Homework 4 Solution

DATA 2020

April 21, 2018

1 Problem 1 (ISL 7.1) [5 points]

It was mentioned in ISL chapter 7 that a cubic regression spline with one knot at ξ can be obtained using a basis of the form x, x^2, x^3, $(x-\xi)^3_+$, where $(x-\xi)^3_+ = (x-\xi)^3$ if $x > \xi$ and equals 0 otherwise. We will now show that a function of the form

$$f(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 (x-\xi)^3_+$$

is indeed a cubic regression spline.

(a) Find a cubic polynomial

$$f_1(x) = a_1 + b_1 x + c_1 x^2 + d_1 x^3$$

such that $f(x) = f_1(x)$ for all $x \le \xi$. Express a_1, b_1, c_1, and d_1 in terms of β_0, β_1, β_2, β_3, and β_4.
When $x \le \xi$, we have

$$f(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$$

since $(x-\xi)^3_+ = 0$ when $x \le \xi$. Therefore,

$$\begin{aligned} a_1 &= \beta_0 \\ b_1 &= \beta_1 \\ c_1 &= \beta_2 \\ d_1 &= \beta_3 \end{aligned}$$

(b) Find a cubic polynomial

$$f_2(x) = a_2 + b_2 x + c_2 x^2 + d_2 x^3$$

such that $f(x) = f_2(x)$ for all $x > \xi$. Express a_2, b_2, c_2, d_2 in terms of β_0, β_1, β_2, β_3, and β_4.
When $x > \xi$, we have

$$\begin{aligned} f(x) =& \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 (x^3 - 3x^2\xi + 3x\xi^2 - \xi^3) \\ =& (\beta_0 - \beta_4\xi^3) + (\beta_1 + 3\beta_4\xi^2)x + (\beta_2 - 3\beta_4\xi)x^2 + (\beta_3 + \beta_4)x^3 \end{aligned}$$

Therefore,

$$\begin{aligned} a_2 &= \beta_0 - \beta_4\xi^3 \\ b_2 &= \beta_1 + 3\beta_4\xi^2 \\ c_2 &= \beta_2 - 3\beta_4\xi \\ d_2 &= \beta_3 + \beta_4 \end{aligned}$$

We have now established that f(x) is a piecewise polynomial.

(c) Show that $f_1(\xi) = f_2(\xi)$. That is, $f(x)$ is continuous at ξ.
We have

$$f_1(\xi) = \beta_0 + \beta_1\xi + \beta_2\xi^2 + \beta_3\xi^3$$

and

$$\begin{aligned}f_2(\xi) =&(\beta_0 - \beta_4\xi^3) + (\beta_1 + 3\beta_4\xi^2)\xi + (\beta_2 - 3\beta_4\xi)\xi^2 + (\beta_3 + \beta_4)\xi^3\\ =&\beta_0 + \beta_1\xi + \beta_2\xi^2 + (-\beta_4 + 3\beta_4 - 3\beta_4 + \beta_3 + \beta_4)\xi^3\\ =&\beta_0 + \beta_1\xi + \beta_2\xi^2 + \beta_3\xi^3\end{aligned}$$

Therefore, $f_1(\xi) = f_2(\xi)$.

(d) Show that $f_1'(\xi) = f_2'(\xi)$. That is, $f'(x)$ is continuous at ξ.

We have the derivatives be

$$\begin{aligned}f_1'(x) =&\beta_1 + 2\beta_2 x + 3\beta_3 x^2\\ \Rightarrow f_1'(\xi) =&\beta_1 + 2\beta_2\xi + 3\beta_3\xi^2\\ f_2'(x) =&(\beta_1 + 3\beta_4\xi^2) + 2(\beta_2 - 3\beta_4\xi)x + 3(\beta_3 + \beta_4)x^2\\ \Rightarrow f_2'(\xi) =&(\beta_1 + 3\beta_4\xi^2) + 2(\beta_2 - 3\beta_4\xi)\xi + 3(\beta_3 + \beta_4)\xi^2\\ =&\beta_1 + 2\beta_2\xi + (3\beta_4 - 6\beta_4 + 3\beta_3 + 3\beta_4)\xi^2\\ =&\beta_1 + 2\beta_2\xi + 3\beta_3\xi^2\end{aligned}$$

Therefore, $f_1'(\xi) = f_2'(\xi)$.

(e) Show that $f_1''(\xi) = f_2''(\xi)$. That is, $f''(x)$ is continuous at ξ.

