# Homework 5

*Allan Kimaina*

# Chapter 7 Question 2:

Continuous probability simulation: the logarithms of weights (in pounds) of men in the United States are approximately normally distributed with mean 5.13 and standard deviation 0.17; women with mean 4.96 and standard deviation 0.20. Suppose 10 adults selected at random step on an elevator with a capacity of 1750 pounds.

`What is the probability that the elevator cable breaks?`

[1] 0.043

We assumed the proportion of women to men is .52 (provided in the chapter but not in this question)

# Chapter 7 Question 8:

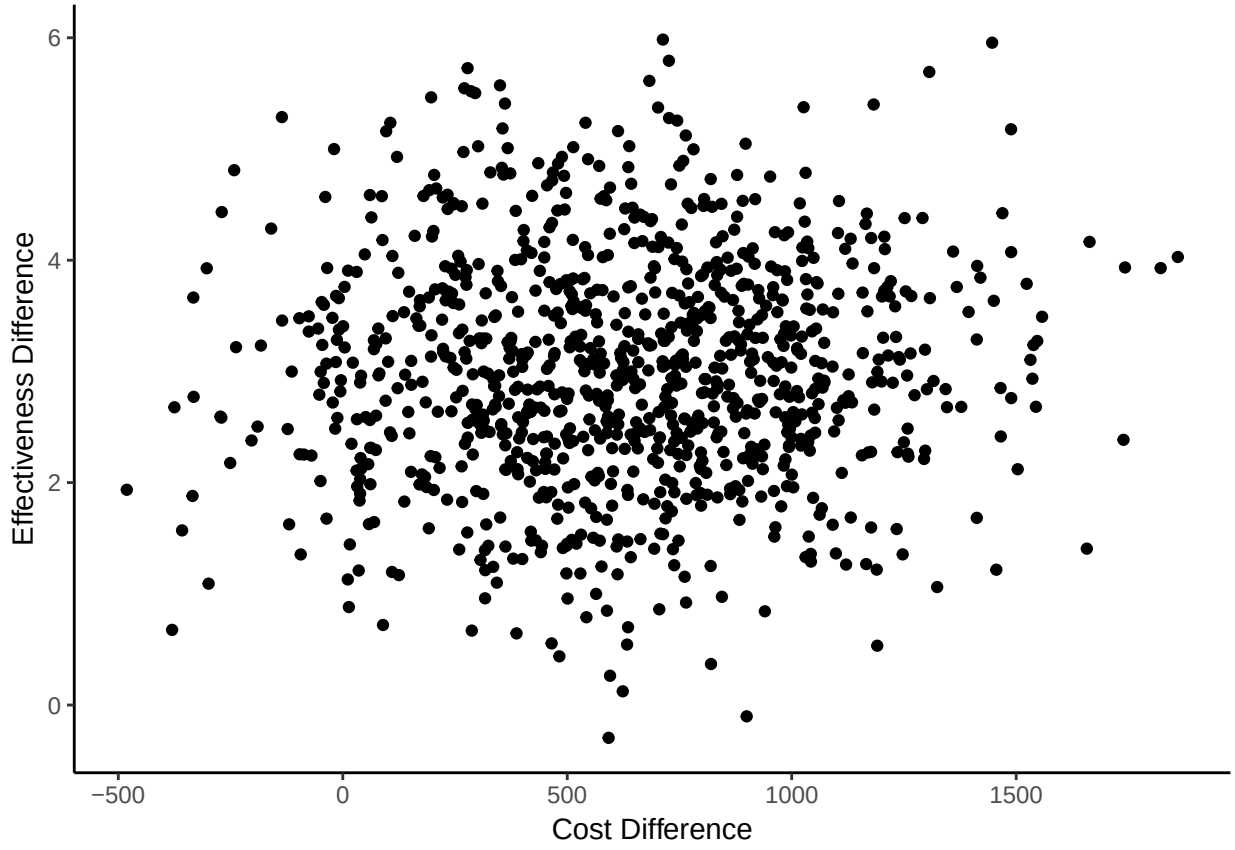
Inference for the ratio of parameters: a (hypothetical) study compares the costs and effectiveness of two different medical treatments.

- In the first part of the study, the difference in costs between treatments A and B is estimated at $600 per patient, with a standard error of $400, based on a regression with 50 degrees of freedom.
- In the second part of the study, the difference in effectiveness is estimated at 3.0 (on some relevant measure), with a standard error of 1.0, based on a regression with 100 degrees of freedom.
- For simplicity, assume that the data from the two parts of the study were collected independently.

Inference is desired for the incremental cost-effectiveness ratio: the difference between the average costs of the two treatments, divided by the difference between their average effectiveness. (This problem is discussed further by heitjan, Moskowitz, and Whang, 1999.)

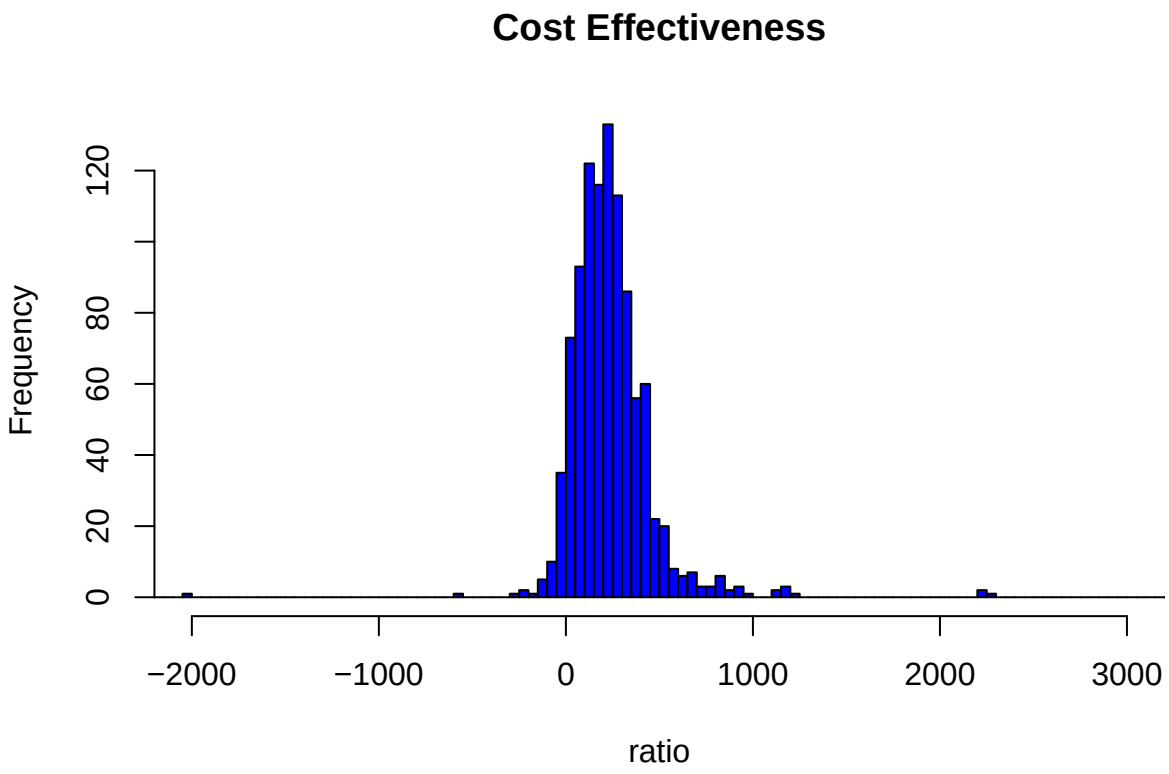
## Part A:

`Create 1000 simulation draws of the cost difference and the effectiveness difference, and make a scatterplot of these draws.`




## Part B:

`Use simulation to come up with an estimate, 50% interval, and 95% interval for the incremental cost-effectiveness ratio.`

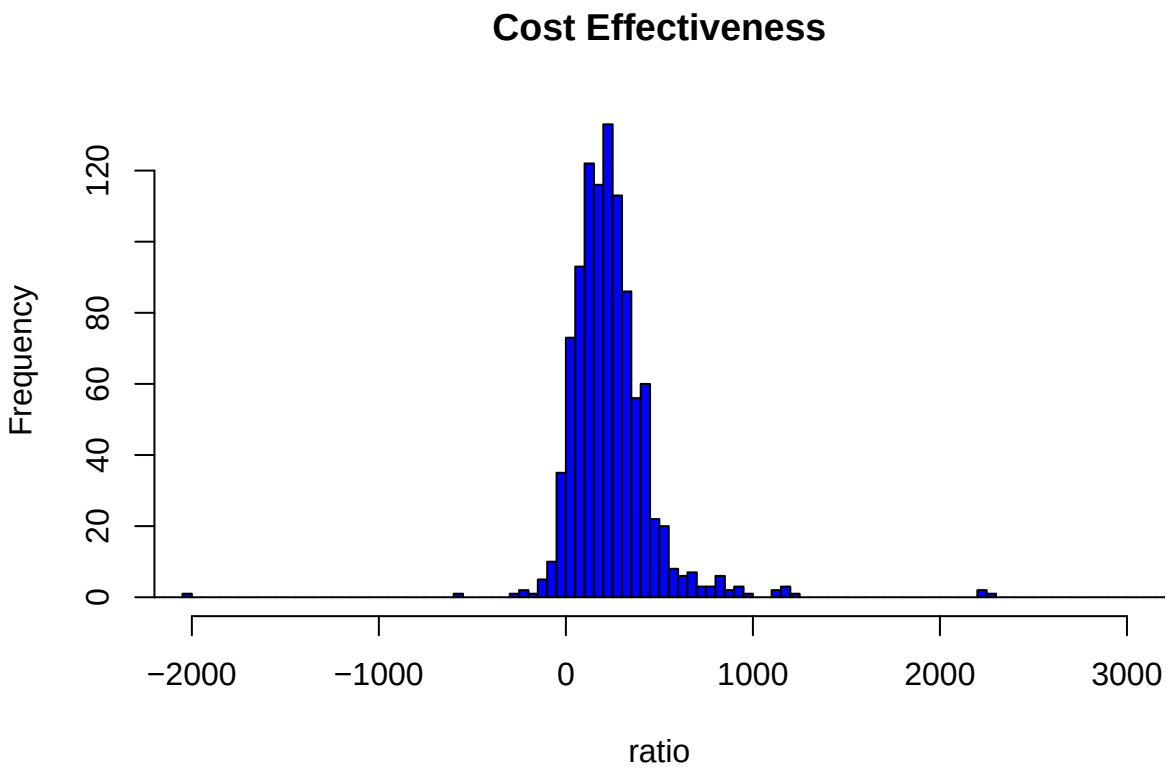



25% 75%

112.5502 323.3060 2.5% 97.5% -38.53523 737.84576

### Part C:

Repeat this problem, changing the standard error on the difference in effectiveness to 2.0.




25% 75%

112.5502 323.3060 2.5% 97.5% -38.53523 737.84576

# Chapter 8 Question 1:

Fitting the wrong model: suppose you have 100 data points that arose from the following model: y =3+0 .1x1+0.5x2+error, with errors having a t distribution with mean 0, scale 5, and 4 degrees of freedom.We shall explore the implications of fitting a standard linear regression to these data.

## Part A:

`Simulate data from this model. For simplicity, suppose the values of x1 are simply the integers from 1 to 100, and that the values of x2 are random and equally likely to be 0 or 1.3 Fit a linear regression (with normal errors) to these data and see if the 68% confidence intervals for the regression coefficients (for each, the estimates ±1 standard error) cover the true values.`

| | Model 8A |
|---|---|
| (Intercept) | 3.30 (0.22)*** |
| x1 | 0.10 (0.00)*** |
| x2 | 0.45 (0.21)* |
| $R^2$ | 0.89 |
| Adj. $R^2$ | 0.89 |
| Num. obs. | 100 |
| RMSE | 1.03 |

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

Table 1: Statistical models

% latex table generated in R 3.4.3 by xtable 1.8-2 package % Fri Apr 27 16:57:39 2018

| | Coef | Lower | Upper | Coverage |
|---|---|---|---|---|
| 1 | Intercept | 3.08 | 3.52 | FALSE |
| 2 | X1 | 0.09 | 0.10 | TRUE |
| 3 | X2 | 0.24 | 0.66 | TRUE |

After generating 68% confidence intervals for the regression coefficient, all the point estimates except Intercept were not contained in the 68% confidence intervals. However the CI band was too wide making high uncertainty in the point estimate. Repeating this simulation without setting seed would definitely result in difference coverage conclusion due to this wide CI

## Part B:

`Put the above step in a loop and repeat 1000 times. Calculate the confidence coverage for the 68% intervals for each of the three coefficients in the model.`

% latex table generated in R 3.4.3 by xtable 1.8-2 package % Fri Apr 27 16:57:44 2018

| | Coef | Lower | Upper | Coverage |
|---|---|---|---|---|
| 1 | Intercept | 2.77 | 3.23 | TRUE |
| 2 | X1 | 0.10 | 0.10 | TRUE |
| 3 | X2 | 0.29 | 0.70 | TRUE |

After simulating the previous step 1000 times, we calculated the mean of all the point estimates and standard error. All the 3 estimates were contained in the 68% confidence intervals.

## Part C:

`Repeat this simulation, but instead fit the model using t errors (see Exercise 6.6).`

% latex table generated in R 3.4.3 by xtable 1.8-2 package % Fri Apr 27 16:57:49 2018

| | Coef | Lower | Upper | Coverage |
|---|---|---|---|---|
| 1 | Intercept | 1.65 | 4.21 | TRUE |
| 2 | X1 | 0.08 | 0.12 | TRUE |
| 3 | X2 | -0.61 | 1.68 | TRUE |

After the previous step 1000 times using t errors, we calculated the mean of all the point estimates and standard error. All the 3 estimates were contained in the 68% confidence intervals. However the standard error were inflated making the CI band too wide compared to the previous model

# Chapter 8 Question 4:

Model checking for count data: the folder risky.behavior contains data from a study of behavior of couples at risk for HIV; see Exercise 6.1.

## Part A:

`Fit a Poisson regression model predicting number of unprotected sex acts from baseline hIV status. Perform predictive simulation to generate 1000 datasets and record both the percent of observations that are equal to 0 and the percent that are greater than 10 (the third quartile in the observed data) for each. Compare these values to the observed value in the original data.`

### Fit a Poisson regression model

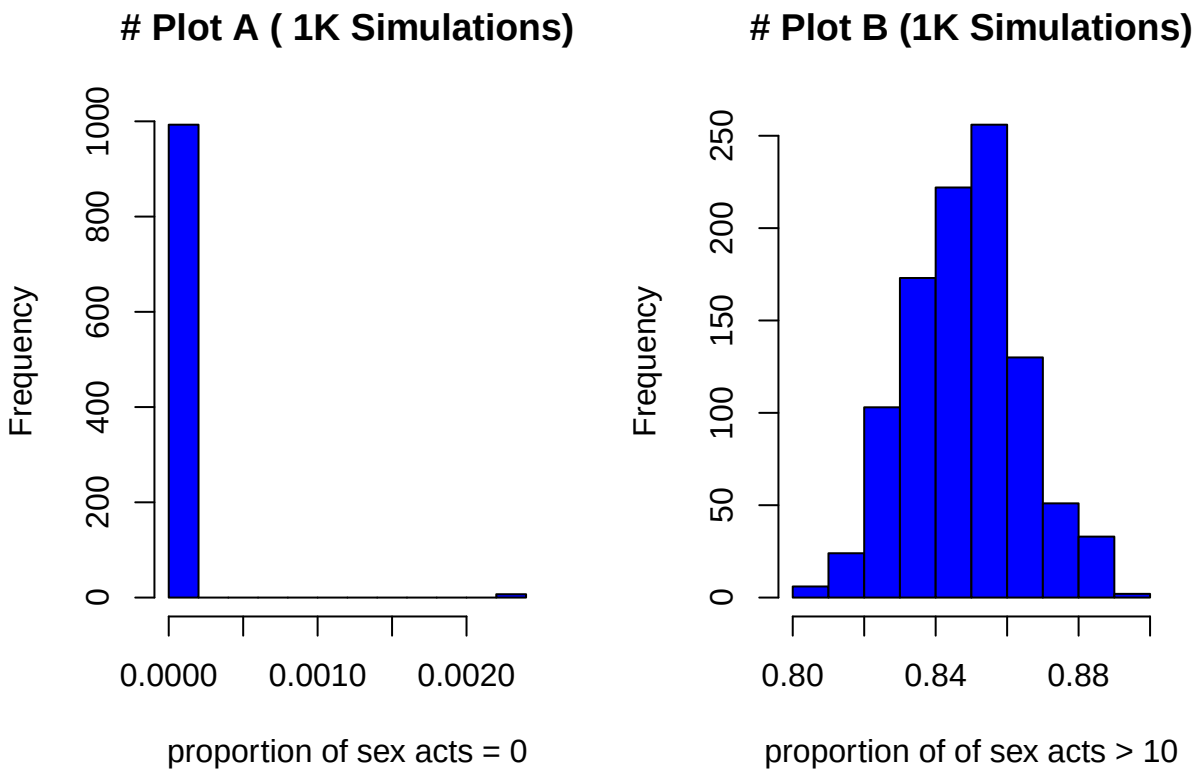
| | Model 1: Poisson |
|---|---|
| (Intercept) | 2.91 (0.01)*** |
| bs_hivpositive | −0.62 (0.03)*** |
| AIC | |
| BIC | |
| Log Likelihood | |
| Deviance | 12938.74 |
| Num. obs. | 434 |

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

Table 2: Statistical models

This model does not fit well, the rule of thumb dictates that the ratio of residual deviance to degree of freedom should be close to 1, however we get a value which is way far from 1.

**1000 Simulations**




**Plot A: Frequency of number of unprotected sex acts at followup = 0**

```
##     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 0.000e+00 0.000e+00 0.000e+00 1.843e-05 0.000e+00 2.304e-03
```

The actual dataset had 29% of the observation having 0 number of unprotected sex acts at followup. However, after performing predictive simulation to generate 1000 datasets, we found out that the percentage of hypothesized observations with number of unprotected sex acts at followup equal to 0 ranged from 0% to 0.2% with most of replication at 0% - all of which were much much lower than the observed test statistic of 29%.

**Plot B: Frequency of number of unprotected sex acts at followup > 10**

```
##  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.7995  0.8364  0.8479  0.8485  0.8594  0.9101
```

The actual dataset had 36% of the observations having number of unprotected sex acts at followup that is greater than 10. However, after performing predictive simulation to generate 1000 datasets, we found out that the percentage of hypothesized observations with number of unprotected sex acts at followup greater than 10 ranged from 84% `(1stIQR)` to 86% `(3rdIQR)` with an average of 85% - all of which were much much higher than the observed test statistic of 36%.

## Part B:

`Repeat (a) using an overdispersed Poisson regression model.`

### Fit an overdispersed Poisson regression model

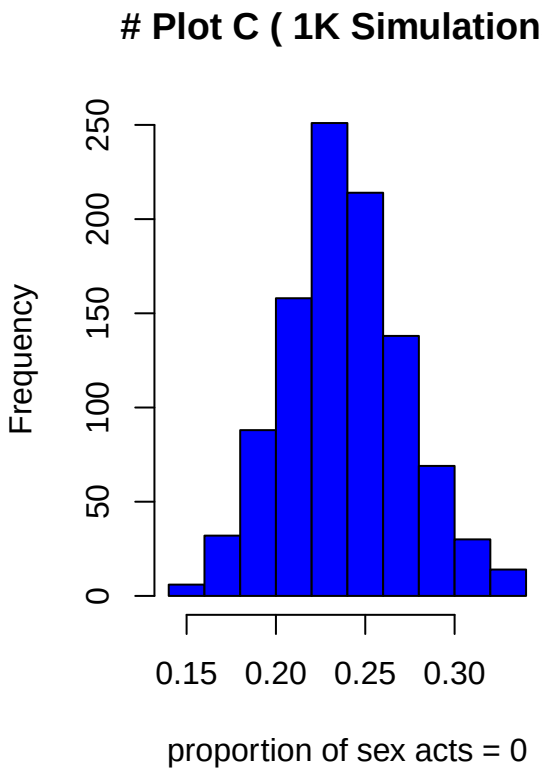
| | Model 2: overdispersed Poisson |
|---|---|
| (Intercept) | 2.91 (0.08)*** |
| bs_hivpositive | −0.62 (0.23)** |
| AIC | |
| BIC | |
| Log Likelihood | |
| Deviance | 12938.74 |
| Num. obs. | 434 |

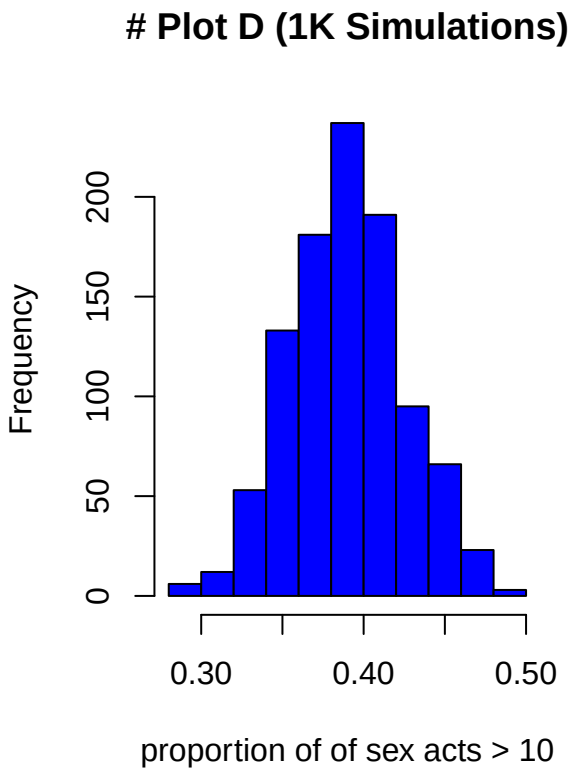
$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

Table 3: Statistical models

Much better model but it does not fit as well, the ratio of residual deviance to degree of freedom is still far from 1.

### 1000 Simulations



**Plot C: Frequency of number of unprotected sex acts at followup = 0**

```
##  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.1290  0.2166  0.2373  0.2388  0.2604  0.3364
```

After performing predictive simulation based on overdispersed model, we found out that the percentage of hypothesized observations with number of unprotected sex acts at followup equal to 0 varied from 21% to 26%, with most of replication averaging at 24% - all of which were reasonably much closer to the observed test statistic of 29% compared to the previous model. In summary, this tells us that this model reasonably fits this aspect of data well, however other aspects of this data might not be well fit by this model.

**Plot D: Frequency of number of unprotected sex acts at followup > 10**

```
##  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.2834  0.3664  0.3871  0.3881  0.4124  0.4862
```

The actual dataset had 36% of the observations having number of unprotected sex acts at followup that is greater than 10. After the predictive simulation, we found out that the percentage of hypothesized observations with number of unprotected sex acts at followup greater than 10 varied from 37% `(1stIQR)` to 41% `(3rdIQR)` with an average of 39% - all of which were reasonably much closer to the observed test statistic of 36% than the previous model. This tells us that this model reasonably fits this aspect of data well, however other aspects of this data might not be well fit by this model.

## Part C:

`Repeat (b), also including ethnicity and baseline number of unprotected sex acts as input variables.`

**Fit an overdispersed Poisson regression model**

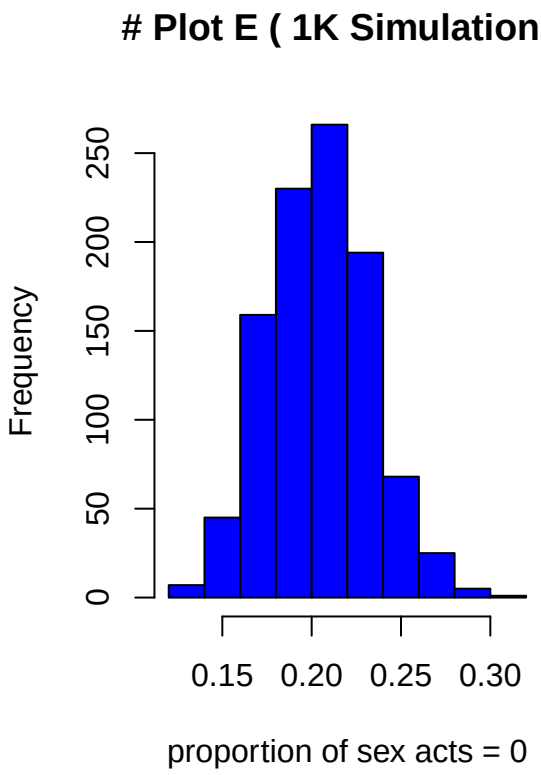
| | Model 3: overdispersed Poisson |
|---|---|
| (Intercept) | 2.54 (0.09)*** |
| bs_hivpositive | −0.50 (0.20)* |
| bupacts | 0.01 (0.00)*** |
| AIC | |
| BIC | |
| Log Likelihood | |
| Deviance | 10707.81 |
| Num. obs. | 434 |

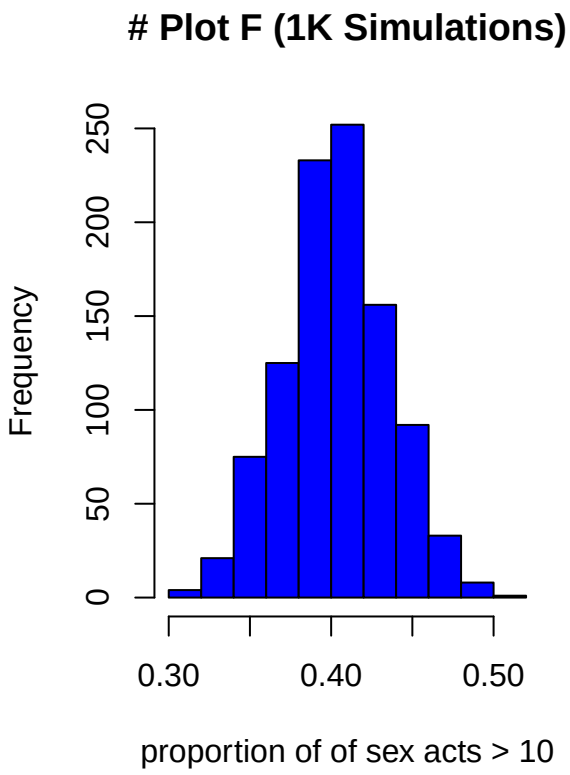
$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

Table 4: Statistical models

Much better model than the previous model but it does not meet the standard, the ratio of residual deviance to degree of freedom is 24 which is still far from 1.

