Homework 08 Solutions

2022-11-14

```
library(arm)
library(ggplot2)
library(HLMdiag)    ## easy extraction of MLM residuals
```

```
## Warning: package 'HLMdiag' was built under R version 4.1.2
```

```
library(gridExtra)  ## For the grid.arrange() function

library(cAIC4)      ## I've asked you not to bother with cAIC for this assignment, but since
```

```
## Warning: package 'cAIC4' was built under R version 4.1.2
```

```
                    ## it is on the assignment I thought it would be useful for you to see it.
                    ## This library provides a function cAIC() for calculating it.
```

Problem 1.

The file allvars.csv contains CD4 percentages (CD4PCT) for a set of 254 young children with HIV who were measured several times over a period of two years (A CD4 count measures the number of CD4 cells in your blood. It's used to check the immune system function in people with HIV https://medlineplus.gov/lab-tests/cd4-lymphocyte-count/]). The dataset also includes the ages of the children at each measurement. This is an example of *growth curve data* in which the Level 1 observations are CD4PCT's at each visit, and the Level 2 "group'' is a child. How the CD4PCT's change over time for each child is the primary question of interest. Because of skewing, you should replace CD4PCT with sqrt(CD4PCT) for all of the following questions.

Problem 1(a).

For the first 12 children only (`newpid` $\leq$ `12`), make a facet plot of CD4PCT vs VISIT number. Use different plotting point colors for children who have treatmnt=1 and treatmnt=0 (Presumably, "treatmnt'' stands for some sort of treatment to improve childrens' HIV status.). Do the same for CD4PCT vs. visage (the age at the child on each visit). (*Note: If you are using `ggplot()`, you may wish to set `scales="free_x"` in the `facet_wrap()` function.*) Which is a better measure of time?

First we read in the data and do a little exploring and munging...

```
cd4 <- read.csv("allvar.csv",header=T)

## It's worth doing a little exploration of the data here,
## especially to look at missing values.

apply(cd4,2,function(x) mean(is.na(x)))
```

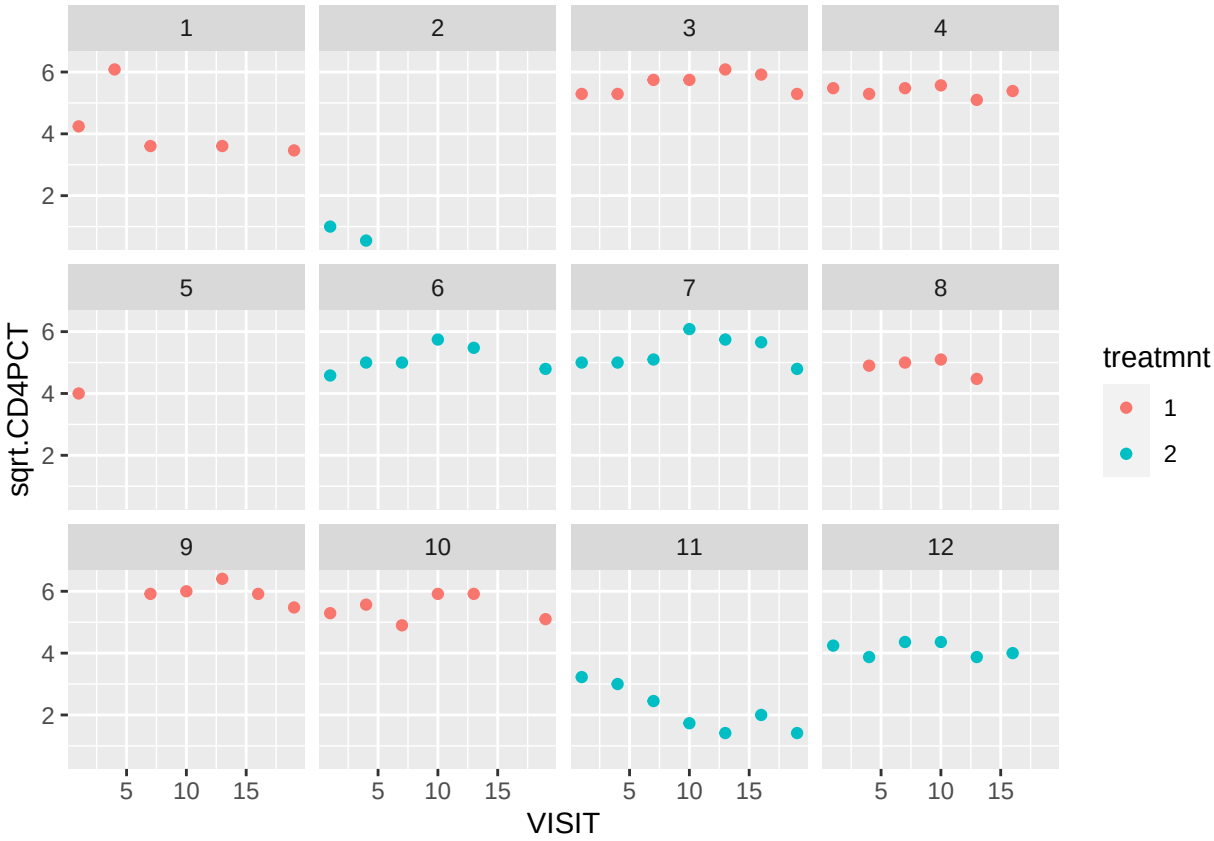
```
##       VISIT      newpid       VDATE      CD4PCT         arv      visage
## 0.000000000 0.000000000 0.000000000 0.142743222 0.103668262 0.109250399
##    treatmnt      CD4CNT     baseage
## 0.000000000 0.146730463 0.007177033
```

```
## we only need the variables newpid, VISIT, visage, CD4PCT and treatmnt
## so let's get rid of some (but not all) missing data problems
## by deleting the other variables

cd4 <- with(cd4,data.frame(newpid=newpid,
                           VISIT=VISIT,
                           visage=visage,
                           CD4PCT=CD4PCT,
                           treatmnt=treatmnt))

## and since we still have some missing data, we will just
## delete the rows that continue to have NA's (not a great
## practice in general, but good enough for this exercise...)
cd4 <- cd4[!apply(cd4,1,function(x) any(is.na(x))),]

## set up the sqrt of the response variable...
cd4$sqrt.CD4PCT <- sqrt(cd4$CD4PCT)

## and for coloring ggplot elements it will better if treatmnt is a factor...
cd4$treatmnt <- as.factor(cd4$treatmnt)

## select the first 12 kids...
first.12 <- (cd4$newpid <= 12)
```

I will show the faceted graph for both `VISIT` *and* `visage` *as the time variable...*

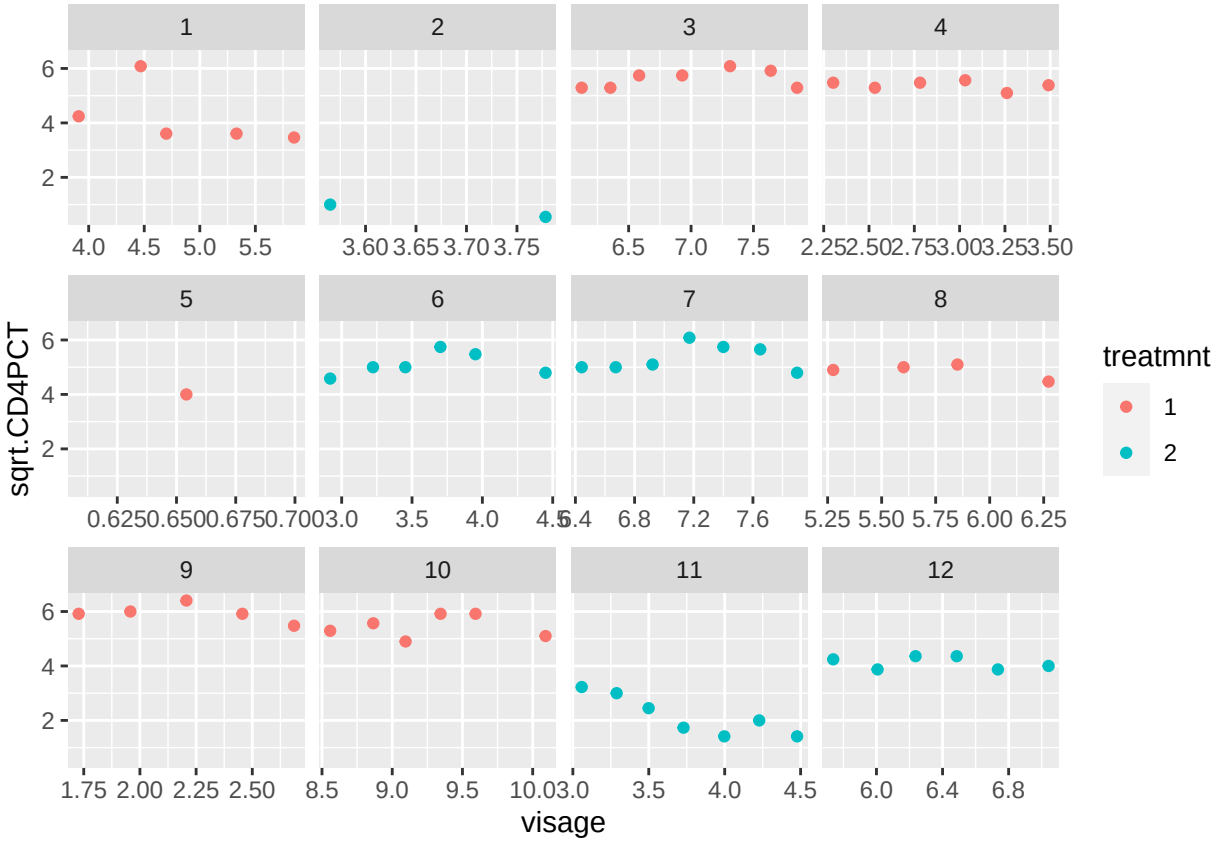
Here it is using VISIT...

```
g <- ggplot(cd4[first.12,],aes(x=VISIT,y=sqrt.CD4PCT)) +
  facet_wrap( ~ newpid) +
  geom_point(aes(color=treatmnt))

g
```


1
2
3
4
5
6
7
8
9
10
11
12
sqrt.CD4PCT
VISIT
treatmnt
1
2

...and here it is using `visage` *as the time variable...*

```
h <- ggplot(cd4[first.12,],aes(x=visage,y=sqrt.CD4PCT)) +
  facet_wrap( ~ newpid, scales="free_x") +
  geom_point(aes(color=treatmnt))

h
```



The graphs look very similar, but `visage`*, which is the actual age of the child, would seem to have more physical/medical meaning than* `VISIT`*, which is just the sequential visit number of each visit. If you have a reason for preferring* `VISIT`*, go for it.*

I will show the solutions below both ways (it will turn out not to make much difference!).

Problem 1(b).

Build a multilevel model with random slopes and intercepts for all the children, using a measure of time (VISIT or visage, whichever you answered in part (a)) as a Level 1 predictor, and treatmnt as a Level 2 predictor. Report the estimated fixed effects and variances from the model, and add the fitted regression lines to your plot from part (a).

