36-463/663: Multilevel & Hierarchical Models
Fall 2016
HW06 – SOLUTIONS

Problem One

A

$Corr(y_i, y_{i'}) = \frac{Cov(y_i,y_{i'})}{\sqrt{Var(y_i)Var(y_{i'}}}$. We know $\sqrt{Var(y_i)Var(y_{i'})} = \tau^2 + \sigma^2$, so now only need the covariance. We find

$$\begin{align} Cov(y_i, y_{i'}) &= Cov(\epsilon_i, \epsilon_{i'}) + Cov(\eta_j, \epsilon_{i'}) + Cov(\eta_{i'}, \epsilon i) + Cov(\eta_j, \eta_{j'}) \tag{1}\\ &= 0 + 0 + 0 + 0. \tag{2} \end{align}$$

In this case, all the covariances are of IID gaussians or are from distinct, independent gaussians, and are all zero. So $Corr(y_i, y_{i'}) = \frac{Cov(y_i,y_{i'})}{\sqrt{Var(y_i)Var(y_{i'}}} = 0$.

B

We do the same computations, but this time find that:

$$\begin{align} Cov(y_i, y_{i'}) &= Cov(\epsilon_i, \epsilon_{i'}) + Cov(\eta_j, \epsilon_{i'}) + Cov(\eta_{i'}, \epsilon i) + Cov(\eta_j, \eta_{j'}) \tag{3}\\ &= 0 + 0 + 0 + Cov(\eta_j, \eta_j) \tag{4}\\ &= Var(\eta_j) \tag{5}\\ Cov(y_i, y_{i'}) &= \tau^2 \tag{6} \end{align}$$

So $Corr(y_i, y_{i'}) = \frac{Cov(y_i,y_{i'})}{\sqrt{Var(y_i)Var(y_{i'})}} = \frac{\tau^2}{\tau^2+\sigma^2}$.

C

If we are excluding our average to those observations in group j, each observation is $y_i = \beta_0 + \eta_j + \epsilon_i$. Then, summing over all entries in j,

$$\begin{align} Var(\overline{y_j}) &= \frac{1}{n_j^2}\Sigma_a\Sigma_b Cov(y_a, y_b) \tag{7}\\ Var(\overline{y_j}) &= \frac{1}{n_j^2}\Sigma_a\Sigma_b Cov(\beta_0 + \eta_j + \epsilon_a, \beta_0 + \eta_j + \epsilon_b). \tag{8} \end{align}$$

The covariances across ϵs for $a \neq b$ are 0, as are the covariances with β_0, and those across ϵs and ηs. This leaves only the n_j matches of ϵs, which have covariance σ^2, and n_j^2 matches of $Cov(\eta_j, \eta_j)$, which is τ^2. So

$$Var(\overline{y_j}) = \frac{1}{n_j^2}(\Sigma_a\Sigma_b I_{a==b} Cov(\epsilon_a, \epsilon_b) + \Sigma_a\Sigma_b Cov(\eta_j, \eta_j)) \tag{9}$$

$$Var(\overline{y_j}) = \frac{1}{n_j^2}(\Sigma_a\Sigma_b I_{a==b}\sigma^2 + \Sigma_a\Sigma_b\tau^2) \tag{10}$$

$$Var(\overline{y_j}) = \frac{1}{n_j^2}(n_j\sigma^2 + n_j^2\tau^2) \tag{11}$$

$$Var(\overline{y_j}) = \frac{1}{n_j}\sigma^2 + \tau^2. \tag{12}$$

D

in part (c) we proved that $Var(\overline{y_i}) = \tau^2 + \frac{\sigma^2}{n_j}$.

It should be clear that $Var(\overline{y_j}) = Var(\overline{y_j^*})$. Now for $Cov(\overline{y_j}, \overline{y_j^*}) = Cov(\frac{1}{n_j}\sum_{i=1}^{n_j} y_i, \frac{1}{n_j}\sum_{i=1}^{n_j} y_i^*) = Cov(\eta_i, \eta_i) = \tau^2$ similar to part (b). So $Corr(\frac{1}{n_j}\sum_{i=1}^{n_j} y_i, \frac{1}{n_j}\sum_{i=1}^{n_j} y_j^*) = \frac{\tau^2}{\tau^2+\frac{\sigma^2}{n_j}}$.

