Stat 414 - Day 28
Case Study Part II Solutions

```
musicians = read.delim("https://www.rossmanchance.com/stat414/data/musicians.
txt" , "\t", header=TRUE)
model0 = lmer(na ~ 1 + (1|id), data=musicians)

musicians$performlarge = as.numeric(musicians$perform_type1 == "LargeEnsemble
")
summary(model1 <- lmer(na ~ performlarge + (performlarge | subjnum), data = m
usicians))

## Linear mixed model fit by REML ['lmerMod']
## Formula: na ~ performlarge + (performlarge | subjnum)
##    Data: musicians
##
## REML criterion at convergence: 2994
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.9892 -0.6827 -0.1977  0.4839  4.1398
##
## Random effects:
##  Groups   Name         Variance Std.Dev. Corr
##  subjnum  (Intercept)   6.3330  2.5165
##           performlarge  0.7429  0.8619   -0.76
##  Residual              21.7712  4.6660
## Number of obs: 497, groups:  subjnum, 37
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)   16.7297     0.4908   34.09
## performlarge  -1.6762     0.5425   -3.09
##
## Correlation of Fixed Effects:
##             (Intr)
## performlarg -0.453

musicians$orchtype = (musicians$instrument1 == "orchestralinstrument")
model2 =lmer(na ~ performlarge*orchtype + (performlarge | subjnum), data = mu
sicians)
summary(model2, corr=FALSE)

## Linear mixed model fit by REML ['lmerMod']
## Formula: na ~ performlarge * orchtype + (performlarge | subjnum)
##    Data: musicians
##
## REML criterion at convergence: 2987
##
## Scaled residuals:
```

```
##     Min      1Q  Median      3Q     Max 
## -1.9404 -0.6625 -0.1771  0.4796  4.1860 
## 
## Random effects:
##  Groups   Name         Variance Std.Dev. Corr 
##  subjnum  (Intercept)   5.655   2.3781        
##           performlarge  0.452   0.6723   -0.63
##  Residual              21.807   4.6698        
## Number of obs: 497, groups:  subjnum, 37
## 
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)                15.9297     0.6415  24.833
## performlarge               -0.9106     0.8452  -1.077
## orchtypeTRUE                1.6926     0.9452   1.791
## performlarge:orchtypeTRUE  -1.4239     1.0992  -1.295
```

Part II: We saw in some of the early data exploration, evidence that subjects with higher baseline levels of negative emotionality tend to have higher performance anxiety levels prior to performances.

Add mpqnem to the model, but first center it. Also include the cross-level interaction to look at how mpqnem explains variation in both the intercepts and the slopes.

```
musicians$mpqnem.c = musicians$mpqnem - mean(musicians$mpqnem)
performlargeF = as.factor(musicians$performlarge) #helps with the effects plo
t
summary(model3 <- lmer(na ~ performlargeF*orchtype + performlargeF*mpqnem.c +
(performlargeF | subjnum), data = musicians), corr = FALSE)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: 
## na ~ performlargeF * orchtype + performlargeF * mpqnem.c + (performlargeF 
|
##     subjnum)
##    Data: musicians
## 
## REML criterion at convergence: 2982.1
## 
## Scaled residuals: 
##     Min      1Q  Median      3Q     Max 
## -2.0544 -0.6364 -0.1584  0.4826  4.0530 
## 
## Random effects:
##  Groups   Name           Variance Std.Dev. Corr 
##  subjnum  (Intercept)     3.2856  1.813         
##           performlargeF1  0.5565  0.746    -0.38
##  Residual                21.8114  4.670         
## Number of obs: 497, groups:  subjnum, 37
## 
```

