Is my survey biased? The importance of measurement invariance

Yusuke Kuroki
Sunny Moon
November 9th, 2017

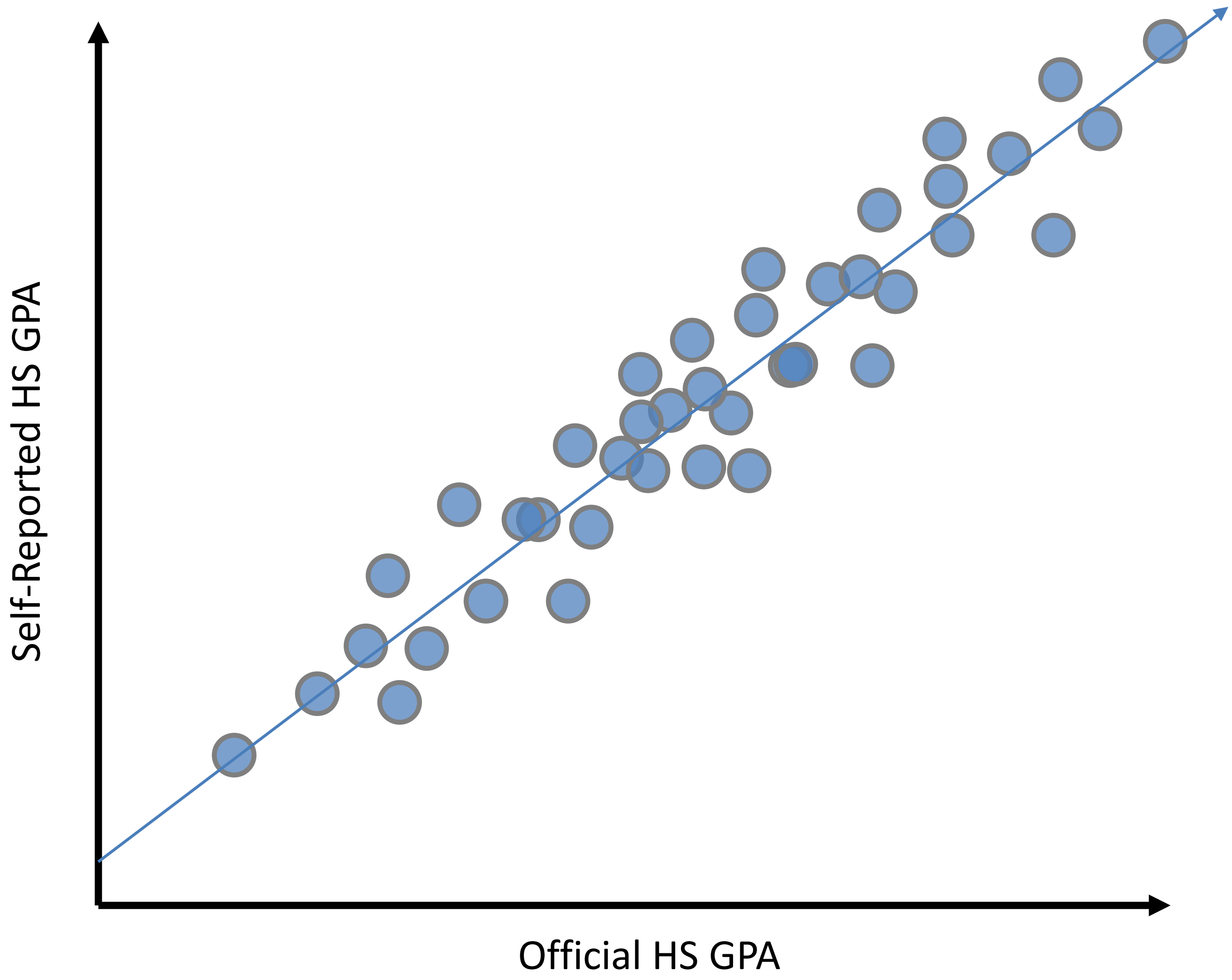
Measurement Invariance

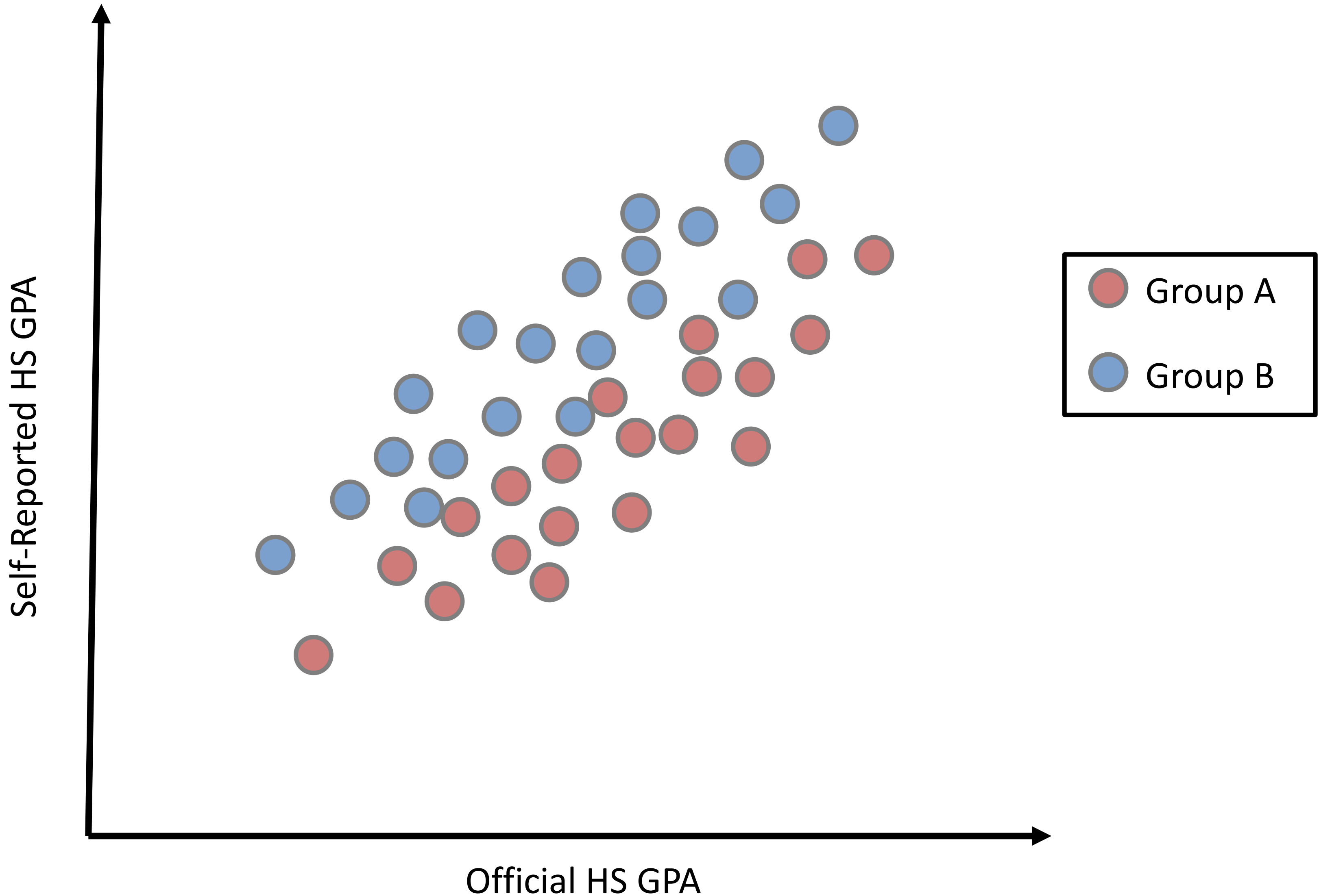
- **Measurement invariance:** the same construct is being measured across groups (or across time).
- This is prerequisite to comparing groups but rarely tested.

“real” vs. Self-report

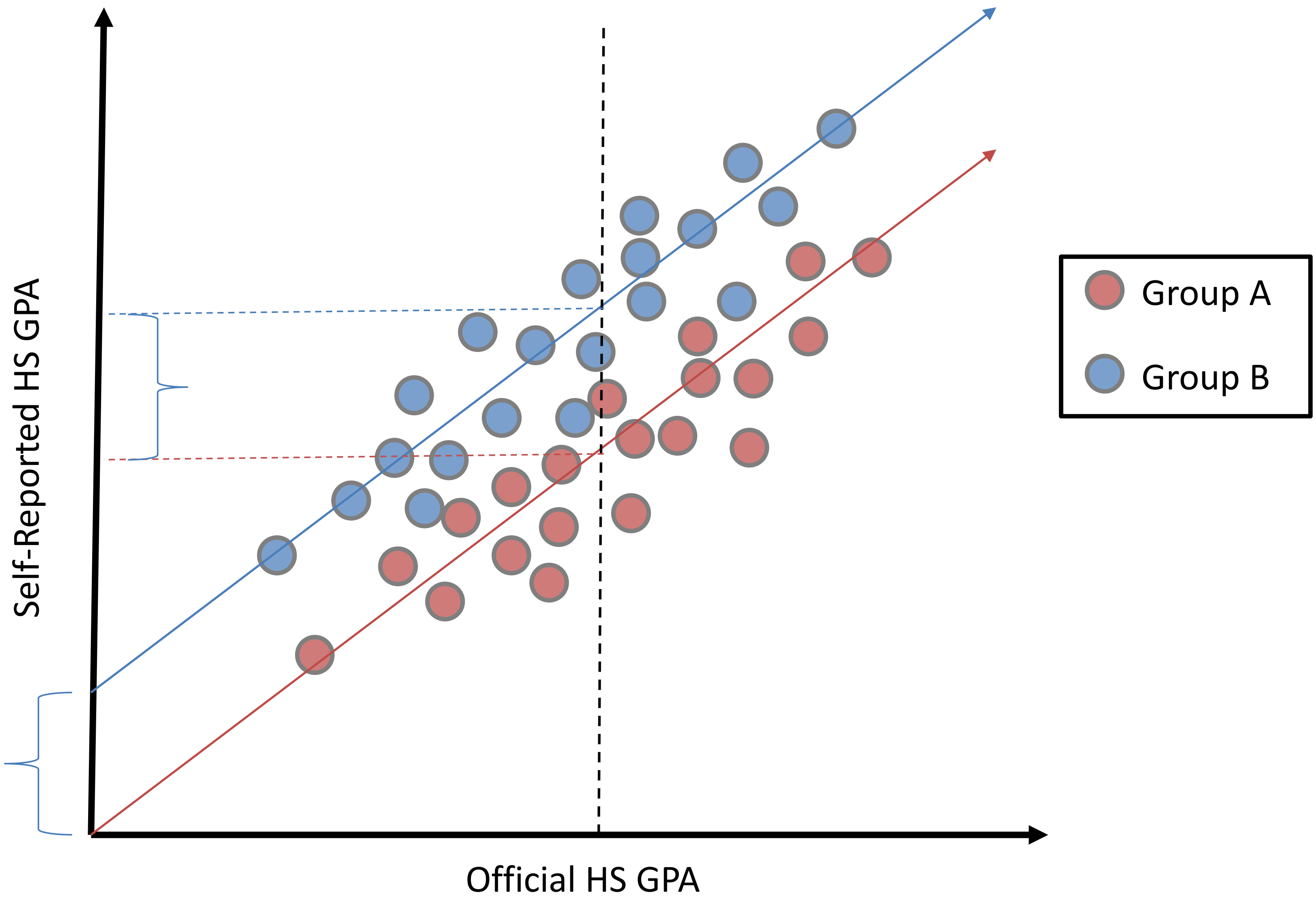
- A professor teaches a course and wants to know if high school GPA is a significant predictor of grades in the course.
- With respect to High school GPA, should a professor asks for the official data from IR or can the professor rely on students’ self-report HS GPA?
- This depends on the relationship between the official and self-reported GPA…

