Stat 414 - Day 18
Case Study

Part 1

Stage fright can be a serious problem for performers, and understanding the personality underpinnings of performance anxiety is an important step in determining how to minimize its impact. Sadler and Miller (2010) studied the emotional state of musicians before performances and factors which may affect their emotional state. Data were collected on 37 undergraduate music majors over the course of an academic year. Students completed diaries prior to performances, including the Positive Affect Negative Affect (PANAS) before each performance. The negative affect measure of this instrument is used as a measure of performance anxiety. Factors include type of performance (solo, large ensemble, small ensemble), audience (orchestral vs. keyboard/vocalist), age, gender, instrument, and years studying.

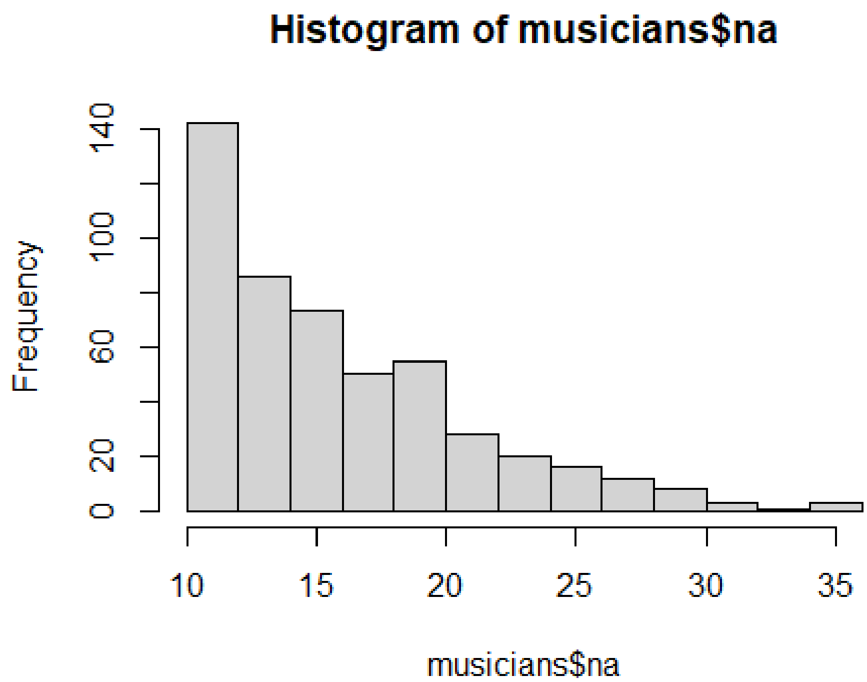
(a) Load in the data and confirm the variables I mentioned.

```
musicians = read.delim("https://www.rossmanchance.com/stat414/data/musicians.txt" ,
"\t", header=TRUE)
head(musicians)
  subjnum id diary previous perform_type1      memory          audience pa na
1       1  1     1        0          Solo Unspecified        Instructor 40 11
2       1  1     2        1 LargeEnsemble      Memory PublicPerformance 33 19
3       1  1     3        2 LargeEnsemble      Memory PublicPerformance 49 14
4       1  1     4        3          Solo      Memory PublicPerformance 41 19
5       1  1     5        4          Solo      Memory          Students 31 10
6       1  1     6        5          Solo      Memory          Students 33 13
  age gender instrument1 years_study mpqab mpqsr mpqpem mpqnem mpqcon
1  18 Female       voice           3    16     7     52     16     30
2  18 Female       voice           3    16     7     52     16     30
3  18 Female       voice           3    16     7     52     16     30
4  18 Female       voice           3    16     7     52     16     30
5  18 Female       voice           3    16     7     52     16     30
6  18 Female       voice           3    16     7     52     16     30
```

(b) First do some data exploration.

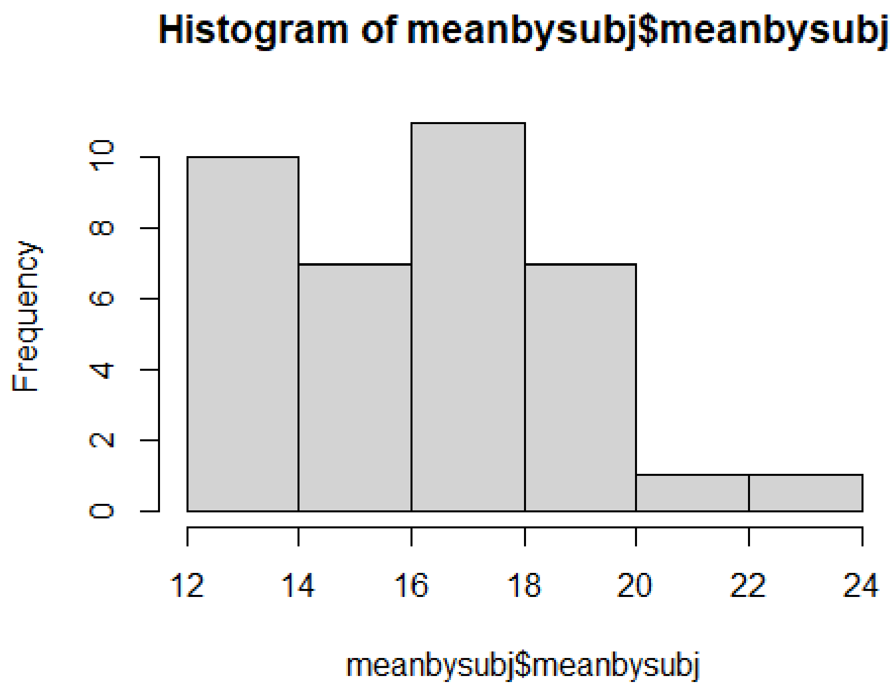
1. Is there much variation in negative anxiety (na) across performances? Is there much variation in na across musicians?

```
hist(musicians$na)
```



negative anxiety by musician

```
#what is the following doing?
meanbysubj <- musicians |>
  group_by(subjnum) |>
  summarise(meanbysubj = mean(na, na.rm = TRUE)) |>
  ungroup()

hist(meanbysubj$meanbysubj)
```



negative anxiety by musician

The first histogram includes all the observations (across performances). The second histogram graphs the mean na for each musician (averaging over the number of performances for that musician). There appears to be quite a bit of variability across performances (ranging from 10 to 35 or so) but less in the averages by subject (which is not unexpected for averages to vary less than individual observations).

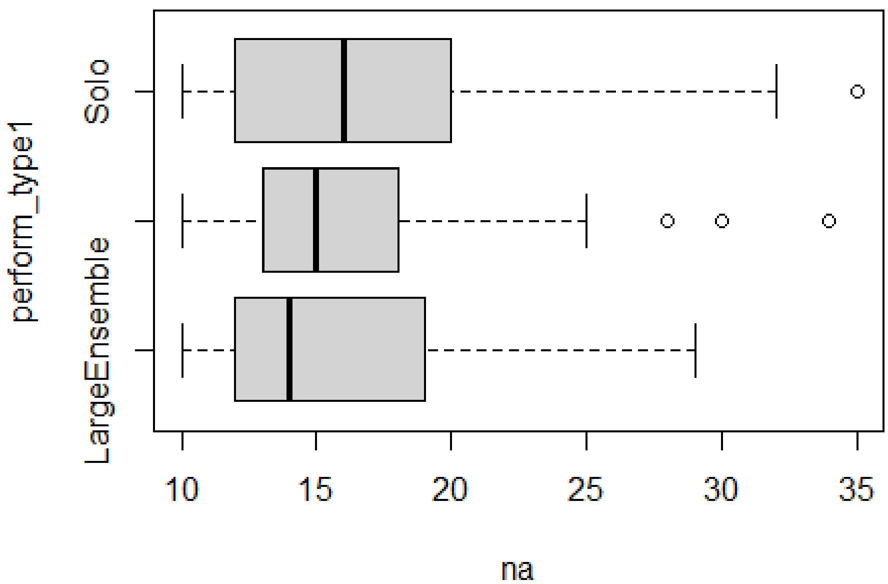
2. Confirm the types of variables/number of categories and start thinking about how you will incorporate these into a model.

```
table(musicians$perform_type1)

LargeEnsemble SmallEnsemble          Solo
          136            82           279
table(musicians$instrument1)

keyboard(pianoorgan)   orchestralinstrument                    voice
                  75                    235                      187
```

Three performance types and three instruments. So slope coefficients will represent differences in mean na between categories (vs. a reference group if using indicator coding)