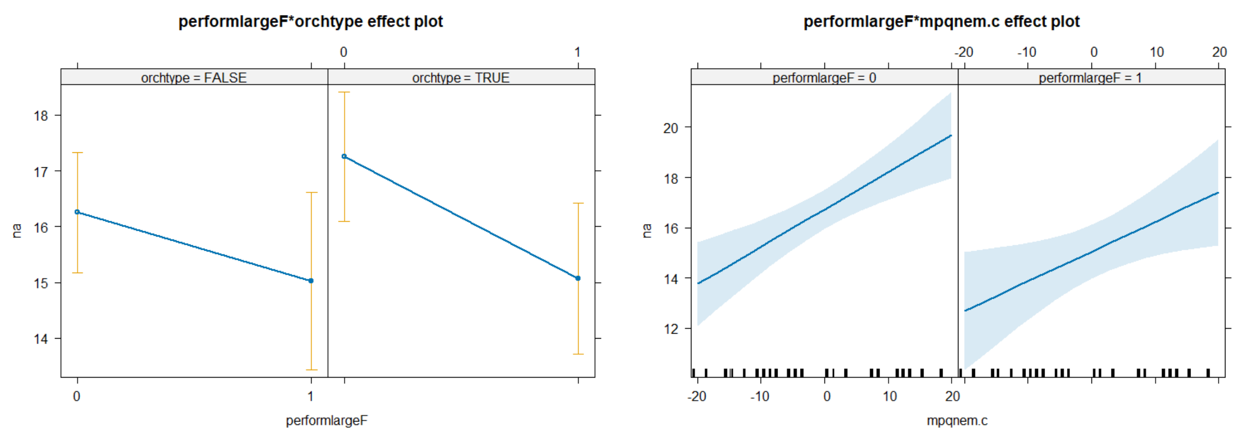
```
## Fixed effects:
##                              Estimate Std. Error t value
## (Intercept)                  16.25679    0.54756  29.689
## performlargeF1               -1.23484    0.84320  -1.464
## orchtypeTRUE                  1.00069    0.81713   1.225
## mpqnem.c                      0.14823    0.03808   3.893
## performlargeF1:orchtypeTRUE -0.94927    1.10620  -0.858
## performlargeF1:mpqnem.c     -0.03018    0.05246  -0.575
```

```
library(effects)
```

```
## Loading required package: carData
```

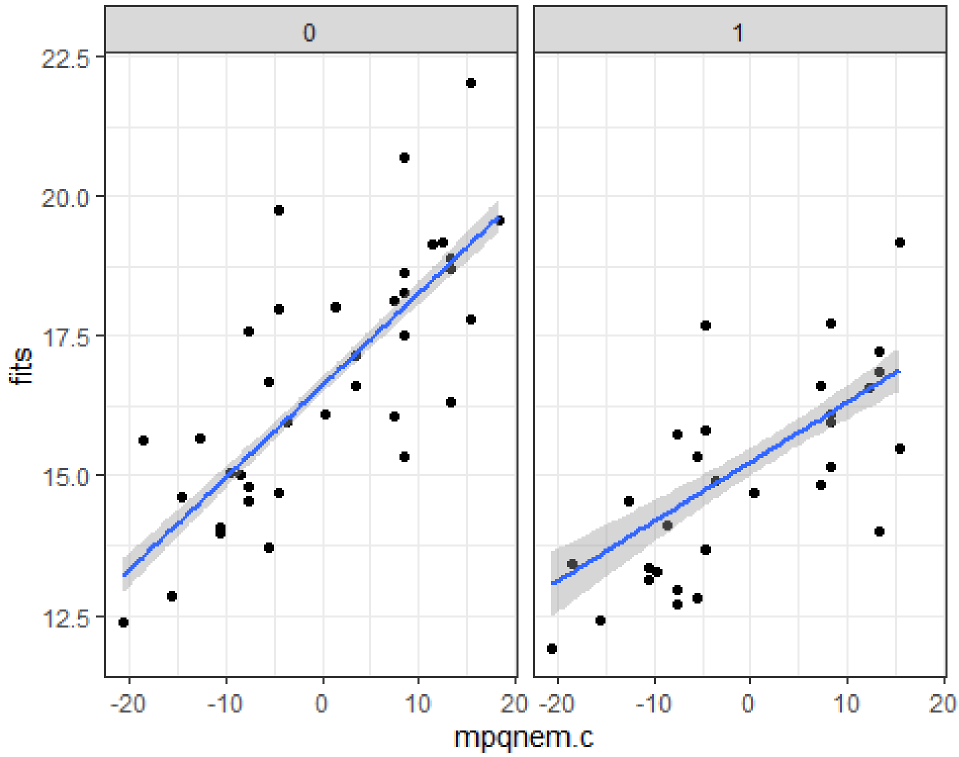
```
## lattice theme set by effectsTheme()
## See ?effectsTheme for details.
```

```
plot(allEffects(model3))
```




```
fits = fitted.values(model3)
ggplot(musicians, aes(y = fits, x= mpqnem.c)) +
  facet_wrap(~performlarge) +
  geom_point() + geom_smooth(method="lm") +
  theme_bw()
```

```
## `geom_smooth()` using formula 'y ~ x'
```