Here's the model, using `VISIT` *as the measure of time:*

```
display(lmer.1 <- lmer(sqrt.CD4PCT ~ 1 + VISIT + treatmnt + (1+VISIT|newpid), data=cd4))
```

```
## lmer(formula = sqrt.CD4PCT ~ 1 + VISIT + treatmnt + (1 + VISIT |
##     newpid), data = cd4)
##             coef.est coef.se
## (Intercept)  4.69     0.13
```

```
## VISIT       -0.03     0.01
## treatmnt2    0.18     0.18
##
## Error terms:
##  Groups   Name        Std.Dev. Corr
##  newpid   (Intercept) 1.40
##           VISIT       0.05     -0.10
##  Residual             0.72
## ---
## number of obs: 1075, groups: newpid, 251
## AIC = 3140.7, DIC = 3100.6
## deviance = 3113.6
```

The overall intercept is $\hat{\beta}_0 = 4.69$*, the overall slope on* `VISIT` *is* $\beta_1 = -0.03$ *and the overall treatment effect is* $\beta_2 = 0.18$*. The residual variances are* $\hat{\sigma}^2 = (0.51)^2 = 0.26$*,* $\hat{\tau}_0^2 = (1.40)^2 = 1.96$ *and* $\hat{\tau}_1^2 = (0.05)^2 = 0.0025$*, and* $\hat{\rho}$ *is a very modest* -0.10*. The only things that seem surprising are*

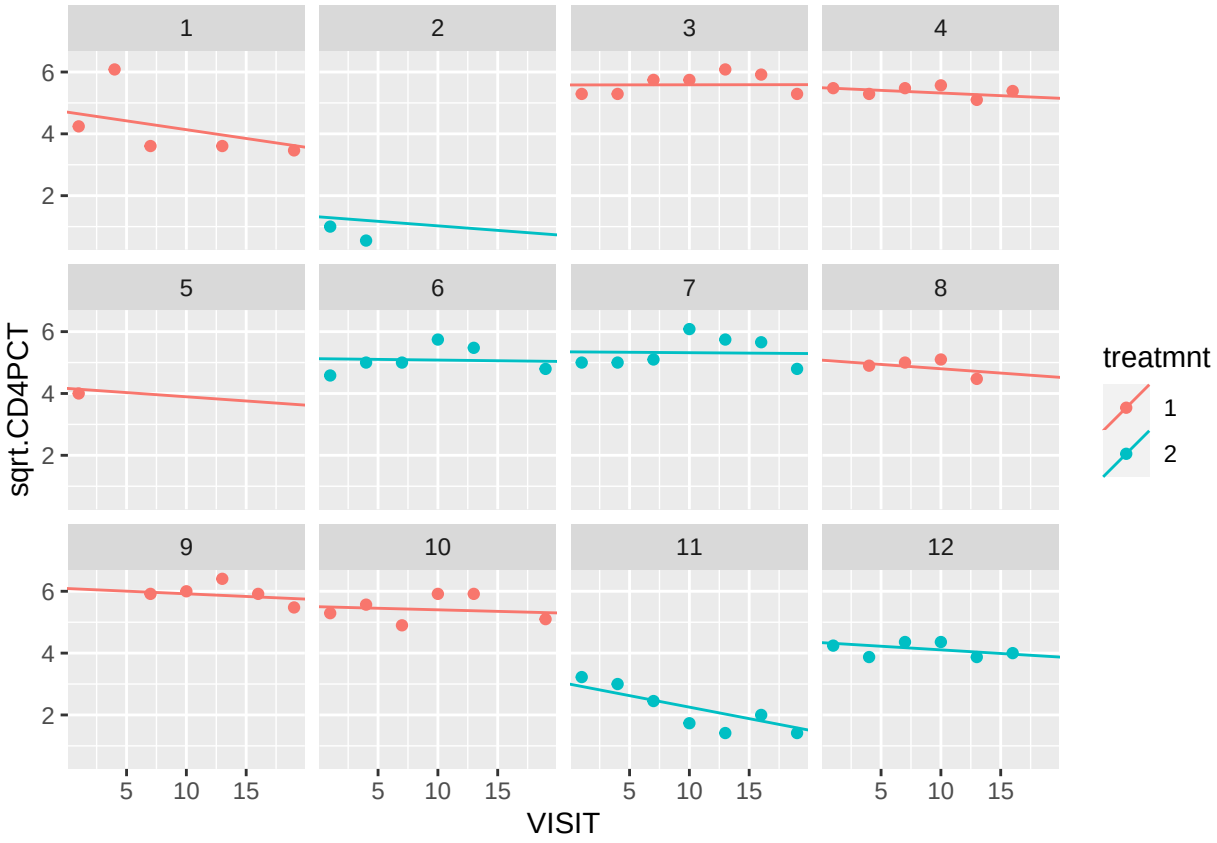
- $\hat{\tau}_0 = 1.40$ *seems large, considering that the most* `sqrt.CD4PCT` *values are between zero and 5;*
- *The treatmt effect,* $\hat{\beta}_2 = 0.18$*, is not significantly different from zero, even though the scatter plots of the first 12 children would suggest otherwise!*

Next we overlay regression lines on the scatterplots from part (a). The regression lines track the data well (and even have different intercepts, even though the the `treatmnt` *effect is nonsignificant in the model summary! [what part of the model could be helping here, do you suppose??])*

(It was a little trickier than I expected, to overlay the regression lines and have them colored differntly for each of the two `treatmnt` *levels, so have a look at the code below also.)*

Here are the overlaid regression lines, using `VISIT` *as the time variable:*

```
## fixef(lmer.1)  ## gives the betas for intercept and slopes on VISIT & treatmnt

## ranef(lmer.1)  ## gives the etas for intercepts and slope on VISIT

coef.1 <- coef(lmer.1)$newpid[,1:3] ## coef.1[,1] = beta0 + eta0  (random intcpt)
                                    ## coef.1[,2] = beta1 + eta1  (random slope)
                                    ## coef.1[,3] = beta2         (coef on treatmnt)

tx <- with(cd4,sapply(split(treatmnt,newpid), function(x) x[1]))  ## one per child

params <- data.frame(newpid=sort(unique(cd4$newpid)),
                     alpha0=coef.1[,1],
                     alpha1=coef.1[,2],
                     beta2=coef.1[,3],
                     tx
                     )

g + geom_abline(data=params[params$newpid<=12,],aes(intercept=alpha0 + ifelse(tx==2,beta2,0),
                                                    slope=alpha1,
                                                    color=tx))
```



Here's the model summary using `visage` *as the measure of time:*

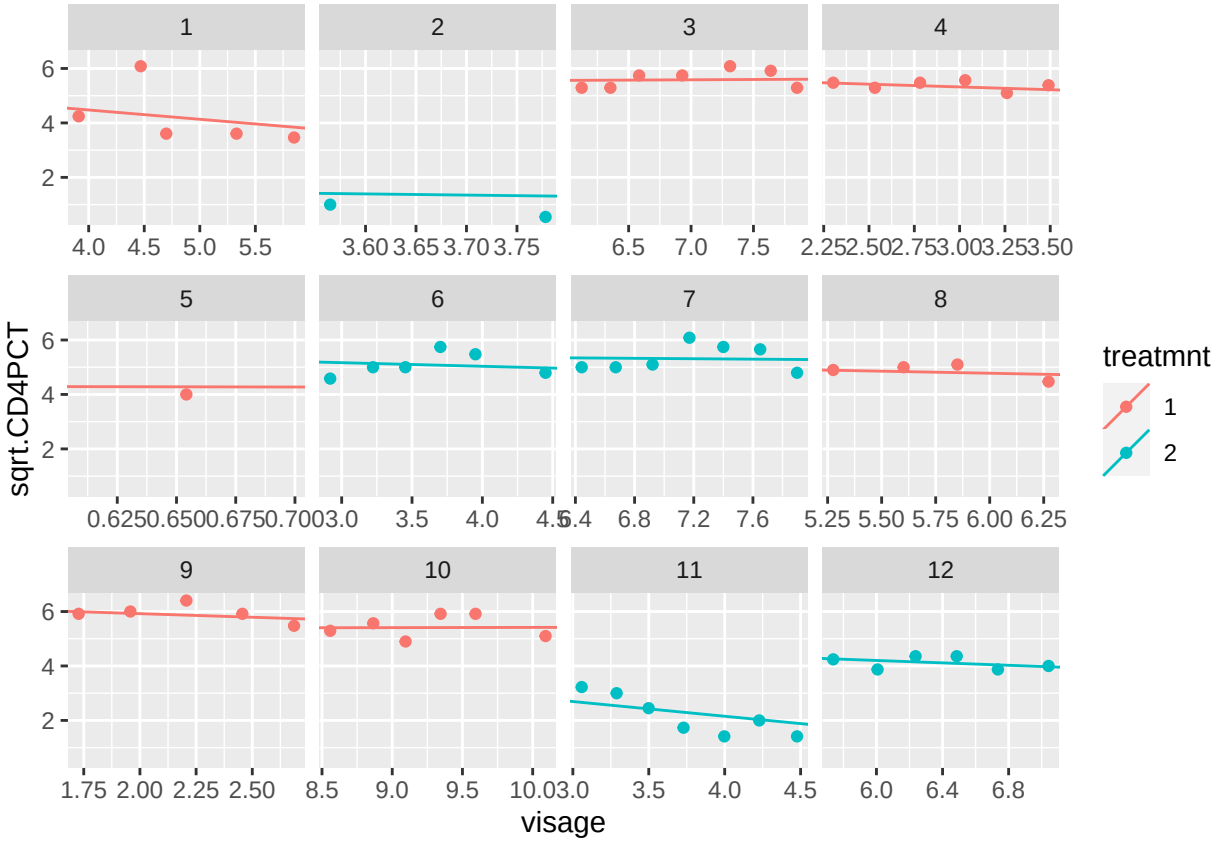
```
display(lmer.2 <- lmer(sqrt.CD4PCT ~ 1 + visage + treatmnt + (1+visage|newpid),data=cd4))
```

```
## lmer(formula = sqrt.CD4PCT ~ 1 + visage + treatmnt + (1 + visage |
##     newpid), data = cd4)
##             coef.est coef.se
## (Intercept)  5.34     0.18
## visage      -0.23     0.04
## treatmnt2    0.23     0.18
##
## Error terms:
##  Groups   Name        Std.Dev. Corr
##  newpid   (Intercept) 1.45
##           visage      0.29     -0.53
##  Residual             0.75
## ---
## number of obs: 1075, groups: newpid, 251
## AIC = 3152, DIC = 3119.5
## deviance = 3128.8
```

The story is much the same using `visage` *as a measure time, as using* `VISIT` *as a measure of time. Again* $\hat{\tau}_0$ *is rather large, and the estimated* `treatmnt` *effect* $\hat{\beta}_2$ *is nonsignificant.*

Here's the facets plot for the first 12 children, using `visage` *for time. Again the regression lines are tracking differences in the children rather well, even though the* `treatmnt` *effect is nonsignificant.*

```
coef.2 <- coef(lmer.2)$newpid[,1:3]  ## coef.2[,1] = beta0 + eta0  (random intcpt)
                                     ## coef.2[,2] = beta1 + eta1  (random slope)
                                     ## coef.2[,3] = beta2         (coef on treatmnt)

tx <- with(cd4,sapply(split(treatmnt,newpid), function(x) x[1]))  ## one per child

params <- data.frame(newpid=sort(unique(cd4$newpid)),
                     alpha0=coef.2[,1],
                     alpha1=coef.2[,2],
                     beta2=coef.2[,3],
                     tx
                     )

h + geom_abline(data=params[params$newpid<=12,],aes(intercept=alpha0 + ifelse(tx==2,beta2,0),
                                                    slope=alpha1,
                                                    color=tx))
```



Problem 1(c).

