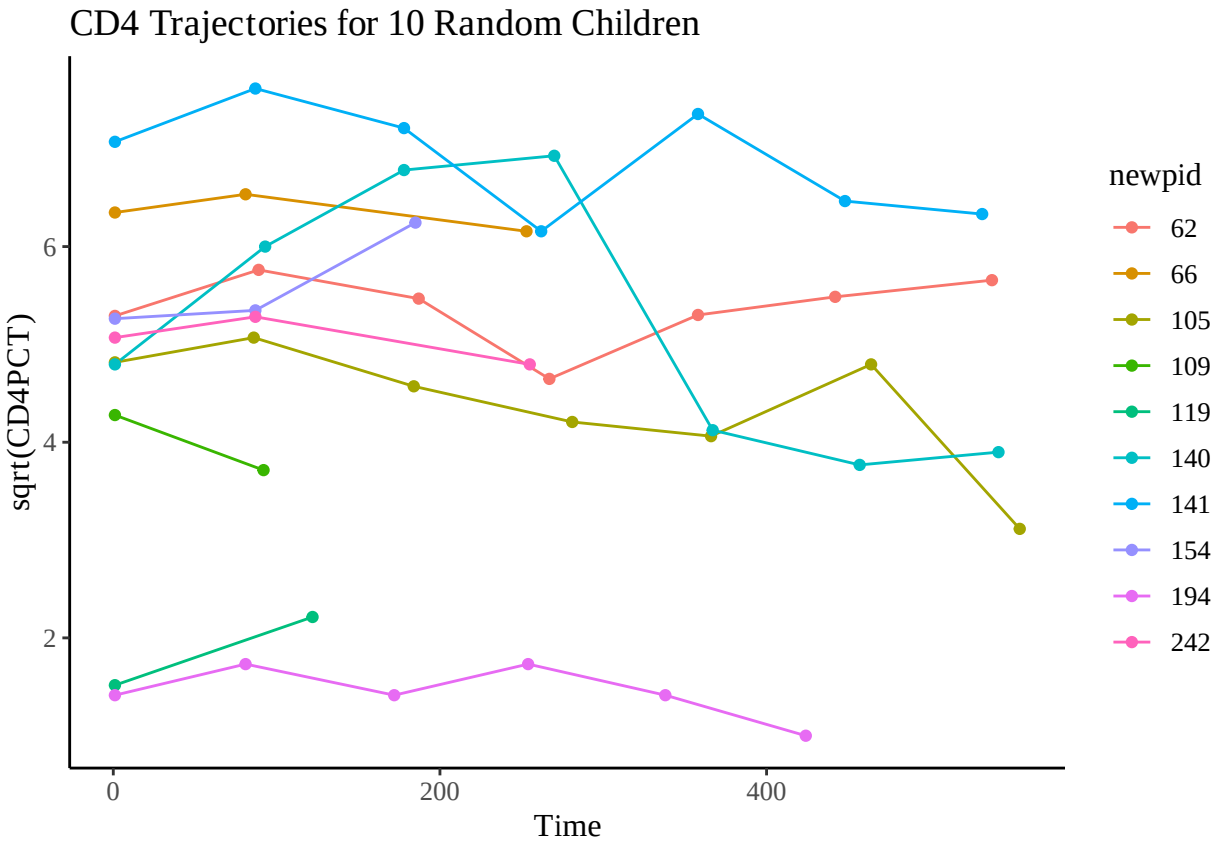
PHP 2517 Homework #1

Blain Morin

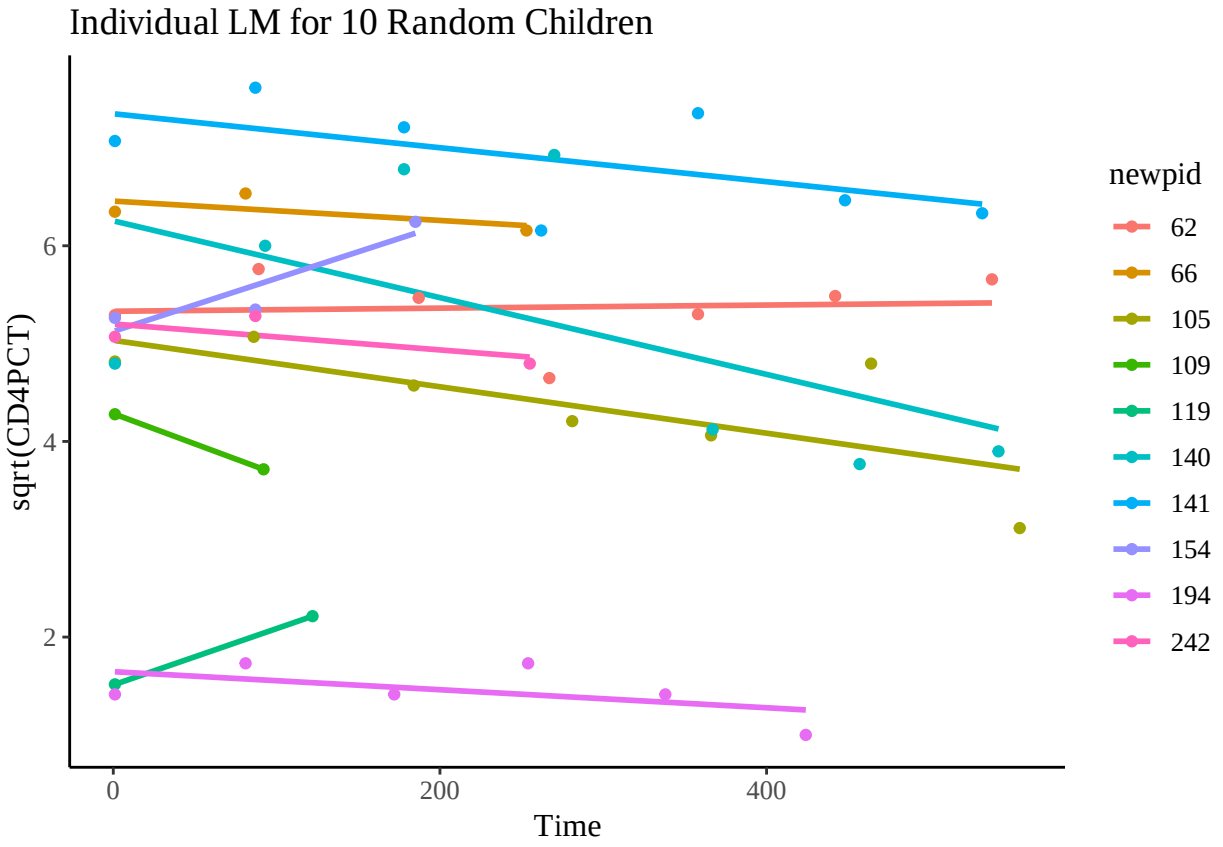
February 11, 2019

Q1 GH Chapter 11: Exercise 4

a.) Graph the outcome (the CD4 percentage, on the square root scale) for 10 children as a function of time.