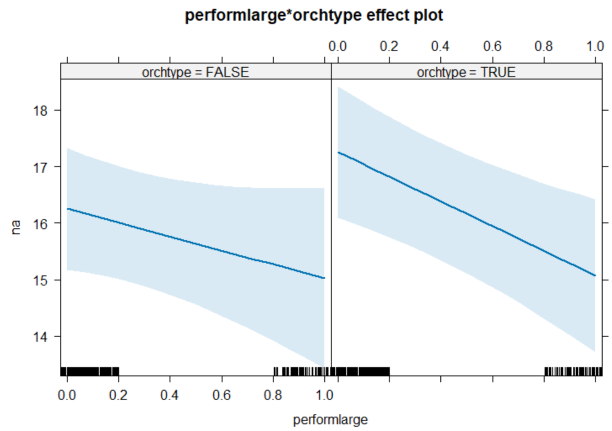


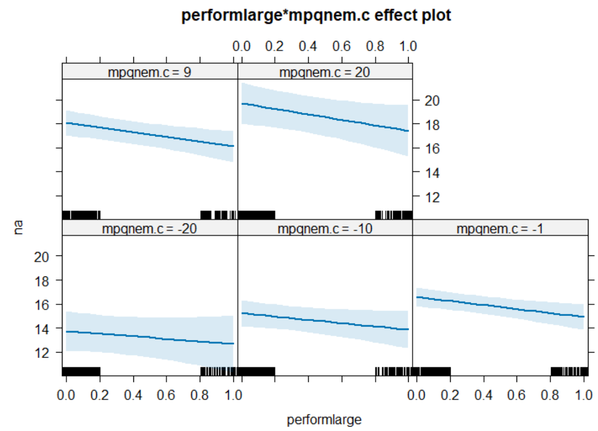
(a) What do you learn about the suggested association between mpqnem and na?

For solo and small ensemble performances (performlarge = 0), on average na increase with larger baseline levels of stress reaction, alienation, and aggression (as measured by the MPQ negative emotionality scale) (slope = 0.148, t = 3.893). For large ensemble performances, the effect of mpqnem on na is smaller (0.148 - 0.03), though the difference in these effects is not statistically significant t = -.575).

For fun, what happens if we don't convert our binary variable into a factor and try to explore the interaction between two quantitative variables.

```
model3b <- lmer(na ~ performlarge*orchtype + performlarge*mpqnem.c + (perform
large | subjnum), data = musicians)
plot(allEffects(model3b))
```





(b) Now how do you describe the new interaction between mpqnem and performance size?

R automatically finds 5 mpqnem values (throughout the range of values in the dataset) and plots the estimated slope between na and performlarge for those 5 values. We can see that for smaller and smaller mpqnem values (moving from top row to bottom row, the centered mpqnem value getting more negative/further below average), the rate of decrease in na from small to large performances also decreases (flattens). In other words, those who have lower mpqnem tend to have lower changes in na between large and other performance types.

Recapping, subjects with higher baseline levels of mpqnem had significantly higher levels of performance anxiety before solos and small ensembles (the slope of mpqnem which you can think of as changing the intercept) and they also had somewhat greater differences (bigger drops) between large ensembles and other performance types (the interaction), controlling for instrument (t = -0.575), although this interaction was not statistically significant.