Try to expand the model by adding an interaction between treatmnt and your measure of time (Beacuse treatmnt is a level-2 variable and the time measure is a level-1 variable, this is somethings called a *cross-level interaction.*). Comment on changes in the estimated fixed effects and variances for this model, vs. the model in part (b). Add the fitted regression lines from this model to your previous facets plot. Does treatmnt account for an interesting amount of the variation in slopes (or intercepts) among children in the study?

OK, here we go with the crosslevel interaction model using `VISIT` *as the time variable:*

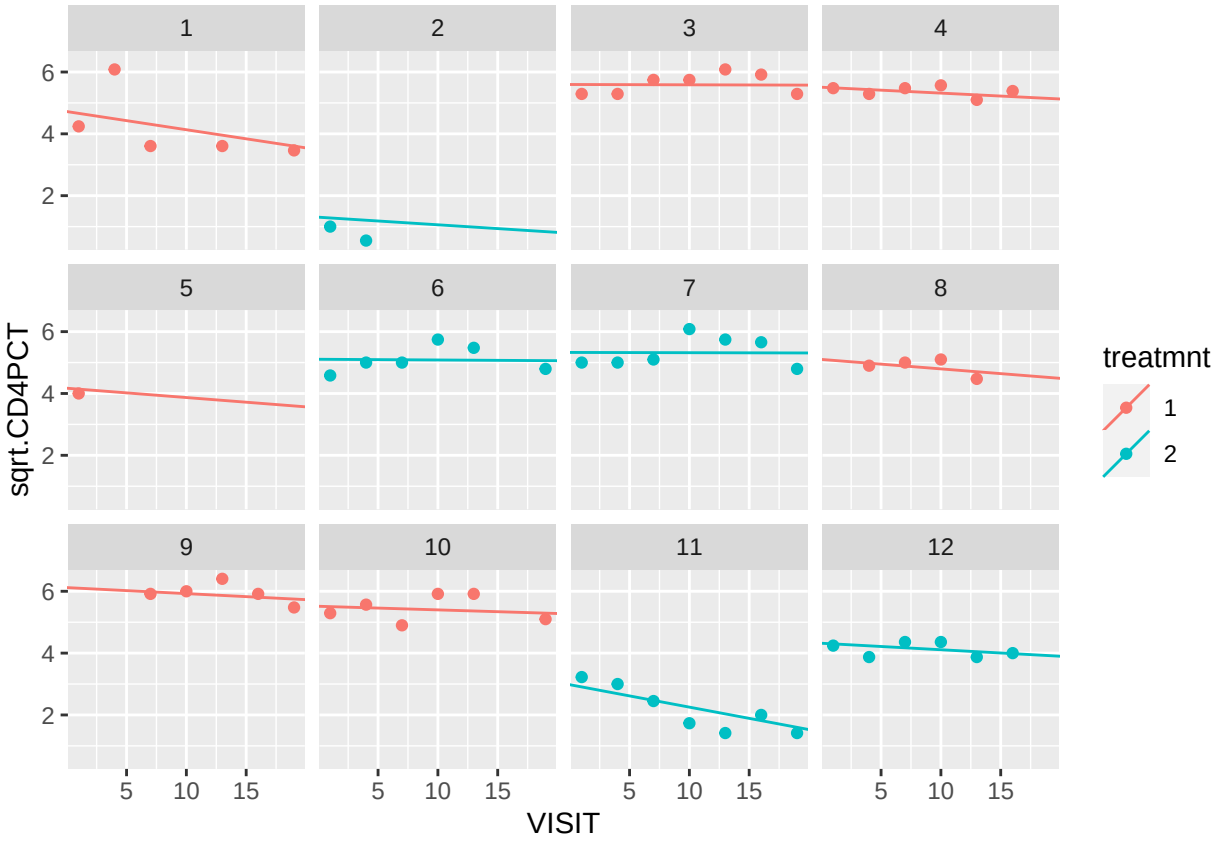
```
display(lmer.3 <- lmer(sqrt.CD4PCT ~ 1 + VISIT * treatmnt + (1+VISIT|newpid), data=cd4))
```

```
## lmer(formula = sqrt.CD4PCT ~ 1 + VISIT * treatmnt + (1 + VISIT |
##     newpid), data = cd4)
##                 coef.est coef.se
## (Intercept)      4.71     0.13
## VISIT           -0.03     0.01
## treatmnt2        0.14     0.19
## VISIT:treatmnt2  0.01     0.01
##
## Error terms:
##  Groups   Name        Std.Dev. Corr
##  newpid   (Intercept) 1.40
##           VISIT       0.05     -0.10
##  Residual             0.72
## ---
## number of obs: 1075, groups: newpid, 251
## AIC = 3149.4, DIC = 3093
## deviance = 3113.2
```

Same story as before: neither the coefficient on `treatmnt` *nor the coefficient on the interaction is significantly different from zero, but* $\hat{\tau}_0$ *is somewhat large for the scale of the data.*

Here are the regression lines plotted over the scatter plots using `VISIT` *for time; again you can see that they are tracking differences between children that we might have thought would be detected by the treatment effect.*

```
coef.3 <- coef(lmer.3)$newpid[,1:4]  ## coef.3[,1] = beta0 + eta0  (random intcpt)
                                     ## coef.3[,2] = beta1 + eta1  (random slope on time)
                                     ## coef.3[,3] = beta2         (coef on treatmnt)
                                     ## coef.3[,4] = beta3         (coef on interaction)

tx <- with(cd4,sapply(split(treatmnt,newpid), function(x) x[1]))  ## one per child

params <- data.frame(newpid=sort(unique(cd4$newpid)),
                     alpha0=coef.3[,1],
                     alpha1=coef.3[,2],
                     beta2=coef.3[,3],
                     beta3=coef.3[,4],
                     tx
                     )

g + geom_abline(data=params[params$newpid<=12,],aes(intercept=alpha0 + ifelse(tx==2,beta2,0),
                                                    slope=alpha1 + ifelse(tx==2,beta3,0),
                                                    color=tx))
```



Here's the cross level interaction model, using `visage` *as time*

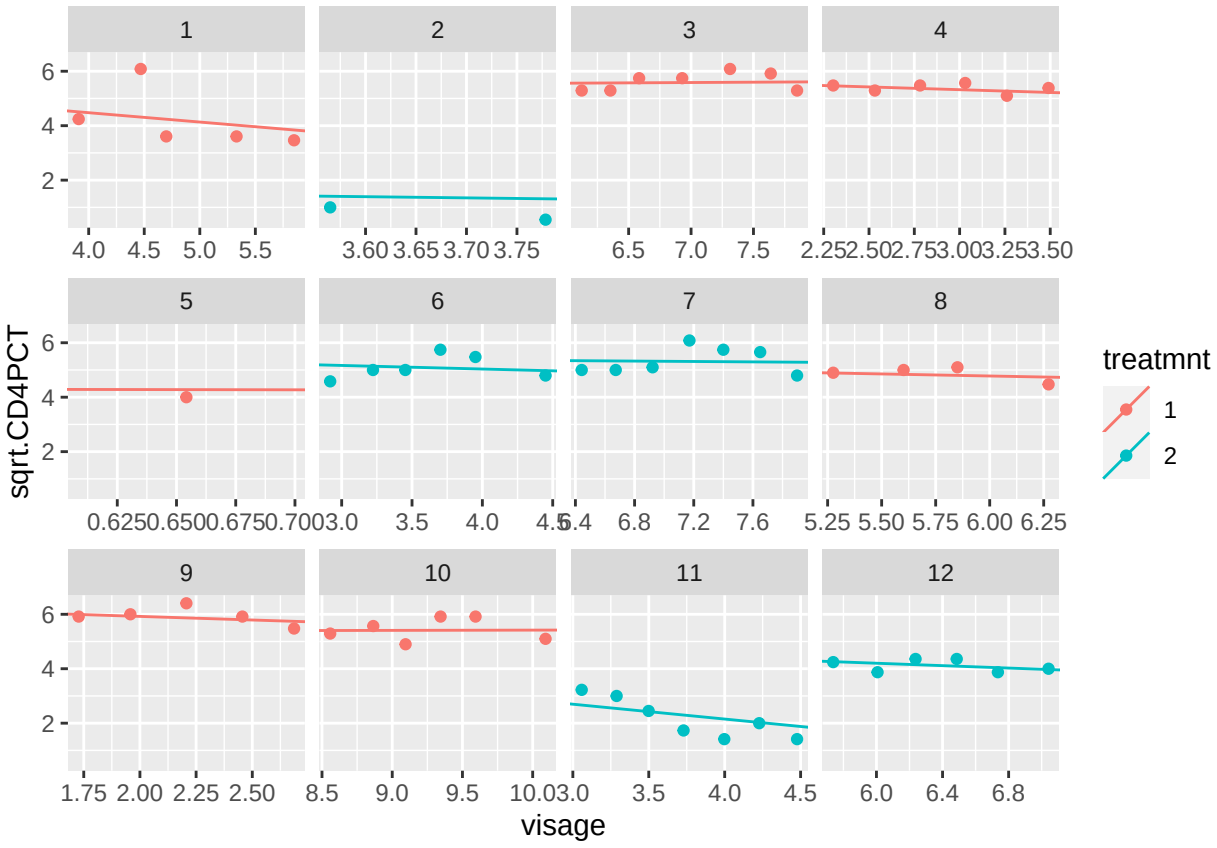
```
display(lmer.4 <- lmer(sqrt.CD4PCT ~ 1 + visage * treatmnt + (1+visage|newpid),data=cd4))
```

```
## lmer(formula = sqrt.CD4PCT ~ 1 + visage * treatmnt + (1 + visage |
##     newpid), data = cd4)
##                   coef.est coef.se
## (Intercept)        5.34     0.21
## visage            -0.23     0.05
## treatmnt2          0.24     0.32
## visage:treatmnt2   0.00     0.08
##
## Error terms:
##  Groups   Name        Std.Dev. Corr
##  newpid   (Intercept) 1.46
##           visage      0.30     -0.53
##  Residual             0.75
## ---
## number of obs: 1075, groups: newpid, 251
## AIC = 3157.3, DIC = 3116.3
## deviance = 3128.8
```

Again, no significant effects for `visage`*, and a rather large* $\hat{\tau}_0$.

