

# Lab 8 - Multiple Indicator, Multiple Causes - MIMIC Models

Adam Garber

Factor Analysis ED 216B - Instructor: Karen Nylund-Gibson

March 30, 2020

## Contents

<b>1</b>	<b>Lab 8 outline</b>	<b>2</b>
1.1	Getting started - following the routine: . . . . .	2
1.2	R-Project instructions: . . . . .	2
1.3	loading (and installing when needed) packages: . . . . .	2
<b>2</b>	<b>~~~~~ Lab 8 - Begin ~~~~~</b>	<b>3</b>
2.1	read in data . . . . .	3
2.2	take a look at the EFA data (same indicators used for lab 4) . . . . .	3
2.3	alternative way to make summary tables using package {gtsummary} . . . . .	3
2.4	prepare dataframe for analysis (select & reorder columns) . . . . .	3
<b>3</b>	<b>Estimate the Unconditional Confirmatory Factor Analysis (CFA) model</b>	<b>4</b>
3.1	Lab exercise: How many parameters are there in this model? . . . . .	4
3.2	Make a simple CFA path diagram using package {Diagrammer} . . . . .	4
3.3	Lab exercise: Count model parameters from the path diagram . . . . .	6
<b>4</b>	<b>MIMIC model 1 - single bivariate covariate</b>	<b>6</b>
<b>5</b>	<b>MIMIC model 2 - probe for covariate -&gt; indicator DIFF</b>	<b>8</b>
<b>6</b>	<b>MIMIC model 3 - specify covariate -&gt; indicator DIFF</b>	<b>10</b>
<b>7</b>	<b>MIMIC model 4 - two covariates &amp; an interaction term</b>	<b>11</b>
7.1	create a path diagram of MIMIC model 4 . . . . .	13
<b>8</b>	<b>MIMIC model 5 - three continuous covariates</b>	<b>13</b>
8.1	create a path diagram of MIMIC model 5 . . . . .	14
8.2	practice some formatting with semPlot::semPaths() . . . . .	14
8.3	read all models and create table . . . . .	14

<b>9 End of Lab 8</b>	<b>15</b>
9.1 References . . . . .	15

---

## 1 Lab 8 outline

- a. Prepare, wrangle, and explore data
  - b. Run an unconditional CFA baseline model
  - c. Specify a MIMIC model with a single binary covariate
  - d. Specify a MIMIC model and probe for DIF
  - e. Specify a MIMIC model with a DIF parameter
  - f. Specify a MIMIC model with two binary covariates & an interaction
  - g. Specify a MIMIC model with three continuous covariates
  - h. Experiment with path diagram notation & formatting
  - i. **We will keep close track of parameters and their status throughout lab**
- 

### 1.1 Getting started - following the routine:

- a. Create an R-Project
- b. Load & install packages (we will test a NEW method today)

### 1.2 R-Project instructions:

- a. click “NEW PROJECT” (upper right corner of window)
- b. choose option “NEW DIRECTORY”
- c. choose location of project (**too many nested folders = bad for ‘MplusObject’ function**)

Within R-studio under the files pane (bottom right):

- a. click “New Folder” and name folder “data”
  - b. click “New Folder” and name folder “mimic\_mplus”
  - c. click “New Folder” and name folder “figures”
- 

### 1.3 loading (and installing when needed) packages:

**We are testing an alternative method for this procedure today (simply run the code below)**

```
if (!require(pacman)) { install.packages("pacman"); library(pacman) }  
  
p_load(knitr, tidyverse, here, semPlot, DiagrammerR, MplusAutomation,  
       rhdf5, texreg, stargazer, gtsummary, gt, kableExtra)
```

---

DATA SOURCE: This lab exercise utilizes the NCES public-use dataset: Education Longitudinal Study of 2002 (Lauff & Ingels, 2014) [See website: nces.ed.gov](http://nces.ed.gov)

## 2 ~~~~~ Lab 8 - Begin ~~~~~

### 2.1 read in data

```
lab_data <- read_csv(here("data", "els_sub5_data.csv"))
```

### 2.2 take a look at the EFA data (same indicators used for lab 4)

```
stargazer(as.data.frame(lab_data), type="text", digits=1)
```