Compare model 2 from last time to model 3b (so both using 0/1 for performance)

```
texreg::screenreg(list(model2, model3b), digits = 3, single.row = TRUE, stars
= 0, custom.model.names = c("no mpqnem", "with mpqnem"), custom.note = "")

##
## ======================================================================
=
##                                      no mpqnem           with mpqnem
## ---------------------------------------------------------------------
## (Intercept)                        15.930 (0.641)     16.257 (0.548)
## performlarge                       -0.911 (0.845)     -1.235 (0.843)
## orchtypeTRUE                        1.693 (0.945)      1.001 (0.817)
## performlarge:orchtypeTRUE          -1.424 (1.099)     -0.949 (1.106)
## mpqnem.c                                               0.148 (0.038)
## performlarge:mpqnem.c                                 -0.030 (0.052)
## ---------------------------------------------------------------------
## AIC                                  3002.981           3002.108
## BIC                                  3036.650           3044.194
## Log Likelihood                      -1493.490          -1491.054
```

```
## Num. obs.                                     497                 497
## Num. groups: subjnum                           37                  37
## Var: subjnum (Intercept)                        5.655               3.286
## Var: subjnum performlarge                       0.452               0.557
## Cov: subjnum (Intercept) performlarge          -1.015              -0.512
## Var: Residual                                  21.807              21.811
## ============================================================================
=
```

(c) How has the model changed from Model 2 (for the common parameters)? Why?

The directions of the effects of instrument and performance type are consistent, but the effect sizes and levels of significance are reduced (e.g., 1.69 to 1.00) because of the relative importance of the negative emotionality term. (So after adjusting for mpqnem, the other variables have less to tell us.) Interpretations will also change slightly to acknowledge that we have controlled for a covariate. Also keep in mind that when interpreting the coefficient of this variable, we won't be talking about changing groups, but will be back to talking about a "1-unit increase in mpqnem."

(d) Interpret the intercept in context.

The estimated mean performance anxiety for solos and small ensembles (performlarge=0) is 16.26 for keyboard players and vocalists (orchtype=0) with an average level of negative emotionality at baseline (mpqnem=31.63).

(e) Interpret the coefficient of the large ensemble (performance) variable.

I want to interpret the coefficient of performlarge, but it is involved in two interactions, so to make them both "go away," I need to "zero out" the other variable. So for keyboard players and vocalists (orchtype=0) with an average level of baseline negative emotionality levels (mpqnem=31.63), the estimated mean decrease in anxiety level is 1.235 points before large ensemble performances compared to smaller performance types.

(f) Interpret the coefficient of the interaction between mpqnem and the large ensemble variable. Try to be more specific (numbers) this time (not just direction).

We still have orchestratype in the model, so we need to hold it constant, then we can say that the slope of mpqnem for smaller performances is 0.148, but is 0.148 - .030 for larger performances. So .030 is the estimated decrease in the mpqnem effect for larger performances. The baseline emotionality doesn't matter as much (though this interaction coefficient is not significant) for larger performances.

And what if we hadn't centered mpqnem?

```
model3c <- lmer(na ~ performlarge*orchtype + performlarge*mpqnem + (performla
rge | subjnum), data = musicians)
texreg::screenreg(list(model3b, model3c), digits = 3, single.row = TRUE, star
s = 0,
custom.model.names = c("mpqnem centered", "mpqnem not centered"), custom.note
= "")
```

```
##
## ==========================================================================
##                                         mpqnem centered    mpqnem not centered
## --------------------------------------------------------------------------
## (Intercept)                               16.257 (0.548)     11.568 (1.221)
## performlarge                              -1.235 (0.843)     -0.280 (1.834)
## orchtypeTRUE                               1.001 (0.817)      1.001 (0.817)
## mpqnem.c                                   0.148 (0.038)
## performlarge:orchtypeTRUE                 -0.949 (1.106)     -0.949 (1.106)
## performlarge:mpqnem.c                     -0.030 (0.052)
## mpqnem                                                        0.148 (0.038)
## performlarge:mpqnem                                          -0.030 (0.052)
## --------------------------------------------------------------------------
## AIC                                       3002.108          3002.108
## BIC                                       3044.194          3044.194
## Log Likelihood                           -1491.054         -1491.054
## Num. obs.                                  497               497
## Num. groups: subjnum                        37                37
## Var: subjnum (Intercept)                     3.286             3.286
## Var: subjnum performlarge                    0.557             0.557
## Cov: subjnum (Intercept) performlarge       -0.512            -0.512
## Var: Residual                               21.811            21.811
## ============================================================================
===
```

(g) What does and does not change in the output? What interpretations will change?

