Problem Set 5

Janette Avelar

5/30/2022

Data Context

For these questions we'll be using data from the The ATLAS program is a team-based intervention designed to decrease steroid use among high school football players. Thirty-one teams from different high schools in the northwest U.S. were randomly assigned to intervention or control conditions. The seven-week intervention consisted of both classroom and weight training sessions. Although the ultimate object of the ATLAS program was to reduce steroid use, a more immediate aim was to increase the adoption of healthy alternatives to steroids such as strength training. Therefore, levels of strength training self-efficacy were measured prior to and immediately following the intervention period.Adolescents Training and Learning to Avoid Steroids (ATLAS) project.

The dataset contains the following variables:
`school` - school ID
`grade` - student grade (9-12)
`stu_id` - individual student ID
`intervention` - indicates whether school was control (*0*) or intervention (*1*)
`stse0` - self-reported strength training self-efficacy (*pretest*)
`stse1` - self-reported strength training self-efficacy (*posttest*)
`use0` - individual steroid use either yes (*1*) or no (*0*) (*pretest*)
`coachtol0` - perception of coach tolerance of steroid use (*pretest*)
`reasons0` - number of reasons for using steroids (*pretest*)
`se0` - self-reported self esteem (*pretest*)
`se1` - self-reported self esteem (*posttest*)

Question 1

Use R to estimate a disaggregated model predicting post-test strength training self-efficacy from the pretest measure of this variable.

```r
dagg <- lm(stse1 ~ stse0, data = ind_dat)
summary(dagg)
```

```
##
## Call:
## lm(formula = stse1 ~ stse0, data = ind_dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.3111 -0.5107  0.2228  0.6889  2.7905
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.85920    0.15953   24.19   <2e-16 ***
## stse0        0.35027    0.02716   12.89   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 1.015 on 1199 degrees of freedom
##   (25 observations deleted due to missingness)
## Multiple R-squared:  0.1218, Adjusted R-squared:  0.1211
## F-statistic: 166.3 on 1 and 1199 DF,  p-value: < 2.2e-16
```

What do the results of this analysis suggest?

The results of our disaggregated model suggest that a pretest score for self-reported strength training self-efficacy of 0 results in a posttest score of 3.86 with an additional 0.35 point increase for each subsequent unit increase in self-reported self-efficacy at pretest.

What is the problem with conducting the analysis in this manner?
Our degrees of freedom is 1199 which does not accurately reflect the degrees of freedom in the model we ran because it is ignoring the variance between our control and intervention groups, thus greatly inflating our chances of committing a Type I error.

Question 2

Estimate an aggregate model predicting mean posttest strength training self-efficacy from mean pretest strength training self-efficacy.

```
#create mean scores
agg_dat <- ind_dat %>%
  group_by(schoolid) %>%
  summarize(stse0_mean = mean(stse0, na.rm = TRUE),
         stse1_mean = mean(stse1, na.rm = TRUE))
#run model with mean scores
agg <- lm(stse1_mean ~ stse0_mean, data = agg_dat)
summary(agg)
```

```
##
## Call:
## lm(formula = stse1_mean ~ stse0_mean, data = agg_dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.65915 -0.20840  0.05652  0.24488  0.54935
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.2963     0.9671   3.409  0.00194 **
## stse0_mean    0.4500     0.1686   2.669  0.01234 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
```

```
## Residual standard error: 0.3566 on 29 degrees of freedom
## Multiple R-squared:  0.1972, Adjusted R-squared:  0.1695 
## F-statistic: 7.122 on 1 and 29 DF,  p-value: 0.01234
```

What do the results of this analysis suggest?
The results of our aggregated model suggest that a pretest score for self-reported strength training self-efficacy of 0 results in a mean posttest score of 3.3 with an additional 0.45 point increase for each subsequent unit increase in mean self-reported self-efficacy at pretest.

Is this analysis consistent with the disaggregated analysis?
The results of our aggregated and disaggregated models are similar, but not entirely consistent, with the biggest difference reflected in our degrees of freedom (1199 vs. 29).

