

Lab 3 - Moderation

Structural Equation Modeling ED 216F - Instructor: Karen Nylund-Gibson

Adam Garber

April 16, 2020

Contents

| | | |
|----------|--|----------|
| 1 | Lab preparation | 1 |
| 1.1 | Creating a version-controlled R-Project with Github | 1 |
| 1.2 | Data source for example 1: | 2 |
| 2 | Begin lab 2 exercise | 2 |
| 2.1 | Estimate moderation example 1 | 6 |
| 2.2 | Create the simple slope plot from Mplus model output | 8 |
| 2.3 | Data source for example 2: | 9 |
| 2.4 | Estimate moderation example 2 | 10 |
| 2.5 | Create the simple slope plot from Mplus model output | 12 |

1 Lab preparation

1.1 Creating a version-controlled R-Project with Github

Download repository here: <https://github.com/garberadamc/SEM-Lab3>

On the Github repository webpage:

- fork your own branch of the lab repository
- copy the repository web URL address from the clone or download menu

Within R-Studio:

- click “NEW PROJECT” (upper right corner of window)
- choose option **Version Control**

- e. choose option Git
 - f. paste the repository web URL path copied from the **clone** or **download** menu on Github page
 - g. choose location of the R-Project (**too many nested folders will result in filepath error**)
-

1.2 Data source for example 1:

The first example utilizes the *Vocabulary and Education* dataset from the National Opinion Research Center General Social Survey. GSS Cumulative Datafile 1972-2016 (Fox, 2008) [See documentation here](#)

This dataset is available via the R-package `{carData}` and can be directly loaded into the R environment.

Note: All models specified in the following exercise are for demonstration only and are **not** theoretically justified or valid.

```
# equatiomatic is not yet on CRAN. Install the development version from GitHub with  
remotes::install_github("datalogorax/equatiomatic", force = TRUE)
```

```
library(tidyverse)  
library(MplusAutomation)  
library(rhdf5)  
library(here)  
library(gt)  
library(gtsummary)  
library(estimatr)  
library(kableExtra)  
library(hrbrthemes)  
library(equatiomatic)  
library(effects)  
library(carData)  
library(Ecdat)  
library(huxtable)  
library(flair)
```

2 Begin lab 2 exercise

Read the dataframe into your R-environment from package `{Ecdat}`

```
data(Vocab)  
  
vocab <- as.data.frame(Vocab)
```

Take a look at focal variables, make a `tribble` table

| Name | Labels |
|------------|---|
| year | Year of the survey |
| sex | Sex of the respondent (Female or Male) |
| education | Education, in years |
| vocabulary | Vocabulary test score: number correct on a 10-word test |

check some basic descriptives with the `{gtsummary}` package

```
table1 <- tbl_summary(vocab,
  statistic = list(all_continuous() ~ "{mean} ({sd})"),
  missing = "no" ) %>%
  bold_labels()
table1
```

| Characteristic | N = 30351 ¹ |
|----------------|------------------------|
| year | 1995 (13) |
| sex | |
| Female | 17148 (56%) |
| Male | 13203 (44%) |
| education | 13.0 (3.0) |
| vocabulary | 6.00 (2.12) |

¹Statistics presented: mean (SD); n (%)

Run a regression of the model with interaction of `sex` and `education` using the `lm` function

```
m1_interact <- lm(formula = ~sex + ~sex:education, data = vocab)
```

Print summary of regression output using package `{huxtable}`

```
huxreg(m1_interact, error_pos = 'right')
```

| | (1) | |
|-------------------|------------|---------|
| (Intercept) | 1.788 *** | (0.064) |
| sexMale | -0.277 ** | (0.094) |
| education | 0.329 *** | (0.005) |
| sexMale:education | 0.008 | (0.007) |
| N | 30351 | |
| R2 | 0.231 | |
| logLik | -61825.557 | |
| AIC | 123661.113 | |