### 2.3 alternative way to make summary tables using package {gtsummary}

```
table_data <- lab_data %>%
  dplyr::select(byincome, mth_test, rd_test, freelnch, bystlang)

table2 <- tbl_summary(table_data,
  by = bystlang,      # split table by group "bystlang" ()
  missing = "no"     # don't list missing data separately
) %>%
  add_n() %>%        # add column with total number of non-missing observations
  add_p() %>%        # test if there's difference between groups
  bold_labels()

table2
```

Characteristic	N	0, N = 117 <sup>1</sup>	1, N = 632 <sup>1</sup>	p-value <sup>2</sup>
byincome	749	8.00 (6.00, 10.00)	10.00 (8.00, 11.00)	<0.001
mth_test	749	48 (41, 55)	52 (45, 58)	<0.001
rd_test	749	46 (40, 52)	51 (44, 58)	<0.001
freelnch	685			<0.001
1		23 (23%)	222 (38%)	
2		9 (8.9%)	67 (11%)	
3		14 (14%)	82 (14%)	
4		9 (8.9%)	72 (12%)	
5		15 (15%)	89 (15%)	
6		18 (18%)	30 (5.1%)	
7		13 (13%)	22 (3.8%)	

<sup>1</sup>Statistics presented: median (IQR); n (%)

<sup>2</sup>Statistical tests performed: Wilcoxon rank-sum test; chi-square test of independence

### 2.4 prepare dataframe for analysis (select & reorder columns)

```
mimic_data <- lab_data %>%
  select(bystlang, freelnch, byincome,          # covariates
```

```

    stolen, t_hurt, p_fight, hit, damaged, bullied, # factor 1 (indicators)
    safe, disrupt, gangs, rac_fight,             # factor 2 (indicators)
    late, skipped, mth_read, mth_test, rd_test) %>%
mutate(
  freelnch = case_when( # Grade 10, percent free lunch - transform to binary
    freelnch < 5 ~ 0,   # < 50%
    freelnch >= 5 ~ 1)) # > 50%

```

---

### 3 Estimate the Unconditional Confirmatory Factor Analysis (CFA) model

#### 3.1 Lab exercise: How many parameters are there in this model?

(no cheating - i.e., jumping ahead)

Number of parameters for the Unconditional CFA model:

- ?? item loadings
- ?? intercepts
- ?? residual variances
- ?? factor variances
- ?? factor co-variance

---

#### 3.2 Make a simple CFA path diagram using package {Diagrammer}

```

# starting simple...

grViz(" digraph CFA_basic {

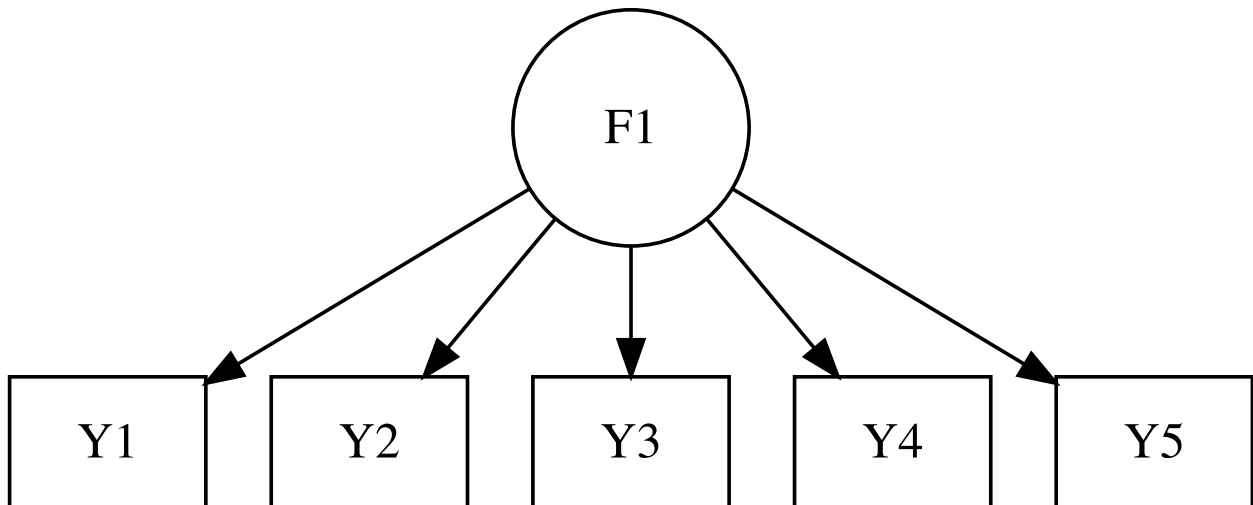
  node [shape=box]
  Y1; Y2; Y3; Y4; Y5;

  node [shape=circle, width = 0.9]
  F1;

  edge []
  F1->{Y1 Y2 Y3 Y4 Y5}

}")