Problem Two - G&H 12.9

```
summary(full_fit = lmer(log.radon ~ floor + log.uranium + (1|county),data = data))
Random effects:
 Groups   Name        Variance Std.Dev.
 county   (Intercept) 0.024462 0.15640
 Residual             0.575230 0.75844
Number of obs: 919, groups: county, 85

Fixed effects:
            Estimate Std. Error t value
(Intercept)  1.46576    0.03794   38.64
floor       -0.66824    0.06880   -9.71
log.uranium  0.72027    0.09176    7.85

Correlation of Fixed Effects:
            (Intr) floor
floor       -0.357
log.uranium  0.145 -0.009

index = 1:nrow(data)
summary(subfit = lmer(log.radon ~ floor + log.uranium + (1|county),data = data[sample(index,0.2*nrow(da
Random effects:
 Groups   Name        Variance Std.Dev.
 county   (Intercept) 0.07807  0.2794
 Residual             0.55835  0.7472
Number of obs: 183, groups:  county, 57

Fixed effects:
            Estimate Std. Error t value
(Intercept)   1.4456     0.0839  17.230
floor        -0.5726     0.1504  -3.808
log.uranium   0.6786     0.1905   3.563
```

```
Correlation of Fixed Effects:
            (Intr) floor
floor       -0.409
log.uranium  0.193  0.018
```

The model fitted to the sampled data has but with larger standard errors. This is not surprising, since there is only 1/5 of the data. The variance components are also fairly similar to the full data set fit (but see next part).

(b) I ran the analysis on randomly sampled subsets of 1/5 of the data a few times; here are the estimated county-level variance (τ^2) and individual residual variance (σ^2) over these runs:

Parameter	Estimates from 1/5 samples				
τ^2	0.009286	0.0000	0.007461	0.06655	0.03364
σ^2	0.602252	0.5238	0.438762	0.56025	0.53395

As we might expect, the estimates of τ^2 vary a lot more than the estimates of σ^2.

(c)

```
index = unique(data$county)
sampled_index = sample(index,0.2*length(index))
subfit = lmer(log.radon ~ floor + log.uranium + (1|county),data = data[data$county %in% sampled_index,]

summary(subfit)
Random effects:
 Groups   Name        Variance Std.Dev.
 county   (Intercept) 0.047496 0.21793
 Residual             0.660934 0.81298
Number of obs: 266, groups: county, 17

Fixed effects:
            Estimate Std. Error t value
(Intercept)  1.46784    0.09177  15.994
floor       -0.71208    0.14122  -5.042
log.uranium  0.77388    0.19377   3.994

Correlation of Fixed Effects:
            (Intr) floor
floor       -0.292
log.uranium  0.094 -0.095
```

It seems the fixed effects coefficient estimates from the cluster sample fitted model are closer to the full data. However, the across county variance estimate still seem variable and different from the full data estimate.

Problem Three: G&H Chapter 12, #6

(a)

$$y_{i,j} = \alpha_i + \epsilon_{i,j}, \epsilon_{i,j} \sim N(0, \sigma^2), \alpha_i \sim N(0, \tau^2)$$

i : instructor index, $y_{i,j}$ course evaluation score for jth evaluation and ith instructor

```
fit = lmer(courseevaluation ~ 1 + (1 | profnumber),data = data)
summary(fit)

Formula: courseevaluation ~ 1 + (1 | profnumber)
   Data: data
 AIC   BIC logLik deviance REMLdev
 650 662.5   -322    639.7     644
Random effects:
 Groups     Name        Variance Std.Dev.
 profnumber (Intercept) 0.14703  0.38344
 Residual               0.17022  0.41257
Number of obs: 463, groups: profnumber, 94

Fixed effects:
            Estimate Std. Error t value
(Intercept)  3.93573    0.04519   87.09

fixef(fit)
3.936
ranef(fit)
1   0.049842665
2  -0.290355041
3  -0.307649827
4   0.067064271
5   0.347270330
.....
```

The fixed effect is the estimated mean intercept of all instructors. The random effect tells us how individual score deviates from the mean intercept. For example, the first teacher is 0.05 above the mean.

