MLM Nested Handout 2 (Spring 2022)

MLM conceptualization, notation and effects

Classroom data example

1. Even though three levels of nesting are present in our classroom data, we now conceptualize MLMs using a simpler model containing an individual-level predictor and effects only for schools (*we will eventually bring classrooms back into the picture*):
   a. This is the model under consideration: $Y_{ijk} = b_0 + b_1 X_{1ijk} + \zeta_k + \varepsilon_{ijk}$; $\zeta_k \sim N(0, \sigma_\zeta^2)$ independent of $\varepsilon_{ijk} \sim N(0, \sigma_\epsilon^2)$.
   b. We now work through some different conceptualizations of the components of this model structure. Each one contributes different insight into what it means to include random effects in a model.

2. Conceptualization #1: random effects are part of a more complex error structure.
   a. Let $\varepsilon'_{ijk} = \zeta_k + \varepsilon_{ijk}$. Then $Y_{ijk} = b_0 + b_1 X_{1ijk} + \varepsilon'_{ijk}$ – the second equation sure looks a lot like 'plain old' regression. But these new *error terms are not independent* across subjects in the same school.
   b. The error term ε'_{ijk} has two parts:
      i. A *structured* component ζ_k (between groups)
      ii. An *unstructured* component ε_{ijk} (within groups)
   c. The indices (subscripts) tell us a lot about what type of variation we are identifying:
      i. ζ_k represents a property of school k ('error' for the school?)
      ii. ε_{ijk} represents error for student i (in classroom j) in school k
   d. Returning to the term $\zeta_k + \varepsilon_{ijk}$, the fact that ζ_k 'repeats' across observations in the same school captures a form of dependence with certain properties:[1]
      i. Covariance between errors from students in the same school:

$$Cov(\varepsilon'_{ijk}, \varepsilon'_{i'j'k}) = Cov(\zeta_k + \varepsilon_{ijk}, \zeta_k + \varepsilon_{i'j'k})$$

$$= Cov(\zeta_k, \zeta_k) + Cov(\varepsilon_{ijk}, \varepsilon_{i'j'k}) + Cov(\zeta_k, \varepsilon_{ijk}) + Cov(\zeta_k, \varepsilon_{i'j'k})$$

$$= Var(\zeta_k) + 0 + 0 + 0 = \sigma_\zeta^2 \text{ (not zero!)}$$

      ii. Note as well that in the same school, $Var(\varepsilon'_{ijk}) = Var(\zeta_k) + Var(\varepsilon_{ijk}) + 2Cov(\zeta_k, \varepsilon_{ijk}) = \sigma_\zeta^2 + \sigma_\varepsilon^2 = Var(\varepsilon'_{i'j'k})$.
         1. The cross-product (covariance) terms in the variance calculation are zero, so it is just the sum of two variances.
      iii. This implies that $Cor(\varepsilon'_{ijk}, \varepsilon'_{i'j'k}) = \frac{\sigma_\zeta^2}{(\sqrt{\sigma_\zeta^2 + \sigma_\varepsilon^2})^2}$, which corresponds to our measure of the proportion of variation that is between schools (ignoring classrooms). It is also the formula for the *Intraclass Correlation Coefficient* (ICC). Reminder: make note of this formula (you have seen it in Handout 1 as well). 1. We have just shown that – net of the mean, so holding the predictors constant – the ICC is the correlation between a random pair of observations within the same school (k), induced by our model.
   e. Identification: we can learn about ζ_k for a specific school k, and identify the effect, by looking for systematic, *mean differences in level* for all students in that school. **HERE IS A WAY TO CONCEPTUALIZE THIS**: Imagine that you ran OLS regression on your outcome, ignoring

---

[1] We will make use of these definitions/identities: Cov(X,Y) = E(XY)- E(X)E(Y); Cor(X,Y) = Cov(X,Y)/sqrt(V(X))sqrt(V(Y)); Var(X+Y)=Var(X)+Var(Y)+2Cov(X,Y).

the nested nature of your data. If, for some schools, the structured error terms $\{\varepsilon'_{ijk}\}_{\cdot k}$ (there are many of these; this is a notation for all errors (including the structured part) for school k, as we have defined them) are typically 'high,' while for other schools they are typically low, then we can associate those high or low levels with each school through component ζ_k.

    i. We will do this, constructively, in a section that follows.