*** p < 0.001; ** p < 0.01; * p < 0.05.

Print the Latex syntax for the regression equation using the package `{equationomatic}`

```
extract_eq(m1_interact)
```

$$\text{vocabulary} = \alpha + \beta_1(\text{sex}_{\text{Male}}) + \beta_2(\text{education}) + \beta_3(\text{sex}_{\text{Male}} \times \text{education}) + \epsilon$$

```
extract_eq(m1_interact, use_coefs = TRUE)
```

$$\text{vocabulary} = 1.79 - 0.28(\text{sex}_{\text{Male}}) + 0.33(\text{education}) + 0.01(\text{sex}_{\text{Male}} \times \text{education}) + \epsilon$$

Plot the interaction effect using the package {effects}

```
int <- effect("sex:education", m1_interact)

plot(int, main="", grid=TRUE,
     x.var = "education", xlab="Education",
     ylab="Vocabulary", multiline=TRUE, confint=list(style="auto"))
```

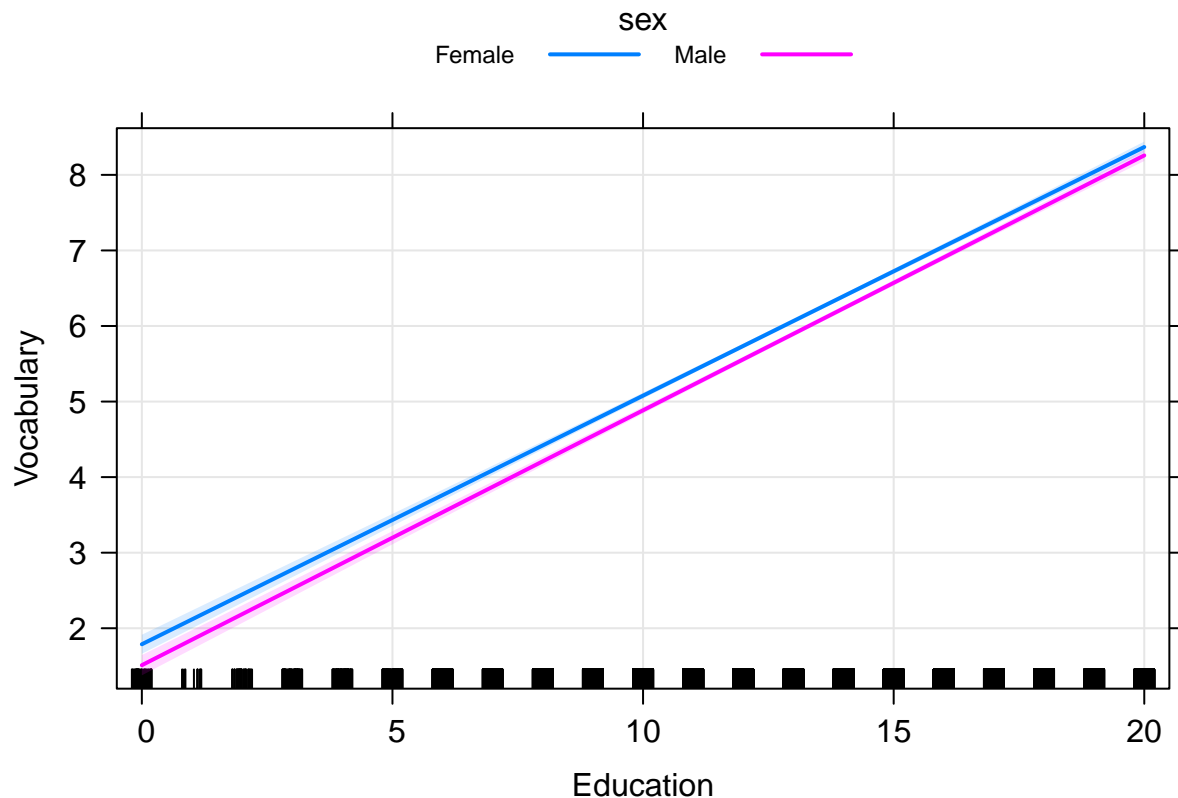


Figure. Example 1 interaction is non-significant (no moderation)

