

# Multiple Group Models & Measurement Invariance

## Theory Construction and Statistical Modeling



**Utrecht  
University**

Kyle M. Lang

Department of Methodology & Statistics  
Utrecht University

# Outline

---

## Multiple Group Models

Between-Group Comparisons

## Measurement Invariance

Testing MI with **lavaan**

## Moderation via Multiple Group SEM



# Multiple Group Models

---

Multiple group models allow us to simultaneously run subgroup analyses.

1. Fit group-specific models to subsets of the data.
2. Constrain parameters across groups to test for between-group differences.

To fix ideas, let's start with an example of multiple group path analysis.

- Outcome: sw
- Predictors: overt and covert
- Group: sex



# Example

---

```
library(dplyr)
library(lavaan)

## Read in the data:
dat1 <- read.table("../data/popular_regr.txt",
                    na.strings = c("-99", "-999"),
                    header = TRUE) %>%
mutate(sex = factor(gender, labels = c("male", "female"))) %>%
filter(!is.na(sex))

## Define the model syntax:
mod1 <- 'sw ~ 1 + overt + covert'

## Fit the model:
fit1 <- sem(mod1, data = dat1, group = "sex")
```

# Example

```
partSummary(fit1, 1:4)
```

lavaan 0.6.16 ended normally after 1 iteration

Estimator	ML	
Optimization method	NLMINB	
Number of model parameters	8	
Number of observations per group:	Used	Total
male	756	824
female	581	660

Model Test User Model:

Test statistic	0.000
Degrees of freedom	0
Test statistic for each group:	
male	0.000
female	0.000

# Example

```
partSummary(fit1, 8:11)
```

Group 1 [male]:

Regressions:

	Estimate	Std.Err	z-value	P(> z )
sw ~				
overt	-0.278	0.059	-4.719	0.000
covert	-0.497	0.039	-12.818	0.000

Intercepts:

	Estimate	Std.Err	z-value	P(> z )
.sw	4.930	0.085	57.938	0.000

Variances:

	Estimate	Std.Err	z-value	P(> z )
.sw	0.336	0.017	19.442	0.000

# Example

```
partSummary(fit1, 13:16)
```

Group 2 [female]:

Regressions:

	Estimate	Std.Err	z-value	P(> z )
sw ~				
overt	-0.232	0.081	-2.871	0.004
covert	-0.558	0.045	-12.295	0.000

Intercepts:

	Estimate	Std.Err	z-value	P(> z )
.sw	5.062	0.106	47.703	0.000

Variances:

	Estimate	Std.Err	z-value	P(> z )
.sw	0.318	0.019	17.044	0.000

# Comparing Nested Models

---

We usually fit multiple group models to test for between-group differences in the model parameters.

- To compare parameters across groups, we need to label the parameters.

```
## Define the model syntax with parameter labels:
mod1 <- '
sw ~ c("b0m", "b0f") * 1 +
      c("b1m", "b1f") * overt +
      c("b2m", "b2f") * covert
'

## Fit the model:
fit1 <- sem(mod1, data = dat1, group = "sex")
```



# Comparing Nested Models

```
partSummary(fit1, 1:4)
```

lavaan 0.6.16 ended normally after 1 iteration

Estimator	ML	
Optimization method	NLMINB	
Number of model parameters	8	
Number of observations per group:	Used	Total
male	756	824
female	581	660

Model Test User Model:

Test statistic	0.000
Degrees of freedom	0
Test statistic for each group:	
male	0.000
female	0.000

# Comparing Nested Models

```
partSummary(fit1, 8:11)
```

Group 1 [male]:

Regressions:

		Estimate	Std.Err	z-value	P(> z )
sw ~					
overt	(b1m)	-0.278	0.059	-4.719	0.000
covert	(b2m)	-0.497	0.039	-12.818	0.000

Intercepts:

		Estimate	Std.Err	z-value	P(> z )
.sw	(b0m)	4.930	0.085	57.938	0.000

Variances:

		Estimate	Std.Err	z-value	P(> z )
.sw		0.336	0.017	19.442	0.000

# Comparing Nested Models

```
partSummary(fit1, 13:16)
```

Group 2 [female]:

Regressions:

		Estimate	Std.Err	z-value	P(> z )
sw	~				
overt	(b1f)	-0.232	0.081	-2.871	0.004
covert	(b2f)	-0.558	0.045	-12.295	0.000

Intercepts:

		Estimate	Std.Err	z-value	P(> z )
.sw	(b0f)	5.062	0.106	47.703	0.000

Variances:

		Estimate	Std.Err	z-value	P(> z )
.sw		0.318	0.019	17.044	0.000

# Comparing Nested Models

---

## Hypotheses

- $H_0$ : Unconstrained Model Fit = Constrained Model Fit
- $H_1$ : Unconstrained Model Fit  $\neq$  Constrained Model Fit

## Large p-value

- No significant difference between the model fits
- No evidence that the coefficients differ across groups

## Small p-value

- Significant difference between the model fits
- Evidence that the parameter differs across groups



# Comparing Nested Models

---

```
## Test equality of regression slopes:  
lavTestWald(fit1, constraints = 'b1m == b1f; b2m == b2f')  
  
$stat  
[1] 1.035523  
  
$df  
[1] 2  
  
$p.value  
[1] 0.5958527  
  
$se  
[1] "standard"
```

# Comparing Nested Models

---

We can also fit the restricted model manually.

- Give parameters the same label to constrain them to equality.

```
## Define the restricted model syntax:
mod0 <- '
sw ~ c("b0", "b0") * 1 +
      c("b1", "b1") * overt +
      c("b2", "b2") * covert
'

## Fit the model:
fit0 <- sem(mod0, data = dat1, group = "sex")
```

# Comparing nested models

```
partSummary(fit0, 1:4)
```

lavaan 0.6.16 ended normally after 11 iterations

Estimator	ML	
Optimization method	NLMINB	
Number of model parameters	8	
Number of equality constraints	3	
Number of observations per group:	Used	Total
male	756	824
female	581	660

Model Test User Model:

Test statistic	5.782
Degrees of freedom	3
P-value (Chi-square)	0.123
Test statistic for each group:	
male	2.042
female	3.741

# Comparing nested models

```
partSummary(fit0, 8:11)
```

Group 1 [male]:

Regressions:

		Estimate	Std.Err	z-value	P(> z )
sw ~					
overt	(b1)	-0.289	0.046	-6.307	0.000
covert	(b2)	-0.506	0.028	-17.774	0.000

Intercepts:

	Estimate	Std.Err	z-value	P(> z )
.sw	(b0) 4.990	0.066	75.137	0.000

Variances:

	Estimate	Std.Err	z-value	P(> z )
.sw	0.336	0.017	19.442	0.000



# Comparing nested models

```
partSummary(fit0, 13:16)
```

Group 2 [female]:

Regressions:

		Estimate	Std.Err	z-value	P(> z )
sw ~					
overt	(b1)	-0.289	0.046	-6.307	0.000
covert	(b2)	-0.506	0.028	-17.774	0.000

Intercepts:

		Estimate	Std.Err	z-value	P(> z )
.sw	(b0)	4.990	0.066	75.137	0.000

Variances:

		Estimate	Std.Err	z-value	P(> z )
.sw		0.320	0.019	17.044	0.000

# Comparing Nested Models

---

```
## Test equality of regression slopes:
```

```
lavTestLRT(fit0, fit1)
```

Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
fit1	0	2319.8	2361.4	0.0000				
fit0	3	2319.6	2345.6	5.7824	5.7824	0.037248	3	0.1227