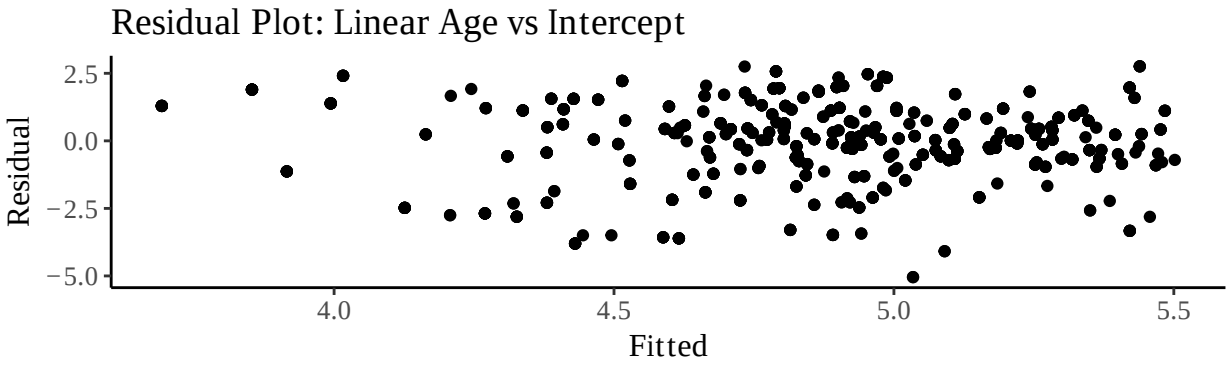


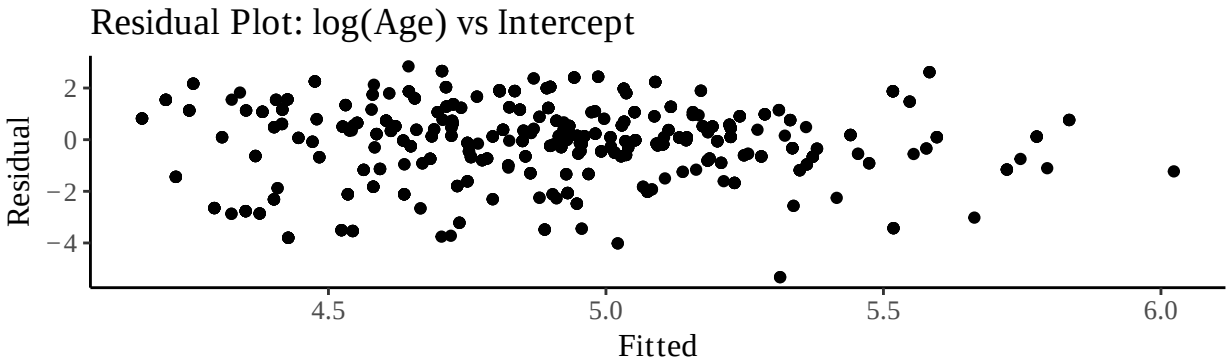
b.) Each child's data has a time course that can be summarized by a linear fit. Estimate these lines and plot them for 10 children.

