MLM Nested Handout 1 (Spring 2022)

Nested grouping structure

Introduction

1. In certain situations, data are naturally nested in some manner. The classic example is students within classrooms within schools. *Why can't we simply use regression?*

   i. The independence of errors assumption is unlikely to hold. This may bias the s.e. of the regression coefficients, and the significance of your findings relies on the accuracy of the s.e.

   ii. We want to 'control' for group effects, somehow. Failure to do so could result in omitted variable bias.

   iii. We are actually interested in quantifying the manner in which groups in our sample differ from one another.

      a. Differences can be in level or response to predictor. E.g. hospital-specific relationship of mortality to patient SES

      b. But... including group indicators and/or group interaction terms is not feasible (*what is one reason that this would be true?*)

2. Example: classroom.csv. This dataset has three levels of nesting: schools, classrooms within schools and students within those classrooms.

```
dat <- read.csv("../Datasets/classroom.csv")
summarizeGroup <- function(x) {
    idCt <- table(x)  #counts of unique ids
    return(list(UniqGroups = length(idCt), minPerGroup = min(idCt), meanPerGroup = mean(idCt),
        maxPerGroup = max(idCt)))
}
t(sapply(dat[, c("schoolid", "classid")], summarizeGroup))  #assuming unique ids across levels of
```

```
##          UniqGroups minPerGroup meanPerGroup maxPerGroup
## schoolid 107        2           11.1215      31
## classid  312        1           3.814103     10
```

   i. In this sample, there are 107 schools, a total of 312 classrooms across all schools, and 1190 students total. Within a school, there are between 2 and 31 students sampled

      a. Actually, classrooms are sampled first at each school, and within a classroom, between 1 and 10 students are studied.

```
head(dat)
```

```
##   sex minority mathkind mathgain   ses yearstea mathknow housepov mathprep
## 1   1        1      448       32  0.46        1       NA    0.082     2.00
## 2   0        1      460      109 -0.27        1       NA    0.082     2.00
## 3   1        1      511       56 -0.03        1       NA    0.082     2.00
## 4   0        1      449       83 -0.38        2    -0.11    0.082     3.25
## 5   0        1      425       53 -0.03        2    -0.11    0.082     3.25
## 6   1        1      450       65  0.76        2    -0.11    0.082     3.25
##   classid schoolid childid
## 1     160        1       1
## 2      160         1         2
## 3      160         1         3
## 4      217         1         4
## 5      217         1         5
## 6      217         1         6
```

ii. Collected at the **student level**: some basic demographics (sex (1=female), minority (0/1), ses); outcomes (math score in spring of kindergarten, called mathkind, and the increase in math score from spring of kindergarten to spring of first grade, called mathgain).

iii. At the **classroom level**, we have information on the teacher (years teaching, math knowledge, math preparation (# courses))

iv. At the **school level**, we have the average household poverty.

3. Say we want to understand the relationship between math scores in kindergarten and ses:

   i. Regression approach: pool the data; ignore nesting. What is the overall slope?

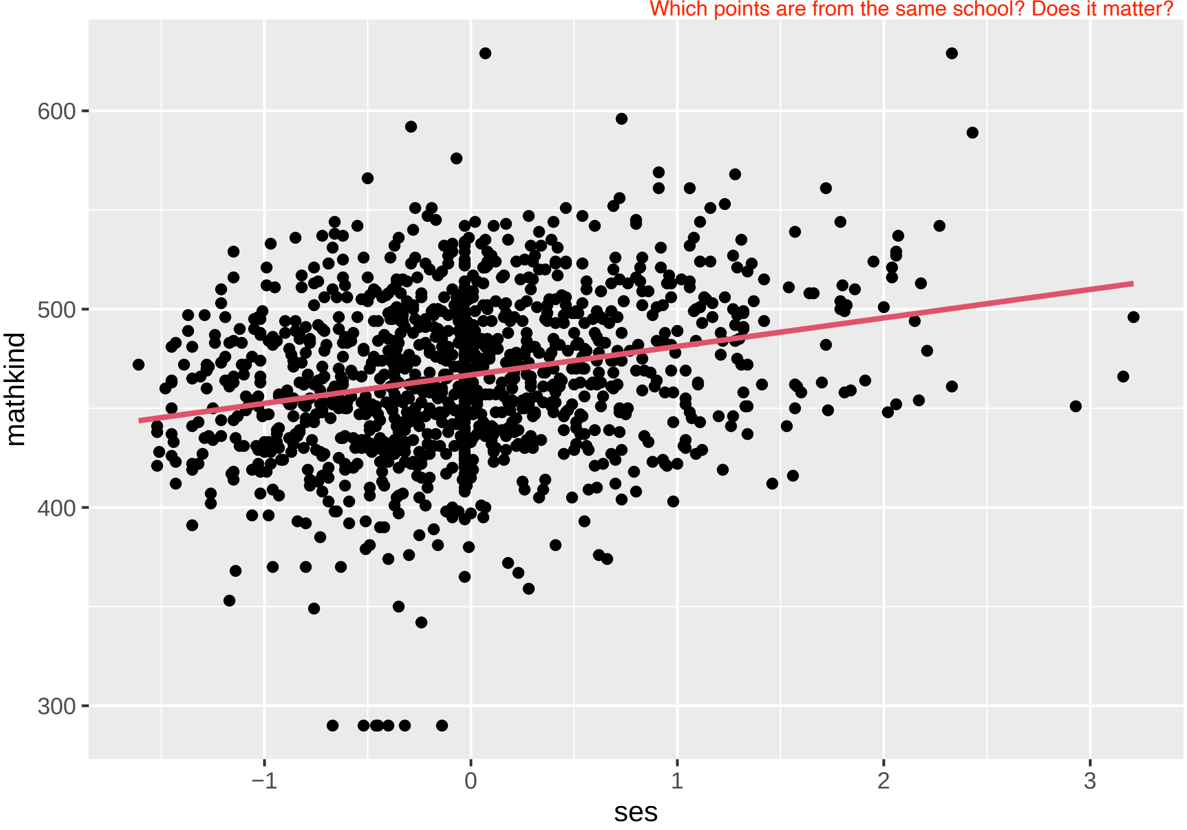
   ii. Graphically:

Completely Pooled Approach (described in G&H)

```
if (vanillaR) {
    plot(dat$ses, dat$mathkind)
    abline(lsfit(dat$ses, dat$mathkind), col = 2)
} else {
    ggplot(dat, aes(x = ses, y = mathkind)) + geom_point() + geom_smooth(method = "lm",
        se = FALSE, col = 2)
}
```

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

Which points are from the same school? Does it matter?



```
if (liveDemo) {
    ggplot(dat, aes(x = ses, y = mathkind, color = schoolid)) + geom_point() + geom_smooth(method =
        se = FALSE, col = 2)  #+theme(legend.position = 'none')
}
```

iii. Numerically:
   a. Implicit: weight each student equally; assume independent observations
   b. Comparing two students differing by one unit of SES, we expect or predict a 14.4 unit difference in MATHKIND on average.

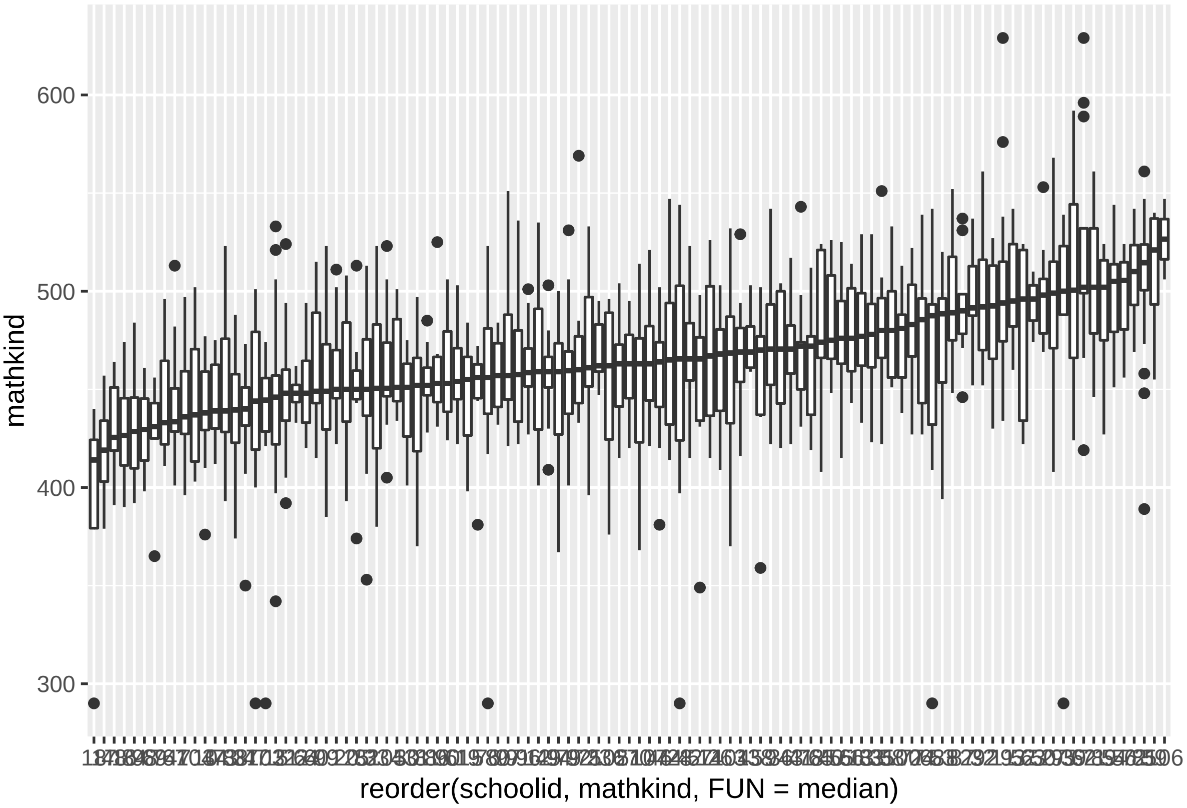
```
fit1 <- lm(mathkind ~ ses, data = dat)
print(summary(fit1))
```

```
##
## 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
```