Filter year to include 1974 and 2016 (emphasizing moderation effect)

```
vocab2 <- vocab %>%
  filter(year %in% c(1974, 2016)) %>%
  mutate(year = droplevels(factor(year)))
```

Run regression with interaction between `year` and `education`

```
# below is R-code to center covariate 'education'
# vocab2 <- vocab %>%
#   mutate(education = scale(education, scale = FALSE)) %>%
#   mutate(year = scale(year, scale = FALSE))

m2_interact <- lm(formula =
  vocabulary ~ sex +
    education +
    year +
    year:education ,
  data=vocab2)

huxreg(m2_interact, error_pos = 'right')
```

| | (1) | |
|--------------------|------------|---------|
| (Intercept) | 1.600 *** | (0.184) |
| sexMale | -0.073 | (0.062) |
| education | 0.375 *** | (0.015) |
| year2016 | 0.373 | (0.272) |
| education:year2016 | -0.078 *** | (0.021) |
| N | 3307 | |
| R2 | 0.244 | |
| logLik | -6607.022 | |
| AIC | 13226.043 | |

*** p < 0.001; ** p < 0.01; * p < 0.05.

Print the Latex syntax for the regression equation using the package `{equationomatic}`

```
extract_eq(m2_interact)
```

$$\text{vocabulary} = \alpha + \beta_1(\text{sex}_{\text{Male}}) + \beta_2(\text{education}) + \beta_3(\text{year}_{2016}) + \beta_4(\text{education} \times \text{year}_{2016}) + \epsilon$$

```
extract_eq(m2_interact, use_coefs = TRUE)
```

$$\text{vocabulary} = 1.6 - 0.07(\text{sex}_{\text{Male}}) + 0.38(\text{education}) + 0.37(\text{year}_{2016}) - 0.08(\text{education} \times \text{year}_{2016}) + \epsilon$$

Plot the interaction effect using the package `{effects}`

```
int2 <- effect("education:year", m2_interact)

plot(int2, main="", grid=TRUE,
x.var = "education", xlab="Education",
ylab="Vocabulary",
multiline=TRUE,
confint=list(style="auto"))
```