3. Conceptualization <mark>#2: effects are latent variables</mark>
   a. Some insight can be gained if one thinks of ζ_k as a latent variable, which is a fancy way of saying the combined effect of unobserved characteristics.
      i. We sometimes say that observations Y_{ijk}, all in school k, share a latent trait ζ_k that reflects something about the school.
      ii. Observations are driven by the (structured) error, so if the errors are driven by a latent trait, then so are the observations.
   b. This can be characterized (for the structured errors) by a path diagram (for the same school k):
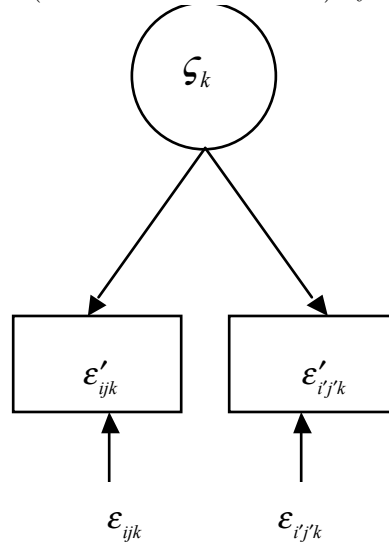
This is really structured error under a simple factor analysis model (unidimensional)

    i. The latent trait ζ_k is shared by structured error terms, $\varepsilon'_{ijk}, \varepsilon'_{i'j'k}$, and thus outcomes, $Y_{ijk}, Y_{i'j'k}$.

4. Conceptualization <mark>#3: random coefficients</mark>
   a. We can regroup the terms in the complex error:
      i. Starting with $Y_{ijk} = b_0 + b_1 X_{1ijk} + \varepsilon'_{ijk}$, we substitute in the expression for the error: $Y_{ijk} = b_0 + b_1 X_{1ijk} + (\zeta_k + \varepsilon_{ijk})$
      ii. And now regroup: $Y_{ijk} = (b_0 + \zeta_k) + b_1 X_{1ijk} + \varepsilon_{ijk}$.
      iii. And finally, rename & re-index: $\underline{Y_{ijk} = (\beta_{0k}) + b_1 X_{1ijk} + \varepsilon_{ijk}}$, where $\beta_{0k} = b_0 + \zeta_k$.
      iv. The term $\beta_{0k} = b_0 + \zeta_k$ is a 'random coefficient' (or random intercept) that varies by school, not by student i in school, or even by classroom (yet).
         1. Every student in school k has regression intercept β_{0k}, not just plain old b_0. Since k can vary across all possible school indices, this yields many school-specific intercepts.
         2. This is not the same as running separate regressions for each school. *(crucial to understand why)*
   b. There are different ways to describe this formulation: *random coefficients, random regression*. It is also closely aligned with 'hierarchical' or 'level' notation, to be introduced next.

5. Conceptualization <mark>#4, separate equations</mark> for different (nested) levels.
   a. School data is naturally hierarchical, and we have already introduced predictors that vary at different "levels" of the nested data structure. We now make this explicit through the use of notation.

b. First, we copy the re-indexed version of our model presented in Section 4, and call this "level 1":

    i. Level 1 is the topmost level and terms that are listed should require the maximal number of subscripts. With nested data, it is usually the subject level, as in outcome Y_{ijk}.

    ii. Level 1: $Y_{ijk} = \beta_{0k} + \beta_{1k} X_{1ijk} + \varepsilon_{ijk}$, $\varepsilon_{ijk} \sim N(0, \sigma_\varepsilon^2)$

    iii. There are many terms in this equation that vary by school: the outcome, the predictor, the error, the intercept and the slope on the predictor. That's right, the *intercept and slope are school-specific.* If we assign a distribution to the intercept term (only) such that it is based on a random variable, such as a draw from a normal distribution, we call this a *random intercepts model.*

    iv. We further specify terms such as β_{0k} at 'lower' levels, in this case level 2 will be used.

    v. Level 2: $\beta_{0k} = b_0 + \zeta_{0k}$, $\zeta_{0k} \sim N(0, \sigma_{\zeta_0}^2)$, and $\beta_{1k} = b_1$, where $\varepsilon_{ijk}, \zeta_{0k}$ are assumed independent. The '0' in the subscript allows for naming additional random effects; e.g., ζ_{1k} or ζ_{2k}.

    vi. Each of these equations can be made more or less complex. For example, we can remove the level 2 error term, ζ_{0k} (yes, this is an error term, but a *structured* or group-level error term), and then we have regression, ignoring any correlation structure in the errors:

        1. Level 1: $Y_{ijk} = \beta_{0k} + \beta_{1k} X_{1ijk} + \varepsilon_{ijk}$, $\varepsilon_{ijk} \sim N(0, \sigma_\varepsilon^2)$

        2. Level 2: $\beta_{0k} = b_0$; $\beta_{1k} = b_1$

c. Why go to so much trouble, notationally?

    i. The levels are organized directly around the nesting principle (an example follows soon).

    ii. There are "rules" associated with what can be included (predictors and error terms) at each level, ensuring some quality control (fewer coding/conceptualizing mistakes).

    iii. There is a sense that each equation is a *mini-regression*, and as such, one can think about explaining variation at different levels with different equations.

    iv. The notation is extremely common in much of the literature.

    v. Two software packages, HLM and MLwiN are based on this organizing principle.

    vi. Moving between this notation and the alternative, used by STATA/SAS/R simply involves algebraic substitution.