iv. We now examine the independence assumption with a simple descriptive exploration, the distribution of mathkind within schools using a stacked boxplot (x-axis is schoolid)
   a. We see some differences, at least in the median within school (qtn: why might schools differ?)
   b. Why does this indicate non-independence *within* school? HINT: what would independence look like?

```
if (vanillaR) {
    ord <- order(unlist(tapply(dat$mathkind, dat$schoolid, median)))
    boxplot(split(dat$mathkind, dat$schoolid)[ord])
} else {
    ggplot(dat, aes(x = reorder(schoolid, mathkind, FUN = median), y = mathkind)) +
        geom_boxplot()
}
```

cf. Jamboard p. 10



v. Another way to examine dependence is to take a random draw of pairs of subjects within each school (repeating for all schools) and correlate the outcomes. If these have a significant correlation, then they are dependent.

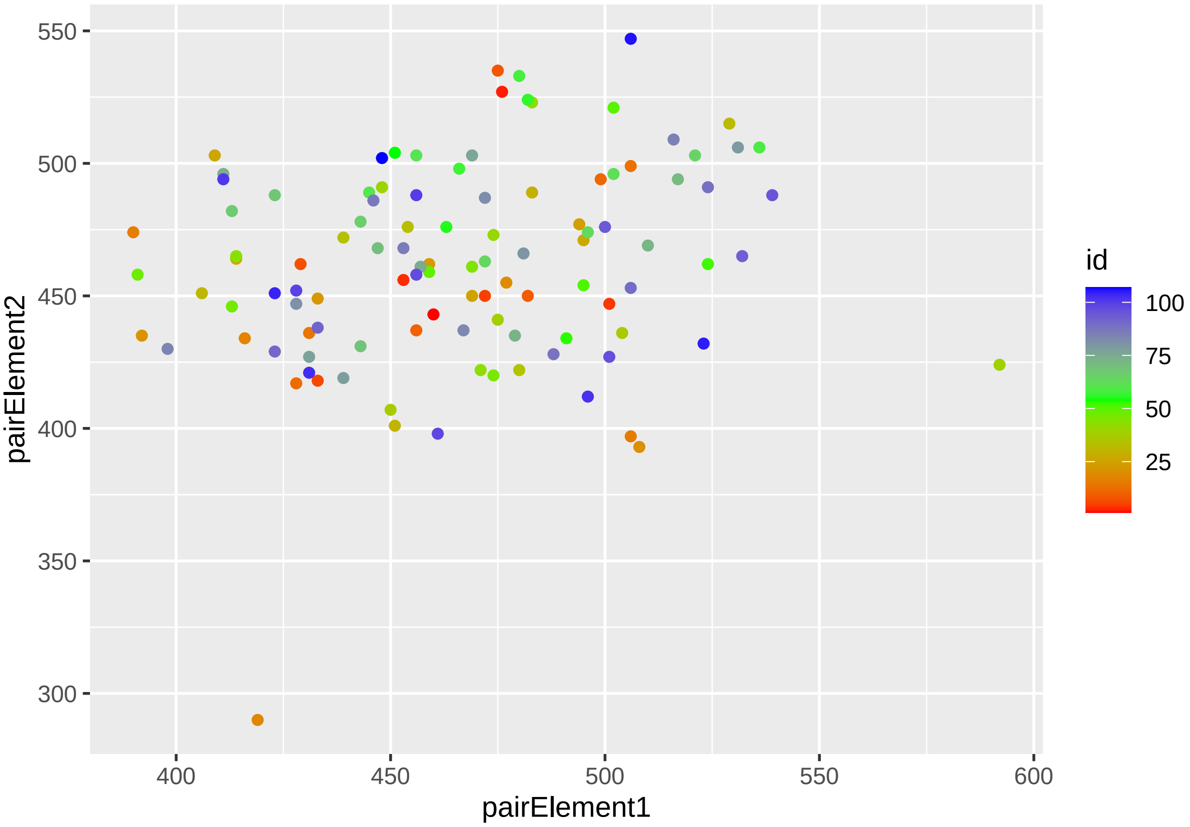
   a. When we do this, we get a correlation of about 0.2~~3~~.

```r
# draw paired sample from schools for correlation analysis:
set.seed(2042001)
x <- matrix(unlist(tapply(dat$mathkind, dat$schoolid, sample, size = 2)), ncol = 2,
    byrow = T)
cor(x)[1, 2]
```

```
## [1] 0.2080421
```

```r
if (vanillaR) {
    plot(x, col = rainbow(dim(x)[1]), pch = 16)
} else {
    x.df <- data.frame(id = row(x)[, 1], pairElement1 = x[, 1], pairElement2 = x[,
        2])
    ggplot(x.df, aes(x = pairElement1, y = pairElement2, col = id)) + geom_point() +
        scale_color_gradientn(colours = rainbow(length(x.df)))
}
```

see Jamboard p. 9



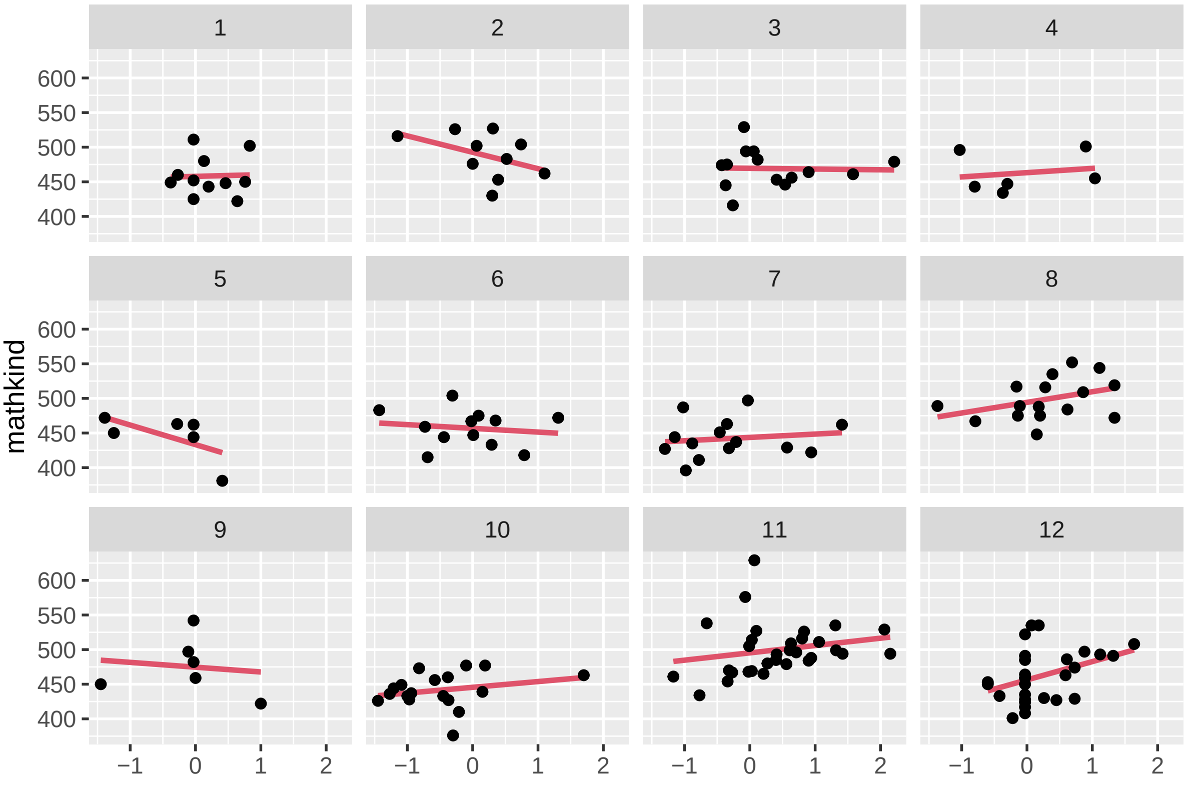
4. Some schools have higher outcomes, in general, but does the *relationship* between outcome and predictor vary by school (this is what we mean by a “response surface” for a predictor or set of them)?

