Homework 3

JP

1/24/2020

In this assignment you will be working with publicly available data from the “High School and Beyond” survey as described by Raudenbush et al (2004) and used in the Rabe-Hesketh and Skrondal text. This data can be found on the course website in Canvas under the “Files” tab.Look for data file “hsb.dta.”Use the STATA commands handout (also available in the Assignments tab) to help you complete this assignment. On the due date please hand in a hardcopy of your STATA output and answers to all questions in this assignment. You can either hand in a document with your answers (e.g., Word doc) with STATA output attached or you can incorporate answers to the questions directly into the STATA output in the form of comments. I prefer typed assignments in Times New Roman size 12 or Arial size 11. \* Raudenbush, S.W., A.S. Bryk, Y.F. Cheong, and R. Congdon. 2004. HLM 6: Hierarchical Linear and Nonlinear Modeling. Lincolnwood, IL: Scientific Software International. The following is a basic description of the variables you will be using in this assignment.

* schoolid = a unique id number assigned to each school in the data set.
* mathach = a continuous measure of mathematics achievement.
* ses = a continuous measure of socioeconomic status based on parental education, occupation, and income. This predictor is mean centered.
* female = a dichotomous indicator of gender [1 = female, 0 = male].

library(tidyverse)

## -- Attaching packages ----------------------------------------------------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.2.0 v purrr 0.3.2  
## v tibble 2.1.3 v dplyr 0.8.3  
## v tidyr 1.0.2 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.4.0

## -- Conflicts -------------------------------------------------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(readstata13)

## Warning: package 'readstata13' was built under R version 3.6.2

library(lme4)

## Loading required package: Matrix

##   
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':  
##   
## expand, pack, unpack

library(psych)

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

# install.packages(c('optimx', 'lmerTest')  
library(optimx)

## Warning: package 'optimx' was built under R version 3.6.2

library(lmerTest)

## Warning: package 'lmerTest' was built under R version 3.6.2

##   
## Attaching package: 'lmerTest'

## The following object is masked from 'package:lme4':  
##   
## lmer

## The following object is masked from 'package:stats':  
##   
## step

options(max.print = 99999)  
options(scipen = 999)  
  
getwd()

## [1] "E:/UO/R Projects/SOC 613/scripts"

set.seed(12420)  
  
data <- read.dta13("E:/UO/R Projects/SOC 613/data/hsb.dta")  
# data <- read.dta13("/Volumes/JPEDROZA/UO/R Projects/SOC 613/data/hsb.dta")  
  
colnames(data)

## [1] "minority" "female" "ses" "mathach" "size" "sector"   
## [7] "pracad" "disclim" "himinty" "schoolid" "mean" "sd"   
## [13] "sdalt" "junk" "sdalt2" "num" "se" "sealt"   
## [19] "sealt2" "t2" "t2alt" "pickone" "mmses" "mnses"   
## [25] "xb" "resid"

data <- data %>%   
 dplyr::select(schoolid,   
 mathach,  
 ses,  
 female)

## Tasks:

1. Quick Exploration of Data:
2. How many respondents are in the sample?

Answer: 7185

data %>%   
 count()

## # A tibble: 1 x 1  
## n  
## <int>  
## 1 7185

1. How many schools are in the sample? \*Hint: recall useful piece of STATA code from session on Random Slopes (see slides).

Answer: 160 different schools.

data %>%   
 group\_by(schoolid) %>%   
 summarize(n = n()) %>%   
 ungroup() %>%   
 count(schoolid) %>%   
 summarize(schools = sum(n))

## # A tibble: 1 x 1  
## schools  
## <int>  
## 1 160

1. Obtain summary statistics (n mean sd min max) for continuous variables mathach and SES.

Answer: Below in code.

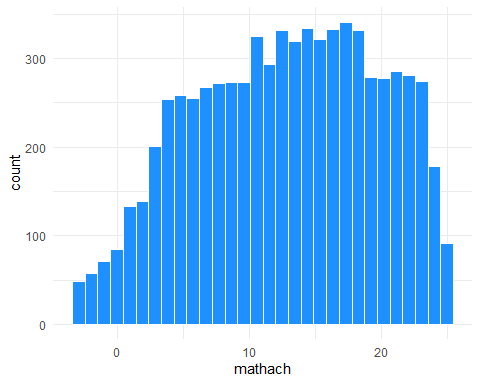
# data %>%  
# dplyr::select(mathach, ses) %>%   
# psych::describe(na.rm = TRUE)  
  
library(sundry)  
data %>%   
 descrips(mathach, ses)

## variable n min max mean sd  
## 1 mathach 7185 -2.832 24.993 12.747852604 6.8782457  
## 2 ses 7185 -3.758 2.692 0.000143354 0.7793552

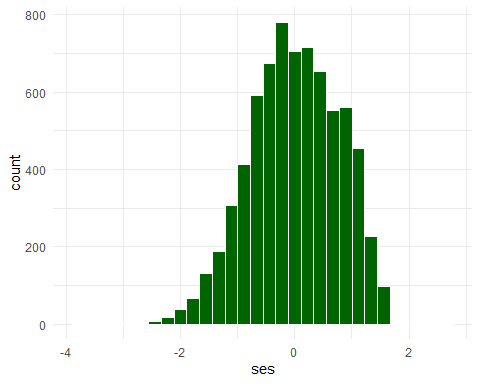
1. Plot histograms for the two continuous variables (mathach and SES).