A constructive basis for random effects

1. While we will not directly estimate effects ζ_k for each school k, we can gather evidence that the information about them exists in our data.

a. Recall conceptualization #1: according to that model, the errors from a regression look like this: $\varepsilon'_{ijk} = \zeta_k + \varepsilon_{ijk}$. We can't easily separate the two components (yet), but we can look at their sum for any grouping.

    i. In regular regression, one examines residuals to evaluate potential omitted variables. For example, one can produce boxplots by an omitted categorical variable. When there are no omitted predictors, these groups should have roughly the same mean and variance.

b. Models that fail to include "effects" for schools essentially omit a variable, and this can be detected in a residual analysis.

c. Below, we run a simple OLS regression of MATHKIND on SES, generate the residuals and then use a boxplot by school to see whether school effects strong enough that we notice their omission.

```
require(lme4)
```

```
## Loading required package: lme4
```

```
## Loading required package: Matrix
```

```
require(lmerTest)
```

```
## Loading required package: lmerTest
```

```
##
## Attaching package: 'lmerTest'
```

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

```
##
##     lmer

## The following object is masked from 'package:stats':
##
##     step
```
```r
dat <- read.csv("../Datasets/classroom.csv")
# simple regression
fit1 <- lm(mathkind ~ ses, data = dat)
print(summary(fit1))
```
```
##
## Call:
## lm(formula = mathkind ~ ses, data = dat)
##
## Residuals:
##       Min      1Q   Median       3Q      Max
## -174.835  -26.399    1.065   26.529  161.150
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  466.845      1.162  401.70   <2e-16 ***
## ses           14.359      1.568    9.16   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 40.08 on 1188 degrees of freedom
## Multiple R-squared:  0.06596,    Adjusted R-squared:  0.06518
## F-statistic:  83.9 on 1 and 1188 DF,  p-value: < 2.2e-16
```
```r
dat$res1 <- residuals(fit1)
if (vanillaR) {
    boxplot(split(dat$res1, dat$schoolid))
} else {
    ggplot(dat, aes(x = factor(schoolid), y = res1)) + geom_boxplot()
}
```

Residual mean by school is not zero

res1

factor(schoolid)
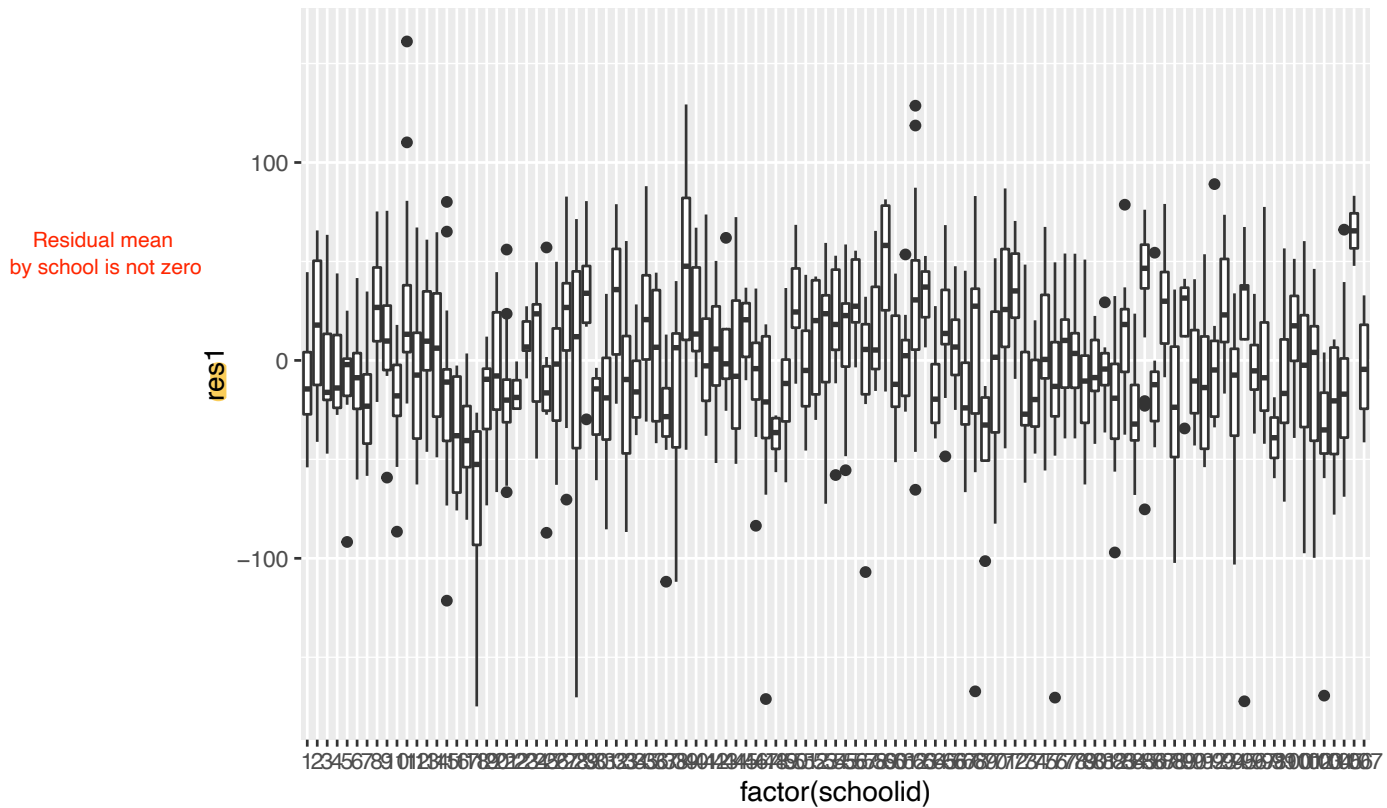
c. It is a bit hard to 'see' what the average shift up or down would be at the school level. Maybe a boxplot will reveal some of the structure (this time, we generate residuals directly):