(b)

$$y_{i,j} = \alpha_i + \epsilon_{i,j}, \alpha_i = \beta_0 + \beta_1 beauty_i + \beta_2 female_i + \beta_3 tenured_i + \eta_i, \epsilon_{i,j} \sim N(0, \sigma^2), \eta_i \sim N(0, \tau^2)$$

```
fit = lmer(courseevaluation ~ 1 + btystdave + female + tenured + (1 | profnumber),data = data)
summary(fit)

Formula: courseevaluation ~ 1 + btystdave + female + tenured + (1 | profnumber)
   Data: data
   AIC   BIC logLik deviance REMLdev
 655.7 680.6 -321.9    629.3   643.7
Random effects:
 Groups     Name        Variance Std.Dev.
 profnumber (Intercept) 0.13234  0.36379
 Residual               0.17027  0.41263
Number of obs: 463, groups: profnumber, 94

Fixed effects:
            Estimate Std. Error t value
(Intercept)  4.06842    0.08415   48.34
btystdave    0.13157    0.05412    2.43
female      -0.22001    0.09157   -2.40
tenured     -0.06179    0.09150   -0.68

Correlation of Fixed Effects:
          (Intr) btystd female
btystdave -0.011
female    -0.604 -0.129
tenured   -0.730  0.131  0.231
```

This is a varying intercept model with the intercept for each group/instructor determined by three group level predictors. For example, the first instructor has an estimated intercept $0.09 + 4.07 + 0.13 * btystdave - 0.22 * female - 0.06 * tenured$

(c) We know that the estimated $\sigma^2 = 0.17, \tau^2 = 0.13$, so we can calculate the intraclass correlation $0.17/(0.17+0.13) = 0.43$. It measure how much information grouping conveys about individuals within the group, with 0 being no information to 1 if all members of a group are identical (and so maximum information).

Problem Four: G&H Chapter 13, #1.

(a) First we can use "lower" or "onecredit" or a combination of them, as a class category variable. For this solution I will use "lower".

A plausible model that retains clustering by professor (which makes sense for the overall problem of relating course ratings in the courses that a professor teaches with his or her overall beauty rating) and allows the coefficient on "lower" to depend on the beauty rating would be

$$\begin{aligned} y_i &= \alpha_{0j[i]} + \alpha_{1j[i]}(lower)_i + \varepsilon_i \ , \ \ \varepsilon_i \sim N(0, \sigma^2) \\ \alpha_{0j} &= \beta_{00} + \beta_{01}(beauty)_j + \eta_{0j} \ , \ \ \eta_{0j} \sim N(0, \tau_0^2) \\ \alpha_{1j} &= \beta_{10} + \beta_{11}(beauty)_j + \eta_{1j} \ , \ \ \eta_{1j} \sim N(0, \tau_1^2) \end{aligned}$$

Substituting the 2nd and 3rd equation into the first, we get the variance components form

$$y_i = \beta_{00} + \beta_{10}(lower)_i + \beta_{01}(beauty)_{j[i]} + \beta_{11}(beauty)_{j[i]}(lower)_i + \eta_{0j[i]} + \eta_{1j[i]}(lower)_i + \varepsilon_i$$

which shows

(i) the intercept $\beta_{00} + \beta_{10}(lower)_i$ does vary by course category;

(ii) the slope on beauty $\beta_{01}(beauty)_{j[i]} + \beta_{11}(beauty)_{j[i]}(lower)_i = (\beta_{01} + \beta_{11}(lower)_i)(beauty)_{j[i]}$ does vary by course category;

(iii) the `lmer` model formula will be `y ~ lower*beauty + (lower|profnumber)`.