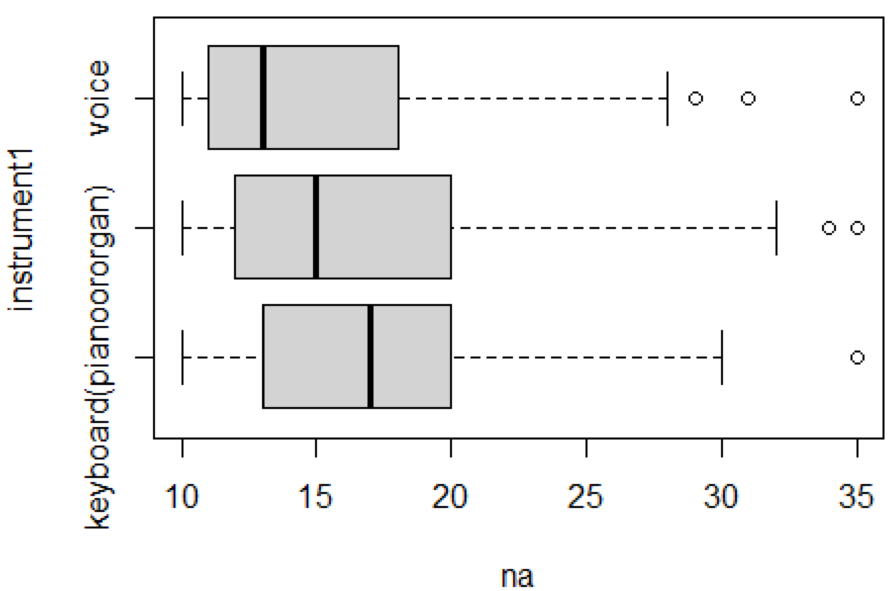
3. Does negative affect seem to vary by performance type? What about instrument? Do you think either will be a useful variable to include?

```
boxplot(na ~ perform_type1, data = musicians, horizontal=TRUE)
```



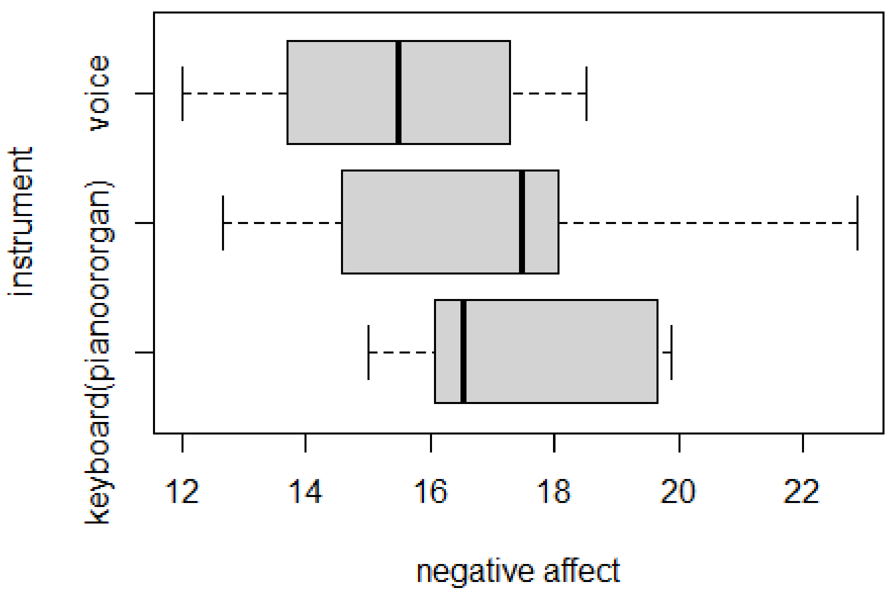
boxplot of negative anxiety by performance type and instrument

```
boxplot(na ~ instrument1, data = musicians, horizontal=TRUE)
```



boxplot of negative anxiety by performance type and instrument

```
#what is the following doing?
meanbysubj <- musicians |>
  group_by(subjnum) |>
  summarise(meanbysubj = mean(na, na.rm = TRUE), instrument = head(instrument1, 1))
|>
  ungroup()

boxplot(meanbysubj$meanbysubj ~ meanbysubj$instrument, horizontal = TRUE, ylab="ins
trument", xlab="negative affect")
```

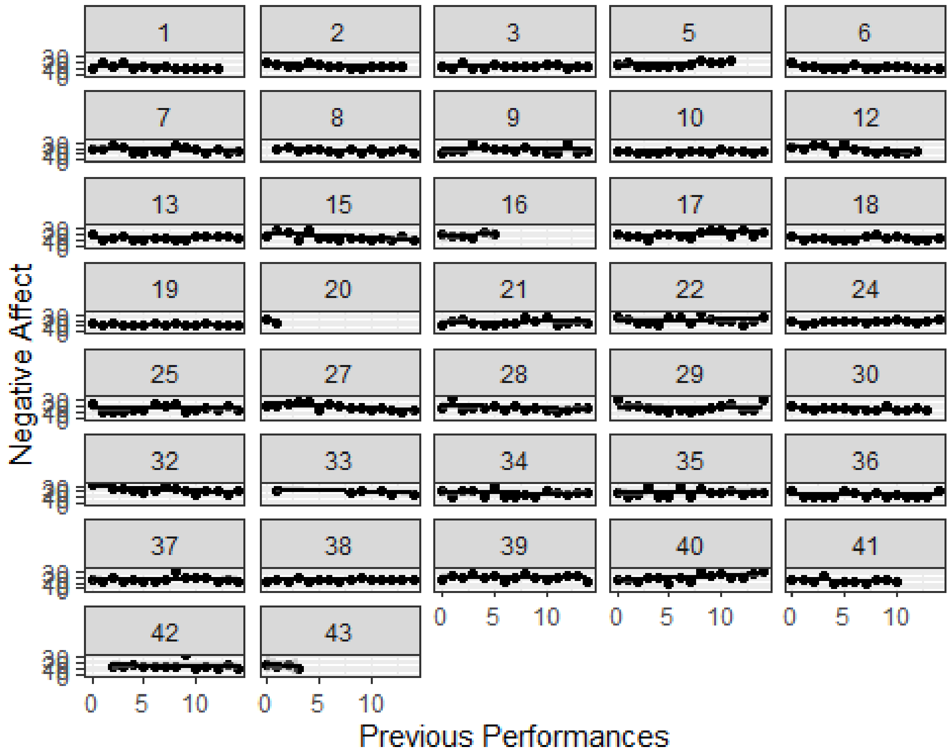


boxplot of negative anxiety by performance type and instrument

There isn't that much variation in na by performance type (from the first graph, ignoring musicians differences). Similarly, we don't see much of a consistent difference in the first boxplot by instrument across all the performances. But when you look at the average anxiety at the musician level, there appears to be less anxiety on average for the vocalists. (The group_by is taking the first instrument listed for each subject since that is a subject-level variable and doesn't change across performances.)

4. Does the relationship between negative affect and number of previous performances appear to differ across musicians? What does that suggest including in the model?

```
ggplot(musicians,aes(x=previous,y=na)) +
  geom_point() + geom_smooth(method="lm",color="black", formula='y~x') +
  facet_wrap(~id,ncol=5) +
  labs(x="Previous Performances",y="Negative Affect") +
  theme_bw()
```

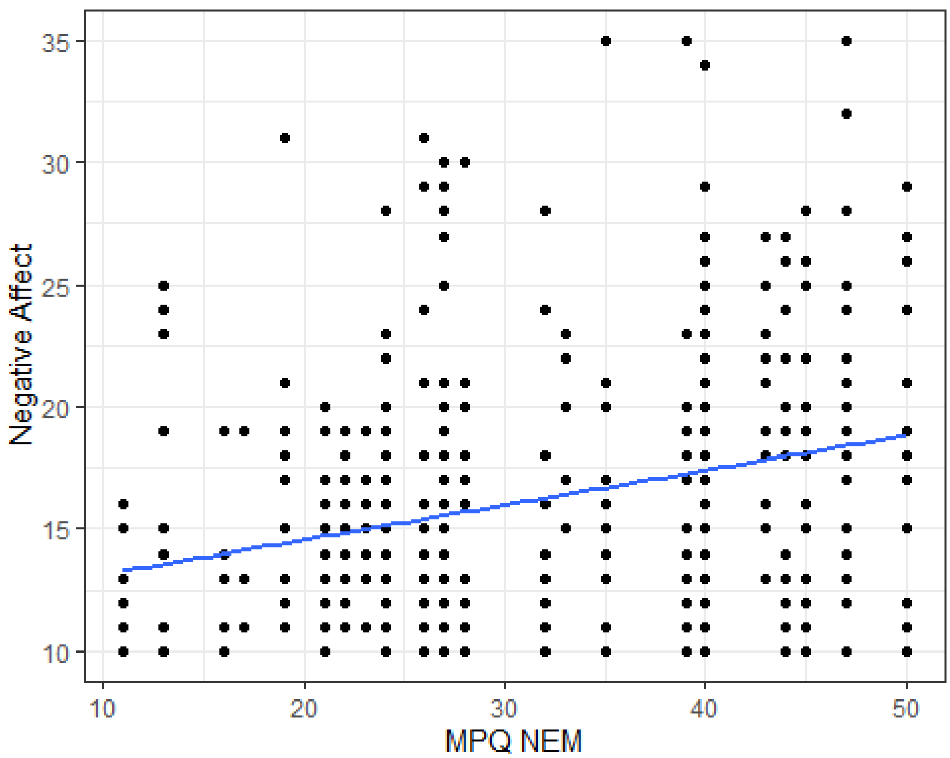


negative anxiety vs. number of previous performances

The main thing I notice is different number of performances across the musicians, but most of these scatterplots look weakly positive? Seeing different slopes between these graphs would suggest including random slopes in the model (an interaction between previous performances and musician).