We have the second derivatives be

$$\begin{aligned}f_1''(x) =&2\beta_2 + 6\beta_3 x\\ \Rightarrow f_1''(\xi) =&2\beta_2 + 6\beta_3\xi\\ f_2''(x) =&2(\beta_2 - 3\beta_4\xi) + 6(\beta_3 + \beta_4)x\\ \Rightarrow f_2''(\xi) =&2(\beta_2 - 3\beta_4\xi) + 6(\beta_3 + \beta_4)\xi\\ =&2\beta_2 + (-6\beta_4 + 6\beta_3 + 6\beta_4)\xi\\ =&2\beta_2 + 6\beta_3\xi\end{aligned}$$

Therefore, $f_1''(\xi) = f_2''(\xi)$.

Therefore, $f(x)$ is indeed a cubic spline.

2 Problem 2 (ISL 7.10) [10 points]

This question relates to the **College** data set.

(a) Split the data into a training set and a test set. Using out-of-state tuition as the response and the other variables as the predictors, perform forward stepwise selection on the training set in order to identify a satisfactory model that uses just a subset of the predictors.

We split half of the data into the training set and the rest half into the test set, and apply forward stepwise selection to identify a satisfactory model with a subset of the predictors. The code for this part:

```r
set.seed(1)
library(ISLR)
library(leaps)
attach(College)
train = sample(length(Outstate), length(Outstate)/2)
test = -train
College.train = College[train, ]
College.test = College[test, ]
reg.fit = regsubsets(Outstate ~ ., data = College.train, nvmax = 17, method = "forward")
reg.summary = summary(reg.fit)

png("2a.png", width = 900, height = 350, res = 130)
par(mfrow = c(1, 3), mai = c(0.6, 0.6, 0.1, 0.1))
plot(reg.summary$cp, xlab = "Number of Variables", ylab = "Cp", type = "l",
     ylim = c(min(reg.summary$cp) - 0.2 * sd(reg.summary$cp), max(reg.summary$cp)))
min.cp = min(reg.summary$cp)
```

```
std.cp = sd(reg.summary$cp)
abline(h = min.cp + 0.2 * std.cp, col = "red", lty = 2)
abline(h = min.cp - 0.2 * std.cp, col = "red", lty = 2)
segments(x0 = 6, x1 = 6, y0 = -100, y1 = reg.summary$cp[6], lty = 2)

plot(reg.summary$bic, xlab = "Number of Variables", ylab = "BIC", type = "l",
     ylim = c(min(reg.summary$bic) - 0.2 * sd(reg.summary$bic), max(reg.summary$bic)))
min.bic = min(reg.summary$bic)
std.bic = sd(reg.summary$bic)
abline(h = min.bic + 0.2 * std.bic, col = "red", lty = 2)
abline(h = min.bic - 0.2 * std.bic, col = "red", lty = 2)
segments(x0 = 6, x1 = 6, y0 = -600, y1 = reg.summary$bic[6], lty = 2)

plot(reg.summary$adjr2, xlab = "Number of Variables", ylab = "Adjusted R2",
    type = "l", ylim = c(0.4, 0.84))
max.adjr2 = max(reg.summary$adjr2)
std.adjr2 = sd(reg.summary$adjr2)
abline(h = max.adjr2 + 0.2 * std.adjr2, col = "red", lty = 2)
abline(h = max.adjr2 - 0.2 * std.adjr2, col = "red", lty = 2)
segments(x0 = 6, x1 = 6, y0 = 0.3, y1 = reg.summary$adjr2[6], lty = 2)
dev.off()

reg.fit = regsubsets(Outstate ~ ., data = College, method = "forward")
coefi = coef(reg.fit, id = 6)
names(coefi)
```

The Mallow's C_p statistic, BIC and adjusted R^2 given by different number of variables are presented in Figure 1, where we realize that the model of size 6 is the minimum size for which the scores are within 0.2 standard deviations of optimum. Therefore, we will pick 6 as the best subset size and find the best 6 variables using the entire data set.

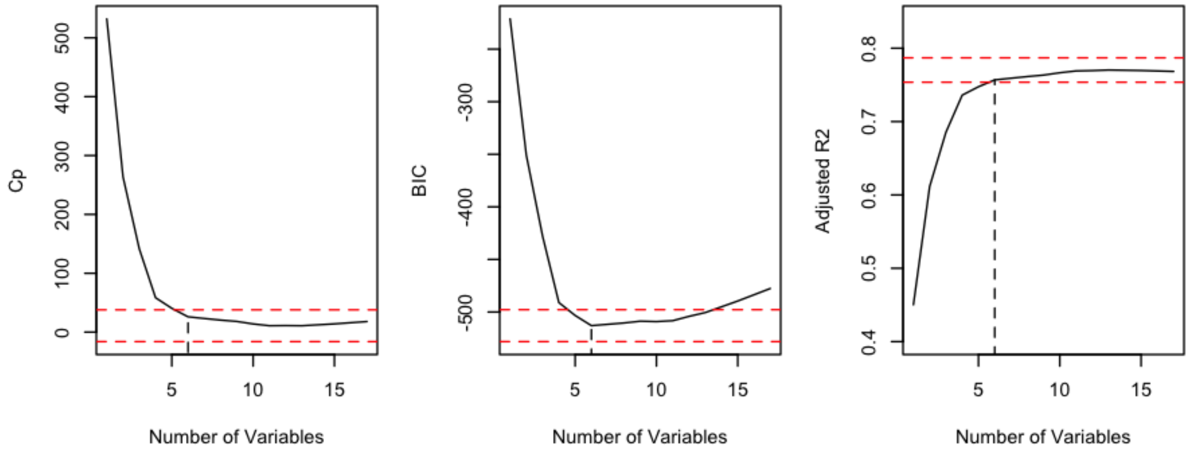
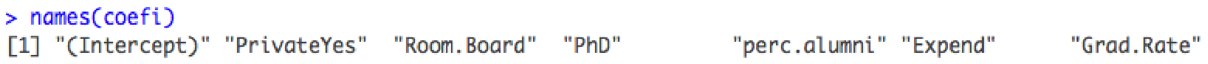