Here's the facet plot...

```
coef.4 <- coef(lmer.4)$newpid[,1:4]  ## coef.4[,1] = beta0 + eta0  (random intcpt)
                                     ## coef.4[,2] = beta1 + eta1  (random slope on time)
                                     ## coef.4[,3] = beta2         (coef on treatmnt)
                                     ## coef.4[,4] = beta3         (coef on interaction)

tx <- with(cd4,sapply(split(treatmnt,newpid), function(x) x[1]))  ## one per child

params <- data.frame(newpid=sort(unique(cd4$newpid)),
                     alpha0=coef.4[,1],
                     alpha1=coef.4[,2],
                     beta2=coef.4[,3],
                     beta3=coef.4[,4],
                     tx
                     )

h + geom_abline(data=params[params$newpid<=12,],aes(intercept=alpha0 + ifelse(tx==2,beta2,0),
                                                    slope=alpha1 + ifelse(tx==2,beta3,0),
                                                    color=tx))
```



It does not appear that `treatmnt` *is having a significant effects on the intercepts or the slopes. We could explore a little further with information criteria:*

```
AIC.ml <- function(M) {AIC(update(M,REML=F))}
BIC.ml <- function(M) {BIC(update(M,REML=F))}
DIC.ml <- function(M) {extractDIC(update(M,REML=F))}  ## from library(arm)
```

```
lmer.0 <- lmer(sqrt.CD4PCT ~ 1 + VISIT + (1+VISIT|newpid),data=cd4)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00205385 (tol = 0.002, component 1)
```

```
res <- rbind(AIC=sapply(list(lmer.0,lmer.1,lmer.2,lmer.3,lmer.4), AIC.ml),
             BIC=sapply(list(lmer.0,lmer.1,lmer.2,lmer.3,lmer.4), BIC.ml),
             DIC=sapply(list(lmer.0,lmer.1,lmer.2,lmer.3,lmer.4), DIC.ml))
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00584617 (tol = 0.002, component 1)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00584617 (tol = 0.002, component 1)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00584617 (tol = 0.002, component 1)
```

```
colnames(res) <- c("lmer.0","lmer.1", "lmer.2", "lmer.3","lmer.4")

t(round(res,2))
```

```
##            AIC     BIC     DIC
## lmer.0 3126.56 3156.44 3114.56
## lmer.1 3127.63 3162.49 3113.63
## lmer.2 3142.76 3177.62 3128.76
## lmer.3 3129.21 3169.06 3113.21
## lmer.4 3144.75 3184.60 3128.75
```

If we restrict our attention to `lmer.1` *through* `lmer.4`*, the winner seems to be* `lmer.1`*, with model formula* `sqrt.CD4PCT ~ 1 + VISIT + treatmnt + (1+VISIT|newpid)`*. Comparing that with* `lmer.0` *which just omits* `treatmnt`*, the simpler* `lmer.0` *model seems to be favored; all of this is consistent with our earlier findings that treatmt doesn't seem to have a significant effect on either the slope on time, or the intercept, of the MLM's we have built.*

NOTE (not to be graded):

What I suspect is going on is that the random part of the intercept, η_{0j}, is accounting for variation between treated and untreated children, that would otherwise show up in the `treatmnt` *effect coefficient. We could check this out by removing the random part of the interecept from one of the models and comparing.*

Below I have fitted the model with fixed intercept and random slope (note the `0+visage` *term in the lmer call; the 0 removes the random intercept η_{0j}). As you can see, the coefficient on* `treatmnt` *is now (barely) significant, and the model still tracks the data well.*

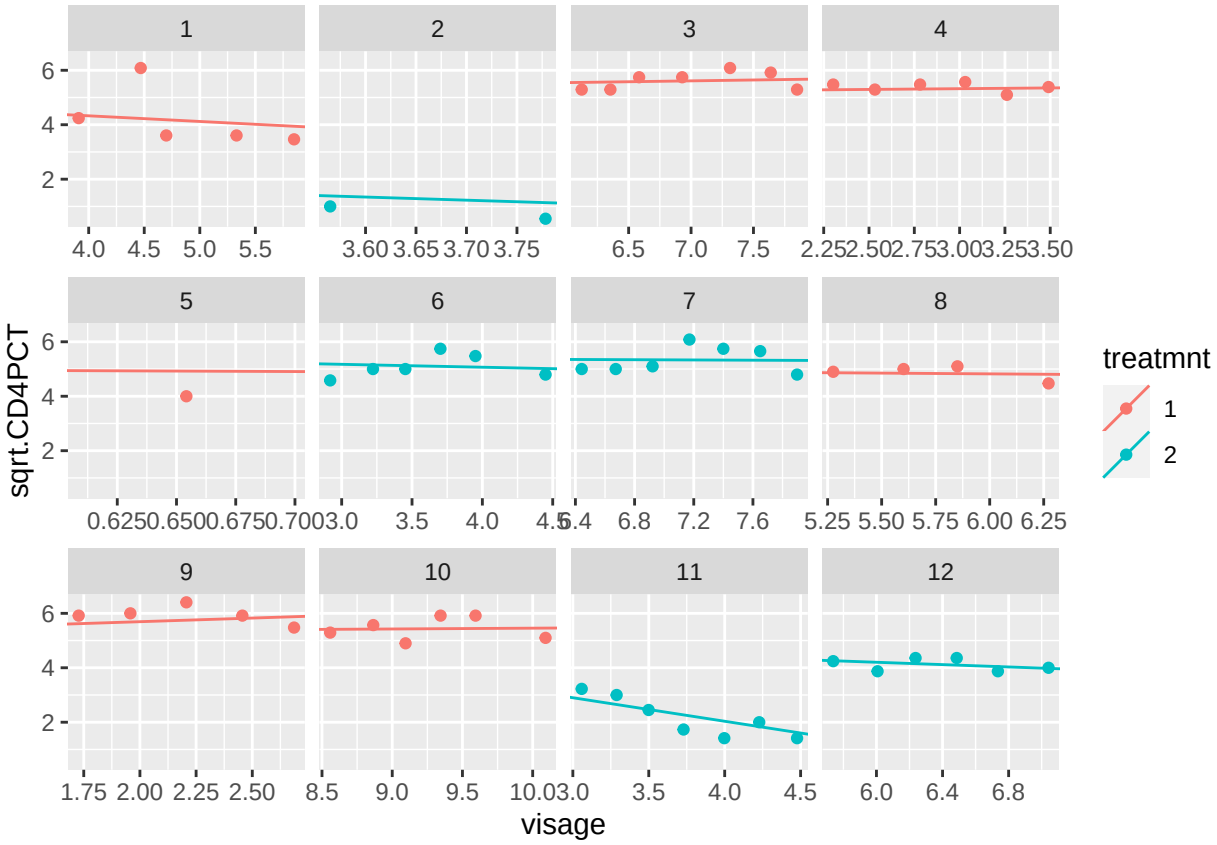
Comparing all these models is also a good exercise in seeing the difference between getting a model that does well with prediction, vs getting a model that might be the "true" model. All of the models we've looked at seem to do well with in-sample prediction (out-of-sample would also be interesting to look at), but they obviously can't all be the "correct/true" model describing the effect of treatment and time on `CD4PCT` *levels.*

```
display(lmer.5 <- lmer(sqrt.CD4PCT ~ 1 + visage + treatmnt + (0+visage|newpid),data=cd4))
```

```
## lmer(formula = sqrt.CD4PCT ~ 1 + visage + treatmnt + (0 + visage |
##     newpid), data = cd4)
##             coef.est coef.se
## (Intercept)  5.16     0.13
## visage      -0.20     0.04
```

```
## treatmnt2    0.33     0.14
##
## Error terms:
##  Groups   Name   Std.Dev.
##  newpid   visage 0.42
##  Residual        0.82
## ---
## number of obs: 1075, groups: newpid, 251
## AIC = 3297.1, DIC = 3267.3
## deviance = 3277.2
```

```
coef.5 <- coef(lmer.5)$newpid[,1:3]  ## coef.5[,1] = beta0          (fixed intcpt)
                                     ## coef.5[,2] = beta1 + eta1  (random slope on time)
                                     ## coef.5[,3] = beta2         (coef on treatmnt)

tx <- with(cd4,sapply(split(treatmnt,newpid), function(x) x[1]))  ## one per child

params <- data.frame(newpid=sort(unique(cd4$newpid)),
                     beta0=coef.5[,1],
                     alpha1=coef.5[,2],
                     beta2=coef.5[,3],
                     tx
                     )

h + geom_abline(data=params[params$newpid<=12,],aes(intercept=beta0 + ifelse(tx==2,beta2,0),
                                                    slope=alpha1,
                                                    color=tx))
```



Problem 2.

Now we will consider in more detail the model below for the CD4 data set:

```
lmer.1 <- lmer(sqrt.CD4PCT ~ 1 + VISIT * treatmnt + (1 + VISIT | newpid),
               data=cd4)
```

Bring together all the residuals (except random effects residuals) that we have talked about, using library(HLMdiag):

```
cd4 <- read.csv("allvar.csv",header=T)

## It's worth doing a little exploration of the data here,
## especially to look at missing values.

apply(cd4,2,function(x) mean(is.na(x)))
```

```
##       VISIT      newpid       VDATE      CD4PCT         arv      visage
## 0.000000000 0.000000000 0.000000000 0.142743222 0.103668262 0.109250399
##    treatmnt      CD4CNT     baseage
## 0.000000000 0.146730463 0.007177033
```

```
## we only need the variables newpid, VISIT, visage, CD4PCT and treatmnt
## so let's get rid of some (but not all) missing data problems
## by deleting the other variables

cd4 <- with(cd4,data.frame(newpid=newpid,
                           VISIT=VISIT,
                           visage=visage,
                           CD4PCT=CD4PCT,
                           treatmnt=treatmnt))

## and since we still have some missing data, we will just
## delete the rows that continue to have NA's (not a great
## practice in general, but good enough for this exercise...)
cd4 <- cd4[!apply(cd4,1,function(x) any(is.na(x))),]

## set up the sqrt of the response variable...
cd4$sqrt.CD4PCT <- sqrt(cd4$CD4PCT)

## and for coloring ggplot elements it will better if treatmnt is a factor...
cd4$treatmnt <- as.factor(cd4$treatmnt)

## select the first 12 kids...
first.12 <- (cd4$newpid <= 12)

## Fit the base model for this exercise...
display(lmer.1 <- lmer(sqrt.CD4PCT ~ 1 + VISIT * treatmnt + (1 + VISIT | newpid),
               data=cd4))
```

```
## lmer(formula = sqrt.CD4PCT ~ 1 + VISIT * treatmnt + (1 + VISIT |
##     newpid), data = cd4)
##                 coef.est coef.se
## (Intercept)      4.71     0.13
## VISIT           -0.03     0.01
## treatmnt2        0.14     0.19
## VISIT:treatmnt2  0.01     0.01
```

```
##
## Error terms:
##  Groups   Name        Std.Dev. Corr
##  newpid   (Intercept) 1.40
##           VISIT       0.05     -0.10
##  Residual             0.72
## ---
## number of obs: 1075, groups: newpid, 251
## AIC = 3149.4, DIC = 3093
## deviance = 3113.2
```

```
r.1 <- hlm_resid(lmer.1,level=1,include.ls=F)
r.1s <- hlm_resid(lmer.1,level=1,include.ls=F,standardize=T)
r.2 <- hlm_resid(lmer.1,level="newpid",include.ls=F)
r.2s <- hlm_resid(lmer.1,level="newpid",include.ls=F,standardize=T)
names(r.1)
```

```
## [1] "id"          "sqrt.CD4PCT" "VISIT"       "treatmnt"    "newpid"
## [6] ".resid"      ".fitted"     ".mar.resid"  ".mar.fitted"
```

```
names(r.1s)
```

```
## [1] "id"              "sqrt.CD4PCT"     "VISIT"           "treatmnt"
## [5] "newpid"          ".std.resid"      ".fitted"         ".chol.mar.resid"
## [9] ".mar.fitted"
```

```
names(r.2)
```

```
## [1] "newpid"           ".ranef.intercept" ".ranef.visit"
```