(b) Following the model given above,

```
summary(fit <- lmer(courseevaluation ~ lower*btystdave + (lower|profnumber), data=data))
Random effects:
 Groups     Name        Variance Std.Dev. Corr
 profnumber (Intercept) 0.1436   0.379
            lower       0.3410   0.584    -0.63
 Residual               0.1467   0.383
Number of obs: 463, groups:  profnumber, 94

Fixed effects:
                Estimate Std. Error t value
(Intercept)      3.92284    0.04972   78.90
lower            0.08763    0.09292    0.94
btystdave        0.08633    0.06301    1.37
lower:btystdave  0.04085    0.11015    0.37
```

Our estimates are

Parameter	β_{00}	β_{01}	β_{10}	β_{11}	σ^2	τ_0^2	τ_1^2
Estimate	3.92284	0.08763	0.08633	0.04085	0.1467	0.1436	0.3410

The η_{0j}'s are the first column and the η_{1j}'s are the second column of the following:

```
ranef(fit)
$profnumber
     (Intercept)        lower
1   0.0475999598 -0.046461537
2  -0.2373790576  0.231701789
3  -0.2752656779  0.268682295
.
.
.
93  0.2169006210 -0.305841630
94 -0.5107574563  0.533169561
```

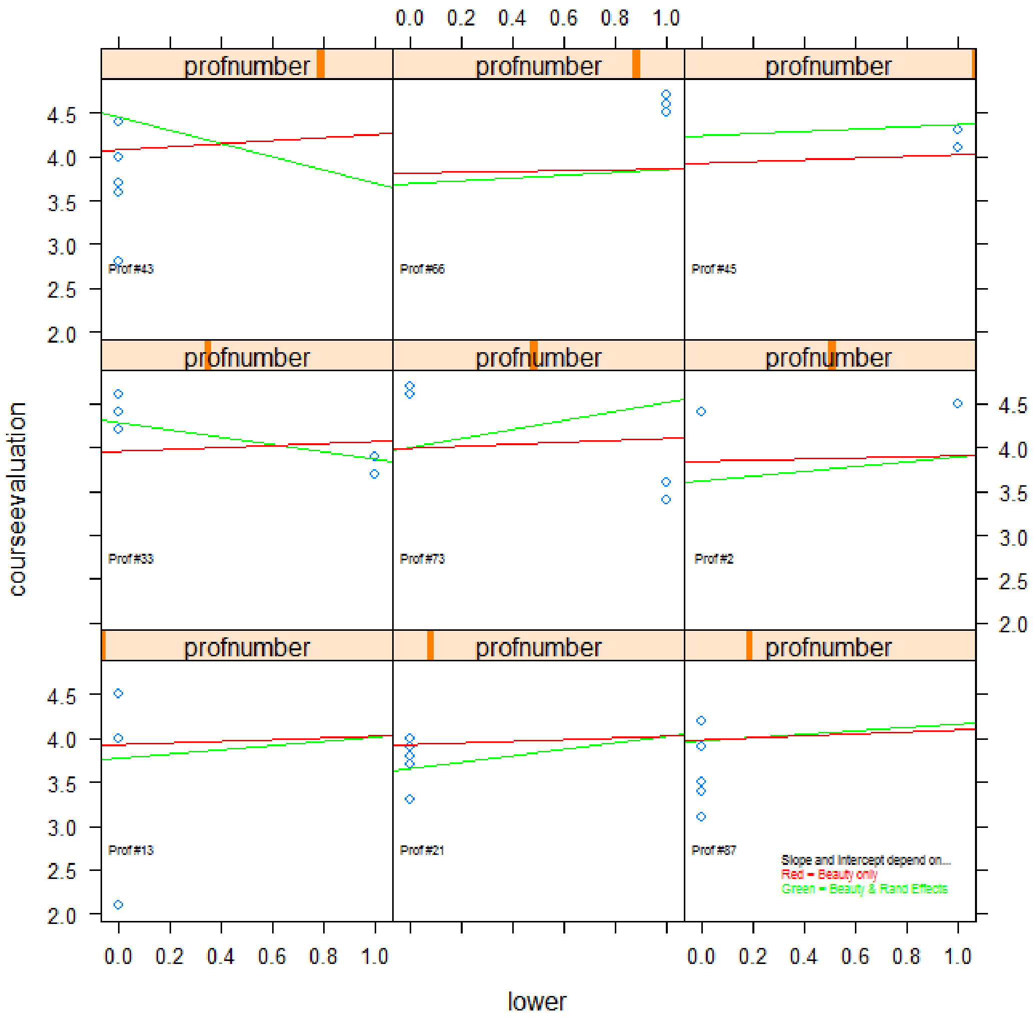
None of the fixed effects are significant in this model, although btystdave nearly is. It would be interesting and worthwhile to augment this model by adding other class-level or professor-level covariates, as a way of understanding better what variables influence course ratings for professors.