Question 3

Conduct an OLS regression within each school predicting posttest strength training self-efficacy from pretest strength training. Plot the lines on a single set of axes.

```
options(scipen=999)
#factor school ID
ind_dat <- ind_dat %>%
  mutate(schoolid = factor(schoolid))
# run model looking at school intercepts
ols_int <- lm(stse1 ~ stse0 + schoolid, data = ind_dat)
summary(ols_int)
```

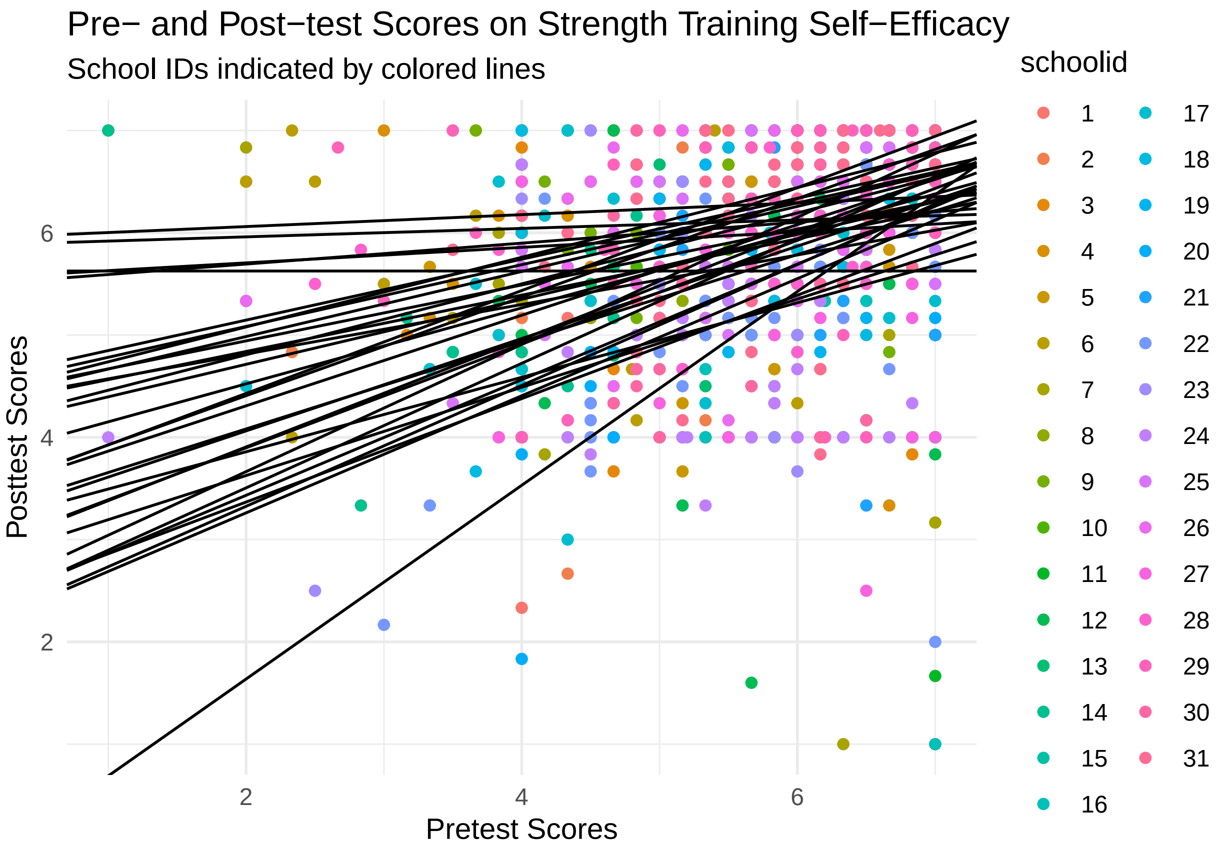
```
## 
## Call:
## lm(formula = stse1 ~ stse0 + schoolid, data = ind_dat)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -5.1353 -0.5368  0.1941  0.6299  2.7151 
## 
## Coefficients:
##             Estimate Std. Error t value             Pr(>|t|)    
## (Intercept)  3.85756    0.23948  16.108 < 0.0000000000000002 ***
## stse0        0.34860    0.02725  12.792 < 0.0000000000000002 ***
## schoolid2    0.36947    0.24106   1.533              0.12562    
## schoolid3   -0.16250    0.20971  -0.775              0.43858    
## schoolid4    0.23755    0.24122   0.985              0.32493    
## schoolid5    0.12621    0.21330   0.592              0.55416    
## schoolid6    0.48539    0.23180   2.094              0.03648 *  
## schoolid7   -0.19671    0.22435  -0.877              0.38077    
## schoolid8    0.38805    0.25569   1.518              0.12937    
## schoolid9    0.35380    0.23531   1.504              0.13297    
## schoolid10   0.46965    0.24221   1.939              0.05274 .  
## schoolid11  -0.64683    0.28174  -2.296              0.02186 *  
## schoolid12  -0.08597    0.21266  -0.404              0.68611    
## schoolid13   0.19731    0.24040   0.821              0.41196    
## schoolid14   0.07878    0.29366   0.268              0.78855    
## schoolid15   0.31177    0.25314   1.232              0.21834    
## schoolid16  -0.58046    0.24620  -2.358              0.01855 *
## schoolid17   0.03865    0.22203   0.174              0.86182
## schoolid18   0.07236    0.20966   0.345              0.73007
## schoolid19   0.05402    0.26166   0.206              0.83648
## schoolid20  -0.74253    0.27230  -2.727              0.00649 **
## schoolid21   0.07776    0.22971   0.338              0.73505
## schoolid22  -0.47426    0.20901  -2.269              0.02344 *
## schoolid23  -0.20163    0.28102  -0.717              0.47321
## schoolid24  -0.62129    0.20999  -2.959              0.00315 **
## schoolid25   0.12629    0.25124   0.503              0.61529
## schoolid26   0.49604    0.23583   2.103              0.03565 *
## schoolid27  -0.25815    0.21925  -1.177              0.23926
## schoolid28   0.30302    0.22841   1.327              0.18488
## schoolid29   0.40093    0.22221   1.804              0.07144 .
## schoolid30  -0.32973    0.28561  -1.155              0.24853
## schoolid31   0.24467    0.21856   1.119              0.26317
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9683 on 1169 degrees of freedom