```



```

cfa_m0 <- mplusObject(
  TITLE = "CFA model0 - LAB 8 mimic models",
  VARIABLE =
    "usevar = stolen-rac_fight;",

  ANALYSIS =
    "estimator = mlr;",

  MODEL =
    "FACTOR_1 by stolen t_hurt p_fight hit damaged bullied;

    FACTOR_2 BY safe disrupt gangs rac_fight;" ,

  PLOT = "type = plot3;",
  OUTPUT = "sampstat standardized residual modindices (3.84);",

  usevariables = colnames(mimic_data),
  rdata = mimic_data)

```

```

cfa_m0_fit <- mplusModeler(cfa_m0,
  dataout=here("mimic_mplus", "lab8_mimic_data.dat"),
  modelout=here("mimic_mplus", "lab8_cfa_model0.inp"),
  check=TRUE, run = TRUE, hashfilename = FALSE)

```

```

# Read in the model to R within the "mimic_mplus" folder
mimic_output1 <- readModels(here("Lab8_FA", "mimic_mplus", "lab8_cfa_model0.out"))

```

```

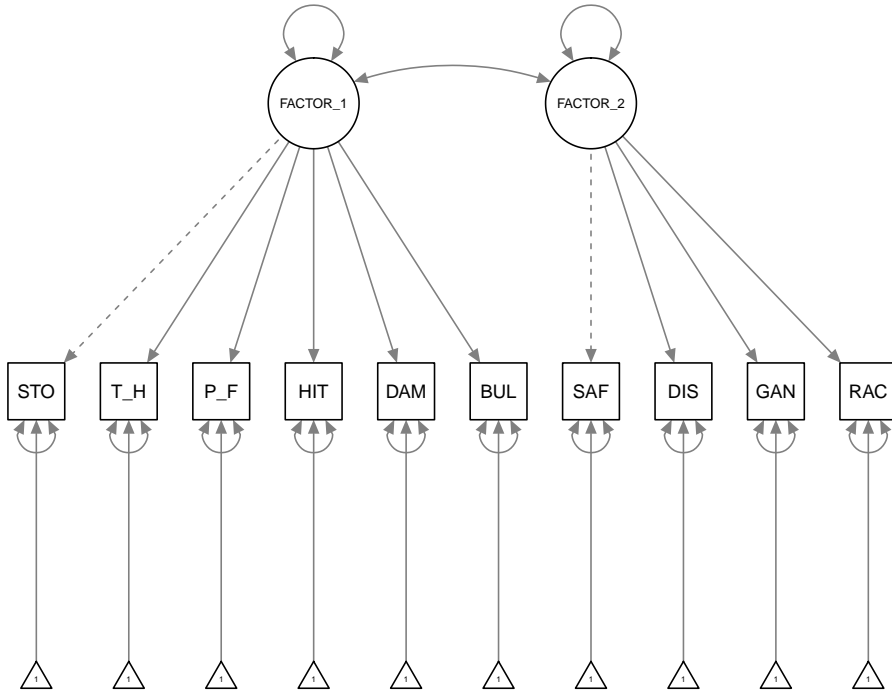
## Reading model: /Users/agarber/github/project-site/Lab8_FA/mimic_mplus/lab8_cfa_model0.out

```

```

# Plot model:
semPaths(mimic_output1,
  # intercepts=FALSE,
  # fixedStyle = c(1)
)

```



*# \*\* comment out the arguments "intercepts" & "fixedStyle" to make all parameters explicit*

---

### 3.3 Lab exercise: Count model parameters from the path diagram

(i.e., count number of arrows)

## 4 MIMIC model 1 - single bivariate covariate

Number of parameters for the MIMIC model 1 = 33

- 8 item loadings (10 items - 2 fixed loadings)
- 10 intercepts
- 10 residual variances
- 2 factor variances
- 1 factor co-variance
- 1 covariate mean
- 1 covariate variance