Figure 1: Mallow's C_p, BIC and adjusted R^2 in Problem 2 Part (a)



```
> names(coefi)
[1] "(Intercept)" "PrivateYes"   "Room.Board"  "PhD"          "perc.alumni" "Expend"       "Grad.Rate"
```

Figure 2: Variables found in Problem 2 Part (a)

The variables found here is presented in Figure 2.

(b) Fit a GAM on the training data, using out-of-state tuition as the response and the features selected in the previous step as the predictors. Plot the results, and explain your findings.

Except that `PrivateYes` is a categorical variable where we will not add splines, we give 2 degrees of freedom to each of the rest variables for smoothing splines. The code for this part:

```
library(gam)
gam.fit = gam(Outstate ~ Private + s(Room.Board, df = 2) + s(PhD, df = 2) +
    s(perc.alumni, df = 2) + s(Expend, df = 2) + s(Grad.Rate, df = 2), data = College.train)
```

```

png("2b.png", height = 600, width = 900, res = 125)
par(mfrow = c(2, 3), mai = c(0.6, 0.6, 0.1, 0.1))
plot(gam.fit, se = T, col = "blue")
dev.off()
```

The results for the fitted model are presented in Figure 3, where we find that

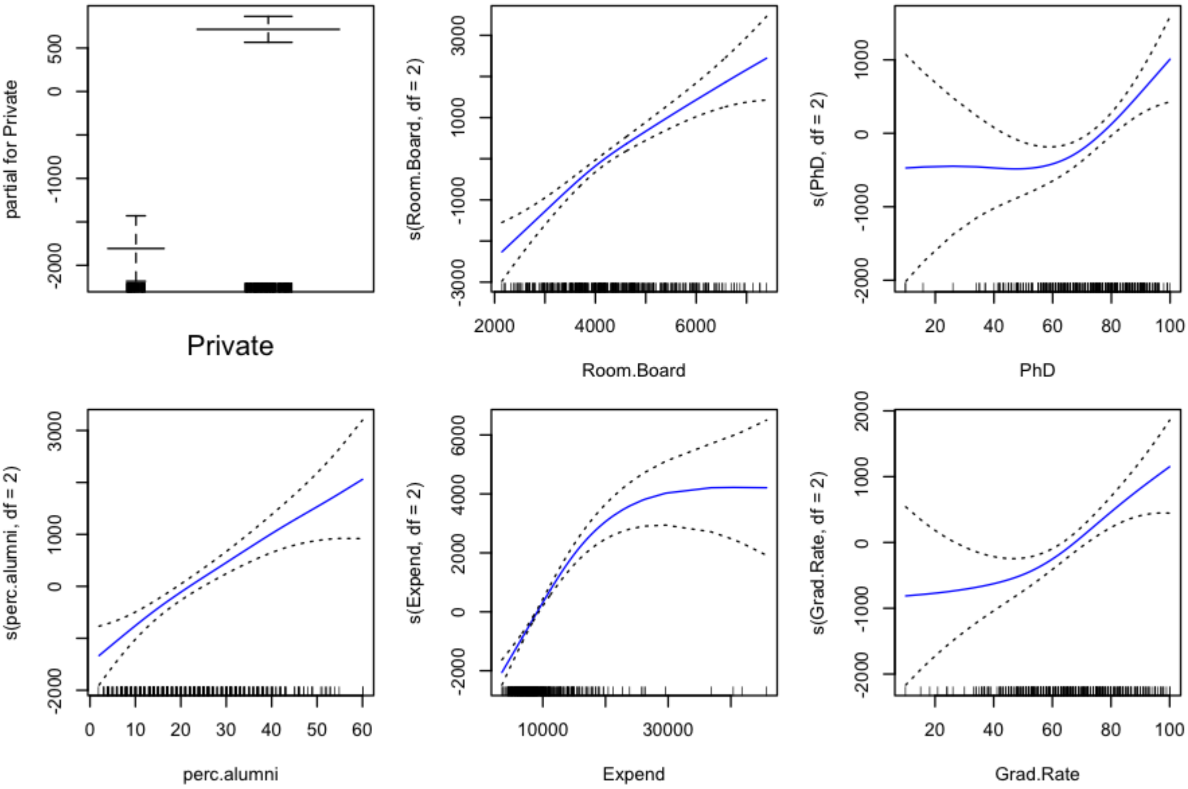


Figure 3: Results for the fitted model in Problem 2 Part (b)