Answers: Below in code.

data %>%   
 ggplot(aes(mathach)) +  
 geom\_histogram(color = 'white', fill = 'dodgerblue', bins = 30) +  
 theme\_minimal()



data %>%   
 ggplot(aes(ses)) +  
 geom\_histogram(color = 'white', fill = 'darkgreen', bins = 30) +  
 theme\_minimal()



1. Obtain frequency statistics for the gender variable female (how many females and males are there in the sample? What proportion of the respondents are female?)

Answer: Below in code. n is the number of males (0) and females (1) and freq is the proportions of each. The proportion of females in the sample is .53 or 53% of the sample.

data %>%   
 group\_by(female) %>%   
 summarize(n = n()) %>%   
 mutate(freq = n/sum(n))

## # A tibble: 2 x 3  
## female n freq  
## <int> <int> <dbl>  
## 1 0 3390 0.472  
## 2 1 3795 0.528

1. For each model listed below, write the model (including any necessary in-between stages such as a micro and macro model) using proper notation. Interpret each parameter in the final combined model (fixed betas and random variance parameters). This time don’t worry about interpreting the parameters that only show up in the “halfway” stage (i.e., in the micro and macro models).
2. Model 1: single-level model, outcome is mathach and SES is a fixed effect predictor. $$

Mathach\_i = \_{0i} + *1(ses\_i) +* {0i}

$$

1. Model 2: random intercept model nesting students in schools, outcome is mathach and SES is a fixed effect predictor.

Micro: $$

Mathach\_{ij} = *{0ij} + 1(ses{ij}) +* {0ij}

$$

Macro: $$

\_{oj} = *0 +* {0j}

$$

Combined: $$

Mathach\_{ij} = *0 + 1(ses{ij}) +* {0j} + \_{0ij}

$$

$$

Level 2: [\_{0j}] N(0, ^2\_{0})

$$

$$

Level 1: [\_{0ij}] N(0,^2\_{0})

$$

Interpretations: \beta\_0 is the BMI for an adolescent (across all schools) \beta\_{0j} is the BMI for school j \epsilon\_{0ij} is the residual differntial BMI for adolescent i in school j \mu\_{0j} is the residual differential BMI for an adolescent in school j \sigma^2\_{\mu0} is the between-school variance of BMI \sigma^2\_{\epsilon0} is the within-school/between-person variance of BMI

1. Model 3: random slope model nesting students in schools (assume covariance =0). Outcome is mathach and SES is a random slope parameter.

Micro: $$

Mathach\_{ij} = *{0j} +* {1j}(ses\_{ij}) + \_{0ij}

$$

Macro 1:(Intercept) $$

\_{0j} = *o +* {0j}

$$

Macro 2: (Slope) $$

\_{1j} = *1 +* {1j}

$$

Combined: $$

Mathach\_{ij} = *0 + 1(ses{ij}) +* {0j} + *{1j}(ses*{ij}) + \_{0ij}

$$

NEED TO INCLUDE THE MATRICES FOR LEVEL 1 AND 2 WITH LEVEL 2 HAVING A COVARIANCE OF ZERO. Specifically, for the covariance-variance matrix.

1. Model 4: random slope model nesting students in schools with covariance allowed to differ from 0 (i.e., be + or -). Outcome is mathach and SES is the random slope parameter. You only need to write a parameter interpretation for the level 2 covariance parameter. The other parameter interpretations will be identical to the parameter interpretations in Model 3. Do write the model itself using correct notation, including any in-between models (micro, macros).

Micro:

Macro 1:(Intercept)

Macro 2: (Slope)

Combined:

NEED THE MATRICES FOR THIS ONE WITH THE CORRECT COVARIANCE.

1. For each model specified in Question 2, fit the model using STATA (or software of your choice). For Models 2, 3 and 4 you will want to store the results for use in Question 4 below. To store the results for use in LR testing use the “estimates store” command.

sub <- data %>%   
 group\_by(schoolid) %>%   
 mutate(n\_per = n())  
  
sub\_add <- sub %>%   
 group\_by(schoolid) %>%   
 summarize(n = n()) %>%   
 ungroup() %>%   
 count(schoolid)  
  
sub <- left\_join(sub, sub\_add, by = 'schoolid')  
  
sub <- sub %>%   
 group\_by(schoolid) %>%  
 mutate(one\_school = !duplicated(n))  
   
colnames(sub)

## [1] "schoolid" "mathach" "ses" "female" "n\_per"   
## [6] "n" "one\_school"

sub <- sub %>%   
 mutate(one\_school = ifelse(one\_school == "TRUE", 1, 0))

# install.packages('sjstats')  
  
single\_model <- lm(mathach ~ ses, data = sub)  
summary(single\_model)

##   
## Call:  
## lm(formula = mathach ~ ses, data = sub)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -19.4382 -4.7580 0.2334 5.0649 15.9007   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 12.74740 0.07569 168.42 <0.0000000000000002 \*\*\*  
## ses 3.18387 0.09712 32.78 <0.0000000000000002 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6.416 on 7183 degrees of freedom  
## Multiple R-squared: 0.1301, Adjusted R-squared: 0.13   
## F-statistic: 1075 on 1 and 7183 DF, p-value: < 0.00000000000000022

confint(single\_model)

## 2.5 % 97.5 %  
## (Intercept) 12.599028 12.895764  
## ses 2.993485 3.374256

ran\_null <- lmer(mathach ~ 1 + (1 | schoolid), data = sub, REML = FALSE)  
summary(ran\_null)