This only impacts the intercept and the coefficient of performlarge, because now they apply to individuals with mpqnem = 0 (which we have none of in the dataset) rather than individuals with average mpqnem.

(h) In these interpretations, when do you need to set "other variables" to zero and when do you need to "hold them constant"?

For interpreting slopes (rather than interactions), set other variables to zero when they are involved in interactions with the variable of interest, otherwise hold them fixed when they are additional variables in the model.

(i) Is model 3 a significantly better fit compared to model 2?

```
anova(model2, model3)

## refitting model(s) with ML (instead of REML)

## Data: musicians
## Models:
## model2: na ~ performlarge * orchtype + (performlarge | subjnum)
## model3: na ~ performlargeF * orchtype + performlargeF * mpqnem.c + (perfor
mlargeF | subjnum)
##        npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## model2    8 3007.2 3040.8 -1495.6   2991.2
## model3   10 2996.4 3038.5 -1488.2   2976.4 14.734  2  0.0006319 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Yes, we have a small p-value (.0006319) for the likleihood ratio test (df = 2), so we conclude that the model with mpqnem significantly improves the fit of the model.

(j) Can you improve model3 further? What term would you suggest dropping and why (See earlier output.)?

The performlarge*mpqnem interaction does not seem necessary (above its t value was like 0.50). Taking it out (but leaving mpqnem in), does not appear to be a significantly worse fit (p-value 0.5534, df = 1). This allows the "intercept" to change with mpqnem, but not the slope of performance type, though the slope does differ person to person.

(k) Is the new model better?

```
summary(model4 <- lmer(na ~ performlargeF*orchtype + mpqnem.c + (performlarge
F | subjnum), data = musicians), corr = FALSE)

## Linear mixed model fit by REML ['lmerMod']
## Formula: na ~ performlargeF * orchtype + mpqnem.c + (performlargeF | subjn
um)
##    Data: musicians
##
## REML criterion at convergence: 2978.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.0396 -0.6388 -0.1571  0.4899  4.0632
##
## Random effects:
##  Groups   Name           Variance Std.Dev. Corr
##  subjnum  (Intercept)     3.2411  1.8003
##           performlargeF1  0.2655  0.5152   -0.39
##  Residual                21.8166  4.6708
## Number of obs: 497, groups:  subjnum, 37
##
## Fixed effects:
##                             Estimate Std. Error t value
## (Intercept)                 16.21301    0.54007  30.020
## performlargeF1              -1.21761    0.83158  -1.464
## orchtypeTRUE                 1.06300    0.80655   1.318
## mpqnem.c                     0.13918    0.03454   4.029
## performlargeF1:orchtypeTRUE -1.02971    1.08219  -0.952

anova(model3, model4)

## refitting model(s) with ML (instead of REML)

## Data: musicians
## Models:
```

```
## model4: na ~ performlargeF * orchtype + mpqnem.c + (performlargeF | subjnu
m)
## model3: na ~ performlargeF * orchtype + performlargeF * mpqnem.c + (perfor
mlargeF | subjnum)
##        npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## model4    9 2994.8 3032.7 -1488.4   2976.8
## model3   10 2996.4 3038.5 -1488.2   2976.4 0.3513  1     0.5534
```

Model 3 (the more complicated model here) is not significantly worse than model 4, so the interaction does not appear to be needed/does not explain significant variation in the slopes of perform large (the impact of large vs. small performance types).

Consider the following model

```
solo = as.numeric(musicians$perform_type1 == "Solo")
model5 <- lmer(na ~ previous + audience + solo + mpqpem + mpqab + orchtype +
mpqnem +
    mpqnem:solo + (previous + audience + solo | subjnum), data = musicians)

## boundary (singular) fit: see help('isSingular')

summary(model5, corr=FALSE)