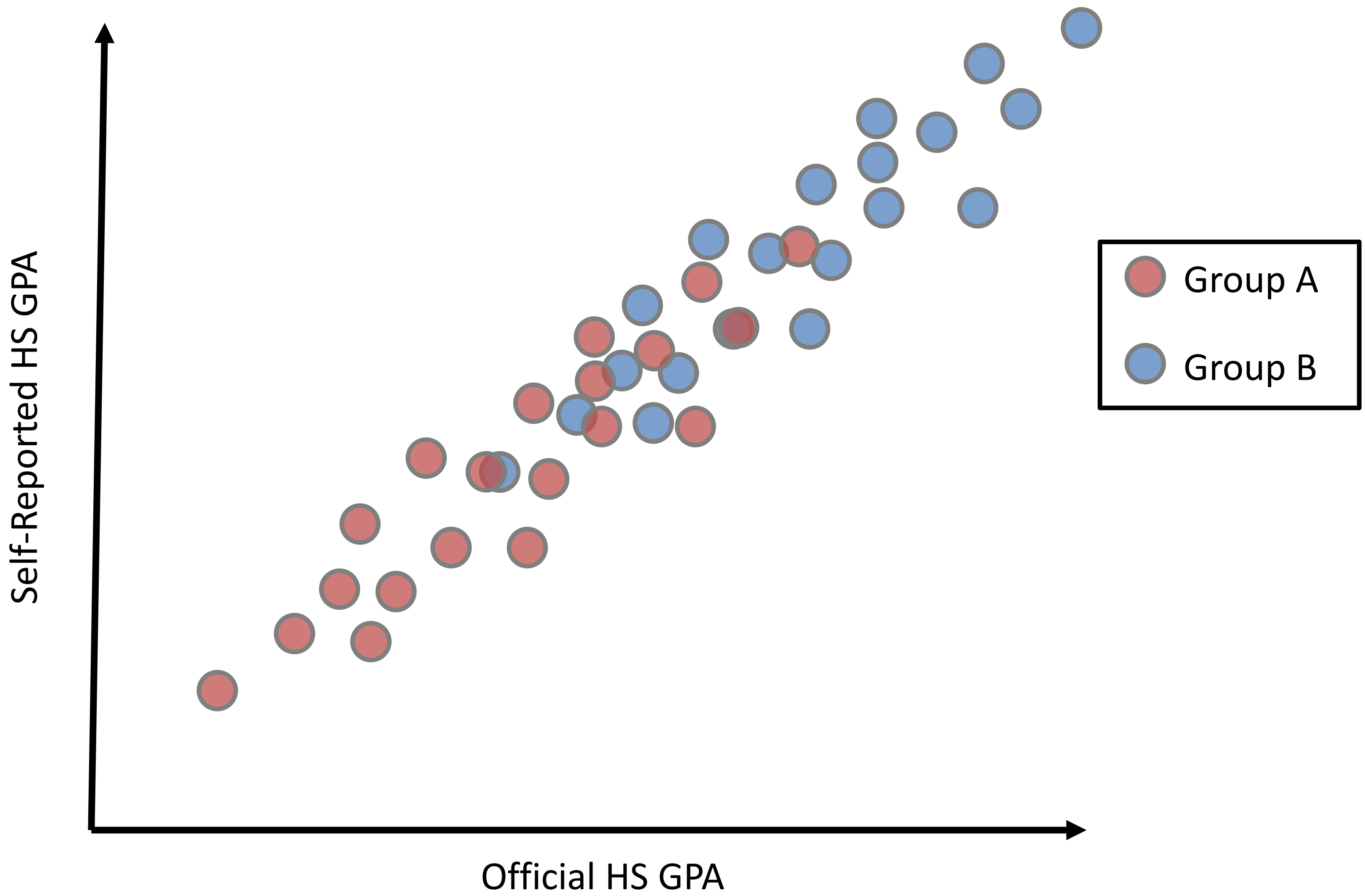
Official HS GPA → Self-report HS GPA

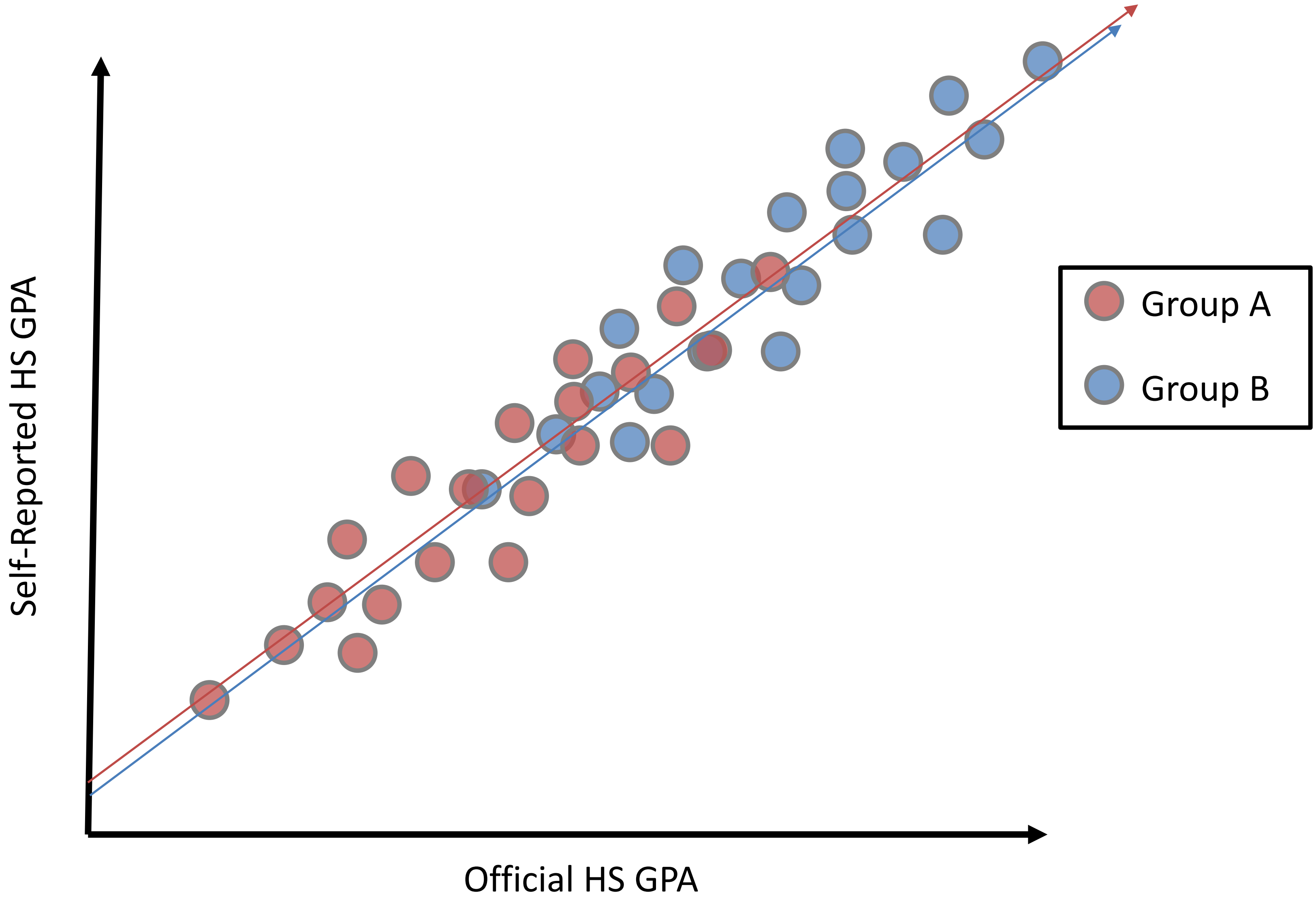

Self-Reported HS GPA
Official HS GPA

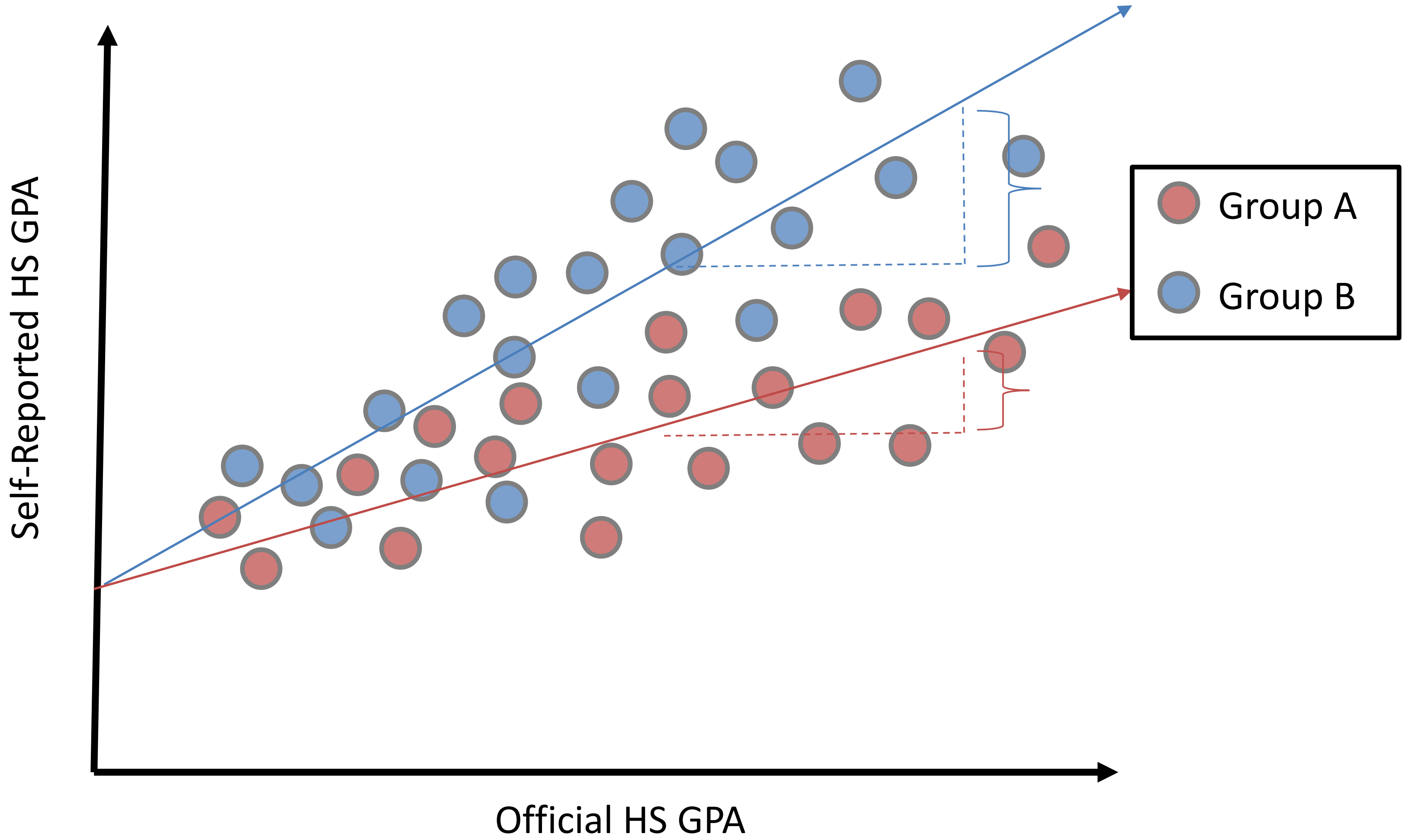


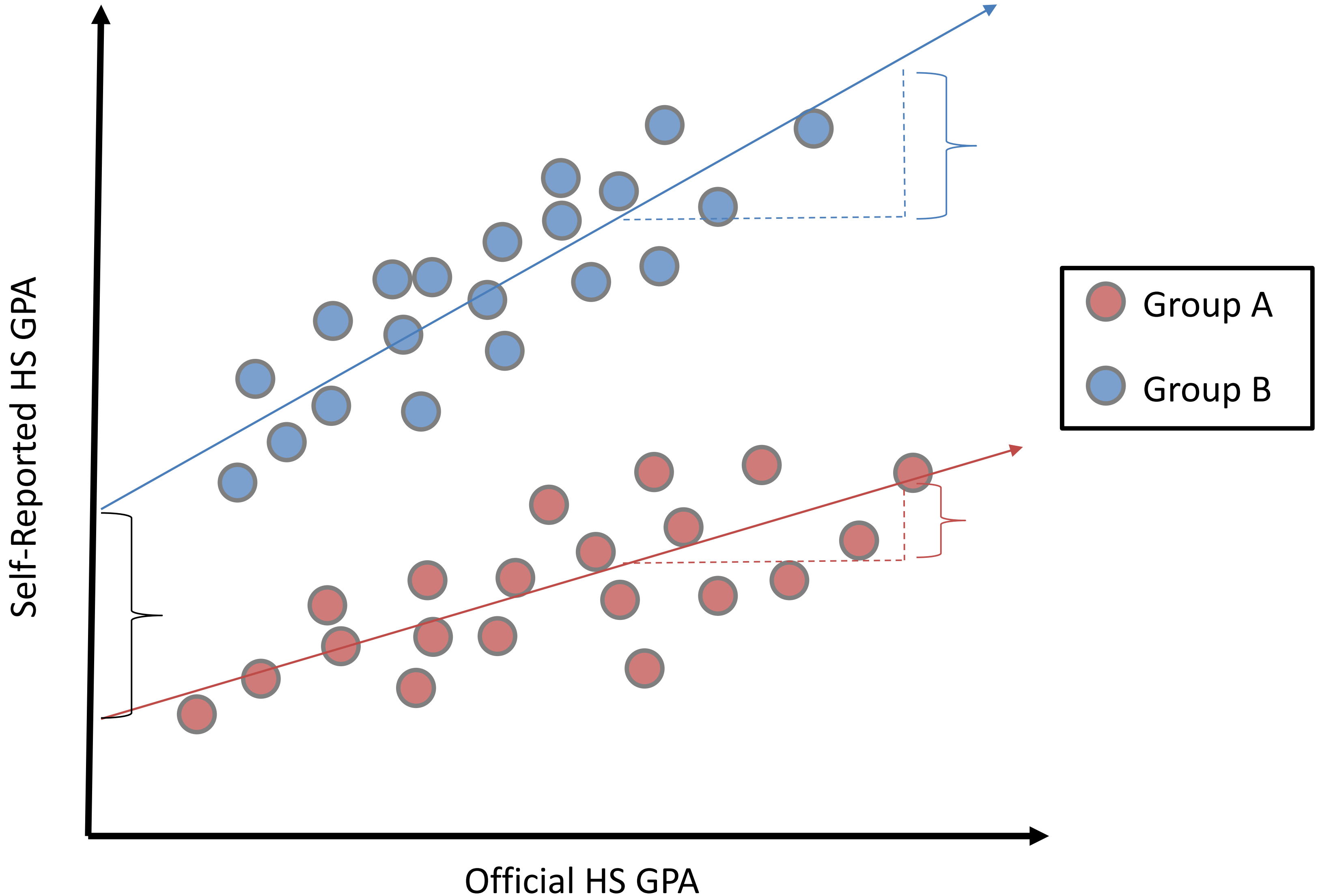
Self-Reported HS GPA
Official HS GPA
Group A
Group B