---

```
grViz(" digraph mimic_path_diagram {
graph [overlap = true, fontsize = 10, # this is the 'graph' statement
fontname = Times,
label=
```

```

'Figure 1: MIMIC model with single covariate.

node [shape = box]                # this is the 'node' statement

A; B; C; D; E;

node [shape = box,
      label = 'Covariate']

X;

node [shape = circle, fixedsize = true,
      width = 0.9, label = 'Factor 1']

F;

edge [color = black]              # this is the 'edge' statement

F->{A B C D E}
X->F

}"))

```

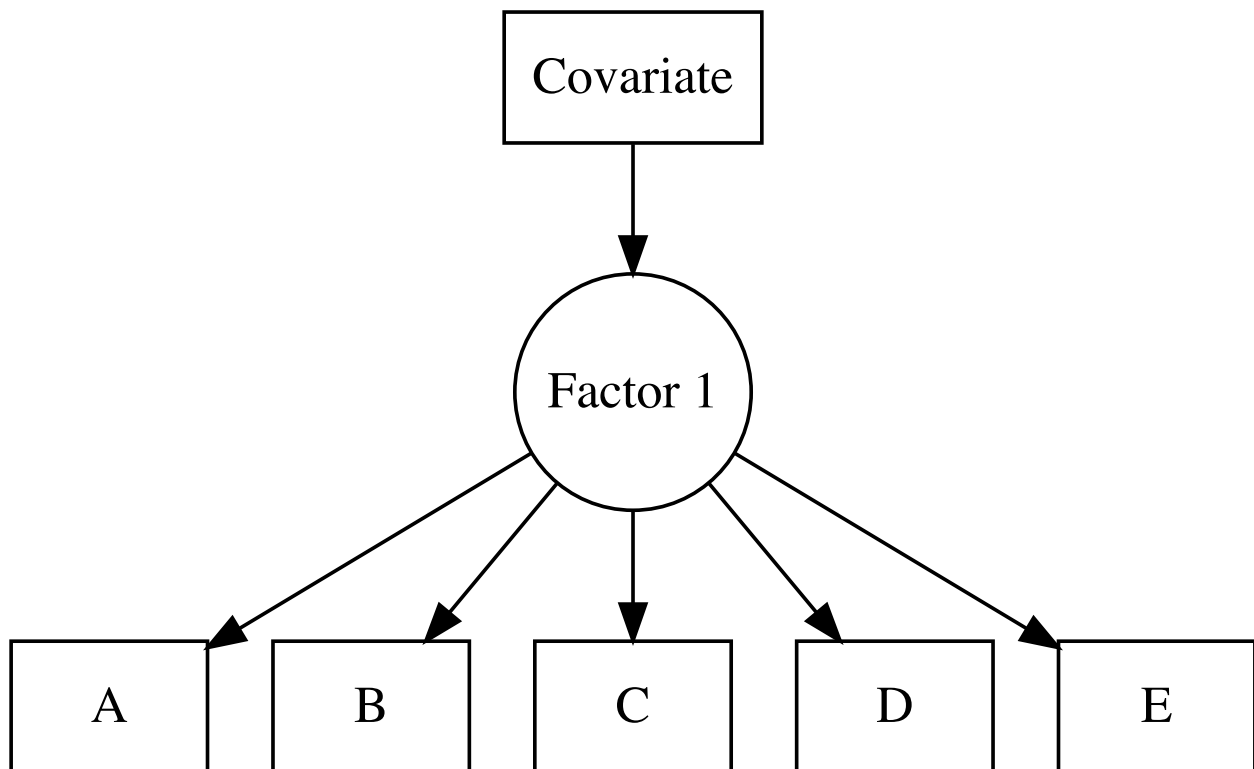


Figure 1: MIMIC model with single covariate.

```

mimic_m1 <- mplusObject(
  TITLE = "MIMIC model1 - LAB 8",
  VARIABLE =
    "usevar = freelnch stolen-rac_fght;",

  ANALYSIS =
    "estimator = mlr;",

  MODEL =
    "FACTOR_1 by stolen t_hurt p_fight hit damaged bullied;

    FACTOR_2 by safe disrupt gangs rac_fght;

    FACTOR_1 on freelnch;

    FACTOR_2 on freelnch;" ,

  PLOT = "type = plot3;",
  OUTPUT = "sampstat standardized residual modindices (3.84);",