  i. We do see a fair amount of variation in the level, represented by the median residual for each school.
  ii. If the errors were dependent errors, we'd see structure (and we do). This observation is crucial: *if the errors were independent, the medians would be much closer to zero for every school.*[2]

d. Let's take the average of these residuals at the school level and use them as a first approximation—an estimate of ζ_k.

 i. This code generates a school-level mean residual:

```
res1mean <- tapply(dat$res1, dat$schoolid, mean, na.rm = T)
```

We visualize these average residuals by case number in the reduced file (this x-axis is meaningless, but useful to spread the observations apart).

```
plot(res1mean)
abline(h = 0, col = 8)
```

---

[2] You can calculate the s.d. of randomly distributed errors when $\sigma_\zeta^2 = 0$ using the Central Limit Theorem, σ_ε^2, and group size.
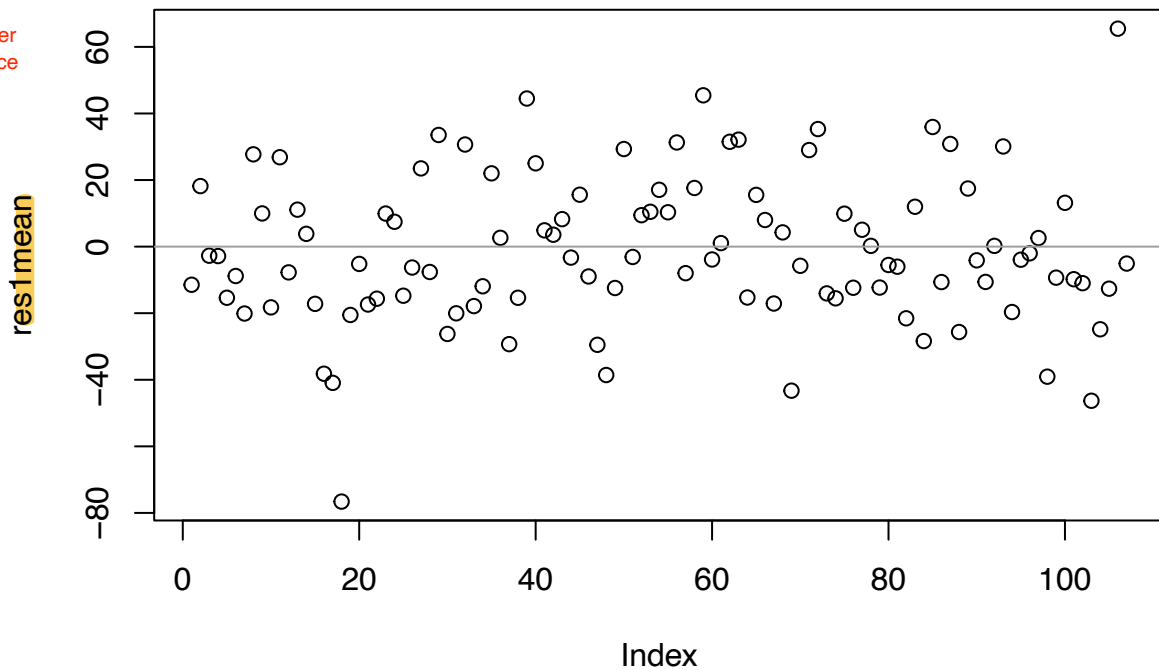
2. (cont.)

    d. (cont.)

        ii. The main point of this plot is that different schools have different mean residuals. *There is only one data point per school,* so they do not represent variance, the data points are averages of residuals. Also note that the x-axis is arbitrary - it represents school ids.

    e. These estimates of ζ_{0k} have a distribution (as they vary by school), which we can examine with density plots and by calculating the standard deviation (square root of variance).

```
plot(density(res1mean))
```

density.default(x = res1mean)