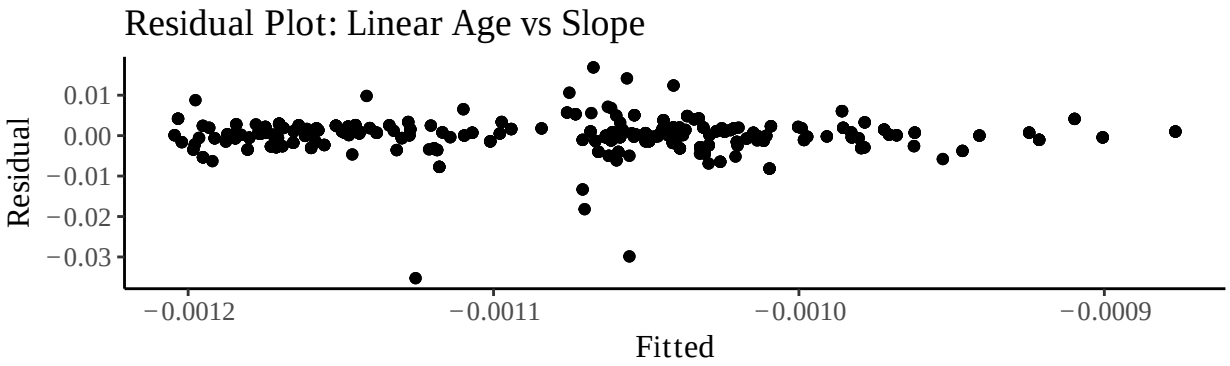


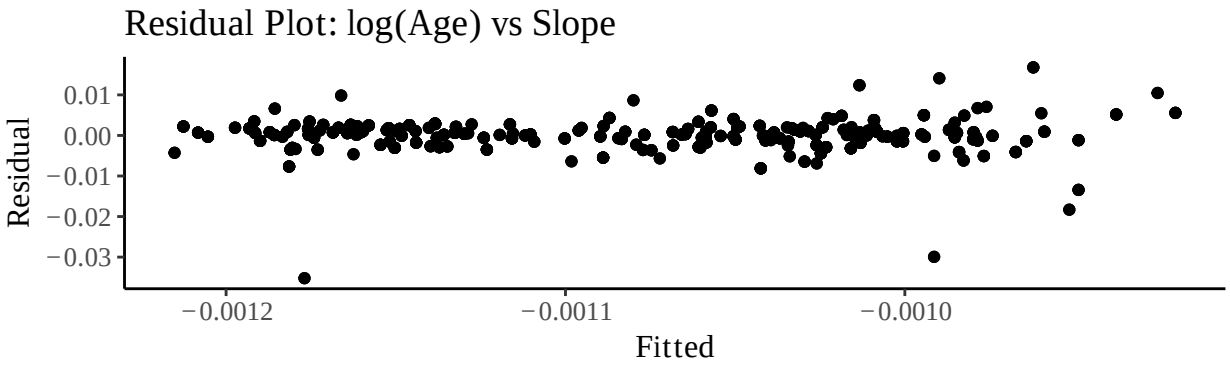
c.) Set up a model for the children's slopes and intercepts as a function of the treatment and age at baseline. Estimate this model using the two-step procedure–first estimate the intercept and slope separately for each child, then fit the between-child models using the point estimates from the first step.

We ran regressed each child's intercept and slope on their treatment and age at baseline. We also checked the log transformation for age at baseline. Here are residual plots for each of the four regressions:









We see that the log transformation of baseline age is a slight improvement for both the intercept and slope regressions. For the intercept plots, the log transformation spreads the fitted values more evenly along the x axis. For the slope plots, the log transformation of basline age seems to help with the heteroskedasticity seen in the linear residual plot.

Here are our regression results for the intercept and slope using the log transformation for baseline age:

Table 1: Regression on Intercept and Slope

	Dependent variable:	
	beta0	beta1
	(1)	(2)
Treatment = 2	0.451***	−0.0001
	(0.092)	(0.0003)
log(baseage)	−0.349***	−0.0001
	(0.065)	(0.0002)
Constant	5.014***	−0.001***
	(0.089)	(0.0002)
Observations	960	936
R^2	0.052	0.0004
Adjusted R^2	0.050	−0.002
Residual Std. Error	1.428 (df = 957)	0.004 (df = 933)
F Statistic	26.155*** (df = 2; 957)	0.170 (df = 2; 933)
Note:		*p<0.1; **p<0.05; ***p<0.01
	Individuals with only one observation do not have a slope estimate.	

We see that treatment and baseline age do not have a significant effect on the slope of the regression, but they do have a significant effect on the intercept.

- On average, being in treatment 2 is associated with a regression intercept that is .451 higher than being in treatment 1, all else equal.
- On average, a 1% increase in baseage is associated with a regression intercept that is $.349 * log(1.01) = .003$ lower, all else equal.

Q2 GH Chapter 12: Exercise 2

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.

Level 1 decribes the within child variability:

$$Level1 : \sqrt{CD4PCT_{ij}} \sim \mathcal{N}(\alpha_j + \beta time_{ij}, \sigma^2)$$

Level 2 decribes the between children variability:

$$Level2 : \alpha_j \sim \mathcal{N}(\mu, \tau^2)$$

Here is the regression result:

```
### Run random intercept model a
rand.int.a = lmer(sqrt(CD4PCT) ~ time + (1 | newpid), data = cd4)

### Make a table
```

```
texreg(rand.int.a, caption.above = TRUE,
       caption = "Random Intercept Model",
       custom.model.names = "sqrt(CD4PCT)",
       float.pos = 'H',
       digits = 3)
```

Table 2: Random Intercept Model

	sqrt(CD4PCT)
(Intercept)	4.796***
	(0.103)
time	−0.001***
	(0.000)
AIC	2825.035
BIC	2844.503
Log Likelihood	-1408.518
Num. obs.	960
Num. groups: newpid	221
Var: newpid (Intercept)	1.981
Var: Residual	0.592

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

The coefficient on time, β, is -.001. This means that for each additional day, we expect the square root of CD4 percentage to decrease by .001.

b.) Extend the model in (a) to include child-level predictors (that is, group-level predictors) for treatment and age at baseline. Fit using lmer() and interpret the coefficients on time, treatment, and age at baseline.