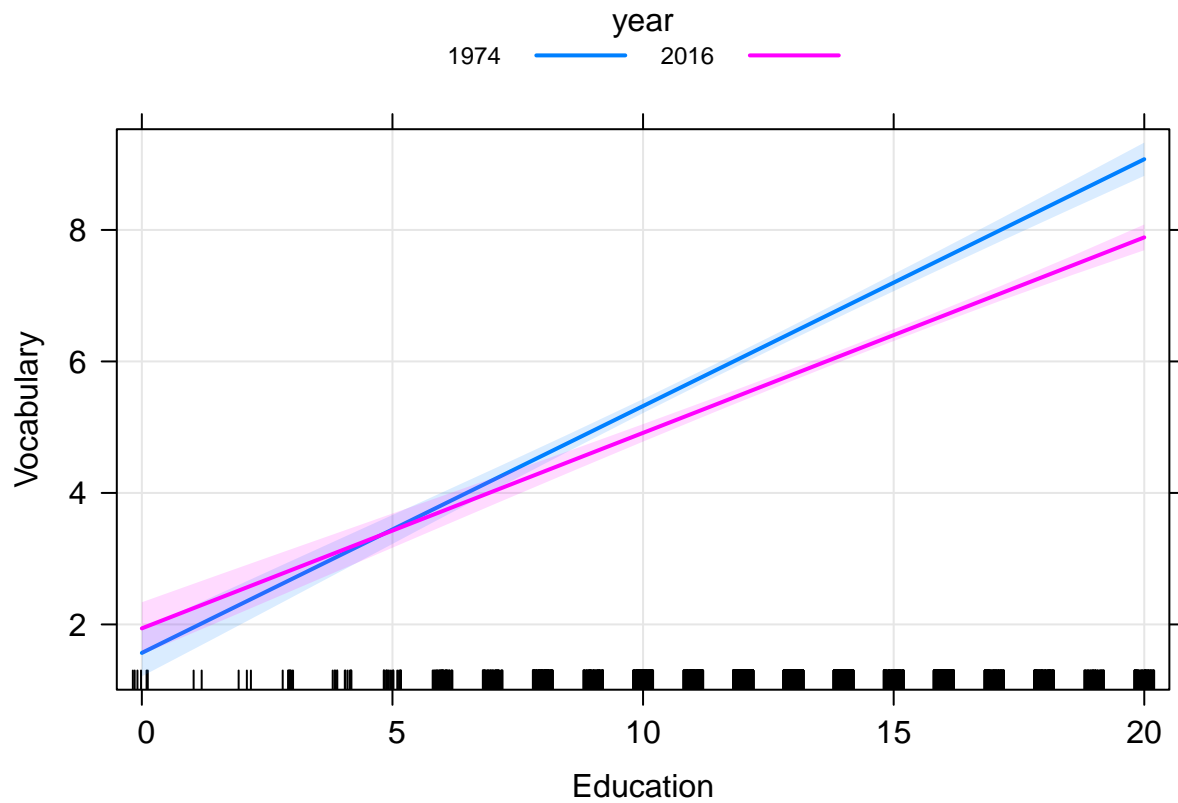
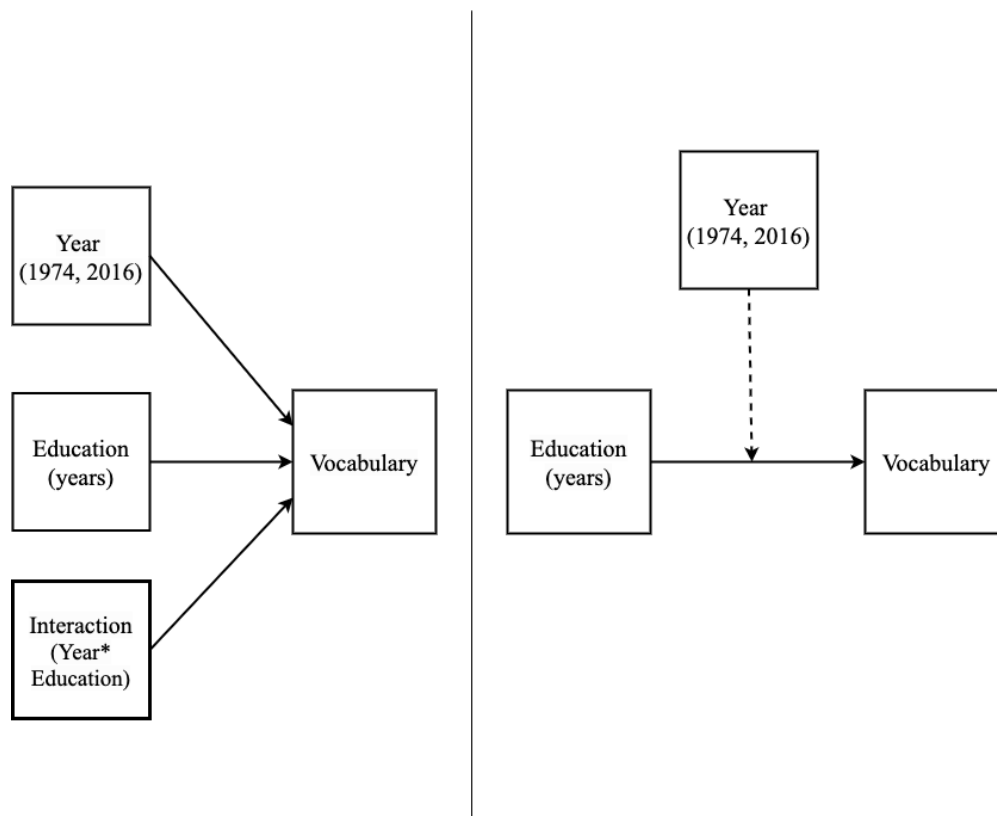


Figure. Example 2 interaction term is significant (moderation)