# MEASUREMENT INVARIANCE

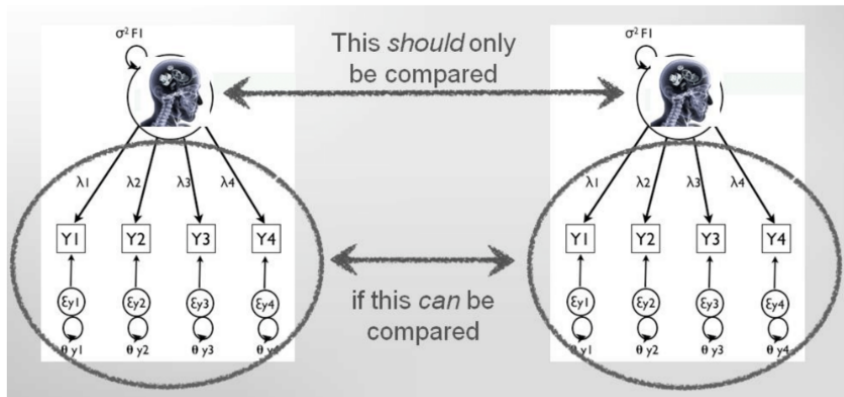
# Measurement Invariance

---

When fitting multiple group models with latent factors, we need to establish *measurement invariance* across groups.

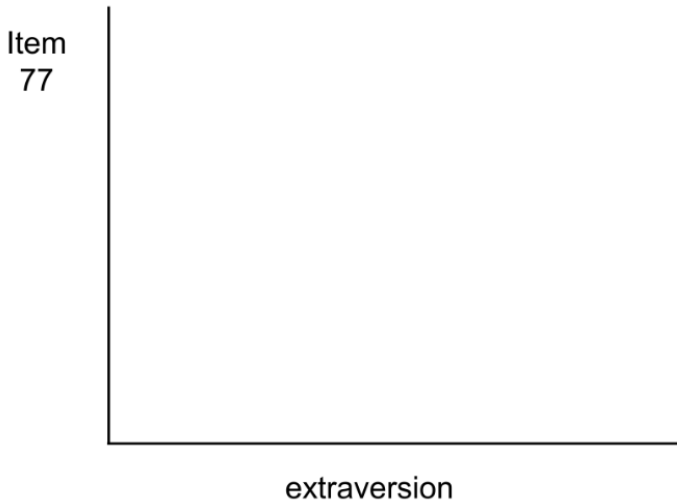
- Construct validity:
  - Is the model measuring the same thing in both (all) groups?
- Can we make a fair comparison between groups?
  - Did the groups understand the questions in the same way?
- Same latent score should result in the same observed scores.
  - Equal slopes (factor loadings)
  - Equal intercepts (item means)