N = 107   Bandwidth = 7.085
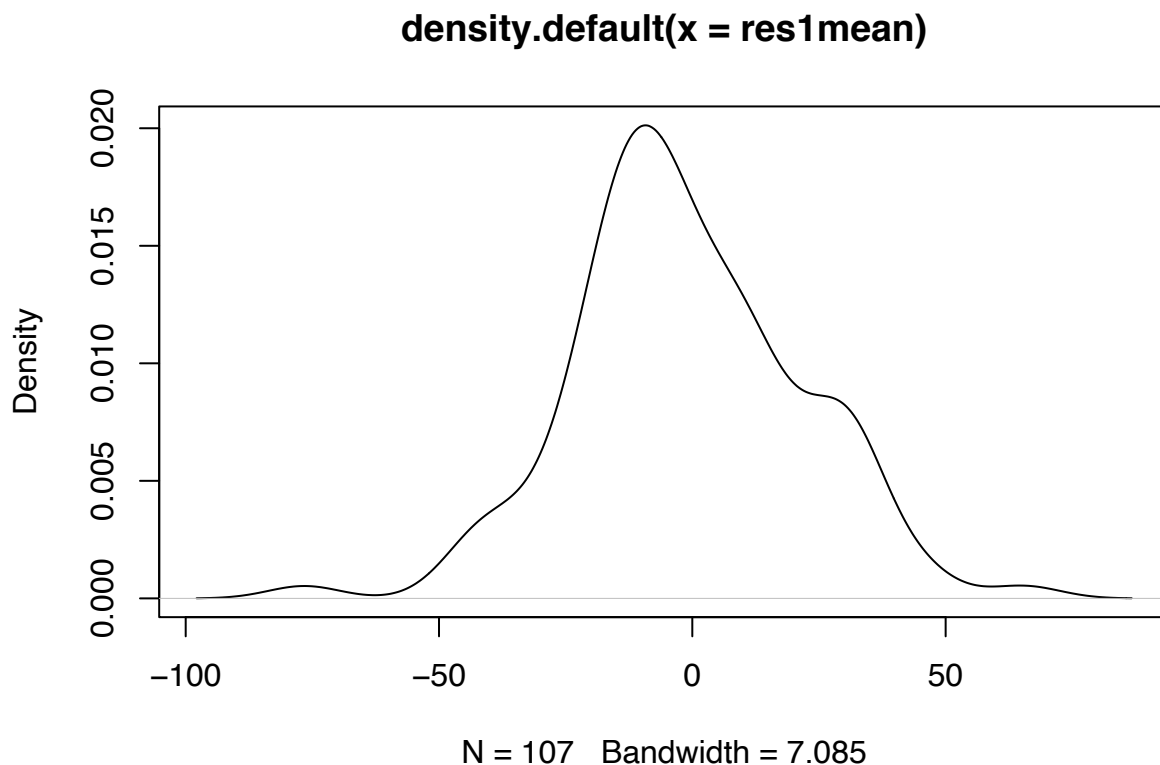
e. (cont.)
   i. The standard deviation is pretty large at 22.43
   ii. Note that because we are using R, the vector **res1mean** contains only 107 means, one for each school. If we wish to *weight* by the number of students sampled in each school, we could do it in several ways in R, but let's try using the weights option of the density function.

```
wts <- as.numeric(prop.table(table(dat$schoolid)))
plot(density(res1mean, weights = wts))  #relies on the ordering of res1mean and table() by schoolid
```

density.default(x = res1mean, weights = wts)