2.1 Estimate moderation example 1

1. covariate: Years of education (**education**)
2. moderator: Year of the survey with 2-levels 1974 and 2016 (**year**)
3. outcome: Vocabulary test score - number correct on a 10-word test (**vocabulary**)



```
education_2sd <- 2*sqrt(9.85) # 6.28

m1_model <- mplusObject(
  TITLE = "m5 model indirect - Lab 3",
  VARIABLE =
    "usevar =
      year          ! covariate/moderator
      education     ! covariate
      vocabulary    ! outcome
      int_yred;     ! interaction of year and education",

  DEFINE =
    "center education (grandmean);
      int_yred = year*education; ! create interaction term ",

  ANALYSIS =
    "estimator = MLR" ,

  MODEL =
    "[vocabulary](b0);
      vocabulary on
      year(b1)
      education(b2)
      int_yred(b3); " ,
```

```

MODELCONSTRAINT =
"LOOP(x,-1,1,0.01);
PLOT(y1974 y2016);
new(hi_y1974 lo_y1974 hi_y2016 lo_y2016 diff_hi);
y1974 = b0 + b2*x;
y2016 = b0 + b1 + (b2+b3)*x;

hi_y1974 = b0 + b2*(6.28);
lo_y1974 = b0 + b2*(-6.28);
hi_y2016 = b0 + b1 + (b2 + b3)*(6.28);
lo_y2016 = b0 + b1 + (b2 + b3)*(-6.28);
diff_hi = hi_y2016 - hi_y1974; ",

OUTPUT = "sampstat standardized modindices (3.84)",

PLOT = "type=plot3;",

usevariables = colnames(vocab2),
rdata = vocab2)

m1_model_fit <- mplusModeler(m1_model,
                             dataout=here("mplus_files", "Lab3.dat"),
                             modelout=here("mplus_files", "model1_Lab3.inp"),
                             check=TRUE, run = TRUE, hashfilename = FALSE)

```

2.2 Create the simple slope plot from Mplus model output

Extract the output parameters generated using the model constraint

```

simp_slope <- data.frame(m1_model_fit[["results"]][["parameters"]][["unstandardized"]]) %>%
  filter(paramHeader == "New.Additional.Parameters") %>%
  filter(param != "DIFF_HI") %>%
  select(param, est, se) %>%
  mutate(year = case_when(
    param %in% c("HI_Y1974", "LO_Y1974") ~ "1974",
    param %in% c("HI_Y2016", "LO_Y2016") ~ "2016")) %>%
  mutate(education = case_when(
    param %in% c("HI_Y1974", "HI_Y2016") ~ 9,
    param %in% c("LO_Y1974", "LO_Y2016") ~ -9))

```

Plot the interaction effect with `ggplot` using theme from `{hrbrthemes}` package

```

# un-center 'education' so values on x-axis are on the original scale
plot_data <- simp_slope %>% mutate(education = education + 12.9)

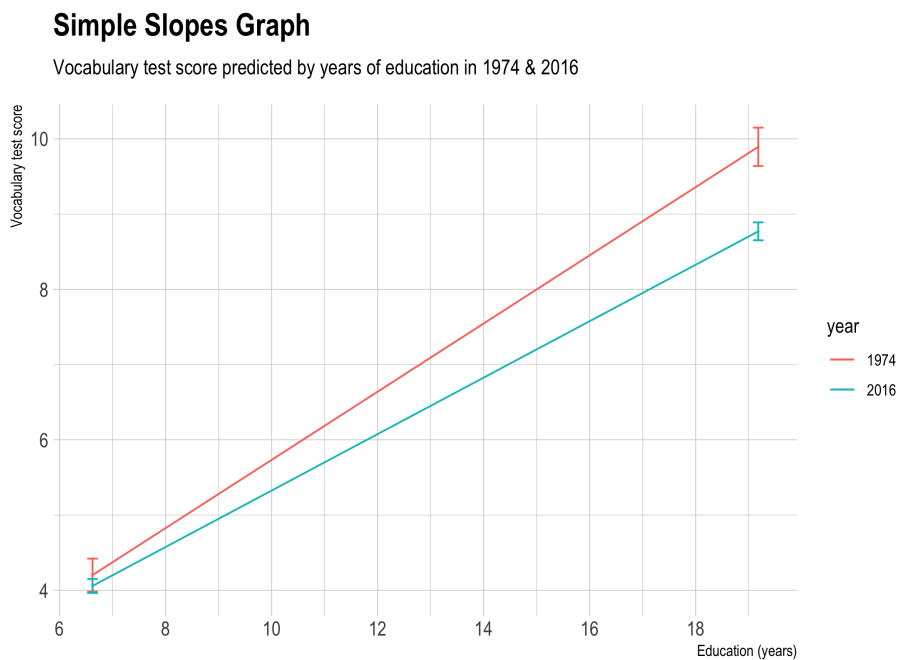
ggplot(plot_data, aes(x=education, y=est, color=year, group=year)) +
  geom_point(size=0) +