i. The relationship between out-of-state tuition and room and board costs, percentage of alumni who donate is almost linear even though more than 1 degree of freedom is given to these 2 predictors.

ii. Although the relationship between out-of-state tuition and percentage of faculty with Ph.D.'s, instructional expenditure per student, graduation rate seem non-linear, we should also be cautious that the non-linear relationship comes from a small number of data points, or in other words, there are a small number of data points bending the regression line towards them. The wider confidence interval around those areas also remind us to be cautious about the inference there.

(c) Evaluate the model obtained on the test set, and explain the results obtained.

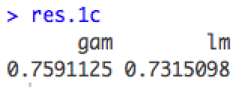
We evaluate the model by comparing the test R^2 obtained from linear regression and GAM:

```
gam.pred = predict(gam.fit, College.test)
gam.err = mean((College.test$Outstate - gam.pred)^2)

gam.tss = mean((College.test$Outstate - mean(College.test$Outstate))^2)
test.rss = 1 - gam.err/gam.tss

lm.2c <- lm(Outstate ~ Private + Room.Board + PhD + perc.alumni + Expend + Grad.Rate, data = College.train)
lm.pred <- predict(lm.2c, College.test)
lm.err <- mean((College.test$Outstate - lm.pred)^2)
test.lm.rss <- 1 - lm.err / gam.tss
res.1c <- c(test.rss, test.lm.rss)
names(res.1c) <- c("gam", "lm")
```

```
res.1c
```



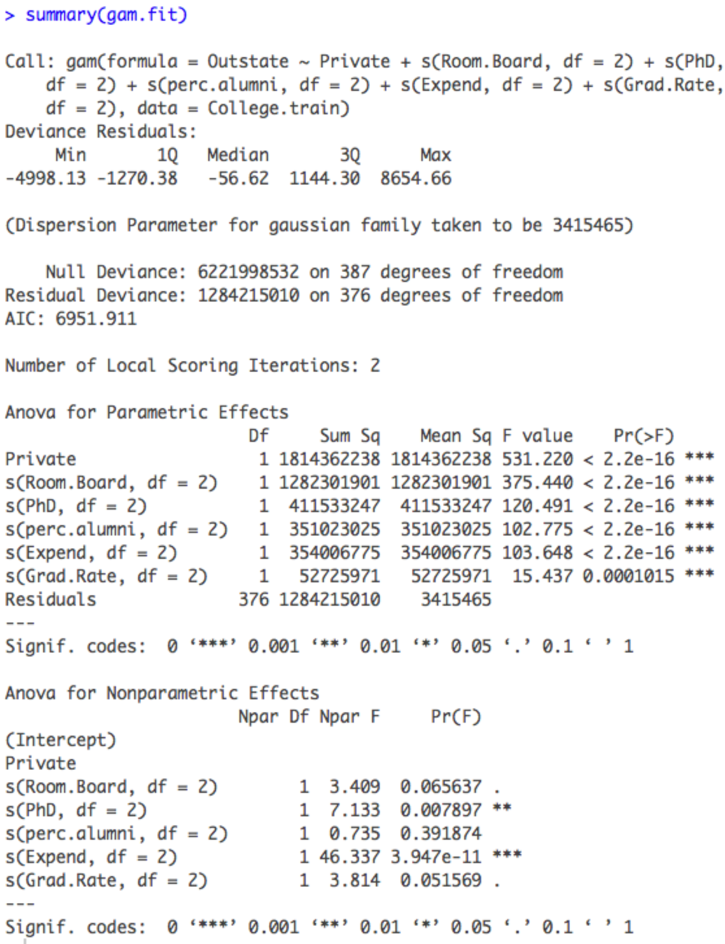
```
> res.1c
      gam        lm 
0.7591125 0.7315098 
```

Figure 4: R^2 on the test set in Problem 2 Part (c)

In Figure 4, where we compared the R^2 on the test given by linear regression model and GAM, We obtain a test R^2 of 0.77 using GAM, which is a slight improvement over a test R^2 of 0.74 obtained using linear regression.