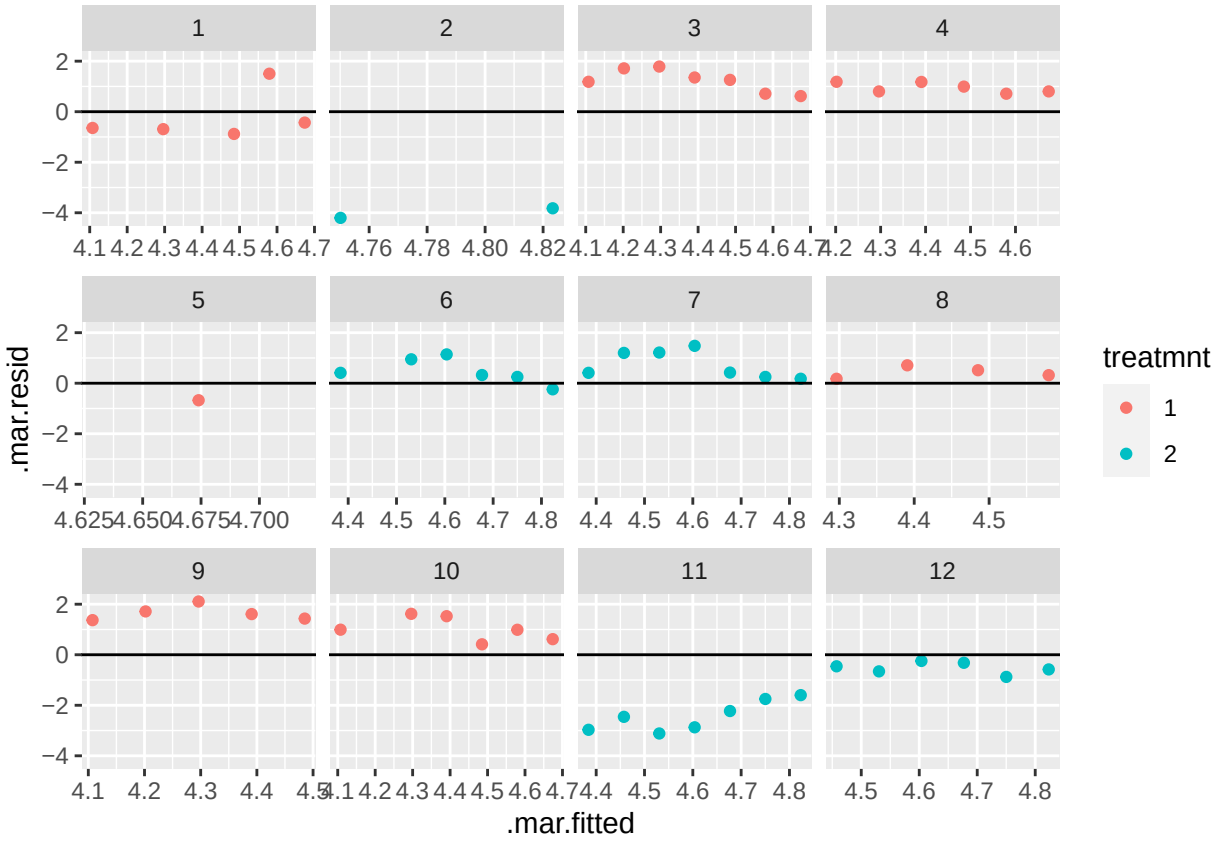
```
names(r.2s)
```

```
## [1] "newpid"               ".std.ranef.intercept" ".std.ranef.visit"
```

Problem 2(a).

Make a facets plot of the marginal residuals, as a function of the marginal fitted values (use `scales="free_x"` if needed to make the plot legible). Explain in a sentence or two why a facets plot is not very useful for assessing model fit for this problem, whether we look at the first 12 children, or all 251 children).

```
ggplot(r.1[first.12,],aes(x=.mar.fitted,y=.mar.resid)) +
  facet_wrap( ~ newpid,scales="free_x") +
  geom_point(aes(color=treatmnt)) +
  geom_abline(intercept=0,slope=0)
```



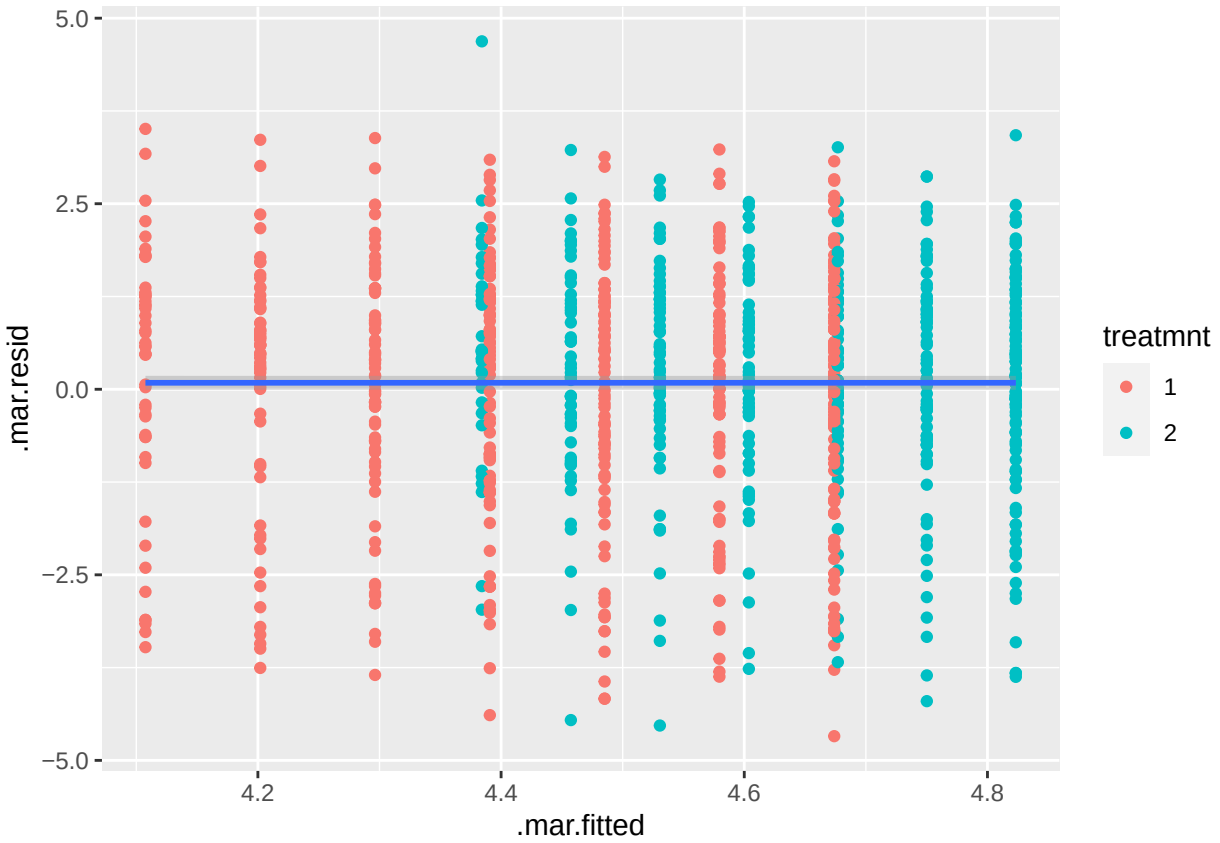
There are very few data points per child, and so the residual plot within facet for each child contains almost no information about fit. (Having said that, there are some hints of misfit, to the extent that several individual children's residuals are entirely above, or entirely below, the zero line.)

Problem 2(b).

Make an ungrouped (that is, no facets) scatter plot of marginal residuals as a function of marginal fitted values, using the full data set (not just the first 12 children). Color the points for `treatmnt=1` kids and treatmnt=2 kids with different colors. Overlay a smooth fit (`geom_smooth` is the easist to use here). Explain, in a couple of sentences (optionally with some math):

- What is causing the dominant structure in this plot, and why that dominant structure is essentially irrelevant for checking the relationship between `sqrt.CD4PCT` and `VISIT`; and
- What in this plot makes you happy or unhappy about having a linear relationship between `sqrt.CD4PCT` and `VISIT` in the model.

```
ggplot(r.1,aes(x=.mar.fitted,y=.mar.resid)) +
  geom_point(aes(color=treatmnt)) +
  geom_smooth()
```



Explain, in a couple of sentences (optionally with some math):

- What is causing the dominant structure in this plot, and why that dominant structure is essentially irrelevant for checking the relationship between `sqrt.CD4PCT` and `VISIT`;

  The dominant structure in the data is the way the residuals are grouped horizontally into 14 groups—7 groups for `treatmnt=1` *and 7 for* `treatmnt=2`*. They are caused by the fact that our main predictor variable is* `VISIT`*, which only takes on 7 values. The spacing of the groups is not the same between* `treatmnt=1` *and* `treatmnt=2` *because the model contains a* `treatmnt` $\times$ `VISIT` *interaction, so that the slope on* `VISIT` *is different for children in the* `treatmnt=1` *and* `treatmnt=2` *groups.*

  This does not affect our assessment of the fit of the model, since to check the model fit we are interested in vertical patterns (trends, curves, outliers, changing variance, etc.) in the residuals, not horizontal (grouping) patterns.

  and

- What in this plot makes you happy or unhappy about having a linear relationship between `sqrt.CD4PCT` and `VISIT` in the model.

  The plot actually looks pretty wonderful. The variance of the residuals looks pretty much constant, as a function of the fitted values, and the trend modeled with the smooth really does look like a horizontal line at 0.