## Linear mixed model fit by maximum likelihood . t-tests use  
## Satterthwaite's method [lmerModLmerTest]  
## Formula: mathach ~ 1 + (1 | schoolid)  
## Data: sub  
##   
## AIC BIC logLik deviance df.resid   
## 47121.8 47142.4 -23557.9 47115.8 7182   
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -3.06262 -0.75365 0.02676 0.76070 2.74184   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## schoolid (Intercept) 8.553 2.925   
## Residual 39.148 6.257   
## Number of obs: 7185, groups: schoolid, 160  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) 12.6371 0.2436 157.6209 51.87 <0.0000000000000002 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

ran\_int <- lmer(mathach ~ ses + (1 | schoolid), data = sub, REML = FALSE)  
summary(ran\_int)

## Linear mixed model fit by maximum likelihood . t-tests use  
## Satterthwaite's method [lmerModLmerTest]  
## Formula: mathach ~ ses + (1 | schoolid)  
## Data: sub  
##   
## AIC BIC logLik deviance df.resid   
## 46649.0 46676.5 -23320.5 46641.0 7181   
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -3.1263 -0.7277 0.0220 0.7578 2.9186   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## schoolid (Intercept) 4.729 2.175   
## Residual 37.030 6.085   
## Number of obs: 7185, groups: schoolid, 160  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) 12.6576 0.1873 149.1760 67.57 <0.0000000000000002 \*\*\*  
## ses 2.3915 0.1057 6837.3052 22.63 <0.0000000000000002 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## ses 0.003

ran\_slope\_zerocov <- lmer(mathach ~ ses + (ses || schoolid), data = sub, REML = FALSE,  
 control = lmerControl(optimizer ="Nelder\_Mead"))  
summary(ran\_slope\_zerocov)

## Linear mixed model fit by maximum likelihood . t-tests use  
## Satterthwaite's method [lmerModLmerTest]  
## Formula: mathach ~ ses + (ses || schoolid)  
## Data: sub  
## Control: lmerControl(optimizer = "Nelder\_Mead")  
##   
## AIC BIC logLik deviance df.resid   
## 46646.7 46681.1 -23318.4 46636.7 7180   
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -3.12438 -0.73150 0.02257 0.75489 2.93129   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## schoolid (Intercept) 4.8097 2.193   
## schoolid.1 ses 0.4096 0.640   
## Residual 36.8234 6.068   
## Number of obs: 7185, groups: schoolid, 160  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) 12.6530 0.1895 147.1571 66.76 <0.0000000000000002 \*\*\*  
## ses 2.3967 0.1180 159.8371 20.32 <0.0000000000000002 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## ses 0.000

confint(ran\_slope\_zerocov, method = 'profile', level = .95)

## Computing profile confidence intervals ...

## 2.5 % 97.5 %  
## .sig01 1.9171609 2.5169128  
## .sig02 0.1310682 0.9726241  
## .sigma 5.9680634 6.1712226  
## (Intercept) 12.2783813 13.0263467  
## ses 2.1604488 2.6338582

as.data.frame(VarCorr(ran\_slope\_zerocov))

## grp var1 var2 vcov sdcor  
## 1 schoolid (Intercept) <NA> 4.8096704 2.1930961  
## 2 schoolid.1 ses <NA> 0.4096306 0.6400239  
## 3 Residual <NA> <NA> 36.8234438 6.0682323

ran\_slope <- lmer(mathach ~ ses + (ses | schoolid), data = sub, REML = FALSE,  
 control = lmerControl(optimizer ="Nelder\_Mead"))  
summary(ran\_slope)

## Linear mixed model fit by maximum likelihood . t-tests use  
## Satterthwaite's method [lmerModLmerTest]  
## Formula: mathach ~ ses + (ses | schoolid)  
## Data: sub  
## Control: lmerControl(optimizer = "Nelder\_Mead")  
##   
## AIC BIC logLik deviance df.resid   
## 46648.5 46689.7 -23318.2 46636.5 7179   
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -3.12298 -0.73010 0.02184 0.75598 2.94309   
##   
## Random effects:  
## Groups Name Variance Std.Dev. Corr   
## schoolid (Intercept) 4.7853 2.1875   
## ses 0.3983 0.6311 -0.11  
## Residual 36.8315 6.0689   
## Number of obs: 7185, groups: schoolid, 160  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) 12.6656 0.1891 146.4054 66.98 <0.0000000000000002 \*\*\*  
## ses 2.3949 0.1177 158.4812 20.35 <0.0000000000000002 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## ses -0.046

confint(ran\_slope, method = 'profile', level = .95)

## Computing profile confidence intervals ...

## Warning in FUN(X[[i]], ...): non-monotonic profile for .sig02

## Warning in confint.thpr(pp, level = level, zeta = zeta): bad spline fit  
## for .sig02: falling back to linear interpolation

## Warning in regularize.values(x, y, ties, missing(ties)): collapsing to  
## unique 'x' values

## 2.5 % 97.5 %  
## .sig01 1.91159297 2.5112992  
## .sig02 -1.00000000 0.3078605  
## .sig03 0.08461397 0.9666699  
## .sigma 5.96867353 6.1719565  
## (Intercept) 12.28852891 13.0406704  
## ses 2.15925022 2.6317003

as.data.frame(VarCorr(ran\_slope))

## grp var1 var2 vcov sdcor  
## 1 schoolid (Intercept) <NA> 4.7852815 2.1875286  
## 2 schoolid ses <NA> 0.3983042 0.6311135  
## 3 schoolid (Intercept) ses -0.1558100 -0.1128585  
## 4 Residual <NA> <NA> 36.8315447 6.0688998

1. Using the results from STATA:
2. Was SES a statistically significant fixed effect predictor in the Models, and if so which models? What was the nature of the relationship between SES and mathach overall?