  usevariables = colnames(mimic_data),
  rdata = mimic_data)

mimic_m1_fit <- mplusModeler(mimic_m1,
                             dataout=here("mimic_mplus", "lab8_mimic_data.dat"),
                             modelout=here("mimic_mplus", "lab8_mimic_model1.inp"),
                             check=TRUE, run = TRUE, hashfilename = FALSE)

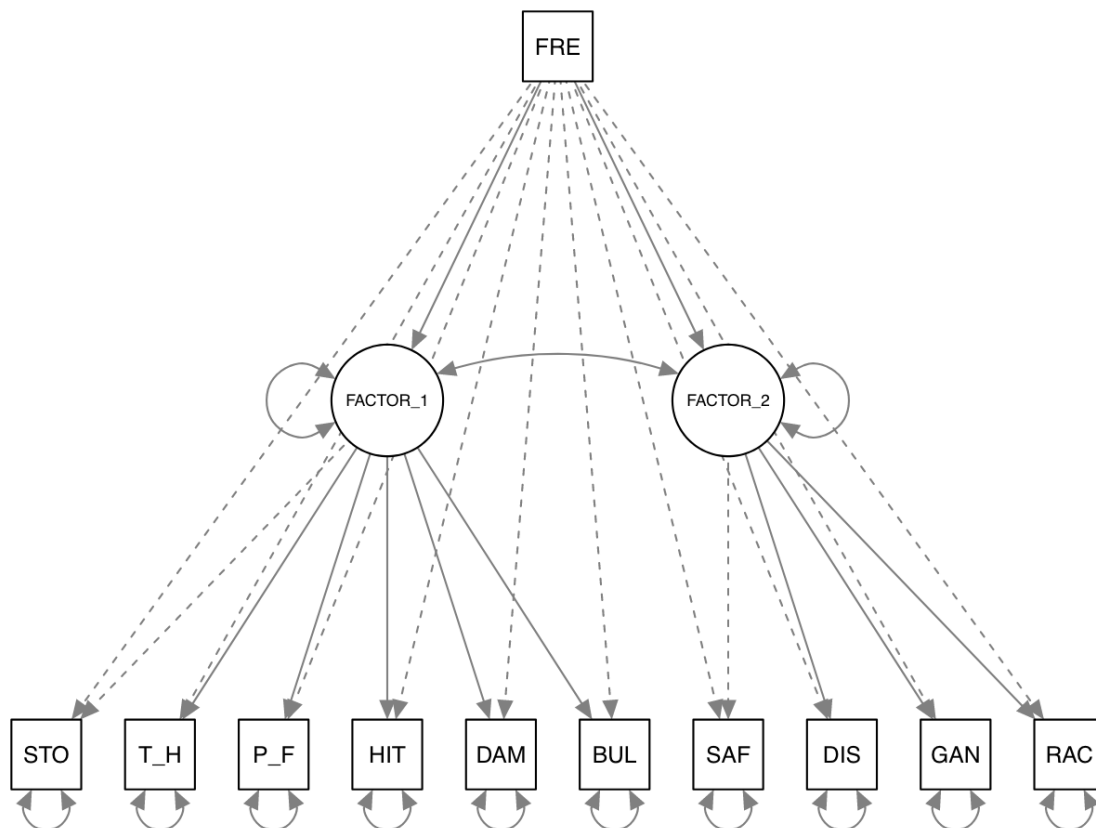
```

---

## 5 MIMIC model 2 - probe for covariate -> indicator DIFF

---





```

mimic_m2 <- mplusObject(
  TITLE = "MIMIC model2 - LAB 8",
  VARIABLE =
    "usevar = freelnch stolen-rac_fght;",

  ANALYSIS =
    "estimator = mlr;",

  MODEL =
    "FACTOR_1 by stolen t_hurt p_fight hit damaged bullied;

    FACTOR_2 by safe disrupt gangs rac_fght;

    FACTOR_1 on freelnch;

    FACTOR_2 on freelnch;

    stolen-rac_fght on freelnch@0; ! to check DIFF see modification indices ",

  PLOT = "type = plot3;",
  OUTPUT = "sampstat standardized residual modindices (.1);",

  usevariables = colnames(mimic_data),
  rdata = mimic_data)