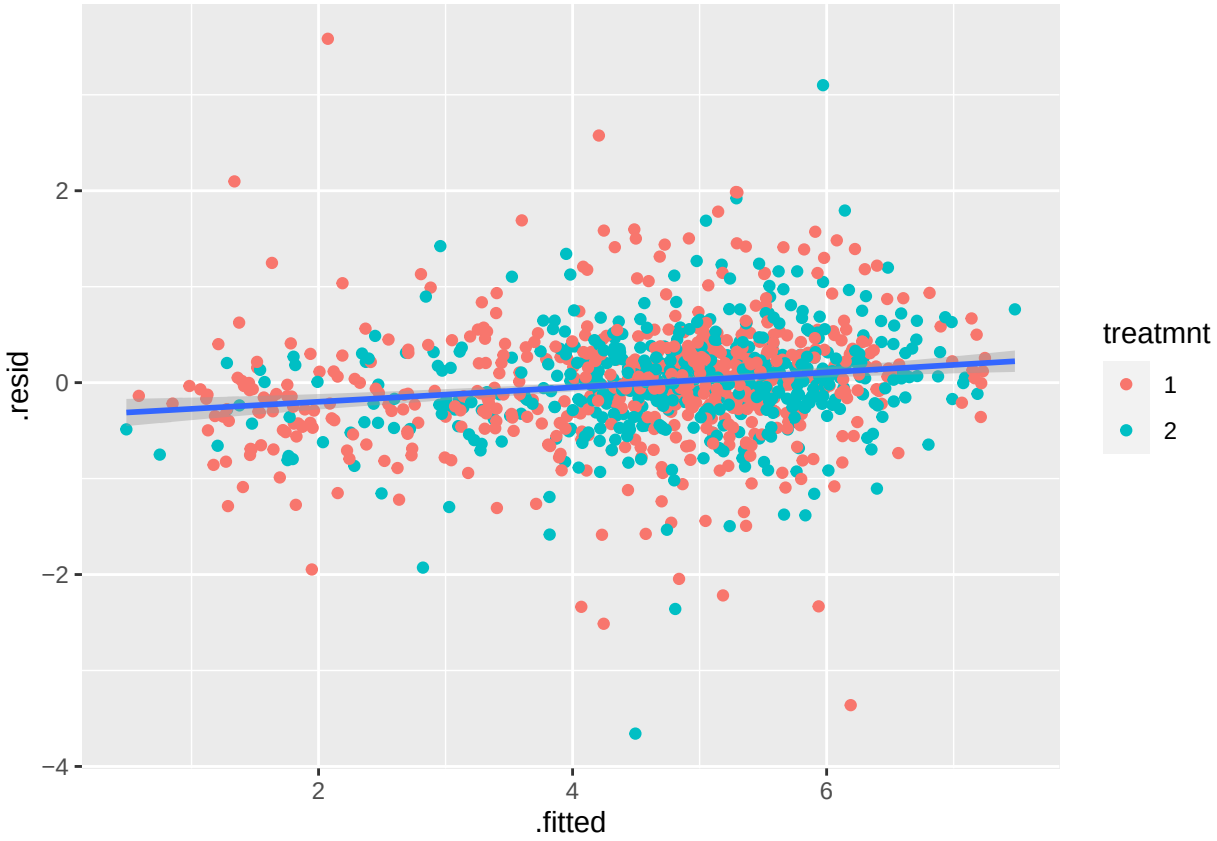
Problem 2(c).

Make an ungrouped (that is, no facets) scatter plot of conditional residuals as a function of conditional fitted values, using the full data set (not just the first 12 children). Color the points for `treatmnt=1` kids and

`treatmnt=2` kids with different colors. Overlay a smooth fit (`geom_smooth` is the easist to use here). Explain, in a couple of sentences (optionally with some math):

- Why the dominant structure in the marginal residuals is not also present in this plot of conditional residuals
- What might be causing the trend you see in this plot to be different from the trend in the plot of the marginal residuals.

```
ggplot(r.1,aes(x=.fitted,y=.resid)) +
  geom_point(aes(color=treatmnt)) +
  geom_smooth()
```



Explain, in a couple of sentences (optionally with some math):

- Why the dominant structure in the marginal residuals is not also present in this plot of conditional residuals

  The marginal residuals plot shows

$$\begin{aligned} \text{.mar.resid}_i &= \text{sqrt.CD4PCT}_i - \hat{\beta}_0 - \hat{\beta}_1 \cdot \text{VISIT}_i - \hat{\beta}_3 \cdot \text{treatmnt}_{j[i]} - \hat{\beta}_4 \cdot \text{VISIT}_i \cdot \text{treatmnt}_{j[i]} \\ &\text{vs.} \\ \text{.mar.fitted}_i &= \hat{\beta}_0 + \hat{\beta}_1 \cdot \text{VISIT}_i + \hat{\beta}_3 \cdot \text{treatmnt}_{j[i]} + \hat{\beta}_4 \cdot \text{VISIT}_i \cdot \text{treatmnt}_{j[i]} \end{aligned}$$

  and because ***`VISIT`*** *and* ***`treatmnt`*** *are both discrete, the fitted values will only take on a few discrete values, causing the horizontal grouping in the marginal plot.*

The conditional residuals plot shows

$$\begin{aligned} .\text{resid}_i &= \text{sqrt.CD4PCT}_i - \hat{\beta}_0 - \hat{\beta}_1 \cdot \text{VISIT}_i - \hat{\beta}_3 \cdot \text{treatmnt}_{j[i]} - \hat{\beta}_4 \cdot \text{VISIT}_i \cdot \text{treatmnt}_{j[i]} \\ &\quad -\eta_{0j[i]} - \eta_{1j[i]} \cdot \text{VISIT}_i \\ &\text{vs.} \\ .\text{fitted}_i &= \hat{\beta}_0 + \hat{\beta}_1 \cdot \text{VISIT}_i + \hat{\beta}_3 \cdot \text{treatmnt}_{j[i]} + \hat{\beta}_4 \cdot \text{VISIT}_i \cdot \text{treatmnt}_{j[i]} \\ &\quad +\eta_{0j[i]} + \eta_{1j[i]} \cdot \text{VISIT}_i \end{aligned}$$

The additional continuous (because the η's are continuous) terms $\eta_{0j[i]} + \eta_{1j[i]} \cdot VISIT_i$ essentially "jitter" the fitted values so they no longer show the horizonal grouping structure of the marginal residuals.

- What might be causing the trend you see in this plot to be different from the trend in the plot of the marginal residuals.

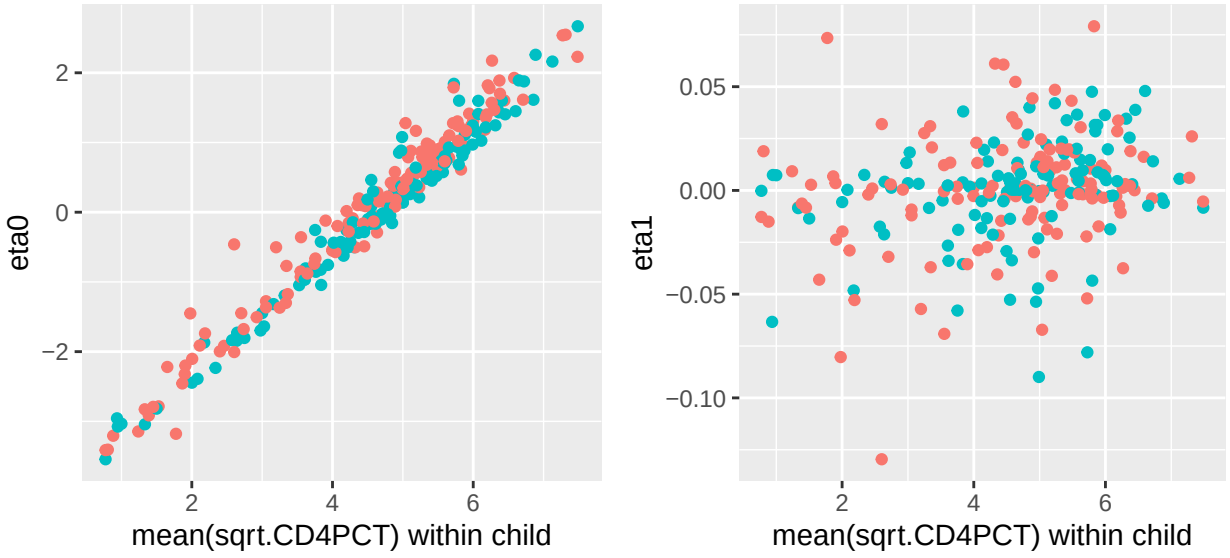
  The trend shows that when the conditional fitted value $\hat{y}_{cond}$ is larger, it tends to underpredict y = sqrt.CD4PCT (positive residual), and when $\hat{y}_{cond}$ is smaller, it tends to overpredict y (negative residual).

  The only difference between the two plots is the presence of the $\eta_{0j[i]} + \eta_{1j[i]} \cdot VISIT_i$ terms.

  In ordinary regression,we know that residuals and fitted values will be uncorrelated: that is why we don't see overall increasing or overall decreasing trends in the marginal residual plots (as far as the math is concerned, marginal residuals and marginal fitted values behave exactly like ordinary regression). The fact that conditional residuals and conditional fitted values are correlated is due to the values we use for the η's: $\hat{\eta} = E[\eta|\text{the data}]$, and "the data" includes y. Since, therefore, the conditional fitted values depend on y, it's not surprising that the residuals and fitted values are correlated.

  The plots below explore the association between the $\hat{\eta}$'s and y. We can see that, although there's not much association between $\hat{\eta}_{1j}$ and y, there is quite a strong association between $\hat{\eta}_{0j}$ and y.

```
y.grouped <- with(cd4,sapply(split(sqrt.CD4PCT,newpid),mean))
tx.grouped <- with(cd4,sapply(split(treatmnt,newpid),function(x) x[1]))

g1 <- ggplot(r.2,aes(x=y.grouped,y=.ranef.intercept)) +
  geom_point(aes(color=tx.grouped)) +
  xlab("mean(sqrt.CD4PCT) within child") +
  ylab("eta0") +
  theme(legend.position="none")

g2 <- ggplot(r.2,aes(x=y.grouped,y=.ranef.visit)) +
  geom_point(aes(color=tx.grouped)) +
  xlab("mean(sqrt.CD4PCT) within child") +
  ylab("eta1") +
  theme(legend.position="none")

grid.arrange(g1,g2,ncol=2)
```



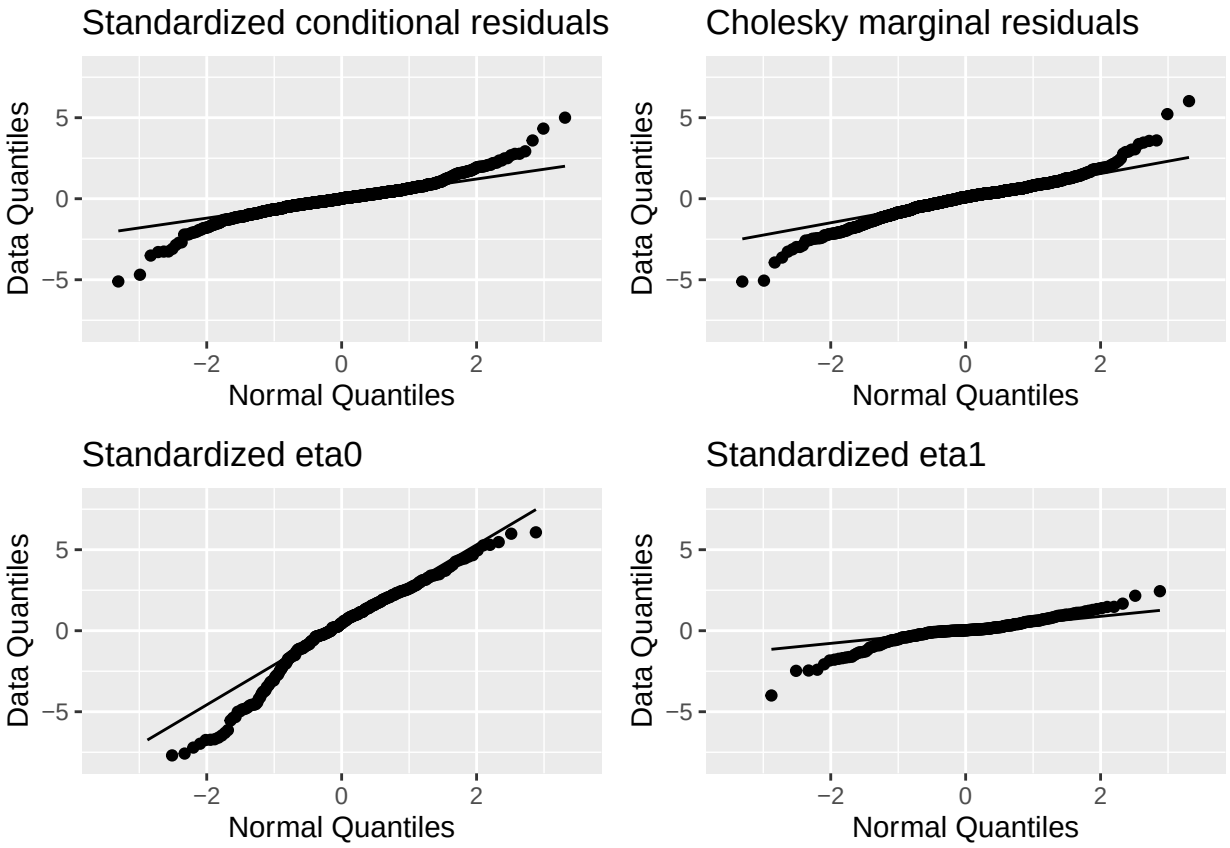
Problem 2(d).