For the fixed model (single\_model), ses was a significant predictor, with increases in ses being associated with an increase in mathach. For the random intercept model (ran\_int), ses was a significant predictor, with increases in ses being associated with an increase in mathach. For the random slope model with a zero covariance (ran\_slope\_zerocov), the fixed effect of ses was significantly associated with an increase in mathach. For the random slope model allowing for covariance (ran\_slope), the fixed effect of ses was significantly associated with an increase in mathach.

1. Use likelihood ratio tests to compare the random intercept and random slope model, and then to compare the random slope model with the random slope model that allows for covariance.

anova(ran\_null, ran\_int)

## Data: sub  
## Models:  
## ran\_null: mathach ~ 1 + (1 | schoolid)  
## ran\_int: mathach ~ ses + (1 | schoolid)  
## Df AIC BIC logLik deviance Chisq Chi Df  
## ran\_null 3 47122 47142 -23558 47116   
## ran\_int 4 46649 46677 -23321 46641 474.81 1  
## Pr(>Chisq)   
## ran\_null   
## ran\_int < 0.00000000000000022 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova(ran\_int, ran\_slope\_zerocov)

## Data: sub  
## Models:  
## ran\_int: mathach ~ ses + (1 | schoolid)  
## ran\_slope\_zerocov: mathach ~ ses + (ses || schoolid)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## ran\_int 4 46649 46677 -23321 46641   
## ran\_slope\_zerocov 5 46647 46681 -23318 46637 4.2593 1 0.03904  
##   
## ran\_int   
## ran\_slope\_zerocov \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova(ran\_slope\_zerocov, ran\_slope)

## Data: sub  
## Models:  
## ran\_slope\_zerocov: mathach ~ ses + (ses || schoolid)  
## ran\_slope: mathach ~ ses + (ses | schoolid)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## ran\_slope\_zerocov 5 46647 46681 -23318 46637   
## ran\_slope 6 46648 46690 -23318 46636 0.2762 1 0.5992

1. Which model is the best fit? In your own words, what do these results indicate about the relationship between SES and math achievement scores? Are there “school effects” and if so, how are schools relevant to this relationship between SES and math achievement in the HSB sample?

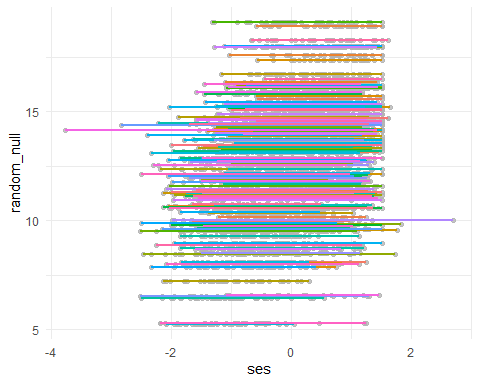
According to the likelihood ratio test, there was a significant increase in fit from the random intercept model to the random slopes model with zero covariance. However, there was no difference in fit between the zero covariance random slopes model and the random slopes model where covariance was allowed. Therefore, the best fitting model is the random slopes model with no covariance.

1. Use the “predict” command and other code from the in-class example to create twoway plots of the following models:

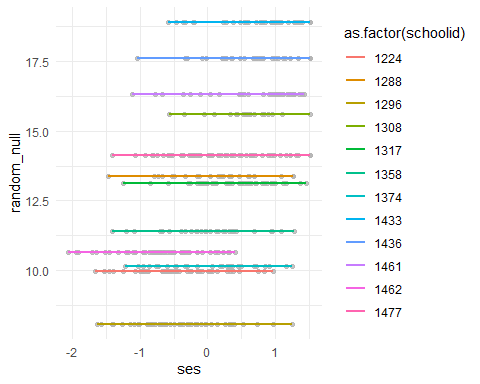
sub$fixed\_value <- predict(single\_model, newdata = sub)  
sub$fixed\_value <- as.numeric(sub$fixed\_value)  
sub$random\_null <- predict(ran\_null, newdata = sub)  
sub$random\_int <- predict(ran\_int, newdata = sub)  
sub$random\_nocv <- predict(ran\_slope\_zerocov, newdata = sub)  
sub$random\_cv <- predict(ran\_slope, newdata = sub)  
  
null\_mu0 <- ranef(ran\_null, condVar = TRUE)  
null\_mu0 <- as.data.frame(null\_mu0)  
  
null\_mu0 <- null\_mu0 %>%   
 rename(null\_term = term,  
 schoolid = grp,  
 null\_diff = condval,  
 null\_se = condsd)  
  
null\_mu0$schoolid <- as.character(null\_mu0$schoolid)  
null\_mu0$schoolid <- as.numeric(null\_mu0$schoolid)  
  
int\_mu0 <- ranef(ran\_int, condVar = TRUE)  
int\_mu0 <- as.data.frame(int\_mu0)  
  