Again, level 1 describes the within child variability:

$$Level1 : CD4PCT_{ij} \sim \mathcal{N}(\alpha_j + \beta time_{ij}, \sigma^2)$$

Level 2 describes the between children variability (now includes group level predictors):

$$Level2 : \alpha_j \sim \mathcal{N}(\mu + \gamma_0 Treat + \gamma_1 log(baseage), \tau^2)$$

Here is the code we used to fit the model:

```
### Run random intercept model b
rand.int.b = lmer(sqrt(CD4PCT) ~ time + treatmnt + log(baseage) + (1|newpid), data = cd4)

### Make a table
texreg(rand.int.b, caption.above = TRUE,
       caption = "Random Intercept Model with Child Level Predictors",
       custom.model.names = "sqrt(CD4PCT)",
       float.pos = 'H',
       digits = 3)
```

Table 3: Random Intercept Model with Child Level Predictors

	sqrt(CD4PCT)
(Intercept)	4.888***
	(0.181)
time	−0.001***
	(0.000)
treatmnt2	0.321
	(0.196)
log(baseage)	−0.263*
	(0.128)
AIC	2825.901
BIC	2855.103
Log Likelihood	-1406.951
Num. obs.	960
Num. groups: newpid	221
Var: newpid (Intercept)	1.926
Var: Residual	0.593

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

- The coefficient on time (β) is -.001. This means that we expect the square root of CD4% to decrease by .001 for each additional day on average, all else equal.
- The coefficeint on treatmnt2 (γ_0) is .32. This means that we expect the square root of CD4% to be .32 higher for the children in treatment group 2 than the children in treatment group 1 on average, all else equal.
- The coefficient on log(baseage) (γ_1) is -.263. This means that for a 1% increase in baseage, we expect the square root of CD4% to decrease by ($.263 * log(1.01) = .0026$).

c.) Investigate the change in partial pooling from (a) to (b) both graphically and numerically.

Since treatment and age at baseline are group level predictors, we expect that they may help explain some of the between group variation and leave the within group variation unchanged. First, we numerically compare the differences in the estimated within group variance (σ^2) and between group variance (τ^2) between model (a) and model (b):

```
### Get model summaries
a.model = summary(rand.int.a)

b.model = summary(rand.int.b)

### Extract sigma and tau
a.vars = round(c(sigma = a.model$sigma,
           tau = a.model$varcor[[1]][1]), 3)

b.vars = round(c(sigma = b.model$sigma,
           tau = b.model$varcor[[1]][1]), 3)


### Make a table
c.data = bind_rows(a.vars, b.vars)

rownames(c.data) = c("Model A", "Model B")
```

```
stargazer(c.data, header = FALSE,
          summary = FALSE,
          title = "Sources of Variation")
```

Table 4: Sources of Variation

	sigma	tau
Model A	0.77	1.981
Model B	0.77	1.926

As expected, we see that the within child standard error (σ) is basically unchanged between model (a) and model (b). We see that there is a slight reduction in the between children standard errors (τ), from 1.981 to 1.926. Thus, the group level predictors(treatment and age at baseline), explain a small amount of the variation between children.

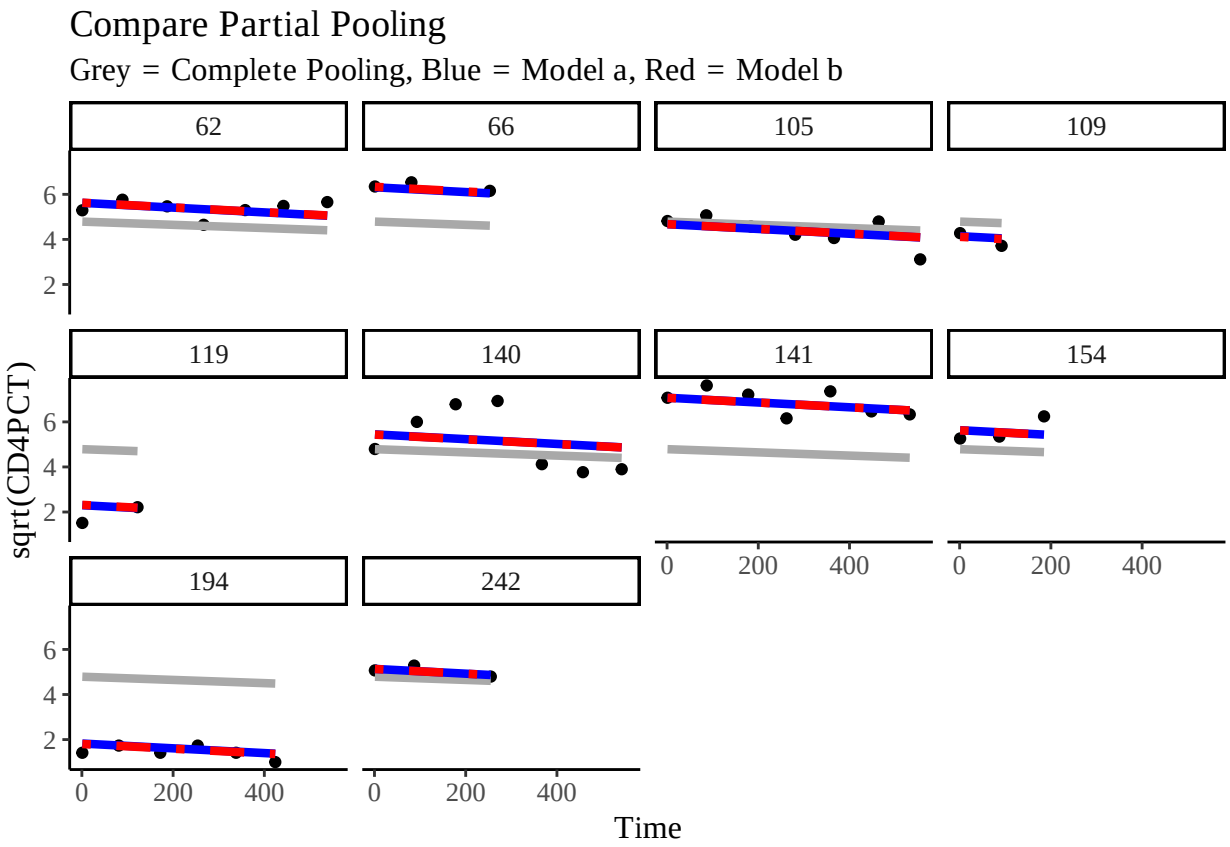
We also look at the difference in partial pooling graphically:

```
### Complete pooling case
complete.pooling = lm(sqrt(CD4PCT) ~ time, data = cd4)

### Get fitted values
cd4$complete.pooling = predict(complete.pooling)
cd4$model.a = predict(rand.int.a)
cd4$model.b = predict(rand.int.b)

### Get data for 10 IDs
ten.kid.data2 = cd4 %>%
  filter(newpid %in% as.numeric(as.matrix(ten.kid.ids))) %>%
  mutate(newpid = as.factor(newpid))

### Plot for 10 random kids
ten.kid.data2 %>%
  ggplot(aes(x = time)) +
  facet_wrap(~newpid) +
  geom_point(aes(y = sqrt(CD4PCT))) +
  geom_line(aes(y = complete.pooling), color = 'darkgrey', size = 1.5) +
  geom_line(aes(y = model.a), color = 'blue', linetype = 'solid', size = 1.5) +
  geom_line(aes(y = model.b), color = 'red', linetype = 'dotdash', size = 1.5) +
  xlab("Time") +
  ggtitle("Compare Partial Pooling") +
  labs(subtitle = "Grey = Complete Pooling, Blue = Model a, Red = Model b") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"))
```