## Linear mixed model fit by REML ['lmerMod']
## Formula: na ~ previous + audience + solo + mpqpem + mpqab + orchtype +
##     mpqnem + mpqnem:solo + (previous + audience + solo | subjnum)
##    Data: musicians
##
## REML criterion at convergence: 2882.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.1917 -0.6060 -0.1115  0.5345  3.9996
##
 Random effects:
  Groups   Name                      Variance Std.Dev. Corr
  subjnum  (Intercept)               14.47670 3.8048
           previous                   0.07077 0.2660   -0.65
           audienceJuriedRecital     18.30710 4.2787   -0.64 -0.12
           audiencePublicPerformance 12.79788 3.5774   -0.83  0.33  0.58
           audienceStudents           8.21838 2.8668   -0.63  0.00  0.84  0.66
           solo                       0.76408 0.8741   -0.68  0.48  0.21  0.90  0.49
  Residual                           15.28466 3.9096
 Number of obs: 497, groups:  subjnum, 37
##
## Fixed effects:
##                            Estimate Std. Error t value
## (Intercept)                 8.37010    1.91388   4.373
## previous                   -0.14304    0.06248  -2.289
## audienceJuriedRecital       4.07333    1.03114   3.950
## audiencePublicPerformance   3.06406    0.89251   3.433
```

```
## audienceStudents           3.61108    0.76802   4.702
## solo                       0.51382    1.39644   0.368
## mpqpem                    -0.08310    0.02408  -3.451
## mpqab                      0.20379    0.04741   4.299
## orchtypeTRUE               1.53064    0.58390   2.621
## mpqnem                     0.11460    0.03591   3.191
## solo:mpqnem                0.08303    0.04159   1.997
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

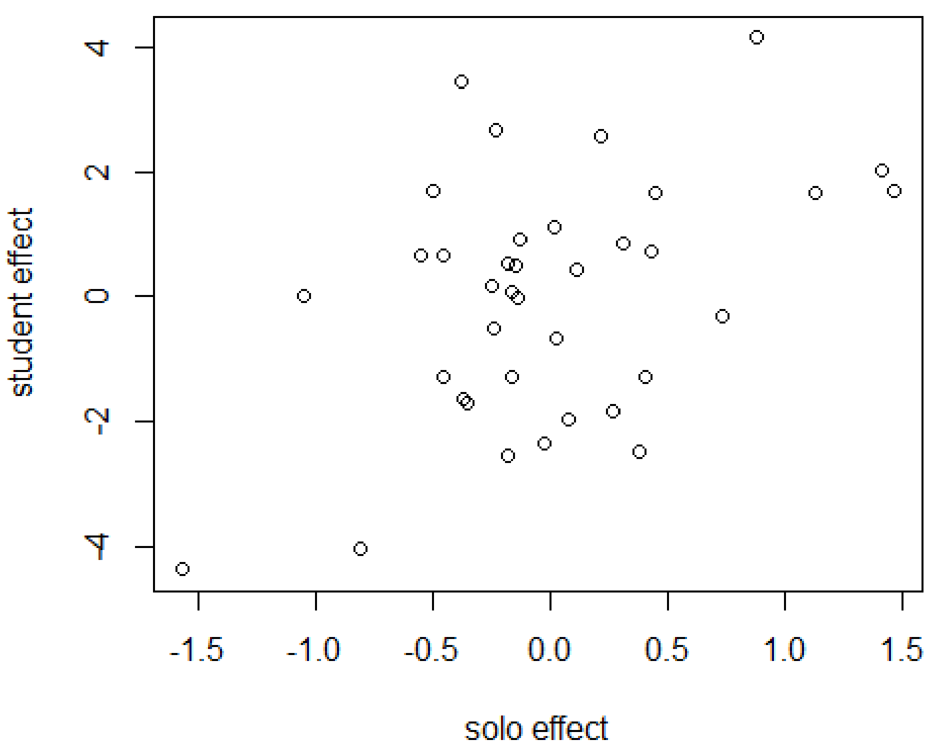
(l) Summarize what is going on with the "audience" variable (in the R code and in the output)

We are entering "audience" as a categorical variable in the "fixed" and "random" inputs. We get 3 coefficients for this four-category variable, and all of them are given random slopes and are included in the covariance estimates.

(m) How many variance/covariance terms are there? Interpret one of the correlations.

There are 15 correlations and 7 variances (probably why it's having trouble converging). The correlation between solo and audience students is 0.50, which says if the effect of a solo performance tends to be large, so does the effect of having students in the audience rather than the instructor.

```
plot(ranef(model5)$subjnum[,5]~ranef(model5)$subjnum[,6], xlab = "solo effect
", ylab="student effect")
```



If a solo performance makes you more nervous then performing in front of students (rather than instructor) tends to as well.