int\_mu0 <- int\_mu0 %>%   
 rename(ranint\_term = term,  
 schoolid = grp,  
 ran\_int\_diff = condval,  
 ran\_int\_se = condsd)  
  
int\_mu0$schoolid <- as.character(int\_mu0$schoolid)  
int\_mu0$schoolid <- as.numeric(int\_mu0$schoolid)  
  
nocv\_mu0 <- ranef(ran\_slope\_zerocov, condVar = TRUE)  
nocv\_mu0 <- as.data.frame(nocv\_mu0)  
  
nocv\_mu0 <- nocv\_mu0 %>%   
 rename(nocv\_term = term,  
 schoolid = grp,  
 nocv\_diff = condval,  
 nocv\_se = condsd)  
  
nocv\_mu0$schoolid <- as.character(nocv\_mu0$schoolid)  
nocv\_mu0$schoolid <- as.numeric(nocv\_mu0$schoolid)  
  
cv\_mu0 <- ranef(ran\_slope, condVar = TRUE)  
cv\_mu0 <- as.data.frame(cv\_mu0)  
  
cv\_mu0 <- cv\_mu0 %>%   
 rename(cv\_term = term,  
 schoolid = grp,  
 cv\_diff = condval,  
 cv\_se = condsd)  
  
cv\_mu0$schoolid <- as.character(cv\_mu0$schoolid)  
cv\_mu0$schoolid <- as.numeric(cv\_mu0$schoolid)  
  
sub <- left\_join(sub, null\_mu0, by = 'schoolid')  
sub <- left\_join(sub, int\_mu0, by = 'schoolid')  
sub <- left\_join(sub, nocv\_mu0, by = 'schoolid')  
sub <- left\_join(sub, cv\_mu0, by = 'schoolid')  
  
sub <- sub %>%   
 dplyr::select(-grpvar.x,  
 -grpvar.x.x,  
 -grpvar.y,  
 -grpvar.y.y)

* Model A: Random Intercept model without ses as a predictor (i.e., null model).

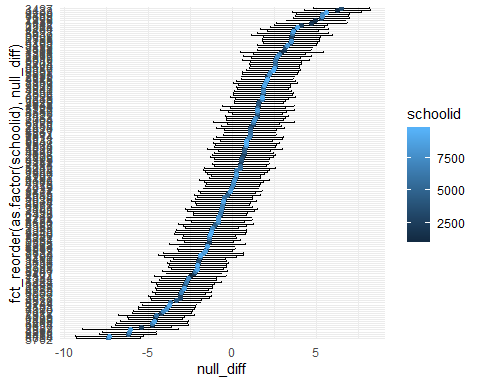
sub %>%   
 ggplot(aes(ses, random\_null)) +  
 geom\_point(color = 'gray70', alpha = .3) +  
 geom\_smooth(se = FALSE, method = 'lm', aes(color = as.factor(schoolid))) +  
 theme\_minimal() +  
 theme(legend.position = 'none')



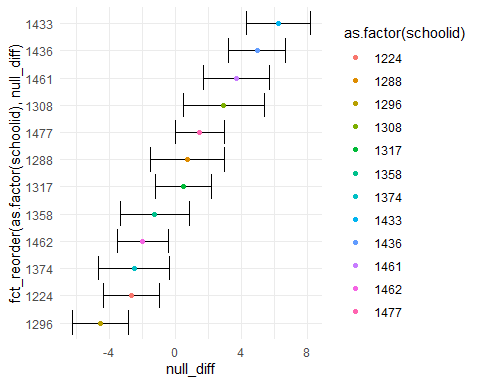
sub %>%   
 filter(schoolid < 1478) %>%   
 ggplot(aes(ses, random\_null)) +  
 geom\_point(color = 'gray70', alpha = .3) +  
 geom\_smooth(se = FALSE, method = 'lm', aes(color = as.factor(schoolid))) +  
 theme\_minimal()



sub %>%   
 ggplot(aes(fct\_reorder(as.factor(schoolid), null\_diff), null\_diff)) +  
 geom\_errorbar(aes(ymin = null\_diff + qnorm(0.025)\*null\_se,  
 ymax = null\_diff + qnorm(0.975)\*null\_se)) +  
 geom\_point(aes(color = schoolid)) +  
 theme\_minimal() +  
 coord\_flip()



sub %>%   
 filter(schoolid < 1478) %>%   
 ggplot(aes(fct\_reorder(as.factor(schoolid), null\_diff), null\_diff)) +  
 geom\_errorbar(aes(ymin = null\_diff + qnorm(0.025)\*null\_se,  
 ymax = null\_diff + qnorm(0.975)\*null\_se)) +  
 geom\_point(aes(color = as.factor(schoolid))) +  
 theme\_minimal() +  
 coord\_flip()

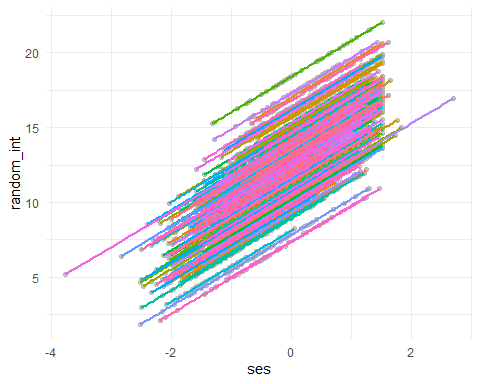


* Model B: Random Intercept model with ses as a fixed effect predictor.