Graphically, we see that there is not much difference between the red and blue lines. Thus, there is not much difference in the partial pooling. This confirms what we observed from the small change in τ seen in Table 4.

Q3 GH Chapter 12: Exercise 5

Using the radon data, include county sample size as a group-level predictor and write the varying-intercept model. Fit this model using lmer().

Level 1 describes the within county variation:

$$Level1 : log(radon)_{ij} \sim \mathcal{N}(\alpha_j, \sigma^2)$$

Level 2 describes the between county variation:

$$Level2 : \alpha_j \sim \mathcal{N}(\mu + \gamma countysamples_j, \tau^2)$$

Here is the code and a table of the regression results:

```
### Read in data
srrs2 = read.table ("srrs2.dat", header=T, sep=",")

### Get only MN
### Then transform radon into log radon
```

```r
### Then count observations in each county
mn = srrs2 %>%
  filter(state2 == "MN") %>%
  mutate(log.radon = log(ifelse (activity==0, .1, activity))) %>%
  group_by(county) %>%
  mutate(county.samples = n()) %>%
  ungroup()


### Specify model
q3 = lmer(log.radon ~ county.samples + (1 | county), data = mn)

### Output a table
texreg(q3, caption.above = TRUE,
       caption = "Random Intercept Model with County Level Predictor",
       custom.model.names = "log(radon)",
       float.pos = 'H',
       digits = 3)
```

Table 5: Random Intercept Model with County Level Predictor

	log(radon)
(Intercept)	1.329***
	(0.060)
county.samples	−0.005**
	(0.002)
AIC	2951.235
BIC	2971.555
Log Likelihood	-1471.618
Num. obs.	1188
Num. groups: county	96
Var: county (Intercept)	0.127
Var: Residual	0.639

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

Bonus

a.)

$$var(\bar{Y}) = var(1/N * \sum_{j=1}^{J}\sum_{i=1}^{n} Y_{ij})$$

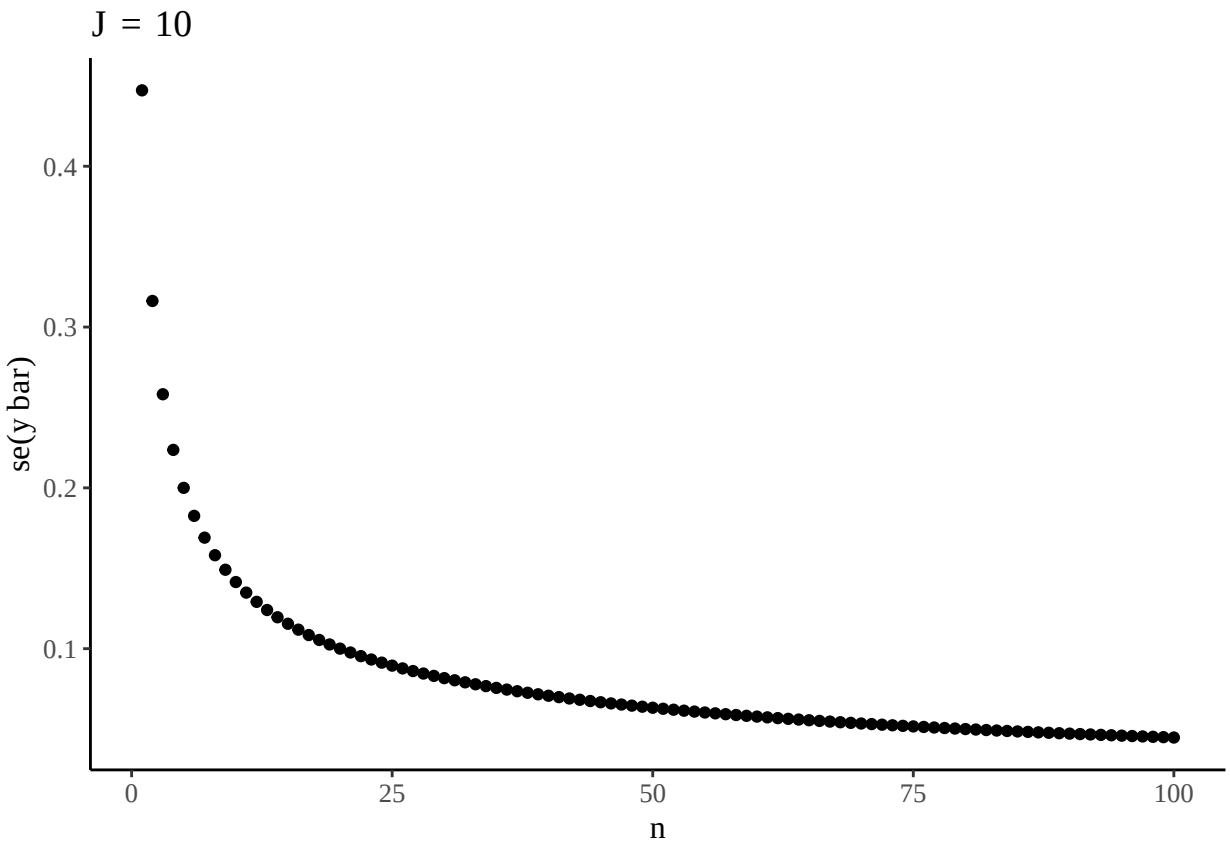
$$= 1/N^2 * \sum_{j=1}^{J}\sum_{i=1}^{n} var(Y_{ij})$$