   i. Here is an attempt to visualize some of the potential variation in the relationship of mathkind to ses by separately plotting the first 12 schoolids:

```
if (vanillaR) {
    xyplot(mathkind ~ ses | schoolid, data = dat[dat$schoolid < 13, ], type = c("p",
        "r"))  #close enough
} else {
    ggplot(data = subset(dat, schoolid < 13), aes(y = mathkind, x = ses)) + geom_smooth(method = "l
        se = FALSE, col = 2) + geom_point() + facet_wrap(~schoolid, nrow = 3)
}
```

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

cf. Jamboard (p 1,2)



concepts of (completely) pooled and unpooled (from G&H)

ii. The fitted lines are 'mini-regressions,' or 'un-pooled' regressions. What if we look at the coefficients from each of these? What might be a downside of doing this (is it practical to do this)? Would we get the same slope in each regression? How might we combine the school-specific slopes?
   a. One concern: small samples yield imprecise estimates
   b. Another concern: how to combine (meaningfully)?
   c. More subtle: do we want to assess the degree to which these slopes differ? Is this of intrinsic interest to us? (it may be)

iii. Below are the coefficients ($\hat{\beta}$) from 107 separate regressions (one per school) of mathkind on ses. We plot these first in ascending order with 95% confidence bounds, then plot the density of the coefficients taken as an ensemble.
   a. The median in this ensemble of coefficients is 8.3, their mean is 8.6; a weighted mean is 10.3. Pooled (ignoring grouping) estimate is 14.4 (red line). *Which one should you report?*
   b. We follow the plot with the distribution of the coeffients. The range of school-level effects (the effect of SES) is substantial (mainly between -50 and 80).

```
set.seed(2042001)
beta.ses <- 0
beta.se <- 0
school.n <- 0
idList <- unique(dat$schoolid)
len <- length(idList)
# slopes for SES by school, one for each.
for (i in 1:len) {
    b <- dat$schoolid == idList[i]
    school.n[i] <- sum(b)
    fit1 <- lm(dat$mathkind[b] ~ dat$ses[b])
    beta.ses[i] <- fit1$coef[2]
    beta.se[i] <- summary(fit1)$coef[2, 2]
```

Loop through each school separately

```
}
# means using different weights:

# unweighted
wtd.beta1 <- mean(beta.ses)
print(wtd.beta1)
```

```
## [1] 8.6352
```

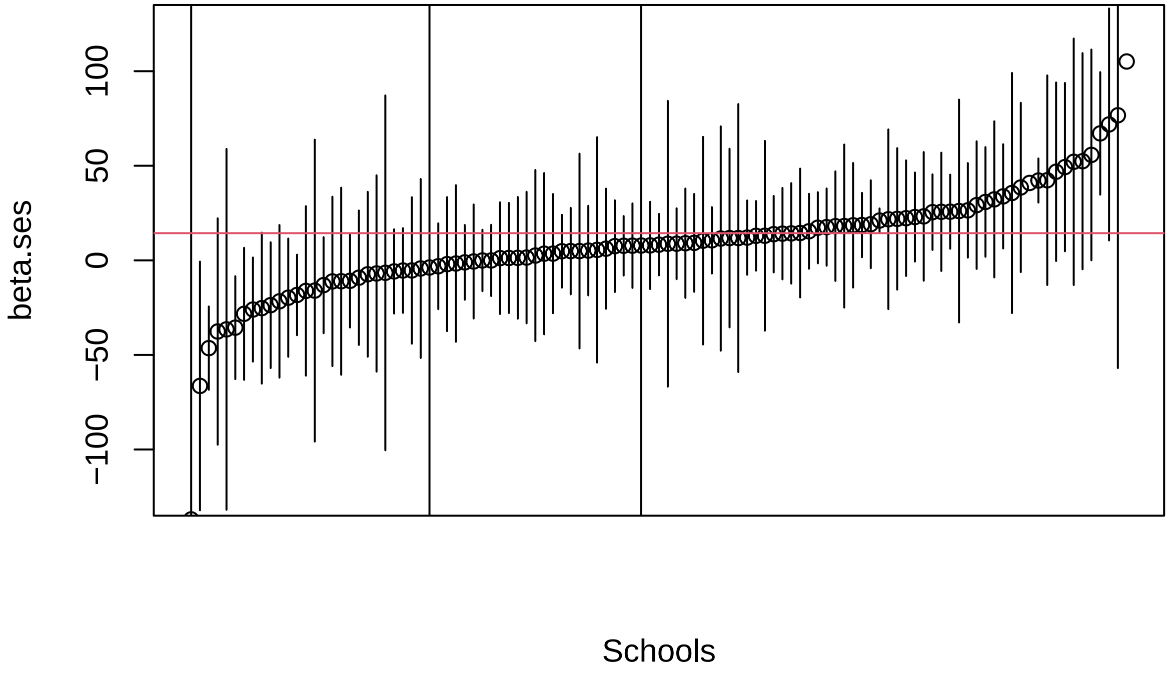
```
# wtd by school size
wtd.beta2 <- sum(beta.ses * school.n)/sum(school.n)
print(wtd.beta2)
```

```
## [1] 10.27034
```

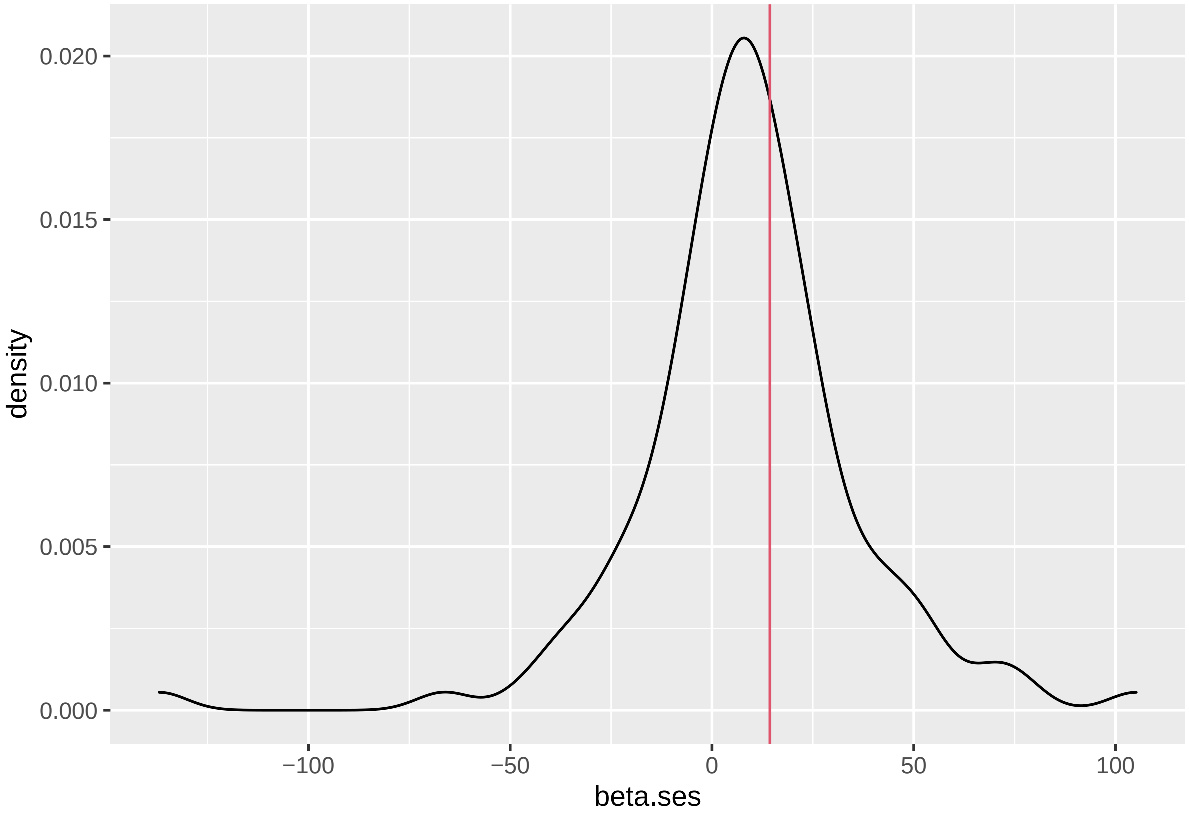
```
# OLS regression-based weights:
coef1 <- lsfit(dat$ses, dat$mathkind)$coef
print(coef1)
```

```
## Intercept         X
## 466.84525  14.35929
```

```
# PLOT the point ests of slopes:
ord <- order(beta.ses, decreasing = F)
plot(idList, beta.ses[ord], xlab = "Schools", ylab = "beta.ses", ylim = c(-125, 125),
    axes = F)
box()
axis(2)
segments(idList, beta.ses[ord] + 1.96 * beta.se[ord], idList, beta.ses[ord] - 1.96 *
    beta.se[ord])
abline(h = coef1[2], col = 2)
```



```
# density
if (vanillaR) {
    plot(density(beta.ses))
    abline(v = coef1[2], col = 2)  #line at regression point.
} else {
    ggplot(data = NULL, aes(beta.ses)) + geom_density() + geom_vline(xintercept = coef1[2],
        col = 2)
}
```