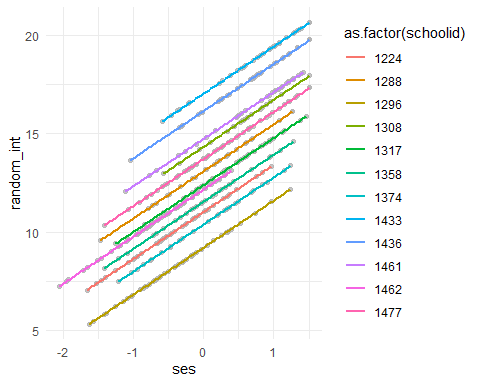
sub %>%   
 colnames()

## [1] "schoolid" "mathach" "ses" "female"   
## [5] "n\_per" "n" "one\_school" "fixed\_value"   
## [9] "random\_null" "random\_int" "random\_nocv" "random\_cv"   
## [13] "null\_term" "null\_diff" "null\_se" "ranint\_term"   
## [17] "ran\_int\_diff" "ran\_int\_se" "nocv\_term" "nocv\_diff"   
## [21] "nocv\_se" "cv\_term" "cv\_diff" "cv\_se"

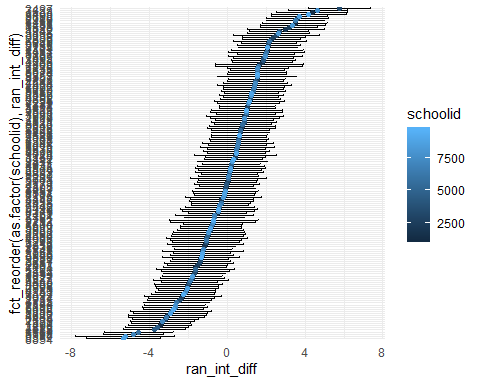
sub %>%   
 ggplot(aes(ses, random\_int)) +  
 geom\_point(color = 'gray70', alpha = .3) +  
 geom\_smooth(se = FALSE, method = 'lm', aes(color = as.factor(schoolid))) +  
 theme\_minimal() +  
 theme(legend.position = 'none')



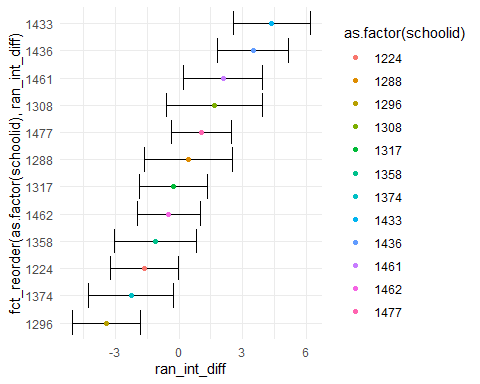
sub %>%   
 filter(schoolid < 1478) %>%   
 ggplot(aes(ses, random\_int)) +  
 geom\_point(color = 'gray70', alpha = .3) +  
 geom\_smooth(se = FALSE, method = 'lm', aes(color = as.factor(schoolid))) +  
 theme\_minimal()



sub %>%   
 ggplot(aes(fct\_reorder(as.factor(schoolid), ran\_int\_diff), ran\_int\_diff)) +  
 geom\_errorbar(aes(ymin = ran\_int\_diff + qnorm(0.025)\*ran\_int\_se,  
 ymax = ran\_int\_diff + qnorm(0.975)\*ran\_int\_se)) +  
 geom\_point(aes(color = schoolid)) +  
 theme\_minimal() +  
 coord\_flip()



sub %>%   
 filter(schoolid < 1478) %>%   
 ggplot(aes(fct\_reorder(as.factor(schoolid), ran\_int\_diff), ran\_int\_diff)) +  
 geom\_errorbar(aes(ymin = ran\_int\_diff + qnorm(0.025)\*ran\_int\_se,  
 ymax = ran\_int\_diff + qnorm(0.975)\*ran\_int\_se)) +  
 geom\_point(aes(color = as.factor(schoolid))) +  
 theme\_minimal() +  
 coord\_flip()

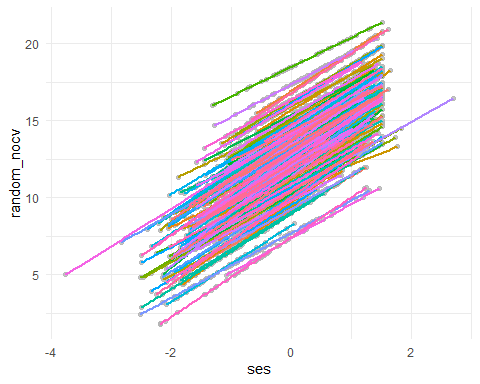


* Model C: Random Slope model with ses as a fixed effect predictor, and treating ses as random slope.