(n) Suggest a variable not collected in these data that might make sense for a Level 3 grouping variable. Explain your reasoning.

Perhaps musicians attend different schools and na as well as the effects of some of these variables could differ across schools.

Let's try some other fancy output functions

```
#install.packages("jtools")
jtools::summ(model3)
```

```
## MODEL INFO:
## Observations: 497
## Dependent Variable: na
## Type: Mixed effects linear regression
##
## MODEL FIT:
## AIC = 3002.11, BIC = 3044.19
## Pseudo-R² (fixed effects) = 0.11
## Pseudo-R² (total) = 0.22
##
## FIXED EFFECTS:
## ------------------------------------------------------------------------
##                                       Est.   S.E.   t val.    d.f.      p
## -------------------------------- ------- ------ -------- ------- ------
## (Intercept)                          16.26   0.55    29.49   30.80   0.00
## performlargeF1                       -1.23   0.87    -1.42   34.02   0.16
## orchtypeTRUE                          1.00   0.82     1.22   32.99   0.23
## mpqnem.c                              0.15   0.04     3.87   31.93   0.00
## performlargeF1:orchtypeTRUE          -0.95   1.13    -0.84   25.90   0.41
## performlargeF1:mpqnem.c              -0.03   0.05    -0.56   25.44   0.58
## ------------------------------------------------------------------------
##
## p values calculated using Kenward-Roger standard errors and d.f.
##
## RANDOM EFFECTS:
## ----------------------------------------
##    Group       Parameter      Std. Dev.
## ---------- ---------------- -----------
##   subjnum     (Intercept)       1.81
##   subjnum    performlargeF1     0.75
##   Residual                      4.67
## ----------------------------------------
##
## Grouping variables:
## ---------------------------
##    Group    # groups   ICC
## --------- ---------- ------
```

```
##  subjnum      37      0.13
## ---------------------------

#install.packages("stargazer")
stargazer::stargazer(model3, type="text")  #can use type="html"?

##
## ====================================================
##                             Dependent variable:
##                         ---------------------------
##                                     na
## ----------------------------------------------------
## performlargeF1                    -1.235
##                                   (0.843)
##
## orchtype                           1.001
##                                   (0.817)
##
## mpqnem.c                          0.148***
##                                   (0.038)
##
## performlargeF1:orchtype           -0.949
##                                   (1.106)
##
## performlargeF1:mpqnem.c           -0.030
##                                   (0.052)
##
## Constant                          16.257***
##                                   (0.548)
##
## ----------------------------------------------------
## Observations                        497
## Log Likelihood                  -1,491.054
## Akaike Inf. Crit.                3,002.108
## Bayesian Inf. Crit.              3,044.194
## ====================================================
## Note:                    *p<0.1; **p<0.05; ***p<0.01

#install.packages("sjPlot")
sjPlot::tab_model(model3)
```

		na	
Predictors	*Estimates*	*CI*	*p*
(Intercept)	16.26	15.18 – 17.33	**<0.001**
performlargeF [1]	-1.23	-2.89 – 0.42	0.144
orchtypeTRUE	1.00	-0.60 – 2.61	0.221
mpqnem c	0.15	0.07 – 0.22	**<0.001**
performlargeF [1] × orchtypeTRUE	-0.95	-3.12 – 1.22	0.391
performlargeF [1] × mpqnem c	-0.03	-0.13 – 0.07	0.565
Random Effects			
σ^2	21.81		
τ_{00} subjnum	3.29		
τ_{11} subjnum.performlargeF1	0.56		
ρ_{01} subjnum	-0.38		
ICC	0.13		
N subjnum	37		
Observations	497		
Marginal R^2 / Conditional R^2	0.108 / 0.221		

(o) Which were you able to use? What are some advantages and disadvantages of the different outputs?

The main difference is that some give p-values and when they do the p-value algorithms can differ! Some also don't report the random effects! It says stargazer works better with html but it didn't work well for me. sjPlot looked awesome for me but only works with html output. sjPlot also seems to work best of these to list multiple models to compare.