##   (25 observations deleted due to missingness)
## Multiple R-squared:  0.221,  Adjusted R-squared:  0.2003
## F-statistic:  10.7 on 31 and 1169 DF,  p-value: < 0.0000000000000022
```

```
# run model looking at school slopes
ols_slopes <- lm(stse1 ~ stse0*schoolid, data = ind_dat)
summary(ols_slopes)
```

```
##
## Call:
## lm(formula = stse1 ~ stse0 * schoolid, data = ind_dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.0250 -0.5057  0.1973  0.6151  2.6041
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.882327   1.134612   2.540  0.01121 *
## stse0             0.507282   0.182539   2.779  0.00554 **
## schoolid2         1.598075   1.325142   1.206  0.22808
## schoolid3         1.447316   1.358725   1.065  0.28701
## schoolid4         2.671350   1.417582   1.884  0.05976 .
## schoolid5         0.539508   1.470692   0.367  0.71381
## schoolid6         2.994973   1.252621   2.391  0.01697 *
## schoolid7         2.743462   1.367295   2.006  0.04504 *
## schoolid8         0.561215   2.018995   0.278  0.78109
## schoolid9         3.061622   1.514175   2.022  0.04341 *
## schoolid10        0.541095   1.573394   0.344  0.73098
## schoolid11        0.247525   1.461980   0.169  0.86558
## schoolid12        0.277280   1.402144   0.198  0.84327
## schoolid13        2.606981   1.646518   1.583  0.11362
## schoolid14        1.195759   1.357887   0.881  0.37872
## schoolid15       -0.461432   2.213528  -0.208  0.83491
## schoolid16       -0.526589   1.496541  -0.352  0.72500
## schoolid17        1.406223   1.285309   1.094  0.27415
## schoolid18       -0.027227   1.360875  -0.020  0.98404
## schoolid19        0.897765   1.939866   0.463  0.64360
## schoolid20       -3.139963   1.583575  -1.983  0.04763 *
## schoolid21       -0.594369   1.804415  -0.329  0.74192
## schoolid22       -0.765969   1.332904  -0.575  0.56563
## schoolid23        0.347190   1.643312   0.211  0.83271
## schoolid24       -0.119038   1.312934  -0.091  0.92777
## schoolid25        1.513712   1.532474   0.988  0.32348
## schoolid26        1.515254   1.341617   1.129  0.25896
## schoolid27       -0.738998   1.405399  -0.526  0.59911
## schoolid28        1.466696   1.446591   1.014  0.31085
## schoolid29        1.668281   1.337133   1.248  0.21241
## schoolid30       -0.579799   2.068783  -0.280  0.77933
## schoolid31        1.231539   1.469359   0.838  0.40212
## stse0:schoolid2  -0.205437   0.219836  -0.935  0.35024
## stse0:schoolid3  -0.265092   0.220517  -1.202  0.22956
## stse0:schoolid4  -0.431388   0.239980  -1.798  0.07250 .