5. This example shows us that we might need models that allow for:
   i. Correlation within groups
   ii. Group-specific intercepts (level differences)
   iii. Variation in the response (or “returns”) associated with a predictor on the outcome, by group. We eventually call these random slopes.

Modeling: first pass

1. How can we write down a mathematical (statistical) model that accounts for some of this nested information, for our classroom example?
   i. The usual regression model, $MATHKIND_i = b_0 + b_1 SES_i + \varepsilon_i$, in which the index i tracks the student to which we are referring, fails to capture anything about students being nested within schools. What if we added another subscripted index that tracked this?
2. We write an equation: $MATHKIND_{ij} = b_0 + b_1 SES_{ij} + \varepsilon_{ij}$. Notationally, $MATHKIND_{ij}$ is the outcome for the i^{th} student in the j^{th} school, with corresponding predictor SES_{ij}, and error ε_{ij}.
   i. To uniquely identify a subject, we need both i and j; this allows there to be more than one student indexed by i in two different schools.
      a. i.e., student identifiers should be thought of as pairs (student, school): (1,1), (2,1),. . . , (1,2),(2,2),. . . *even if the actual IDs used in the dataset are unique across schools.*
   ii. Why go to this trouble of referring to schools in the subscript index? Because one way to account for the nested relationships is by introducing an “effect” indexed by group membership, and we need to associate effects with groups (here, schools).

a. An effect is simply a way to account for a difference. At one extreme, we could model each school as having its own intercept – this is one type of effect. It could be implemented using indicator variables – one for each school (omitting a single reference school). This would involve 106 new parameters in our school example. Completely unpooled (G&H)

b. If it is reasonable to assume that differences in level between schools (on average) follow a normal distribution and that *we are not interested in specific levels* for each school, but simply adjusting for those differences, then we can use a *random effect* to represent school (group) differences. With random effects:

- The assumption is that we have a sample from a population. cf. Jamboard (p. 3,4)
- We must also assume that these effects and errors are independent.

iii. Notationally, we use a Greek letter to represent a random effect and include enough indices to uniquely identify it:
$MATHKIND_{ij} = b_0 + b_1 SES_{ij} + \zeta_j + \varepsilon_{ij}$. School j gets a unique shift up or down, based on the value of ζ_j, its random effect.

iv. There is more to do: There are two random (stochastic) components to the model and they must be further specified. We usually assume $\zeta_j \sim N(0, \sigma_\zeta^2)$ [read this as, "zeta sub j is distributed as a normal random variable with mean 0 and variance sigma sub zeta squared] and $\varepsilon_{ij} \sim N(0, \sigma_\varepsilon^2)$, independently of one another.

3. With all of these components in place, we can fit the above model, $MATHKIND_{ij} = b_0 + b_1 SES_{ij} + \zeta_j + \varepsilon_{ij}$, using the method of maximum likelihood (this is what is commonly used in most statistical models you will encounter). What is different about this model (compared to regression)?

i. One concern: what is the role of the ζ_j term in the estimation? We do not plan to estimate each of these (107) separately, instead, we only estimate the variance of their distribution, σ_ζ^2.

a. While we do not estimate them individually, they are still part of the model, so one can think of each subject in school j as having a school-specific intercept, ζ_j. Including the intercepts produces better estimates (in a manner to be made more precise) of the effects for the remaining predictors.

ii. While the error terms ε_{ij} are independent, outcomes are not, and this is our first attempt to capture this in a model. The way that we can quantify the dependence is by comparing observations within and between groups (here, indexed by j).

a. We find that under this model, the correlation between two outcomes is known precisely:

$$\text{Cor}(MATHKIND_{ij}, MATHKIND_{i'j}) = \frac{\sigma_\zeta^2}{\sigma_\zeta^2 + \sigma_\varepsilon^2}, \text{ when } i \neq i'$$

Assuming same covariate pattern

- We assume j is the same (group or school) in this comparison. So observations are correlated in a prescribed manner, within schools.

b. $Cor(MATHKIND_{ij}, MATHKIND_{i'j'}) = 0$, when $j \neq j'$ (different groups, or schools in our example).

c. This is the proportion of variation between groups as a fraction of the total and also known as the Intraclass Correlation Coefficient (ICC).

4. Let's fit the above model (more on 'fitting' near the end of class today) for our schools example.

i. In R, there are two commonly used libraries for linear models with random effects: nlme and lme4. The lme4 library and corresponding `lmer` command will primarily be used in this course.

a. `lmer(mathkind~ses+(1|schoolid))` is the syntax for a basic random intercept model.

b. The nested structure is specified inside parentheses and uses a vertical bar, '|' to indicate the name of the grouping factor. The '1' indicates a random intercept.

c. The default fitting method is known as REML – we will discuss this later.

ii. Question: Do you have *intuition* for what a random effect is?

```
require(lme4)
```

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

```r
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
lme1 <- lmer(mathkind ~ ses + (1 | schoolid), data = dat)
print(summary(lme1))
```

```
## 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
##  Residual             1308     36.16
## Number of obs: 1190, groups:  schoolid, 107
##
## 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
```

ests. of sigma^2 zeta, epsilon, respectively.

```r
rand(lme1)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## mathkind ~ ses + (1 | schoolid)
##                  npar  logLik   AIC   LRT Df Pr(>Chisq)
## <none>              4 -6019.3 12047
## (1 | schoolid)      3 -6077.4 12161 116.1  1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

5. Discussion of model fit results
    i. The components of this nested data model are divided into two sections.
        a. Random effects: listed as schoolid (intercept) is the estimated variance $\hat{\sigma}_\zeta^2$ in level differences

ζ_j between schools. The value, 309, should be compared to the estimated residual variance, $\hat{\sigma}^2_{\varepsilon}$, 1308.

b. Fixed effects table (this label is controversial). The effect of a one unit change in SES on MATHKIND is 10.72, controlling for differences between schools, if our model and its assumptions are correct.

ii. The fraction of total variance accounted for by differences between groups is 309/(309+1308) = 0.19. This is also an estimate of the correlation between subjects within schools (and is the ICC).

iii. To know whether the random effects were 'warranted', we conduct a likelihood ratio (LR) test, as the model with and without the random effects are nested. The null tested is $H_0 : \sigma^2_{\zeta} = 0$ using lmerTest package's `rand` function.

a. The additional effects are warranted (p<.0001).

Simpler baseline models

1. Some researchers prefer to start with simpler initial models. Typically, these include at least one random effect, but no predictors. This is called an *unconditional means model* (UMM).

    i. *Why do it?* It documents the extent to which variation appears to be between vs. within grouping factor(s).

    ii. For school data, we can fit

    $$MATHKIND_{ij} = b_0 + \zeta_j + \varepsilon_{ij}, \text{ with } \zeta_j \sim N(0, \sigma^2_{\zeta}), \varepsilon_{ij} \sim N(0, \sigma^2_{\varepsilon}), \text{ indep.}$$

    a. Note: for those who are familiar with panel data models, clearly the UMM is more appropriate as a baseline model for nested data as contrasted to panel data (why?).

    iii. Our lmer model simply drops the predictor SES, leaving: `lmer(mathkind~(1|schoolid))`