**1000 Simulations**




**Plot E: Frequency of number of unprotected sex acts at followup = 0**

```
##    Min.  1st Qu. Median  Mean    3rd Qu. Max
##  0.1198  0.1843  0.2028  0.2041  0.2235  0.2857
```

After performing predictive simulation based on overdispersed model with added predictor (bupacts), we found out that the percentage of hypothesized observations with number of unprotected sex acts at followup equal to 0 varied from 18% to 22%, with most of replication averaging at 20% - most of which were closer to the observed test statistic of 29%

**Plot F: Frequency of number of unprotected sex acts at followup > 10**

```
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.3111  0.3802  0.4032  0.4033  0.4263  0.4908
```

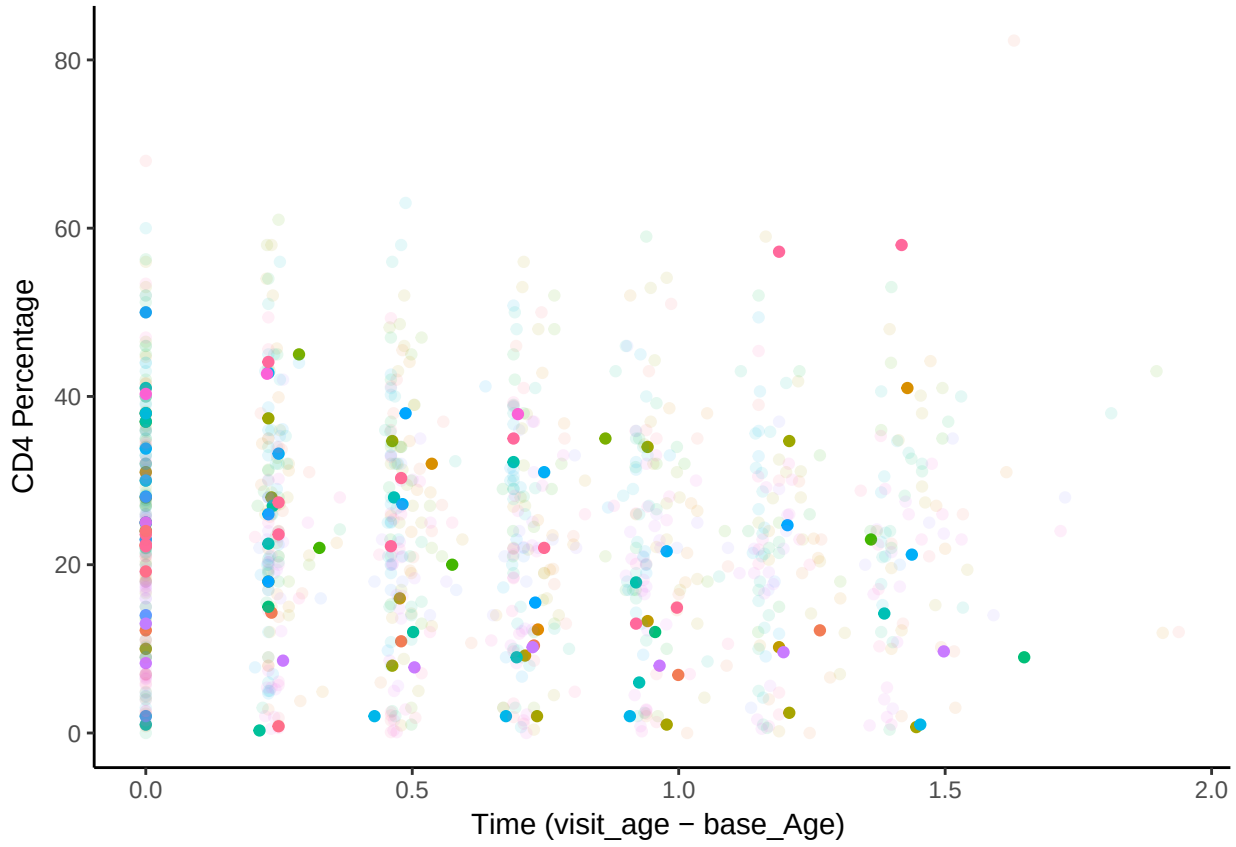
The actual dataset had 36% of the observations having number of unprotected sex acts at followup that is greater than 10. After the predictive simulation, we found out that the percentage of hypothesized observations with number of unprotected sex acts at followup greater than 10 varied from 38% `(1stIQR)` to 42% `(3rdIQR)` with an average of 40% - all of which were closer to the observed test statistic of 36%.

In summary this model really does well in compromising and balancing between the 2 aspect of data we have accessed. The upshot is that this model fit the data much much better compared to the previous 2 model

# Chapter 11 Question 4:

The folder cd4 has CD4 percentages for a set of young children with HIV who were measured several times over a period of two years. The dataset also includes the ages of the children at each measurement.

**Time Function**



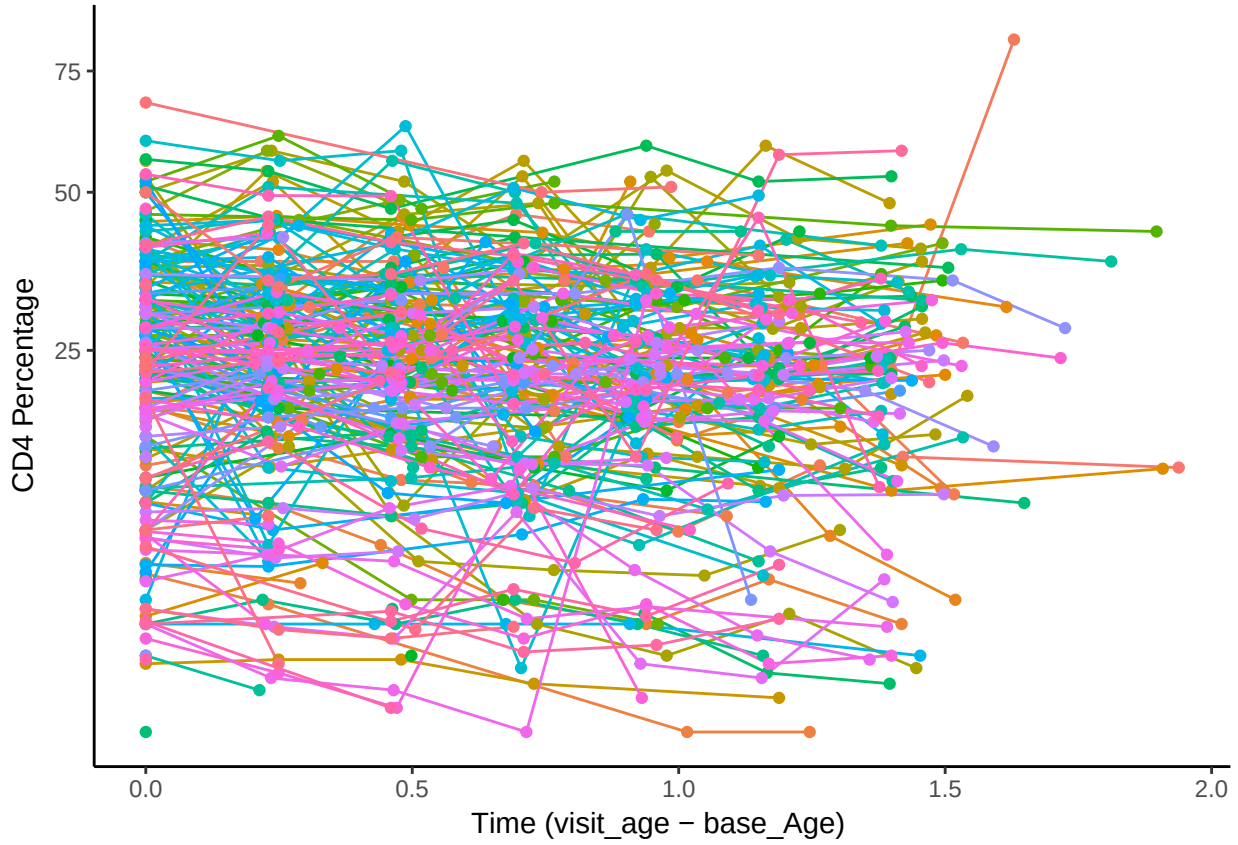

We define our time function as visit_age - base_Age. This ensured that all children visit started from day 0 and had a serial/sequential progression of events as shown in the graph above. Some data points (children) have been grey scaled for interchangeability.

It is also interesting to note that most visit occurred quarterly

## Part A:

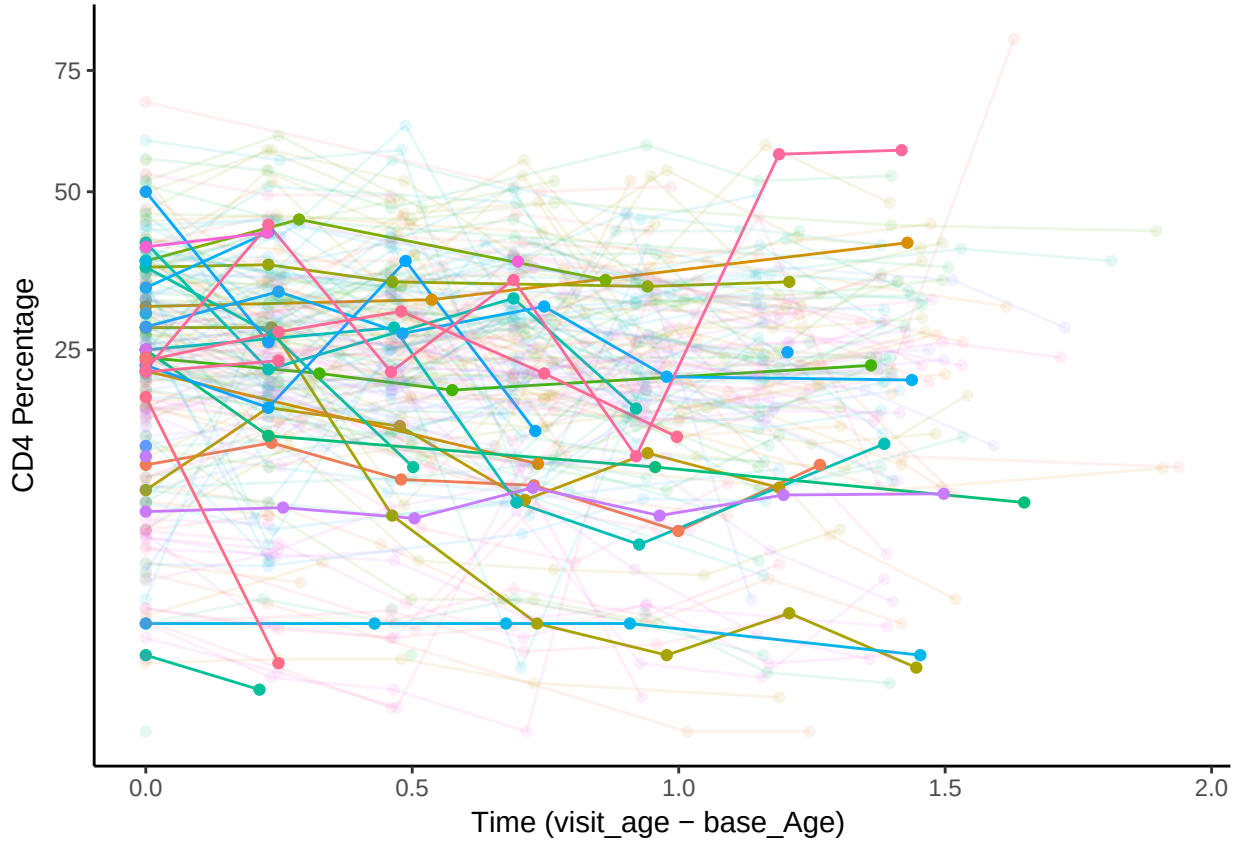
`Graph the outcome (the CD4 percentage, on the square root scale) for each child as a function of time.`

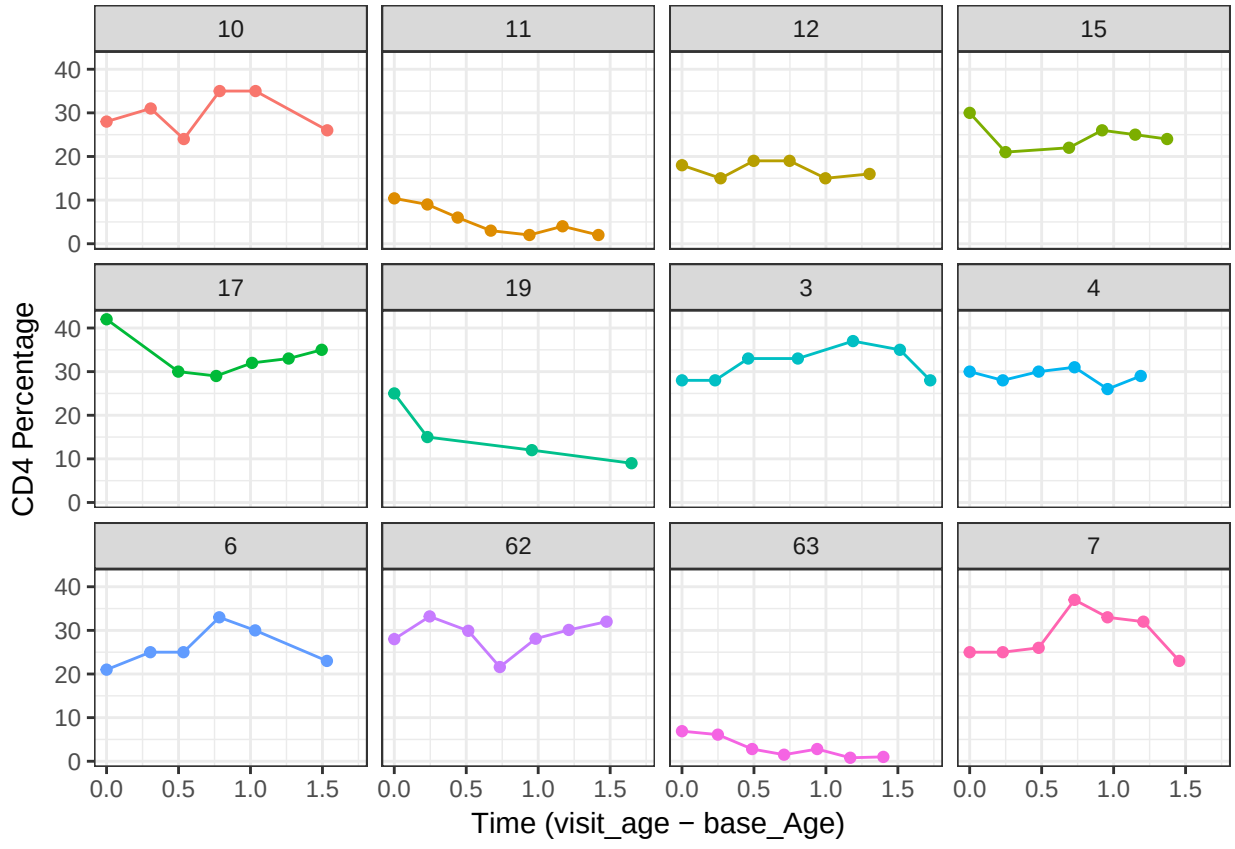
**Overall Plot - all 254 children**




The plot above is difficult synthesize and inter prate, let's try plotting 5% of the children. The below shows 5% of the children (colored) and the remaining 90% of children (grey scale).

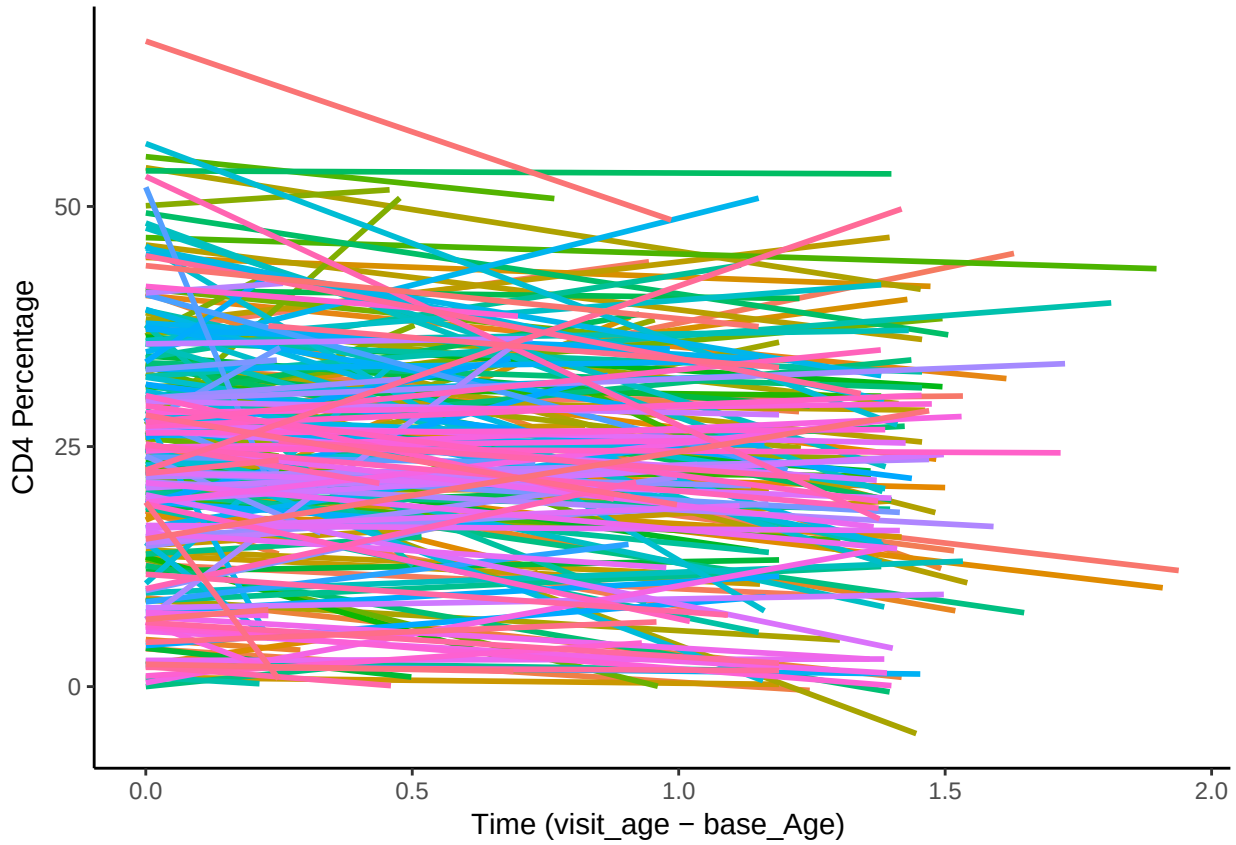
**Plot for 5% of all children**



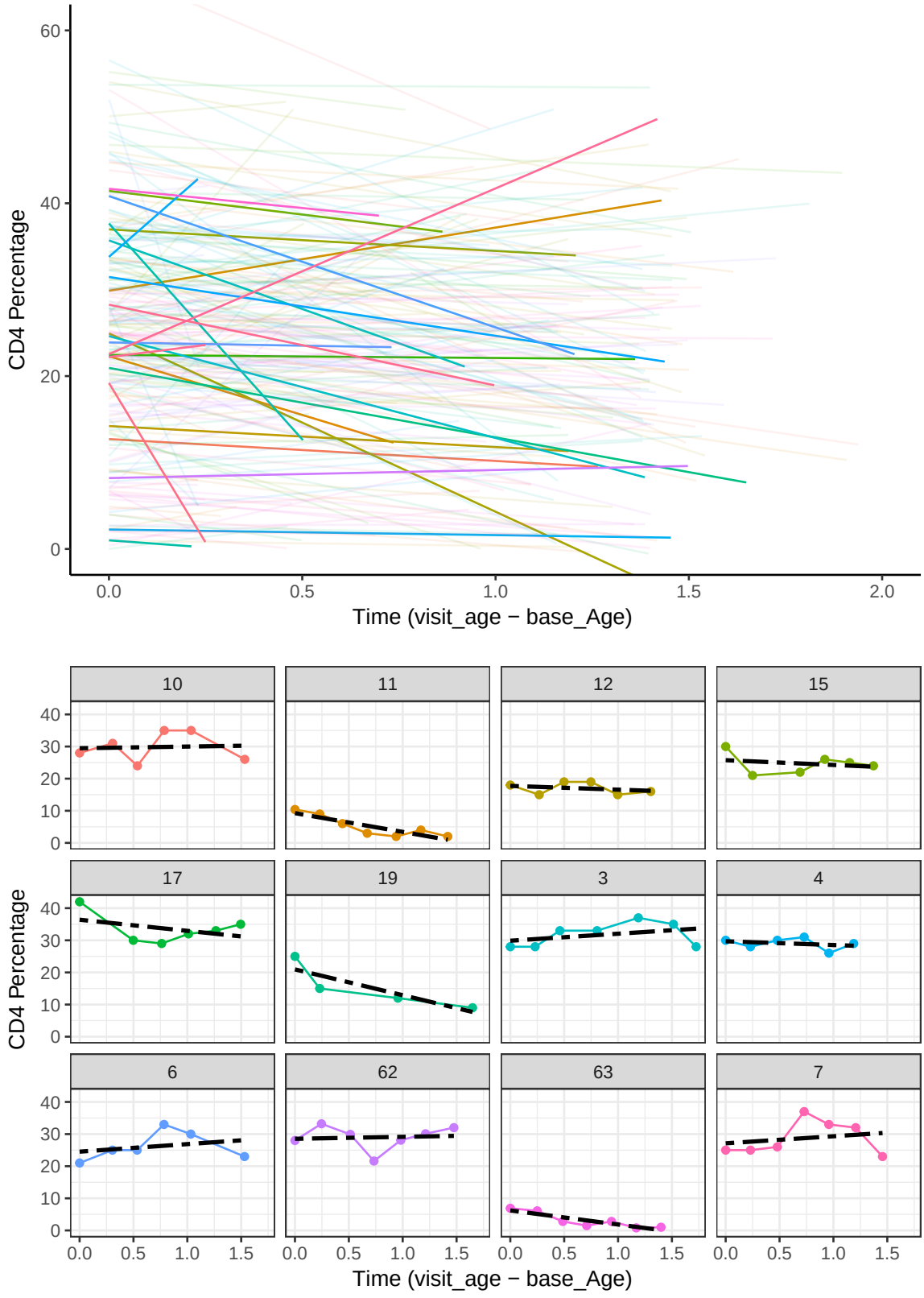



## Part B:

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

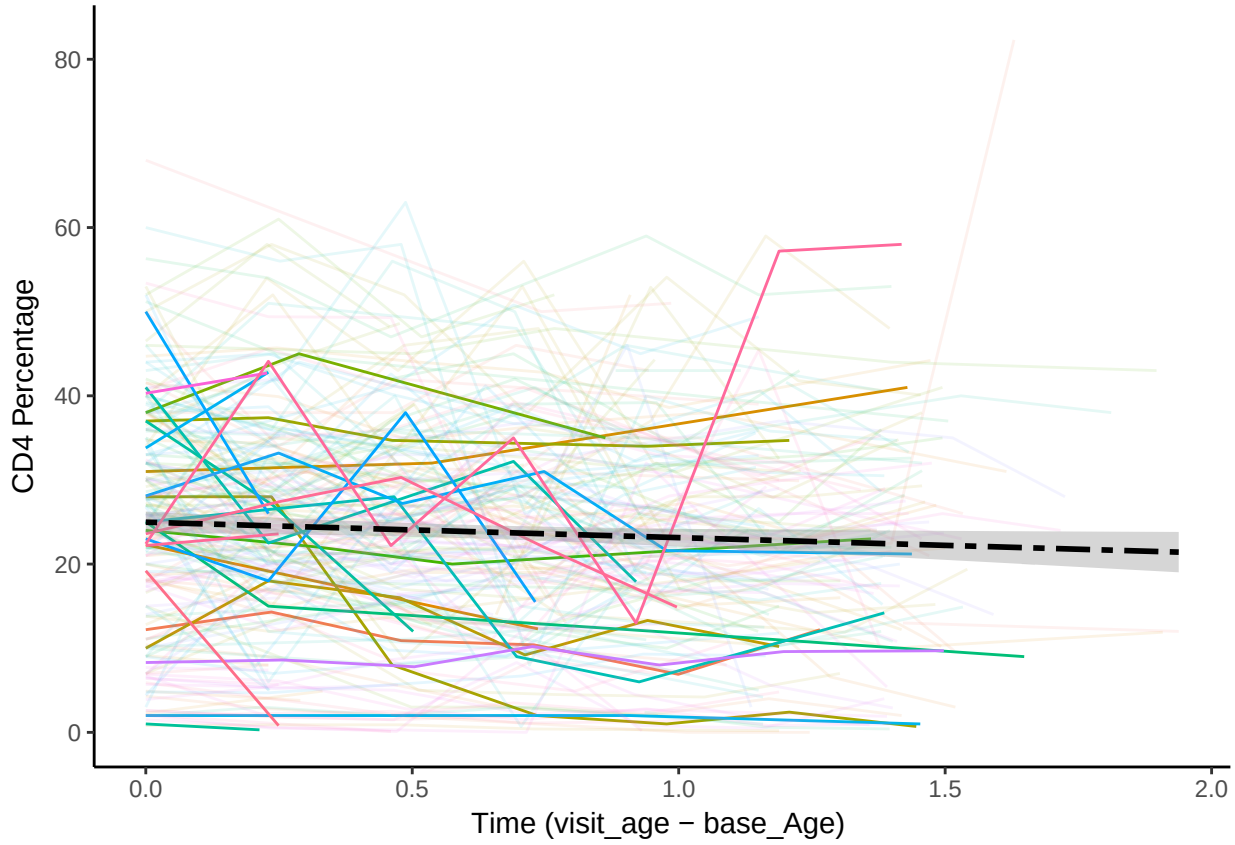



From the above plot, it is difficult to decipher the magnitude and direction of each regression line. The next page provides a plot for only a subset of these children which is more interpretable.




For some children, cd4 percent increases over time. However, for most children, cd4 percent decreases with time indicating that time is negatively associated with the outcome.

This is rather odd! we expect that most of the children to have increasing cd4 percent over time. Lets take a look at the general trend of all children in general (Group level regression) as shown below.




The dotted line is the population level regression and apparently it has a negative slope. It's likely that some treatment/ARVs works better than others or maybe other child level factors might be confounded with the design variables. This is beyond the scope of this analysis, therefore we shall not make any adjustment or include other predictors.

## Part 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.`

| | Intercept Model | Slope Model |
|---|---|---|
| (Intercept) | 28.66 (0.35)*** | −3.77 (0.28)*** |
| baseage | −1.05 (0.08)*** | 0.17 (0.06)** |
| treatmnt2 | 3.32 (0.34)*** | −0.51 (0.27) |
| $R^2$ | 0.05 | 0.00 |
| Adj. $R^2$ | 0.05 | 0.00 |
| Num. obs. | 5722 | 5693 |
| RMSE | 12.61 | 10.24 |

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

Table 5: Statistical models

# Chapter 12 Question 2:

Continuing with the analysis of the CD4 data from Exercise 11.4:

## Part 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.`

| | Model 12.2 A |
|---|---|
| (Intercept) | 25.04 (0.80)*** |
| time | −3.00 (0.51)*** |
| AIC | 7889.03 |
| BIC | 7908.94 |
| Log Likelihood | -3940.52 |
| Num. obs. | 1072 |
| Num. groups: newpid | 250 |
| Var: newpid (Intercept) | 128.77 |
| Var: Residual | 53.19 |

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

Table 6: Statistical models

Given that this model is a random intercept model, it only accounts for "idiosyncratic" variation that is due to individual level differences in baseline cd4 percent. It also assumes that time effects on cd4 percent is fixed meaning that every child cd4 percent varies systematically with time.

- **Time** - Given that time is a fixed effect on every child, each year on the study is associated with a decreased in cd4 percentage by 3 % on average. In this model, time was statistically significant and had a substantial effect on cd4 percentage.

## Part 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.`

| | Model 12.2 B |
|---|---|
| (Intercept) | 27.71 (1.55)*** |
| time | −2.96 (0.51)*** |
| treatmnt2 | 1.21 (1.50) |
| baseage | −0.95 (0.33)** |
| AIC | 7884.23 |
| BIC | 7914.10 |
| Log Likelihood | -3936.12 |
| Num. obs. | 1072 |
| Num. groups: newpid | 250 |
| Var: newpid (Intercept) | 123.51 |
| Var: Residual | 53.21 |

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

Table 7: Statistical models

In general, adding child-level predictors (i.e, group-level predictors) definitely improves the model. In fact it increased the precision of our estimates. Note that the variance of our random effect estimate has reduced from 128 to 123 meaning some of the variations has been explained by addition of fixed effect predictors like base age and treatment. Conversely, residual has not reduced by much possibly because we haven't added another random effect predictor that is due to child-specific idiosyncrasies.