N = 107   Bandwidth = 7.085
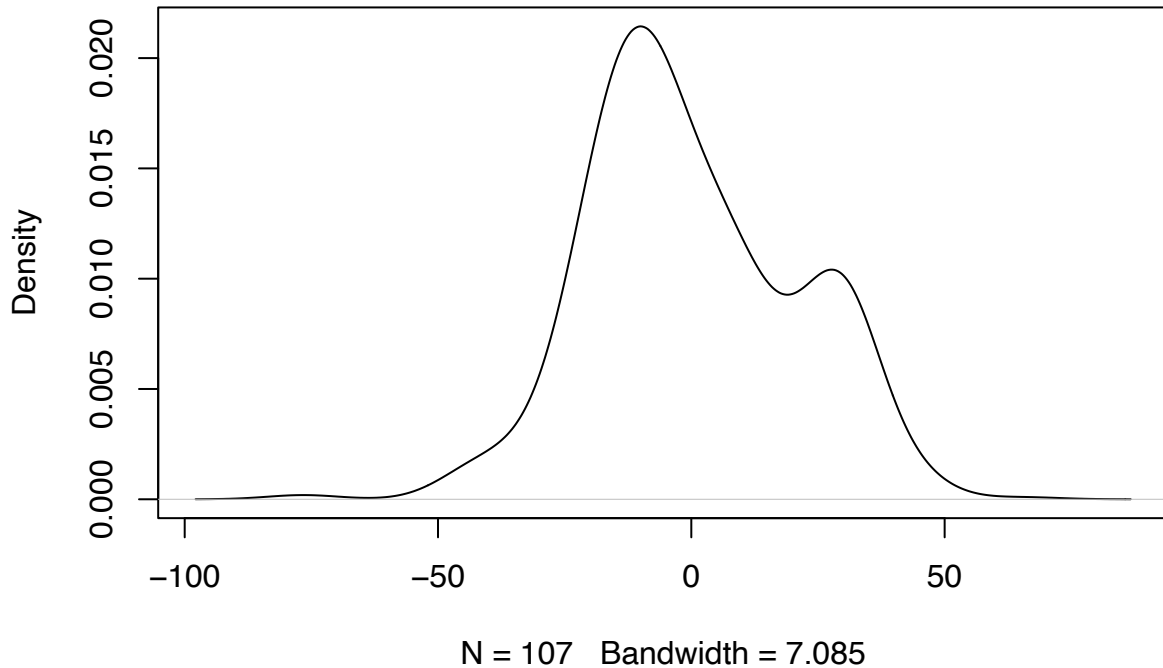
e. (cont.)
  iii. And you can see a bimodality you get from the individual student perspective, but this isn't really how we conceptualize the data generation.
f. We can think of average residuals as a 'crude' estimate of the structured, school-level error component ζ_k. Or at least evidence that schools differ in a consistent (structured) manner.
  i. The distribution was reasonably symmetric. We computed the std. dev. of these terms, which was about 20 or 22. This is a crude estimate of the variance component σ_ζ^2. It tends to *overestimate* the variance - but understanding this will require more work.
g. We now run a mixed model to derive a better estimate of the variance component (shortly, we'll see why it's 'better')

```
lmer.fit1 <- lmer(mathkind ~ ses + (1 | schoolid), data = dat)
summary(lmer.fit1)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: mathkind ~ ses + (1 | schoolid)
##    Data: dat
##
## REML criterion at convergence: 12038.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.7423 -0.5716  0.0188  0.6343  3.7763
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  schoolid (Intercept) 309      17.58         est. of sig^2_zeta
##  Residual             1308     36.16
## Number of obs: 1190, groups:  schoolid, 107
```

8

```
##
## Fixed effects:
##              Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)  465.746      2.058   98.752 226.337  < 2e-16 ***
## ses           10.722      1.589 1187.995   6.747 2.35e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##     (Intr)
## ses 0.034
```

g. (cont.)

   i. The variance component estimate (as s.d.) is $\hat{\sigma}_\zeta = 17.6$, which is a bit smaller than our prior estimates.

3. Adding in a second level of nesting (classroom level)

a. Running a mixed effects model is the proper approach in many nested data applications. We now run a mixed model with three levels.

```
fit3 <- lmer(mathkind ~ ses + (1 | schoolid/classid), data = dat)
print(summary(fit3))
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: mathkind ~ ses + (1 | schoolid/classid)
##    Data: dat
##
## REML criterion at convergence: 12038.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.7444 -0.5722  0.0112  0.6320  3.7507
##
## Random effects:
##  Groups          Name        Variance Std.Dev.
##  classid:schoolid (Intercept)   20.44    4.521       est. of sig^2_eta
##  schoolid        (Intercept)  301.52   17.364       est. of sig^2_zeta
##  Residual                    1294.08   35.973       est. of sig^2_epsilon
## Number of obs: 1190, groups:  classid:schoolid, 312; schoolid, 107
##
## Fixed effects:
##              Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)  465.749      2.058   99.030 226.344  < 2e-16 ***
## ses           10.684      1.589 1187.053   6.725 2.72e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##     (Intr)
## ses 0.035
```

b. These two variance components are estimated simultaneously along with the within subject error variance σ_ϵ^2.

Model Selection

1. Comparing nested models
   a. A common way to determine what components of a model are necessary (fixed effects or random effects), from a statistical perspective, is to conduct a significance test on some subset of the model's parameters. The test we have already introduced and will use extensively is a likelihood ratio test (LRT or LR test).
      i. In this setting, parameters are added to an existing model, establishing a "nesting" of one model in the other that facilitates the LR test.
   b. When all of the parameters considered are regression coefficients (fixed effects), then the tests are for statistical differences from zero, either singly, or as a vector. T-tests of single coefficents and Wald tests for blocks of coefficients may be used.