Self-Reported HS GPA
Official HS GPA
Group A
Group B


Self-Reported HS GPA
Official HS GPA
Group A
Group B

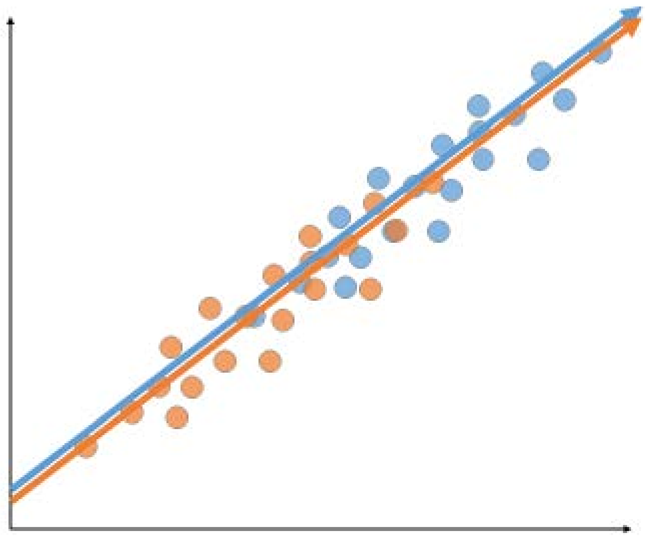

Self-Reported HS GPA
Official HS GPA
Group A
Group B


Self-Reported HS GPA
Official HS GPA
Group A
Group B

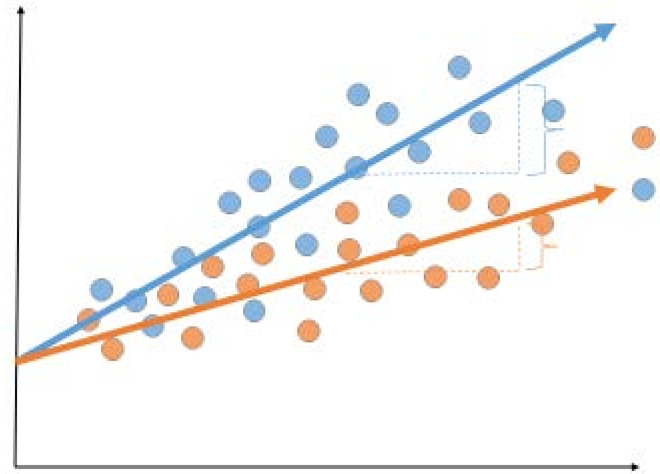


Self-Reported HS GPA
Official HS GPA
Group A
Group B

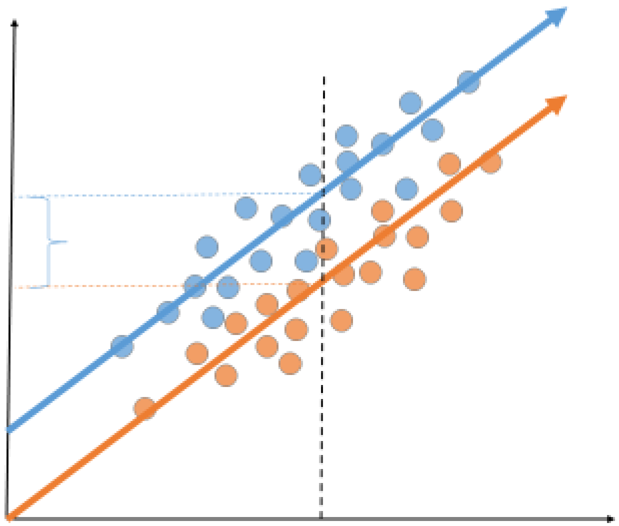
equal slopes and intercepts



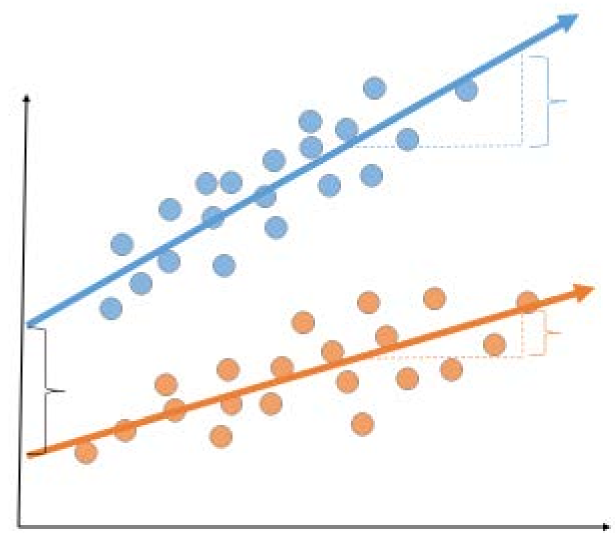
diff slopes, equal intercepts



equal slopes, diff. intercepts



diff. slopes, diff. intercepts

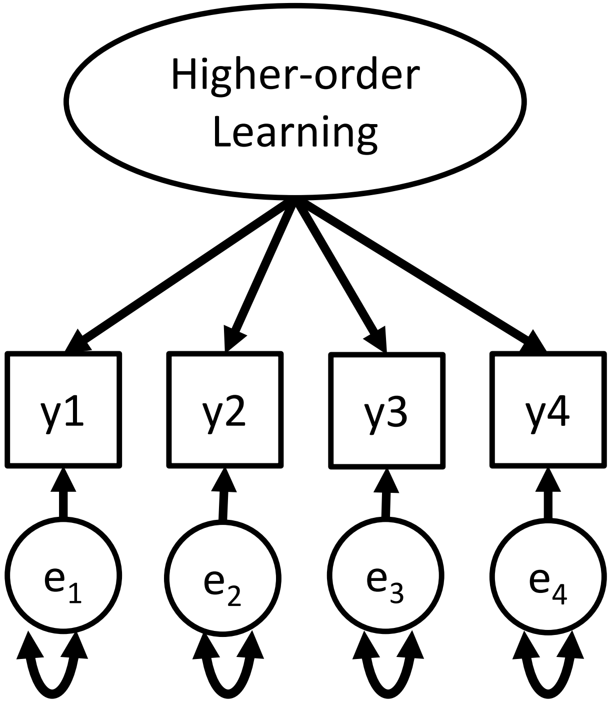


Confirmatory Factor Analysis (CFA)

- CFA allows us to examine the relationship between latent and observed variables.
- For example, in NSSE*, Higher-order learning* is a latent factor measured by four items.

During the current school year, how much has your coursework emphasized the following:

1. Applying facts, theories, or methods to practical problems or new situations
2. Analyzing an idea, experience, or line of reasoning in depth by examining its parts
3. Evaluating a point of view, decision, or information source
4. Forming a new idea or understanding from various pieces of information



Confirmatory Factor Analysis (CFA)

- CFA allows us to examine the relationship between *latent* and *observed* variables.
  - Observed – directly measured.
  - Latent – not directly measured but inferred from the observed variables.

- A path diagram is a popular way to describe your theory (though it may not be accurate)

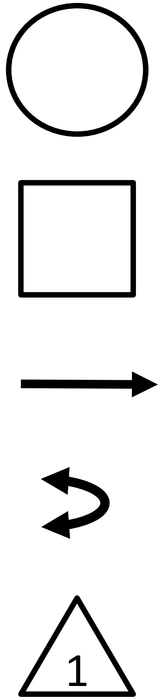
latent

observed

Regression or loading

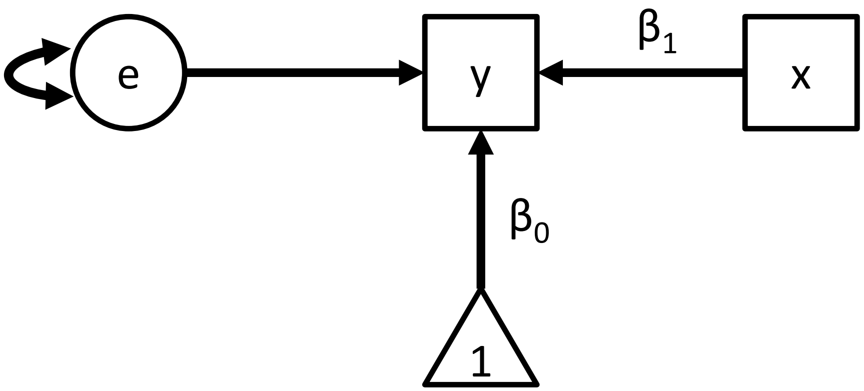
(co)variance

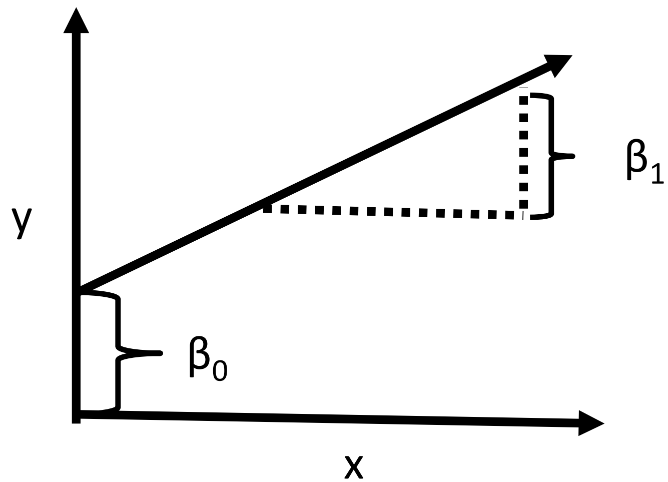
Vector of 1s
(used for intercepts)