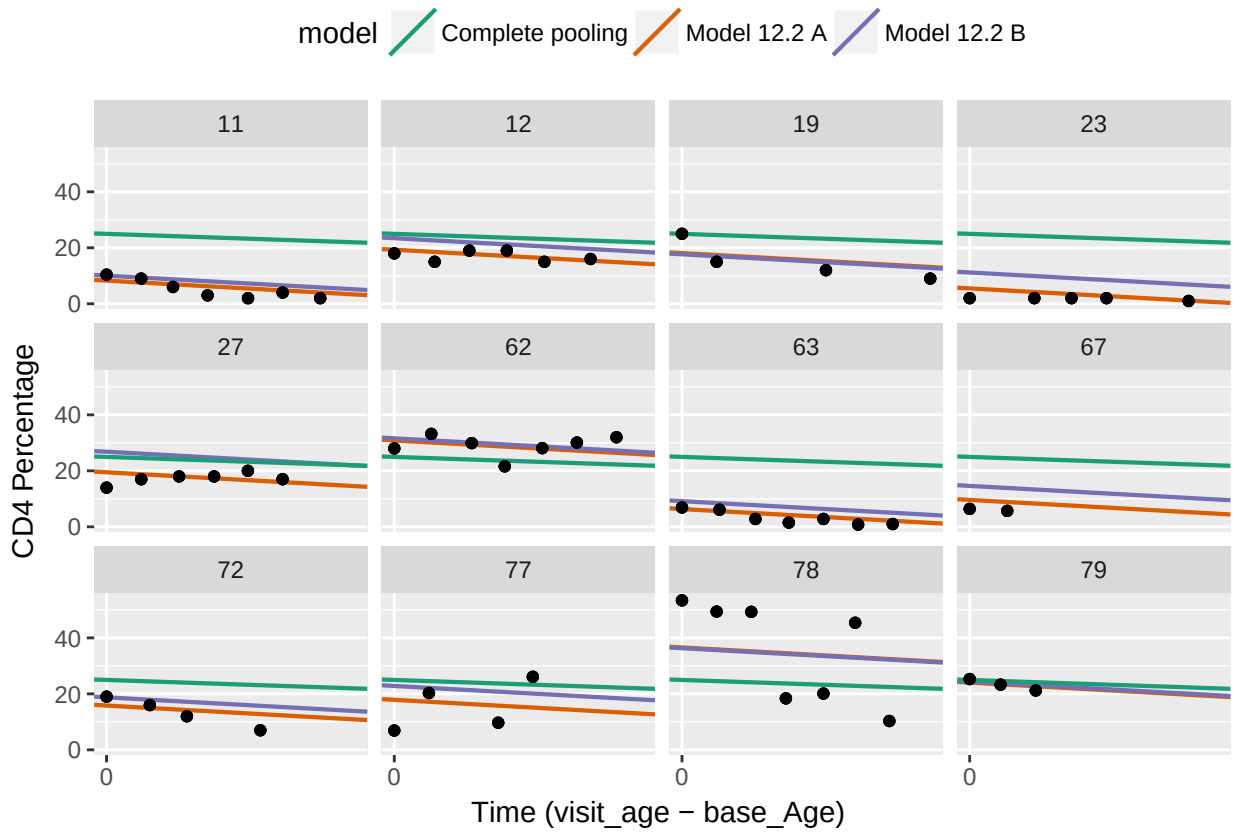
### Interpretation

- **Treatment** - Children who were on second treatment had 1.2125% higher cd4 percent on average than children on first treatment. This predictor had substantial effect on cd4 percent however it was insignificant.
- **Baseline Age** - 1 year difference in baseline age is associate with changing cd4 percentage by 0.9485 % on average. Meaning children who were enrolled into the study early had higher cd4 percentage. This predictor was highly significant.
- **Time** - Each year on treatment was associated with a decreased in cd4 percentage by 2.9615 % on average. This predictor was highly significant.

## Part C:

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

### Graphically



A few observation to make.

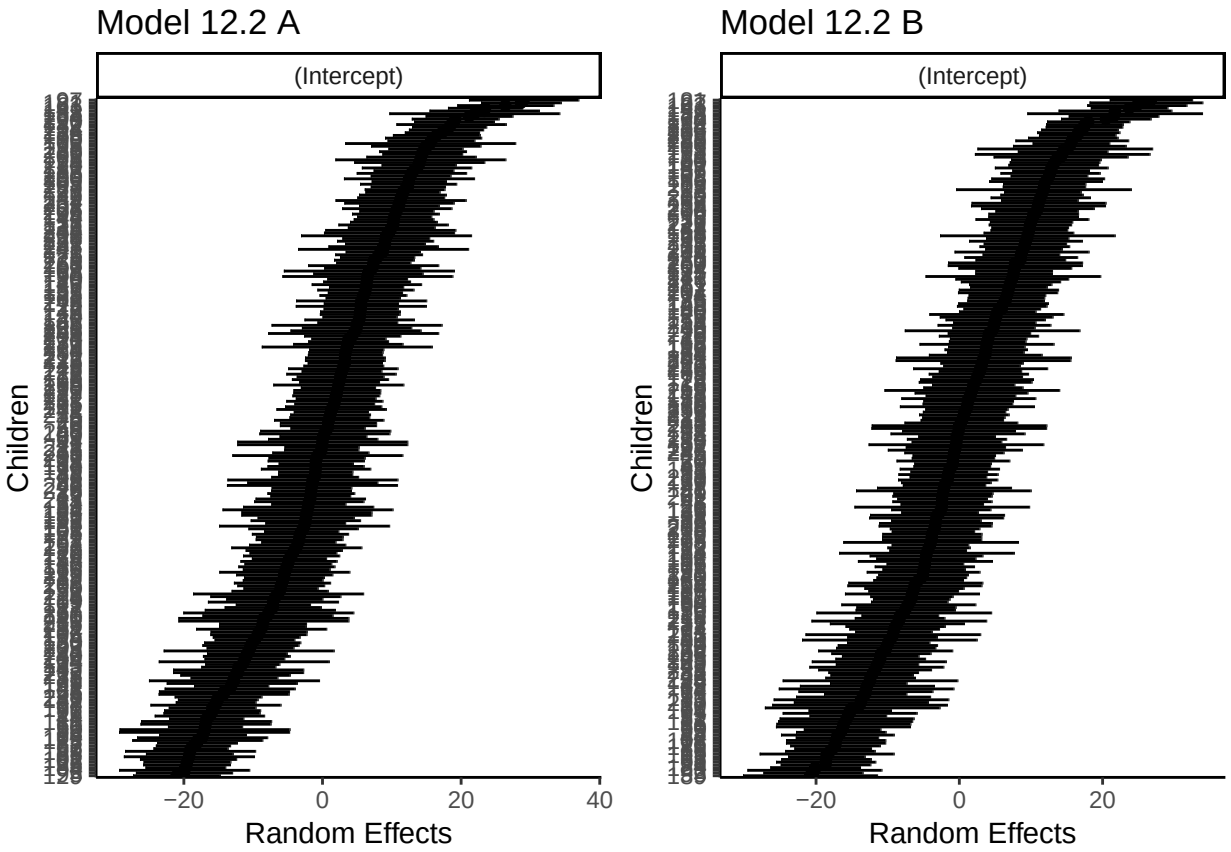
- From the plot, observe that the partial pooling model with added child-level predictors (Model 12.2B) is pulled substantially towards the complete pooling model (group average) compared to Model 12.2.A
- If you take a look at children with less observations (incomplete data), you will notice that Model 12.2.B is pulled more towards the complete pooling model (group average) compared to Model 12.2.A. See child 67, 79, 77

**Numerically**

% latex table generated in R 3.4.3 by xtable 1.8-2 package % Fri Apr 27 16:58:16 2018

| | Df | AIC | BIC | logLik | deviance | Chisq | Chi Df | Pr(>Chisq) |
|---|---|---|---|---|---|---|---|---|
| model_chap12_q2a | 4 | 7889.03 | 7908.94 | -3940.52 | 7881.03 | | | |
| model_chap12_q2b | 6 | 7884.23 | 7914.10 | -3936.12 | 7872.23 | 8.80 | 2 | 0.0123 |

Using ANOVA to check the changes between the 2 models, we find out that adding additional child-level predictors actually reduced the residual sum of squares . This reduction was statistically significant. We also had lower AIC and BIC in the second model implying that the additional regressors actually improved the model

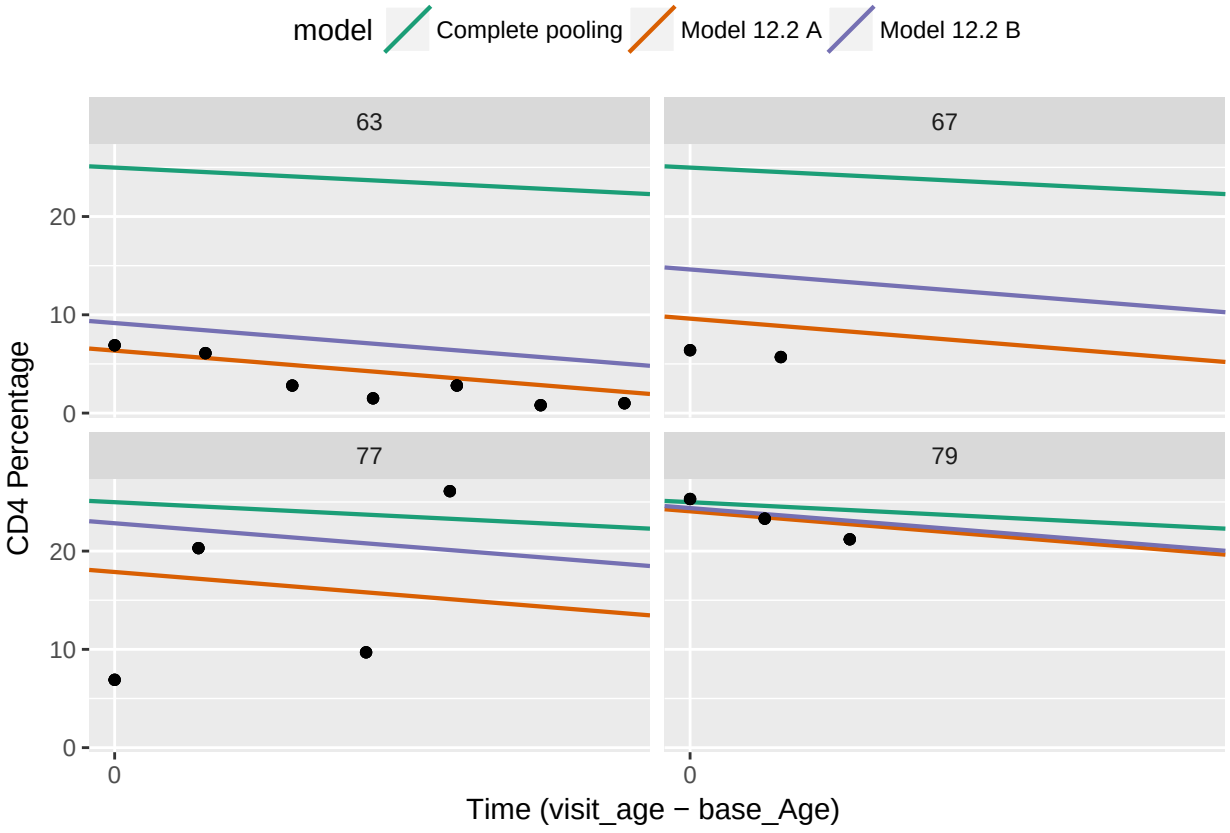



From the above plot, notice the scale of each plot. Model 12.2.A has higher random error than Model 12.2.B. This implies that adding child-level predictors reduces the variance of our random effect estimate. We noticed a substantial reduction in the variance of the intercept - 128 to 123. This is mainly because some of the variations had been explained by the addition of fixed effect predictors like base age and treatment.

## Part D:

Compare results in (b) to those obtained in part (c).

As seen in the previous question, We observed that by adding child-level predictors (i.e, group-level predictors) the model improved significantly hence increasing the precision of our estimates. The addition of the 2 fixed effect predictors like base age and treatment made the variance of the random effect estimates to reduce by 5 units essentially by compromising between group level and individual level variation. However, the residual did not reduced by much possibly because we did not add another random effect predictor that is due to child-specific idiosyncrasies e.g ARVS.



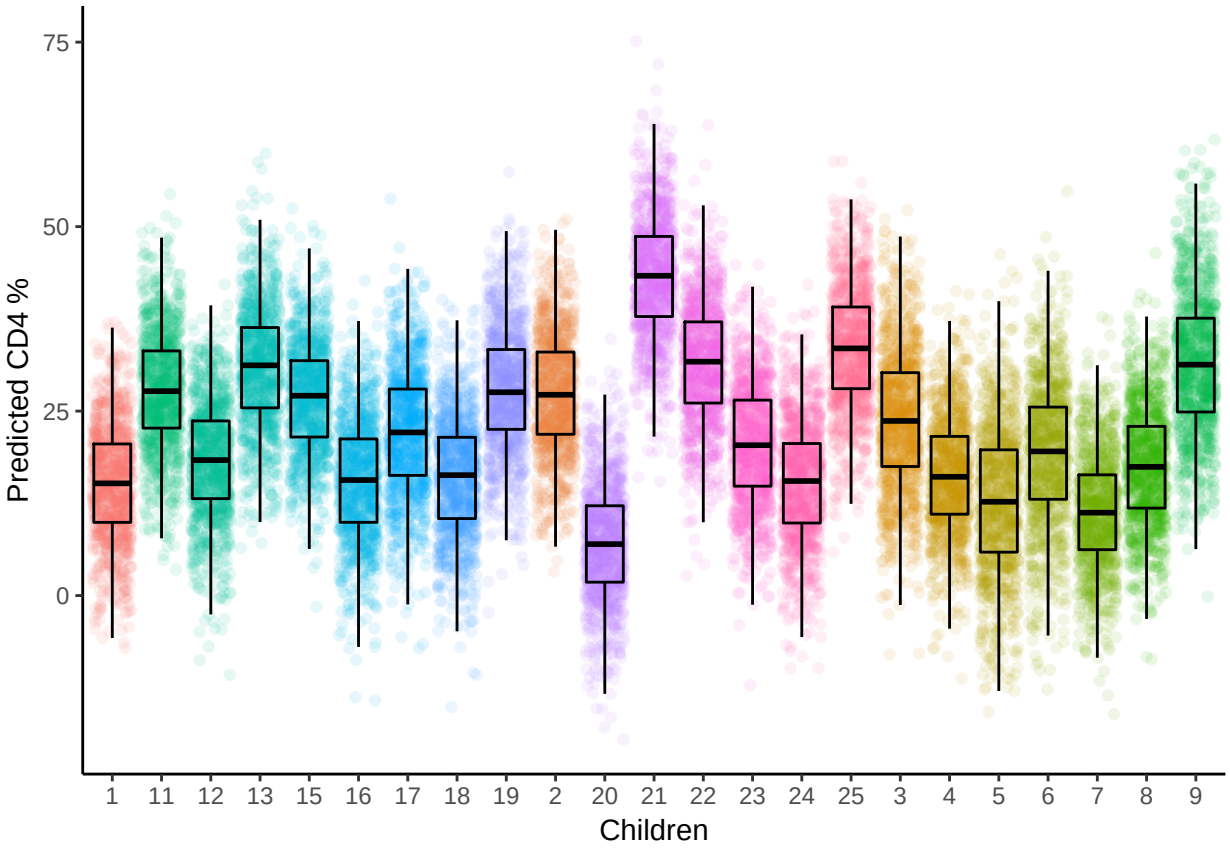

In summary, this shrinkage effect observed in model 12.2.B ensures that our model compromised between individual-level effects and group level effects at the same regularizing extreme estimates and taming children with less visits/observations as depicted by child 67,79 and 77.

# Chapter 12 Question 3:

`Predictions for new observations and new groups:`

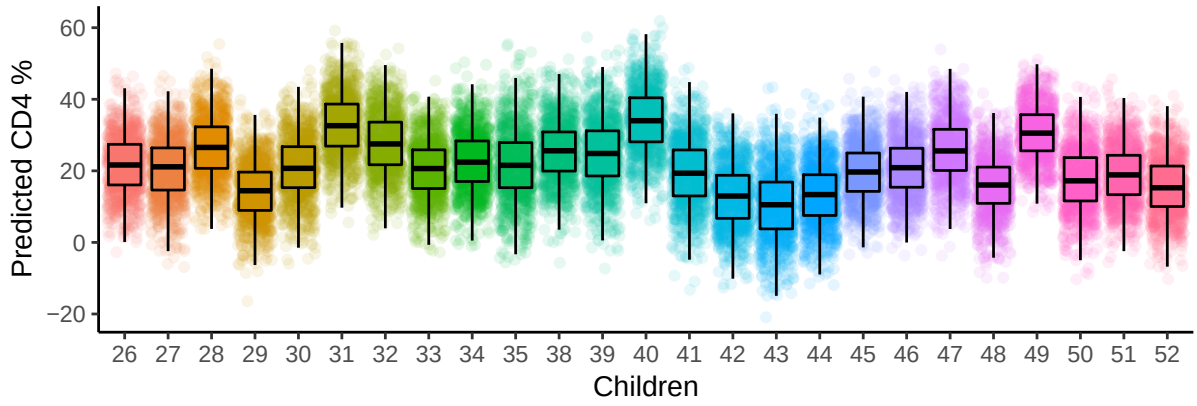
## Part A:

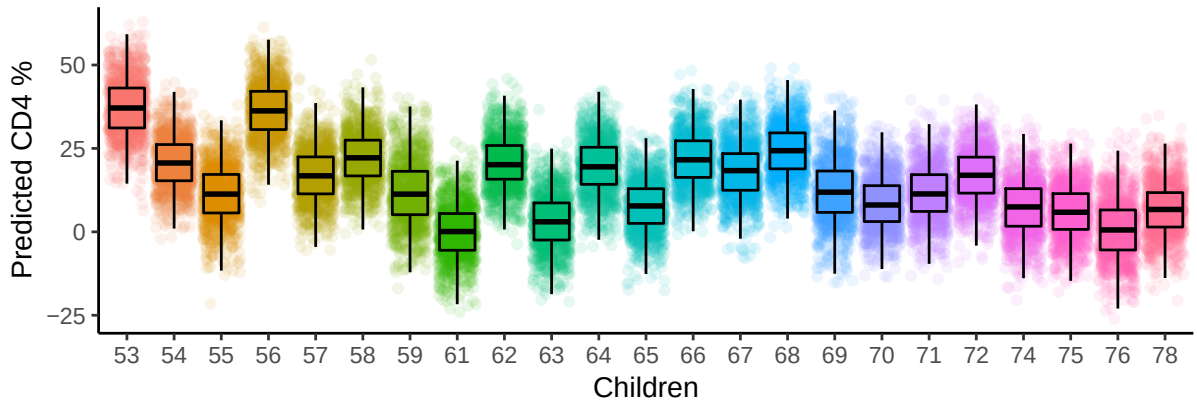
`Use the model fit from Exercise 12.2(b) to generate simulation of predicted CD4 percentages for each child in the dataset at a hypothetical next time point.`

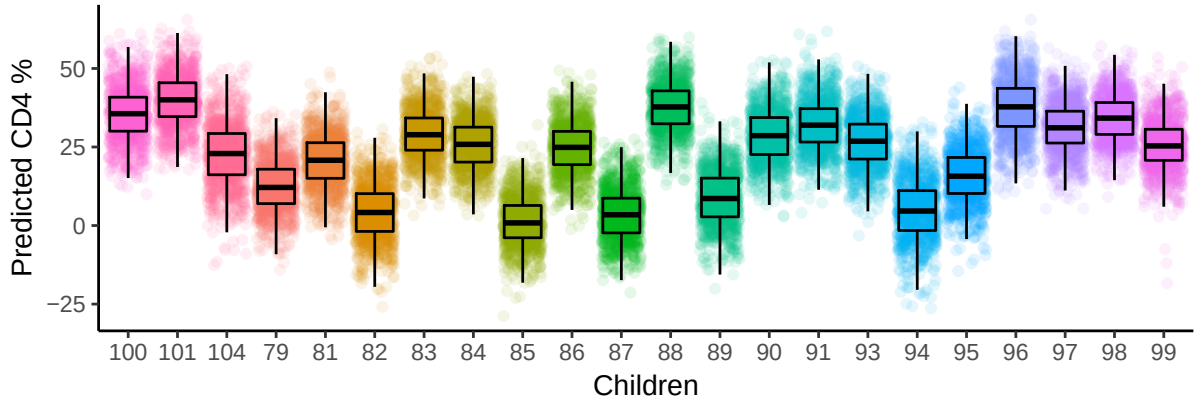


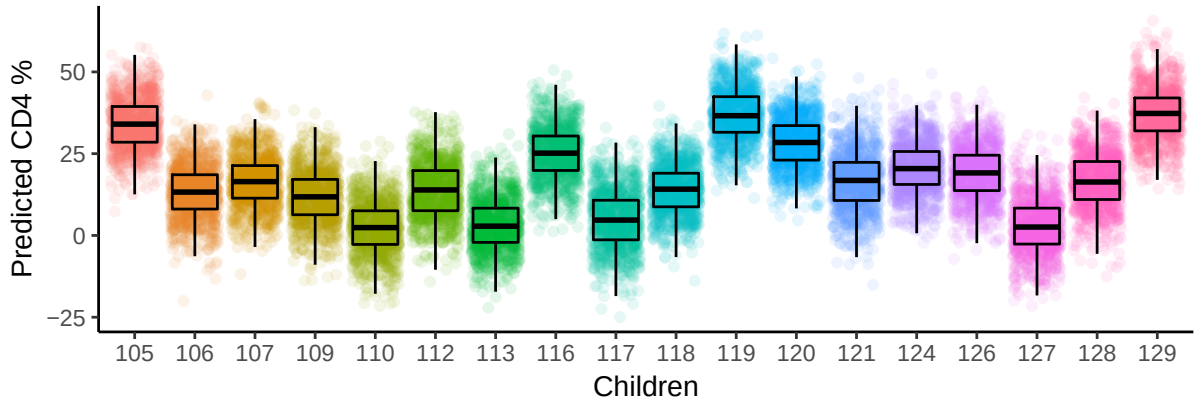

Since most appointments/visits occurred every quarterly as shown in Chapter 11 Question 4, we made our hypothetical next time point as the sum of maximum visit time for each child + 0.25 year. We then generated 1000 predictive simulation for each child. Each box plot represent each child and each geometric point represent each predicted simulated cd4% (1000). Notice the wide variation around the mean. This is because We did not use the classical predict function, instead we simulated from the distribution of regression coefficients. This in turn ensured we captured the uncertainity in regression coefficients that is necessary for replicating new data.

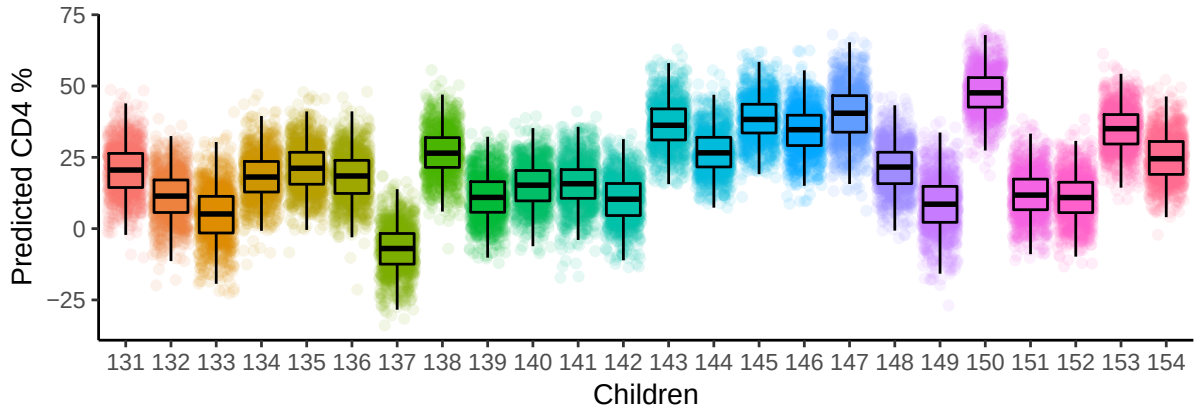
In general most children with multiple visits had much more precise prediction interval while children with less observation had less precise prediction interval. Additionally children who had visits/observations greater 1.75 years had less precise prediction interval compared to children who had observations less than 1.75. This 2 aspects are also related to the shrinkage effects discussed in the previous question.

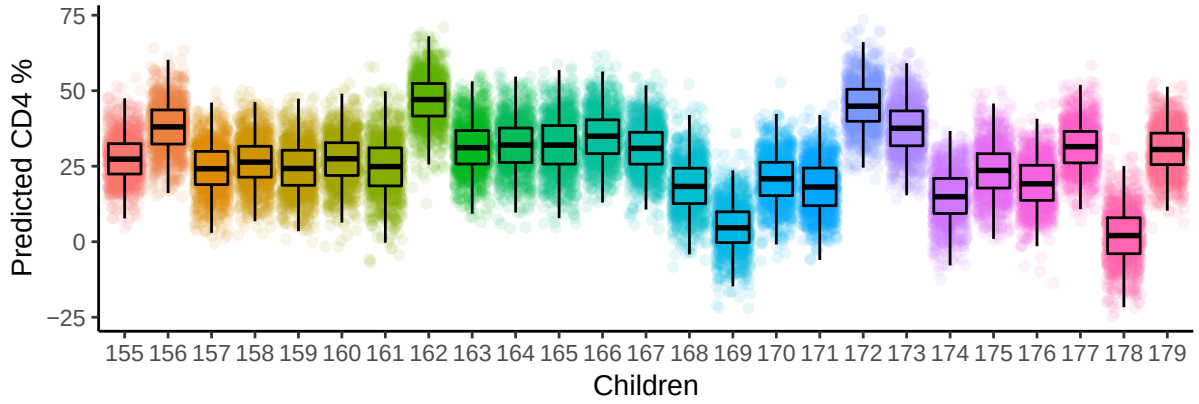

Predicted CD4 %
60
40
20
0
−20
26 27 28 29 30 31 32 33 34 35 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52
Children

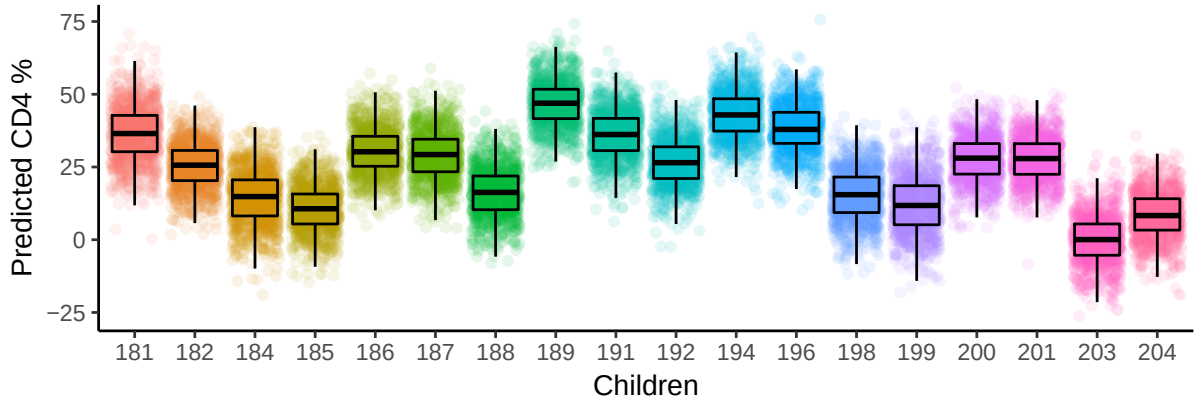

Predicted CD4 %
50
25
0
−25
53 54 55 56 57 58 59 61 62 63 64 65 66 67 68 69 70 71 72 74 75 76 78
Children

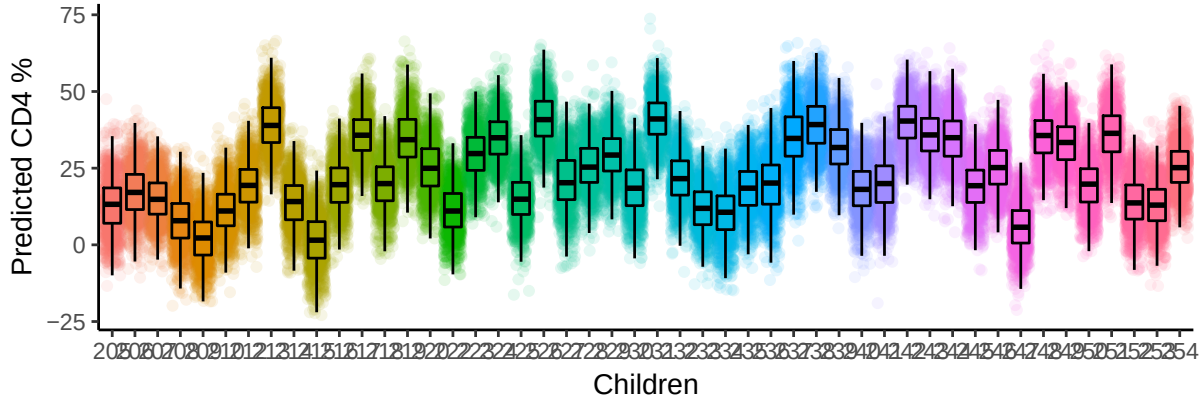

Predicted CD4 %
50
25
0
−25
100 101 104 79 81 82 83 84 85 86 87 88 89 90 91 93 94 95 96 97 98 99
Children