(d) For which variables, if any, is there evidence of a non-linear relationship with the response?

```
summary(gam.fit)
```



```
> summary(gam.fit)

Call: gam(formula = Outstate ~ Private + s(Room.Board, df = 2) + s(PhD, 
    df = 2) + s(perc.alumni, df = 2) + s(Expend, df = 2) + s(Grad.Rate, 
    df = 2), data = College.train)
Deviance Residuals:
     Min       1Q   Median       3Q      Max 
-4998.13 -1270.38   -56.62  1144.30  8654.66 

(Dispersion Parameter for gaussian family taken to be 3415465)

    Null Deviance: 6221998532 on 387 degrees of freedom
Residual Deviance: 1284215010 on 376 degrees of freedom
AIC: 6951.911 

Number of Local Scoring Iterations: 2 

Anova for Parametric Effects
                        Df     Sum Sq    Mean Sq F value    Pr(>F)    
Private                  1 1814362238 1814362238 531.220 < 2.2e-16 ***
s(Room.Board, df = 2)    1 1282301901 1282301901 375.440 < 2.2e-16 ***
s(PhD, df = 2)           1  411533247  411533247 120.491 < 2.2e-16 ***
s(perc.alumni, df = 2)   1  351023025  351023025 102.775 < 2.2e-16 ***
s(Expend, df = 2)        1  354006775  354006775 103.648 < 2.2e-16 ***
s(Grad.Rate, df = 2)     1   52725971   52725971  15.437 0.0001015 ***
Residuals              376 1284215010    3415465                      
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Anova for Nonparametric Effects
                       Npar Df Npar F     Pr(F)    
(Intercept)                                        
Private                                            
s(Room.Board, df = 2)        1  3.409  0.065637 .  
s(PhD, df = 2)               1  7.133  0.007897 ** 
s(perc.alumni, df = 2)       1  0.735  0.391874    
s(Expend, df = 2)            1 46.337 3.947e-11 ***
s(Grad.Rate, df = 2)         1  3.814  0.051569 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 5: Summary of GAM in Problem 2 Part (d)

From the model summary in Figure 5, non-parametric Anova test shows a strong evidence of non-linear relationship between response and Expend or PhD, and a moderately strong non-linear relationship (using p value of 0.05) between response and Room.Board or Grad.Rate. However, we still need to be cautious as we have seen in part (b) that there are a small number of points pulling the regression line towards them leading to the non-linear relationship.

3 Problem 3 (Based on GH 12.1) [3 points]

Using data of your own (possibly imaginary or based on previous data sets seen in the class) that are appropriate for a multilevel model, write the model in the five ways discussed in Section 12.5 and give an interpretation of the model.

Suppose that we impose another group level, i.e. hospitals, on the pain data set, so that the patients are nested within the hospitals, and the pain measurements taken on the same individual nested within the individuals. Also suppose that we have hospital-level factors, number of admissions per month, number of beds, etc. If we have following model,

$$\begin{aligned}
y_{ijk} =& \alpha_{jk} + \boldsymbol{\beta}_j^T \mathbf{x}_{ijk} + \epsilon_{ijk} & \epsilon_{ijk} \sim N(0, \sigma_\epsilon^2)\\
=& \alpha_{jk} + \beta_{j,1} x_{ijk,1} + \beta_2 x_{ijk,2} + \beta_3 x_{ijk,3} + \cdots + \epsilon_{ijk} & \\
\alpha_{jk} =& a_{0k} + \mathbf{a}^T \mathbf{w}_{jk} + \eta_{jk} & \eta_{jk} \sim N(0, \sigma_\eta^2)\\
=& a_{0k} + a_1 w_{jk,1} + a_2 w_{jk,2} + \cdots + \eta_{jk} & \\
\beta_{j,1} =& b_0 + \mathbf{b}^T \mathbf{w}_{jk} + \gamma_{jk} & \gamma_{jk} \sim N(0, \sigma_\gamma^2)\\
a_{0k} =& c_0 + \mathbf{c}^T \mathbf{u}_k + \xi_k & \xi_k \sim N(0, \sigma_\xi^2)
\end{aligned}$$

with notations:

- y_{ijk}: pain level measured at time i on patient j in hospital k;