$$= 1/N^2 * N(\sigma^2 + \tau^2)$$

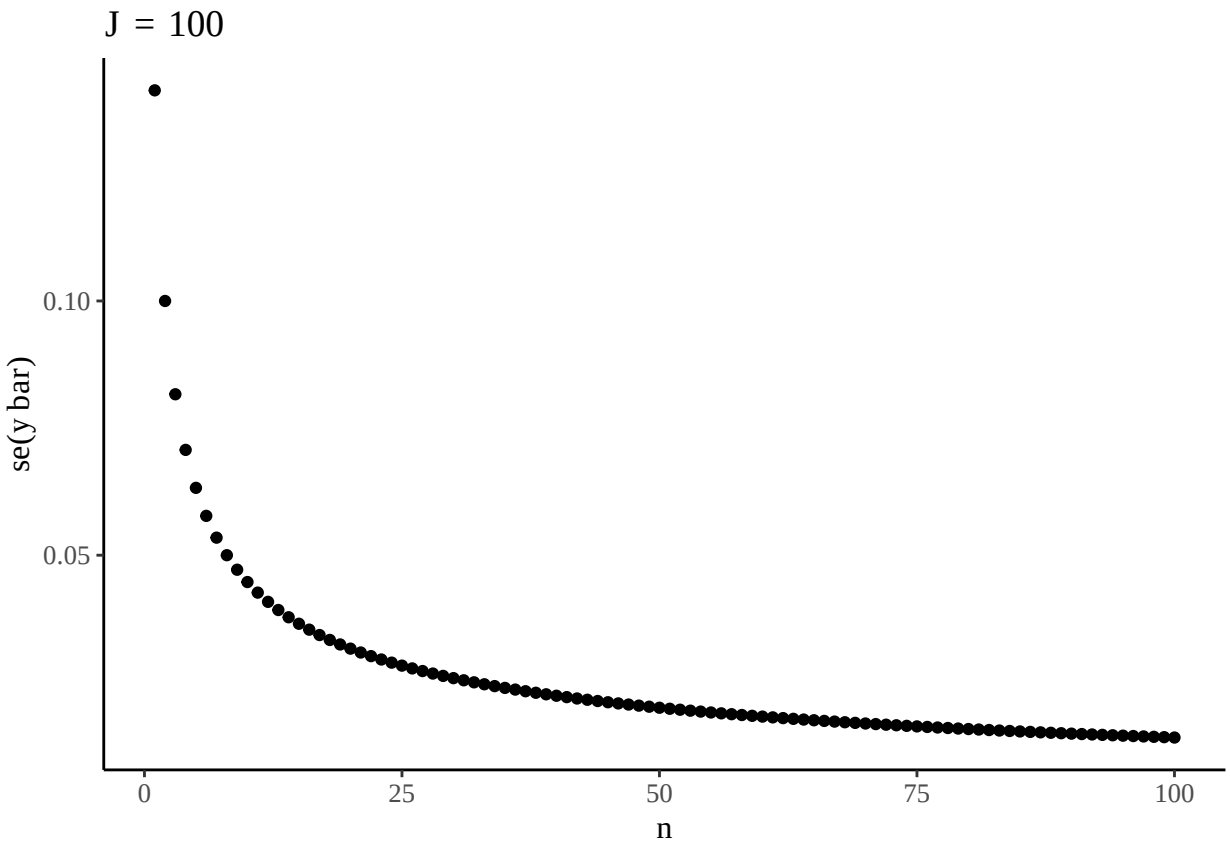
$$= \frac{(\sigma^2 + \tau^2)}{nJ}$$

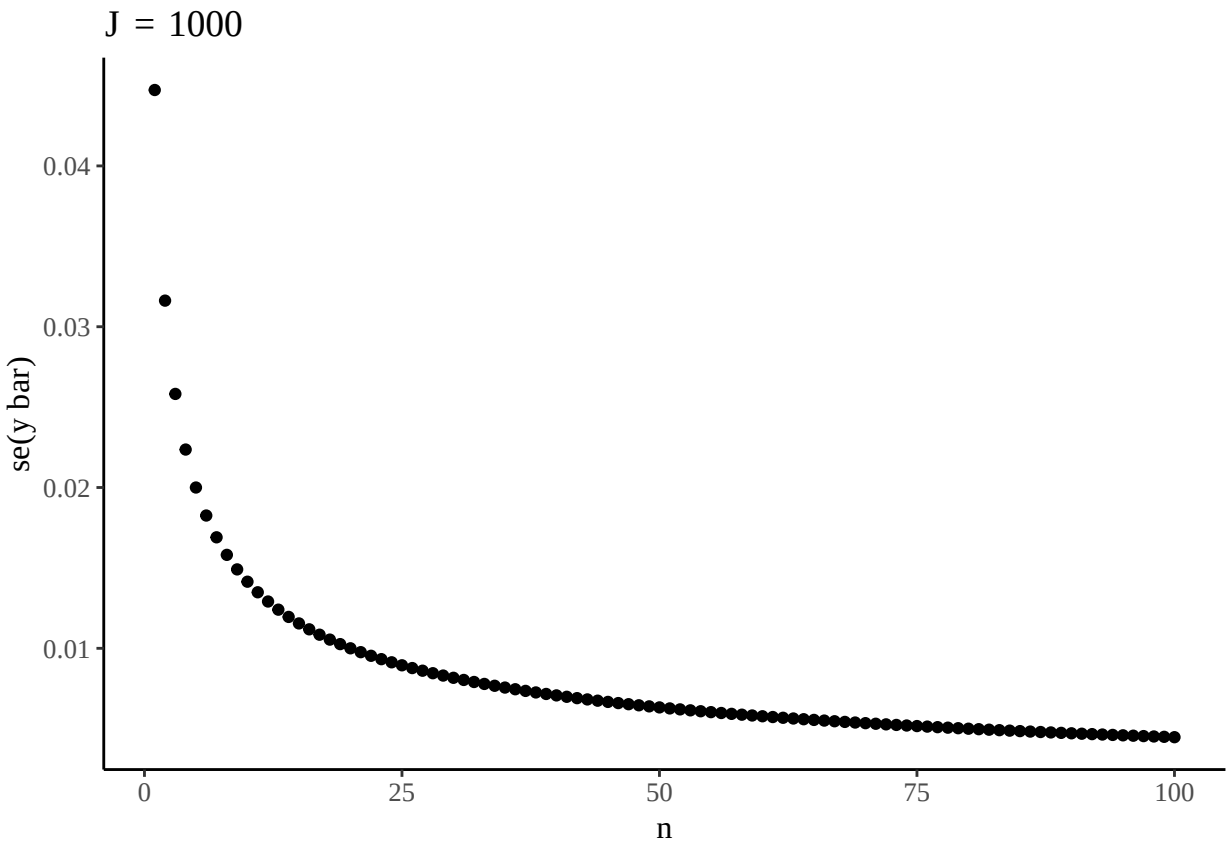
$$s.e.(\bar{Y}) = \sqrt{\frac{(\sigma^2 + \tau^2)}{nJ}}$$

b.)



c.)





d.)

We see that the the standard error of the sample mean decreases at a decreasing rate as the number of observations within each cluster (n) increases. The same is true of the number of clusters (J).