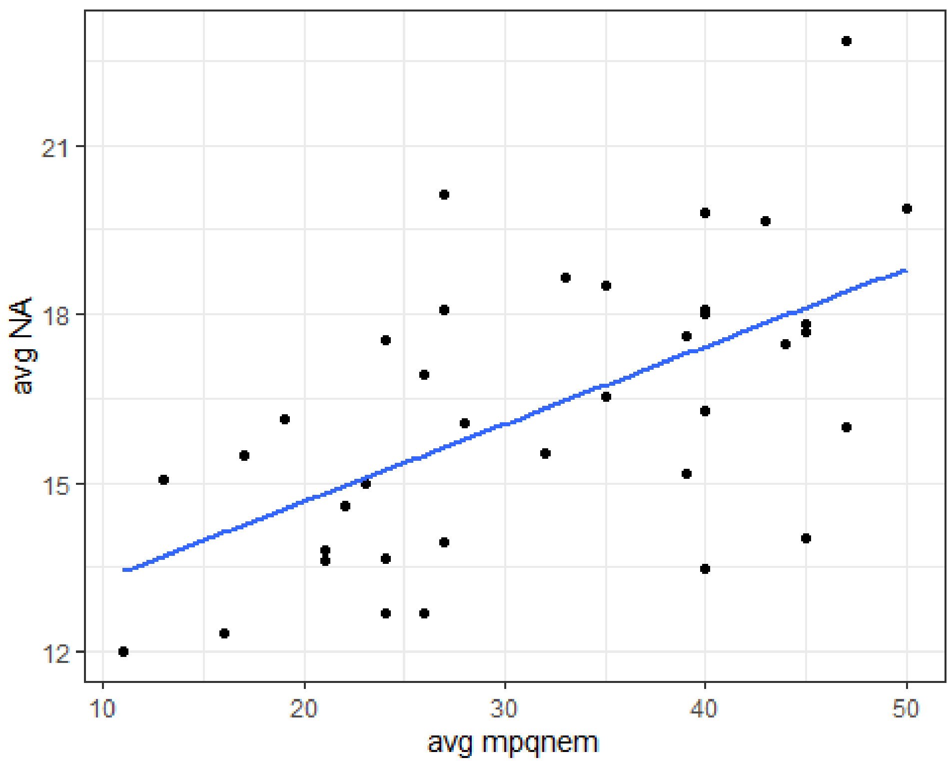
The following plots look at the musicians' negative emotionality composite scale from the MPQ instrument (mpqnem).

```
ggplot(musicians, aes(x = mpqnem, y = na)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  labs(x = "MPQ NEM", y = "Negative Affect" ) +
  theme_bw()
```



negative anxiety MPQ score

```
meanbysubj <- musicians |>
  group_by(subjnum) |>
  summarise(meanbysubj = mean(na, na.rm = TRUE), instrument = head(instrument1, 1),
mpqnem = mean(mpqnem,  na.rm = TRUE)) |>
  ungroup()

#run these two lines together
ggplot(meanbysubj, aes(x = mpqnem, y = meanbysubj)) +
  geom_point()+
  geom_smooth(method = "lm", se = FALSE, formula = 'y~x') +
  labs(x = "avg mpqnem", y = "avg NA") +
  theme_bw()
```



negative anxiety MPQ score

Note, mpgnem is a musician level variable (a person's overall disposition). It's not totally clear from the graph above, but I verified that this value stayed constant for each musician. In this case, it doesn't matter whether we use the first value or the average (they are identical). If we do see some changes in values over the course the study for any of the musicians, then you could treat as a Level 1 variable or you could take the average, realizing this will ignore any "measurement error" in that value.

(c) What patterns do you notice? Why is it expected? What's the difference between the two plots? Is one better than the other?

The first graph looks at na vs. mpqnem for each performance. Because mpqnem is a subject-level variable that's the same for every musician, that's why we get the stacks. If we instead aggregate ot the musician level and look at mean negative anxiety (averaging across the performances) vs. mpqnem, we have have one observation per musician. For the second graph (aggregating), we see that the average na tends to be larger for individual with more negative emotionality. The second graph is probably better but in many cases will tell the same story about the relationship (see next paragraph). The only worry would be if the number of performances were very different across the performers (and could throw off some of the group comparisons). (We don't mind showing lots of dots for the same individual when exploring in the graphs, we worry about the "inflated sample size" more when carrying out tests of significance/inference.)

The two plots can differ when the number of performances differs among musicians. Suppose, for instance, one vocalist who is very anxious had twice as many performances as anybody else. In that case, plot (a) would show a higher level of anxiety for vocalists compared to other instruments than plot (b). On the other hand, if a keyboardist had only one performance in which she felt no anxiety, that one performance would count just as much in plot (b) as a keyboardist with 20 performances averaged together, so plot (b) would show a lower anxiety level for keyboardists.

Some of the questions we might want to explore: *Which characteristics of individual performances are most associated with performance anxiety? Which characteristics of student musicians are most associated with performance anxiety? Are any of these associations statistically significant? Does the significance remain after controlling for other covariates?*

But of course, need to account for lack of independence in performances by the same musician.

(d) Identify Level 1 and Level 2. Identify some variables at each level.

Level 1: performance (variable: who the performance was in front of)
Lever 2: musician (variables: type of instrument, baseline anxiety measures)

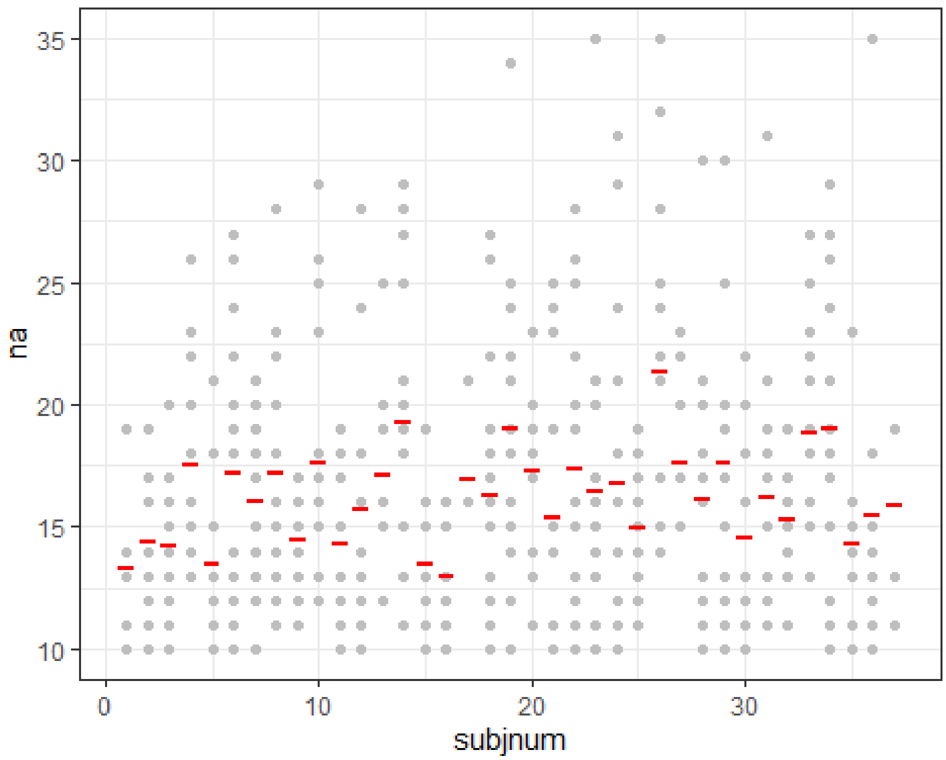
Start with a random intercepts ("null") model to assess the variation in performance anxiety ("na") among the musicians.