- α_{jk}: intercept for patient j in hospital k which may depend on second (patient)/third (hospital) level predictors;
- $\mathbf{x}_{ijk}$: vector for first level predictors (i.e. time i/temperature i when the pain level measurement is taken on patient j in hospital k);
- $\boldsymbol{\beta}_j$: coefficients for the first level predictors for patient j;
  - $\beta_{j,1}$: coefficient for first level predictor x_{ijk} for patient j;
  - β_2: coefficient for the first level predictors common for all the patients;
- a_{0k}: intercept for hospital k when all the second level predictors are 0 which may depend on third (hospital) level predictors;
- a_1: coefficient for second level predictors common for all hospitals;
- $\mathbf{w}_{jk}$: vector for second (patient) level predictors (i.e. age, sex, race, etc. for patient j in hospital k);
- b_0: slope for $x_{ijk,1}$ when all the second level predictors are 0 common for all hospitals;
- c_0; grand average intercept for pain level prediction;
- $\mathbf{u}_k$: vector for third (hospital) level predictors (i.e. number of admissions per month, number of beds, etc. for hospital k).

In the above model, we assume that:

- Pain level measurements will depend on time and temperature in the first level;
- The intercept will vary by patient and hospital depending on the second and third level predictors with fixed relationship;
- One of the coefficients of the first level predictors, i.e. time, will vary by patient depending on second level predictors.

The 5 model forms will be:

(a) *Allowing regression coefficients to vary across groups:*

$$\begin{aligned}
y_i =& \alpha_{jk[i]} + b_{j[i]} x_{i,1} + \mathbf{b}^T \mathbf{w}_i \cdot x_{i,1} + \beta_2 x_{i,2} + \beta_3 x_{i,3} + & \\
& \cdots + \mathbf{a}^T \mathbf{w}_i + \mathbf{c}^T \mathbf{u}_i + \epsilon_i & \epsilon \sim N(0, \sigma_\epsilon^2)\\
\alpha_{jk} =& c_0 + \eta_{jk} + \xi_k & \eta_{jk} \sim N(0, \sigma_\eta^2)\\
& & \xi_k \sim N(0, \sigma_\xi^2)\\
b_j =& b_0 + \gamma_{jk} & \gamma_{jk} \sim N(0, \sigma_\gamma^2)
\end{aligned}$$

where $\mathbf{x}_i$ are the first level predictors, $\mathbf{w}_i$ are the second (patient) level predictors and $\mathbf{u}_i$ are the third (hospital) level predictors.

(b) *Combining separate local regressions:*

Within patient j in hospital k:

$$y_i \sim N(\alpha_{jk} + \beta_{j,1}x_{i,1} + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \cdots, \sigma_\epsilon^2)$$

for $i = 1, \ldots, n_{jk}$ with $\mathbf{x}_i$ being the first level predictors.

The second (patient) level model:

$$\begin{aligned} \alpha_j &\sim N(a_{0k} + \mathbf{a}^T \mathbf{w}_j, \sigma_\eta^2) \\ \beta_{j,1} &\sim N(b_0 + \mathbf{b}^T \mathbf{w}_j, \sigma_\gamma^2) \end{aligned}$$

where $\mathbf{w}_j$ are the second (patient) level predictors.

The third (hospital) level model:

$$a_{0k} \sim N(c_0 + \mathbf{c}^T \mathbf{u}_k, \sigma_\xi^2)$$

where $\mathbf{u}_k$ are the third (hospital) level predictors.

(c) *Modeling the coefficients of a large regression model:*

$$y_i \sim N(\mathbf{X}_i \boldsymbol{\beta}, \sigma_\epsilon^2)$$

where $\mathbf{X}$ includes vectors corresponding to

- $\mathbf{c}_0$: a constant term;
- $\mathbf{W}\mathbf{x}_1$: interaction between $\mathbf{x}_1$ and second (patient) level predictors;
- $\mathbf{X}_{2,\ldots}$: first level predictors;
- $\mathbf{W}$: second (patient) level predictors;
- $\mathbf{U}$: third (hospital) level predictors.

at the second level of the model, assuming we have equal number, J, of patients in K hospitals, we have