Predicted CD4 %
50
25
0
−25
105 106 107 109 110 112 113 116 117 118 119 120 121 124 126 127 128 129
Children


Predicted CD4 %
75
50
25
0
−25
131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154
Children

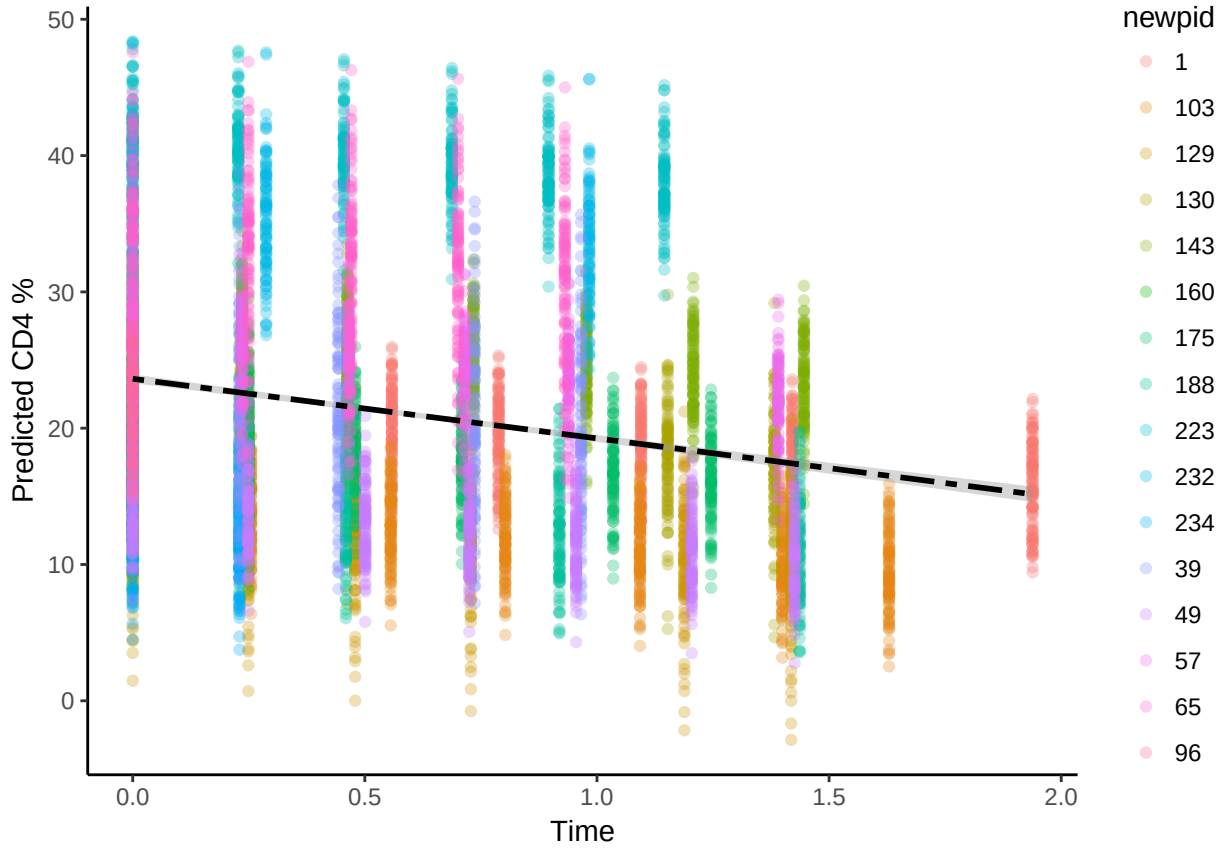

Predicted CD4 %
75
50
25
0
−25
155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179
Children


Predicted CD4 %
75
50
25
0
−25
181 182 184 185 186 187 188 189 191 192 194 196 198 199 200 201 203 204
Children

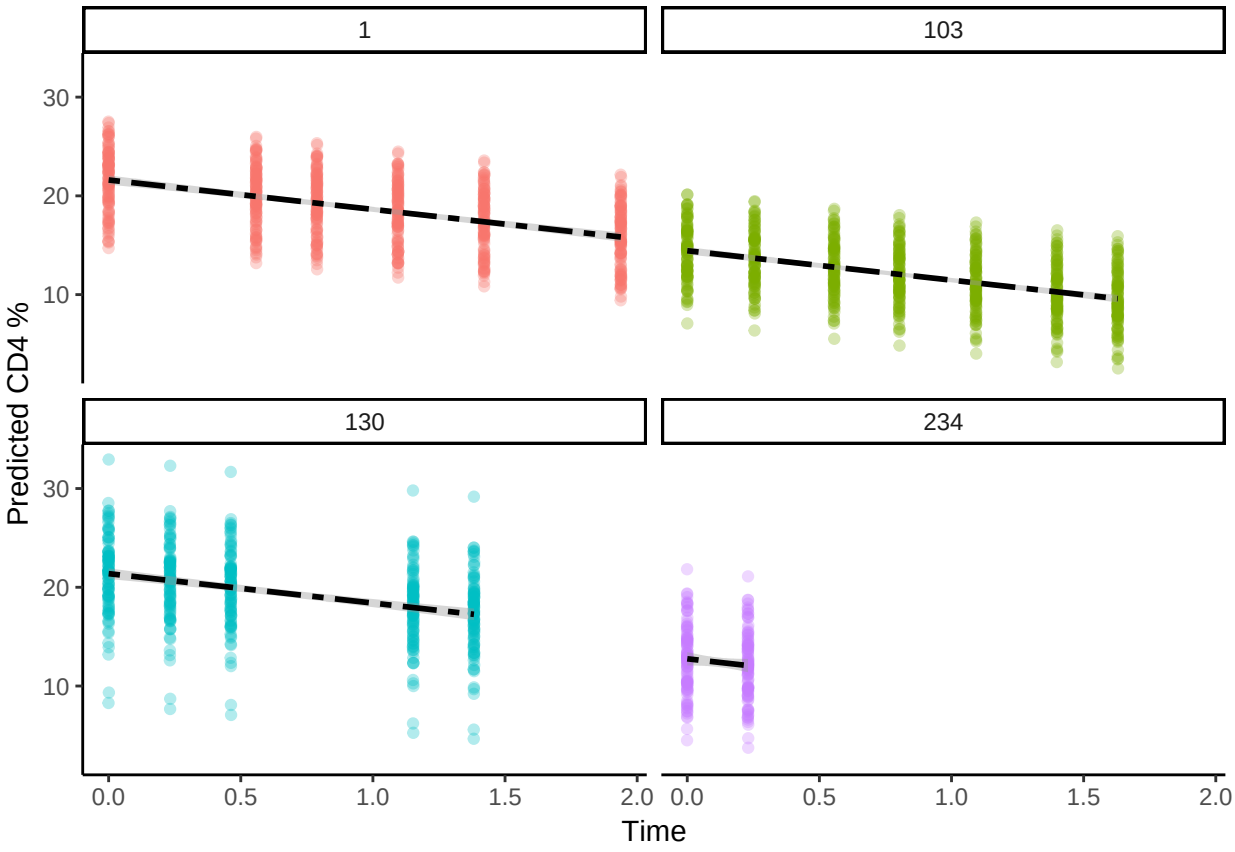

Predicted CD4 %
75
50
25
0
−25
Children

## Part B:

`Use the same model fit to generate simulations of CD4 percentages at each of the time periods for a new child who was 4 years old at baseline.`

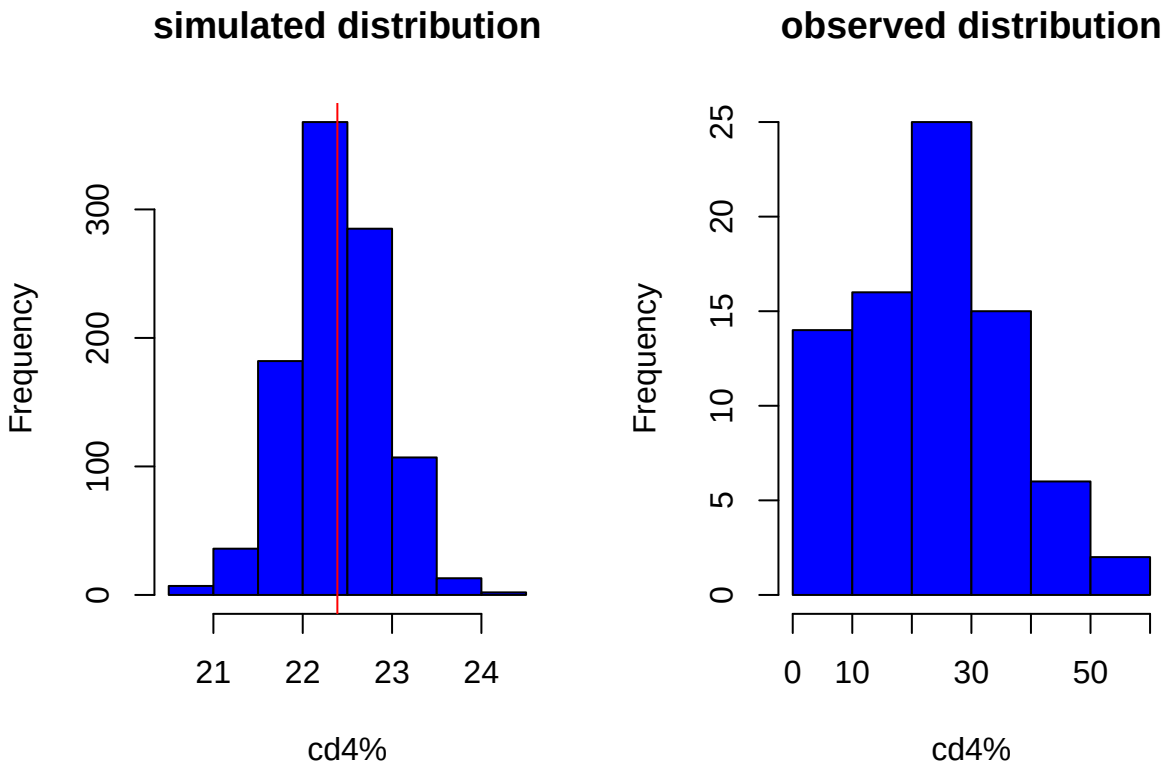



We identified 16 children who had a baseline age of 4 years or (+-.25). From the plot it is clear that children who get into the study with a baseline age of 4 years have higher cd4% decline rate. Most of their cd4 % declined by an average of 10 units


1
103
130
234
Predicted CD4 %
Time
30
20
10
0.0
0.5
1.0
1.5
2.0

## Chapter 12 Question 4:

Posterior predictive checking: continuing the previous exercise, use the fitted model from Exercise 12.2(b) to simulate a new dataset of CD4 percentages (with the same sample size and ages of the original dataset) for the final time point of the study, and record the average CD4 percentage in this sample. Repeat this process 1000 times and compare the simulated distribution to the observed CD4 percentage at the final time point for the actual data.




A few observation to make. In general we notice the shrinkage effect depicted by the model when we try to extrapolate. The simulated distribution of the predicted cd4 % ranges from 21 to 24 with a mean of about 22.5. On the other hand the observed distribution ranges from 0 to 60. Comparing to the observed distribution, simulated prediction have been pulled substantially towards (group average). It is clear that the model can be terrible at extrapolating beyond the 3rd IQR (the final time point).

In summary, this shrinkage effect observed in model 12.2.B ensures that our model compromised between individual-level effects and group level effects at the same regularizing extreme estimates and taming children with less visits/observations

## Source Code

```
#' ---
#' title: "Homework 5"
#' author: "Allan Kimaina"
#' header-includes:
#' - \usepackage{pdflscape}
#' - \newcommand{\blandscape}{\begin{landscape}}
#' - \newcommand{\elandscape}{\end{landscape}}
#' output:
#'   pdf_document: default
#'   html_document: default
#' ---
#'
#'
#knitr::opts_chunk$set(echo = F)

# loadl lm  package
library(plyr)
library(dplyr)
library("tibble")
library(car)
library(sjPlot)
library(sjmisc)
library(sjlabelled)
library(ggpubr)
library(ggpmisc)
library(gridExtra)
library(stargazer)
library(e1071)
library(jtools)
library(effects)
library(multcompView)
library(ggplot2)
library(ggrepel)
library(MASS)
library(broom)
library(ggcorrplot)
library(leaps)
library(relaimpo)
library(olsrr)

# aestetics
library(xtable)# latex table
library(texreg) # for latex table

# load GLM packages
library(ROCR)
library(arm)
library(foreign)
library(nnet)
library(VGAM)
library(ordinal)
```

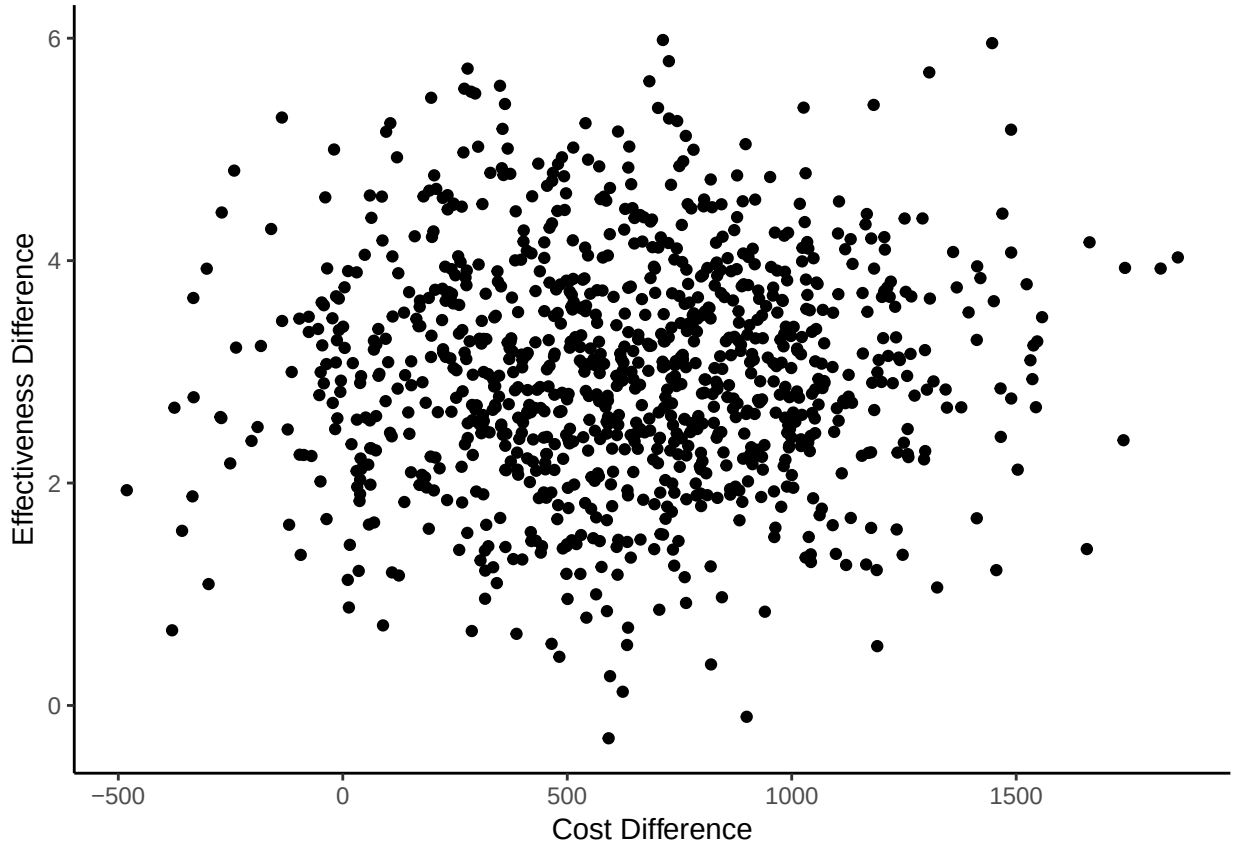
```r
library(ModelGood)
library(InformationValue)
library(rms)
library(AER) # for overdispersion
library(metRology)
library(hett)
library(lme4)




# load required data
riskyBehaviours.df=read.csv("data/risky_behaviors.csv")
# Data wrangling
riskyBehaviours.df$couples <- factor(riskyBehaviours.df$couples)
riskyBehaviours.df$women_alone <- factor(riskyBehaviours.df$women_alone)

# https://stackoverflow.com/questions/14510277/scale-back-linear-regression-coefficients-in-r-from-scal
# scaling
riskyBehaviours.df$cBupacts <- (riskyBehaviours.df$bupacts - mean(riskyBehaviours.df$bupacts)) / (2 * s

#hist(riskyBehaviours.df$fupacts)

#'
#'
#' \onecolumn
#' # Chapter 7 Question 2:
#'
#' Continuous probability simulation: the logarithms of weights (in pounds) of men in the United States
#'
#' ``What is the probability that the elevator cable breaks?``
#'
#'
set.seed(111)
men_weight <- 5.13
men_sd <- .17
prop_female <- .52
women_weight <- 4.96
women_sd <- .20
n <- 10
num_sims <- 1000
capacity <- 1750
elevatorCableBreaks <- rep(NA, num_sims)
for (j in 1:num_sims) {
  weights <- rep(NA, n)
  is_female <- rbinom(n=n, size=1, prob=prop_female)
  for (i in 1:n) {
    log_weight <- ifelse(is_female[i]==1,
                         rnorm(n=1, mean=women_weight, sd=women_sd),
                         rnorm(n=1, mean=men_weight, sd=men_sd)
                         )
    weights[i]<- exp(log_weight)
  }
```

```
  elevatorCableBreaks[j] <- ifelse(capacity<sum(weights), 1, 0)
}
sum(elevatorCableBreaks) / num_sims
```

```
## [1] 0.043
```

```
#'
#'
#'  We assumed the proportion of women to men is .52 (provided in the chapter but not in this question)
#' \onecolumn
#'
#' # Chapter 7 Question 8:
#' Inference for the ratio of parameters: a (hypothetical) study compares the costs and effectiveness o
#'
#'  * In the first part of the study, the difference in costs between treatments A and B is estimated a
#'
#'  *  In the second part of the study, the difference in effectiveness is estimated at 3.0 (on some re
#'
#'  *   For simplicity, assume that the data from the two parts of the study were collected independent
#'
#' Inference is desired for the incremental cost-effectiveness ratio: the difference between the averag
#'
#' ## Part A:
#' `` Create 1000 simulation draws of the cost difference and the effectiveness difference, and make a
#'
#'
set.seed(111)
num_sims <- 1000

# initialize with NA
cost_diff = rep(NA, num_sims)
effect_diff = rep(NA, num_sims)

n_cost = 52
n_effect = 102

for (i in 1:num_sims) {
  # cost difference
  treatment = rbinom(n = n_cost, size = 1, prob = 0.5)
  X = cbind(1, treatment)
  sigma = sqrt(400 ^ 2 / solve((t(X) %*% X))[2, 2])
  error = rnorm(n_cost, mean = 0, sd = sigma)
  cost = 100000 + 600 * treatment + error
  Cost_model = lm(cost ~ treatment)
  cost_diff[i] = coef(Cost_model)[2]

  # effect difference
  treatment = rbinom(n = n_effect, size = 1, prob = 0.5)
  X = cbind(1, treatment)
  sigma = sqrt(1 / solve((t(X) %*% X))[2, 2])
  error = rnorm(n_effect, mean = 0, sd = sigma)
  effect = 100000 + 3 * treatment + error
  effect_model = lm(effect ~ treatment)
  effect_diff[i] = coef(effect_model)[2]
```

```
}



costEffect.df <- data.frame(cost=cost_diff, effect=effect_diff)

ggplot(data=costEffect.df, aes(x=cost, y=effect)) +
      geom_point() +
      labs(x="Cost Difference", y="Effectiveness Difference") +
      theme_classic()+
      guides(fill=FALSE)
```




```
#'
#'
#'
#' ## Part B:
#' ``Use simulation to come up with an estimate, 50% interval, and 95% interval for the incremental cos
#'
#'


costEffect.df$ratio <- costEffect.df$cost/costEffect.df$effect

hist(costEffect.df$ratio,
```

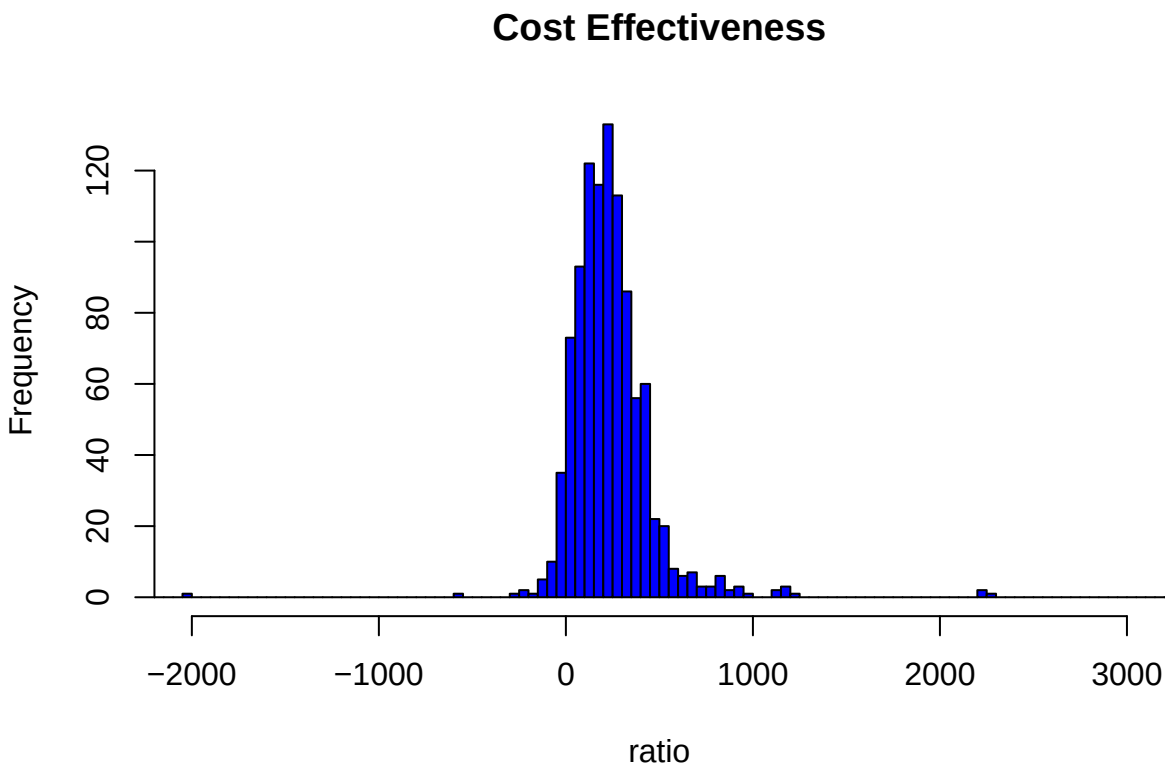
```
     col="blue",
     main = "Cost Effectiveness", xlim = c(-2000,3000), xlab="ratio",
     breaks=300)


#'
#'
#'
quantile(costEffect.df$ratio, c(.25, .75))
```

```
##      25%      75%
## 112.5502 323.3060
```

```
quantile(costEffect.df$ratio, c(.025, .975))
```

```
##      2.5%     97.5%
## -38.53523 737.84576
```

```
#stargazer(summary(costEffect.df))


#'
#'
#' ## Part C:
#' ``Repeat this problem, changing the standard error on the difference in effectiveness to 2.0.``
#'
#'
num_sims <- 1000
effect_diff = rep(NA, num_sims)
n_cost = 52
n_effect = 102

for (i in 1:num_sims) {
  # effect difference
  treatment = rbinom(n = n_effect, size = 1, prob = 0.5)
  X = cbind(1, treatment)
  sigma = sqrt(1 / solve((t(X) %*% X))[2, 2])
  error = rnorm(n_effect, mean = 0, sd = sigma)
  effect = 100000 + 2 * treatment + error
  effect_model = lm(effect ~ treatment)
  effect_diff[i] = coef(effect_model)[2]

}


costEffect.df$ratio <- costEffect.df$cost/costEffect.df$effect
hist(costEffect.df$ratio,
     col="blue",
     main = "Cost Effectiveness", xlim = c(-2000,3000), xlab="ratio",
     breaks=300)
```




```
#'
#'
#'
#'
#'
quantile(costEffect.df$ratio, c(.25, .75))
```

```
##      25%      75%
## 112.5502 323.3060
```

```
quantile(costEffect.df$ratio, c(.025, .975))
```

```
##      2.5%     97.5%
## -38.53523 737.84576
```

```
#'
#'
#'
#' \onecolumn
#'
#' # Chapter 8 Question 1:
#' Fitting the wrong model: suppose you have 100 data points that arose from the following model: y =3+
#'
#'
#' ## Part A:
#' ``Simulate data from this model. For simplicity, suppose the values of x1 are simply the integers fr
#'
#'
```

```r
set.seed(111)
x1 <- 1:100
x2 <- rbinom(100, 1, 0.5)
error <- rnorm(100, 0, 1)

y = 3 + .1*x1 + .5*x2 + error

model_8.1.a <- lm(y ~ x1 + x2)
texreg(list(model_8.1.a),
       custom.model.names = c("Model 8A"),
       single.row=TRUE,  float.pos = "h")
```

```
##
## \begin{table}[h]
## \begin{center}
## \begin{tabular}{l c }
## \hline
##  & Model 8A \\
## \hline
## (Intercept) & $3.30 \; (0.22)^{***}$ \\
## x1          & $0.10 \; (0.00)^{***}$ \\
## x2          & $0.45 \; (0.21)^{*}$   \\
## \hline
## R$^2$       & 0.89                   \\
## Adj. R$^2$  & 0.89                   \\
## Num. obs.   & 100                    \\
## RMSE        & 1.03                   \\
## \hline
## \multicolumn{2}{l}{\scriptsize{$^{***}p<0.001$, $^{**}p<0.01$, $^*p<0.05$}}
## \end{tabular}
## \caption{Statistical models}
## \label{table:coefficients}
## \end{center}
## \end{table}
```

```r
#'
#'
#'
coverage_test = c(3, 0.1, 0.5)
regression_coef = as.data.frame(summary(model_8.1.a)$coefficients)
int_coverage = cbind(regression_coef$Estimate-regression_coef$`Std. Error`,
                     regression_coef$Estimate+regression_coef$`Std. Error`)

int_coverage_test = cbind(coverage_test>=int_coverage[,1]) & (coverage_test<=int_coverage[,2])
rownames(int_coverage) <- c("Intercept","X1","X2")
rownames(int_coverage_test) <- c("Intercept","X1","X2")
test_matrix <- merge(int_coverage, int_coverage_test, by = "row.names", all = TRUE)
colnames(test_matrix) <- c("Coef","Lower","Upper","Coverage")
xtable(test_matrix, comment=FALSE)
```

```
## % latex table generated in R 3.4.3 by xtable 1.8-2 package
## % Fri Apr 27 17:00:12 2018
## \begin{table}[ht]
## \centering
```

```
## \begin{tabular}{rlrrl}
##   \hline
##  & Coef & Lower & Upper & Coverage \\
##   \hline
## 1 & Intercept & 3.08 & 3.52 & FALSE \\
##   2 & X1 & 0.09 & 0.10 & TRUE \\
##   3 & X2 & 0.24 & 0.66 & TRUE \\
##    \hline
## \end{tabular}
## \end{table}
```

```
#'
#' After generating 68% confidence intervals for the regression coefficient, all the point estimates ex
#'
#' ## Part B:
#' `` Put the above step in a loop and repeat 1000 times. Calculate the confidence coverage for the 68%
#'
#'
set.seed(111)
coefs <- array(NA, c(3, 1000))
se <- array(NA, c(3, 1000))

for (i in 1:ncol(coefs)) {
  x1 <- 1:100
  x2 <- rbinom(100, 1, 0.5)
  error <-rnorm(100, 0, 1)

  y = 3 + 0.1*x1 + 0.5*x2 + error

  lm.model <- summary(lm(y ~ x1 + x2))
  coefs[1,i] <- tidy(lm.model)[1,2]
  coefs[2,i] <- tidy(lm.model)[2,2]
  coefs[3,i] <- tidy(lm.model)[3,2]

  se[1,i] <- tidy(lm.model)[1,3]
  se[2,i] <- tidy(lm.model)[2,3]
  se[3,i] <- tidy(lm.model)[3,3]
}

mean_coef <- rowMeans(coefs)
mean_se <- rowMeans(se)

int_coverage<- cbind(mean_coef + (-1 * mean_se),
                     mean_coef + (1 * mean_se))

int_coverage_test = cbind(coverage_test>=int_coverage[,1]) & (coverage_test<=int_coverage[,2])
rownames(int_coverage) <- c("Intercept","X1","X2")
rownames(int_coverage_test) <- c("Intercept","X1","X2")
test_matrix <- merge(int_coverage, int_coverage_test, by = "row.names", all = TRUE)
colnames(test_matrix) <- c("Coef","Lower","Upper","Coverage")
xtable(test_matrix, comment=FALSE)
```

```
## % latex table generated in R 3.4.3 by xtable 1.8-2 package
## % Fri Apr 27 17:00:16 2018
## \begin{table}[ht]
```

```
## \centering
## \begin{tabular}{rlrrl}
##   \hline
##  & Coef & Lower & Upper & Coverage \\
##   \hline
## 1 & Intercept & 2.77 & 3.23 & TRUE \\
##   2 & X1 & 0.10 & 0.10 & TRUE \\
##   3 & X2 & 0.29 & 0.70 & TRUE \\
##    \hline
## \end{tabular}
## \end{table}
```

```
#'
#'
#' After simulating the previous step 1000 times, we calculated the mean of all the point estimates and
#'
#' ## Part C:
#' ``Repeat this simulation, but instead fit the model using t errors (see Exercise 6.6).``
#'
#'
set.seed(111)
coefs <- array(NA, c(3, 1000))
se <- array(NA, c(3, 1000))

for (i in 1:ncol(coefs)) {
  x1 <- 1:100
  x2 <- rbinom(100, 1, 0.5)
  error <- rt.scaled(100, df = 4, mean = 0, sd = 5)
  y = 3 + 0.1*x1 + 0.5*x2 + error

  lm.model <-  summary(tlm(y ~ x1 + x2))

  coefs[1,i] <- lm.model$loc.summary$coefficients[1,1]
  coefs[2,i] <- lm.model$loc.summary$coefficients[2,1]
  coefs[3,i] <- lm.model$loc.summary$coefficients[3,1]

  se[1,i] <- lm.model$loc.summary$coefficients[1,2]
  se[2,i] <- lm.model$loc.summary$coefficients[2,2]
  se[3,i] <- lm.model$loc.summary$coefficients[3,2]
}

mean_coef <- rowMeans(coefs)
mean_se <- rowMeans(se)

int_coverage<- cbind(mean_coef + (-1 * mean_se),
                     mean_coef + (1 * mean_se))

int_coverage_test = cbind(coverage_test>=int_coverage[,1]) & (coverage_test<=int_coverage[,2])
rownames(int_coverage) <- c("Intercept","X1","X2")
rownames(int_coverage_test) <- c("Intercept","X1","X2")
test_matrix <- merge(int_coverage, int_coverage_test, by = "row.names", all = TRUE)
colnames(test_matrix) <- c("Coef","Lower","Upper","Coverage")
xtable(test_matrix, comment=FALSE)
```