Code

```
model0 = lmer(na ~ 1 + (1 | subjnum), data = musicians)
summary(model0)
Linear mixed model fit by REML ['lmerMod']
Formula: na ~ 1 + (1 | subjnum)
   Data: musicians

REML criterion at convergence: 3006

Scaled residuals:
   Min     1Q Median     3Q    Max
-1.904 -0.689 -0.208  0.528  4.129

Random effects:
 Groups   Name        Variance Std.Dev.
 subjnum  (Intercept)  4.95    2.22
 Residual             22.46    4.74
Number of obs: 497, groups:  subjnum, 37

Fixed effects:
            Estimate Std. Error t value
(Intercept)   16.237      0.428    37.9
```

```
#model0 =
#summary(model0)

fits0 = fitted.values(model0, level = 1)
```

```
ggplot(musicians,  aes(y = fits0, x = subjnum, group= factor(subjnum))) +
  geom_point(aes(y=na, x= subjnum), col="grey") +
  geom_boxplot(color = "red") +  #of the one value, to show a line
  theme_bw()
```



graph of fitted random intercepts model

(e) What is the ICC?

4.95/(4.95 + 22.46) = .1806. About 18% of the variation in na is among musicians, the rest is across performances within musicians.

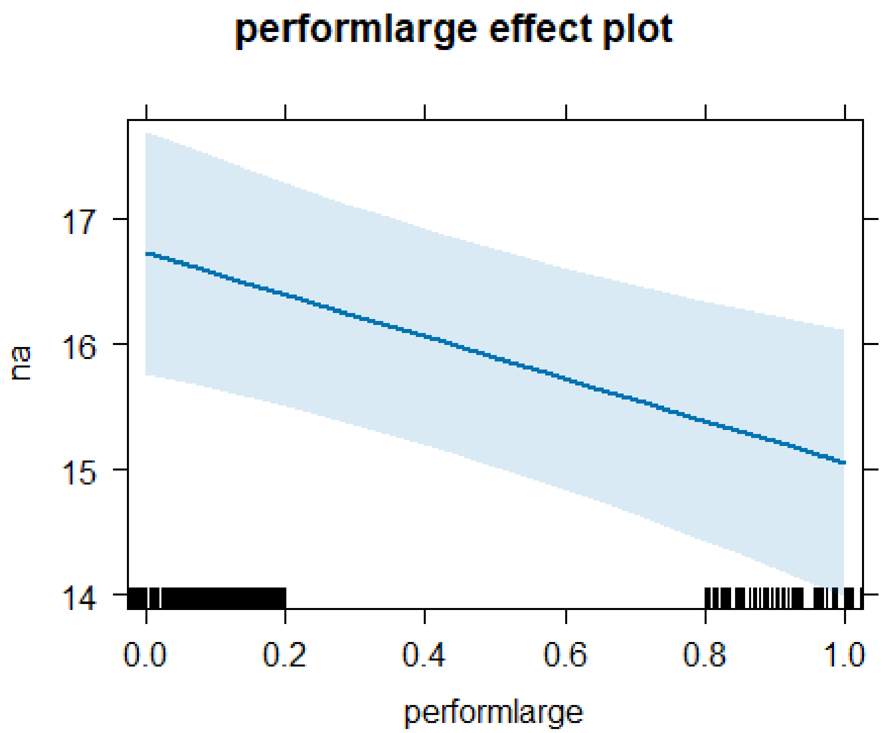
So again, what we are doing with a multilevel model is not that different from including "subject" in the model to account for that source of variation (and source of dependence). The multilevel model has two main consequences: shrinkage, and the ability to ask different research questions, like about level 2 variables explaining some of the subject to subject differences.

Suppose we want to predict performance anxiety based on the type of performance (large ensemble or not). Fit a model that looks at the effects of type of performance (large vs. small/solo), allowing this effect to vary by musician.

ⓘ Code

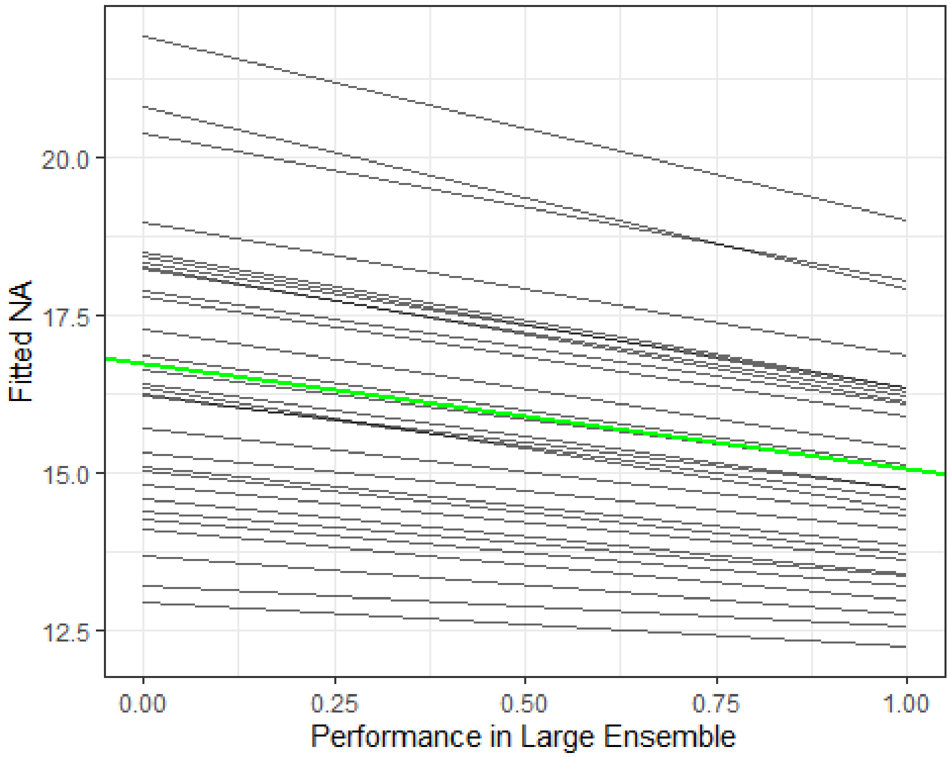
```
musicians$performlarge = as.numeric(musicians$perform_type1 == "LargeEnsemble")
head(cbind(musicians$perform_type1, musicians$performlarge))
     [,1]            [,2]
[1,] "Solo"          "0"
[2,] "LargeEnsemble" "1"
[3,] "LargeEnsemble" "1"
[4,] "Solo"          "0"
[5,] "Solo"          "0"
[6,] "Solo"          "0"
model1 = lmer(na ~ performlarge + (1 + performlarge | subjnum), data = musicians
)
summary(model1, corr=FALSE)
Linear mixed model fit by REML ['lmerMod']
Formula: na ~ performlarge + (1 + performlarge | subjnum)
   Data: musicians

REML criterion at convergence: 2994

Scaled residuals:
   Min     1Q Median     3Q    Max
-1.989 -0.683 -0.198  0.484  4.140

Random effects:
 Groups   Name         Variance Std.Dev. Corr
 subjnum  (Intercept)   6.333   2.517
          performlarge  0.743   0.862    -0.76
 Residual              21.771   4.666
Number of obs: 497, groups:  subjnum, 37

Fixed effects:
             Estimate Std. Error t value
(Intercept)    16.730      0.491   34.09
performlarge   -1.676      0.542   -3.09
```

```
#First convert performance type to a binary variable, just to simplify things a bit
initially
musicians$performlarge = as.numeric(musicians$perform_type1 == "LargeEnsemble")
head(musicians$performlarge)
[1] 0 1 1 0 0 0
#model1 =
#summary(model1)

#install.packages("effects")
#library(effects)
plot(allEffects(model1))
```



fitted model 1

```
fits1 = fitted.values(model1, level =1)

ggplot(musicians,
       aes(x = performlarge, y = fits1, group = factor(subjnum))) +
  geom_line(alpha = 0.6) +
  geom_abline(
    intercept = fixef(model1)[1],
    slope     = fixef(model1)[2],
    color     = "green",
    linewidth = 1
  ) +
  theme_bw() +
  labs(
    x = "Performance in Large Ensemble",
    y = "Fitted NA",
   )
```



fitted model 1

(f) Explain in plain language what it means for this model to have "random intercepts." Hint: What does the fixed slope for the 0/1 performance type variable represent?) What does it mean for this model to have "random slopes" for that variable?

