smaller $N$, especially with larger models

• **Limitation** of omnibus $H_0$: LRT of overall model fit confounds 2 sources of misfit:
  1. Group differences in population form
  2. Overall approximation discrepancy
     – Even if a hypothesized model only approximates a population process, it may be **equally** well specified for both groups
     – Equal fit within each group does not necessarily imply groups' population models are equal
Meade et al. (2008):  
- Grey cross-loadings close to 0 in population, not estimated  
- Good approximate fit (CFI > .99, RMSEA < .03)  
- Identical population configurations  
- $\chi^2$ will "over" reject as $N \to \infty$  
- CFI/RMSEA/SRMR will also over reject, as $N \to 0$
Hypothetical (straw man) unequal population configurations:

- Substantial cross-loading in Group 1 not estimated
- Minor secondary factors omitted from Group 2
  - Equivalent to residual covariances
- Model yields same $F_{ML}$ when fit to each group
- Identical population configurations
Follow-Up Tests?

• If the omnibus $H_0$ of configural invariance is rejected, what is best practice to modify the model?

• Modification index for each fixed parameter, estimate of $1-df \Delta \chi^2$ if it were freed

• Simultaneously or separately for groups?
  – Multiparameter Lagrange multipliers are available in lavaan ($\text{lavTestScore()}$ function)
  – For 4 groups, estimate a $4-df \Delta \chi^2$ if a parameter were freed in all groups
Follow-Up Tests?

• **Limitations:**
  – Multiple testing leads to inflated Type I error rates
  – Capitalization on chance leads to incorrect models
    • Overfitting to sample-specific nuances
  – Bonferroni correction can control Type I error rate, but more items/groups make it very conservative
Summary of Limitations

• LRT of overall model fit does not test the correct $H_0$ of group equivalence for configural model
• Fit indices have unknown distributions, and fixed cutoffs perform poorly
  – Inconsistent error rates across model sizes & $Ns$
• Inflated Type I errors for follow-up tests
  – Or low power with Bonferroni adjustment

• Permuting group membership can address each of these limitations
Illustration of Permutation Method

<table>
<thead>
<tr>
<th>Item</th>
<th>...</th>
<th>Item</th>
<th>Grouping</th>
<th>Original Indicator:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>...</td>
<td>k</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>...</td>
<td>86</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>...</td>
<td>41</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>...</td>
<td>22</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>...</td>
<td>57</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>16</td>
<td>...</td>
<td>24</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>...</td>
<td>38</td>
<td></td>
<td>2</td>
</tr>
</tbody>
</table>
## Illustration of Permutation Method

<table>
<thead>
<tr>
<th>Item</th>
<th>...</th>
<th>Item</th>
<th>Grouping Indicator:</th>
<th>Original</th>
<th>Permutation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>k</td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>...</td>
<td>86</td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>...</td>
<td>41</td>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>...</td>
<td>22</td>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>...</td>
<td>57</td>
<td></td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>...</td>
<td>24</td>
<td></td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>...</td>
<td>38</td>
<td></td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>
## Illustration of Permutation Method

<table>
<thead>
<tr>
<th>Item</th>
<th>...</th>
<th>Item</th>
<th>Grouping</th>
<th>Original</th>
<th>Permutation</th>
<th>Permutation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>k</td>
<td>Indicator:</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>...</td>
<td>86</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>...</td>
<td>41</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>...</td>
<td>22</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>...</td>
<td>57</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>...</td>
<td>24</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>...</td>
<td>38</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>
## Illustration of Permutation Method

<table>
<thead>
<tr>
<th>Item</th>
<th>...</th>
<th>Item</th>
<th>Grouping Indicator:</th>
<th>Original</th>
<th>Permutation 1</th>
<th>Permutation 2</th>
<th>Permutation 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>...</td>
<td>k</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>...</td>
<td>86</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>...</td>
<td>41</td>
<td></td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>...</td>
<td>22</td>
<td></td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>...</td>
<td>57</td>
<td></td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>16</td>
<td>...</td>
<td>24</td>
<td></td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>...</td>
<td>38</td>
<td></td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>
Permutation Randomization Test