## stse0:schoolid5  -0.064349   0.239891  -0.268  0.78856
## stse0:schoolid6  -0.466014   0.209042  -2.229  0.02599 *
## stse0:schoolid7  -0.507258   0.225824  -2.246  0.02488 *
## stse0:schoolid8  -0.025572   0.330391  -0.077  0.93832
## stse0:schoolid9  -0.449015   0.246643  -1.821  0.06894 .
## stse0:schoolid10 -0.004435   0.259020  -0.017  0.98634
## stse0:schoolid11 -0.143002   0.251891  -0.568  0.57034
## stse0:schoolid12 -0.055715   0.228233  -0.244  0.80719
## stse0:schoolid13 -0.405013   0.272912  -1.484  0.13807
## stse0:schoolid14 -0.189510   0.238913  -0.793  0.42782
## stse0:schoolid15  0.114406   0.347293   0.329  0.74190
## stse0:schoolid16 -0.001632   0.245146  -0.007  0.99469
## stse0:schoolid17 -0.231911   0.212970  -1.089  0.27641
## stse0:schoolid18  0.018091   0.219554   0.082  0.93434
## stse0:schoolid19 -0.135980   0.325550  -0.418  0.67625
## stse0:schoolid20  0.439847   0.265278   1.658  0.09758 .
## stse0:schoolid21  0.101598   0.286161   0.355  0.72263
## stse0:schoolid22  0.065195   0.219460   0.297  0.76647
## stse0:schoolid23 -0.081028   0.280315  -0.289  0.77259
## stse0:schoolid24 -0.075839   0.215100  -0.353  0.72447
## stse0:schoolid25 -0.233951   0.260014  -0.900  0.36844
## stse0:schoolid26 -0.166613   0.221702  -0.752  0.45250
## stse0:schoolid27  0.083728   0.227756   0.368  0.71322
## stse0:schoolid28 -0.189236   0.232164  -0.815  0.41519
## stse0:schoolid29 -0.209588   0.218923  -0.957  0.33859
## stse0:schoolid30  0.057901   0.353863   0.164  0.87006
## stse0:schoolid31 -0.160682   0.241930  -0.664  0.50672
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.9599 on 1139 degrees of freedom
##   (25 observations deleted due to missingness)
## Multiple R-squared:  0.2541, Adjusted R-squared:  0.2142
## F-statistic: 6.362 on 61 and 1139 DF,  p-value: < 0.00000000000000022
```

```
#extract intercepts and slopes
ints_slopes <- ind_dat %>%
  split(.$schoolid) %>%
  map(~ lm(stse1 ~ stse0, data = .)) %>%
  map_dfc("coefficients") %>%
  cbind(term = c("intercept", "b_strength"), .) %>%
  gather(schoolid, estimate, -term) %>%
  spread(term, estimate)
#plot
ggplot(data = ind_dat, aes(stse0, stse1, color = schoolid)) +
  geom_point() +
  geom_abline(slope = ints_slopes$b_strength, intercept = ints_slopes$intercept) +
  theme_minimal() +
  labs(title = "Pre- and Post-test Scores on Strength Training Self-Efficacy",
       subtitle = "School IDs indicated by colored lines",
       x = "Pretest Scores",
       y = "Posttest Scores")
```

```
## Warning: Removed 25 rows containing missing values (geom_point).
```