(c) Because of the changes in the current version of `ggplot2`, this plot is easier to make using the `xyplot` function in `library(lattice)`...

The basic plot we make will have one subgraph per professor, with “level” on the X axis and course evaluation on the Y axis. The slope and intercept of the line relating “level” to course evaluation will be determined by (1) the beauty rating of that professor, and (2) the random effects in the model.

The next page contains the code to make the plot, and the plot is on the following page after that.

```
attach(data)
profsubset <- sample(sort(unique(profnumber)),9)
  # take 9 professors at random
profbeauty <- btystdave[match(sort(unique(profnumber)), profnumber)]
  # get the unique beauty value for each professor
xyplot(courseevaluation ~ lower | profnumber,
       subset=profnumber %in% profsubset,
       panel=function(x,y) {
         panel.xyplot(x,y)
         j <- profsubset[panel.number()]
         b00 <- fixef(fit)[1]
         b01 <- fixef(fit)[2]
         b10 <- fixef(fit)[3]
         b11 <- fixef(fit)[4]
         eta0 <- ranef(fit)$profnumber[j,1]
         eta1 <- ranef(fit)$profnumber[j,2]
         panel.abline(b00+b10*profbeauty[j]+eta0,
                      b10+b11*profbeauty[j]+eta1,col="Green")
         panel.abline(b00+b10*profbeauty[j],
                      b10+b11*profbeauty[j],col="Red")
         panel.text(0.05,2.75,paste("Prof #",j,sep=""),cex=0.5)
       }
       )
trellis.focus("toplevel")
panel.text(.73,.163,"Slope and intercept depend on...",cex=.5,pos=4)
panel.text(.73,.150,"Red = Beauty only",col="Red",cex=.5,pos=4)
panel.text(.73,.137,"Green = Beauty & Rand Effects",col="Green",cex=.5,pos=4)
trellis.unfocus()
detach(data)
```



Because “lower” can only be 0 (for lower-division classes) and 1 (for upper division classes), the individual course ratings cluster in lines above 0 and 1 in each subplot.

We see from the plot that the beauty rating does affect the slope and intercept for the relationship between “lower” and course evaluation somewhat, and that the random effects can be bigger than the effect of beauty.

Problem Five - G&H 12.2

0.1 A

Write a model predicting CD4 percentage as a function of time with varying intercepts across children. Fit using lmer() and interpret the coefficient for time.

We used ’age’ as a variable for time, since it wasn’t clear what to use for that, and we thought just using the visit date would be inappropriate. We also could have used ”VISIT” which are the visit numbers, but, without knowing anything about visit scheduling, we thought using age would be more appropriate.

```
Linear mixed model fit by REML ['lmerMod']
Formula: CD4PCT ˜ visage + (1 | newpid)
   Data: d

REML criterion at convergence: 7903.5

Scaled residuals:
    Min      1Q  Median      3Q     Max
-4.4747 -0.4553 -0.0562  0.3799  6.6349

Random effects:
 Groups   Name        Variance Std.Dev.
 newpid   (Intercept) 125.5    11.201
 Residual              53.7     7.328
Number of obs: 1075, groups:  newpid, 251

Fixed effects:
            Estimate Std. Error t value
(Intercept)  29.7404     1.3150  22.616
visage       -1.5833     0.2717  -5.827

Correlation of Fixed Effects:
       (Intr)
visage -0.820
```

A lot of the variation here is attributable to patient identity. Different children have very different base CD4 levels. But, within each child, this model says that you can expect CD4 to decrease by 1.6 every year. The decrease over time appears significant, and looks like it should be about 1.3 to 1.9 per year.