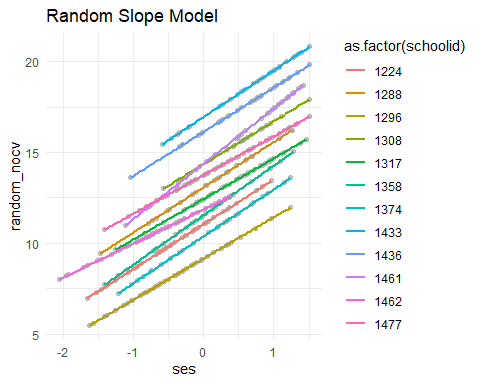
sub %>%   
 colnames()

## [1] "schoolid" "mathach" "ses" "female"   
## [5] "n\_per" "n" "one\_school" "fixed\_value"   
## [9] "random\_null" "random\_int" "random\_nocv" "random\_cv"   
## [13] "null\_term" "null\_diff" "null\_se" "ranint\_term"   
## [17] "ran\_int\_diff" "ran\_int\_se" "nocv\_term" "nocv\_diff"   
## [21] "nocv\_se" "cv\_term" "cv\_diff" "cv\_se"

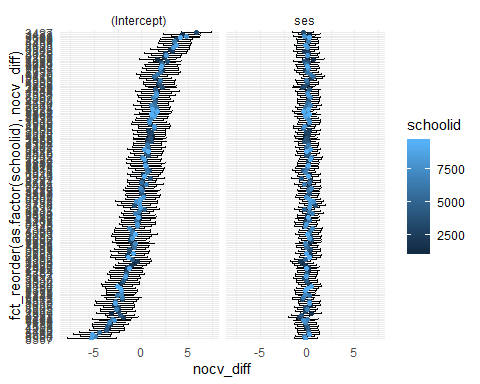
sub %>%   
 ggplot(aes(ses, random\_nocv)) +  
 geom\_point(color = 'gray70', alpha = .3) +  
 geom\_smooth(se = FALSE, method = 'lm', aes(color = as.factor(schoolid))) +  
 theme\_minimal() +  
 theme(legend.position = 'none')



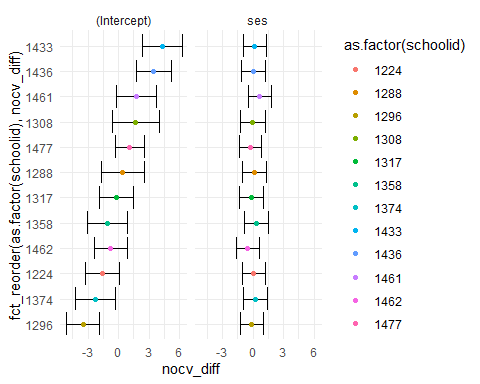
sub %>%   
 filter(schoolid < 1478) %>%   
 ggplot(aes(ses, random\_nocv)) +  
 geom\_point(color = 'gray70', alpha = .3) +  
 geom\_smooth(se = FALSE, method = 'lm', aes(color = as.factor(schoolid))) +  
 theme\_minimal() +  
 labs(title = 'Random Slope Model')



sub %>%   
 ggplot(aes(fct\_reorder(as.factor(schoolid), nocv\_diff), nocv\_diff)) +  
 geom\_errorbar(aes(ymin = nocv\_diff + qnorm(0.025)\*nocv\_se,  
 ymax = nocv\_diff + qnorm(0.975)\*nocv\_se)) +  
 geom\_point(aes(color = schoolid)) +  
 theme\_minimal() +  
 facet\_wrap(~nocv\_term) +  
 coord\_flip()



sub %>%   
 filter(schoolid < 1478) %>%   
 ggplot(aes(fct\_reorder(as.factor(schoolid), nocv\_diff), nocv\_diff)) +  
 geom\_errorbar(aes(ymin = nocv\_diff + qnorm(0.025)\*nocv\_se,  
 ymax = nocv\_diff + qnorm(0.975)\*nocv\_se)) +  
 geom\_point(aes(color = as.factor(schoolid))) +  
 theme\_minimal() +  
 facet\_wrap(~nocv\_term) +  
 coord\_flip()

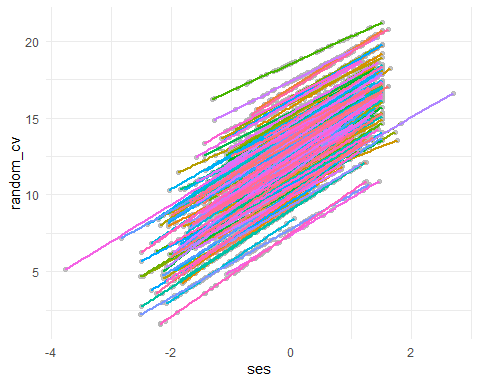


* Model D: Random Slope model with ses as a fixed effect predictor, and treating ses as random slope, and allowing covariance to be unstructured.

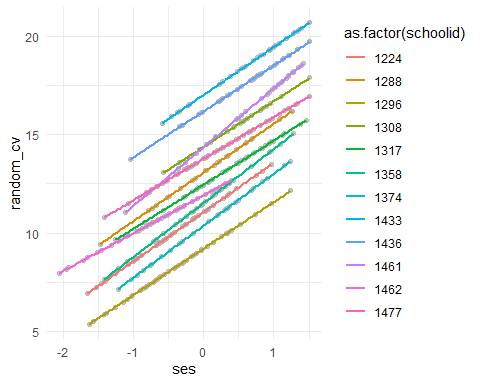
sub %>%   
 colnames()

## [1] "schoolid" "mathach" "ses" "female"   
## [5] "n\_per" "n" "one\_school" "fixed\_value"   
## [9] "random\_null" "random\_int" "random\_nocv" "random\_cv"   
## [13] "null\_term" "null\_diff" "null\_se" "ranint\_term"   
## [17] "ran\_int\_diff" "ran\_int\_se" "nocv\_term" "nocv\_diff"   
## [21] "nocv\_se" "cv\_term" "cv\_diff" "cv\_se"