Does it look like the ANCOVA assumption (homogeneity of regression) would be met?
No, from the plot there appears to be distinct clustering with a set of schools with more stagnant scores (and one which appears to go down), and varying degrees of slope severity.

Question 4

Finish the intercepts- and slopes-as-outcomes analysis: use the group-level intervention variable to (separately) predict intercepts and slopes.

```
#first left-join ints_slopes with original dataset to ensure intervention variable is in our dataset
ols_dat <- ind_dat %>%
  select(schoolid, intervention) %>%
  mutate(intervention = factor(intervention, levels = c(0, 1), labels = c("control", "intervention"))) %
  left_join(ints_slopes, by = "schoolid")
#use int_slopes to regress intercepts on predictor
ints_model <- lm(intercept ~ intervention, data = ols_dat)
summary(ints_model)
```

```
##
## Call:
## lm(formula = intercept ~ intervention, data = ols_dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.11987 -0.57303 -0.09895  0.55960  1.68837
##
## Coefficients:
##                          Estimate Std. Error t value            Pr(>|t|)
## (Intercept)               2.86224    0.03458   82.77 <0.0000000000000002 ***
## interventionintervention  1.82466    0.05249   34.76 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9109 on 1224 degrees of freedom
## Multiple R-squared:  0.4968, Adjusted R-squared:  0.4964
## F-statistic:  1208 on 1 and 1224 DF,  p-value: < 0.00000000000000022
```

```
#now use int_slopes to regress slopes on predictor
slopes_model <- lm(b_strength ~ intervention, data = ols_dat)
summary(slopes_model)
```

```
##
## Call:
## lm(formula = b_strength ~ intervention, data = ols_dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.24526 -0.12318  0.03217  0.09951  0.45967
##
## Coefficients:
##                           Estimate Std. Error t value            Pr(>|t|)
## (Intercept)               0.487456   0.005643   86.38 <0.0000000000000002 ***
## interventionintervention -0.246296   0.008567  -28.75 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1487 on 1224 degrees of freedom
## Multiple R-squared:  0.4031, Adjusted R-squared:  0.4026
## F-statistic: 826.5 on 1 and 1224 DF,  p-value: < 0.00000000000000022
```

Summarize the results of the analysis:
Our model predicts a posttest score of 2.86 for self-reported strengths training self-efficacy for our control group with a pretest score of 0, which is predicted to increase by 0.49 points for each increase in pretest score. For our groups that received the intervention, a posttest score of 4.69 ($2.86224intercept + 1.82466slope$) is expected given the same pretest score with a decrease of -0.25 for every unit increase in pretest score.

Question 5

Compute the ICC of posttest strength training self-efficacy.

```
#first we need to run our null model
rcr_0 <- lmer(stse1 ~ 1 + (1 | schoolid),
              data = ind_dat)
summary(rcr_0)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: stse1 ~ 1 + (1 | schoolid)
##    Data: ind_dat
##
## REML criterion at convergence: 3575.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.6419 -0.6038  0.2387  0.7667  1.6968
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  schoolid (Intercept) 0.1147   0.3386
##  Residual             1.0636   1.0313
## Number of obs: 1215, groups:  schoolid, 31
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  5.87897    0.06845   85.88
```

```
#now estimate ICC
#between group variance / total variance
0.1147/(0.1147+1.0636)
```

```
## [1] 0.09734363
```

In your estimation, is an ICC of this size likely to be problematic? Is most of the variance between or within groups?
Our ICC is about 0.1, meaning only about 10% of the variance is due to differences between groups. Given that we ran an experimental design, this doesn't bode well for our findings as it indicates that the majority of our variance is within groups, occurring at the individual level rather than due to our intervention.