Simple regression

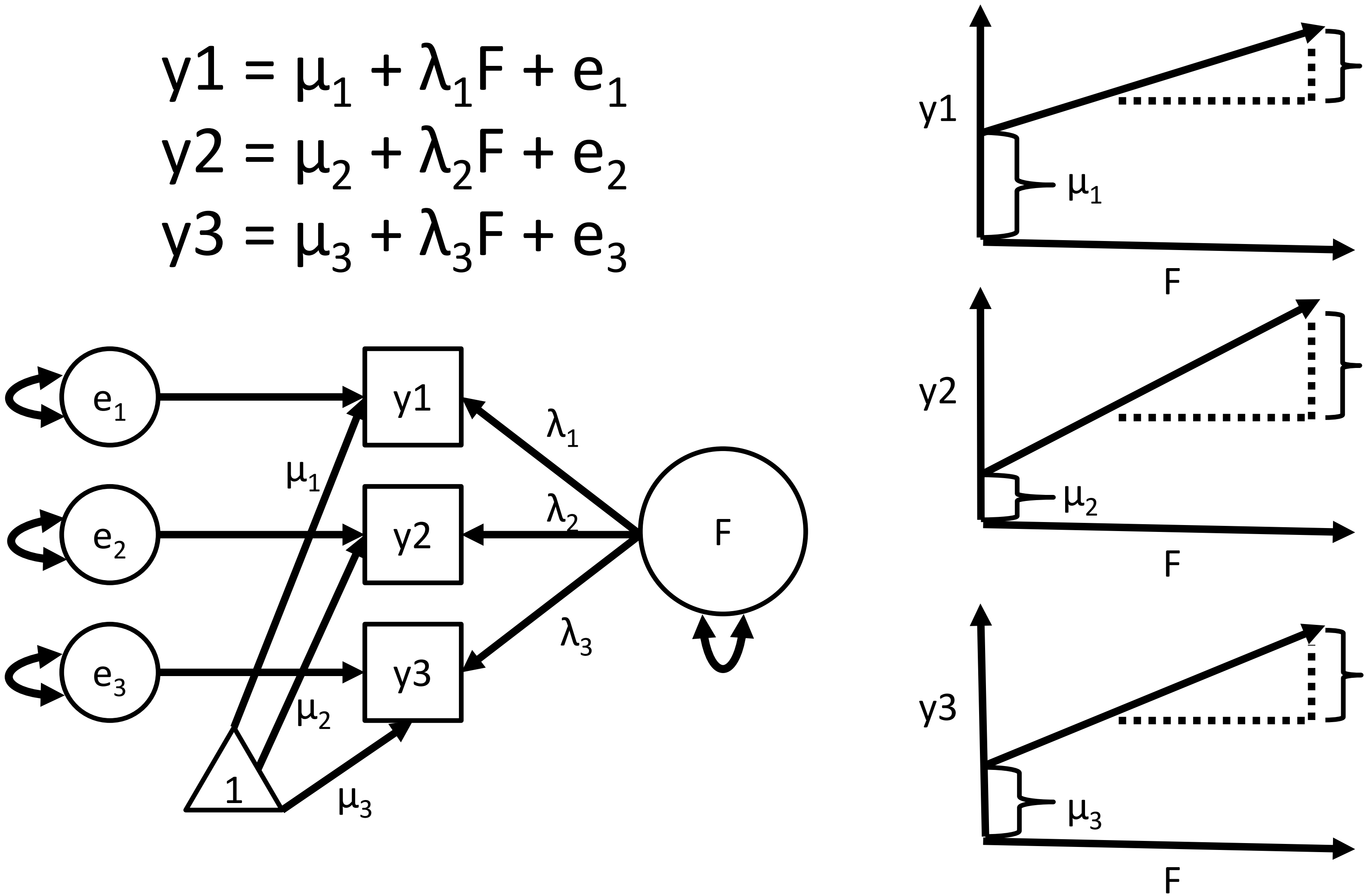
$$y = \beta_0 + \beta_1 x + e$$





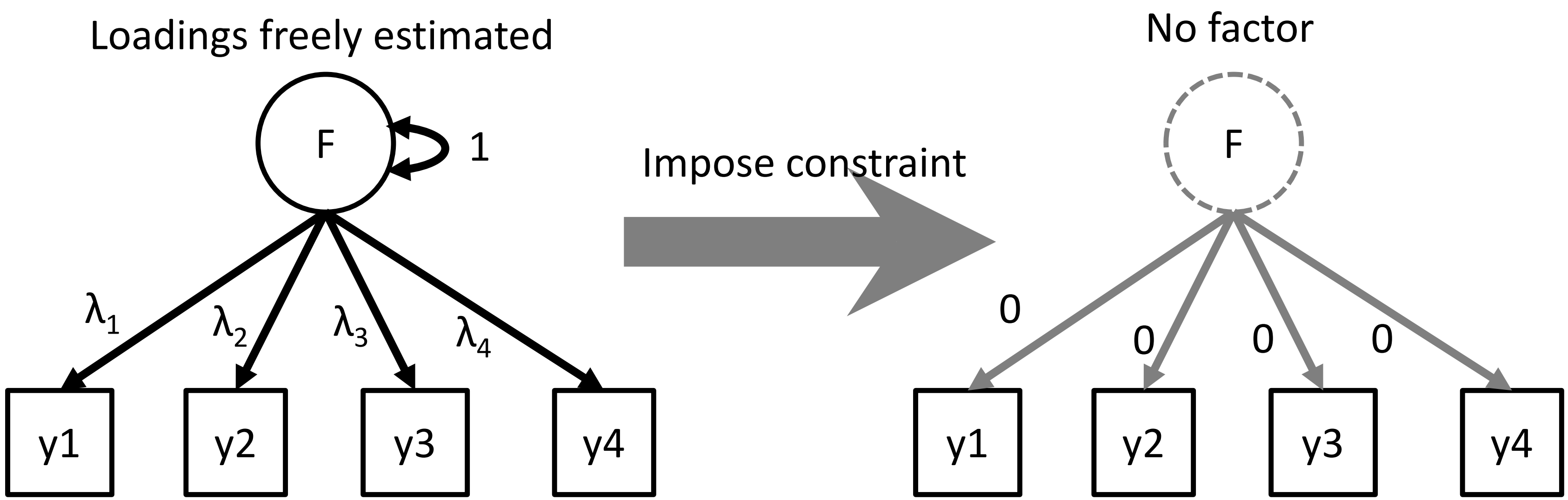
One factor model

$$y1 = \mu_1 + \lambda_1 F + e_1$$
$$y2 = \mu_2 + \lambda_2 F + e_2$$
$$y3 = \mu_3 + \lambda_3 F + e_3$$



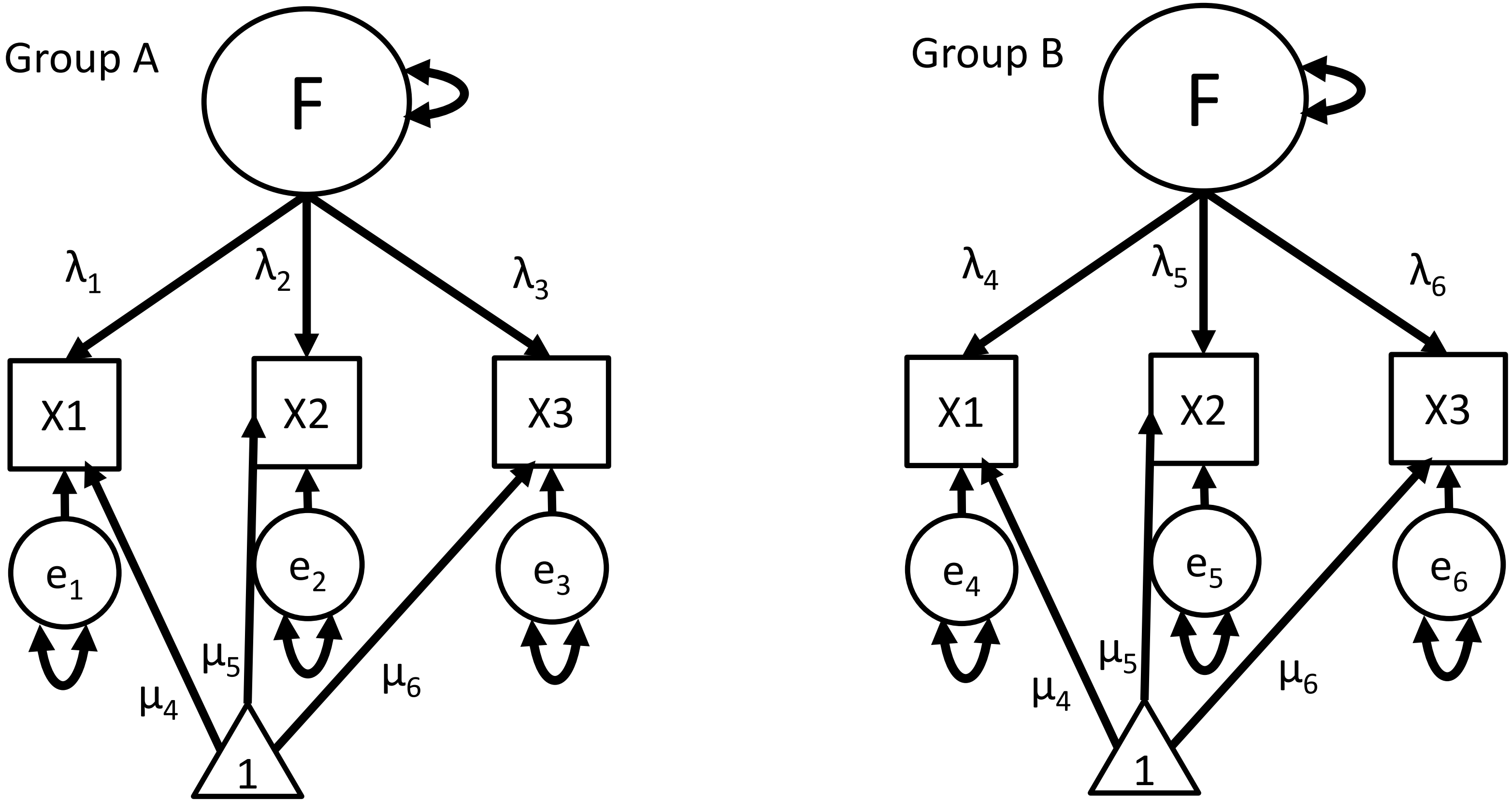
Model evaluation

- RMSEA: < 0.05 = good, .05 to .08 = acceptable
- Comparative Fit Index (CFI): > 0.95 = good, > 0.90 = acceptable
- Tucker-Lewis Index (TLI): > 0.95 = good
- SRMR: < 0.08 good
- Chi-square: this can be used compare models, if they are nested



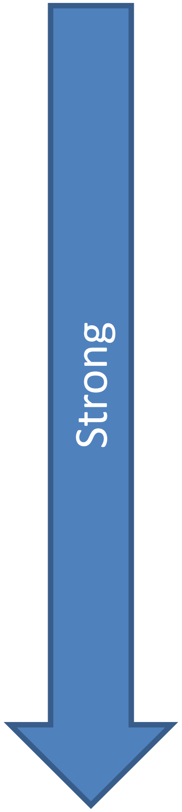
Multiple-group CFA

This allows simultaneously estimate parameters for multiple groups



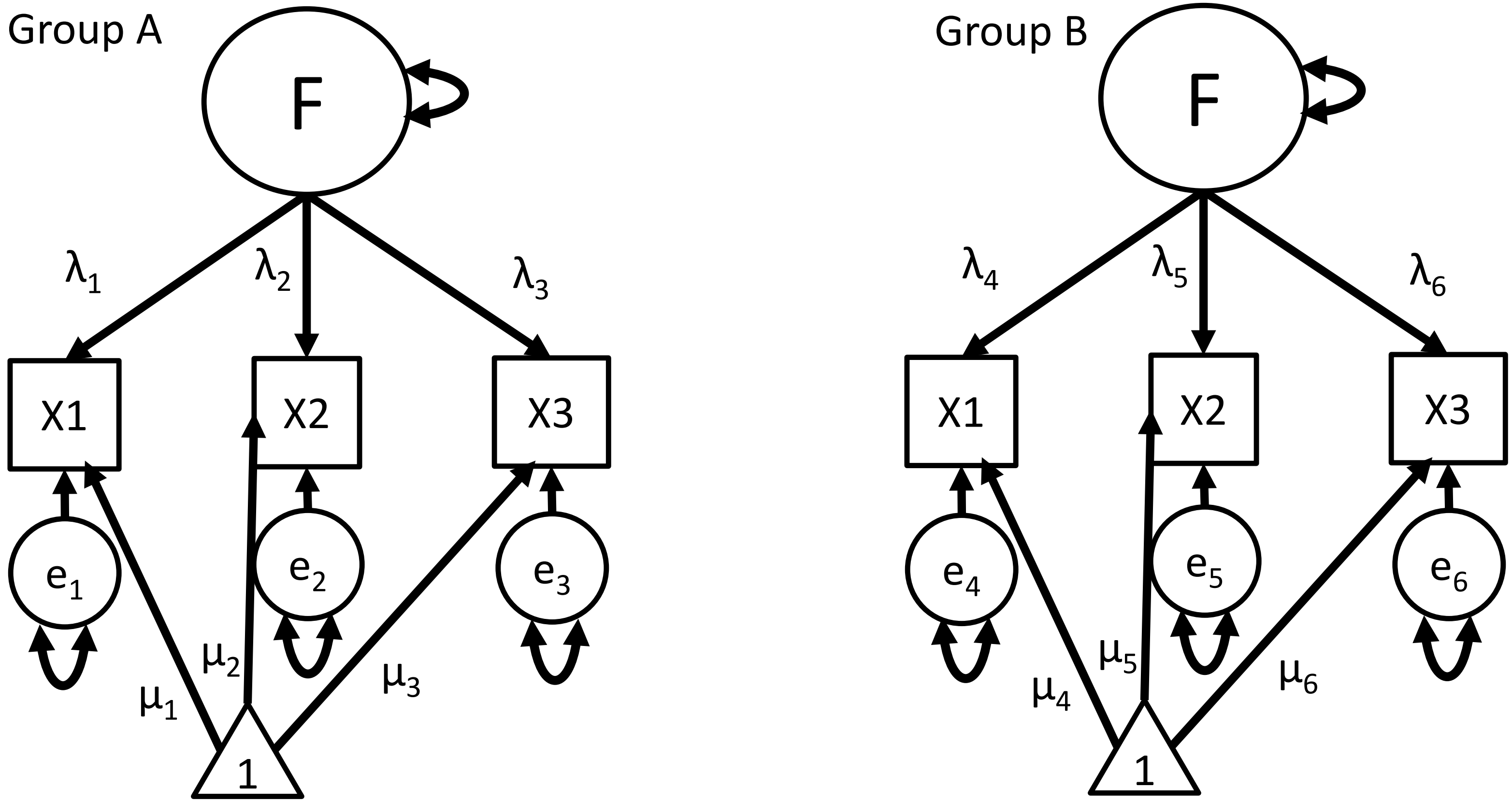
Levels of MI

1. Configural invariance: same factor loading pattern across groups.
2. Metric invariance: factor loadings equal across groups (aka weak invariance).
3. Scalar invariance: loadings & intercept equal across groups (aka strong invariance).
4. Strict invariance: residual variances equal across groups.