# Visualizing Measurement Invariance



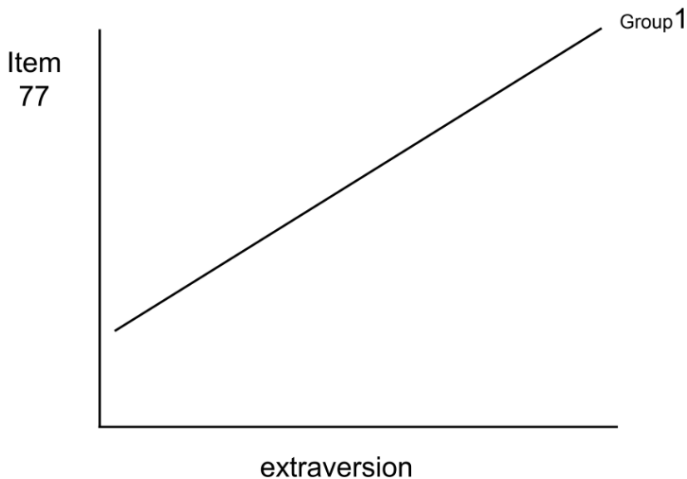
# Visualizing Measurement Invariance

---

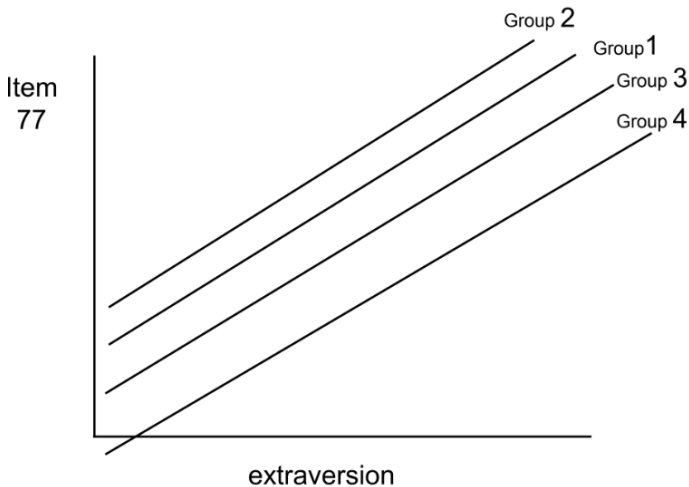


# Visualizing Measurement Invariance

---

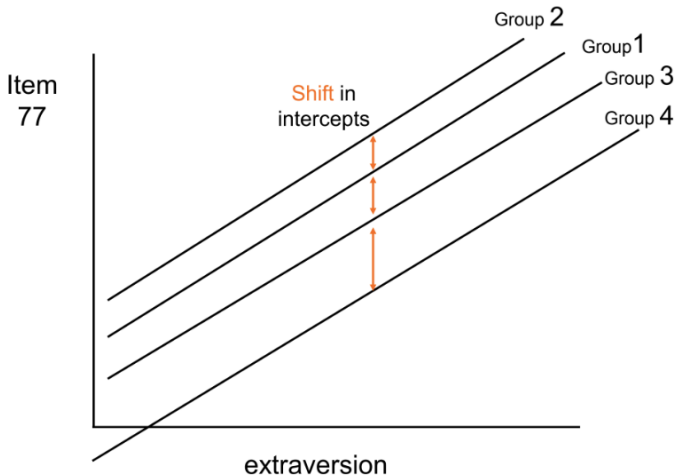


# Visualizing Measurement Invariance

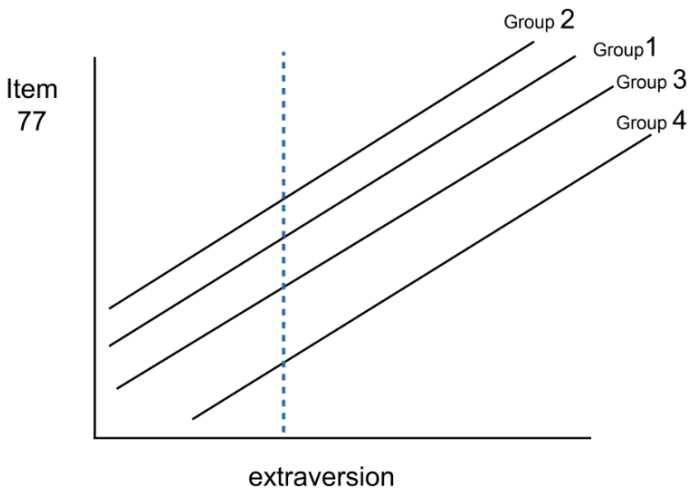




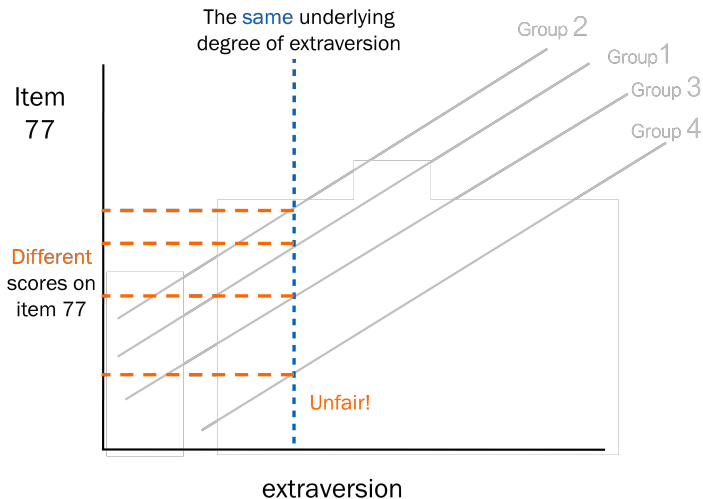
# Visualizing Measurement Invariance



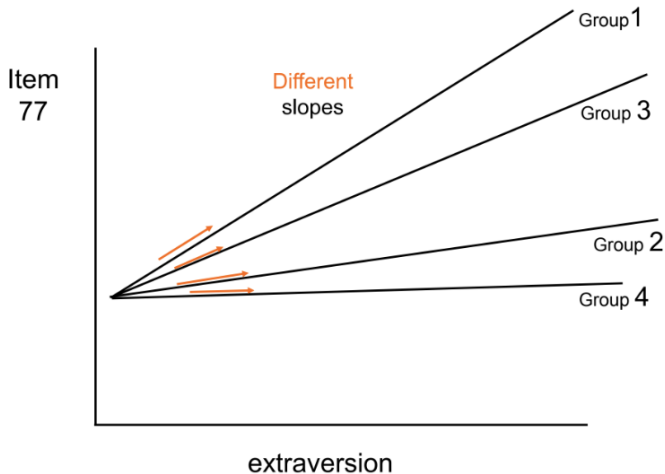
# Visualizing Measurement Invariance



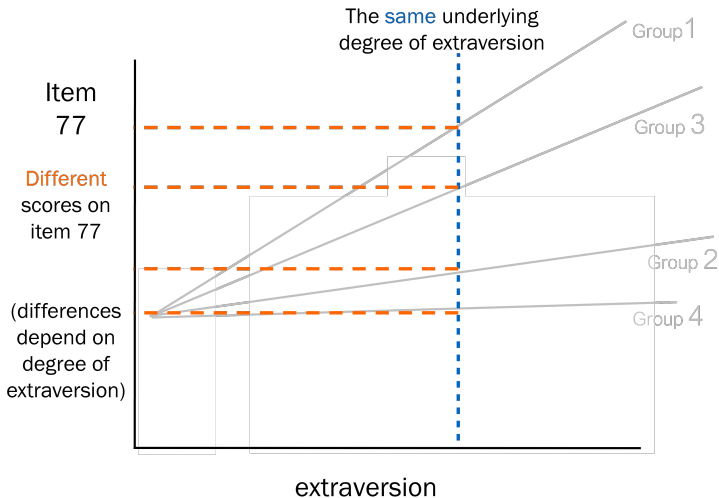
# Visualizing Measurement Invariance



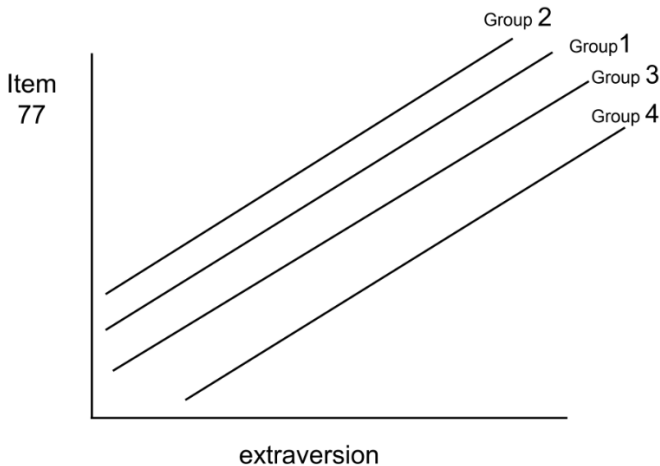
# Visualizing Measurement Invariance



# Visualizing Measurement Invariance



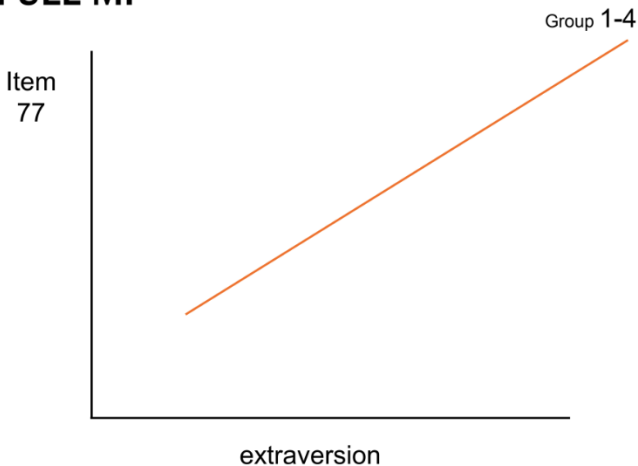
# Visualizing Measurement Invariance



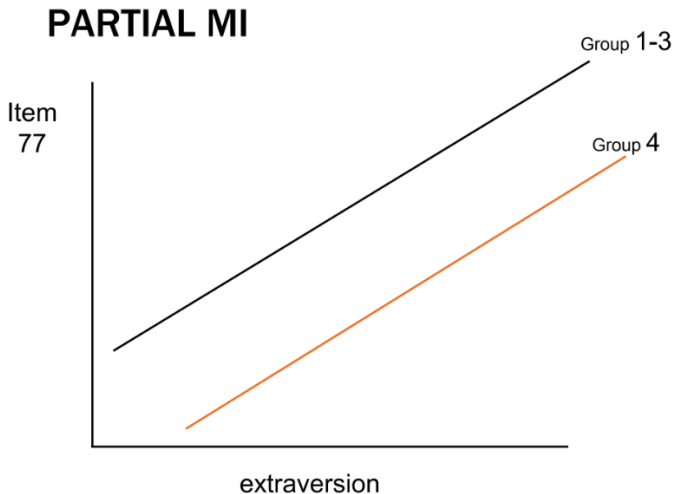
# Visualizing Measurement Invariance

---

## FULL MI



# Visualizing Measurement Invariance



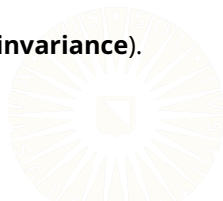


# Measurement Invariance Procedure

---

We establish measurement invariance by sequentially testing more restrictive models.