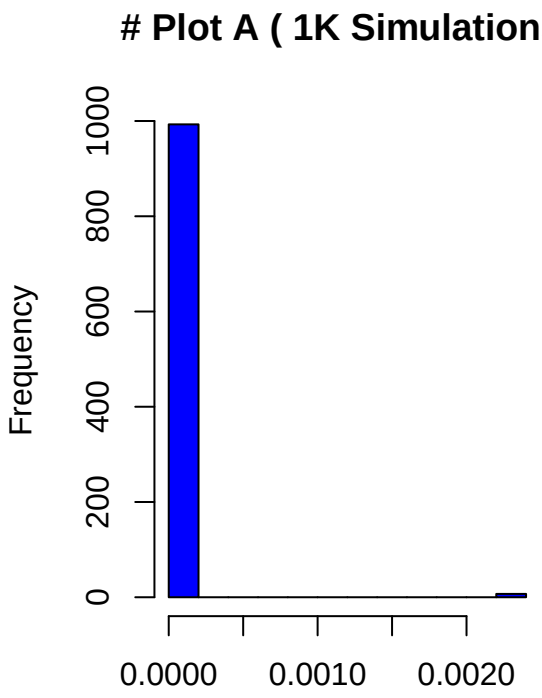
```
## % latex table generated in R 3.4.3 by xtable 1.8-2 package
```

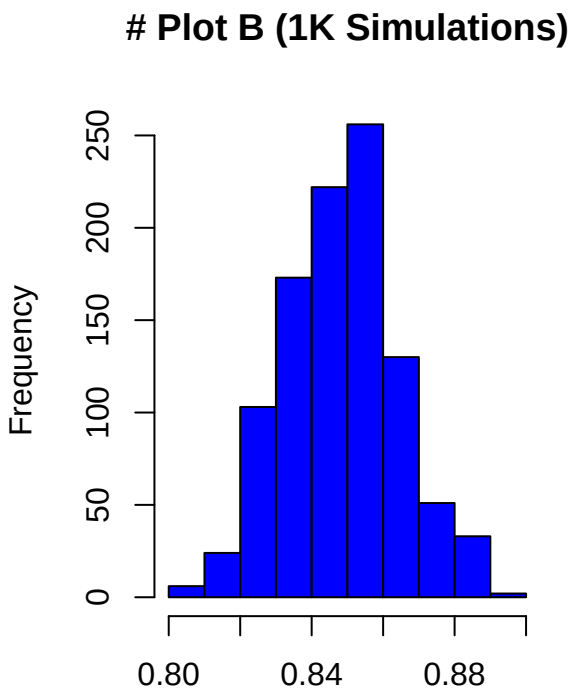
```
## % Fri Apr 27 17:00:21 2018
## \begin{table}[ht]
## \centering
## \begin{tabular}{rlrrl}
##   \hline
##  & Coef & Lower & Upper & Coverage \\
##   \hline
## 1 & Intercept & 1.65 & 4.21 & TRUE \\
##   2 & X1 & 0.08 & 0.12 & TRUE \\
##   3 & X2 & -0.61 & 1.68 & TRUE \\
##    \hline
## \end{tabular}
## \end{table}
```

```
#'
#' After  the previous step 1000 times using t errors, we calculated the mean of all the point estimate
#' However the standard error were inflated making the CI band  too wide compared to the previous model
#' \onecolumn
#'
#' # Chapter 8 Question 4:
#' Model checking for count data: the folder risky.behavior contains data from a study of behavior of c
#'
#' ## Part A:
#' ``Fit a Poisson regression model predicting number of unprotected sex acts from baseline hIV status.
#'
#' ### Fit a Poisson regression model
#'
riskyBehaviours.glm1 <- glm(fupacts ~ bs_hiv, family=poisson, data=riskyBehaviours.df)
texreg(list(riskyBehaviours.glm1),
       custom.model.names = c("Model 1: Poisson"),
       single.row=TRUE,  float.pos = "h")
```

```
##
## \begin{table}[h]
## \begin{center}
## \begin{tabular}{l c }
## \hline
##  & Model 1: Poisson \\
## \hline
## (Intercept)     & $2.91 \; (0.01)^{***}$  \\
## bs\_hivpositive & $-0.62 \; (0.03)^{***}$ \\
## \hline
## AIC             &                         \\
## BIC             &                         \\
## Log Likelihood  &                         \\
## Deviance        & 12938.74                \\
## Num. obs.       & 434                     \\
## \hline
## \multicolumn{2}{l}{\scriptsize{$^{***}p<0.001$, $^{**}p<0.01$, $^*p<0.05$}}
## \end{tabular}
## \caption{Statistical models}
## \label{table:coefficients}
## \end{center}
## \end{table}
```

```r
#'
#' This model does not fit well, the rule of thumb dictates that the ratio of residual deviance to degr
#'
#' ### 1000 Simulations
#'
# data wrangling and cleaning
riskyBehaviours.df$bs_hiv_bin <- ifelse(riskyBehaviours.df$bs_hiv=="negative",0,1)
X = cbind(1,as.numeric(riskyBehaviours.df$bs_hiv_bin))

# simulate 1000
n.sims <- 1000
riskyBehaviours.sims1 <- arm::sim(riskyBehaviours.glm1, n.sims)
n<- length(riskyBehaviours.df$fupacts)
y.rep <- array(NA, c(n.sims,n))
beta <- coef(riskyBehaviours.sims1)

# do 1000 simulations
for(i in 1:n.sims){
  y.hat <- exp(X%*%beta[i,])
  y.rep[i,]<-rpois(n,y.hat)
}

# test statistics
test.rep <- rep(NA, n.sims)
test.rep.gt10 <- rep(NA, n.sims)
for (i in 1:n.sims){
  test.rep[i]<- mean(y.rep[i,]==0)
  test.rep.gt10[i]<-mean(y.rep[i,]>10)
}

#summary(rowSums(y.hat == 0)/ncol(y.hat))
#summary(rowSums(y.hat > 10)/ncol(y.hat))

real.gt.0= mean(riskyBehaviours.df$fupacts == 0)
real.gt.10= mean(riskyBehaviours.df$fupacts > 10)

# summary(test.rep)
# summary(test.rep.gt10)
par(mfrow = c(1, 2))
hist(test.rep, main="# Plot A ( 1K Simulations)", xlab="proportion of sex acts = 0",  col="blue")
hist(test.rep.gt10, main="# Plot B (1K Simulations)", xlab="proportion of of sex acts > 10",  col="blue
```



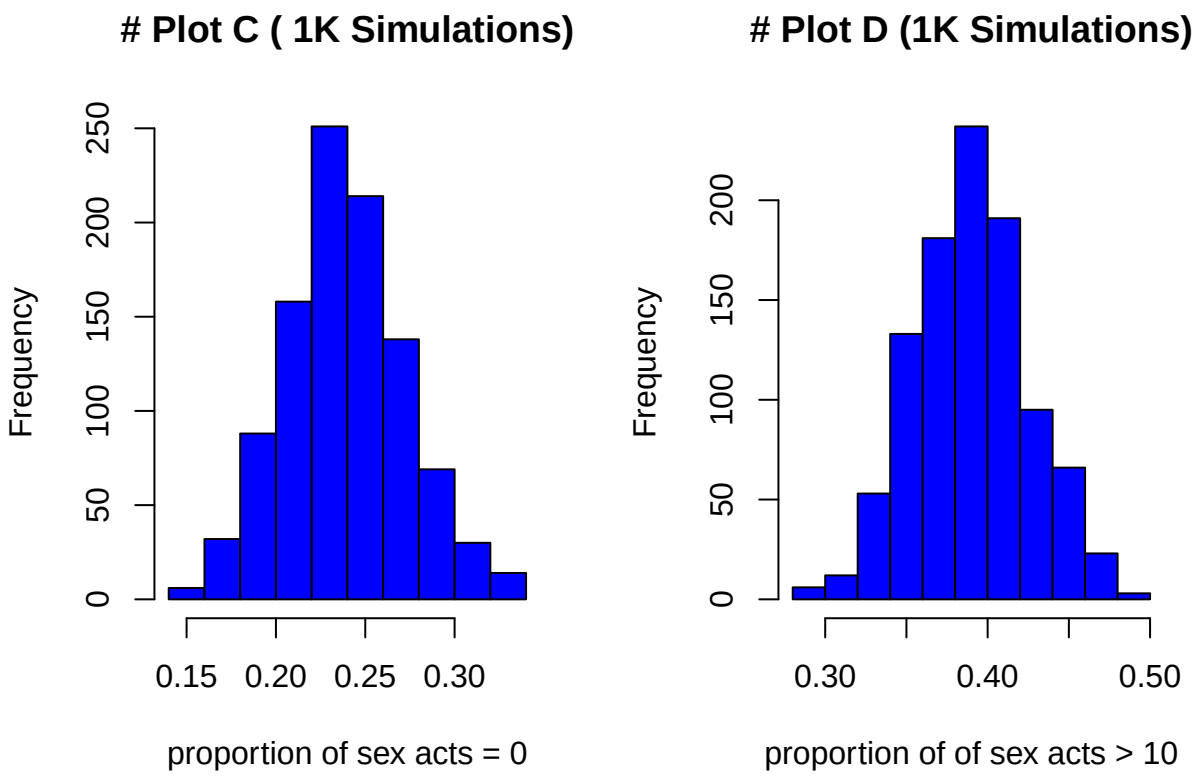

```
#'
#'
#' ### Plot A:  Frequency of number of unprotected sex acts at followup = 0
#'
#' ```
#' ##     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
#' ## 0.000e+00 0.000e+00 0.000e+00 1.843e-05 0.000e+00 2.304e-03
#' ```
#'
#' The actual dataset had 29% of the observation having  0 number of unprotected sex acts at followup.
#'
#' ### Plot B:  Frequency of  number of unprotected sex acts at followup > 10
#'
#' ```
#' ##  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
#' ## 0.7995  0.8364  0.8479  0.8485  0.8594  0.9101
#' ```
#'
#' The actual dataset had 36% of the observations having number of unprotected sex acts at followup tha
#'
#' ## Part B:
#' ``Repeat (a) using an overdispersed Poisson regression model.``
#'
#' ### Fit an overdispersed Poisson regression model
#'
riskyBehaviours.glm2 <- glm(fupacts ~ bs_hiv, family=quasipoisson, data=riskyBehaviours.df)
```

```
texreg(list(riskyBehaviours.glm2),
       custom.model.names = c("Model 2: overdispersed Poisson"),
       single.row=TRUE,  float.pos = "h")
```

```
##
## \begin{table}[h]
## \begin{center}
## \begin{tabular}{l c }
## \hline
##  & Model 2: overdispersed Poisson \\
## \hline
## (Intercept)     & $2.91 \; (0.08)^{***}$ \\
## bs\_hivpositive & $-0.62 \; (0.23)^{**}$ \\
## \hline
## AIC             &                        \\
## BIC             &                        \\
## Log Likelihood  &                        \\
## Deviance        & 12938.74               \\
## Num. obs.       & 434                    \\
## \hline
## \multicolumn{2}{l}{\scriptsize{$^{***}p<0.001$, $^{**}p<0.01$, $^*p<0.05$}}
## \end{tabular}
## \caption{Statistical models}
## \label{table:coefficients}
## \end{center}
## \end{table}
```

```
#'
#' Much better model but it does not fit as well, the ratio of residual deviance to degree of freedom i
#'
#' ### 1000 Simulations
#'

# simulate 1000
n.sims <- 1000
riskyBehaviours.sims2 <- arm::sim(riskyBehaviours.glm2, n.sims)
n<- length(riskyBehaviours.df$fupacts)
y.rep <- array(NA, c(n.sims,n))
beta <- coef(riskyBehaviours.sims2)

# do 1000 simulations
overdisp <- summary(riskyBehaviours.glm2)$dispersion
for(i in 1:n.sims){
  y.hat <- exp(X%*%beta[i,])
  a <- y.hat/(overdisp-1) # dispersion param
  y.rep[i,]<-rnegbin(n,y.hat, a)
}

# test statistics
test.rep <- rep(NA, n.sims)
test.rep.gt10 <- rep(NA, n.sims)
for (i in 1:n.sims){
  test.rep[i]<- mean(y.rep[i,]==0)
  test.rep.gt10[i]<-mean(y.rep[i,]>10)
```

```
}

#summary(rowSums(y.hat == 0)/ncol(y.hat))
#summary(rowSums(y.hat > 10)/ncol(y.hat))

real.gt.0= mean(riskyBehaviours.df$fupacts == 0)
real.gt.10= mean(riskyBehaviours.df$fupacts > 10)

# summary(test.rep)
# summary(test.rep.gt10)
par(mfrow = c(1, 2))
hist(test.rep, main="# Plot C ( 1K Simulations)", xlab="proportion of sex acts = 0",  col="blue")
hist(test.rep.gt10, main="# Plot D (1K Simulations)", xlab="proportion of of sex acts > 10",  col="blue
```



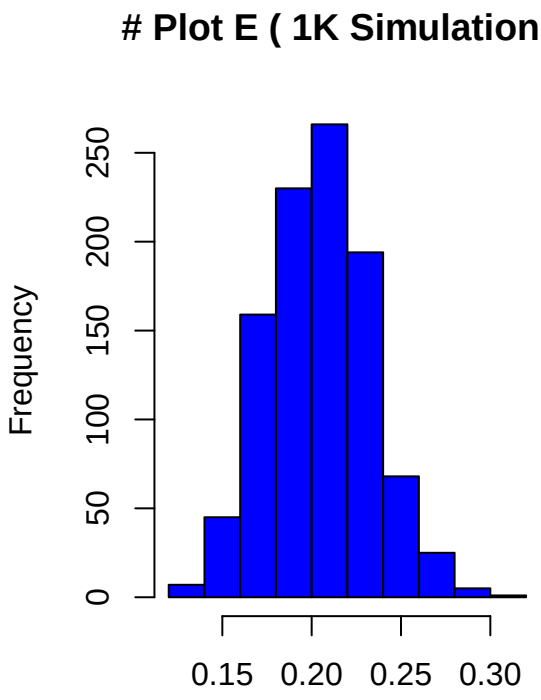

```
#'
#'
#' ### Plot C: Frequency of number of unprotected sex acts at followup = 0
#'
#' ```
#' ##  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
#' ## 0.1290  0.2166  0.2373  0.2388  0.2604  0.3364
#' ```
#'
#' After performing predictive simulation based on overdispersed model, we found out that the percentag
#'
#' ### Plot D: Frequency of  number of unprotected sex acts at followup > 10
```

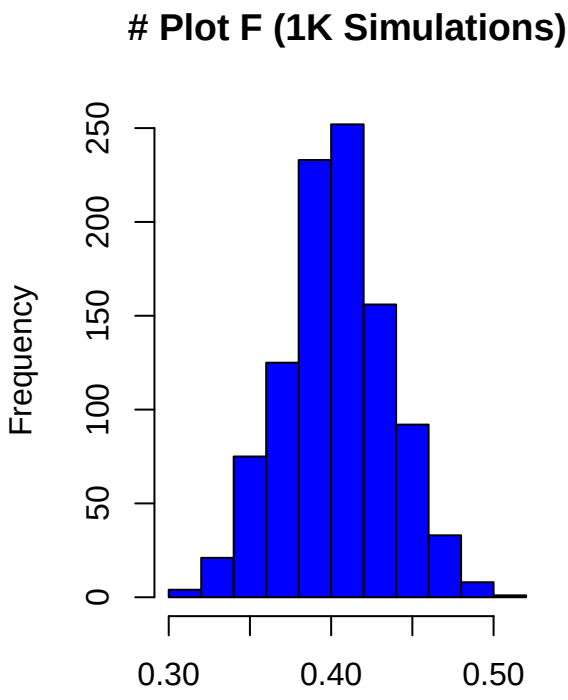
```
#'
#' ```
#' ##  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
#' ## 0.2834  0.3664  0.3871  0.3881  0.4124  0.4862
#' ```
#'
#' The actual dataset had 36% of the observations having number of unprotected sex acts at followup tha
#'
#'
#' ## Part C:
#' ``Repeat (b), also including ethnicity and baseline number of unprotected sex acts as input variable
#'
#' ### Fit an overdispersed Poisson regression model
#'
riskyBehaviours.glm3 <- glm(fupacts ~ bs_hiv + bupacts, family=quasipoisson, data=riskyBehaviours.df)
texreg(list(riskyBehaviours.glm3),
       custom.model.names = c("Model 3: overdispersed Poisson"),
       single.row=TRUE,  float.pos = "h")
```

```
##
## \begin{table}[h]
## \begin{center}
## \begin{tabular}{l c }
## \hline
##  & Model 3: overdispersed Poisson \\
## \hline
## (Intercept)     & $2.54 \; (0.09)^{***}$ \\
## bs\_hivpositive & $-0.50 \; (0.20)^{*}$  \\
## bupacts         & $0.01 \; (0.00)^{***}$ \\
## \hline
## AIC             &                        \\
## BIC             &                        \\
## Log Likelihood  &                        \\
## Deviance        & 10707.81               \\
## Num. obs.       & 434                    \\
## \hline
## \multicolumn{2}{l}{\scriptsize{$^{***}p<0.001$, $^{**}p<0.01$, $^*p<0.05$}}
## \end{tabular}
## \caption{Statistical models}
## \label{table:coefficients}
## \end{center}
## \end{table}
```

```
#'
#' Much  better model than the previous model but it does not meet the standard, the ratio of residual
#'
#' ### 1000 Simulations
#'

# simulate 1000
n.sims <- 1000
 riskyBehaviours.sims3<- arm::sim(riskyBehaviours.glm3, n.sims)
n<- length(riskyBehaviours.df$fupacts)
y.rep <- array(NA, c(n.sims,n))
```

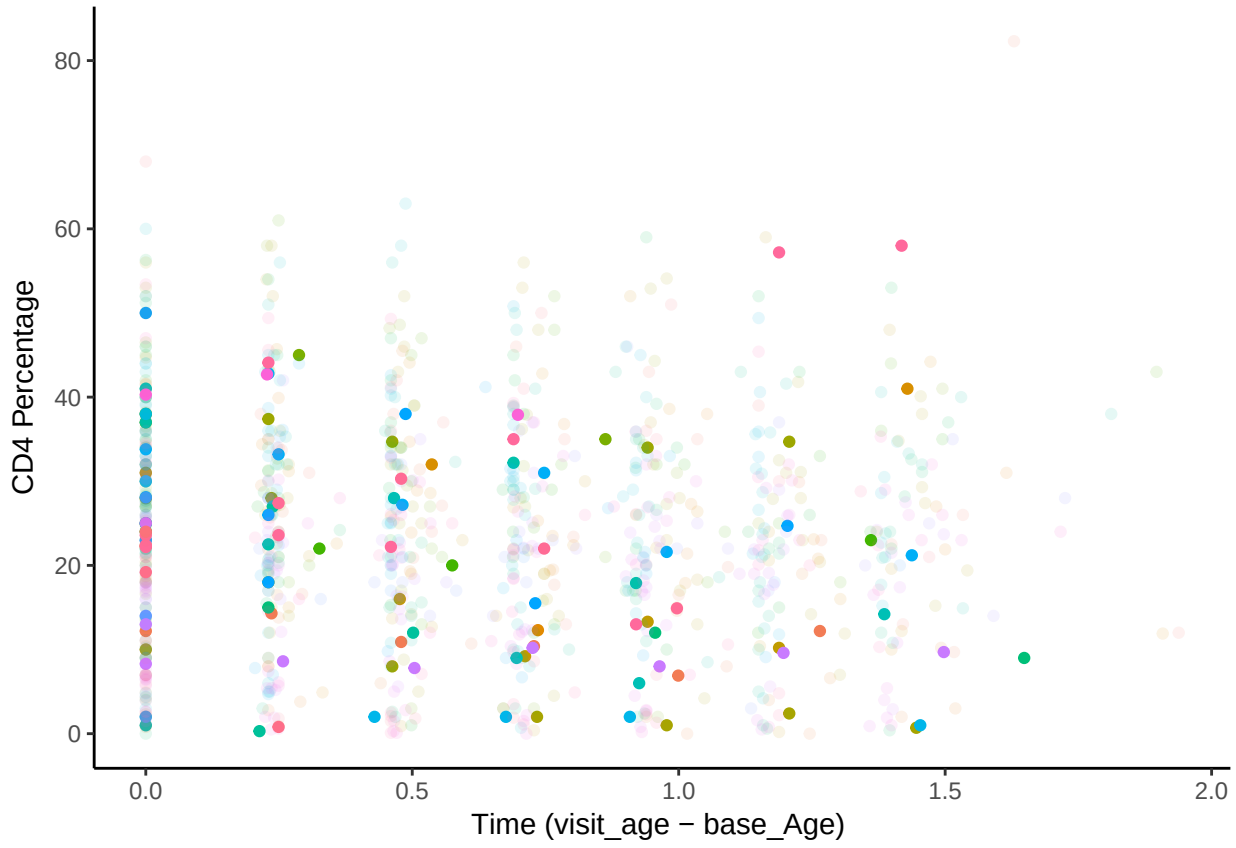
```r
beta <- coef(riskyBehaviours.sims3)
X = cbind(1,as.numeric(riskyBehaviours.df$bs_hiv_bin),riskyBehaviours.df$bupacts)

# do 1000 simulations
overdisp <- summary(riskyBehaviours.glm3)$dispersion
for(i in 1:n.sims){
  y.hat <- exp(X%*%beta[i,])
  a <- y.hat/(overdisp-1) # dispersion param
  y.rep[i,]<-rnegbin(n,y.hat, a)
}

# test statistics
test.rep <- rep(NA, n.sims)
test.rep.gt10 <- rep(NA, n.sims)
for (i in 1:n.sims){
  test.rep[i]<- mean(y.rep[i,]==0)
  test.rep.gt10[i]<-mean(y.rep[i,]>10)
}

#summary(rowSums(y.hat == 0)/ncol(y.hat))
#summary(rowSums(y.hat > 10)/ncol(y.hat))

real.gt.0= mean(riskyBehaviours.df$fupacts == 0)
real.gt.10= mean(riskyBehaviours.df$fupacts > 10)

# summary(test.rep)
# summary(test.rep.gt10)
par(mfrow = c(1, 2))
hist(test.rep, main="# Plot E ( 1K Simulations)", xlab="proportion of sex acts = 0",  col="blue")
hist(test.rep.gt10, main="# Plot F (1K Simulations)", xlab="proportion of of sex acts > 10",  col="blue
```



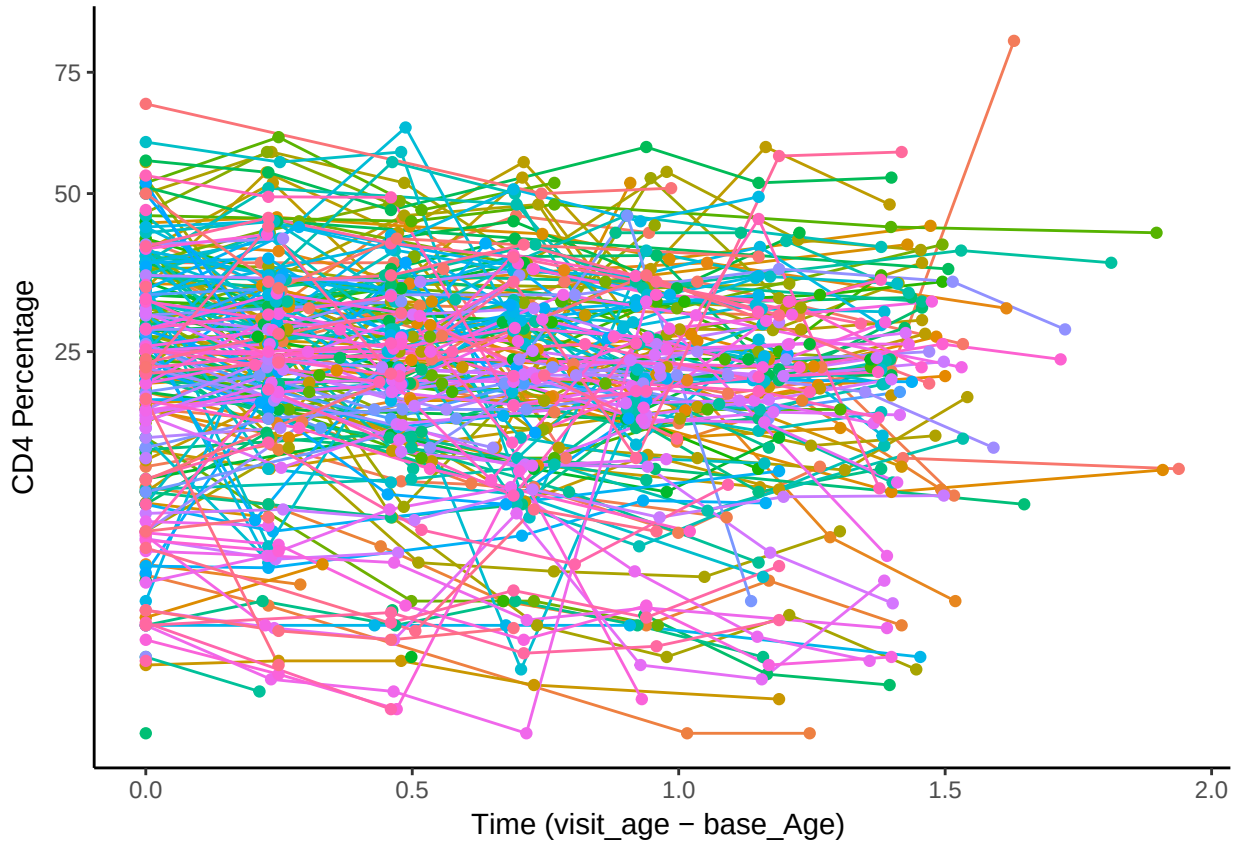

```
#'
#'
#' ### Plot E: Frequency of number of unprotected sex acts at followup = 0
#'
#' ```
#' ##    Min.  1st Qu. Median  Mean    3rd Qu. Max
#' ##  0.1198  0.1843  0.2028  0.2041  0.2235  0.2857
#' ```
#'
#' After performing predictive simulation based on overdispersed model with added predictor (bupacts),
#'
#' ### Plot F: Frequency of  number of unprotected sex acts at followup > 10
#'
#' ```
#' ##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
#' ## 0.3111  0.3802  0.4032  0.4033  0.4263  0.4908
#' ```
#'
#' The actual dataset had 36% of the observations having number of unprotected sex acts at followup tha
#'
#'
#' In summary this model really does well in compromising and balancing between the 2 aspect of data we
#'
#'
#' \onecolumn
#'
```

```r
#' # Chapter 11 Question 4:
#' The folder cd4 has CD4 percentages for a set of young children with HIV who were measured several ti
#'
#' ### Time Function
#'
# load required data
cd4.df=read.csv("data/cd4.csv")
cd4.df$newpid= as.character(cd4.df$newpid)
cd4.df$treatmnt= as.factor(cd4.df$treatmnt)

# transfrom data
time<-cd4.df$visage-cd4.df$baseage
cd4.df<-cbind(cd4.df,time)

# define ggplot alpha level
cd4.df = ddply(cd4.df, .(newpid), function(x){
  x$alpha = ifelse(runif(n = 1) > 0.9, 1, 0.1)
  x$is_test = factor(ifelse(x$newpid<5, 0, 1))
  x
})

# define timne function
time_function_plot <- ggplot(cd4.df, aes(x = time, y = cd4pct) )+
                    geom_point()+
                    labs(y = "CD4 Percentage", x="Time (visit_age - base_Age) ")+
                    aes(alpha=alpha, color=factor(newpid)) +
                    theme_classic()+
                    theme(legend.position="none")
time_function_plot
```