```

```

mimic_m2_fit <- mplusModeler(mimic_m2,
                             dataout=here("mimic_mplus", "lab8_mimic_data.dat"),
                             modelout=here("mimic_mplus", "lab8_mimic_model2.inp"),
                             check=TRUE, run = TRUE, hashfilename = FALSE)

mimic_output2 <- readModels(here("mimic_mplus", "lab8_mimic_model2.out"))

# Plot model:
semPaths(mimic_output2,
         intercepts=FALSE,
         #fixedStyle = c(1)
         )

```

---

## 6 MIMIC model 3 - specify covariate -> indicator DIFF

Number of parameters for MIMIC model 3 = 34

- 8 indicator loadings (10 items - 2 fixed loadings)
- 10 intercepts
- 10 residual variances
- 2 factor variances
- 1 factor co-variance
- 1 covariate mean
- 1 covariate variance
- 1 DIF (covariate -> indicator)

---

```

grViz(" digraph mimic_mode_3 {

graph [overlap = true, fontsize = 12, fontname = Times]

  node [shape = box]
  stolen; t_hurt; p_fight; hit; damaged; bullied; safe; disrupt; gangs; rac_fght;

  node [shape = box, label = 'Percent Free Lunch']
  X;

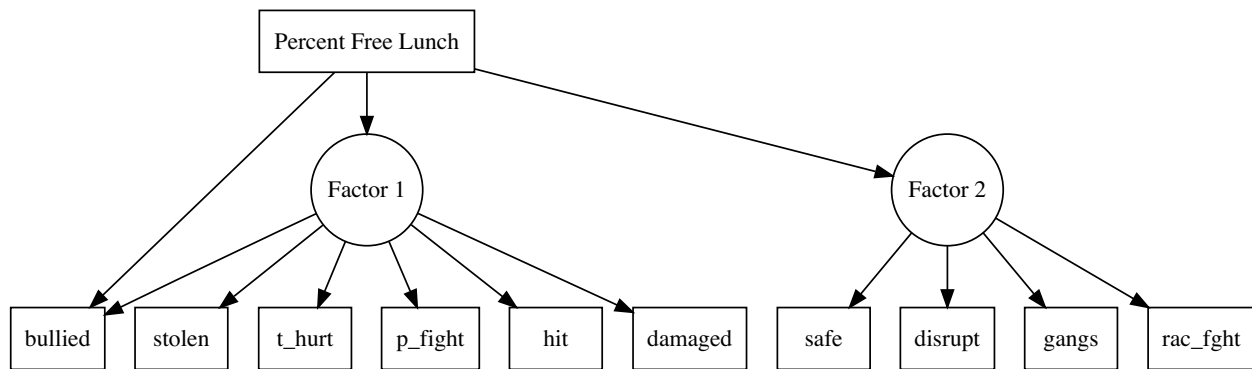
  node [shape = circle, fixedsize = true, width = 0.9, label = 'Factor 1']
  F1;

  node [shape = circle, fixedsize = true, width = 0.9, label = 'Factor 2']
  F2;

  edge [color = black]

  F1->{stolen t_hurt p_fight hit damaged bullied}
  F2->{safe disrupt gangs rac_fght}
  X->F1 X->F2 X->bullied
}
")