```
lme2 <- lmer(mathkind ~ (1 | schoolid), data = dat)
print(summary(lme2))
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: mathkind ~ (1 | schoolid)
##    Data: dat
##
## REML criterion at convergence: 12085.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.8223 -0.5749  0.0005  0.6454  3.6237
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  schoolid (Intercept)  364.3   19.09
##  Residual             1344.5   36.67
## Number of obs: 1190, groups:  schoolid, 107
##
## Fixed effects:
##             Estimate Std. Error     df t value Pr(>|t|)
## (Intercept)   465.23       2.19 103.20   212.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
rand(lme2)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
```

```
## mathkind ~ (1 | schoolid)
##                 npar  logLik   AIC    LRT Df Pr(>Chisq)
## <none>             3 -6042.8 12092
## (1 | schoolid)     2 -6119.3 12243 152.96  1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

iii. The fixed effects are uninteresting – simply a constant is reported.

iv. LR test indicates significant random effects (they are needed)

v. The variances of the random effects are interesting. Using the formula $ICC = \frac{\sigma^2_\zeta}{\sigma^2_\zeta+\sigma^2_\varepsilon}$, we see that 364.3/(364.3+1344.5)=21% of the variance is between schools (Reminder: ICC stands for intraclass correlation coefficient). This suggests that if we had school-level predictors, they might 'explain' some of this variation. Similarly, the 79% of the variation that is within schools might be explainable via student level (within-group) predictors.

2. We have been focusing on schools as the grouping factor, but we have an even smaller grouping factor: classrooms. We can just as easily fit a model with groups determined by classrooms, which happen to be nested within schools.

   i. Three layers of nesting can be made explicit in the notation. Student i is in classroom j in school k (we had to 'insert' classroom before school, so the index for school becomes k). So one version of a model for outcomes is:

   $$MATHKIND_{ijk} = b_0 + \eta_{jk} + \varepsilon_{ijk}, \text{ with } \eta_{jk} \sim N(0, \sigma^2_\eta) \text{ and } \varepsilon_{ijk} \sim N(0, \sigma^2_\varepsilon); \text{ with } \eta \perp \varepsilon$$

   ii. In this model, the random effect associated with classrooms introduces a new symbol, η_{jk}, to distinguish it from any school-level effects. The subscripts reflect the fact that classrooms are nested within schools. As before random effects and error are assumed independent of one another

      a. Notice that we have added an 'extra' index to the residual term).

   iii. Notationally, we prefer to index classroom effects by jk because there could be two classrooms indexed by '$j = 1$' in two different schools (or just to organize the construction of effects). η_{jk} represents the effect for the j^{th} classroom in the k^{th} school.

   iv. With lmer, we use the syntax: `lmer(mathkind~(1|classid))` which indicates that there is a random intercept for the groups determined by classrooms (note that in this data, classid is unique between school as well. Had it not been, we would use `schoolid:classid` to refer to the unique index). The resulting fit is given next.

```r
lme3 <- lmer(mathkind ~ (1 | classid), data = dat)
print(summary(lme3))
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: mathkind ~ (1 | classid)
##    Data: dat
##
## REML criterion at convergence: 12149.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.5541 -0.5719 -0.0173  0.6295  3.5807
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  classid  (Intercept)  425.5   20.63
##  Residual             1305.8   36.14
## Number of obs: 1190, groups:  classid, 312
##
## Fixed effects:
```

```
##             Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  465.866      1.629 272.864   285.9   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
rand(lme3)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## mathkind ~ (1 | classid)
##                npar  logLik   AIC    LRT Df Pr(>Chisq)
## <none>            3 -6074.7 12155
## (1 | classid)     2 -6119.3 12243 89.221  1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

iv. (cont.)

a. The LR test indicates significant random effects for classrooms (this should not be surprising, since it was true for schools – *why does this follow?*)

b. Using the formula $ICC = \frac{\sigma_\eta^2}{\sigma_\eta^2+\sigma_\varepsilon^2}$, we see that 425/(425+1306)=25% of the variance is between classrooms (ignoring schools).

c. With more of the variance 'explained' by differences between classrooms, is this a better model (than the school-level model)? Perhaps, but this is not the way to make the assessment.

Two levels of nesting (schools and classrooms)

1. We now extend our example to include school-level effects in addition to the classroom level effects.

   i. The general outcome is Y_{ijk}, for student i in classroom j in school k.

   ii. For now, we assume that there is only one observation per student (no repeated measures).

   iii. We introduced notation that specifies *effects* for classrooms and schools separately, but their different names η_{jk}, ζ_k allow us to use them both in the same model. Notice how our random effects use just enough indices to be identifiable:
   a. ζ_k suffices for schools: we don't need a classroom index, because all classrooms in school k will get this effect.
   b. η_{jk} represents the effect for the j^{th} classroom in the k^{th} school; all students in that classroom will get this effect.
   c. An unconditional means model for the outcome (no covariates), that includes effects for both schools and classrooms would then be specified as:

$$Y_{ijk} = b_0 + \eta_{jk} + \zeta_k + \varepsilon_{ijk}$$

Doubly nested

   d. Reminders: $MATHKIND_{ijk} = Y_{ijk}$ in our example; The random effects (η_{jk}, ζ_k) are taken as mutually independent and normally distributed, with level-specific variances, $\sigma_\eta^2, \sigma_\zeta^2$, resp.; both are assumed independent of ε_{ijk} as well.

   - We write these distributional assumptions as: $\eta_{jk} \sim N(0, \sigma_\eta^2)$, $\zeta_k \sim N(0, \sigma_\zeta^2)$, independently of each other and $\varepsilon_{ijk} \sim N(0, \sigma_\varepsilon^2)$.

2. We fit the model $Y_{ijk} = b_0 + \eta_{jk} + \zeta_k + \varepsilon_{ijk}$, in which schools and classrooms are modeled together. We interpret parameters in such a model as follows: net of school effects, how much additional systematic variation exists at the classroom level (in this case, level differences)?

   i. In the following R code, the order of the two random effects is important. Since classrooms are nested within schools, schoolid should be listed first.

   ii. Failure to organize the nesting properly can result in a misspecified model – in this case, nesting schools within classrooms will generate more than one school effect for the same school, which is not correct.

a. Note that two sets of group identifiers and a slightly different syntax is used.

b. In computing the likelihood ratio test of significance for the random effects variance, the syntax changes as well.

```
lme4 <- lmer(mathkind ~ (1 | schoolid/classid), data = dat)
print(summary(lme4))
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: mathkind ~ (1 | schoolid/classid)
##    Data: dat
##
## REML criterion at convergence: 12085.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.8198 -0.5749  0.0012  0.6303  3.5893
##
## Random effects:
##  Groups           Name        Variance Std.Dev.
##  classid:schoolid (Intercept)   32.0    5.657
##  schoolid         (Intercept)  352.8   18.784
##  Residual                     1323.4   36.379
## Number of obs: 1190, groups:  classid:schoolid, 312; schoolid, 107
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  465.220      2.191 103.579   212.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Notation: classid:schoolid here indicates interaction (unique combinations)

```
lm0 <- lm(mathkind ~ 1, data = dat)  # linear model with same 'fixed' effects but no random effects
anova(lme4, lm0, refit = F)  #test the significance of random effects (jointly).
```

```
## Warning in anova.merMod(lme4, lm0, refit = F): some models fit with REML = TRUE,
## some not

## Data: dat
## Models:
## lm0: mathkind ~ 1
## lme4: mathkind ~ (1 | schoolid/classid)
##      npar   AIC   BIC  logLik deviance  Chisq Df Pr(>Chisq)
## lm0     2 12245 12255 -6120.4    12241
## lme4    4 12093 12113 -6042.5    12085 155.78  2  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

iii. The estimated variances are: $\hat{\sigma}^2_\eta = 32.0$; $\hat{\sigma}^2_\zeta = 352.8$.

iv. We can extend the formula for proportion of variance ‘explained’ by each effect. We simply include only one random effect variance at a time in the numerator, and sum all variance components in the denominator.

a. So, 353/(353+32+1323), or about 21% of the variation is between schools, while 32/(353+32+1323), or 2% is between classrooms, *net of schools*; the remaining variation is within classrooms at the student level.

v. Important: the last line of output gives an LR test that suggests that the model is improved

random intercepts
(L2 & L3)

significantly when both effects are included *(as compared to linear regression)*.

3. We now add a covariate to the model $Y_{ijk} = b_0 + b_1 SES_{ijk} + \eta_{jk} + \zeta_k + \varepsilon_{ijk}$, in which schools and classrooms are modeled together. We interpret the effect for that covariate as follows: *net of school and classroom effects, what is the impact of a student's SES on our outcome, MATHKIND?*