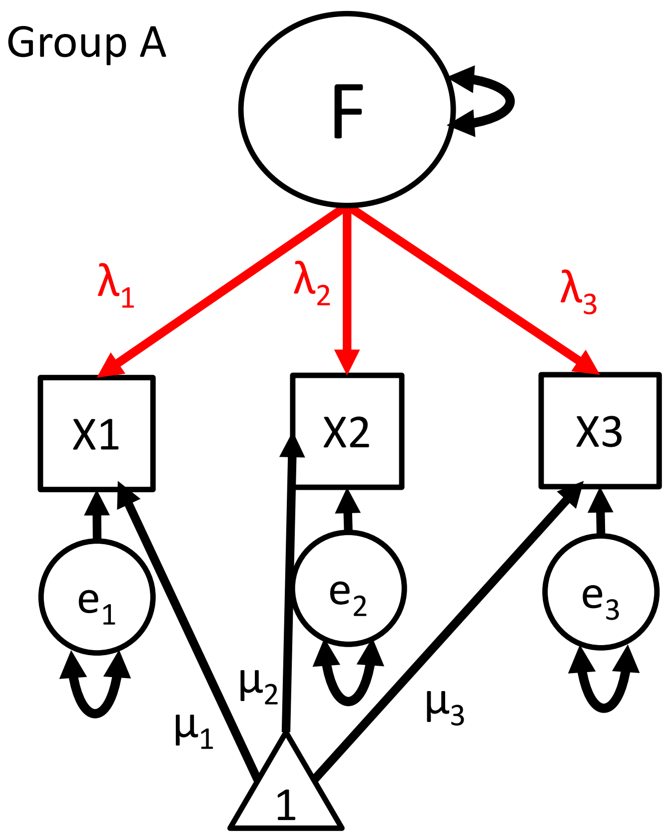
Configural invariance

Parameters are free to vary across groups



Metric invariance (weak)

Factor loadings are held equal across groups

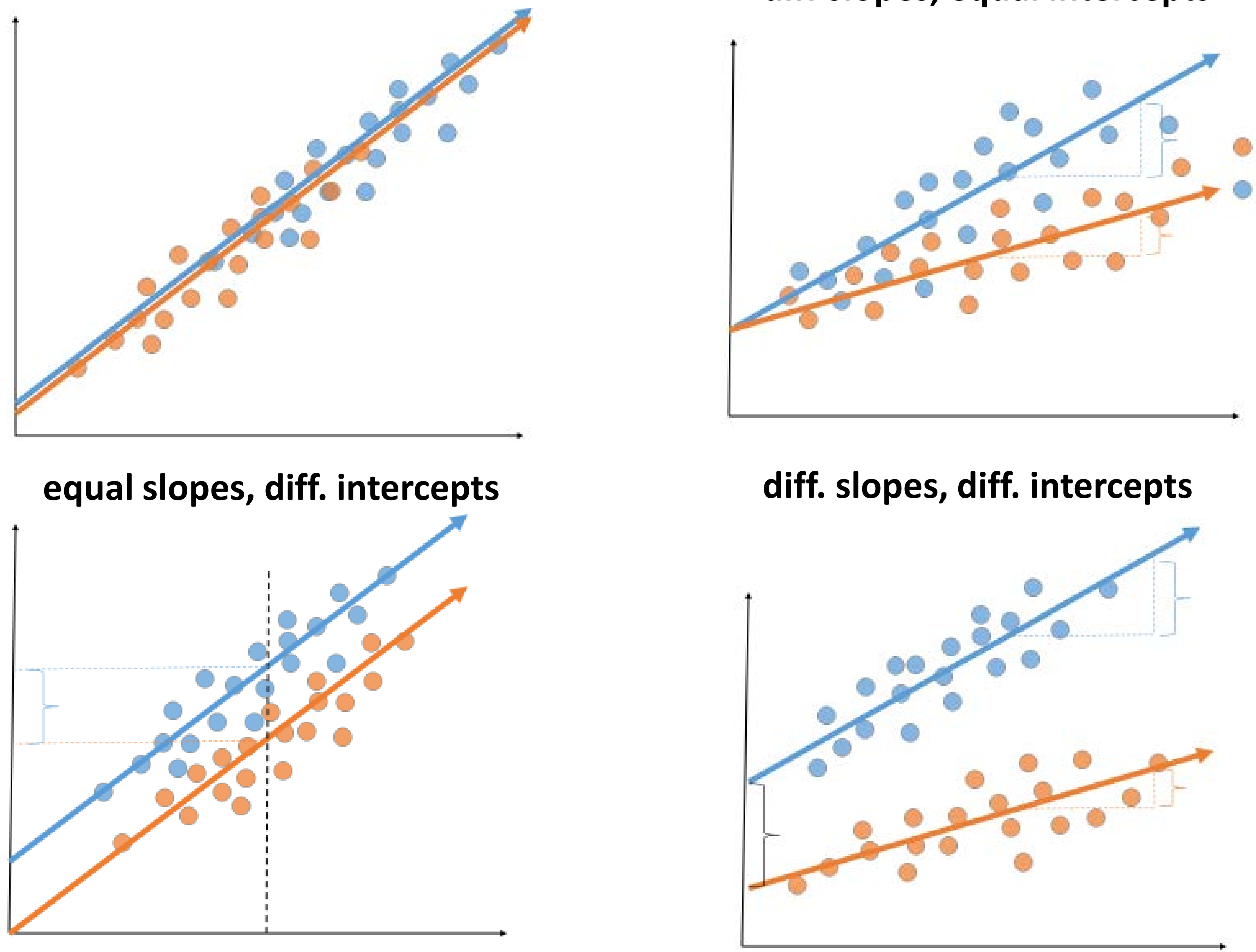


equal slopes and intercepts

diff slopes, equal intercepts

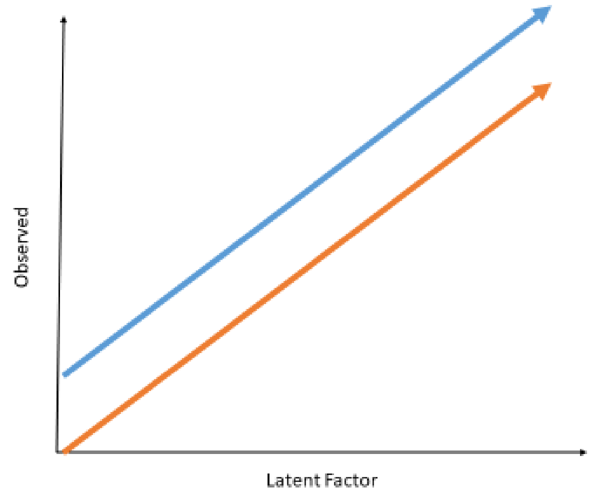
equal slopes, diff. intercepts

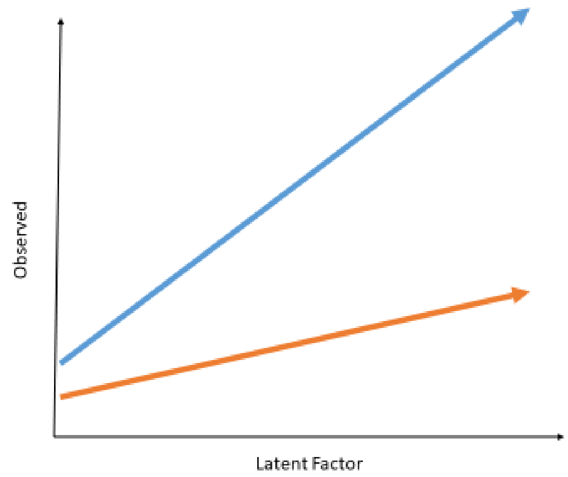
diff. slopes, diff. intercepts



Metric invariance (weak)

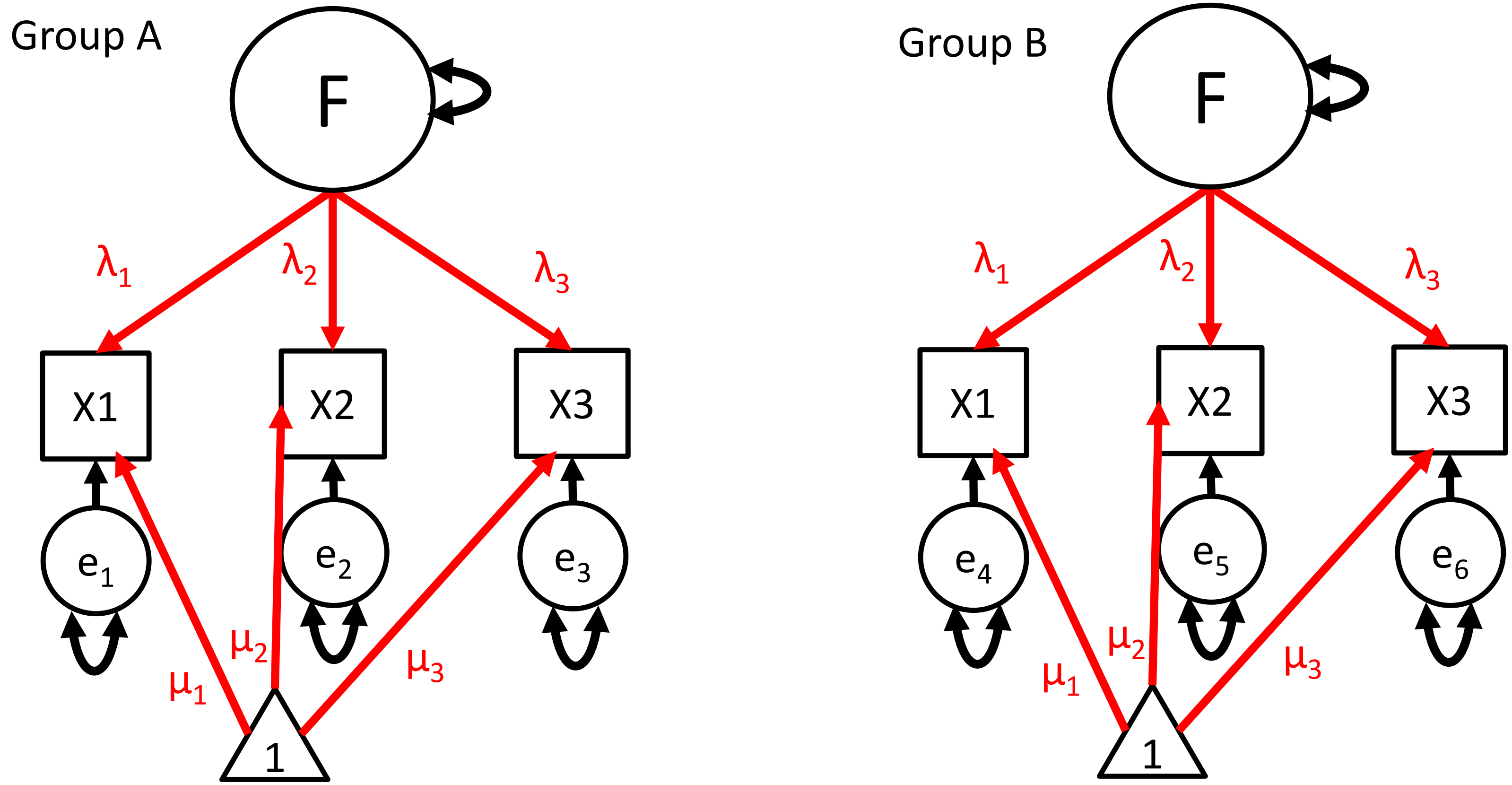
- **Are factor loadings equal?**
- Factor loadings, like regression weights, shows us the relationship between a latent factor and observed variables.
- Compare the fit of the metric invariance model with the fit of the configural model using a chi-square difference test.
- If not significantly different, the factor loadings are invariant.
- This suggests that the same construct is being measured.