1. Test model separately for each group (**configural invariance**).
  - The measurement models must fit the data in all groups.
2. Test equality of loadings across groups (**metric/weak invariance**).
  - Must be equal to compare linear associations between groups.
3. Test equality of intercepts across groups (**scalar/strong invariance**).
  - Must be equal to compare means across groups.
4. Test equality of measurement error variances (**strict invariance**).
  - Controversial
  - Generally considered to be too restrictive



# Example

---



Carin Hill  
Leon Jackson  
Deon Meiring  
J. Aleweyn Nel

Ian Rothmann  
Michael Temane  
Velichko H. Valchev  
Fons J. R. van de Vijver

Nel, J. A., Valchev, V. H., Rothmann, S., van de Vijver, F. J. R., Meiring, D., & de Bruin, G. P. (2012). Exploring the personality structure in the 11 languages of South Africa. *Journal of Personality*, 80, 915–948.

# Example

---

```
## Read in the data:
sapi <- read.table("../data/sapi.txt",
                  header = TRUE,
                  na.strings = "-999") %>%
mutate(sex = factor(Gender, labels = c("male", "female"))) %>%
filter(!is.na(sex))

## Define the model syntax:
cfaMod <- '
having_fun  =~ Q77 + Q84 + Q196
being_liked =~ Q44 + Q63 + Q98
'
```

# Example

---

```
## Fit the models:
```

```
configFit <- cfa(cfaMod,  
  data = sapi,  
  std.lv = TRUE,  
  group = "sex",  
  missing = "FIML")
```

```
weakFit <- cfa(cfaMod,  
  data = sapi,  
  std.lv = TRUE,  
  group = "sex",  
  group.equal = "loadings",  
  missing = "FIML")
```

```
strongFit <- cfa(cfaMod,  
  data = sapi,  
  std.lv = TRUE,  
  group = "sex",  
  group.equal = c("loadings", "intercepts"),  
  missing = "FIML")
```

# Example

The `compareFit()` function from the **semTools** package runs several different model comparison tests.

```
library(semTools)
compareFit(configFit, weakFit, strongFit) %>% summary()
```

```
##### Nested Model Comparison #####
```

Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
configFit	16	15354	15540	85.083				
weakFit	20	15348	15514	86.483	1.3992	0.0000	4	0.8443
strongFit	24	15347	15494	93.862	7.3791	0.0415	4	0.1172