Appendix: R Code

```
### Set knit options
knitr::opts_chunk$set(echo = FALSE, warning = FALSE, message = FALSE, cache = TRUE)


### Load required libraries
library(lme4)
library(ggplot2)
library(arm)
library(dplyr)
library(stargazer)
library(readr)
library(extrafont)
library(grid)
library(gridExtra)
library(sjPlot)
library(texreg)
```

```r
### Load Data
cd4 = read_csv("cd4.csv")

### Set Seed
set.seed(121)


### Get 10 random IDs
ten.kid.ids = cd4 %>%
  select(newpid) %>%
  distinct() %>%
  sample_n(size = 10)

### Get data for 10 IDs
ten.kid.data = cd4 %>%
  filter(newpid %in% as.numeric(as.matrix(ten.kid.ids))) %>%
  mutate(newpid = as.factor(newpid))

### Plot for 10 random individuals
ten.kid.data %>%
  ggplot(aes(x = time, y = sqrt(CD4PCT))) +
  geom_line(aes(group = newpid, color = newpid)) +
  geom_point(aes(color = newpid)) +
  ylab("sqrt(CD4PCT)") +
  xlab("Time") +
  ggtitle("CD4 Trajectories for 10 Random Children") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"))




### Plot individually fitted LMs
ten.kid.data %>%
  ggplot(aes(x = time, y = sqrt(CD4PCT))) +
  stat_smooth(aes(group = newpid, color = newpid), method = "lm", se = FALSE) +
  geom_point(aes(color = newpid)) +
  ylab("sqrt(CD4PCT)") +
  xlab("Time") +
  ggtitle("Individual LM for 10 Random Children") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"))


### Change treatment to factor
cd4 = cd4 %>%
  mutate(treatmnt = as.factor(treatmnt))

cd4 = cd4[complete.cases(cd4),]

### Function that gets intercepts from LM
get.ints = function(x, y) {
```

```
  model.i = lm(y ~ x)
  return(coef(model.i)[1])

}


### Function that gets slopes from LM
get.slopes = function(x, y) {

  model.i = lm(y ~ x)
  return(coef(model.i)[2])

}

### Get ints and slopes for each individual regression
cd4 = cd4 %>%
  group_by(newpid) %>%
  mutate(beta0 = get.ints(x = time, y = sqrt(CD4PCT))) %>%
  mutate(beta1 = get.slopes(x = time, y = sqrt(CD4PCT))) %>%
  ungroup()

### Regress intercept on tx and age
b0.model = lm(beta0 ~ treatmnt + baseage, data = cd4)
b0.model1 = lm(beta0 ~ treatmnt + log(baseage), data = cd4)


### REsidual plots for intercept regression
b0.resid.plot = ggplot(b0.model) +
  geom_point(aes(x = .fitted, y = .resid)) +
  xlab("Fitted") +
  ylab("Residual") +
  ggtitle("Residual Plot: Linear Age vs Intercept") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"))


b0.resid.plot1 = ggplot(b0.model1) +
  geom_point(aes(x = .fitted, y = .resid)) +
  xlab("Fitted") +
  ylab("Residual") +
  ggtitle("Residual Plot: log(Age) vs Intercept") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"))



### Regress slope on tx and age
b1.model = lm(beta1 ~ treatmnt + baseage, data = cd4)
b1.model1 = lm(beta1 ~ treatmnt + log(baseage), data = cd4)


### Residual plots for slope
b1.resid.plot = ggplot(b1.model) +
```

```
  geom_point(aes(x = .fitted, y = .resid)) +
  xlab("Fitted") +
  ylab("Residual") +
  ggtitle("Residual Plot: Linear Age vs Slope") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"))


b1.resid.plot1 = ggplot(b1.model1) +
  geom_point(aes(x = .fitted, y = .resid)) +
  xlab("Fitted") +
  ylab("Residual") +
  ggtitle("Residual Plot: log(Age) vs Slope") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"))

### Arrange plots
grid.arrange(b0.resid.plot,
             b0.resid.plot1,
             b1.resid.plot,
             b1.resid.plot1,
             nrow = 4)

stargazer(b0.model1, b1.model1,
          header = FALSE,
          table.placement = 'H',
          title = "Regression on Intercept and Slope",
          covariate.labels = c("Treatment = 2",
                               "log(baseage)"),