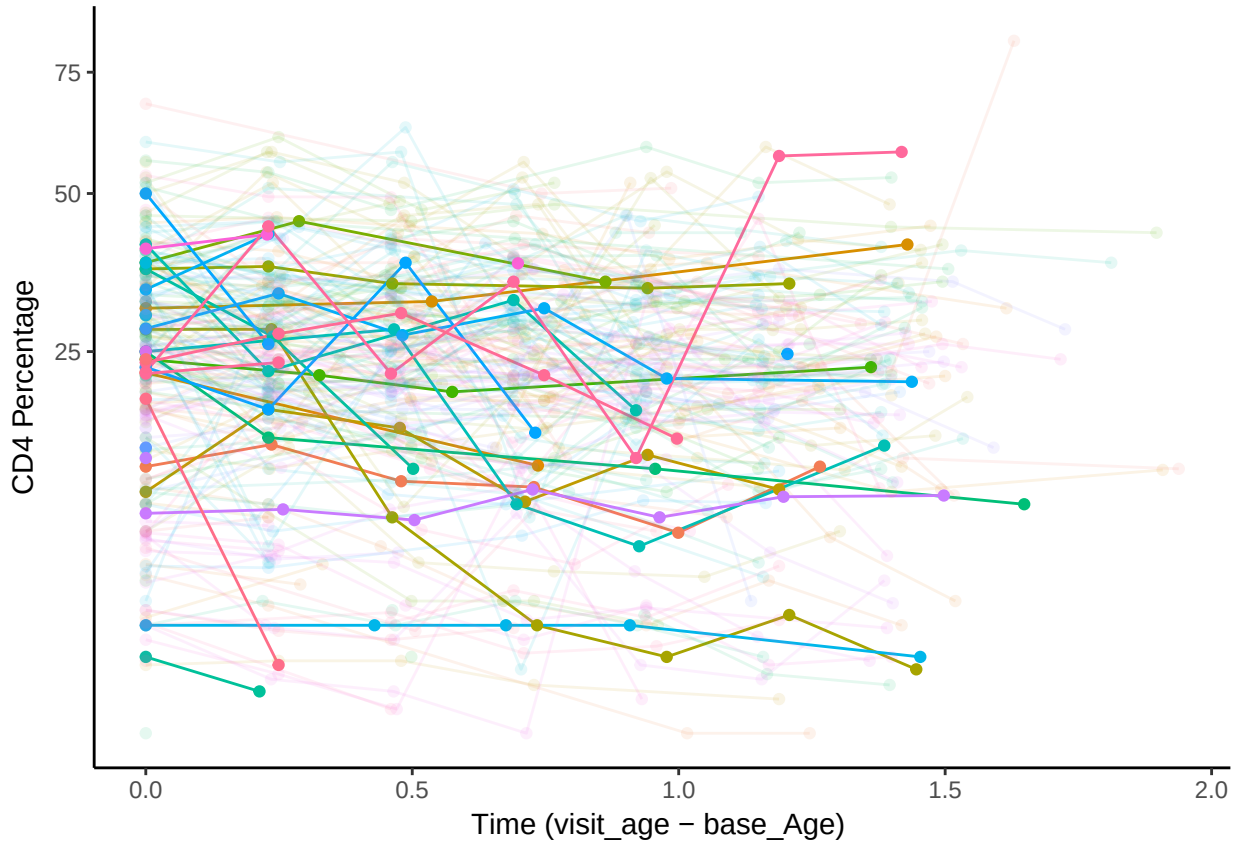

```
#'
#'
#'
#' We define our time function as visit_age - base_Age. This ensured that all children visit started fr
#'
#' It is also interesting to note that most visit occurred quarterly
#'
#'
#' ## Part A:
#' ``Graph the outcome (the CD4 percentage, on the square root scale) for each child as a function of t
#'
#'
#'
#' ### Overall Plot - all 254 children
#'


cd4_c11_q4.plot<- ggplot(cd4.df, aes(x = time, y = cd4pct)) +
    guides(colour=FALSE) +
    labs(y = "CD4 Percentage", x="Time (visit_age - base_Age) ")+
    theme_classic()+ guides(fill=FALSE)+
    theme(legend.position="none") +
    scale_y_sqrt()
    #geom_smooth(se=FALSE, colour="black", method = "lm", size=2)
    #facet_wrap(~person_id)
```

```
cd4_c11_q4.plot +
      geom_line()+
  geom_point()+
aes(color  = factor(newpid))
```

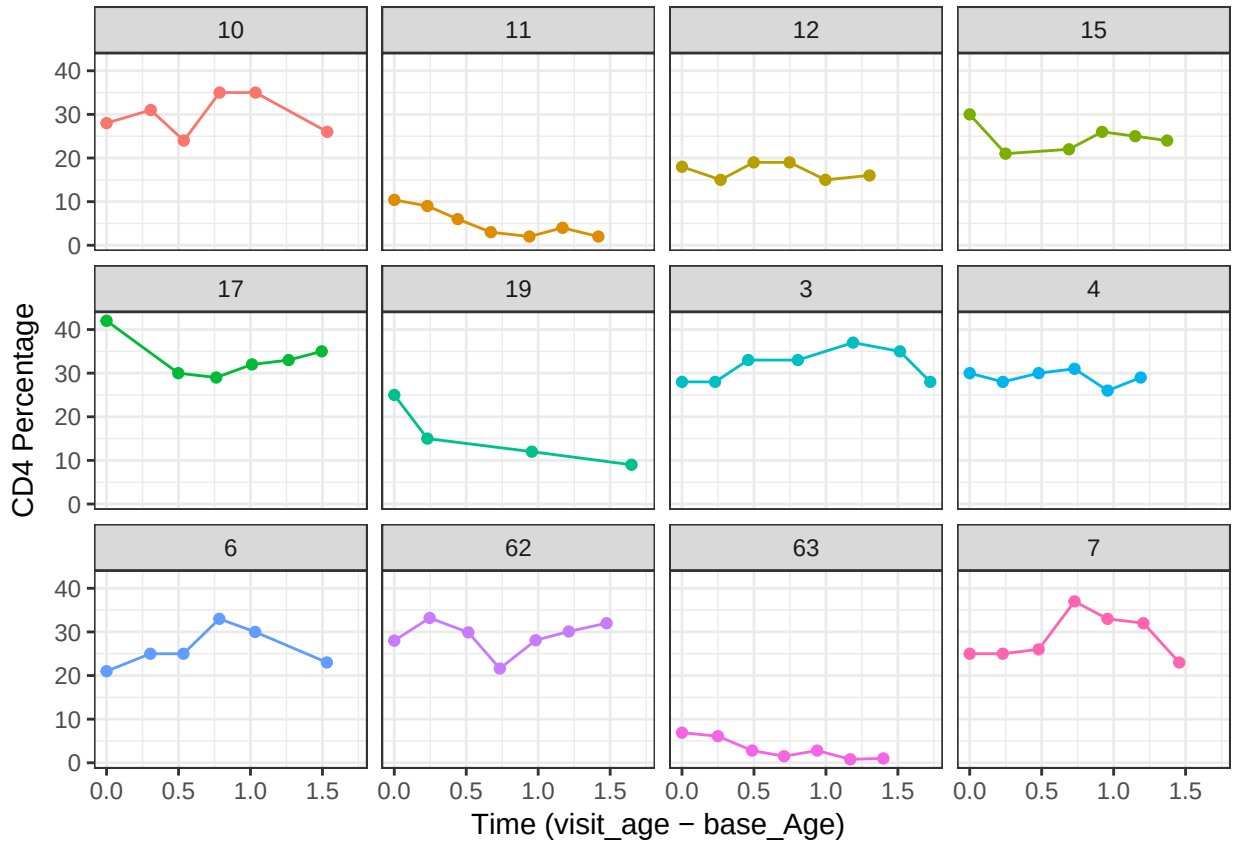



```
#'
#'
#' The plot above is difficult synthesize and inter prate, let's try plotting 5% of the children.
#' The below shows 5% of the children (colored) and the remaining 90% of children (grey scale).
#'
#' ### Plot for 5% of all children
#'


cd4_c11_q4a.plot.2<-  cd4_c11_q4.plot+
      geom_line(aes(alpha=alpha, color=factor(newpid)) )+
   geom_point(aes(alpha=alpha, color=factor(newpid)) )+
  guides(alpha=FALSE)
  #geom_smooth(se=T, colour="black", method = "lm", size=1, linetype=6)

subset <- c(3,4,6,7,10,11,12,15,17,19,62,63)
cd4_c11_q4a.plot.3<- ggplot(cd4.df[cd4.df$newpid %in% subset,],
                            aes(x = time, y = cd4pct)) +
    guides(colour=FALSE) +
    labs(y = "CD4 Percentage", x="Time (visit_age - base_Age) ")+
    theme_bw()+
```

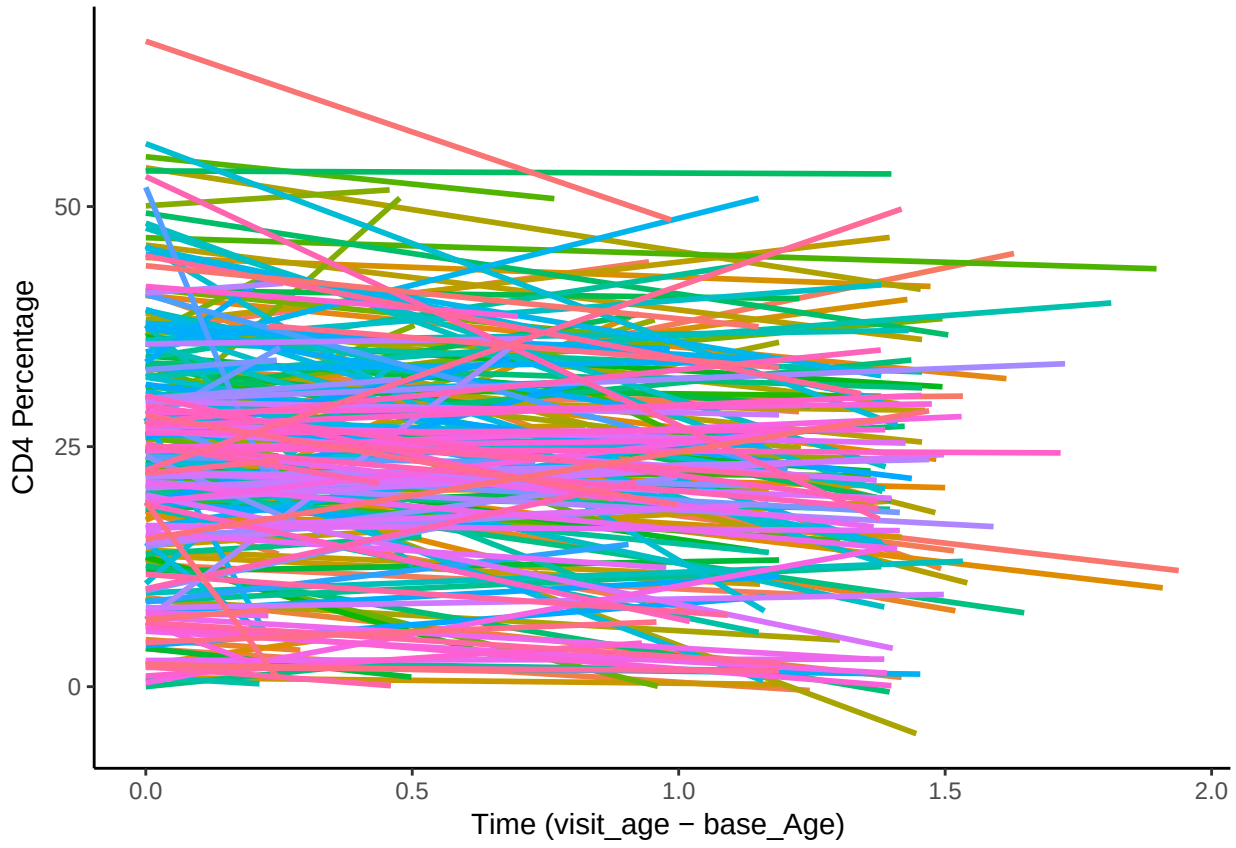
```
    theme(legend.position="none") +
    geom_line()+
    geom_point()+
    aes(color  = factor(newpid)) +
  # geom_smooth(se=T, colour="red", method = "lm",  linetype=6)+
    facet_wrap(~factor(newpid))

 #grid.arrange(cd4_c11_q4a.plot.2, cd4_c11_q4a.plot.3, nrow = 2.)
cd4_c11_q4a.plot.2
```



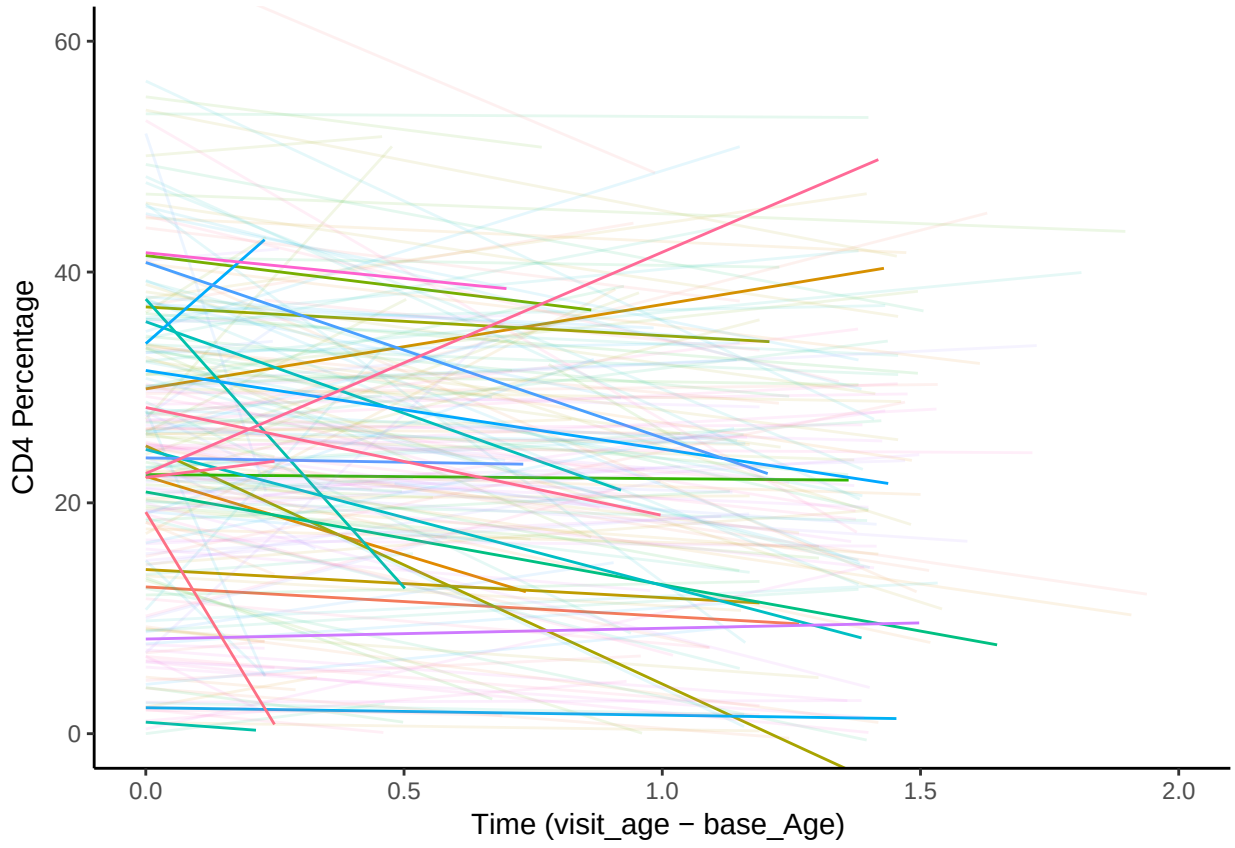

```
cd4_c11_q4a.plot.3
```



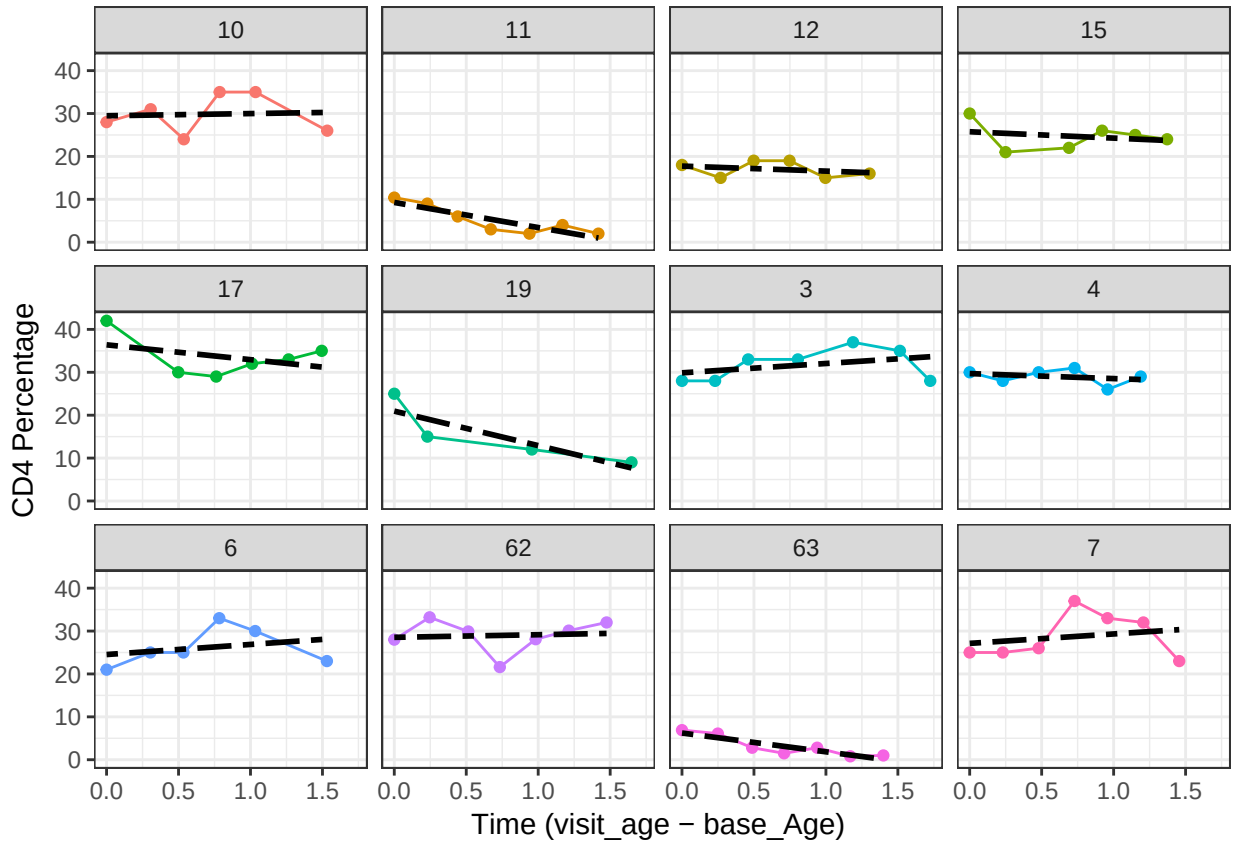

```
#'
#'
#'
#'
#'
#'
#' ## Part B:
#' ``Each child's data has a time course that can be summarized by a linear fit. Estimate these lines a
#'
#'
#'

cd4_c11_q4b.plot.3<- ggplot(cd4.df, aes(x = time, y = cd4pct, color=factor(newpid)) )+
                     #  geom_point()+
                     geom_smooth(se=F,method = "lm", linetype=1) +
                     labs(y = "CD4 Percentage", x="Time (visit_age - base_Age) ")+
                     theme_classic()+
                     theme(legend.position="none")

cd4_c11_q4b.plot.3
```



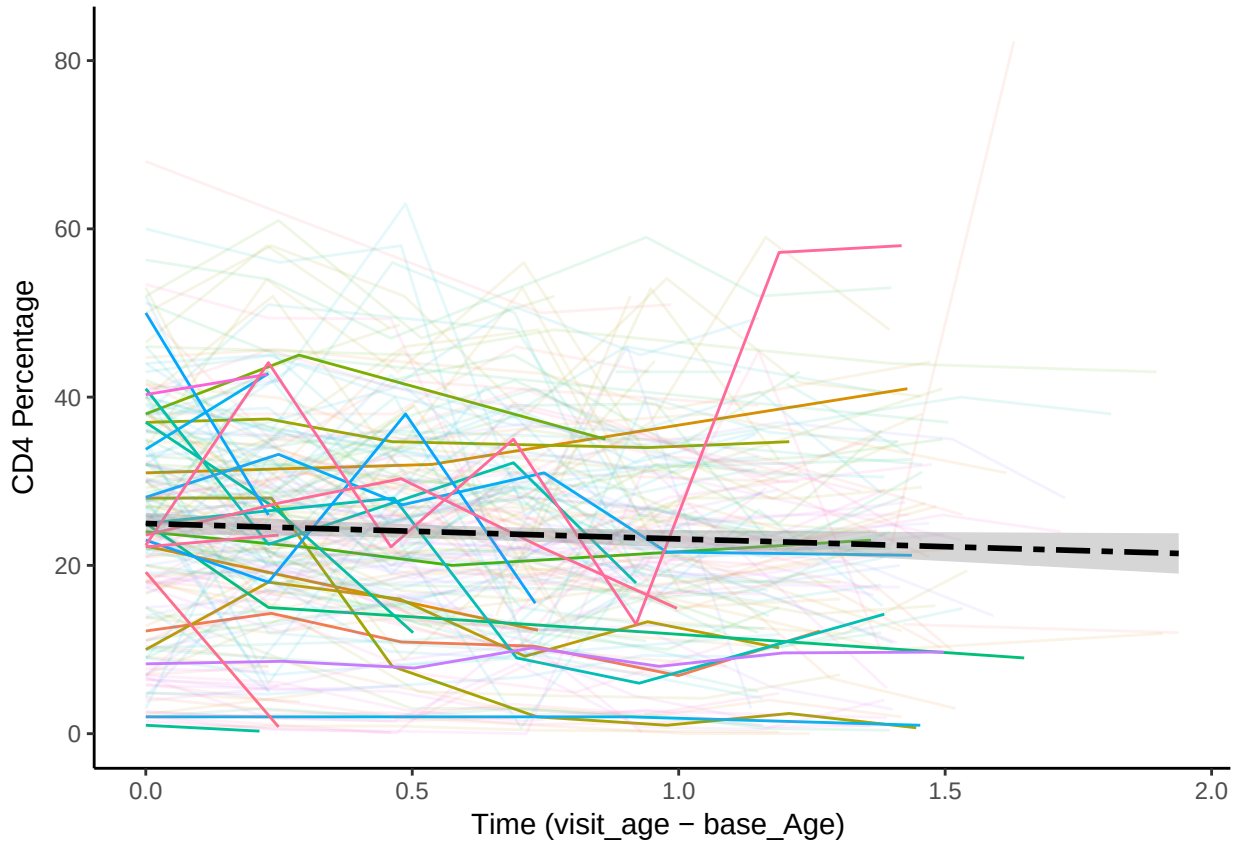

```
#'
#'
#' From the above plot, it is difficult to decipher the magnitude and direction of each regression line
#'
#'

cd4_c11_q4b.plot.4 <- ggplot(cd4.df)+
                      #geom_point(
                      #  aes(x = time, y = cd4pct, alpha=alpha, color=factor(newpid) ) )+
                      geom_line(stat = "smooth",
                                method = lm,
                                aes(x = time, y = cd4pct, alpha=alpha, color=factor(newpid))
                                ) +
                      labs(y = "CD4 Percentage", x="Time (visit_age - base_Age) ")+
                       coord_cartesian(ylim = c(0,60), xlim = c(0, 2))+
                      theme_classic()+
                      theme(legend.position="none")

cd4_c11_q4b.plot.4
```




```
#'
#'
#'
#'

# selective subsetting
subset <- c(3,4,6,7,10,11,12,15,17,19,62,63)
cd4_c11_q4b.plot.2<- ggplot(cd4.df[cd4.df$newpid %in% subset,],
                            aes(x = time, y = cd4pct)) +
    guides(colour=FALSE) +
    labs(y = "CD4 Percentage", x="Time (visit_age - base_Age) ")+
    theme_bw()+
    theme(legend.position="none") +
    geom_line()+
    geom_point()+
    aes(color  = factor(newpid)) +
  # geom_smooth(se=T, colour="red", method = "lm",  linetype=6)+
    facet_wrap(~factor(newpid)) +
    geom_smooth(se=F, colour="black", method = "lm",  linetype=6)
cd4_c11_q4b.plot.2
```




```
#'
#'
#' For some children, cd4 percent increases over time. However, for most children, cd4 percent decrease
#'
#' This is rather odd! we expect that most of the children to have increasing cd4 percent over time. Le
#'
#'
cd4_c11_q4b.plot<- ggplot(cd4.df, aes(x = time, y = cd4pct)) +
 geom_line() +
 # geom_point()+
    guides(colour=FALSE) +
    labs(y = "CD4 Percentage", x="Time (visit_age - base_Age) ")+
    theme_classic()+ guides(fill=FALSE)+
    theme(legend.position="none") +
    aes(alpha=alpha, color=factor(newpid)) +
    geom_smooth(se=T, colour="black", method = "lm", linetype=6)

cd4_c11_q4b.plot
```




```
#'
#'
#' The dotted line is the population level regression and apparently it has a negative slope. It's like
#'
#'
#' ## Part C:
#' ``Set up a model for the children's slopes and intercepts as a function of the treatment and age at
#'
#'
# step 1
df_1 = cd4.df %>% na.omit()  %>%
  group_by(newpid) %>%
  dplyr::select(newpid,cd4pct,time) %>%
  mutate(intercept = coef(lm(cd4pct~time))[1]) %>%
  mutate(slope = coef(lm(cd4pct~time))[2]) %>%
  dplyr::select(newpid, intercept,slope)

# step 2
df_2 = inner_join(cd4.df,df_1,by="newpid")

model.intercept = lm(intercept~baseage+treatmnt, data =df_2 )
model.slope = lm(slope~baseage+treatmnt, data =df_2 )

texreg(list(model.intercept,model.slope),
       custom.model.names = c("Intercept Model", "Slope Model"),
       single.row=TRUE,  float.pos = "h")
```

```
##
## \begin{table}[h]
## \begin{center}
## \begin{tabular}{l c c }
## \hline
##  & Intercept Model & Slope Model \\
## \hline
## (Intercept) & $28.66 \; (0.35)^{***}$ & $-3.77 \; (0.28)^{***}$ \\
## baseage     & $-1.05 \; (0.08)^{***}$ & $0.17 \; (0.06)^{**}$   \\
## treatmnt2   & $3.32 \; (0.34)^{***}$  & $-0.51 \; (0.27)$       \\
## \hline
## R$^2$       & 0.05                    & 0.00                    \\
## Adj. R$^2$  & 0.05                    & 0.00                    \\
## Num. obs.   & 5722                    & 5693                    \\
## RMSE        & 12.61                   & 10.24                   \\
## \hline
## \multicolumn{3}{l}{\scriptsize{$^{***}p<0.001$, $^{**}p<0.01$, $^*p<0.05$}}
## \end{tabular}
## \caption{Statistical models}
## \label{table:coefficients}
## \end{center}
## \end{table}
```

```
#'
#'
#' \onecolumn
#'
#' # Chapter 12 Question 2:
#' Continuing with the analysis of the CD4 data from Exercise 11.4:
#'
#' ## Part A:
#' ``Write a model predicting CD4 percentage as a function of time with varying intercepts across child
#'
#'
model_chap12_q2a = lmer(cd4pct~1+time+(1|newpid),data=cd4.df,REML=F) # varying intercepts

texreg(list(model_chap12_q2a),
       custom.model.names = c("Model 12.2 A"),
       single.row=TRUE,  float.pos = "h")
```

```
##
## \begin{table}[h]
## \begin{center}
## \begin{tabular}{l c }
## \hline
##  & Model 12.2 A \\
## \hline
## (Intercept)             & $25.04 \; (0.80)^{***}$ \\
## time                    & $-3.00 \; (0.51)^{***}$ \\
## \hline
## AIC                     & 7889.03                 \\
## BIC                     & 7908.94                 \\
## Log Likelihood          & -3940.52                \\
## Num. obs.               & 1072                    \\
## Num. groups: newpid     & 250                     \\
```

```
## Var: newpid (Intercept) & 128.77                  \\
## Var: Residual           & 53.19                   \\
## \hline
## \multicolumn{2}{l}{\scriptsize{$^{***}p<0.001$, $^{**}p<0.01$, $^*p<0.05$}}
## \end{tabular}
## \caption{Statistical models}
## \label{table:coefficients}
## \end{center}
## \end{table}
```