```




---

```

mimic_m3 <- mplusObject(
  TITLE = "MIMIC model3 - LAB 8",
  VARIABLE =
    "usevar = freelnch stolen-rac_fght;",

  ANALYSIS =
    "estimator = mlr;",

  MODEL =
    "FACTOR_1 by stolen t_hurt p_fight hit damaged bullied;

    FACTOR_2 by safe disrupt gangs rac_fght;

    FACTOR_1 FACTOR_2 on freelnch;

    bullied on freelnch; ",

  PLOT = "type = plot3;",
  OUTPUT = "sampstat standardized residual modindices (3.84);",

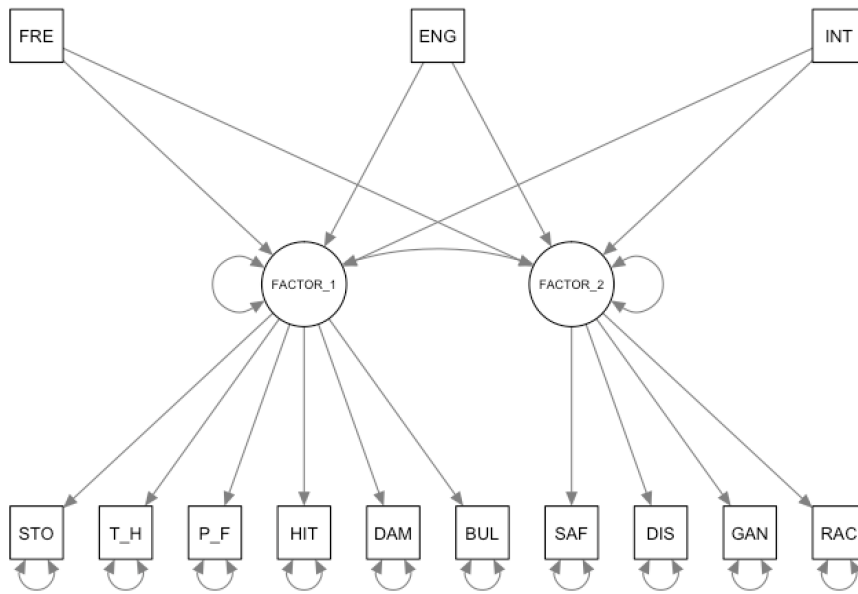
  usevariables = colnames(mimic_data),
  rdata = mimic_data)

mimic_m1_fit <- mplusModeler(mimic_m3,
  dataout=here("mimic_mplus", "lab8_mimic_data.dat"),
  modelout=here("mimic_mplus", "lab8_mimic_model3.inp"),
  check=TRUE, run = TRUE, hashfilename = FALSE)
  
```

---

## 7 MIMIC model 4 - two covariates & an interaction term

---



```

mimic_m4 <- mplusObject(
  TITLE = "MIMIC model4 - LAB 8",
  VARIABLE =
    "usevar = freelnch stolen-rac_fght eng_2nd int;",

  ANALYSIS =
    "estimator = mlr;",

  DEFINE =
    "if bystlang == 1 THEN eng_2nd=0;
     if bystlang == 0 THEN eng_2nd=1;
     int = eng_2nd*freelnch;",

  MODEL =
    "FACTOR_1 by stolen t_hurt p_fight hit damaged bullied;

     FACTOR_2 by safe disrupt gangs rac_fght;

     FACTOR_1 FACTOR_2 on freelnch eng_2nd int; ",

  PLOT = "type = plot3;",
  OUTPUT = "sampstat standardized residual modindices (3.84);",

  usevariables = colnames(mimic_data),
  rdata = mimic_data)

mimic_m4_fit <- mplusModeler(mimic_m4,
  dataout=here("mimic_mplus", "lab8_mimic_data.dat"),
  modelout=here("mimic_mplus", "lab8_mimic_model4.inp"),
  check=TRUE, run = TRUE, hashfilename = FALSE)

```

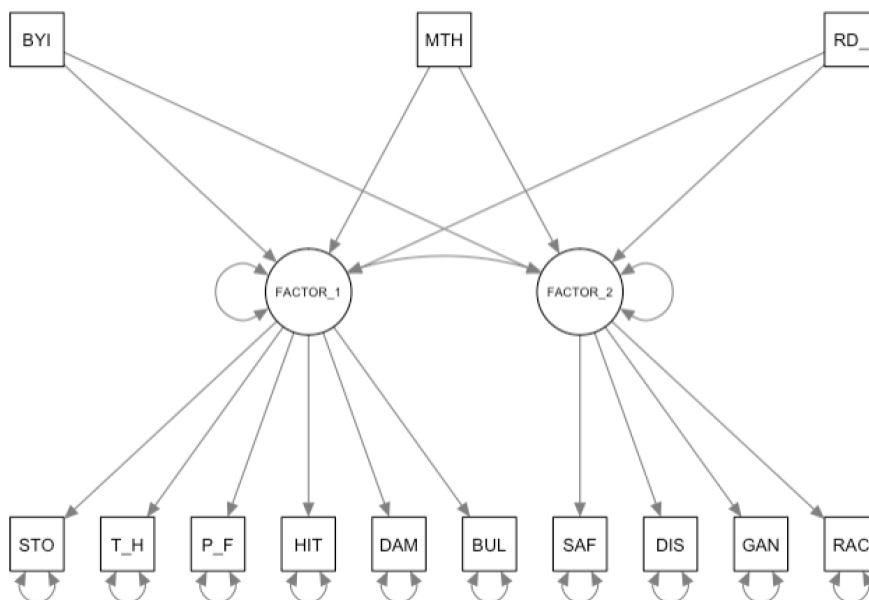
## 7.1 create a path diagram of MIMIC model 4

```
# Read in the model to R within the "cfa_mplus" folder
mimic_output4 <- readModels(here("mimic_mplus", "lab8_mimic_model4.out"))

# Plot model:
semPaths(mimic_output4,
         intercepts=FALSE,
         fixedStyle = c(1))
```