The random intercepts pertain to the small and solo performances (performlarge = 0), so they tell us about the musician to musician variability in (average) anxiety for the small and solo performances. (Different musicians have more/less anxieity on average during small and solo performances.) The random slopes represent the change in anxiety between smaller performances and large performances. So the random slopes allow the "effect" of the performance type to vary by musician. (Some musicians may have similar anxiety for both types of performances - flatter slope, some may have less anxiety with larger performances compared to smaller performances - more "negative" slope. We probably don't expect any positive slopes, but could be!)

(g) What does $\hat{\sigma}$ represent here in this new model? What do the two Level 2 variance components represent?

The Level 1 variance is the model-estimated variability within subjects' performances (after adjusting for type of performance). In other words, the "average" unexplained variation in the na measures of different performances by the same performer.

The Level 2 variances represent the variation in the intercepts (na for smaller performances) and slopes (change in na between small and large performances) and the covariance between the intercepts and slopes.

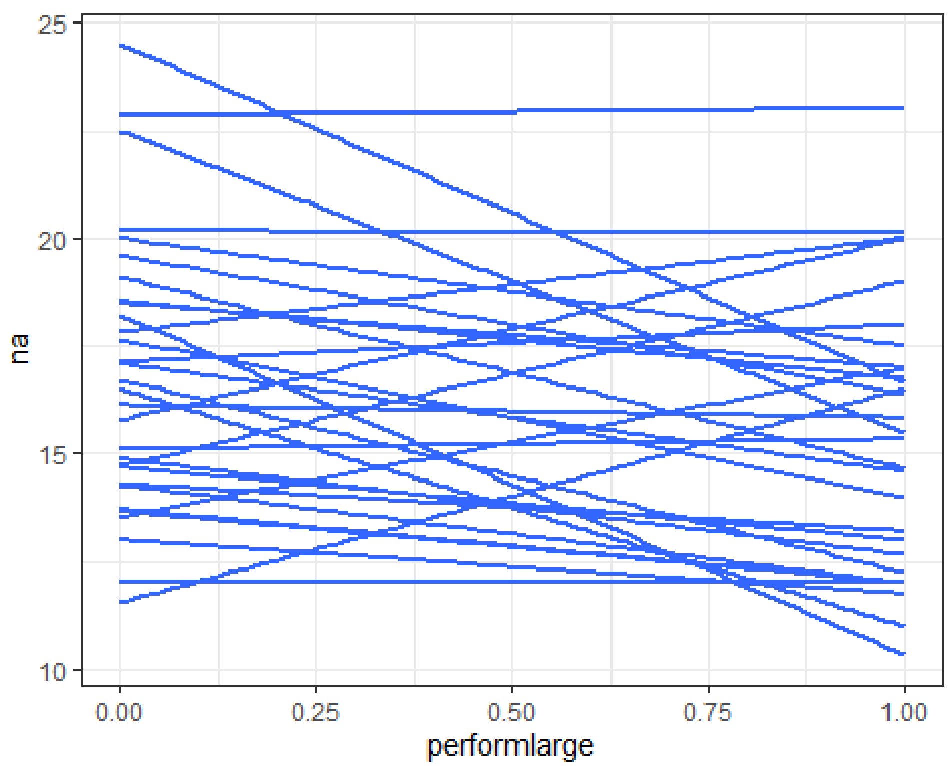
(See k) The negative correlation between the intercepts and slopes indicates that subjects with more anxiety with small performances tend to have larger decreases (more negative slopes) in their anxiety with larger performances (corresponding to the "fanning in" we see in the above graph). In other words, those with less anxiety in small performance tend to see less of a difference (slope closer to zero) between large and small performances.

(h) The above graph shows the fitted equations from the multilevel model for each performer. How do you think the graph will differ if we fit a separate OLS line for each performer?

There would very likely be more variability among the OLS lines vs. all going in the same general direction as the model (above graph) won't perfectly describe each individual. In other words, we see shrinkage in the above graph where the lines are encouraged to have the same general pattern and lines from the multilevel model will be a weighted combination of the performer's own (OLS) line and the overall green line averaged across all the performers.

Code

```
#Below are the OLS fits.
ggplot(musicians) +
  aes(x = performlarge, y = na, group= subjnum) +
  stat_smooth(method = "lm",  se = FALSE, formula = "y~x") +
  theme_bw()
```



fitted OLS line for each musician

(i) Interpret your model output: Do the signs of the coefficients of the fixed effects make sense in context? What do you learn about the effect of large ensemble performances on anxiety? How much of the performance-to-performance variation is explained by the type of performance? How did the intercept variance change? Does this surprise you?

We see a negative coefficient of performlarge, showing a decrease of 1.676 in anxiety (on average across the performers) moving from small/solo performances to large performances (makes sense to me that anxiety would go down for larger performances compared to smaller). The effect appears to be statistically significant (t = -3.09 < -2.00). The residual variance (performance to performance within a musician) is now 21.77 from 22.46, a reduction of (22.46 - 21.77)/22.46 = .0307, or about 3%. In other words, the type of performance explains about 3% of the unexplained within-performer variability, which is not a lot.

The intercept variance is now 6.33 (and represents the variability in intercepts for the small performances), up from 4.95, so it has increased (but keep in mind these two "intercept variances" for these two models mean different things). This increase in the variation of the intercepts is a little surprising (but we've seen it before when we allow for random slopes) but reflects a hidden relationship between performance type and musician. Some musicians are more likely to have more large performances than other musicians, and it turns out the musicians with a higher proportion of large performances tended to be more anxious than musicians with a lower proportion of large performances. For example, musician 22 has a lot of large performances, but is actually a more anxious musician overall, so after accounting for type of performance, the 'residual' for that musician (that musician's effect) needs to be larger. This happens for several musicians and so the unexplained musician-to-musician variability for small performances increases.

Recall what we learned about how the (Level 2) variance changes with x in a random slopes model: $Var(y_{ij}) = \tau_0^2 + x_{ij}\tau_1^2 + 2x_{ij}\tau_{01}$. For this model, when x = 1 (larger performances) we have 6.333 + 1(.7429) + 21(-.76)(2.5165)(0.8619) = 3.77. When x = 0 (smaller performances), we have 6.333 (variability in na scores is larger for smaller performances than for larger performances). The point is, for large performances, there is less variation among musicians (3.77) than when we didn't account for performance size (4.95). So when we force the slopes (performance-size effect) to be the same for every musician (not random slopes, ignoring performance type) we were getting more of an estimated 'average' variability in na across musicians.

(j) Which is larger, the variation in the intercepts or in the slopes? What does that tell you in context?

The variation in the intercepts is larger, so there is more person-to-person variation in negative anxiety (in the small performances) than in the effect of performance size on negative anxiety.

(k) Interpret the slope/intercept correlation in this context. Are the effects "fanning in" or "fanning out"? Or do they cross over? Why does this relationship between slopes and intercepts make sense in context?

The correlation is negative. The variance is minimized at (.76)(2.5165)(.8619) ≈ 1.65, so beyond our 0/1 values for performance size. This means the lines are fanning in. Because the slopes are generally negative, the musicians who are more anxious with small performances tend to have a bigger decrease (smaller/more negative slope) in anxiety in moving to large performances. Those who are less anxious with small performances tend to decrease for large performances as well, but don't have as much 'room' to drop. This is also consist with the larger variation in na values with small performances, but more consistency in the larger performances.

(l) Write out a (new) model (by level and then composite) that also uses the type of performance (large ensemble or not) with random intercepts and slopes that depend on type of instrument (orchestral or not).