0.2 B

Extend your model to include child-level predictors for treatment and age at baseline. Fit using lmer() and interpret the coefficients on time, treatment, and age at baseline.

```
Linear mixed model fit by REML ['lmerMod']
Formula: model
   Data: d

REML criterion at convergence: 7868.2

Scaled residuals:
    Min      1Q  Median      3Q     Max
-4.4799 -0.4553 -0.0554  0.3841  6.8081

Random effects:
 Groups   Name        Variance Std.Dev.
 newpid   (Intercept) 125.23   11.191
 Residual              53.27    7.299
```

```
Number of obs: 1072, groups:  newpid, 250

Fixed effects:
            Estimate Std. Error t value
(Intercept)  26.5005     2.6212  10.110
visage       -2.9639     0.5084  -5.830
baseage       2.0154     0.6118   3.294
treatmnt      1.2092     1.5089   0.801

Correlation of Fixed Effects:
         (Intr) visage baseag
visage   -0.099
baseage  -0.150 -0.841
treatmnt -0.849  0.011 -0.011
```

We first look at whether or not including the additional variables improves our model.

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
m1	4	7911.97	7931.89	-3951.98	7903.97			
m2	6	7884.23	7914.10	-3936.12	7872.23	31.73	2	0.0000

It looks like including the group level (i.e. child-level) variables here definitely improves our model.
Our interpretations for each coefficient are:

- for time, ”visage”, it looks like the decrease in CD4 over time has become more substantial, now that we have included other child-level variables. We should expect a decrease of about 2.5 to 3.5 in CD4 each year, in any given child.

- for treatment, we estimated an increase in CD4 of 1.2. But I wouldn’t trust this very much, considering the standard error, the increase is closer to -0.3 to 2.8. It’s hard to tell if the treatment impacted CD4 at all using this model.

- for base age, it looks like children who were included in the study earlier have higher CD4. I’m somewhat skeptical of this estimate, since base age ought to be somewhat associated with ’treatment’. A child who has been on treatment longer ought to have different CD4 from someone who just was put on treatment. We should expect some kind of interaction between (visit age - base age)*(treatment), rather than a simple effect from treatment, if time on treatment matters at all.