---

## 8 MIMIC model 5 - three continuous covariates



---

```
mimic_m5 <- mplusObject(
  TITLE = "MIMIC model5 - LAB 8",
  VARIABLE =
    "usevar = byincome mth_test rd_test stolen-rac_fght;",
  ANALYSIS =
    "estimator = mlr;",
  MODEL =
    "FACTOR_1 by stolen t_hurt p_fight hit damaged bullied;
    FACTOR_2 by safe disrupt gangs rac_fght;
```

```

    FACTOR_1 FACTOR_2 on byincome mth_test rd_test; ",

PLOT = "type = plot3;",
OUTPUT = "sampstat standardized residual modindices (3.84);",

usevariables = colnames(mimic_data),
rdata = mimic_data)

mimic_m5_fit <- mplusModeler(mimic_m5,
                            dataout=here("mimic_mplus", "lab8_mimic_data.dat"),
                            modelout=here("mimic_mplus", "lab8_mimic_model5.inp"),
                            check=TRUE, run = TRUE, hashfilename = FALSE)

```

---

## 8.1 create a path diagram of MIMIC model 5

```

# Read in the model to R
mimic_output5 <- readModels(here("mimic_mplus", "lab8_mimic_model5.out"))

```

```

# Plot model:
semPaths(mimic_output5,
         intercepts=FALSE,
         fixedStyle = c(1)
         )

```

*# \*\* Lab exercise: comment out the "intercepts" & "fixedStyle" arguments and then count model parameters*

---

## 8.2 practice some formatting with semPlot::semPaths()

```

semPaths(mimic_output5,
         "stdyx", # plot the standardized parameter estimates (see output section: STDYX)
         intercepts=FALSE,
         fixedStyle = c(1),
         color= list(lat = c("light blue"," light green")),
         sizeMan = 10, sizeInt = 10, sizeLat = 10,
         edge.label.cex=.8,
         fade=FALSE
         )

```

---

## 8.3 read all models and create table

```

all_models <- readModels(here("mimic_mplus"))

table <- LatexSummaryTable(all_models,
  keepCols=c(
    "Filename", "Parameters", "ChiSqM_Value",
    "CFI", "TLI", "SRMR", "RMSEA_Estimate",
    "RMSEA_90CI_LB", "RMSEA_90CI_UB"),
  sortBy = "Filename")

table %>%
  kable(booktabs = T, linesep = "",
    col.names = c(
      "Model", "Par", "ChiSq",
      "CFI", "TLI", "SRMR", "RMSEA",
      "Lower CI", "Upper CI")) %>%
  kable_styling(c("striped"),
    full_width = F, position = "left")

```

---

## 9 End of Lab 8

---

### 9.1 References

- Hallquist, M. N., & Wiley, J. F. (2018). MplusAutomation: An R Package for Facilitating Large-Scale Latent Variable Analyses in Mplus. *Structural equation modeling: a multidisciplinary journal*, 25(4), 621-638.
- Horst, A. (2020). Course & Workshop Materials. GitHub Repositories, [https://https://allisonhorst.github.io/](https://allisonhorst.github.io/)
- Muthén, L.K. and Muthén, B.O. (1998-2017). *Mplus User's Guide*. Eighth Edition. Los Angeles, CA: Muthén & Muthén
- R Core Team (2017). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>
- Wickham et al., (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43), 1686, <https://doi.org/10.21105/joss.01686>

# UC SANTA BARBARA