# Example

```
##### Model Fit Indices #####
      chisq df pvalue rmsea   cfi   tli   srmr       aic       bic
configFit 85.083↑ 16   .000 .094 .939 .886 .039↑ 15354.393 15540.159
weakFit   86.483 20   .000 .082 .942↑ .913 .039 15347.793 15514.004
strongFit 93.862 24   .000 .077↑ .939 .923↑ .041 15347.172↑ 15493.829↑

##### Differences in Fit Indices #####
      df rmsea   cfi   tli   srmr       aic       bic
weakFit - configFit  4 -0.011  0.002 0.026 0.000 -6.601 -26.155
strongFit - weakFit  4 -0.005 -0.003 0.011 0.002 -0.621 -20.175
```

Metric/Weak invariance holds.

- The weakly invariant model fits just as well as the configurally invariant model.

Scalar/Strong invariance holds.

- The strongly invariant model fits just as well as the weakly invariant model.

# What if invariance fails?

---

## Configural Invariance

- The within-group measurement models don't hold (for some groups).

## Weak Invariance

- The model defines valid constructs in each group.
- We cannot make any comparisons across groups.

## Strong Invariance

- We can compare linear associations across groups.
- We cannot compare means across groups.

## Strict Invariance

- Doesn't matter



# Partial Invariance

---

When weak or strong invariance fail, we can sometimes establish satisfactory *partial invariance*.

- Free the fewest possible number of constraints.
- Try to achieve good fit without freeing *too many* constraints.
- We can claim that the measurement models are comparable enough.





# MODERATION VIA MULTIPLE GROUP SEM

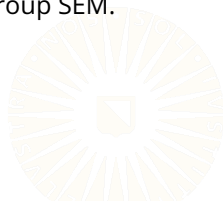


# Multiple Group SEM for Moderation

---

When our moderator is a categorical variable, we can use multiple group CFA/SEM to test for moderation.

- Categorical moderators define groups.
- Significant moderation with categorical moderators implies between-group differences in the focal effect.
- We can directly test these hypotheses with multiple group SEM.



# Example

---

We must first establish measurement invariance.

```
## Read the data and subset to only high school and college graduates:
bfi <- readRDS("../data/bfiData2.rds") %>%
  filter(educ %in% c("highSchool", "college"))

## Specify the (configurally invariance) measurement model:
mod0 <- '
agree =~ A1 + A2 + A3 + A4 + A5
open  =~ O1 + O2 + O3 + O4 + O5
'

## Estimate the unrestricted model:
out0 <- cfa(mod0, data = bfi, std.lv = TRUE, group = "educ")
```

# Example

---

```
### Fit the configurally invariant model:
configFit <- cfa(mod0, data = bfi, std.lv = TRUE, group = "educ")

## Fit the weakly invariant model:
weakFit <- cfa(mod0,
  data = bfi,
  std.lv = TRUE,
  group = "educ",
  group.equal = "loadings")

## Fit the strongly invariant model:
strongFit <- cfa(mod0,
  data = bfi,
  std.lv = TRUE,
  group = "educ",
  group.equal = c("loadings", "intercepts")
)
```

# Example

```
compareFit(configFit, weakFit, strongFit) %>% summary()
```

```
##### Nested Model Comparison #####
```

```
Chi-Squared Difference Test
```

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)	
configFit	68	75336	75694	418.25					
weakFit	76	75358	75669	455.70	37.451	0.055855	8	9.505e-06	***
strongFit	84	75461	75726	575.15	119.446	0.108654	8	< 2.2e-16	***

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# Example

```
##### Model Fit Indices #####
      chisq df pvalue rmsea   cfi   tli  srmr       aic       bic
configFit 418.253† 68   .000 .066  .906† .875  .046† 75336.292† 75693.810
weakFit   455.705 76   .000 .065† .898  .879† .050  75357.744 75669.130†
strongFit 575.151 84   .000 .070  .868  .858  .056  75461.190 75726.445

##### Differences in Fit Indices #####
      df rmsea   cfi   tli  srmr       aic       bic
weakFit - configFit  8 -0.001 -0.008  0.004 0.004  21.451 -24.680
strongFit - weakFit  8  0.005 -0.030 -0.021 0.006 103.446  57.315
```

Invariance doesn't really hold here, but we'll move forward for the sake of pedagogical demonstration.