Use standardized residuals and standardized random effects estimates to assess the normality of ϵ_i, η_{0j} and η_{1j} in the fitted model, and to check for any outliers. Include qq plots for each, and accompany each plot with a sentence or two describing what is good or bad in that plot.

```
g1 <- ggplot(r.1s,aes(sample=.std.resid)) +
  geom_qq() +
  geom_qq_line() +
  xlim(-3.5,3.5) +
  ylim(-8,8) +
  xlab("Normal Quantiles") +
  ylab("Data Quantiles") +
  ggtitle("Standardized conditional residuals")

g2 <- ggplot(r.1s,aes(sample=.chol.mar.resid)) +
  geom_qq() +
  geom_qq_line() +
  xlim(-3.5,3.5) +
  ylim(-8,8) +
  xlab("Normal Quantiles") +
  ylab("Data Quantiles") +
  ggtitle("Cholesky marginal residuals")

g3 <- ggplot(r.2s,aes(sample=.std.ranef.intercept)) +
  geom_qq() +
  geom_qq_line() +
  xlim(-3.5,3.5) +
  ylim(-8,8) +
  xlab("Normal Quantiles") +
  ylab("Data Quantiles") +
  ggtitle("Standardized eta0")

g4 <- ggplot(r.2s,aes(sample=.std.ranef.visit)) +
  geom_qq() +
```

```
  geom_qq_line() +
  xlim(-3.5,3.5) +
  ylim(-8,8) +
  xlab("Normal Quantiles") +
  ylab("Data Quantiles") +
  ggtitle("Standardized eta1")

grid.arrange(g1,g2,g3,g4,ncol=2)
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



The "middle" part of all four plots seem to follow the normal distribution fairly well, but the tails fail in various ways:

- *The standardized conditional and marginal residuals both have longer tails than the Normal distribution. There seem to be two low outliers and two high outliers among these residuals.*
- *The Cholesky marginal residual seem a little more extreme that the standardized conditional residuals, which make sense since $X\beta + Z\eta$ should do a better job predicting y than $X\beta$ alone. There seem to be two low outliers and two high outliers among these residuals. It would be useful to know if these are the same data points as in the first QQ plot.*
- *The standardized $\hat{\eta}_0$'s have a long left tail and slighltly short right tail. In fact the left tail for η_0 is the most extreme of all four plots. There don't appear to be any clear outliers among the $\hat{\eta}_0$'s.*
- *The distribution of $\hat{\eta}_1$ is closest to Normal, with a right tail that is only a little longer than the Normal, and a left tail that is the least extreme of all four plots. There may be one low outlier among the $\hat{\eta}_1$'s.*

Problem 3.

Continuing with the CD4 data...

> **Note:** *Because I am not covering cAIC very much in class I have asked you not to compare models with cAIC (that is, no credit lost if you do not consider cAIC). However, since it is on the assignment, I wanted to (a) make some notes on it here; and (b) show what the results using cAIC would be in the solutions below.*
>
> *Here are notes on cAIC:*
>
> - *Earlier in the semester we worked with a quantity called "CAIC". Since this quantity is called "cAIC" you might think these are the same. They are actually not at all the same!*
>   - *CAIC is only appropriate for regular lm() models. It is a version of AIC that has been adjusted for small sample size:*
>
> $$\begin{aligned} AIC &= -2logLik(M) + 2(k+1) \quad \text{, where } k = df = p+1. \\ CAIC &= -2logLik(M) + 2\frac{n+2}{n-k}(k+1) \end{aligned}$$
>
>   - *cAIC is designed for lmer() models. Like DIC, it is a version of AIC that has been adjusted for reduced degrees of freedom due to the presence of random effects. The mathematics is a bit much for the amount of time/space that we have in this class, but here's the overview from 10,000 feet:*
>
> **DIC:** $DIC = -2logLik(M) + 2k_{eff}$
>
> * $LogLik(M)$ *based on marginal model* $f(\underline{Y}|\underline{\beta}, \underline{\omega}, \sigma^2)$
> * k_{eff} *is estimated from the curvature of the likelihood* $f(\underline{Y}|\underline{\beta}, \underline{\omega}, \sigma^2)$*, which is driven by the size of the* τ^2*'s*
>
> **cAIC:** $cAIC = -2logLik(M) + 2k_{eff}$
>
> * $logLik(M)$ *based on the conditional model* $f(\underline{Y}|\underline{\eta}, \underline{\beta}, \underline{\omega}, \sigma^2)$
> * k_{eff} *is a different "model curvature" estimate*
>
> - *You can use the cAIC() function from library(cAIC4) to calculate cAIC.*
> - *Since cAIC and DIC are both adjustments to AIC to account for random* η*'s, when there are no* η*'s in the problem, both cAIC and DIC reduce to just AIC.*
>
> *In the solutions below I have just integrated work with cAIC with all the other work on this problem.*

Problem 3(a).

Make a table giving values of AIC, BIC, DIC, and cAIC:

```
sqrt.CD4PCT ~ 1 + visage +            (1+visage|newpid)
sqrt.CD4PCT ~ 1 + visage + treatmnt + (1+visage|newpid)
sqrt.CD4PCT ~ 1 + visage * treatmnt + (1+visage|newpid)
sqrt.CD4PCT ~ 1 + VISIT  +            (1+VISIT|newpid)
sqrt.CD4PCT ~ 1 + VISIT  + treatmnt + (1+VISIT|newpid)
sqrt.CD4PCT ~ 1 + VISIT  * treatmnt + (1+VISIT|newpid)
```

Comment briefly on any similarities or differences in how the different criteria choose fixed effects.

```
lmer.1 <- lmer(sqrt.CD4PCT ~ 1 + visage +           (1+visage|newpid),data=cd4,REML=F)
lmer.2 <- lmer(sqrt.CD4PCT ~ 1 + visage + treatmnt + (1+visage|newpid),data=cd4,REML=F)
lmer.3 <- lmer(sqrt.CD4PCT ~ 1 + visage * treatmnt + (1+visage|newpid),data=cd4,REML=F)
lmer.4 <- lmer(sqrt.CD4PCT ~ 1 + VISIT  +           (1+VISIT|newpid),data=cd4,REML=F)
lmer.5 <- lmer(sqrt.CD4PCT ~ 1 + VISIT  + treatmnt + (1+VISIT|newpid),data=cd4,REML=F)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00584617 (tol = 0.002, component 1)
```

```
lmer.6 <- lmer(sqrt.CD4PCT ~ 1 + VISIT  * treatmnt + (1+VISIT|newpid),data=cd4,REML=F)

DIC <- function(M) {
         if(class(M)=="lm") { x <- AIC(M) }
         else { x <- unname(extractDIC(M)) }
         return(x)
       }

CAIC <- function(M) {cAIC(M)$caic}

IC <- function(M) {
    x <-c(AIC=AIC(M),BIC=BIC(M),DIC=DIC(M),cAIC=CAIC(M))
    return(x)
}

IC.table <- function(...) {
    Mlist <- list(...)
    x <- suppressWarnings(lapply(Mlist,IC))
    x <- data.frame(matrix(unlist(x),ncol=4,byrow=T))
    names(x) <- c("AIC","BIC","DIC","cAIC")
    models <- sapply(Mlist,formula)
    models <- abbreviate(substr(models,14,200),30)
    rownames(x) <- models
    return(x)
}

IC.table(lmer.1,lmer.2,lmer.3,lmer.4,lmer.5,lmer.6)
```

```
##                                      AIC      BIC      DIC     cAIC
## 1+visage+(1+visage|newpid)     3142.447 3172.327 3130.447 2697.868
## 1+visag+treatmnt+(1+visg|nwpd) 3142.755 3177.616 3128.755 2698.827
## 1+visag*treatmnt+(1+visg|nwpd) 3144.755 3184.595 3128.755 2699.562
## 1+VISIT+(1+VISIT|newpid)       3126.563 3156.443 3114.563 2642.798
## 1+VISIT+tretmnt+(1+VISIT|nwpd) 3127.625 3162.486 3113.625 2643.307
## 1+VISIT*tretmnt+(1+VISIT|nwpd) 3129.215 3169.056 3113.215 2644.733
```

Comment briefly on any similarities or differences in how the different criteria choose fixed effects.

Here's an overall sum of the best (minimizing the criterion) models for each criterion, with second and third place winners as well:

AIC: *Best model is lmer.4, with lmer.5 a close second.*

BIC: *Best model is lmer.4, with lmer.5 a substantially less close second.*

DIC: *Best model is lmer.6, with lmer.4 and lmer.5 almost indistinguishably close.*

cAIC: *Best model is lmer.4, with lmer.5 and lmer.6 almost indistinguishably close.*

All the criteria prefer having `VISIT` *as a predictor rather than* `visage`*. (This kind of surprises me because visage has more physical meaning, but so be it. . .) Consistent with our findings on earlier hw that treatmnt did not seem like an important predictor,* `lmer.4` *(without* `treatmnt` *in the model) is first or second with all the criteria, with* `lmer.5` *(main effect for* `treatmt`*, but no interaction) a strong second or third with all the criteria.*

Problem 3(b).

Now let's look at the random effects in the model

```
sqrt.CD4PCT ~ 1 + VISIT + treatmnt + (1+VISIT|newpid)
```

Again, make a table giving values of AIC, BIC, DIC, and cAIC for the following models:

```
sqrt.CD4PCT ~ 1 + VISIT + treatmnt
sqrt.CD4PCT ~ 1 + VISIT + treatmnt + (1|newpid)
sqrt.CD4PCT ~ 1 + VISIT + treatmnt + (0+VISIT|newpid)
sqrt.CD4PCT ~ 1 + VISIT + treatmnt + (1+VISIT|newpid)
```

```
lm.7 <- lm(sqrt.CD4PCT ~ 1 + VISIT + treatmnt,data=cd4)
lmer.8 <- lmer(sqrt.CD4PCT ~ 1 + VISIT + treatmnt + (1|newpid),data=cd4,REML=F)
lmer.9 <- lmer(sqrt.CD4PCT ~ 1 + VISIT + treatmnt + (0+VISIT|newpid),data=cd4,REML=F)
lmer.10 <- lmer(sqrt.CD4PCT ~ 1 + VISIT + treatmnt + (1+VISIT|newpid),data=cd4,REML=F)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00584617 (tol = 0.002, component 1)
```

```
IC.table(lm.7,lmer.8,lmer.9,lmer.10)
```

```
##                                     AIC      BIC      DIC     cAIC
## 1 + VISIT + treatmnt           4009.810 4029.730 4009.810 4009.810
## 1+VISIT+treatmnt+(1|newpid)    3153.879 3178.779 3143.879 2728.147
## 1+VISIT+tretmnt+(0+VISIT|nwpd) 3688.619 3713.520 3678.619 3490.226
## 1+VISIT+tretmnt+(1+VISIT|nwpd) 3127.625 3162.486 3113.625 2643.307
```

You'll fit the first model with lm(), and the others with lmer(). Comment briefly on any similarities or differences in how the different criteria choose random effects. (Note that only DIC and cAIC have a strong theoretical justification here).

There's a convergence warning for one of the models, which you can (and should) pursue using the methods in the R notes from lecture.

All four information criteria strongly favor at least a random intercept (`lm.7` *and* `lmer.9` *are very strongly disfavored). Among the remaining two models, the model with random slope and random intercept is strongly favored by all the criteria.*

Problem 3(c).

Repeat part (b) but with the interaction `VISIT * treatmnt` in each model instead of just the main effects `VISIT + treatmnt`.