Question 6

Write down the first- and second-level equations for a random coefficient model in which posttest strength training self-efficacy is predicted by pretest strength training self-efficacy, intervention, and their interaction.

Level 1:

$$L1 : Y_{ij} = b_{0j} + e_{ij}$$

where b_{0j} = our intercept at 0 pretest score and e_{ij} is our individual-level error term.

Level 2:

$$L2 : b_{0j} = g_{00} + u_{0j}$$

where g_{00} = our intercept for control group and u_{0j} is our group-level error term.

Full model:

$$Y_{ij} = g_{00} + g_{01}pre_j + g_{10}control_{ij} + g_{11}pre_{ij} * control_j + (u_{0j} + u_{1j}pre_{ij} + e_{ij})$$

What is the meaning of each parameter? g_{00} = expected Y-intercept with a pretest score of 0 for the control group

g_{01} = expected difference in intercept given a change in pretest score

g_{10} = expected slope given assignment to either control (0) or experimental (1) group

g_{11} = effect of assignment on slope

u_{0j} = random error from predicting our intercepts

u_{1j} = random error from predicting our slopes

e_{ij} = variance at the individual level

Question 7

Examine the model from Problem 6. Suppose the slope of pretest strength training self-efficacy was found not to vary significantly across schools. **How would this change model? How would it change your interpretation of the data?**

If the slope of pretest strength training self-efficacy did not vary significantly across schools I would not continue analyzing using MLM, but would rather focus on within-group differences because it would not be important to find a predictor that explained the variance in slopes. I'd turn to individual-level predictors that might explain the variance within schools and would interpret the data in light of individual-level differences and predictors that may or may not hold across schools.

Question 8

Run the model from Problem 6 in R using `lmer()`.

```
#factor intervention in ind_dat
ind_dat <- ind_dat %>%
  mutate(intervention = factor(intervention, levels = c(0, 1), labels = c("control", "intervention")))
#run model
rcr_1 <-  lmer(stse1 ~ 1 + stse0*intervention + (1 + stse0 | schoolid),
              data = ind_dat)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.653934 (tol = 0.002, component 1)
```

```
summary(rcr_1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: stse1 ~ 1 + stse0 * intervention + (1 + stse0 | schoolid)
##    Data: ind_dat
##
## REML criterion at convergence: 3356
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -5.4912 -0.5413  0.2192  0.6468  2.3216
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  schoolid (Intercept) 0.131861 0.36313
##           stse0       0.001539 0.03923  -0.82
##  Residual             0.920001 0.95917
## Number of obs: 1201, groups:  schoolid, 31
##
## Fixed effects:
##                                Estimate Std. Error t value
## (Intercept)                     2.84927    0.24435  11.661
## stse0                           0.48983    0.03918  12.504
## interventionintervention        1.89635    0.34007   5.576
## stse0:interventionintervention -0.25631    0.05544  -4.624
##
## Correlation of Fixed Effects:
##             (Intr) stse0  intrvn
## stse0       -0.961
## intrvntnntr -0.719  0.691
## sts0:ntrvnt  0.679 -0.707 -0.957
## optimizer (nloptwrap) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 0.653934 (tol = 0.002, component 1)
```

What are the parameters?

How do the results compare to the results from the intercepts- and slopes-as-outcomes model?

I wasn't able to answer this question because my model wouldn't run. I get the following warning message regarding convergence failure:

```
Warning message:  In checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv,  :
Model failed to converge with max|grad| = 0.653934 (tol = 0.002, component 1)
```

I tried to do some troubleshooting on my own, but as it was not required for the assignment and I'm short on time this week and unable to go to office hours I went ahead and turned in. But if you have any suggestions for why this wasn't working I'd love some feedback!