          notes = "Individuals with only one observation do not have a slope estimate.")


### Run random intercept model a
rand.int.a = lmer(sqrt(CD4PCT) ~ time + (1 | newpid), data = cd4)

### Make a table
texreg(rand.int.a, caption.above = TRUE,
       caption = "Random Intercept Model",
       custom.model.names = "sqrt(CD4PCT)",
       float.pos = 'H',
       digits = 3)


### Run random intercept model b
rand.int.b = lmer(sqrt(CD4PCT) ~ time + treatmnt + log(baseage) + (1|newpid), data = cd4)

### Make a table
texreg(rand.int.b, caption.above = TRUE,
       caption = "Random Intercept Model with Child Level Predictors",
       custom.model.names = "sqrt(CD4PCT)",
       float.pos = 'H',
       digits = 3)
```

```r
### Get model summaries
a.model = summary(rand.int.a)

b.model = summary(rand.int.b)

### Extract sigma and tau
a.vars = round(c(sigma = a.model$sigma,
           tau = a.model$varcor[[1]][1]), 3)

b.vars = round(c(sigma = b.model$sigma,
           tau = b.model$varcor[[1]][1]), 3)


### Make a table
c.data = bind_rows(a.vars, b.vars)

rownames(c.data) = c("Model A", "Model B")

stargazer(c.data, header = FALSE,
          summary = FALSE,
          title = "Sources of Variation")


### Complete pooling case
complete.pooling = lm(sqrt(CD4PCT) ~ time, data = cd4)

### Get fitted values
cd4$complete.pooling = predict(complete.pooling)
cd4$model.a = predict(rand.int.a)
cd4$model.b = predict(rand.int.b)

### Get data for 10 IDs
ten.kid.data2 = cd4 %>%
  filter(newpid %in% as.numeric(as.matrix(ten.kid.ids))) %>%
  mutate(newpid = as.factor(newpid))

### Plot for 10 random kids
ten.kid.data2 %>%
  ggplot(aes(x = time)) +
  facet_wrap(~newpid) +
  geom_point(aes(y = sqrt(CD4PCT))) +
  geom_line(aes(y = complete.pooling), color = 'darkgrey', size = 1.5) +
  geom_line(aes(y = model.a), color = 'blue', linetype = 'solid', size = 1.5) +
  geom_line(aes(y = model.b), color = 'red', linetype = 'dotdash', size = 1.5) +
  xlab("Time") +
  ggtitle("Compare Partial Pooling") +
  labs(subtitle = "Grey = Complete Pooling, Blue = Model a, Red = Model b") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"))



### Read in data
```

```
srrs2 = read.table ("srrs2.dat", header=T, sep=",")

### Get only MN
### Then transform radon into log radon
### Then count observations in each county
mn = srrs2 %>%
  filter(state2 == "MN") %>%
  mutate(log.radon = log(ifelse (activity==0, .1, activity))) %>%
  group_by(county) %>%
  mutate(county.samples = n()) %>%
  ungroup()


### Specify model
q3 = lmer(log.radon ~ county.samples + (1 | county), data = mn)

### Output a table
texreg(q3, caption.above = TRUE,
       caption = "Random Intercept Model with County Level Predictor",
       custom.model.names = "log(radon)",
       float.pos = 'H',
       digits = 3)


n = 1:100
J = 10
se.ybar = sqrt(2/(n*J))
b.graph.data = data.frame(n = n, se = se.ybar)

b.graph.data %>%
  ggplot(aes(x = n, y = se)) +
  geom_point() +
  xlab("n") +
  ylab("se(y bar)") +
  ggtitle("J = 10") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"))



n = 1:100
J = 100
se.ybar = sqrt(2/(n*J))
b.graph.data = data.frame(n = n, se = se.ybar)

b.graph.data %>%
  ggplot(aes(x = n, y = se)) +
  geom_point() +
  xlab("n") +
  ylab("se(y bar)") +
  ggtitle("J = 100") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"))
```

```r
n = 1:100
J = 1000
se.ybar = sqrt(2/(n*J))
b.graph.data = data.frame(n = n, se = se.ybar)

b.graph.data %>%
  ggplot(aes(x = n, y = se)) +
  geom_point() +
  xlab("n") +
  ylab("se(y bar)") +
  ggtitle("J = 1000") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"))
```