```



```
geom_line() +
geom_errorbar(aes(ymin=est-se, ymax=est+se),
              width=.2) +
scale_x_continuous(breaks = c(seq(6,20,2))) +
labs(title = "Simple Slopes Graph",
      subtitle = "Vocabulary test score predicted by years of education in 1974 & 2016",
      x = "Education (years)",
      y = "Vocabulary test score") +
theme_ipsum()

ggsave(here("figures", "m1_simple_slope.png"), height = 6, width = 8)
```



2.3 Data source for example 2:

The next example utilizes the **Effects on Learning of Small Class Sizes (Star)** dataset from the *Introduction to Econometrics* textbook. (Stock et al., 2003) [See documentation here](#)

This dataset is available via the R-package {Ecdat} and can be directly loaded into the R environment.

Read the dataframe into your R-environment from package {Ecdat}

```
data(Star)

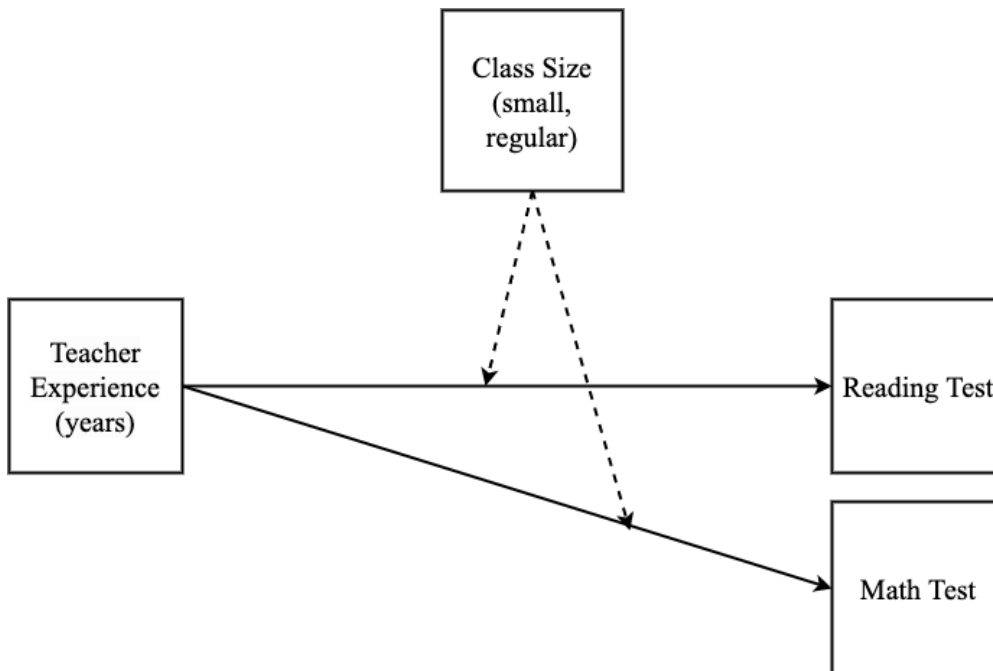
star_data <- as.data.frame(Star)
```

Take a look at the variables in the **Star** dataset

| Name | Labels |
|----------|---|
| tmathssk | total math scaled score |
| treadssk | total reading scaled score |
| classk | type of class (small, regular, regular with aide) |
| totexpk | years of total teaching experience |