2. Selecting random effects (first pass)
   a. A natural consideration in building mixed effects models is the presence and complexity of the random effects.
   b. The very first question to ask is whether *any* random effects are needed. We will use the classroom data, beginning with a model for MATHGAIN with no covariates: $MATHGAIN_{ijk} = b_0 + \zeta_k + \varepsilon_{ijk}$, with $\zeta_k \sim N(0, \sigma_\zeta^2)$ and $\varepsilon_{ijk} \sim N(0, \sigma_\varepsilon^2)$ independently of one another. Recall that this is called the *unconditional means model*. The 'extra' subscripts are intentionally included, as they will become important as we add random effects for classrooms.
      i. We want to use an LRT to compare a model with no random effects to one with random effects for schools, but it can be challenging to fit a model *with no random effects* (yes, this is plain old regression, but we need to be able to compare likelihoods, and that isn't set up (in a simple manner) under R's lm and lmer functions, so one way around this is to use the function `ranova` in the lmerTest library in R). `ranova` makes this possible by listing what is *deleted* and then computing an LRT.

```
lmer.fit2 <- lmer(mathgain ~ (1 | schoolid), data = dat)
ranova(lmer.fit2)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## mathgain ~ (1 | schoolid)
##                  npar  logLik   AIC    LRT Df Pr(>Chisq)
## <none>              3 -5888.3 11783
## (1 | schoolid)      2 -5904.8 11814 32.881  1  9.796e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

   c. A more interesting comparison would be whether we need to add a random classroom effect, *after controlling for school effects*, and this is an LRT.
      i. If we add random classroom effects to the last model and then compare likelihoods with an LRT, we have an answer to this question.
      ii. The model to examine is $MATHGAIN_{ijk} = b_0 + \eta_{jk} + \zeta_k + \varepsilon_{ijk}$, with the usual normality and independence assumptions on all random terms.

```
lmer.fit3 <- lmer(mathgain ~ (1 | schoolid/classid), data = dat)
anova(lmer.fit2, lmer.fit3, refit = F)
```

```
## Data: dat
## Models:
## lmer.fit2: mathgain ~ (1 | schoolid)
## lmer.fit3: mathgain ~ (1 | schoolid/classid)
##           npar   AIC   BIC  logLik deviance  Chisq Df Pr(>Chisq)
## lmer.fit2    3 11783 11798 -5888.3    11777
```

```
## lmer.fit3    4 11777 11797 -5884.4     11769 7.9048  1     0.00493 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

  2. (cont.)
      c. (cont.)
          ii. (cont.)
              1. We used the `anova` command, which does a sequential LRT based on the listed model fits.
              2. The chi-square test, on one degree of freedom (corresponding to the additional variance component) is highly significant, with p<0.01. Conclusion: inclusion of classroom effects is warranted.
              3. Question: can we use the LRT given by ranova() applied to the more complex model? Answer: That function divises two LR tests; it is likely that we can find the test that we require (see R code that follows). *Look at what was deleted.*

```
ranova(lmer.fit3)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## mathgain ~ (1 | classid:schoolid) + (1 | schoolid)
##                       npar  logLik   AIC    LRT Df Pr(>Chisq)
## <none>                   4 -5884.4 11777
## (1 | classid:schoolid)   3 -5888.3 11783 7.9048  1   0.004930 **    drops classroom effects
## (1 | schoolid)           3 -5888.3 11783 7.8875  1   0.004978 **    drops school effects
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

  2. (cont.)
      c. (cont.)
          ii. (cont.) 4. Question: *I thought we found, previously, that the classroom effects were not needed?* **Why does this result differ?**   In contrast to Handout 1

3. LRTs for mixed effects models: comparing fixed effects parameters.
    a. Many mixed effects models are fit using 'restricted' maximum likelihood (REML). This is the default in R.
        i. Researchers prefer REML over what is sometimes referred to as "full" ML because the estimation of *variance components* is unbiased using REML.
        ii. Unfortunately, we cannot use LR tests to compare the fixed effects of models fit under REML (there is another way, to be given soon).
            1. If you try to use REML fits to compare fixed effects, R will refit the model using ML.
    b. When the comparison of interest involves the fixed effects, you can force the fit to use ML (in R you let the functions that compare models call the 'refitting' procedure).
        i. We normally do not compare models that change both fixed and random effects at the same time, but it is possible under ML (not REML).
    c. Note: if you are evaluating the inclusion of a single, continuous predictor to a model, you can just look at the p-value (fixed effects output).
        i. Here, we explore the assessment of 'blocks' of predictors.
    d. Example: let's add SES and (teacher-level) MATHKNOW to our model as a 'block' and evaluate whether either is significant (the null hypothesis is that the regression coefficients for both are identically zero).
        i. The new model is: $MATHGAIN_{ijk} = b_0 + b_1 SES_{ijk} + b_2 MATHKNOW_{jk} + \eta_{jk} + \zeta_k + \varepsilon_{ijk}$, with the usual normality and independence assumptions on all random terms, and will be compared to the model fit in the prior section.
        ii. We first do this naively, using REML:

```
lmer.fit4 <- lmer(mathgain ~ ses + mathknow + (1 | schoolid/classid), data = dat)
summary(lmer.fit4)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: mathgain ~ ses + mathknow + (1 | schoolid/classid)
##    Data: dat
##
## REML criterion at convergence: 10684.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.3702 -0.6106 -0.0342  0.5357  5.5805
##
## Random effects:
##  Groups            Name        Variance Std.Dev.
##  classid:schoolid (Intercept)  104.64   10.229
##  schoolid         (Intercept)   92.11    9.598
##  Residual                     1018.67   31.917
## Number of obs: 1081, groups:  classid:schoolid, 285; schoolid, 105
##
## Fixed effects:
##             Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)  57.8169     1.5468   91.6576  37.379   <2e-16 ***
## ses           0.8847     1.4602 1039.2449   0.606   0.5447
## mathknow      2.4341     1.3093  211.2083   1.859   0.0644 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) ses
## ses       0.039
## mathknow  0.009 -0.042
```

```
print(try(anova(lmer.fit3, lmer.fit4, refit = F)))
```

```
## Error in anova.merMod(lmer.fit3, lmer.fit4, refit = F) :
##   models were not all fitted to the same size of dataset
## [1] "Error in anova.merMod(lmer.fit3, lmer.fit4, refit = F) : \n  models were not all fitted to
## attr(,"class")
## [1] "try-error"
## attr(,"condition")
## <simpleError in anova.merMod(lmer.fit3, lmer.fit4, refit = F): models were not all fitted to the
```

d. (cont.)

   iii. It doesn't quite look like SES or MATHKNOW is significant, but we don't really know if they are jointly non-significant until we do the proper test in the `anova`, but ...

   iv. The test we thought we could do 'fails' *for a different reason*: the problem occurred because in *fitting a model with more predictors*, we lost nine cases due to missingness. **LRTs are only legitimate when the included observations are the same in both models.**

     1. To remedy the situation, we can choose a subsample of observations from the dataset before performing the model estimation. We can identify which observations were included in the model by establishing an indicator variable ('in_sample') with that information:

```
save.options <- options()   #so that old options can be restored
options(na.action = "na.pass")
mm <- model.matrix(~mathgain + ses + mathknow, data = dat)
```

```
in_sample <- apply(is.na(mm), 1, sum) == 0  # these rows aren't missing anything
options(save.options)
# re-fit mlms using only fully observed observations, even in model with fewer
# predictors.
fit7 <- lmer(mathgain ~ 1 + (1 | schoolid/classid), data = dat, subset = in_sample)
print(summary(fit7))
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: mathgain ~ 1 + (1 | schoolid/classid)
##    Data: dat
##  Subset: in_sample
##
## REML criterion at convergence: 10693.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.3056 -0.6115 -0.0396  0.5509  5.6208
##
## Random effects:
##  Groups            Name        Variance Std.Dev.
##  classid:schoolid (Intercept)  116.9    10.813
##  schoolid         (Intercept)   73.4     8.567
##  Residual                     1021.0    31.953
## Number of obs: 1081, groups:  classid:schoolid, 285; schoolid, 105
##
## Fixed effects:
##             Estimate Std. Error     df t value Pr(>|t|)
## (Intercept)   57.761      1.499 95.000   38.54   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
fit8 <- lmer(mathgain ~ ses + mathknow + (1 | schoolid/classid), data = dat, subset = in_sample)
print(summary(fit8))
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: mathgain ~ ses + mathknow + (1 | schoolid/classid)
##    Data: dat
##  Subset: in_sample
##
## REML criterion at convergence: 10684.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.3702 -0.6106 -0.0342  0.5357  5.5805
##
## Random effects:
##  Groups            Name        Variance Std.Dev.
##  classid:schoolid (Intercept)  104.64   10.229
##  schoolid         (Intercept)   92.11    9.598
##  Residual                     1018.67   31.917
## Number of obs: 1081, groups:  classid:schoolid, 285; schoolid, 105
##
## Fixed effects:
```

```
##            Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)  57.8169     1.5468   91.6576  37.379   <2e-16 ***
## ses           0.8847     1.4602 1039.2449   0.606   0.5447
## mathknow      2.4341     1.3093  211.2083   1.859   0.0644 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) ses
## ses       0.039
## mathknow  0.009 -0.042
```

```
print(anova(fit7, fit8, refit = T))  #force ML fit.
```

```
## refitting model(s) with ML (instead of REML)
## Data: dat
## Subset: in_sample
## Models:
## fit7: mathgain ~ 1 + (1 | schoolid/classid)
## fit8: mathgain ~ ses + mathknow + (1 | schoolid/classid)
##      npar   AIC   BIC  logLik deviance  Chisq Df Pr(>Chisq)
## fit7    4 10704 10724 -5347.9    10696
## fit8    6 10704 10734 -5346.1    10692 3.5921  2      0.166
```

  d. (cont.)

     v. In this set of runs, we 'refit' with ML to properly compare the fixed effects as a block.

    vi. The new predictors are found to be jointly non-significant, p>0.05.

   vii. In general, *when fitting a series of sequential models, it is best if you subset the observations to reflect the 'last' (or largest) model's non-missing predictors.*