0.3 C

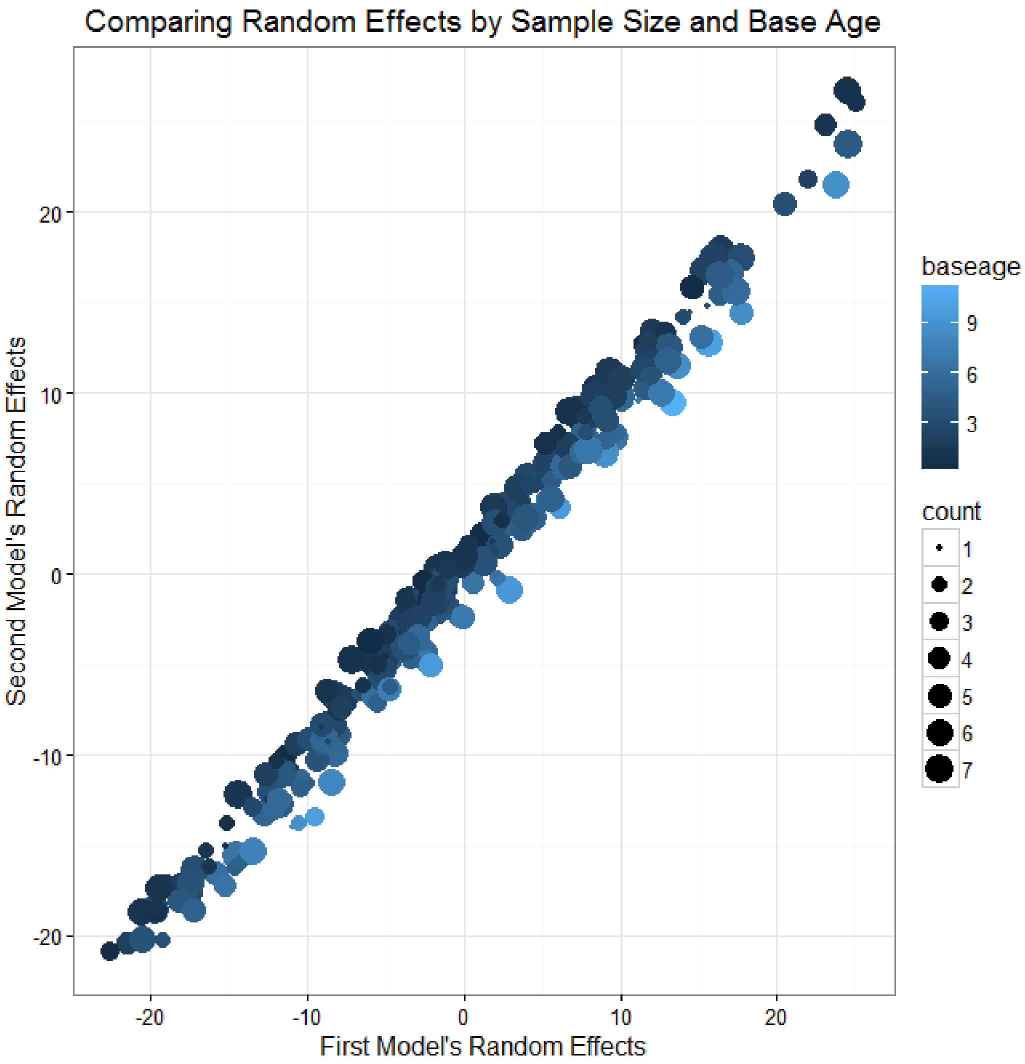
Investigate the change in partial pooling from (A) to (B) both graphically and numerically.