```
lme5 <- lmer(mathkind ~ ses + (1 | schoolid/classid), data = dat)
print(summary(lme5))
```

```
## 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
##  schoolid         (Intercept)  301.52  17.364
##  Residual                     1294.08  35.973
## 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
```

```
lm1 <- lm(mathkind ~ ses, data = dat)  # linear model with ses 'fixed' effects but no random effec
anova(lme5, lm1, refit = F)
```

```
## Warning in anova.merMod(lme5, lm1, refit = F): some models fit with REML = TRUE,
## some not
```

```
## Data: dat
## Models:
## lm1: mathkind ~ ses
## lme5: mathkind ~ ses + (1 | schoolid/classid)
##      npar   AIC   BIC  logLik deviance  Chisq Df Pr(>Chisq)
## lm1     3 12166 12181 -6079.8    12160
## lme5    5 12048 12074 -6019.2    12038 121.23  2  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

i. In terms of proportion of variation, 302/(302+20+1294), or about 19% of the *total* variation is between schools, while 20/(302+20+1294), or 1% is between classrooms, *net of schools and SES*; the remaining variation is within classrooms at the student level.

ii. The last lines give an LR test that suggests that the model is improved significantly when both effects are included. If you want to test the need for just one effect, such as σ^2_η, you need another procedure.

a. The estimate of $\hat{\sigma}^2_\eta = 20$, which is very small. We can 'request' confidence intervals as follows:

```
ci <- confint(lme5)
```

```
## Computing profile confidence intervals ...
```

```
ci[1:3, ]^2  #only display for variance components (and square to make variances)
```

```
##              2.5 %    97.5 %
## .sig01    0.0000  114.8900  sig^2_eta
## .sig02  190.7281  446.5601  sig^2_zeta
## .sigma 1179.5292 1418.8220  sig^2_eps
```

2. (cont.)
   - ii. (cont.)
     - a. (cont.)
       - But…confidence bounds for variance terms can never cross zero (conceptually, and estimation is constrained to prevent it), as variances are always positive.
     - b. We need another way to test $H_0 : \sigma^2_\eta = 0$.
   - iii. To assess whether adding classroom effects, in addition to school effects is warranted, we do a likelihood ratio (LR) test comparing the a model without these effects to one that contains them. The models are nested, and the only difference between them is that in one, $\sigma^2_\eta = 0$, so the LR test will determine how unusual our results are, assuming this variance to be zero under the null. Sequentially:
     a. We run the simpler model and store its estimates (this has already been done, as model `lme1`).
     b. We run the more complex model and store its estimates (already done, as model `lme5`).
     c. The LR test used the `anova` command (unless we specify refit=F, it automatically refits using ML not REML–to be explained in subsequent handouts).
     d. Note that we show you two ways to attempt to derive this test. The first is direct (fit 2 nested models); the second derives multiple tests, and the ordering is such that it tests each random effect in the presence of the other, so we get the same result.

```
anova(lme1, lme5, refit = F)
```

```
## Data: dat
## Models:
## lme1: mathkind ~ ses + (1 | schoolid)
## lme5: mathkind ~ ses + (1 | schoolid/classid)
##      npar   AIC   BIC  logLik deviance  Chisq Df Pr(>Chisq)
## lme1    4 12047 12067 -6019.3    12039                          Sequentially add params.
## lme5    5 12048 12074 -6019.2    12038 0.2565  1     0.6125
```

```
rand(lme5)
```

```
## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## mathkind ~ ses + (1 | classid:schoolid) + (1 | schoolid)
##                         npar  logLik   AIC    LRT Df Pr(>Chisq)
## <none>                     5 -6019.2 12048
## (1 | classid:schoolid)     4 -6019.3 12047  0.257  1     0.6125  Result after removing one param.
## (1 | schoolid)             4 -6045.9 12100 53.359  1  2.778e-13 ***Removal of School Level effects is not warranted
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ii. (cont.)

a. The p-value (0.61) suggests that we don't need classroom level effects (we do not reject the null–*Are you surprised?*), but a "better" p-value divides this by 2 to reflect the evaluation on the boundary of the parameter space.[1]

OLS & MLE (sketch)

1. In this course, we rely exclusively on the theory associated with Maximum Likelihood Estimation (MLE) and you may not have been exposed to it very much (APSTA-GE.2122 delves into the details). These notes are an attempt to give you an idea of the method and contrast it to OLS, which you know from regression.

i. Example: determining the mean of a population of known variance by examining a sample.

a. This is a classic example. It seems easy – just take the mean of the sample, and you have a good estimate of the mean of the population from which it was derived, right? Yes, but why is this a good idea?

b. There are certainly more complicated situations in which finding the "best" parameters isn't straightforward… and *you are in this situation in this course!*

ii. To make progress with this concern, we have to make the assumptions more explicit. Let us say that we have a population of size N (very large) and we know that it is normally distributed with standard deviation one. We just don't know the mean of this population, which we will call μ.

iii. We say that the data generating process (DGP) – the mechanism by which observations, X, are generated, is $X \sim N(\mu, 1)$ (a normal distribution with s.d.=1, so the variance is also 1).

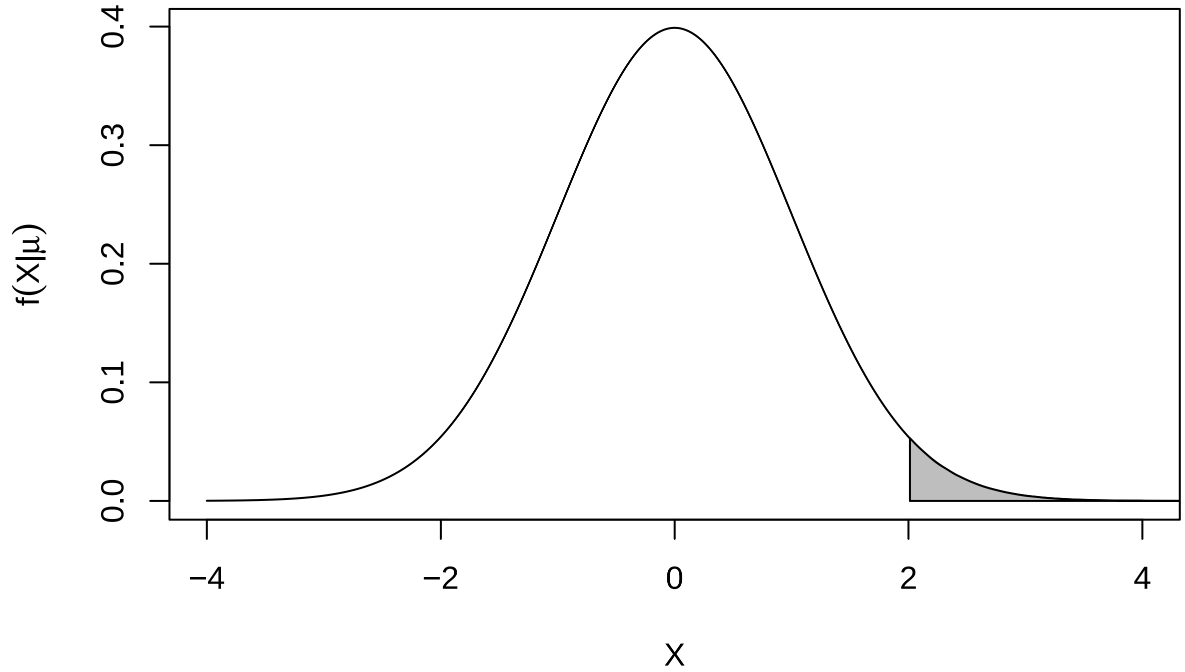
iv. The probability of observing a single X from this distribution derives from the density:

$$f(X|\mu) = \frac{1}{\sqrt{2\pi}} e^{\frac{-(X-\mu)^2}{2}}$$

This is called the density of X given μ (so μ is fixed); you can think of it as defining the shape of the histogram you would get if you sampled millions of observations of X (so you know which X values are more or less likely). Here is what it looks like for $\mu = 0$:

```r
set.seed(2042001)
# normal mean example: standard normal density displayed:
grd <- seq(-4, 4, length = 9 * 100)
plot(grd, dnorm(grd), type = "l", xlab = "X", ylab = expression(f(paste(X, "|", mu))))
dens <- density(rnorm(1e+07))
x1 <- min(which(dens$x >= 2))
x2 <- max(which(dens$x < 5))
with(dens, polygon(x = c(x[c(x1, x1:x2, x2)]), y = c(0, y[x1:x2], 0), col = "gray"))
```

[1] See Stram, D., & Lee, J. (1994). Variance Components Testing in the Longitudinal Mixed Effects Model. Biometrics, 50(4), 1171-1177. doi:10.2307/2533455



iv. (cont.)

   a. This is a familiar picture, and it allows you to calculate a probability. For example, P(X>2) is given by the shaded area under the curve. And X "near" 0 is most likely, while X "near" 4 is very unlikely.
   b. With a simple switch of the ordering, we define what we call the likelihood of μ given X [this is different, conceptually]. $\mathcal{L}(\mu|X) = f(X|\mu)$. The conceptual difference is that we ask ourselves, which values of μ are most likely to have *given rise to the observed data*, X (thus conditioning on X), assuming the model is correct?

2. We can also evaluate the density of M *multiple* independent (standard normal) observations $(X_1, \ldots, X_M)$. It is:

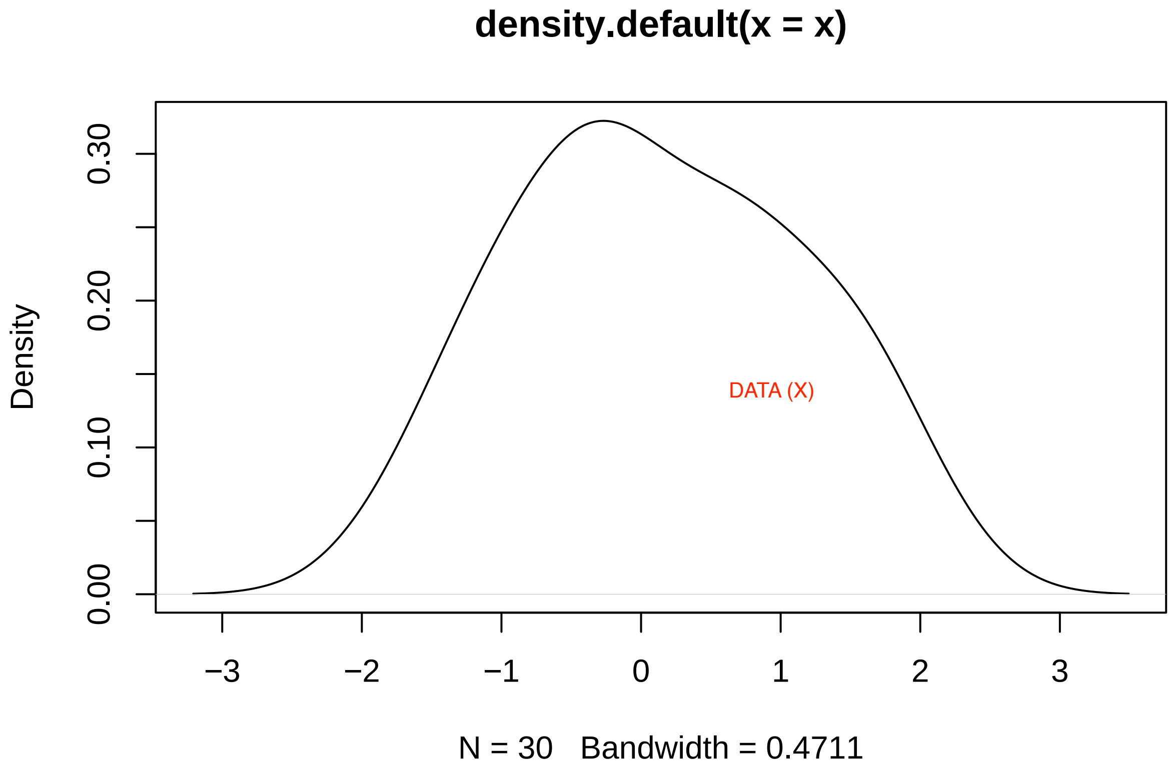
$$f(X_1, ..., X_M|\mu) = \prod_{i=1}^{M} \frac{1}{\sqrt{2\pi}} e^{\frac{-(X_i-\mu)^2}{2}} = \frac{1}{(2\pi)^{M/2}} e^{\frac{-\sum\limits_{i=1}^{M}(X_i-\mu)^2}{2}}$$

and the corresponding likelihood is $\underline{\mathcal{L}(\mu|X_1, ..., X_M) = f(X_1, ..., X_M|\mu)}$

3. In real situations, we observe $X_1, \ldots, X_M$ and wish to infer μ. The principle of Maximum Likelihood says that we should choose μ so that the log-likelihood, defined as $\ell(\mu|X_1, ..., X_M) = \log f(X_1, ..., X_M|\mu) = -\frac{M}{2}\log(2\pi) - \frac{1}{2}\sum\limits_{i=1}^{M}(X_i - \mu)^2$ is maximized (over the range of valid μ).

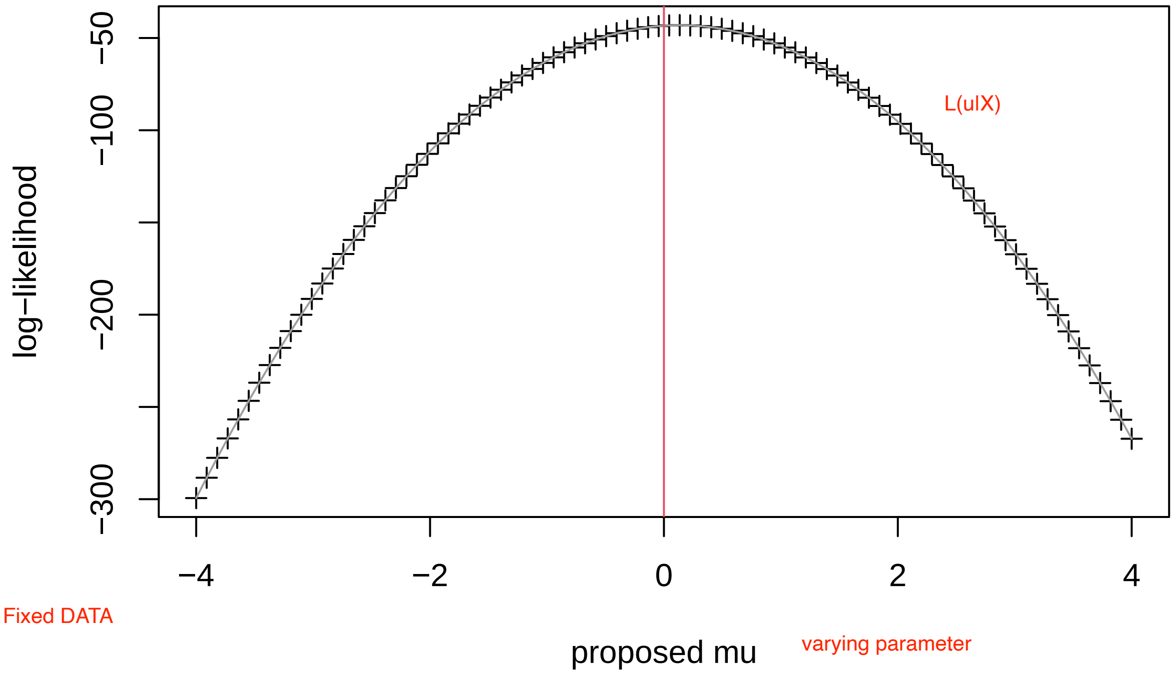
   i. Example (cont.): Say we have some number of independent observations, X, from a standard normal distribution, so the truth is that $\mu = 0$. Here is a density plot from a draw of 30:

```
# sample from DGP:
N <- 30
x <- rnorm(N)
plot(density(x))
```



ii. To apply the Principle of Maximum Likelihood, we calculate $\ell(\mu|X_1, ..., X_M) = \log f(X_1, ..., X_M|\mu)$ for a range of μ and report the value that maximizes the log-likelihood ℓ. We did this for a grid of points between -4 and 4 and we get this plot.