```
lm.11 <- lm(sqrt.CD4PCT ~ 1 + VISIT * treatmnt,data=cd4)
lmer.12 <- lmer(sqrt.CD4PCT ~ 1 + VISIT * treatmnt + (1|newpid),data=cd4,REML=F)
lmer.13 <- lmer(sqrt.CD4PCT ~ 1 + VISIT * treatmnt + (0+VISIT|newpid),data=cd4,REML=F)
lmer.14 <- lmer(sqrt.CD4PCT ~ 1 + VISIT * treatmnt + (1+VISIT|newpid),data=cd4,REML=F)

IC.table(lm.11,lmer.12,lmer.13,lmer.14)
```

```
##                                     AIC      BIC      DIC     cAIC
```

```
## 1 + VISIT * treatmnt           4010.343 4035.244 4010.343 4010.343
## 1+VISIT*treatmnt+(1|newpid)    3155.181 3185.062 3143.181 2729.893
## 1+VISIT*tretmnt+(0+VISIT|nwpd) 3690.150 3720.030 3678.150 3491.763
## 1+VISIT*tretmnt+(1+VISIT|nwpd) 3129.215 3169.056 3113.215 2644.733
```

The story here is essentially the same as in part (b): The model with random slope and random intercept is strongly favored by all four criteria.

Not part of the required answer here, but, comparing parts (b) and (c) it seems like the cross-level interaction is not needed.

Problem 4.

IMRAD and IDMRAD.

Problem 4(a).

(You do not need to turn anything in for this part.) Compare the papers

- `COVID breakthrough rates in England - PIIS1473309921004606.pdf` and `COVID breakthrough... (appendix) 1-s2.0-S1473309921004606-mmc1.pdf`
- `menu pricing IDMRAD paper with technical appendix.pdf`

The COVID paper is a published IMRAD paper with a separate technical appendix. The "menu pricing'' paper is a simple IDMRAD paper (with technical appendix attached) that you can use as a model for your final project paper. Note similarities and differences. In particular, when you write your paper, the technical appendix should show all the R code and results that you used to answer the questions posed for the data analysis (A small portion of the results in the appendix actually makes it into the main paper). This is a little different from the published COVID paper, which really just provides some supplementary details in the technical appendix that didn't fit in the main paper.

There really is nothing to turn in here. I just want you to see

- *An example of a "data" section (present in the "menu pricing" paper but not in the COVID paper)*
- *Difference in the content of the technical appendices of the two papers*
  - *The technical appendix of the COVID paper doesn't contain much more computational detail than is in the main paper. It is mainly other supplemental information for the paper.*
  - *The technical appendix for the "menu pricing" paper is much more all of the technical detail and computation that justifies or "backs up" the assertions in the main part of the paper. With the more computational technical appendix I will be able to see how your computation corresponds to what you have written in the main body of the paper.*

  Note also all of the cross-references in the main body of the paper to specific sections and pages of the technical appendix, in <u>both</u> papers. This is really needed so that I do not have to leaf through your appendix to find the right computation each time I want to check an assertion you made in the main part of your paper. **Please include lots and lots of cross-references to specific pages of the technical appendix when you write your IDMRAD paper!**
- *The liberal use of subsections in the COVID paper to make material easy to find, for a reader who is in a hurry.* **Please use subsections to organize the material in your IDMRAD paper to make things easy to find for a reader in a hurry—whether it is me, or the TA, or your peer reviewers!**

Problem 3(b).

(You do need to turn something in for this part.) Refer again to the two papers in part (a). The difference between an IMRAD paper and an IDMRAD paper is that the "Data'' part of the **Methods** section of an IMRAD paper is taked out of **Methods** and made into its own **Data** section between **Introduction** and **Methods**. \[-0.75em]

Find the **Methods** section of the COVID paper, several pages in, and within the **Methods** section find the subsections that describe the data (rather than describing methods of analysis). Copy and paste these sections (including subsection titles) into a separate **Data** section, and turn that section in as an answer to this question.

The **Data** *section should consist of the following three subsections of the* **Methods** *section of this paper:*

- *Study design and participants*
- *Risk factor variable definitions*
- *Disease severity, duration, and symptom definitions*

The full data section would look like this:

Data.

Study design and participants

This prospective, community-based, nested, case-control study used data from UK-based, adult (≥ -18 years) participants of the COVID Symptom Study logged through a free mobile phone app developed by ZOE (London, UK) and King's College London (London, UK).[22] *The app was launched in the UK on March 24, 2020, and by July 4, 2021, had nearly 4 · 5 million unique participants providing data by self-report or proxy report. At registration, each participant reported baseline demographic information (eg, age, sex, ethnicity, weight, height, and health-care worker status), geographical location, and information on health risk factors, including comorbidities, lifestyle, frailty, visits to hospital, and adherence to mask-wearing guidance. Participants were encouraged to self-report any of 32 pre specified symptoms (appendix p 1) daily, providing prospective longitudinal information on incident symptoms. Those with new symptoms were prompted to book and take a SARS-CoV-2 test. All users were encouraged to record any SARS-CoV-2 testing results (whether prompted by the app or otherwise), and, from Dec 11, 2020, any SARS-CoV-2 vaccination and subsequent symptoms.*[22] *Users with missing or inconsistent information were excluded from our analysis. The inclusion process for cases and controls is shown in the appendix (p 14).*

Cases had received a first or second dose of a COVID-19 vaccine since Dec 8, 2020; had either a positive RT-PCR test or lateral flow antigen test (LFAT) at least 14 days after their first vaccination (but before their second; cases 1) or a positive RT-PCR test or LFAT at least 7 days after their second vaccination (cases 2); and had no positive SARS-CoV-2 test before vaccination. If more than one positive test result was reported, only the first positive test was selected. To identify risk factors for postvaccination infection, we selected two control groups among the vaccinated (since Dec 8, 2020) UK-based adult users of the COVID Symptom Study app who had not tested positive for SARS-CoV-2 before vaccination: a control group of users reporting a negative RT-PCR test or LFAT at least 14 days after their first vaccination but before their second (controls 1) and a control group of users reporting a negative RT-PCR test or LFAT at least 7 days after their second vaccination (controls 2). Controls 1 and controls 2 were matched (1:1) with cases 1 and cases 2, respectively, by use of the date of the postvaccination COVID-19 test, health-care worker status, and sex. If multiple negative tests were reported, the last test date was used for matching.

To compare the disease profile of SARS-CoV-2 infection before and after vaccination, we sub-selected participants from cases 1 and cases 2 who had used the app for at least 14 consecutive days after testing positive for SARS-CoV-2 (denoted as cases 3 and cases 4, respectively), so that symptoms of infection could be assessed. Controls for the disease profile analysis were those who reported a SARS-CoV-2-positive RT-PCR test or LFAT, were unvaccinated until data censoring, and had used the app for at least 14 consecutive days after the test. Among these users, two groups were formed: controls 3 and controls 4, matching (1:1) with cases 3 and cases 4, respectively, by the date of the positive COVID-19 test, health-care worker status, sex, body-mass

index (BMI), and age. Individuals in all case and control groups who did not report an RT-PCR or LFAT test after Dec 8, 2020, were excluded.

For all control groups, we used a matching algorithm based on minimum Euclidean distance[23] *between the vectors of the covariates, with age, BMI, and the date of the test as numerical variables, and sex as a binary variable multiplied by 100 to ensure balance between covariate strengths. We considered health-care worker status, coded as a a categorical variable in the app, as a numerical variable (0=not a health-care worker; 1=health-care worker who does not interact with patients; 2=health-care worker who does not treat patients; 3=health-care worker who interacts with patients; 4=health-care worker who treats patients). Participants could only choose one of these options.*

All app users provided digital informed consent for data usage for COVID-19-related research. In the UK, the app and the study were approved by King's College London's ethics committee (REMAS number 18210; reference LRS-19/20–18210).

Risk factor variable definitions

For this analysis, the outcome variable was case status (a self-reported positive RT-PCR test or LFAT for SARS-CoV-2). We considered a priori-defined risk factors for SARS-CoV-2 infection based on previous evidence in unvaccinated individuals:16–18 age; BMI; self-reported comorbidities (ie, cancer, diabetes, asthma, lung disease, heart disease, and kidney disease), analysed individually as binary variables; dependency level (frailty) assessed by the PRISMA-7 questionnaire, which is embedded in app registration,[24,25] *as a binary variable (PRISMA-7 score ≥ -3 defined as frail and <3 defined as not frail);*[26] *local area Index of Multiple Deprivation (IMD; a score ranging from 1 [most deprived] to 10 [least deprived] estimating relative locality deprivation derived from postal code and lower layer super output area) divided into low IMD (1–3), intermediate IMD (4–7), and high IMD (8–10) groups;*[27] *and four healthy lifestyle factors (no current smoking, no obesity [BMI <30 kg/m²], physical activity at least once per week [non-sedentary], and a healthy diet pattern; appendix p 17). We also calculated a healthy lifestyle score on the basis of these four lifestyle factors,*[28] *by which participants received 1 point for each healthy lifestyle factor and the sum of the scores gave a total healthy lifestyle score ranging from 0 to 4, with higher scores indicating a healthier lifestyle (appendix p 17).*

Disease severity, duration, and symptom definitions

To compare the disease profile in vaccinated versus unvaccinated individuals testing positive for SARS-CoV-2, we assessed disease severity (asymptomatic or sympto- matic; more than five symptoms or five or fewer symptoms reported in the first week of illness;20 and self-reported presentation to hospital or no hospital presentation), illness duration (duration of <28 days or ≥ -28 days), and individual symptom reports. Vaccination status was the exposure. For cases, controls 3, and controls 4, symptoms were considered within a window between 3 days before the COVID-19 test date and up to 14 days after the test date (appendix p 1). This window was used because it might have taken up to 3 days to request an RT-PCR test and receive a result following symptom onset, and symptoms can occur up to 14 days following SARS-CoV-2 exposure.[29]