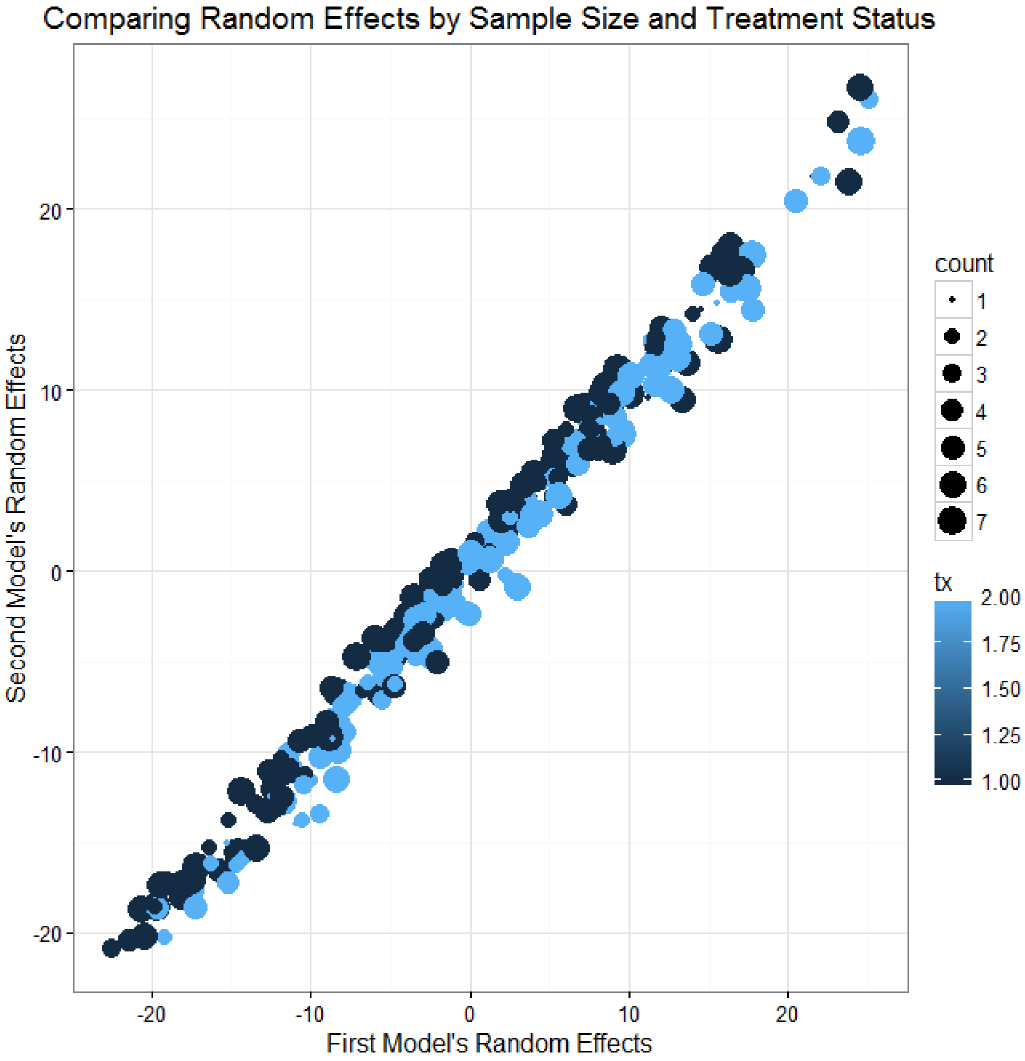
As an example, we plot the random effects from the first model against those in the second model, sized by the number of observations in each group (i.e. for each child.) The first plot colors the plotted points by base age; and the second colors the plotted points by treatment vs. control.

In this case, it was easier to do the plots using ggplot2 (though the 2nd plot could be cleaned up somewhat since there are only two colors that matter, darkest blue for tx=1 and lightest blue for tx=2).

The code is on the next page, and the plots are on the pages following that.

```
reduced.data <- data[with(data,!is.na(visage+baseage+treatmnt)),]
## ensuring that the two models are fitted to exactly the same data sets...

summary(fit.a <- lmer(CD4PCT ˜ visage + (1 | newpid),data=reduced.data))

summary(fit.b <- lmer(CD4PCT ˜ visage + baseage + treatmnt + (1 | newpid),
                      data=reduced.data))

plotdata <- data.frame(mod1=ranef(fit.a)[[1]][,1],mod2=ranef(fit.b)[[1]][,1],
                       count=as.vector(table(reduced.data$newpid)),
                       baseage=sapply(split(reduced.data$baseage,
                                            reduced.data$newpid),
                                            function(x) x[1]),
                       tx=sapply(split(reduced.data$treatmnt,
                                       reduced.data$newpid),
                                       function(x) x[1]))

ggplot(plotdata,aes(x=mod1,y=mod2)) +
    geom_point(aes(size=count,color=baseage)) +
    xlab("First Model’s Random Effects") +
    ylab("Second Model’s Random Effects") +
    ggtitle("Comparing Random Effects by Sample Size and Base Age")

ggplot(plotdata,aes(x=mod1,y=mod2)) +
    geom_point(aes(size=count,color=tx)) +
    xlab("First Model’s Random Effects") +
    ylab("Second Model’s Random Effects") +
    ggtitle("Comparing Random Effects by Sample Size and Treatment Status")
```


Comparing Random Effects by Sample Size and Base Age
Second Model's Random Effects
First Model's Random Effects
20
10
0
-10
-20
-20
-10
0
10
20
baseage
9
6
3
count
1
2
3
4
5
6
7



It looks like the random effects in model 1 were incorporating some information from ’treatment’ and ’base age’. Generally the random effects are very consistent from model to model.

Numerically, the pooling does not seem very different. We see variance in the random effects of 125.5 and 125.2, which are not very different.