Level 1: $na_{ij} = \beta_{0j} + \beta_{1j} largeperf_{ij} + \epsilon_{ij};$
Level 2: $\beta_{0j} = \beta_{00} + \beta_{01} instrument_j + u_{0j};$

$$\beta_{1j} = \beta_{10} + \beta_{11} instrument_j + u_{1j}.$$

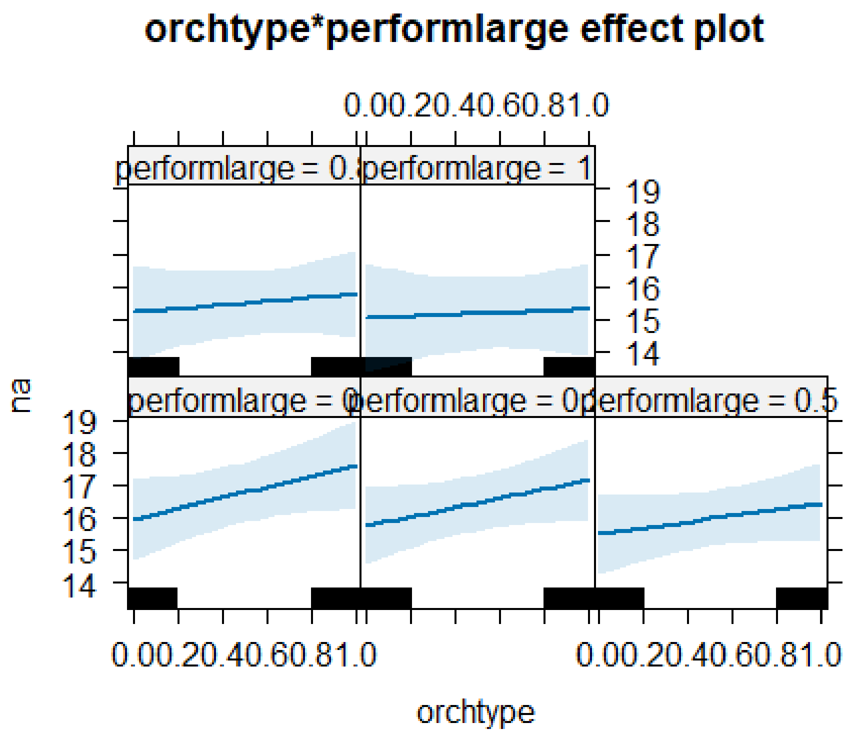
Composite: $na_{ij} = \beta_{00} + \beta_{01} instrument_j + u_{0j} + \beta_{10} largeperf_{ij} +$
$\beta_{11} largeperf_{ij} imesinstrument_j + u_{1j} imeslargeperf_{ij} + \epsilon_{ij}$

Fit the model for (l): Make a binary variable for orchestra. Include orchestra, largeperformance, and their interaction as fixed effects, and then random intercepts and random slopes for performance type).

ⓘ Code

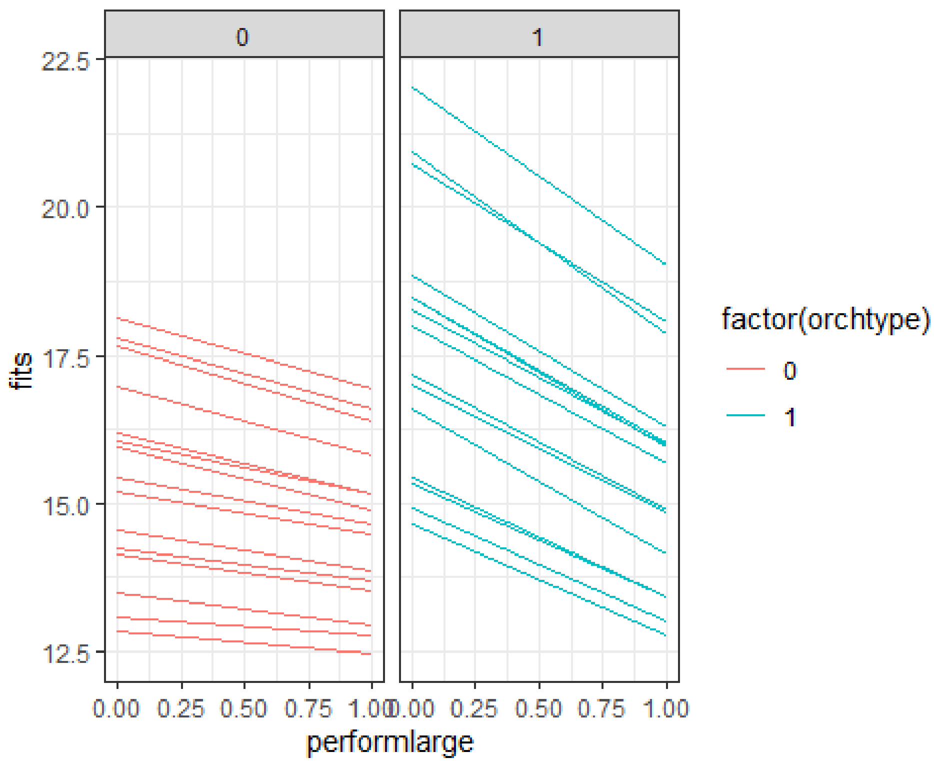
```
musicians$orchtype = ifelse(musicians$instrument1 == "orchestralinstrument", 1,
0)
model2 =  lmer(na ~ orchtype*performlarge + (1 + performlarge | subjnum), data =
musicians)
summary(model2, corr=FALSE)
Linear mixed model fit by REML ['lmerMod']
Formula: na ~ orchtype * performlarge + (1 + performlarge | subjnum)
   Data: musicians

REML criterion at convergence: 2987

Scaled residuals:
   Min     1Q Median     3Q    Max
-1.940 -0.663 -0.177  0.480  4.186

Random effects:
 Groups   Name         Variance Std.Dev. Corr
 subjnum  (Intercept)   5.655   2.378
          performlarge  0.452   0.672    -0.63
 Residual              21.807   4.670
Number of obs: 497, groups:  subjnum, 37

Fixed effects:
                      Estimate Std. Error t value
(Intercept)             15.930      0.641   24.83
orchtype                 1.693      0.945    1.79
performlarge            -0.911      0.845   -1.08
orchtype:performlarge   -1.424      1.099   -1.30
```

```
musicians$orchtype = ifelse(musicians$instrument1 == "orchestralinstrument", 1, 0)
#model2 =
#summary(model2, corr=FALSE)

plot(allEffects(model2))
```



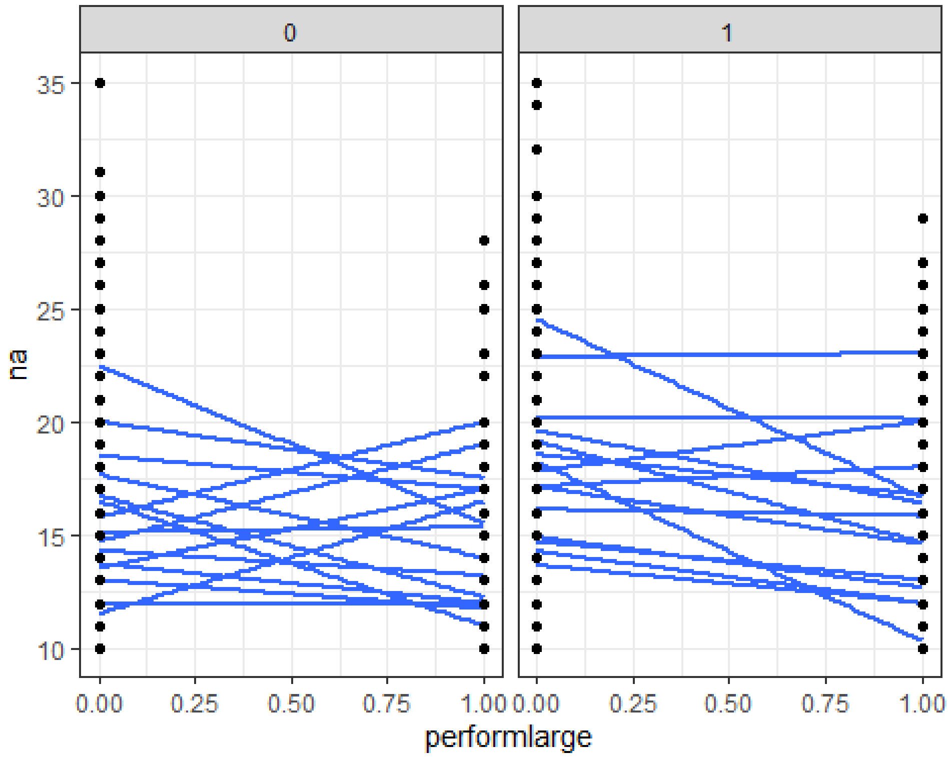
fitted model 2

```
fits = fitted.values(model2, level =1)
ggplot(musicians, aes(y = fits, x= performlarge, group = factor(subjnum), col=facto
r(orchtype))) +
  facet_wrap(~orchtype) +
  geom_line()+
  theme_bw()
```



fitted model 2

```
#"raw data"
ggplot(musicians) +
  aes(x = performlarge, y = na, group= subjnum) +
  stat_smooth(method = "lm",  se = FALSE, formula = "y~x") +
  facet_wrap(~orchtype) +
  geom_point() +
  theme_bw()
```



fitted model 2

(m) Interpret the interaction between performance type and orchestra type in context.

If the musician has an orchestral instrument (rather than voice or keyboard), then they experience a (1.42) larger decrease (on average) in anxiety moving from small to large performance types than the voice and keyboardists. However, this may not be statistically significant with a t-value of -1.295 > -2.00.

(n) How much variability in the intercepts does including type of instrument explain? How much variability in the slopes?

Now these comparisons make a bit more sense because we are focusing on changes in Level 2 variance from adding a Level 2 variable.