# Example

Specifying (unconstrained) structural parameters models moderation of those parameters by the grouping factor.

- Each group get's their own estimate of the structural effects.

```
## Specify a structural model:
mod3 <- '
agree =~ A1 + A2 + A3 + A4 + A5
open  =~ O1 + O2 + O3 + O4 + O5

agree ~ open
'

## Estimate the model with strong invariance constraints:
out3 <- sem(mod3,
  data = bfi,
  std.lv = TRUE,
  group = "educ",
  group.equal = c("loadings", "intercepts")
)
```

# Example

---

Each group gets their own slope estimate.

```
partSummary(out3, c(8, 10, 14, 16))
```

Group 1 [highSchool]:

Regressions:

	Estimate	Std.Err	z-value	P(> z )
agree ~				
open	-0.321	0.040	-7.957	0.000

Group 2 [college]:

Regressions:

	Estimate	Std.Err	z-value	P(> z )
agree ~				
open	-0.203	0.051	-3.972	0.000



# Example

---

To test for moderation, we constrain the focal effects to be equal across groups and conduct a model comparison test.

```
## Specify the restricted model:
mod4 <- '
agree =~ A1 + A2 + A3 + A4 + A5
open  =~ O1 + O2 + O3 + O4 + O5

agree ~ c(beta, beta) * open
'

## Estimate the model:
out4 <- sem(mod4,
  data = bfi,
  std.lv = TRUE,
  group = "educ",
  group.equal = c("loadings", "intercepts")
)
```

# Example

---

Now, the slopes are equal in both groups.

```
partSummary(out4, c(8, 10, 14, 16))
```

Group 1 [highSchool]:

Regressions:

		Estimate	Std.Err	z-value	P(> z )
agree ~					
open	(beta)	-0.278	0.032	-8.621	0.000

Group 2 [college]:

Regressions:

		Estimate	Std.Err	z-value	P(> z )
agree ~					
open	(beta)	-0.278	0.032	-8.621	0.000

# Example

We can use a  $\Delta\chi^2$  test to test for moderation.

- A significant loss of fit would imply moderation.

```
## Do a chi-squared difference test for moderation:
```

```
anova(out3, out4)
```

Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
out3	84	75461	75726	575.15				
out4	85	75463	75722	578.59	3.435	0.045426	1	0.06383 .

---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

In this case, we don't have evidence of moderation.

# Example

---

We could also fit an analagous model using OLS regression.

```
readRDS("../data/bfiData1.rds") %>%  
  filter(educ %in% c("highSchool", "college")) %$%  
  lm(agree ~ open * educ) %>%  
  partSummary(-(1:2))
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	3.24965	0.12321	26.376	<2e-16
open	0.27115	0.03134	8.652	<2e-16
educcollege	0.03975	0.22849	0.174	0.862
open:educcollege	-0.05654	0.05856	-0.965	0.334

Residual standard error: 0.6972 on 2356 degrees of freedom

Multiple R-squared: 0.05314, Adjusted R-squared: 0.05194

F-statistic: 44.08 on 3 and 2356 DF, p-value: < 2.2e-16

# Probing Multiple Group Moderation

---

Testing moderation with multiple group SEM has several advantages.

- Remove measurement error from the estimates
- Test measurement invariance
- All simple effects are directly estimated in the unrestricted model



# Simple Slopes & Intercepts

Group 1 [highSchool]:

Regressions:

	Estimate	Std.Err	z-value	P(> z )
agree ~				
open	-0.321	0.040	-7.957	0.000

Intercepts:

	Estimate	Std.Err	z-value	P(> z )
.agree	0.000			

Group 2 [college]:

Regressions:

	Estimate	Std.Err	z-value	P(> z )
agree ~				
open	-0.203	0.051	-3.972	0.000

Intercepts:

	Estimate	Std.Err	z-value	P(> z )
.agree	0.170	0.056	3.058	0.002