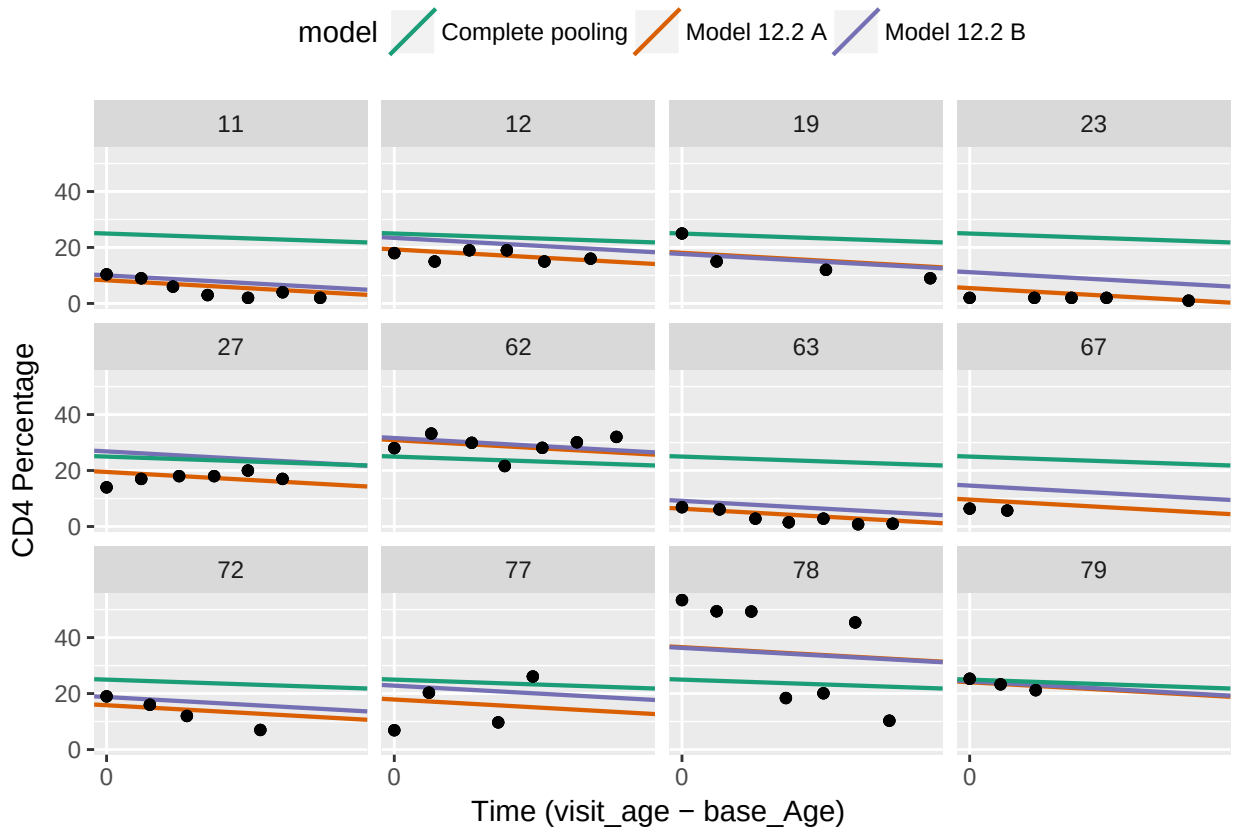
```
#summary(model_chap12_q2a)
#coef(model_chap12_q2a)

#'
#'
#' Given that this model is a random intercept model, it only accounts for "idiosyncratic" variation th
#'
#' * **Time** -  Given that time is a fixed effect on every child, each year on the study is associated
#'
#' ## Part B:
#' ``Extend the model in (a) to include child-level predictors (that is, group-level predictors) for tr
#'
#'
#'
#'
#model_chap12_q2b=lmer(cd4pct~1+time+(1+treatmnt+baseage|newpid),data=cd4.df,REML=F)
model_chap12_q2b=lmer(cd4pct~1+time+treatmnt+baseage+(1|newpid),data=cd4.df,REML=F)



texreg(list(model_chap12_q2b),
       custom.model.names = c("Model 12.2 B"),
       single.row=TRUE,  float.pos = "h")
```

```
##
## \begin{table}[h]
## \begin{center}
## \begin{tabular}{l c }
## \hline
##  & Model 12.2 B \\
## \hline
## (Intercept)             & $27.71 \; (1.55)^{***}$ \\
## time                    & $-2.96 \; (0.51)^{***}$ \\
## treatmnt2               & $1.21 \; (1.50)$        \\
## baseage                 & $-0.95 \; (0.33)^{**}$  \\
## \hline
## AIC                     & 7884.23                 \\
## BIC                     & 7914.10                 \\
## Log Likelihood          & -3936.12                \\
## Num. obs.               & 1072                    \\
## Num. groups: newpid     & 250                     \\
## Var: newpid (Intercept) & 123.51                  \\
## Var: Residual           & 53.21                   \\
## \hline
## \multicolumn{2}{l}{\scriptsize{$^{***}p<0.001$, $^{**}p<0.01$, $^*p<0.05$}}
```

```
## \end{tabular}
## \caption{Statistical models}
## \label{table:coefficients}
## \end{center}
## \end{table}
```

```
#summary(model_chap12_q2b)
#coef(model_chap12_q2b)
#toLatex(mtable(model_chap12_q2b))
#'
#'
#' In general, adding child-level predictors (i.e, group-level predictors) definitely improves the mode
#'
#' ### Interpretation
#'
#' * **Treatment** - Children who were on second treatment had 1.2125% higher cd4 percent on average  t
#'
#' *  **Baseline Age** - 1 year difference in baseline age is associate with changing cd4 percentage by
#'
#' * **Time** -  Each year on treatment was associated with a decreased in cd4 percentage by 2.9615 % o
#'
#' ## Part C:
#' ``Investigate the change in partial pooling from (a) to (b) both graphically and numerically.``
#'
#'
#'
#'
#' ### Graphically
#'
#'

# complete pooling
complete_pooling.fit<-lm(cd4pct~time, data=cd4.df)

df_complete_pooling <- data_frame(
  model = "Complete pooling",
  newpid = unique(cd4.df$newpid),
  intercept = coef(complete_pooling.fit)[1],
  slope_time = coef(complete_pooling.fit)[2])



# model for question 2 A
df = coef(model_chap12_q2a)[["newpid"]]
df$intercept = df$'(Intercept)'
df_chap12_q2a <- df %>%
   dplyr::select(intercept, slope_time = time)  %>%
  as_tibble() %>%
   rownames_to_column("newpid") %>%
  add_column(model = "Model 12.2 A")


# model for question 2 B
df = coef(model_chap12_q2b)[["newpid"]]
```

```r
df$intercept = df$'(Intercept)'
df_chap12_q2b <- df %>%
   dplyr::select(intercept,
         slope_time = time,
         treatmnt_slope= treatmnt2, baseage_slope= baseage )  %>%
  as_tibble() %>%
   rownames_to_column("newpid") %>%
  add_column(model = "Model 12.2 B")


# bind data
df_models <- bind_rows(df_chap12_q2a, df_chap12_q2b, df_complete_pooling) %>%
  left_join(cd4.df, by = "newpid")

# create subset
subset <- c(67,72,77,78,79,11,12,19,27,23,62,63)

# construct plot
model_comparison <- ggplot(df_models[df_models$newpid %in% subset,]) +
  aes(x = time, y = cd4pct) +
  geom_abline(aes(intercept = intercept, slope = slope_time, color = model),
              size = .75) +
   #theme_bw()+
    #geom_line( linetype=6)+
    geom_point()+
  facet_wrap("newpid") +
  labs(y = "CD4 Percentage", x="Time (visit_age - base_Age) ")+
  scale_x_continuous(breaks = 0:4 * 2) +
  scale_color_brewer(palette = "Dark2") +
  theme(legend.position = "top")

model_comparison
```

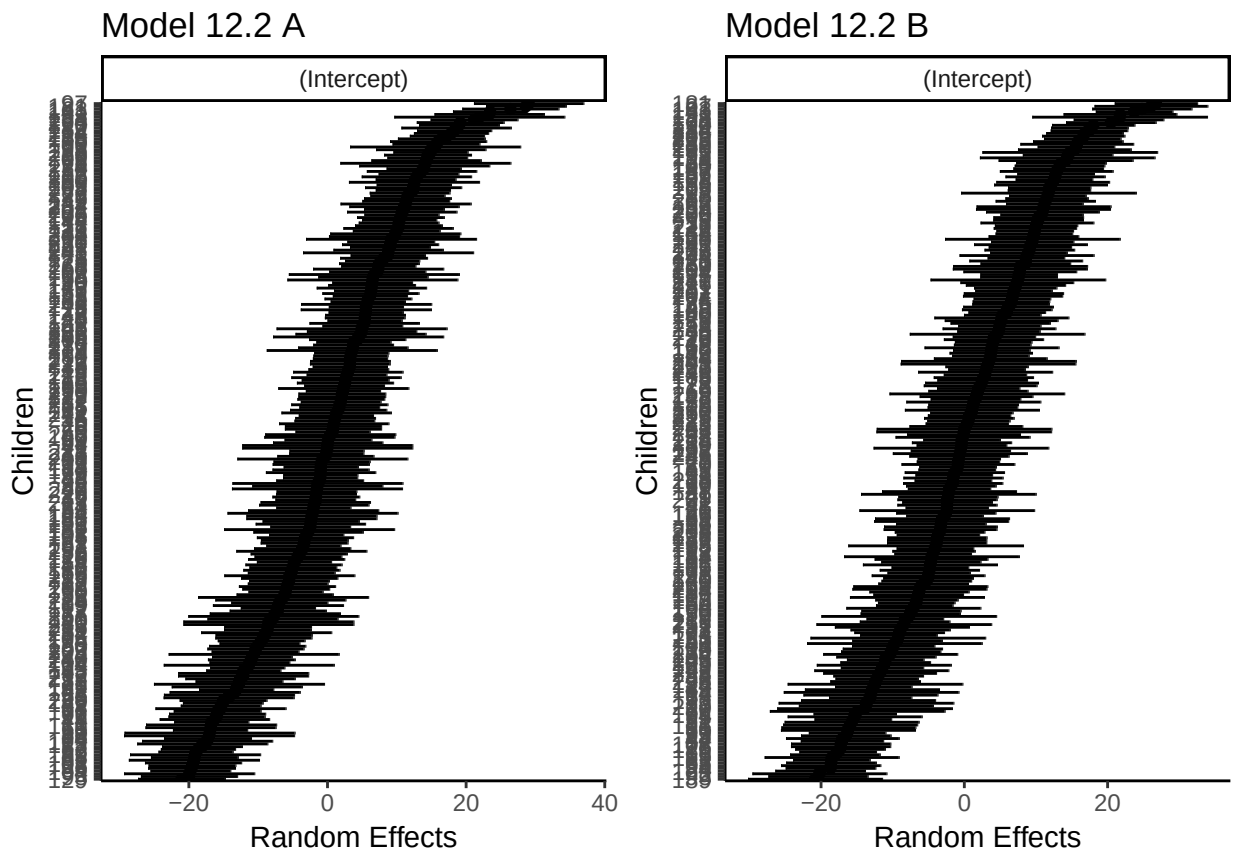



```
#'
#'
#'
#' A few observation to make.
#'
#' * From the plot, observe that the partial pooling model with added child-level predictors (Model 12.
#'
#' * If you take a look at children with less observations (incomplete data), you will notice that Mode
#'
#'
#' ### Numerically
#'

anova_mixd <- anova(model_chap12_q2a,model_chap12_q2b)
xtable(anova_mixd, comment=FALSE)
```

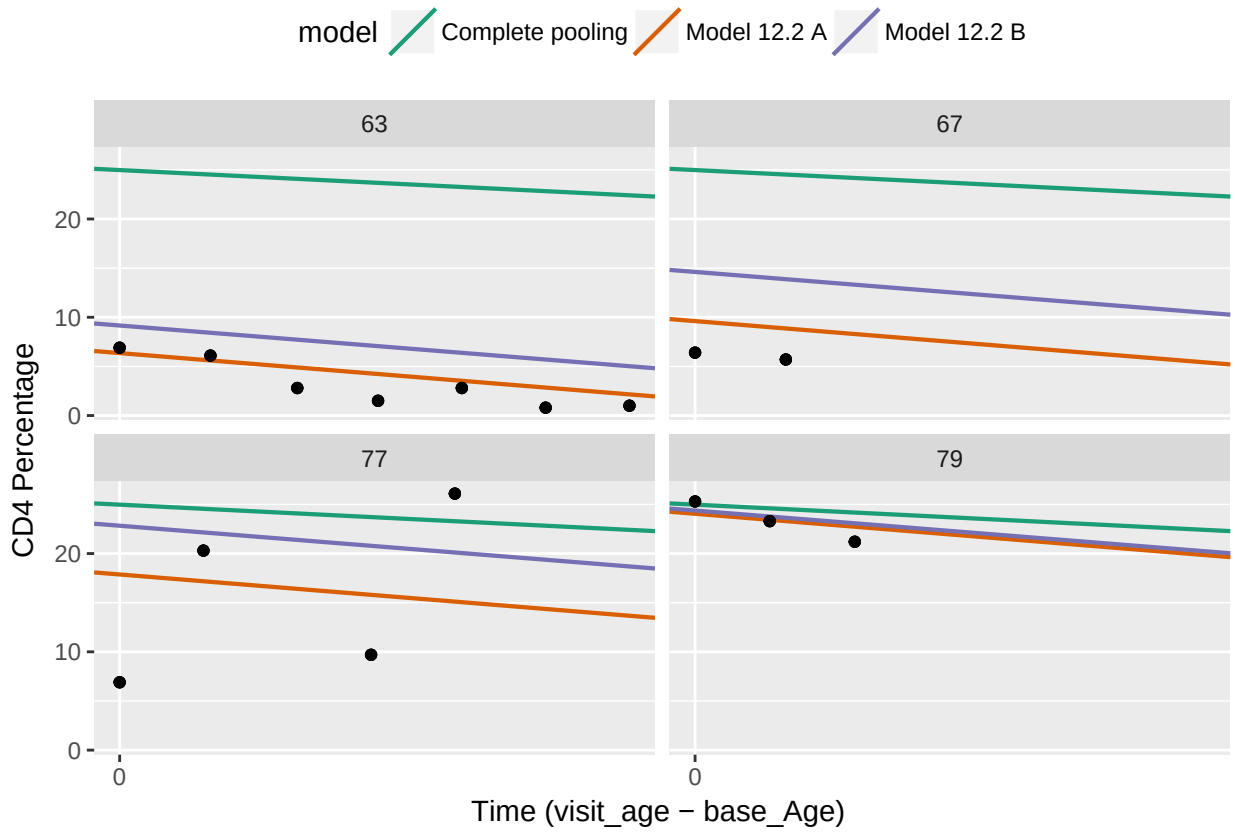
```
## % latex table generated in R 3.4.3 by xtable 1.8-2 package
## % Fri Apr 27 17:00:33 2018
## \begin{table}[ht]
## \centering
## \begin{tabular}{lrrrrrrrr}
##   \hline
##  & Df & AIC & BIC & logLik & deviance & Chisq & Chi Df & Pr($>$Chisq) \\
##   \hline
## model\_chap12\_q2a & 4 & 7889.03 & 7908.94 & -3940.52 & 7881.03 &  &  &  \\
##   model\_chap12\_q2b & 6 & 7884.23 & 7914.10 & -3936.12 & 7872.23 & 8.80 & 2 & 0.0123 \\
```

```
##    \hline
## \end{tabular}
## \end{table}
```

```
#'
#'
#' Using ANOVA to check the changes between the 2 models, we find out that adding additional child-leve
#'
#'
#'
#'
#coef(model_chap12_q2a)
q2a_plot <- as.data.frame(lme4::ranef(model_chap12_q2a,condVar=T)) %>%
    ggplot(aes(y=grp,x=condval))+
    geom_point()+facet_wrap(~term,scales="free_x")+
    geom_errorbarh(aes(xmin=condval-2*condsd,xmax=condval+2*condsd),height=0)+
   labs(x="Random Effects", y="Children") +
      theme_classic()+
      guides(fill=FALSE) +
   ggtitle("Model 12.2 A")

q2b_plot <-  as.data.frame(lme4::ranef(model_chap12_q2b,condVar=T)) %>%
    ggplot(aes(y=grp,x=condval))+
    geom_point()+facet_wrap(~term,scales="free_x")+
    geom_errorbarh(aes(xmin=condval-2*condsd,xmax=condval+2*condsd),height=0)+
    labs(x="Random Effects", y="Children") +
      theme_classic()+
      guides(fill=FALSE)+
   ggtitle("Model 12.2 B")

grid.arrange(q2a_plot,q2b_plot, ncol = 2.)
```



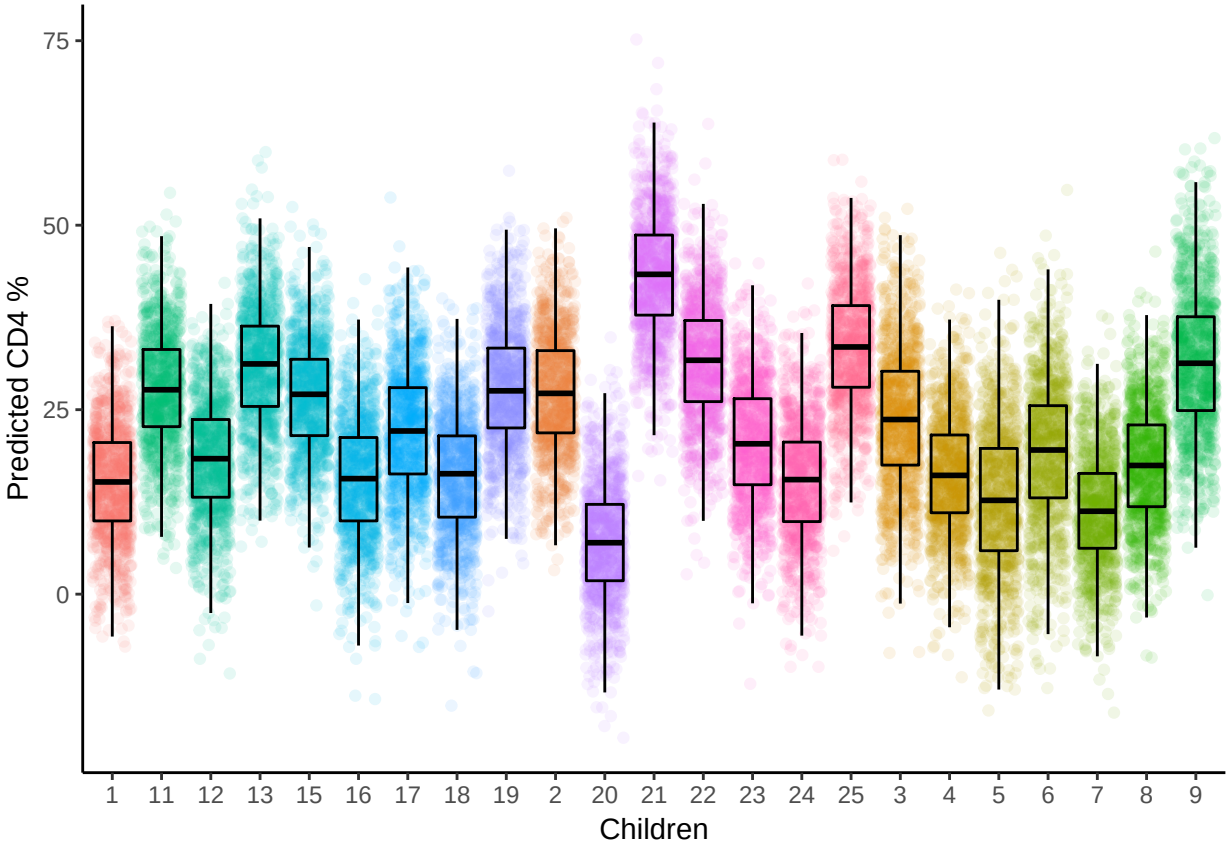

```
#'
#'
#' From the above plot, notice the scale of each plot. Model 12.2.A  has higher random error than Model
#'
#' ## Part D:
#' Compare results in (b) to those obtained in part (c).
#'
#' As seen in the previous question, We observed that by adding child-level predictors (i.e, group-leve
#' However, the residual did not reduced by much  possibly because we did not add another random effect
#'
#'
df_zoom <- df_models %>%
  filter(newpid %in% c("63", "67", "79", "77"))

model_comparison  %+% df_zoom
```




```
#'
#'
#' In summary, this shrinkage effect observed in model 12.2.B ensures that our model compromised betwee
#'
#' \onecolumn
#'
#' # Chapter 12 Question 3:
#'
#' `Predictions for new observations and new groups:`
#'
#' ## Part A:
#' ``Use the model fit from Exercise 12.2(b) to generate simulation of predicted CD4 percentages for ea
#'
#'


set.seed(111)
n.sims = 1000

# create hypothetical next time point
cd4.df3 = cd4.df %>%
            na.omit() %>% # remove missing values
            group_by(newpid) %>% # group by PID
            mutate(current.time = max(time)) %>% # get maximum time for each child
            mutate(future.time = current.time + .25) # add .25 year beacause appintments are done quart
```

```r
# slelect only relevant time
cd4.df4 = cd4.df3 %>%
          dplyr::select(newpid,treatmnt,baseage,future.time) %>%
          rename(time=future.time) %>%
          unique() # make sure it is a one row per patient dataset

# Simulate 1000 times using sim fx
model_chap12_q2b.sim = sim(model_chap12_q2b,n.sims)
# fetch array offixed effect for intercept for each simulation
fixed.intercept = coef(model_chap12_q2b.sim)$fixef[,1]
# get random effect array for intercept for each simulation
rand.eff = coef(model_chap12_q2b.sim)$ranef$newpid
# get the intercept for each person by summing fixed and random effects
sim.intercept = fixed.intercept + rand.eff
# create matrix
sim.intercept = matrix(sim.intercept,nrow=n.sims,ncol=length(unique(cd4.df$newpid)))

# generate x matrix
X = cbind(1,cd4.df4$time,as.numeric(cd4.df4$treatmnt)-1,cd4.df4$baseage)

cd4.df.pred.future.sim = list()
for (simnum in 1:n.sims){
  b.hat = rbind(sim.intercept[simnum,],coef(model_chap12_q2b.sim)$fixef[simnum,2],coef(model_chap12_q2b
  rownames(b.hat) = c("intercept","time","treatmnt","baseage")

  sigma.y.hat <- sigma.hat(model_chap12_q2b.sim)

  cd4.future.sim = rep(NA,nrow(cd4.df4))
  # actual simulation
  for (idnum in 1:length(cd4.future.sim)){
    cd4.future.sim[idnum] = rnorm (n.sims, X[idnum,] %*% b.hat[,idnum], sigma.y.hat)
  }

  cd4.df.pred.future.sim[[simnum]] = cd4.df4 %>%
                  cbind(cd4.future.sim) %>%
                  rename(future.time = time)
}

  cd4.df.pred.future.sim.all = bind_rows(cd4.df.pred.future.sim) %>%
                                      group_by(newpid) %>%
                                      mutate(mean.future.cd4 = mean(cd4.future.sim))

  cd4.df.pred.future.sim.all$index = as.numeric(cd4.df.pred.future.sim.all$newpid)
  cd4.df.set1 = cd4.df.pred.future.sim.all%>%
                filter(index <= 25)

  simulation_plot_12_3 <- ggplot(cd4.df.set1, aes(x=newpid, y =cd4.future.sim, color=factor(index))) +
  geom_jitter(alpha = 0.1) +
    #stat_boxplot(geom="errorbar", width=.5,  color = "black")+
  geom_boxplot(alpha = 0, color = "black") +
  labs(y = "Predicted CD4 %", x="Children")+
  theme_classic()+ guides(fill=FALSE)+
  theme(legend.position="none")
```