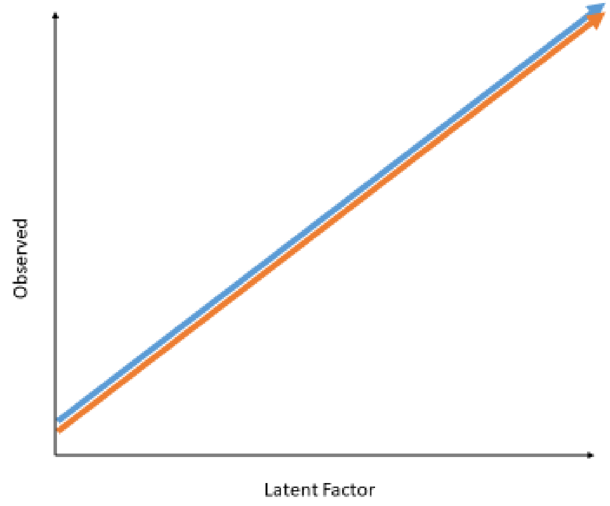
Scalar invariance (strong)

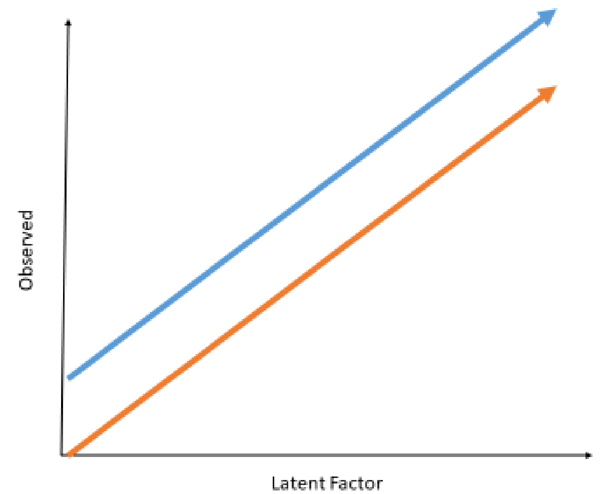
Factor loadings and intercepts are held equal across groups



Scalar invariance (strong)

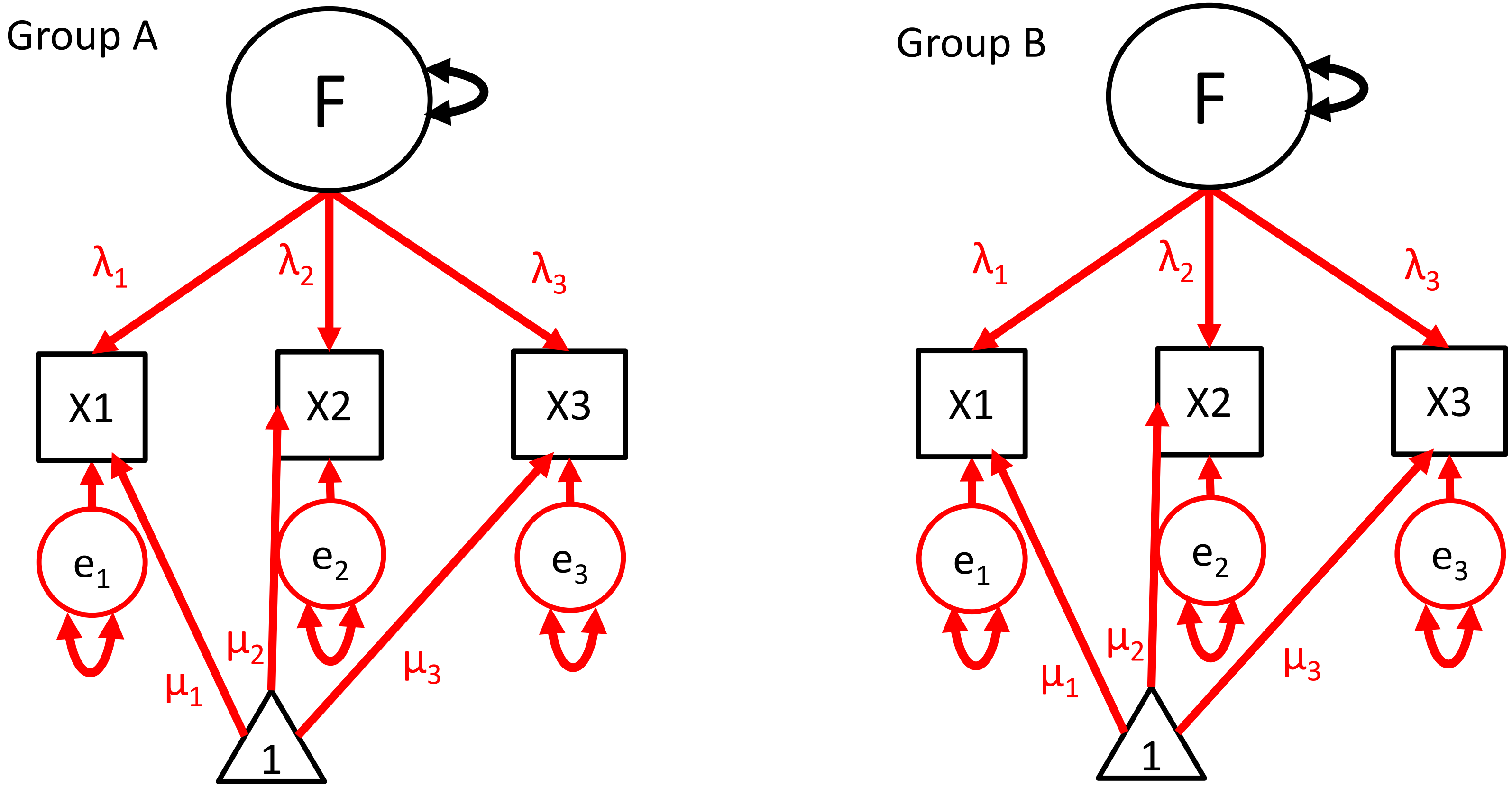
- **Are factor loadings AND intercepts equal?**
- Compare the fit of the scalar invariance model with the fit of the metric invariance model.
- If this model is significantly worse than the previous one, the intercepts are not equals, suggesting that one group tend to give higher or lower item response.





Strict invariance

Factor loadings, intercepts and residuals are held equal across groups



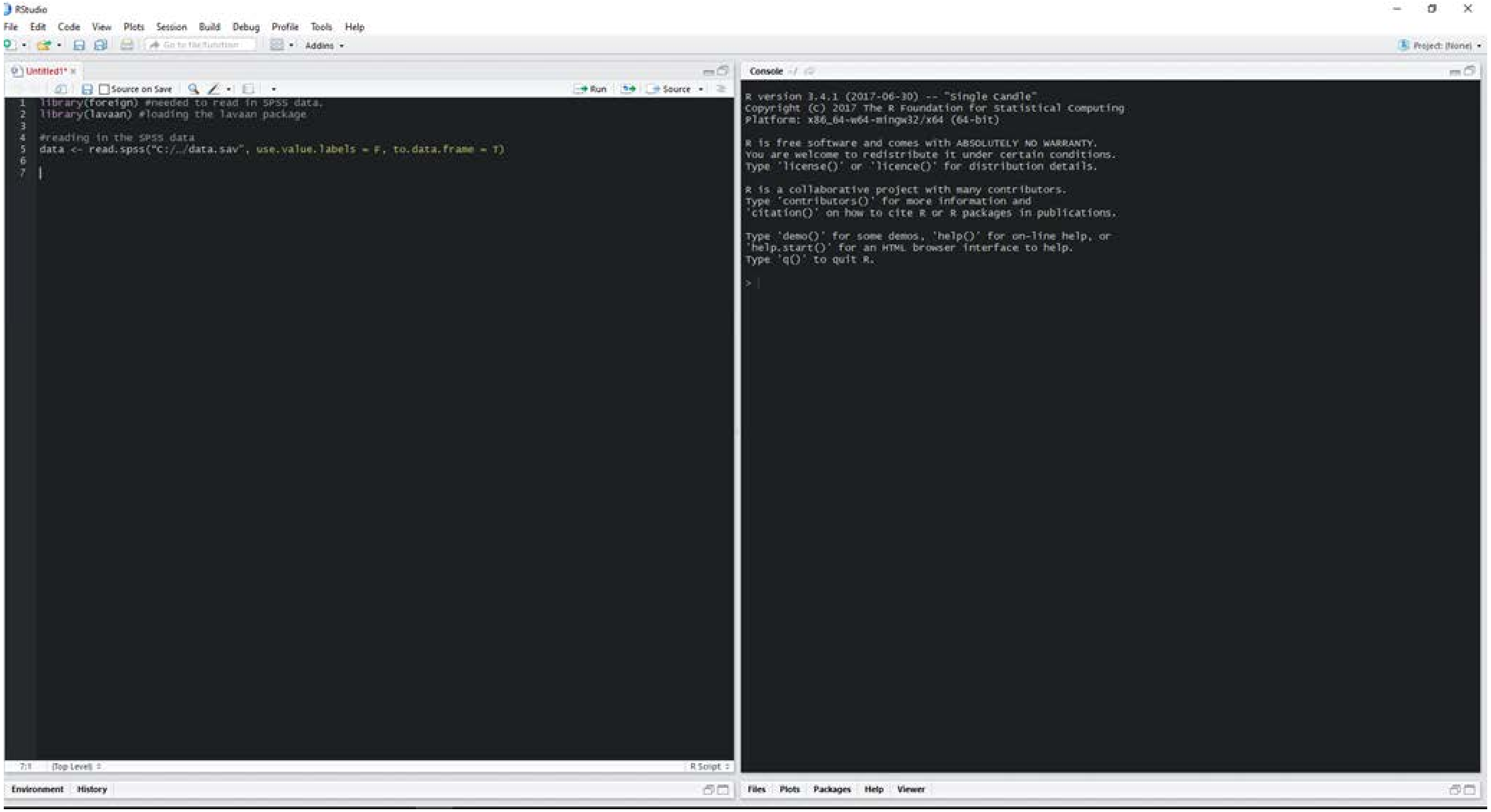
Strict invariance

- **Are factor loadings AND intercepts AND residual variance equal?**
- Compare the fit of the strict invariance model with the fit of the scalar invariance model.
- The strict invariance model is highly constrained model and often rejected in practice.

Software

- Mplus, SAS (proc calis), SPSS (AMOS), STATA (SEM builder), SmartPLS, LISREL, Onyx, EQS, etc.
- R
  - OpenMx
  - sem
  - lavaan
  - semTools

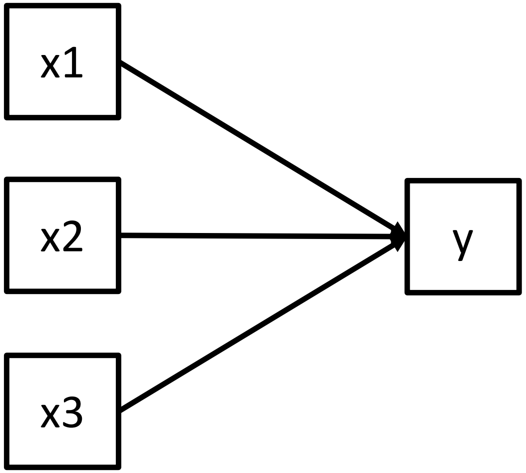
Software Options: R and RStudio



Reading in data

```
library(foreign) #needed to read in SPSS data.
library(lavaan) #loading the lavaan package

#reading in the SPSS data
data <- read.spss("C:/…/data.sav", use.value.labels = F, to.data.frame = T)

#inspect data
head(data)
str(data)
summary(data)
objects(data)
View(data)
```

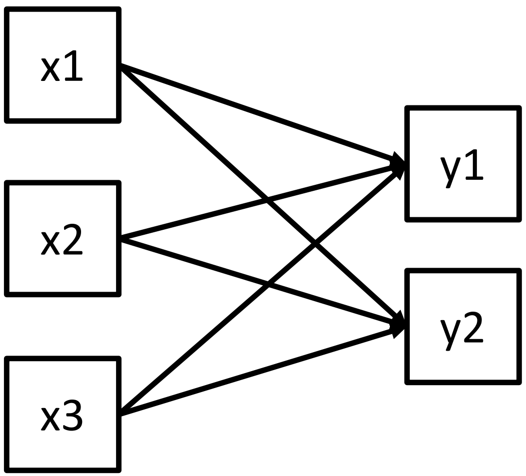
lavaan syntax



Type	Operator	definitions
Latent variable	=~	Is measured by
Regression	~	Is regressed on
(co)variance	~~	Is correlated with
Intercept	~1	intercept

```
# specify the model
model <- ' y  ~ x1 + x2 + x3 '

# fit the model
fit <- cfa(model, data)

# display summary output
summary(fit, fit.measures=TRUE)
```

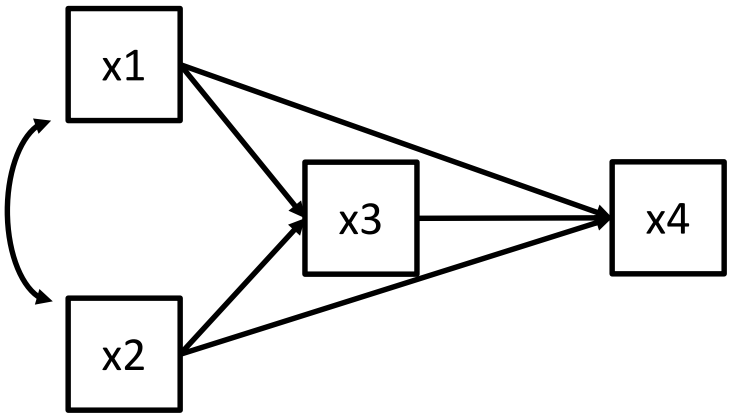
lavaan syntax



Type	Operator	definitions
Latent variable	=~	Is measured by
Regression	~	Is regressed on
(co)variance	~~	Is correlated with
Intercept	~1	intercept

```
# specify the model
model <- ' y1  ~ x1 + x2 + x3
           y2  ~ x1 + x2 + x3 '

# fit the model
fit <- cfa(model, data)

# display summary output
summary(fit, fit.measures=TRUE)
```