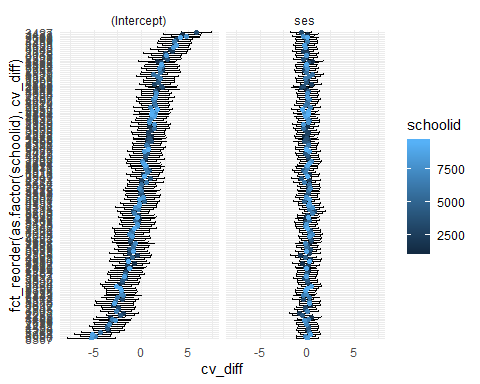
sub %>%   
 ggplot(aes(ses, random\_cv)) +  
 geom\_point(color = 'gray70', alpha = .3) +  
 geom\_smooth(se = FALSE, method = 'lm', aes(color = as.factor(schoolid))) +  
 theme\_minimal() +  
 theme(legend.position = 'none')



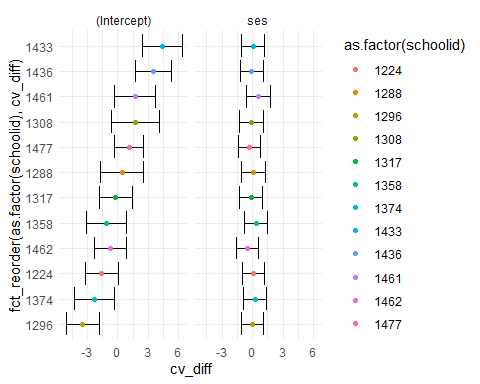
sub %>%   
 filter(schoolid < 1478) %>%   
 ggplot(aes(ses, random\_cv)) +  
 geom\_point(color = 'gray70', alpha = .3) +  
 geom\_smooth(se = FALSE, method = 'lm', aes(color = as.factor(schoolid))) +  
 theme\_minimal()



sub %>%   
 ggplot(aes(fct\_reorder(as.factor(schoolid), cv\_diff), cv\_diff)) +  
 geom\_errorbar(aes(ymin = cv\_diff + qnorm(0.025)\*cv\_se,  
 ymax = cv\_diff + qnorm(0.975)\*cv\_se)) +  
 geom\_point(aes(color = schoolid)) +  
 theme\_minimal() +  
 facet\_wrap(~cv\_term) +  
 coord\_flip()



sub %>%   
 filter(schoolid < 1478) %>%   
 ggplot(aes(fct\_reorder(as.factor(schoolid), cv\_diff), cv\_diff)) +  
 geom\_errorbar(aes(ymin = cv\_diff + qnorm(0.025)\*cv\_se,  
 ymax = cv\_diff + qnorm(0.975)\*cv\_se)) +  
 geom\_point(aes(color = as.factor(schoolid))) +  
 theme\_minimal() +  
 facet\_wrap(~cv\_term) +  
 coord\_flip()



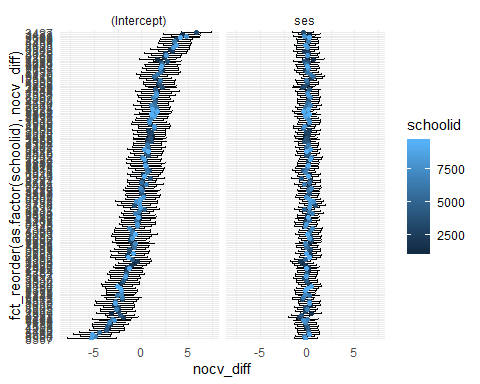
In the two-way plots, have ses on the x-axis versus predicted mathach scores, and have lines drawn by school.

* Hint: there are so many schools in the sample that it may make it difficult to see what is going on if we include all of them. Instead try adding the following to the twoway command in order to only draw lines for schools with id number smaller than 1478:

twoway connected y\_hat xvar if schoolid <1478

1. For whichever model you decided was the “best fit” in Question 4 – obtain estimates of appropriate school-level residuals (mu 0j, mu 1j) for each school in the sample and the SE of these estimates. You do not need to actually print out the values, but please show the code that you use. Use these estimates to generate caterpillar plots. (If you found the “best” model to be the random intercepts one you will not need to include a caterpillar plot for �$#. If random slopes was the best fit, then do include a random slopes plot.)

sub %>%   
 ggplot(aes(fct\_reorder(as.factor(schoolid), nocv\_diff), nocv\_diff)) +  
 geom\_errorbar(aes(ymin = nocv\_diff + qnorm(0.025)\*nocv\_se,  
 ymax = nocv\_diff + qnorm(0.975)\*nocv\_se)) +  
 geom\_point(aes(color = schoolid)) +  
 theme\_minimal() +  
 facet\_wrap(~nocv\_term) +  
 coord\_flip()



sub %>%   
 filter(schoolid < 1478) %>%   
 ggplot(aes(fct\_reorder(as.factor(schoolid), nocv\_diff), nocv\_diff)) +  
 geom\_errorbar(aes(ymin = nocv\_diff + qnorm(0.025)\*nocv\_se,  
 ymax = nocv\_diff + qnorm(0.975)\*nocv\_se)) +  
 geom\_point(aes(color = as.factor(schoolid))) +  
 theme\_minimal() +  
 facet\_wrap(~nocv\_term) +  
 coord\_flip()