- Permuting group membership can address each of these limitations
  1. Fit hypothesized model, save fit measures
  2. Randomly reassign individuals to groups
  3. Fit model(s) to the permuted (re)sample, saving the same fit measures
  4. Repeat 2 and 3 (e.g., 1000 times)
  5. Calculate $p$ values as the proportion of the empirical distribution that exceeds the fit measure from Step 1
Proposed Advice

• To test configural invariance, use permutation for the omnibus test

• If you fail to reject the omnibus $H_0$, modification indices should be inspected for freeing the same parameter across all groups simultaneously
  – Permutated Tukey method no better than Bonferroni

• If you reject the omnibus $H_0$, inspect group-specific minimum fit functions to reveal which group's model differs most from the hypothesized model. Then try testing configural invariance for a subset of groups
  – Or perhaps test any set of groups that have a similar minimum fit function
  – Simulation study forthcoming...
Type I Errors with Approximate Fit

• Replicated Meade et al.'s (2008) simulation
  – Population models had near-zero cross-loadings, so model fit approximately (not perfectly) well
    • Population CFI > .99, RMSEA < .03 in all conditions

• Experimentally manipulated:
  – Model size:
    • Number of factors (2 or 4)
    • Number of items per factor (4 or 8)
  – $N = 100, 200, 400, 800,$ and $1600$ per group

• 2000 replications per 20 conditions
Omnibus $H_0$: Type I errors (LRT)
Omnibus $H_0$: Type I errors (CFI)
Proposed Advice

• To test configural invariance, use permutation for the omnibus test

• If you fail to reject the omnibus $H_0$, modification indices should be inspected for freeing the same parameter across all groups simultaneously
  – Permuted Tukey method no better than Bonferroni

• If you reject the omnibus $H_0$, inspect group-specific minimum fit functions to reveal which group's model differs most from the hypothesized model. Then try testing configural invariance for a subset of groups
  – Or perhaps test any set of groups that have a similar minimum fit function
  – Simulation study forthcoming...
Equivalent but Poor Fit

• Monte Carlo simulation
  – Analysis model omitted 2 substantial residual covariances from data-generating model
    • Population CFI = .933, TLI = 0.875, RMSEA = 0.114
  – Held constant $N = 1000$, manipulated ratios across 4 groups
    • 30:70%, 40:60%, 50:50%, 60:40%, 70:30% in Groups $\{1,2\} : \{3,4\}$

• 4-$df$ Lagrange multipliers available by modifying lavaan's `lavTestScore()`
Omnibus $H_0$: Type I Errors
Proposed Advice

• To test configural invariance, use permutation for the omnibus test

• If you fail to reject the omnibus $H_0$, modification indices should be inspected for freeing the same parameter across all groups simultaneously
  – Permutated Tukey method no better than Bonferroni

• If you reject the omnibus $H_0$, inspect group-specific minimum fit functions to reveal which group's model differs most from the hypothesized model. Then try testing configural invariance for a subset of groups
  – Or perhaps test any set of groups that have a similar minimum fit function
  – Simulation study forthcoming...
Conclusions

• Only permutation provides a valid omnibus test of configural invariance
  – Overall and group-specific fit is a related but distinct issue, which can shed light on differences, if found

• When configural invariance appears tenable, but model modification is necessary, free the same parameter in each group
  – Choose parameters based on theory, manually fit competing models (actual $\Delta \chi^2$)
  – Or calculate multi-$df$ modification indices (estimated $\Delta \chi^2$)
Software

• The permutation method is implemented in the R package `semTools`
  – The function `permuteMeasEq()` accepts fitted `lavaan` models and additional arguments

• I also have examples using `Mplus`, via the R package `MplusAutomation`
  – Feel free to contact me for the R script, or for the working manuscript  
    
    **T.D.Jorgensen@uva.nl** 

• Questions?
References