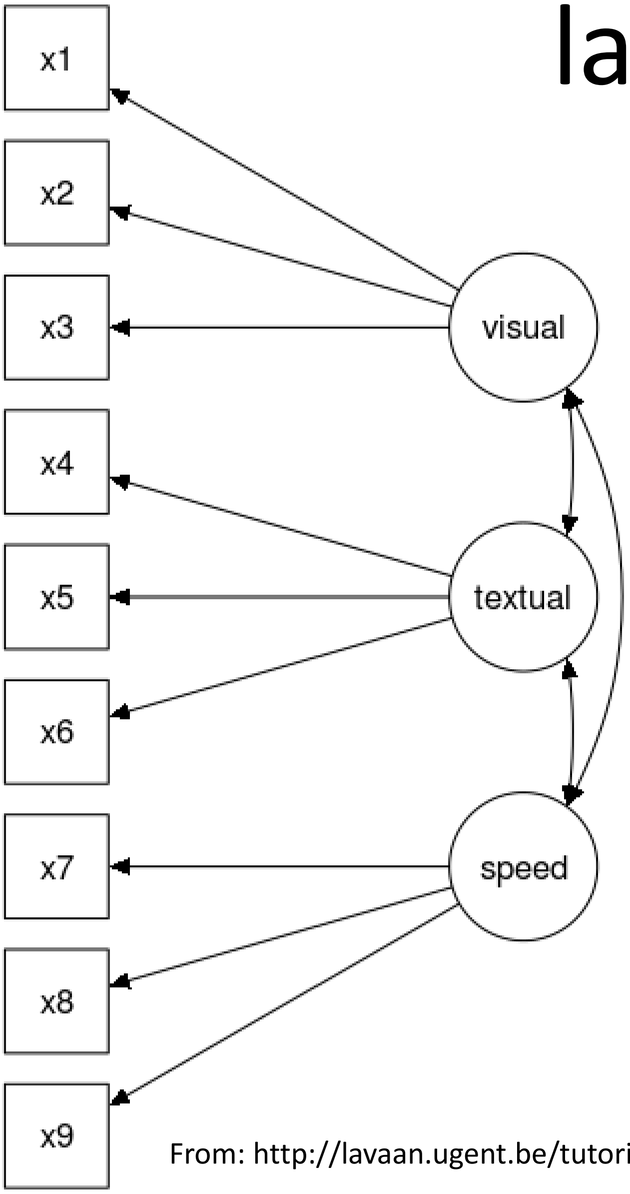
lavaan syntax



Type	Operator	definitions
Latent variable	=~	Is measured by
Regression	~	Is regressed on
(co)variance	~~	Is correlated with
Intercept	~1	intercept

```
# specify the model
model <- ' x3  ~ x1 + x2
           x4  ~ x1 + x2 + x3
          x1 ~~ x2 '

# fit the model
fit <- cfa(model, data)

# display summary output
summary(fit, fit.measures=TRUE)
```

lavaan syntax



Type	Operator	definitions
Latent variable	=~	Is measured by
Regression	~	Is regressed on
(co)variance	~~	Is correlated with
Intercept	~1	intercept

```
# specify the model
HS.model <- ' visual  =~ x1 + x2 + x3
              textual =~ x4 + x5 + x6
              speed   =~ x7 + x8 + x9 '

# fit the model
fit <- cfa(HS.model, data)

# display summary output
summary(fit, fit.measures=TRUE)
```

From: http://lavaan.ugent.be/tutorial/cfa.html

Illustration using NSSE

Sample: two cohorts of first-time freshmen who took NSSE, 2014 and 2016 at Cal State Fullerton.

Latent Variable: *Learning Strategies*

Three indicator items (1 = never, 4 = very often)

- *LSreading*: Identified key information from reading assignments
- *LSnotes*: Reviewed your notes after class
- *LSsummary*: Summarized what you learned in class or from course materials

Grouping Variable: URM (URM vs. non-URM)

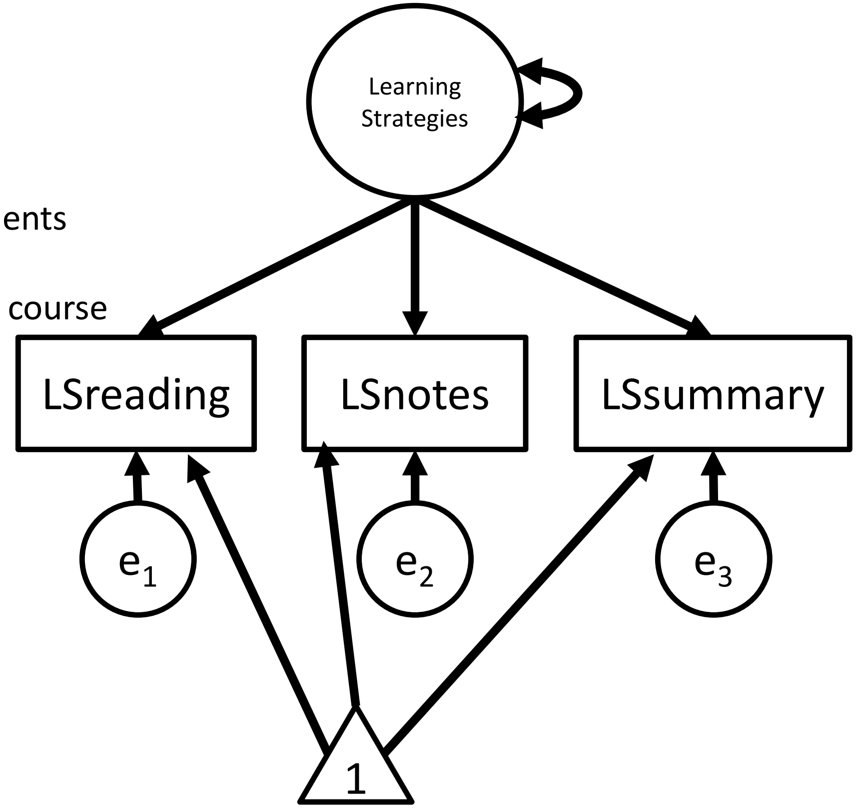


Illustration of MI

#specify your model

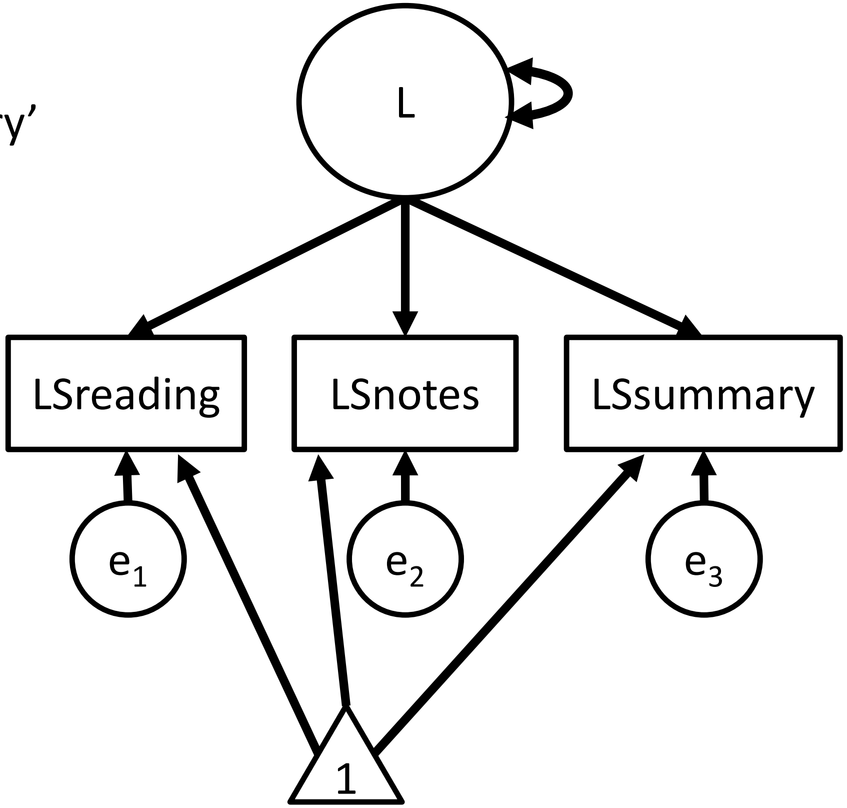
model <- 'L =~ LSreading + LSnotes + LSsummary'

#fit your model

modelresult <- cfa(model, data, missing = 'ML')

#display model

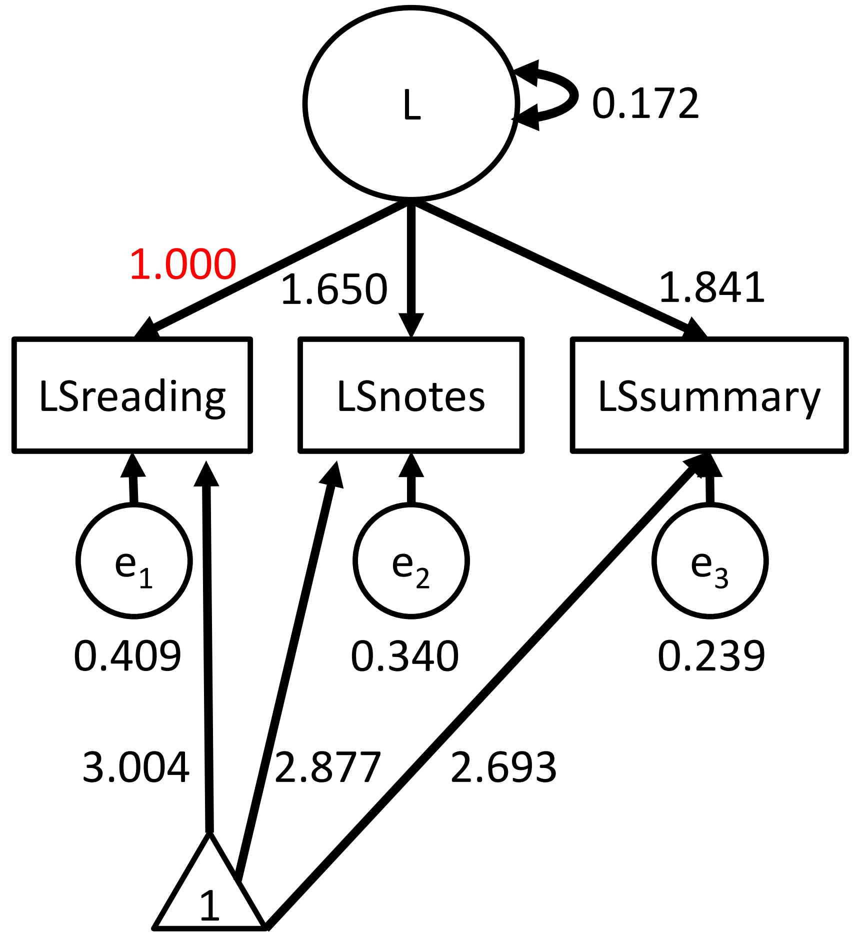
summary(modelresult ,fit.measures=TRUE)



lavaan

model <- 'LS =~ LSreading + LSnotes + LSsummary'

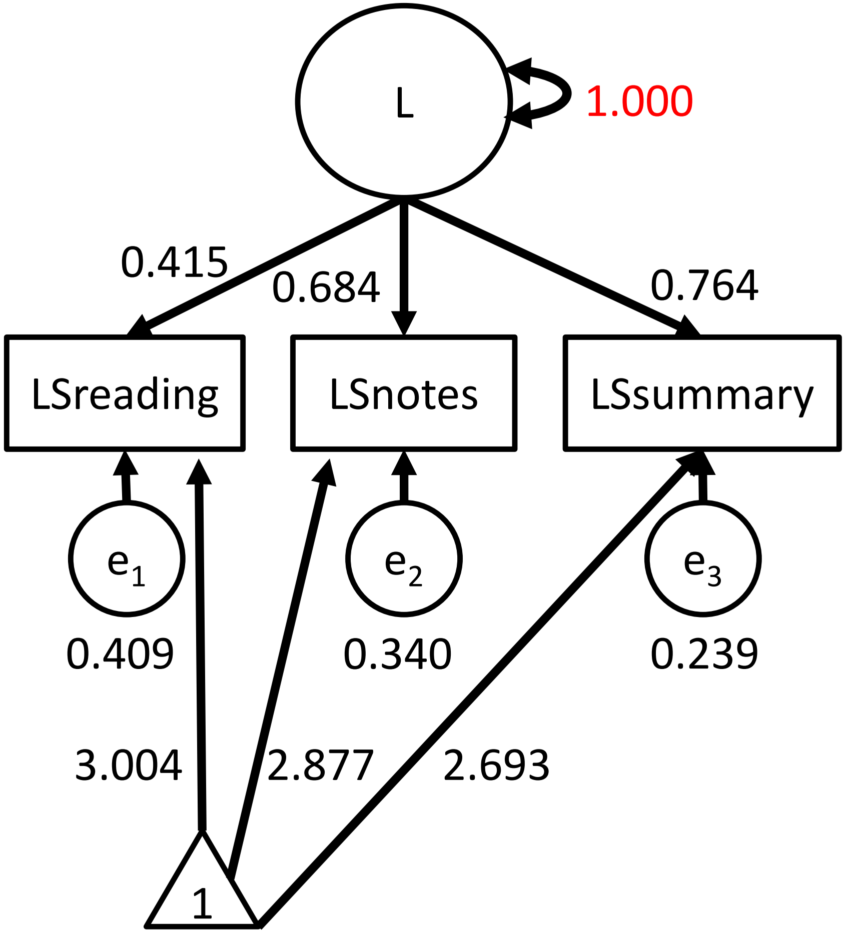
```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  L =~
    LSreading         1.000
    LSnotes           1.650    0.091   18.166    0.000
    LSsummary         1.841    0.106   17.326    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         3.004    0.020  150.237    0.000
   .LSnotes           2.877    0.024  122.025    0.000
   .LSsummary         2.693    0.024  113.057    0.000
    L                 0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         0.409    0.017   23.937    0.000
   .LSnotes           0.340    0.025   13.612    0.000
   .LSsummary         0.239    0.028    8.454    0.000
    L                 0.172    0.017   10.004    0.000
```



lavaan

model <- ‘LS =~ NA*LSreading + LSnotes + Lssummary
LS ~~ 1*LS’