```r
# set up for MLE: need the likelihood function (R codes this as dnorm):
loglik.norm <- function(x, mu, sd = 1) sum(dnorm(x, mean = mu, sd = sd, log = T))
# grid search - crude MLE
grd <- seq(-4, 4, length = 9 * 10)
ll <- rep(NA, length(grd))  #store loglik values here
# compute ll across grid
for (i in 1:length(grd)) ll[i] <- loglik.norm(x, mu = grd[i])
# plot ll across grid
plot(grd, ll, type = "p", pch = 3, ylab = "log-likelihood", xlab = "proposed mu")
lines(grd, ll, col = 8)  #connect the points
abline(v = 0, col = 2)  #true value of mu from population
```



iii. Clearly the choice of μ that maximizes the log-likelihood is 0. It turns out that is also the mean of this sample (to a close approximation).

iv. You may have noticed that we were trying to find the μ that maximizes: $-\frac{M}{2}\log(2\pi) - \frac{1}{2}\sum_{i=1}^{M}(X_i-\mu)^2$ and that since the first term is a constant, this will be a maximum when $\sum_{i=1}^{M}(X_i-\mu)^2$ is minimized. There are lots of ways to find the minimum of that expression (those who know calculus should see that this quadratic is minimized at the sample mean), but for now, just notice that the expression looks a lot like the sum of squared residuals. The term "least squares" should come to mind.

4. If you had a regression model (X as outcome) $X_i = b_0 + \varepsilon_i$ with the usual assumptions of independent normal errors, then an OLS approach to finding an estimate $\hat{b}_0$ would be to compute residuals from the model over different values of b_0 and chose $\hat{b}_0$ such that it minimizes the sum of squared residuals: $\sum_{i=1}^{M}(X_i-b_0)^2$. This should look familiar – it is the criterion used in the maximization of the log-likelihood, above, with b_0 substituted for μ.

   i. For the estimation of unknown mean and known variance, the MLE and the OLS solution are identical and lead to optimizing with respect to the same sum of squares criterion.

5. Example 2: Poisson regression

   i. In this instance, OLS and MLE give different answers, and you would not "trust" the OLS results. We need to be clear, though, that by OLS, we mean applying a sum of squared residuals criterion to search for an "optimal" set of regression parameters.

   ii. The model (for one explanatory predictor, and constant exposure) is: $\ln\lambda_i = b_0 + b_1X_i$, and $Y_i \sim Pois(\lambda_i)$ (Y is a Poisson random variable with rate λ_i). Residuals from the model are of the form:$Y_i - \exp(b_0 + b_1X_i)$, where $\exp(x) = e^x$.

   iii. Using a likelihood-based approach, the density of Y (a count) follows a Poisson distribution: $f(Y_i|X_i, b_0, b_1) = \frac{e^{-\lambda_i}\lambda_i{}^{Y_i}}{Y_i!}$, where $\ln\lambda_i = b_0 + b_1X_i$.

iv. When we extend this to M independent observations, the density becomes:

$$f(Y_1,\dots,Y_M|X_1,\dots,X_M,b_0,b_1) = \frac{e^{-\lambda_1}{\lambda_1}^{Y_i}}{Y_1!}\cdots\frac{e^{-\lambda_M}{\lambda_M}^{Y_M}}{Y_M!}$$

with $\ln\lambda_i = b_0 + b_1 X_i$ for $i = 1,\dots,M$. Change X_i and you change the rate λ_i.

v. A likelihood based approach reverses this equation, so that you try to maximize (over b_0, b_1):

$$\mathcal{L}(b_0,b_1|X_1,\dots,X_M,Y_1,\dots,Y_M) = f(Y_1,\dots,Y_M|X_1,\dots,X_M,b_0,b_1)$$

   a. You can imagine using a grid search through plausible b_0, b_1 until you find one that maximizes that likelihood.
   b. The OLS approach would try to *minimize* $\sum_{i=1}^{M}(Y_i - \exp(b_0 + b_1 X_i))^2$ over plausible b_0, b_1.

vi. We did a grid search over plausible b_0, b_1 using this OLS approach, in which the true model was $Y_i \sim Pois(\lambda_i)$, with $\ln\lambda_i = -0.5 + 0.5X_i$. We took a sample of size 3000, drawing X from a standard normal, and Y from a Poisson process, corresponding to the rates given by the "regression" portion of the model. We would hope to discover that $\hat{b}_0 = -0.5$; $\hat{b}_1 = 0.5$ (the truth).

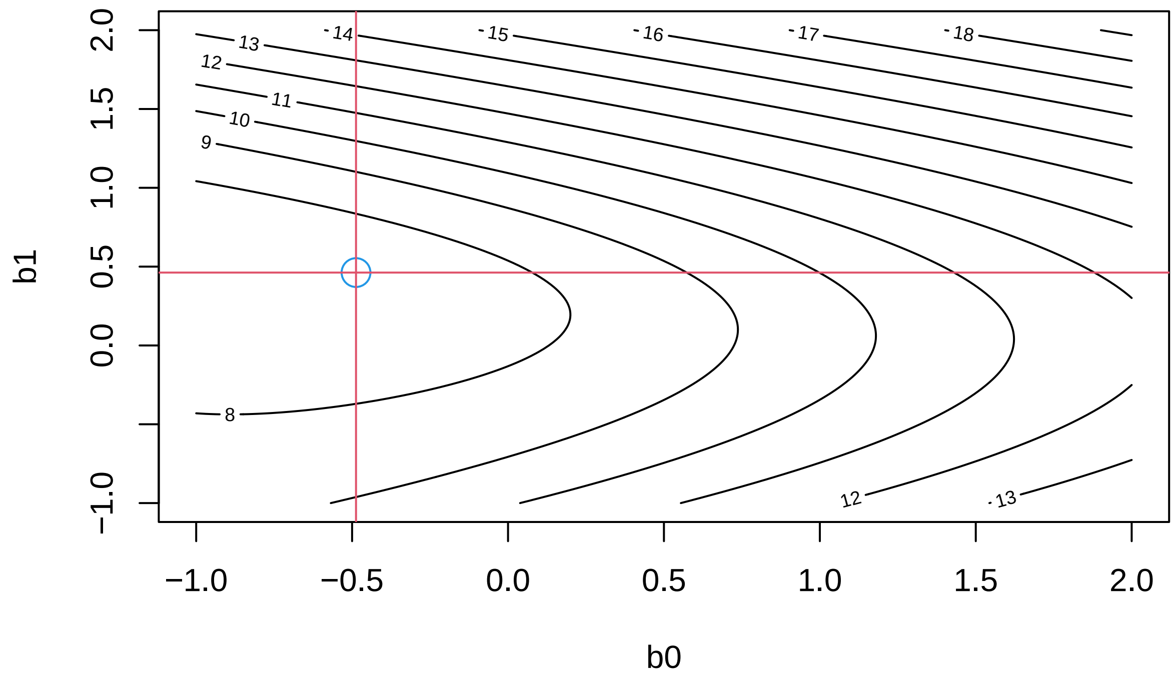
   a. In our simulation, the OLS approach, minimizing sum of squared residuals, we get $\hat{b}_0 = -0.49$; $\hat{b}_1 = 0.46$. Here are some details:

```
# poisson example
set.seed(3)
N <- 3000
x <- rnorm(N)
rt <- exp(-0.5 + 0.5 * x)  #model for poisson rate
y <- rpois(N, rt)  #generate data
# use Sum Sqs criterion to search for best params in poisson model.  this fn
# computes the sum sqs
ssq.pois <- function(betas, x, y) {
    n <- dim(betas)[1]
    muhat <- betas %*% rbind(1, x)
    res <- matrix(rep(y, n), nrow = n, byrow = T) - exp(muhat)
    ssq <- apply(res * res, 1, sum)
    ssq
}
# set up for grid search
grd.len <- 4 * 5 * 10  #*2 for speedup
grd <- seq(-1, 2, length = grd.len)
betas <- as.matrix(merge(grd, grd))  #trick to build the crossproduct of grid values in range (-1,2
ssq <- ssq.pois(betas, x, y)  #this computes all 200*200 search points Sum Sqs rslts
contour(x = grd, y = grd, z = ssq.mat <- matrix(log(ssq), grd.len, grd.len, byrow = F),
    xlab = "b0", ylab = "b1")
min.loc <- which(ssq.mat == min(ssq.mat), arr.ind = T)
cat("Grid-based fit (b0, b1): ", grd[min.loc])
```

```
## Grid-based fit (b0, b1):  -0.4874372 0.4623116
```

```
points(grd[min.loc[1]], grd[min.loc[2]], col = 4, pch = 1, cex = 2)
abline(v = grd[min.loc[1]], col = 2)
abline(h = grd[min.loc[2]], col = 2)
```

DATA (X) is fixed (not an axis!)



v. The MLE approach yields this:

```
fit <- glm(y ~ x, family = "poisson")
summary(fit)
```

```
##
## Call:
## glm(formula = y ~ x, family = "poisson")
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3266  -1.0397  -0.7155   0.5440   3.4077
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.50108    0.02456  -20.40   <2e-16 ***
## x            0.48534    0.02225   21.82   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 3521.1  on 2999  degrees of freedom
## Residual deviance: 3042.7  on 2998  degrees of freedom
## AIC: 6206.9
##
## Number of Fisher Scoring iterations: 5
```

vi. The results are not terribly different, although with 3000 observations, one would expect to be able to recover the DGP parameters fairly precisely. And the MLE-based approach did get closer to the “truth.”

   a. But you must also ask yourself another question: if I use the result from OLS for Poisson regression, *what are my confidence intervals for those estimates*?