Variability in intercepts is now 5.655, down from 6.33, a decrease of `r{(6.33 - 5.655)/6.33} = about 11%

Variability in slopes is now 0.452, down from 0.7429, a decrease of ’r{(.7429 - .452)/.7429} = about 39%

How did the estimate of within group variation change?

Our residual standard error is now 21.807, about the same as 21.77. We don’t expect it to change when we add a Level 2 variable, and the change can be explained by small numerical adjustments as we simultaneously estimate all of these parameters.

(o) Summarize what you learn about the effect of type of instrument on the intercepts and the slopes.

So knowing the type of instrument tells us more about why some musicians see a sharper drop in negative anxiety moving from small to large performance types than others. It explains a bit (but not as much) about why some musicians have more anxiety on average for smaller performances than others.

(p) Maybe with the interaction between performance type and instrument type we no longer need the random slopes… Investigate this. Document how you did so (both the model equations and the R code).

ⓘ Code

```
model2b =   lmer(na ~ performlarge*orchtype + (1 | subjnum), data = musicians)
anova(model2, model2b)
Data: musicians
Models:
model2b: na ~ performlarge * orchtype + (1 | subjnum)
model2: na ~ orchtype * performlarge + (1 + performlarge | subjnum)
        npar  AIC  BIC logLik -2*log(L) Chisq Df Pr(>Chisq)
model2b    6 3004 3029  -1496      2992
model2     8 3007 3041  -1496      2991  0.43  2       0.81
```

```
#model2b =
#anova(model2, model2b)
```

We notice AIC and BIC values are better for model 2b, without the random slopes, and the more complicated model is not significantly better in terms of log likelihood (p-value = .8065). Using the same slope for performance type (difference in anxiety between large and small performances) on everyone, differing only by the type of instrument, seems to be as well-fitting of a model as also allowing a different slope for every musician.

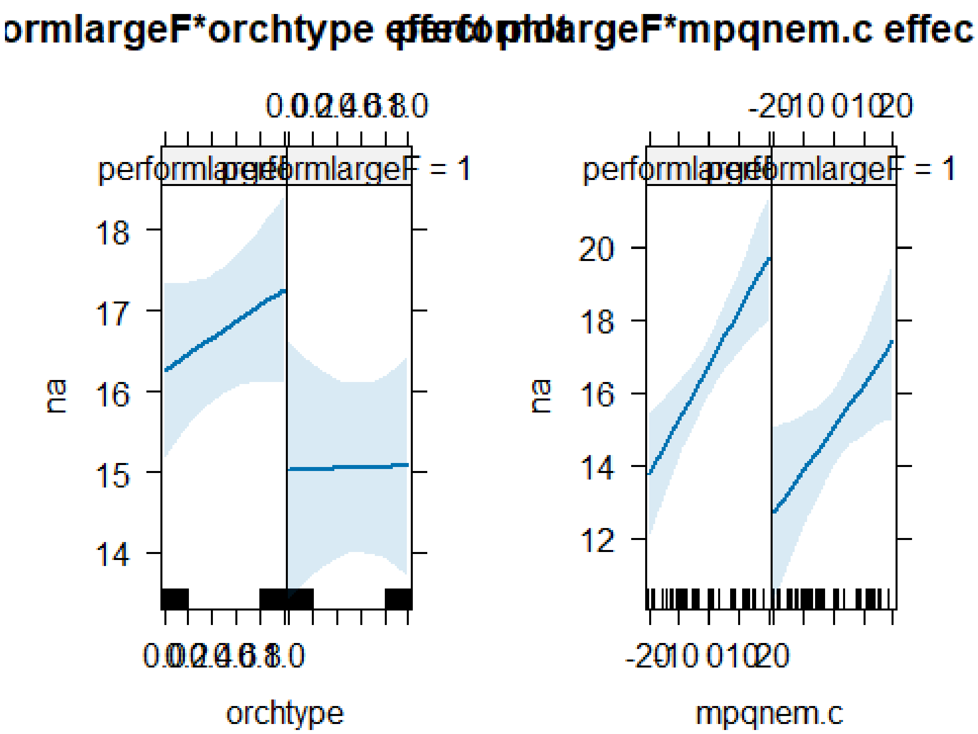
Part II:

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.

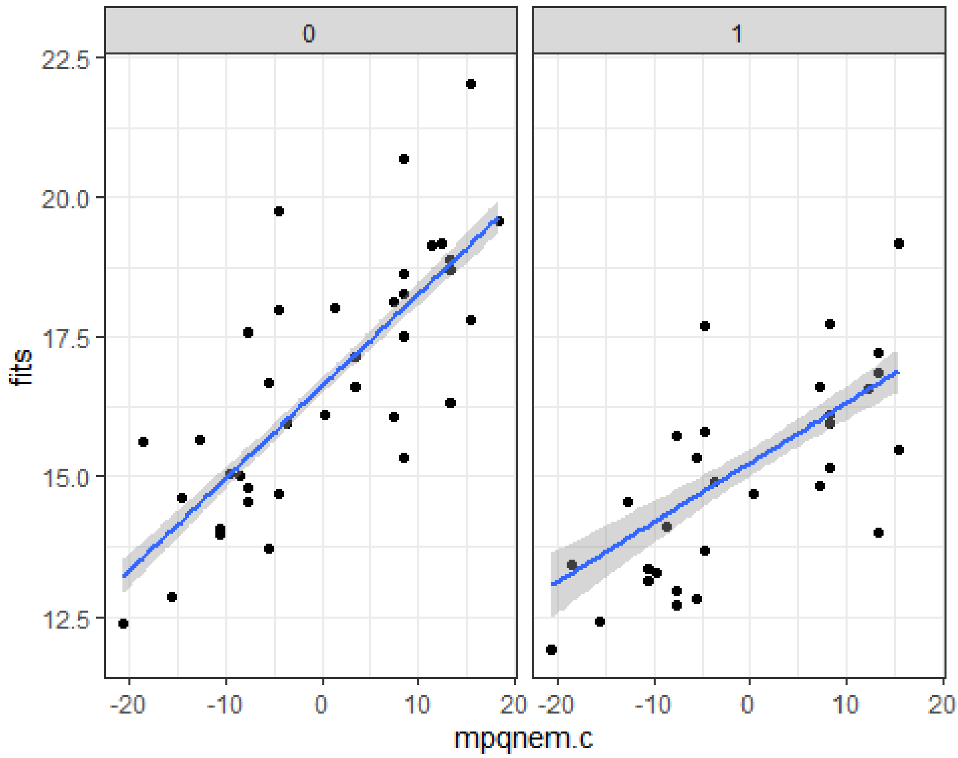
```
musicians$mpqnem.c = musicians$mpqnem - mean(musicians$mpqnem)
performlargeF = as.factor(musicians$performlarge) #helps with the effects plot
summary(model3 <- lmer(na ~ performlargeF*orchtype + performlargeF*mpqnem.c + (perf
ormlargeF | subjnum), data = musicians), corr = FALSE)
Linear mixed model fit by REML ['lmerMod']
Formula:
na ~ performlargeF * orchtype + performlargeF * mpqnem.c + (performlargeF |
```

```
    subjnum)
   Data: musicians

REML criterion at convergence: 2982

Scaled residuals:
   Min      1Q Median      3Q     Max
-2.054 -0.636 -0.158  0.483  4.053

Random effects:
 Groups   Name           Variance Std.Dev. Corr
 subjnum  (Intercept)     3.286   1.813
          performlargeF1  0.557   0.746    -0.38
 Residual                21.811   4.670
Number of obs: 497, groups:  subjnum, 37

Fixed effects:
                        Estimate Std. Error t value
(Intercept)              16.2568     0.5476   29.69
performlargeF1           -1.2348     0.8432   -1.46
orchtype                  1.0007     0.8171    1.22
mpqnem.c                  0.1482     0.0381    3.89
performlargeF1:orchtype  -0.9493     1.1062   -0.86
performlargeF1:mpqnem.c  -0.0302     0.0525   -0.58
plot(allEffects(model3))
```