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  L =~
    LSreading         0.415    0.021   20.008    0.000
    LSnotes           0.684    0.025   26.981    0.000
    LSsummary         0.764    0.026   29.362    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         3.004    0.020  150.237    0.000
   .LSnotes           2.877    0.024  122.025    0.000
   .LSsummary         2.693    0.024  113.057    0.000
    L                 0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
    L                 1.000
   .LSreading         0.409    0.017   23.937    0.000
   .LSnotes           0.340    0.025   13.612    0.000
   .LSsummary         0.239    0.028    8.454    0.000
```



Configural invariance

cfa(model, data, group = 'URM', missing = 'ML')

Non-URM (n=761)

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  L =~
    LSreading         1.000
    LSnotes           1.580    0.120   13.165    0.000
    LSsummary         1.808    0.146   12.428    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         2.977    0.028  107.683    0.000
   .LSnotes           2.866    0.032   88.560    0.000
   .LSsummary         2.678    0.033   81.815    0.000
    L                 0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         0.404    0.024   17.086    0.000
   .LSnotes           0.353    0.034   10.447    0.000
   .LSsummary         0.233    0.039    5.927    0.000
    L                 0.176    0.024    7.288    0.000
```

URM (n=697)

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  L =~
    LSreading         1.000
    LSnotes           1.730    0.139   12.489    0.000
    LSsummary         1.878    0.156   12.055    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         3.033    0.029  104.911    0.000
   .LSnotes           2.890    0.034   83.978    0.000
   .LSsummary         2.709    0.035   78.056    0.000
    L                 0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         0.414    0.025   16.746    0.000
   .LSnotes           0.324    0.037    8.752    0.000
   .LSsummary         0.246    0.041    6.048    0.000
    L                 0.167    0.024    6.848    0.000
```

Metric invariance

cfa(model, data, group = 'URM', missing = 'ML', group.equal = c('loadings'))

Non-URM (n=761)

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  L =~
    LSredng           1.000
    LSnotes (.p2.)    1.652    0.091   18.147    0.000
    LSsmmry (.p3.)    1.842    0.106   17.324    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         2.977    0.027  108.359    0.000
   .LSnotes           2.866    0.033   88.125    0.000
   .LSsummary         2.678    0.033   81.882    0.000
    L                 0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         0.406    0.023   17.508    0.000
   .LSnotes           0.345    0.030   11.365    0.000
   .LSsummary         0.241    0.033    7.287    0.000
    L                 0.167    0.018    9.087    0.000
```

URM (n=697)

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  L =~
    LSredng           1.000
    LSnotes (.p2.)    1.652    0.091   18.147    0.000
    LSsmmry (.p3.)    1.842    0.106   17.324    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         3.033    0.029  104.200    0.000
   .LSnotes           2.890    0.034   84.404    0.000
   .LSsummary         2.709    0.035   77.992    0.000
    L                 0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         0.413    0.024   16.860    0.000
   .LSnotes           0.334    0.031   10.650    0.000
   .LSsummary         0.238    0.034    7.005    0.000
    L                 0.177    0.020    9.013    0.000
```

Compare configural vs metric

config_out <- cfa(model, data, group = 'URM', missing = 'ML')

metric_out <- cfa(model, data, group = 'URM', missing = 'ML', group.equal = c('loadings'))

lavTestLRT(config_out, metric_out)

```
Chi Square Difference Test

           Df    AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
config_out  0 9843.7 9938.8 0.0000
metric_out  2 9840.4 9925.0 0.7136    0.71364       2     0.6999
```

Not sig.

Scalar invariance

cfa(model, data, group = 'URM', missing = 'ML', group.equal = c('loadings', 'intercepts'))

Non-URM (n=761)

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  L =~
    LSredng           1.000
    LSnotes (.p2.)    1.650    0.091   18.164    0.000
    LSsmmry (.p3.)    1.839    0.106   17.345    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSredng (.p8.)    2.994    0.023  130.269    0.000
   .LSnotes (.p9.)    2.862    0.030   94.968    0.000
   .LSsmmry (.10.)    2.675    0.032   84.486    0.000
    L                 0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         0.406    0.023   17.496    0.000
   .LSnotes           0.345    0.030   11.378    0.000
   .LSsummary         0.241    0.033    7.305    0.000
    L                 0.168    0.018    9.094    0.000
```

URM (n=697)

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  L =~
    LSredng           1.000
    LSnotes (.p2.)    1.650    0.091   18.164    0.000
    LSsmmry (.p3.)    1.839    0.106   17.345    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSredng (.p8.)    2.994    0.023  130.269    0.000
   .LSnotes (.p9.)    2.862    0.030   94.968    0.000
   .LSsmmry (.10.)    2.675    0.032   84.486    0.000
    L                 0.020    0.024    0.819    0.413

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         0.413    0.025   16.849    0.000
   .LSnotes           0.334    0.031   10.664    0.000
   .LSsummary         0.238    0.034    7.021    0.000
    L                 0.177    0.020    9.020    0.000
```

Compare metric vs scalar

scalar_out <- cfa(model, data, group = 'URM', missing = 'ML', group.equal = c('loadings', 'intercepts'))

lavTestLRT(metric_out, scalar_out)

```
Chi Square Difference Test

           Df    AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
metric_out  2 9840.4 9925.0 0.7136
scalar_out  4 9837.7 9911.7 2.0114     1.2978       2     0.5226
```

Not sig.

Strict invariance

cfa(model, data, group = 'URM', missing = 'ML', group.equal = c('loadings', 'intercepts', 'residuals'))

Non-URM (n=761)

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  L =~
    LSredng           1.000
    LSnotes (.p2.)    1.650    0.091   18.169    0.000
    LSsmmry (.p3.)    1.840    0.106   17.344    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSredng (.p8.)    2.994    0.023  130.237    0.000
   .LSnotes (.p9.)    2.862    0.030   95.043    0.000
   .LSsmmry (.10.)    2.675    0.032   84.478    0.000
    L                 0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSredng (.p4.)    0.409    0.017   23.936    0.000
   .LSnotes (.p5.)    0.340    0.025   13.626    0.000
   .LSsmmry (.p6.)    0.240    0.028    8.490    0.000
    L                 0.168    0.018    9.136    0.000
```

URM (n=697)

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  L =~
    LSredng           1.000
    LSnotes (.p2.)    1.650    0.091   18.169    0.000
    LSsmmry (.p3.)    1.840    0.106   17.344    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSredng (.p8.)    2.994    0.023  130.237    0.000
   .LSnotes (.p9.)    2.862    0.030   95.043    0.000
   .LSsmmry (.10.)    2.675    0.032   84.478    0.000
    L                 0.020    0.024    0.819    0.413

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSredng (.p4.)    0.409    0.017   23.936    0.000
   .LSnotes (.p5.)    0.340    0.025   13.626    0.000
   .LSsmmry (.p6.)    0.240    0.028    8.490    0.000
    L                 0.177    0.020    9.051    0.000
```

Compare scalar vs strict

```
strict_out <- cfa(model, data, group = 'URM', missing = 'ML', group.equal = c('loadings', 'intercepts', 'residuals'))

lavTestLRT(scalar_out, strict_out)
```

```
Chi Square Difference Test

           Df    AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
scalar_out  4 9837.7 9911.7 2.0114
strict_out  7 9831.9 9890.0 2.1773    0.16585       3     0.9829
```

Not sig.

Alternative: semTools

library(semTools)

model <- 'L =~ LSreading + LSnotes + LSsummary'

measurementInvariance(model, data, strict= TRUE, group = "URM", missing = 'ML')

```
Measurement invariance models:

Model 1 : fit.configural
Model 2 : fit.loadings
Model 3 : fit.intercepts
Model 4 : fit.residuals
Model 5 : fit.means

Chi Square Difference Test

               Df    AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
fit.configural  0 9843.7 9938.8 0.0000
fit.loadings    2 9840.4 9925.0 0.7136    0.71364       2     0.6999
fit.intercepts  4 9837.7 9911.7 2.0114    1.29780       2     0.5226
fit.residuals   7 9831.9 9890.0 2.1773    0.16585       3     0.9829
fit.means       8 9830.6 9883.4 2.8493    0.67200       1     0.4124
```

Issues

- What if the measurement is non-invariant? (Sass, 2011).
  - Use only invariant items.
  - Allows parameters of non-invariant items to vary across groups (partial measurement invariance model)
  - Use all the items if the extent of noninvariance is small.
  - avoid using the scale
- Are survey items considered continuous or ordinal?
  - lavaan can model ordinal data.

Issues

```
scalar_out <- cfa(model, data = data,
                  group.equal = c("loadings", "intercepts"),
                  group.partial = c('L =~ LSnotes', 'LSnotes ~ 1'),
                  group = "URM", missing = 'ML')
```

```
Group 1 [0]:

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  L =~
    LSredng           1.000
    LSnotes           1.594    0.112   14.216    0.000
    LSsmmry (.p3.)    1.839    0.106   17.329    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSredng (.p8.)    2.993    0.023  127.948    0.000
   .LSnotes           2.866    0.032   88.561    0.000
   .LSsmmry (.10.)    2.673    0.032   82.378    0.000
    L                 0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         0.406    0.023   17.431    0.000
   .LSnotes           0.355    0.033   10.745    0.000
   .LSsummary         0.229    0.037    6.200    0.000
    L                 0.172    0.020    8.779    0.000
```

```
Group 2 [1]:

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  L =~
    LSredng           1.000
    LSnotes           1.710    0.123   13.894    0.000
    LSsmmry (.p3.)    1.839    0.106   17.329    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSredng (.p8.)    2.993    0.023  127.948    0.000
   .LSnotes           2.851    0.040   70.580    0.000
   .LSsmmry (.10.)    2.673    0.032   82.378    0.000
    L                 0.023    0.025    0.888    0.375

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .LSreading         0.413    0.024   16.902    0.000
   .LSnotes           0.321    0.036    8.960    0.000
   .LSsummary         0.251    0.038    6.668    0.000
    L                 0.172    0.020    8.469    0.000
```

Issues

library(semTools)

model <- 'L =~ LSreading + LSnotes + LSsummary‘

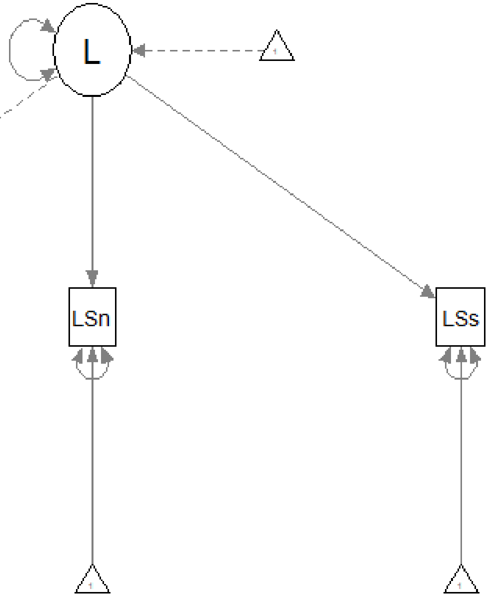
measurementInvarianceCat(model, data = data, strict = T, group = "URM",
ordered = c("LSreading","LSnotes","LSsummary"))

```
Measurement invariance models:

Model 1 : fit.configural
Model 2 : fit.loadings
Model 3 : fit.thresholds
Model 4 : fit.residuals
Model 5 : fit.means

Scaled Chi Square Difference Test (method = "satorra.bentler.2001")

               Df AIC BIC  Chisq Chisq diff Df diff Pr(>Chisq)
fit.configural  0         0.0000
fit.loadings    2         2.3680     4.1477       2     0.1257
fit.thresholds  7         5.3724     6.5088       5     0.2598
fit.residuals  10         5.5467     0.3762       3     0.9451
fit.means      11         7.8282     0.5332       1     0.4653
```

Optional: Plotting the model

library(semPlot)

semPaths(config_out)
semPaths(config_out, “est”) # if you want add estimates to the figure.



Summary

- Measurement invariance is **required** for accurate assessment and evaluation.
- Multiple Group CFA is the most widely used tool for testing measurement invariance.
- Testing for measurement invariance in R is relatively simple.
  - A lot of examples online.

End

Question?