Subset and recode variables to use in moderation model with `select`, `mutate`, and `case_when`

```
mod_data <- star_data %>%  
  select(totexpk, # years of total teaching experience  
         classk, # type of class, a factor with levels (regular,small.class,regular.with.aide)  
         tmathssk, treadssk) %>%  
  mutate(classk = case_when(  
    classk == "small.class" ~ "small.class",  
    classk %in% c("regular.with.aide", "regular") ~ "regular")) %>%  
  mutate(classk = fct_rev(classk))
```



2.4 Estimate moderation example 2

1. covariate: Years of education (`totexpk`)
 2. moderator: type of class (small, regular) (`classk`)
 3. outcome 1: total math scaled score (`tmathssk`)
 4. outcome 2: total reading scaled score (`treadssk`)
-

```

teach_exp_2sd <- sqrt(33.261) # 5.77

m2_model <- mplusObject(
  TITLE = "m2 model indirect - Lab 3",
  VARIABLE =
    "usevar =
      totexpk classk
      tmathssk, treadssk
      tchXclas; ",

  DEFINE =
    "center totexpk (grandmean);
      tchXclas = totexpk*classk; ! create interaction term" ,

  ANALYSIS =
    "estimator = mlr; ",

  MODEL =
    "treadssk on classk totexpk tchXclas;

      [tmathssk](b0);

      tmathssk on
      classk (b1)
      totexpk (b2)
      tchXclas (b3); ",

  MODELCONSTRAINT =
    "LOOP(x,-1,1,0.01);
      PLOT(small regular);
      new(hi_small lo_small hi_regular lo_regular diff_hi);
      small = b0 + b2*x;
      regular = b0 + b1 + (b2+b3)*x;

      hi_small = b0 + b2*(9.3);
      lo_small = b0 + b2*(-9.3);
      hi_regular = b0 + b1 + (b2 + b3)*(9.3);
      lo_regular = b0 + b1 + (b2 + b3)*(-9.3);
      diff_hi = hi_small - hi_regular; ",

  OUTPUT = "sampstat standardized modindices (3.84)",

  PLOT = "type=plot3;",

  usevariables = colnames(mod_data),
  rdata = mod_data)

m2_model_fit <- mplusModeler(m2_model,
  dataout=here("mplus_files", "Lab3_caschools.dat"),
  modelout=here("mplus_files", "model2_Lab2.inp"),
  check=TRUE, run = TRUE, hashfilename = FALSE)

```

2.5 Create the simple slope plot from Mplus model output

```
simp_slope2 <- data.frame(m2_model_fit[["results"]][["parameters"]][["unstandardized"]]) %>%
  filter(paramHeader == "New.Additional.Parameters") %>%
  filter(param != "DIFF_HI") %>%
  select(param, est, se) %>%
  mutate(size = case_when(
    param %in% c("HI_SMALL", "LO_SMALL") ~ "Small",
    param %in% c("HI_REGUL", "LO_REGUL") ~ "Regular")) %>%
  mutate(experience = case_when(
    param %in% c("HI_SMALL", "HI_REGUL") ~ 9.3,
    param %in% c("LO_SMALL", "LO_REGUL") ~ -9.3))

# un-center 'experience' so values on x-axis are on the original scale
mean_exp <- mean(mod_data$totexpk)
plot_data2 <- simp_slope2 %>% mutate(experience = experience + mean_exp)

ggplot(plot_data2, aes(x=experience, y=est, color=size, group=size)) +
  geom_point(size=0) +
  geom_line() +
  geom_errorbar(aes(ymin=est-se, ymax=est+se), width=.25) +
  scale_x_continuous( breaks = c(seq(0,18,2))) +
  labs(title = "Simple Slopes Graph",
       subtitle = "Math test score predicted by years of teaching experience in small & regular classrooms",
       x = "Teaching Experience (years)",
       y = "Math test score") +
  theme_ipsum()

ggsave(here("figures", "m2_simple_slope.png"), height = 6, width = 8)
```

Simple Slopes Graph

Math test score predicted by years of teaching experience in small & regular classrooms