fitted model 3

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



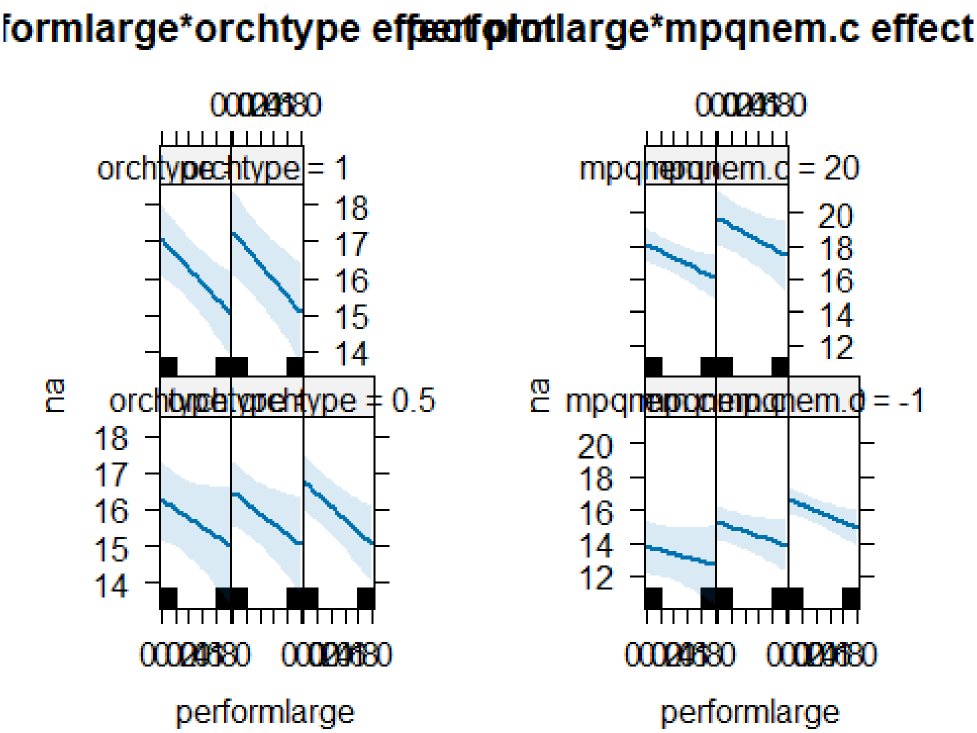
fitted model 3

(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 + (performlarge
| subjnum), data = musicians)
plot(allEffects(model3b))
```



fitted model 3

(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)
orchtype                                  1.693 (0.945)      1.001 (0.817)
performlarge                             -0.911 (0.845)     -1.235 (0.843)
orchtype:performlarge                    -1.424 (1.099)
mpqnem.c                                                     0.148 (0.038)
performlarge:orchtype                                       -0.949 (1.106)
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 + (performlarge |
subjnum), data = musicians)
texreg::screenreg(list(model3b, model3c), digits = 3, single.row = TRUE, stars = 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)
orchtype                                   1.001 (0.817)      1.001 (0.817)
mpqnem.c                                   0.148 (0.038)
performlarge:orchtype                     -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)
Data: musicians
Models:
```

```
model2: na ~ orchtype * performlarge + (1 + performlarge | subjnum)
model3: na ~ performlargeF * orchtype + performlargeF * mpqnem.c + (performlargeF |
subjnum)
       npar  AIC  BIC logLik -2*log(L) Chisq Df Pr(>Chisq)
model2    8 3007 3041  -1496      2991
model3   10 2996 3039  -1488      2976  14.7  2    0.00063 ***
---
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 + (performlargeF | su
bjnum), data = musicians), corr = FALSE)
Linear mixed model fit by REML ['lmerMod']
Formula: na ~ performlargeF * orchtype + mpqnem.c + (performlargeF | subjnum)
   Data: musicians

REML criterion at convergence: 2978

Scaled residuals:
   Min     1Q Median     3Q    Max
-2.040 -0.639 -0.157  0.490  4.063

Random effects:
 Groups   Name           Variance Std.Dev. Corr
 subjnum  (Intercept)     3.241   1.800
          performlargeF1  0.265   0.515    -0.39
 Residual                21.817   4.671
Number of obs: 497, groups:  subjnum, 37

Fixed effects:
                        Estimate Std. Error t value
(Intercept)              16.2130     0.5401   30.02
performlargeF1           -1.2176     0.8316   -1.46
orchtype                  1.0630     0.8066    1.32
mpqnem.c                  0.1392     0.0345    4.03
performlargeF1:orchtype  -1.0297     1.0822   -0.95
anova(model3, model4)
Data: musicians
Models:
```

```
model4: na ~ performlargeF * orchtype + mpqnem.c + (performlargeF | subjnum)
model3: na ~ performlargeF * orchtype + performlargeF * mpqnem.c + (performlargeF |
subjnum)
       npar  AIC  BIC logLik -2*log(L) Chisq Df Pr(>Chisq)
model4    9 2995 3033  -1488      2977
model3   10 2996 3039  -1488      2976  0.35  1       0.55
```

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)
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

Scaled residuals:
   Min     1Q Median     3Q    Max
-2.192 -0.605 -0.112  0.534  3.999

Random effects:
 Groups   Name                       Variance Std.Dev. Corr
 subjnum  (Intercept)                14.4291  3.799
          previous                    0.0707  0.266    -0.65
          audienceJuriedRecital      18.2759  4.275    -0.64 -0.12
          audiencePublicPerformance 12.7839  3.575    -0.83  0.33  0.57
          audienceStudents            8.1899  2.862    -0.63  0.00  0.83  0.66
          solo                        0.7647  0.874    -0.67  0.47  0.20  0.90
 Residual                            15.2859  3.910







  0.49

Number of obs: 497, groups:  subjnum, 37

Fixed effects:
                          Estimate Std. Error t value
(Intercept)                 8.3676     1.9129    4.37
```

```
previous                     -0.1430     0.0625   -2.29
audienceJuriedRecital         4.0735     1.0307    3.95
audiencePublicPerformance     3.0646     0.8923    3.43
audienceStudents              3.6111     0.7675    4.70
solo                          0.5146     1.3964    0.37
mpqpem                       -0.0831     0.0241   -3.45
mpqab                         0.2038     0.0474    4.30
orchtype                      1.5304     0.5836    2.62
mpqnem                        0.1146     0.0359    3.19
solo:mpqnem                   0.0830     0.0416    2.00
optimizer (nloptwrap) convergence code: 0 (OK)
Model failed to converge with max|grad| = 0.0323178 (tol = 0.002, component 1)
```

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

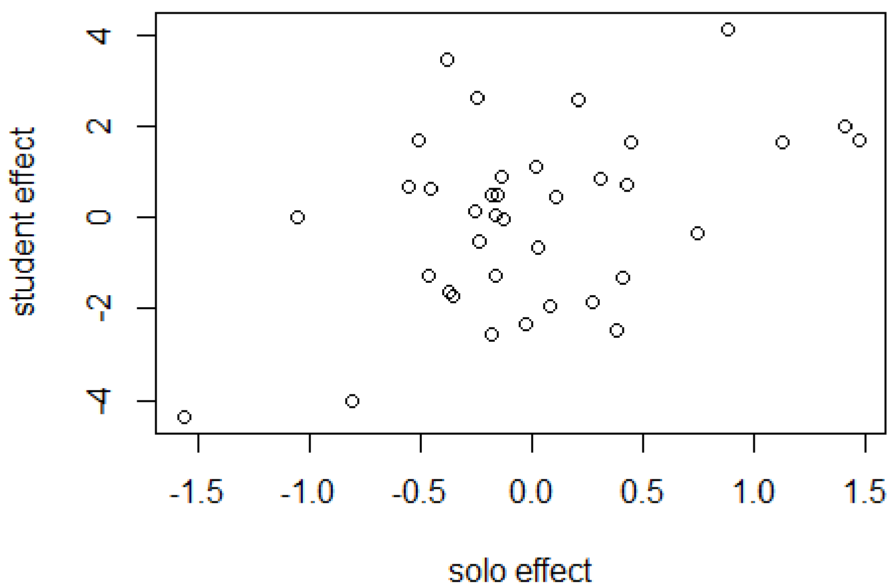
We are entering “audience” as a categorical variable in the “fixed” and “random” inputs. We get 3 coefficients for this four-category variable, and all of them are given random slopes and are included in the covariance estimates.

(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.

ⓘ Code

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



random effects of performance type vs. musician random effects (intercepts)

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    29.69   29.26   0.00
performlargeF1                  -1.23   0.84    -1.46   39.99   0.15
orchtype                         1.00   0.82     1.22   31.37   0.23
mpqnem.c                         0.15   0.04     3.89   30.35   0.00
performlargeF1:orchtype         -0.95   1.11    -0.86   30.45   0.40
performlargeF1:mpqnem.c         -0.03   0.05    -0.58   29.93   0.57
------------------------------------------------------------------

p values calculated using Satterthwaite 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")
#may have to restart R?
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                          16.260***
                                    (0.548)

---------------------------------------------------
Observations                         497
Log Likelihood                  -1,491.000
Akaike Inf. Crit.                3,002.000
Bayesian Inf. Crit.              3,044.000
===================================================
Note:                   *p<0.1; **p<0.05; ***p<0.01
```

(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.