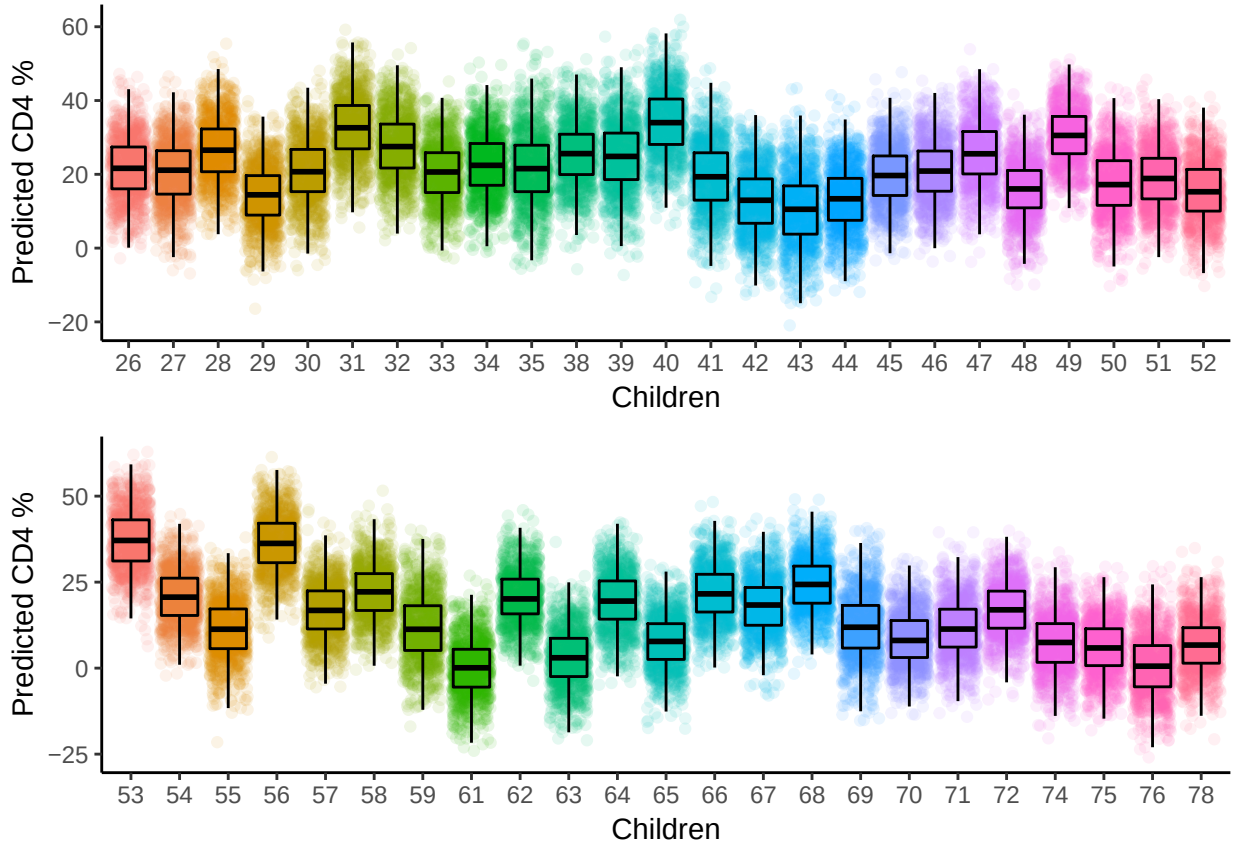
```
simulation_plot_12_3
```




```
plot1.df = cd4.df.pred.future.sim.all %>% dplyr::filter(index > 25 & index <=52)
plot2.df = cd4.df.pred.future.sim.all %>% dplyr::filter(index > 52 & index <=78)
plot3.df =  cd4.df.pred.future.sim.all %>% dplyr::filter(index > 78 & index <=104)
plot4.df =  cd4.df.pred.future.sim.all %>% dplyr::filter(index > 104 & index <=129)
plot5.df =  cd4.df.pred.future.sim.all %>% dplyr::filter(index > 129 & index <=154)
plot6.df =  cd4.df.pred.future.sim.all %>% dplyr::filter(index > 154 & index <=179)
plot7.df =  cd4.df.pred.future.sim.all %>% dplyr::filter(index > 179 & index <=204)
plot8.df =  cd4.df.pred.future.sim.all %>% dplyr::filter(index > 204)
```

```
#'
#'
#' Since most appointments/visits occurred every quarterly as shown in Chapter 11 Question 4, we made o
#'
#' In general most children with multiple visits had much more precise prediction interval while childr
#'
#'
#'
#'
#'
```

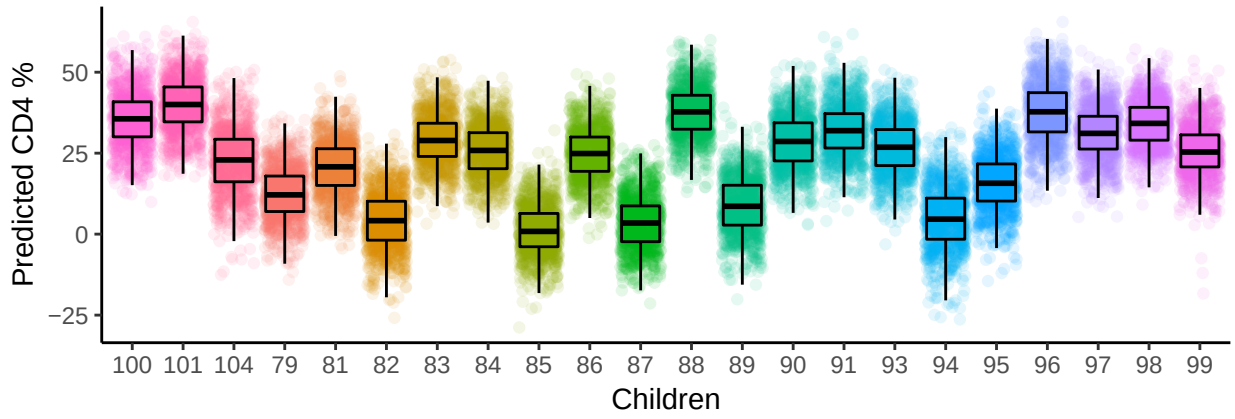
```
grid.arrange( simulation_plot_12_3  %+% plot1.df,
               simulation_plot_12_3  %+% plot2.df,

              nrow =2., ncol = 1.)
```

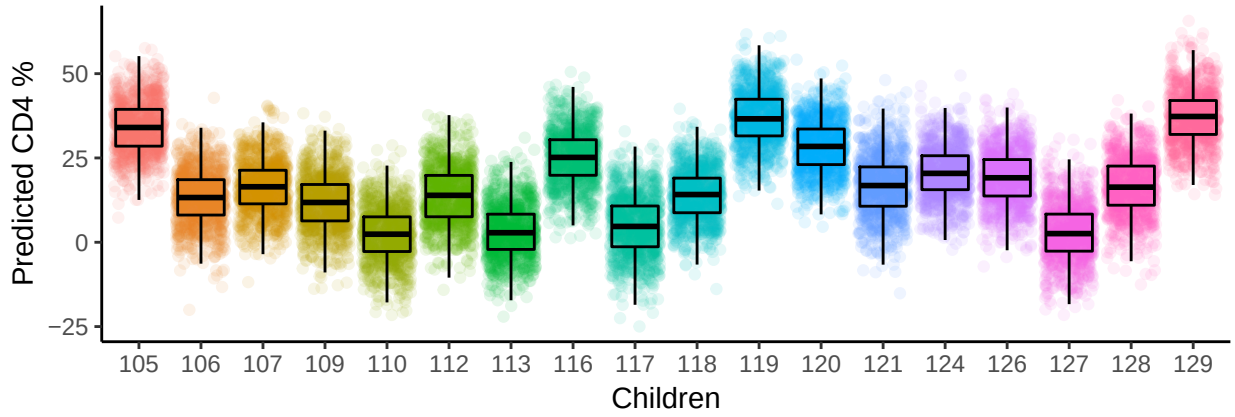




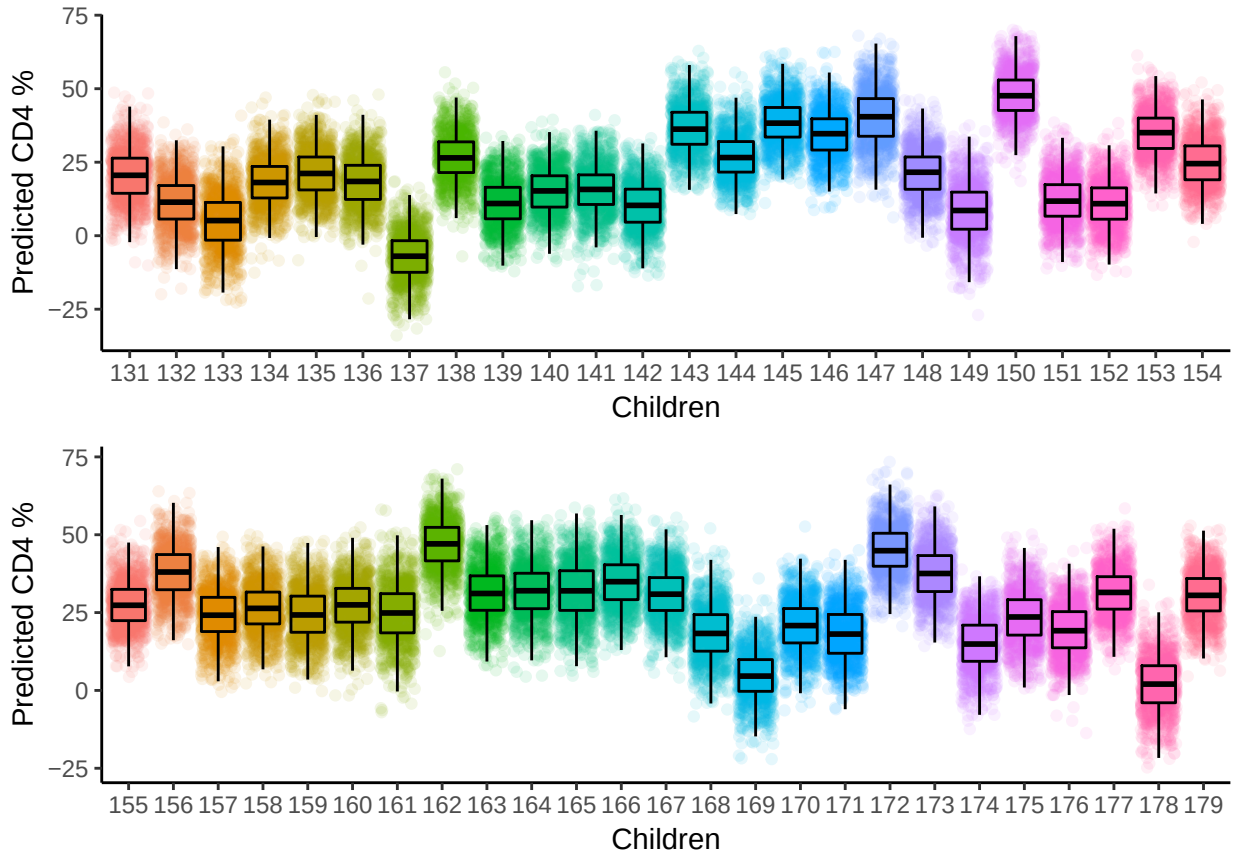


```
#'
#'
#'

 grid.arrange(
               simulation_plot_12_3  %+% plot3.df,
               simulation_plot_12_3  %+% plot4.df,
              nrow =2., ncol = 1.)
```

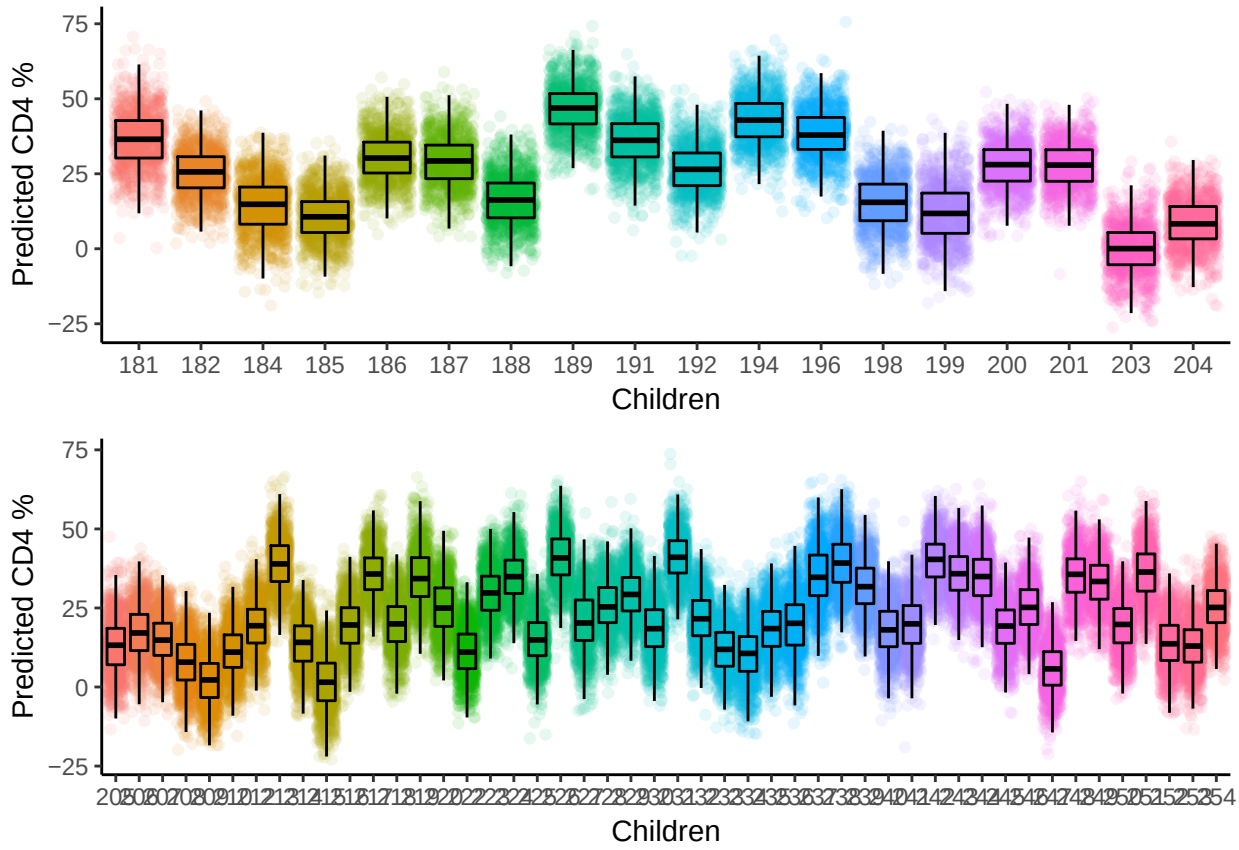



```
#'
#'
#'

    grid.arrange( simulation_plot_12_3  %+% plot5.df,
                  simulation_plot_12_3  %+% plot6.df,
                nrow =2., ncol = 1.)
```

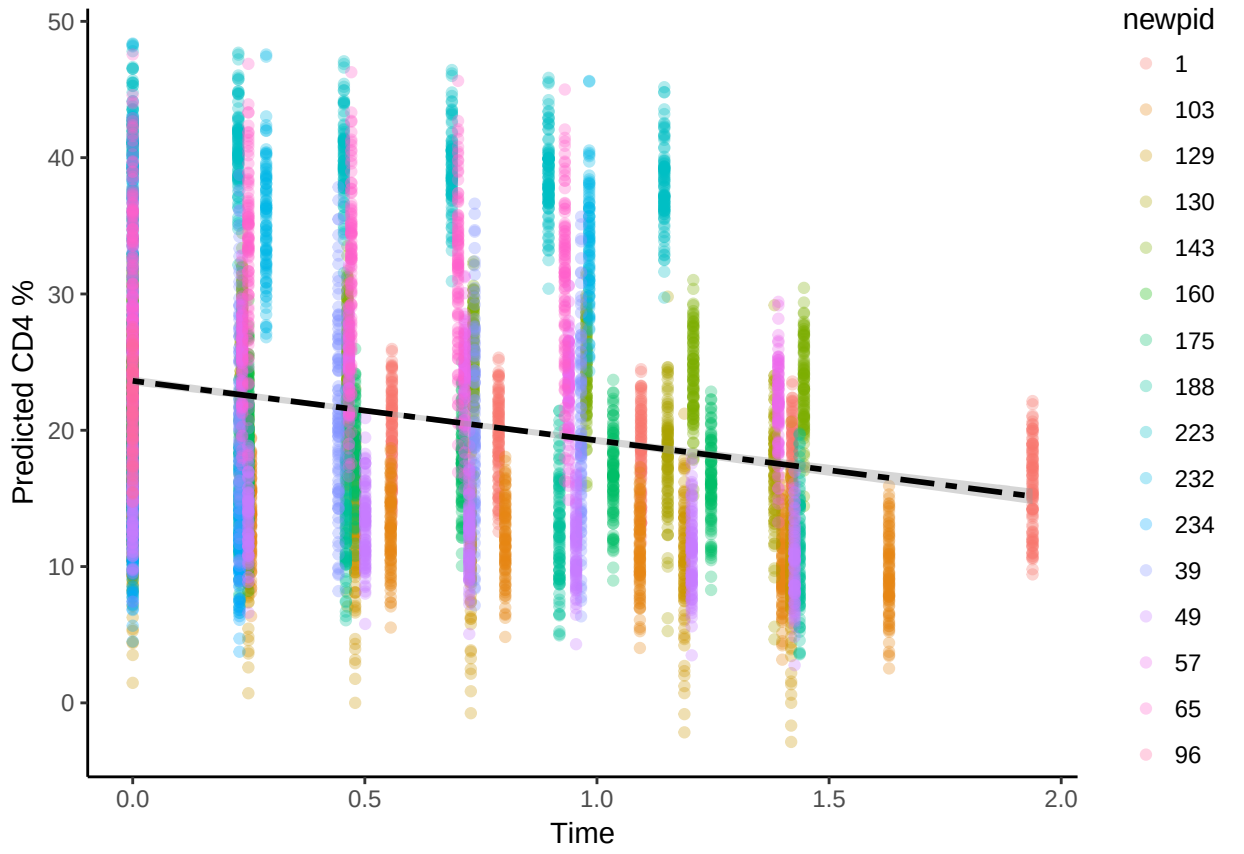



```
#'
#'
#'

    grid.arrange(
                simulation_plot_12_3  %+% plot7.df,
                simulation_plot_12_3  %+% plot8.df,
               nrow =2., ncol = 1.)
```

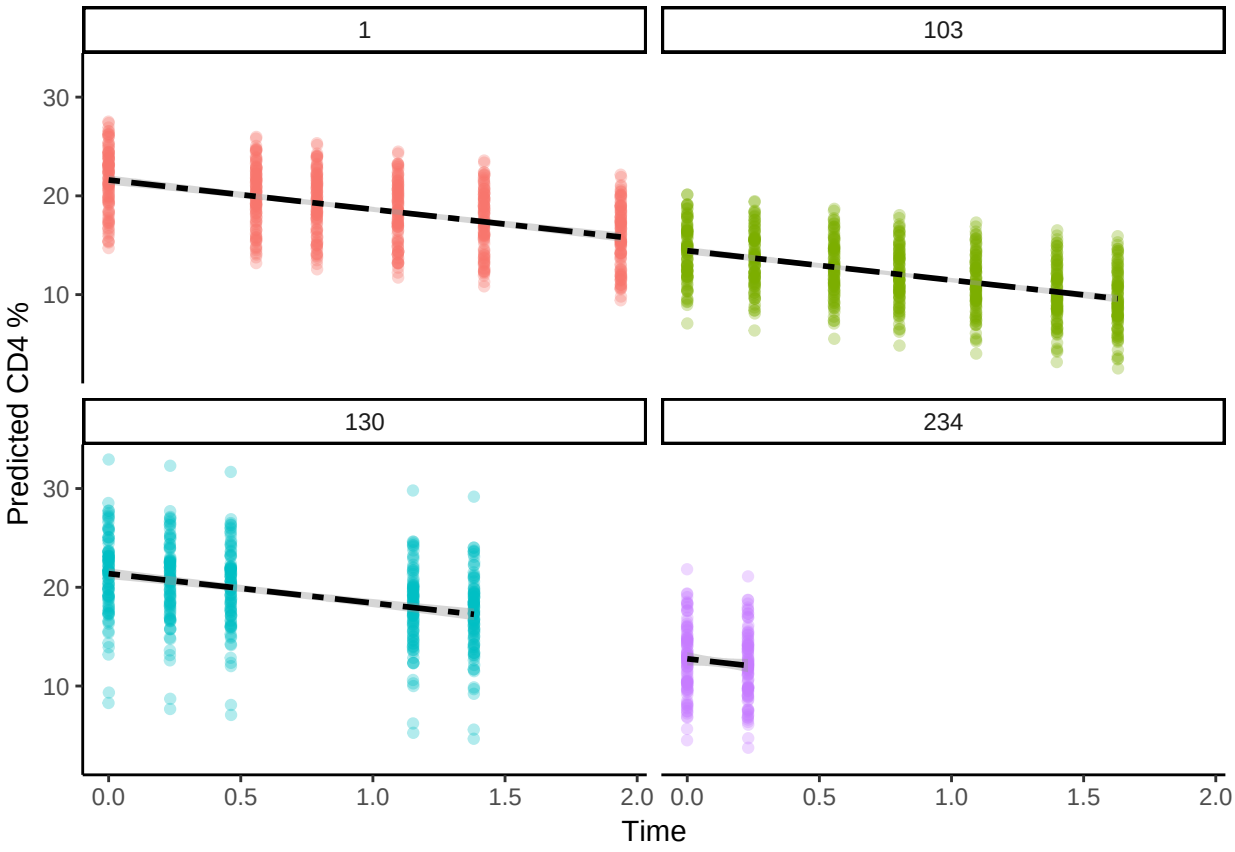



```
#'
#'
#'
#' ## Part B:
#' ``Use the same model fit to generate simulations of CD4 percentages at each of the time periods for
#'
#'

n.sims = 100

cd4.df4 = cd4.df %>%
  dplyr::select(newpid,treatmnt,baseage,time)

# Simulates the fit model 1000 times
model_chap12_q2b.sim = sim(model_chap12_q2b,n.sims)
# Extracts fixed effect for intercept for each simulation
fixed.intercept = coef(model_chap12_q2b.sim)$fixef[,1]
# Extracts random effect for intercept for each person for each simulation
rand.eff = coef(model_chap12_q2b.sim)$ranef$newpid
# Computes intercept for each person for each simulation
sim.intercept = fixed.intercept + rand.eff
sim.intercept = matrix(sim.intercept,nrow=n.sims,ncol=length(unique(cd4.df4$newpid)))

X = cbind(1,cd4.df4) %>% dplyr::select(1,time,treatmnt,baseage,newpid) %>% dplyr::mutate(treatmnt =as.nu

cd4.df.4y.pred.sim = list()
```

```r
for (simnum in 1:n.sims){
  b.hat = rbind(sim.intercept[simnum,],coef(model_chap12_q2b.sim)$fixef[simnum,2],coef(model_chap12_q2b
  rownames(b.hat) = c("intercept","time","treatmnt","baseage")
  colnames(b.hat) = unique(X$newpid)

  cd4.4y.pred.sim = rep(NA,nrow(cd4.df4))
  for (i in 1:length(cd4.4y.pred.sim)){
    cd4.4y.pred.sim[i] = as.vector(t(X[i,1:4])) %*% as.vector(t(b.hat[,as.character(X[i,"newpid"])]))
  }

  cd4.df.4y.pred.sim[[simnum]] = cd4.df4 %>% cbind(cd4.4y.pred.sim)
}

cd4.df.4y.pred.sim.all = bind_rows(cd4.df.4y.pred.sim) %>%
                         arrange(newpid) %>%
                         group_by(newpid) %>%
                         mutate(mean.4yo.cd4 = mean(cd4.4y.pred.sim)) %>%
                         filter(baseage>3.75 & baseage < 4.25)

ggplot(cd4.df.4y.pred.sim.all,
       aes(x=time,y=cd4.4y.pred.sim)) +
      geom_jitter(alpha = 0.3, aes(color=newpid)) +
    geom_smooth(se=T, colour="black", method = "lm", linetype=6) +
    #stat_boxplot(geom="errorbar", width=.5,  color = "black")+
  #geom_boxplot(alpha = 0, color = "black") +
  labs(y = "Predicted CD4 %", x="Time")+
  theme_classic()+ guides(fill=FALSE)
```

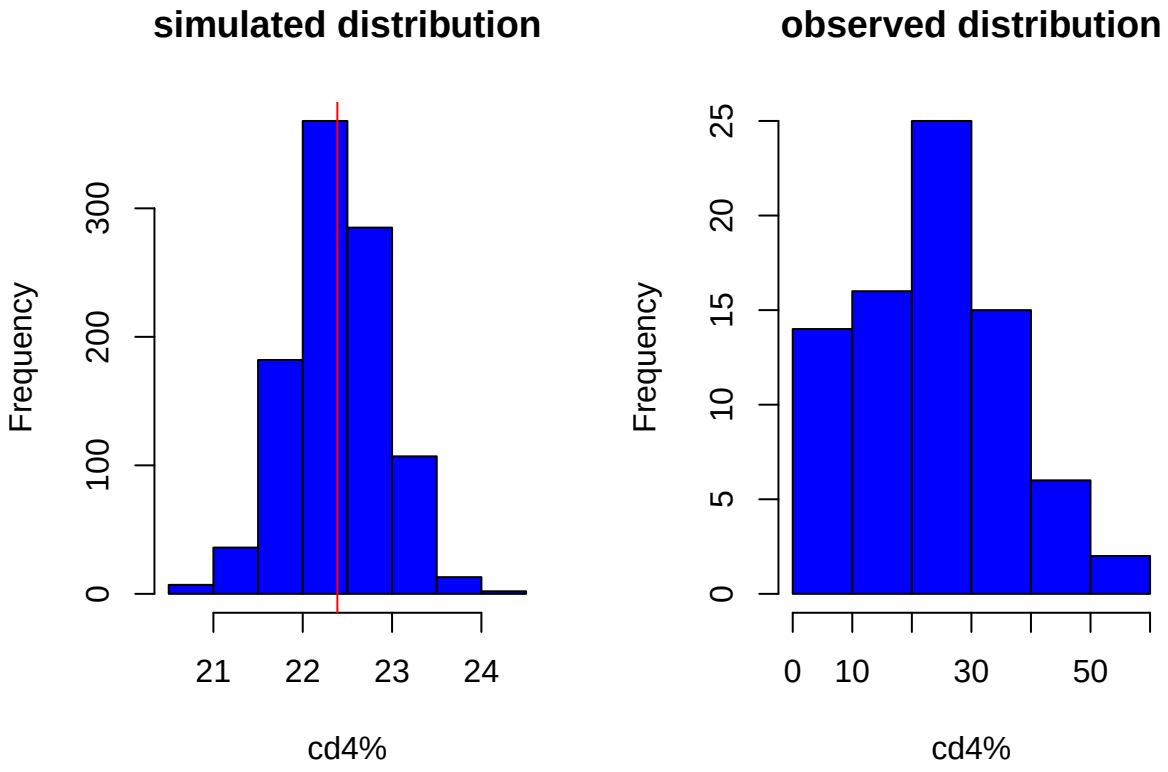



```
 # theme(legend.position="none")


#'
#'
#'  We identified  16 children who had a baseline age of 4  years or  (+-.25). From the plot it is clea
#'
#'
#'
cd4.df.4y.pred.plot<-ggplot(cd4.df.4y.pred.sim.all,
       aes(x=time,y=cd4.4y.pred.sim,color=factor(newpid))) +
      geom_jitter(alpha = 0.3) +
        geom_smooth(se=T, colour="black", method = "lm", linetype=6) +
        #stat_boxplot(geom="errorbar", width=.5,  color = "black")+

      labs(y = "Predicted CD4 %", x="Time")+
      facet_wrap(~newpid)+
      theme_classic()+ guides(fill=FALSE)+
      theme(legend.position="none")

 plot1.df = cd4.df.4y.pred.sim.all[cd4.df.4y.pred.sim.all$newpid %in% c(1,103,234,130),]
cd4.df.4y.pred.plot  %+% plot1.df
```




```
#'
#'
#' \onecolumn
#'
#' # Chapter 12 Question 4:
#' ``Posterior predictive checking: continuing the previous exercise, use the fitted model from Exercis
#'
#'
#'
n.sims = 1000
# get uniqui child id
child_id = cd4.df%>%
  na.omit()%>%
  distinct(newpid, treatmnt, baseage)

# count num of id
num_id = dim(child_id)[1]

final_time_point.df = cd4.df%>%
  filter(visitno == 19)%>%
  dplyr::select(time, cd4pct)

final_time = mean(final_time_point.df$time)
y = array(NA, c(n.sims, num_id))
average = array(NA, n.sims)
```

```
for(s in 1:num_id){
  sigma.y.hat = sigma.hat(model_chap12_q2b)$sigma$data
  coef.hat = as.matrix(coef(model_chap12_q2b)$newpid)[s,]
  variable = child_id[s,]
  y[,s] = rnorm (n.sims, coef.hat %*% c(1, final_time, variable$treatmnt, variable$baseage), sigma.y.hat
}

predicted = apply(y, 1, mean)
actual = cd4.df%>%
   na.omit()%>%
  filter(visitno == 19)%>%
  dplyr::select(cd4pct)

par(mfrow = c(1, 2))
hist(predicted, main = "simulated distribution", col="blue",  xlab="cd4%")
abline(v=mean(predicted),col="red")
hist(actual$cd4pct, main = "observed distribution", col="blue", xlab="cd4%")
```




```
#'
#'
#'
#'
#' A few observation to make. In general we notice the shrinkage effect depicted by the model when we t
#'
#' In summary, this shrinkage effect observed in model 12.2.B ensures that our model compromised betwee
#'
```

```
#'
#' \onecolumn
#'
#' # Source Code
#'
#'


#'
```