- Interaction between $\mathbf{x}_1$ and patient indicators: the corresponding J coefficients to the number of patients follow a normal distribution,

$$\beta_l \sim N(b_0, \sigma_\gamma^2)$$

and at the third level of the model, we have

- The corresponding K coefficients to the number of hospitals follow a normal distribution:

$$\beta_l \sim N(0, \sigma_\xi^2)$$

- The corresponding J coefficients to the interaction between patient and each hospital k indicators follow a normal distribution:

$$\beta_l \sim N(0, \sigma_\eta^2)$$

(d) *Regression with multiple error terms:*

$$\begin{aligned} y_i &\sim N(\mathbf{X}_i \boldsymbol{\beta} + (b_0 + \gamma_{j[i]})x_{i,1} + \eta_{jk[i]} + \xi_{k[i]}, \sigma_\epsilon^2) \\ \gamma_j &\sim N(0, \sigma_\gamma^2) \\ \eta_{jk} &\sim N(0, \sigma_\eta^2) \\ \xi_k &\sim N(0, \sigma_\xi^2) \end{aligned}$$

where $\mathbf{X}$ includes vectors corresponding to

- $\mathbf{c}_0$: a constant term;
- $\mathbf{W}\mathbf{x}_1$: interaction between $\mathbf{x}_1$ and second (patient) level predictors;
- $\mathbf{X}_{2,...}$: first level predictors;
- $\mathbf{W}$: second (patient) level predictors;
- $\mathbf{U}$: third (hospital) level predictors.

and $j[i]$ represents the patient that contains pain level measurements i; $jk[i]$ represents the pain level measurements i nested within patient j and hospital k; $k[i]$ represents the measurement i taken in hospital k.

(e) *Large regression with correlated errors:*

$$\begin{aligned} y_i =& \mathbf{X}_i \boldsymbol{\beta} + (b_0 + \gamma_{j[i]}) x_{i,1} + \epsilon_i^{\text{all}}, \qquad \boldsymbol{\epsilon}^{\text{all}} \sim N(0, \boldsymbol{\Sigma}) \\ \gamma_j \sim& N(0, \sigma_\gamma^2) \end{aligned}$$

where we will have $\mathbf{X}$ be the same as that in the previous form of model. We will break $\boldsymbol{\Sigma}$ into parts for expression,

- Within patient j,

$$\boldsymbol{\Sigma}_{jj} = \begin{pmatrix} \sigma_\epsilon^2 + \sigma_\eta^2 + \sigma_\xi^2 & \sigma_\eta^2 + \sigma_\xi^2 & \cdots & \sigma_\eta^2 + \sigma_\xi^2 \\ \sigma_\eta^2 + \sigma_\xi^2 & \sigma_\epsilon^2 + \sigma_\eta^2 + \sigma_\xi^2 & \cdots & \sigma_\eta^2 + \sigma_\xi^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_\eta^2 + \sigma_\xi^2 & \sigma_\eta^2 + \sigma_\xi^2 & \cdots & \sigma_\epsilon^2 + \sigma_\eta^2 + \sigma_\xi^2 \end{pmatrix}$$

- For patient j and l in the same hospital k,

$$\boldsymbol{\Sigma}_{jl} = \begin{pmatrix} \sigma_\xi^2 & \sigma_\xi^2 & \cdots & \sigma_\xi^2 \\ \sigma_\xi^2 & \sigma_\xi^2 & \cdots & \sigma_\xi^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_\xi^2 & \sigma_\xi^2 & \cdots & \sigma_\xi^2 \end{pmatrix}$$

- For patient j and l in different hospitals,

$$\boldsymbol{\Sigma}_{jl} = 0$$

4 Problem 4 (GH 12.9)

In this exercise, you will explore the relationship between the number of observations and number of groups on the performance of a multilevel model.

(a) Take a simple random sample of one-fifth of the radon data seen in class. Fit the varying-intercept model with floor as an individual-level predictor and log uranium as a county-level predictor, and compare your inferences to what was obtained by fitting the model to the entire dataset. (Compare inferences for the individual-level and group-level standard deviations, the slopes for floor and log uranium, the average intercept, and the county-level intercepts.)

The model we will be fitting:

$$\begin{aligned} \log(\text{radon})_{ij} =& \beta_{0j} + \beta_1 \text{floor}_{ij} + \epsilon_{ij} \\ \beta_{0j} =& \gamma_0 + \gamma_1 \log(\text{uranium})_j + \eta_j \\ \epsilon_{ij} \sim& N(0, \sigma_\epsilon^2) \\ \eta_j \sim& N(0, \sigma_\eta^2) \end{aligned}$$

The code for this part:

```
library(dplyr)
library(lme4)
library(arm)
srrs2    <- read.table ("../../LectureNotes/srrs2.dat", header=T, sep=",")
```

```
srrs2.mn <- srrs2[srrs2$state2 == "MN",]
srrs2.mn <- srrs2.mn %>%
  mutate(fips = stfips * 1000 + cntyfips)

cty <- read.table ("../../LectureNotes/cty.dat", header=T, sep=",")
cty <- cty %>%
  mutate(fips = 1000 * stfips + ctfips)
cty.mn <- cty %>%
  filter(fips %in% srrs2.mn$fips)
cty.mn <- cty.mn[!duplicated(cty.mn$fips),]

# remove the rows whose county is not in cty
srrs2.mn <- srrs2.mn %>%
  filter(fips %in% cty.mn$fips)
srrs2.mn <- srrs2.mn %>%
  left_join(cty.mn, by = "fips")

# Select only the used columns
srrs2.mn <- srrs2.mn %>%
  select(county, Uppm, activity, floor)

srrs2.mn <- srrs2.mn %>%
  mutate(log.radon = ifelse(activity == 0, log(0.01), log(activity))) %>%
  mutate(log.u     = log(Uppm))

# Subset 1/5 of the data set
srrs2.mn.4a <- srrs2.mn[sample.int(n = dim(srrs2.mn)[1],
                                   size = floor(dim(srrs2.mn)[1] / 5),
                                   replace = F),]

fit.sub.4a  <- lmer(log.radon ~ floor + log.u + (1|county), data = srrs2.mn.4a)
fit.full.4a <- lmer(log.radon ~ floor + log.u + (1|county), data = srrs2.mn)

display(fit.sub.4a)
display(fit.full.4a)

png("4a.png", width = 600, height = 450, res = 100)
par(mai = c(0.8, 0.8, 0.1, 0.1))
plot(coef(fit.sub.4a)$county[,1], type = "l", lwd = 2, col = "pink",
     ylab = "County-level Intercepts", xlab = "Counties in the Subset")
tmp <- rownames(coef(fit.full.4a)$county) %in%
  rownames(coef(fit.sub.4a)$county)
lines(coef(fit.full.4a)$county[tmp,1], lty = 2, lwd = 2, col = "yellowgreen")
legend("bottomleft", c("Subset", "Entire"),
       lty = 1:2,
       col = c("pink", "yellowgreen"),
       lwd = 2)
dev.off()
```

The comparison of the inferences of the model fitted to the subset and the model fitted to the entire dataset is presented in Table 1 and Figure 6, where we find

- The standard error of the coefficient estimate for floor, log uranium level and average intercept given by the model fitted on the entire data set is smaller than that given by the other model since we have a larger data set; similar result could also be found in Figure 6, where the county-level intercepts are less variable given by the entire dataset model than those given by the subset model.
- The unexplained variance in the response, log radon level, are almost the same given by the 2 data sets.
- The estimates for the coefficients are reasonably different or almost the same, given the decreased accuracy by the smaller data set.

(b) Repeat step (a) a few times, with a different random sample each time, and summarize how the estimates vary.

Here, we repeat step (a) 1,000 times, and compare the estimates compared in part (a) to those given by the model fitted to the entire data set.

The code for this part:

```
sigma.eps <- NULL
sigma.eta <- NULL
beta.1    <- NULL
gamma.1   <- NULL
```

	Model Representation	1/5 of the dataset	Entire dataset	1/5 of the counties
Individual s.d.	σ_ϵ	0.84	0.82	0.65
Group s.d.	σ_η	0.22	0.17	0.29
Floor	β_1 (s.e.)	-0.09 (0.08)	-0.48 (0.05)	-0.54 (0.18)
log(Uranium)	γ_1 (s.e.)	0.75 (0.18)	0.75 (0.09)	0.78 (0.23)
Average Intercept	γ_0 (s.e.)	1.35 (0.08)	1.39 (0.04)	1.40 (0.10)

Table 1: Comparing inferences of the 2 models in Problem 4 Part (a)

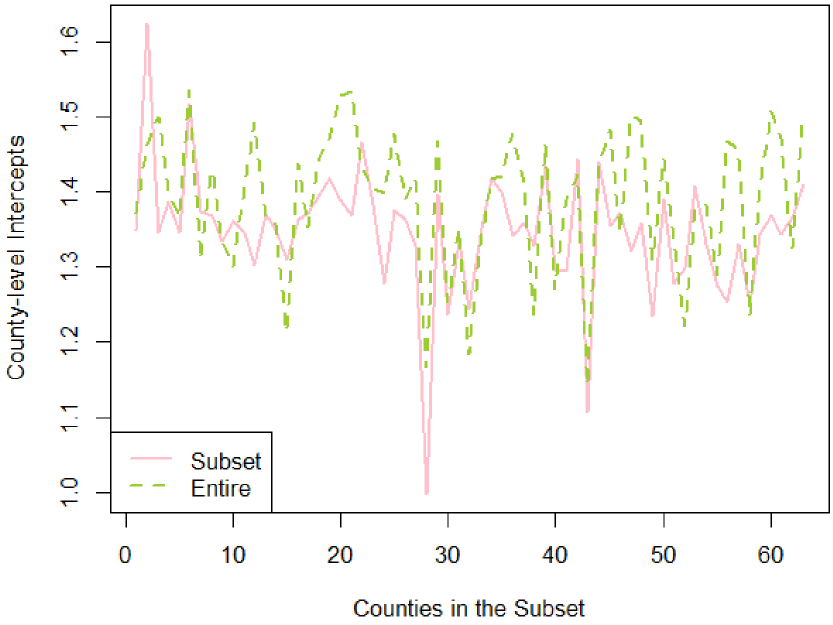


Figure 6: Comparing county-level intercepts from the models in Problem 4 Part (a) and (c)

```
gamma.0   <- NULL

R <- 1000
cty.int   <- data.frame(full = coef(fit.full.4a)$county[, 1],
                        matrix(NA, nrow = length(unique(srrs2.mn$county)),
                               ncol = R))
row.names(cty.int) <- rownames(coef(fit.full.4a)$county)

for (i in 1:R){

  srrs2.mn.4b <- srrs2.mn[sample.int(n = dim(srrs2.mn)[1],
                                     size = floor(dim(srrs2.mn)[1] / 5),
                                     replace = F),]
  fit.sub.4b  <- lmer(log.radon ~ floor + log.u + (1|county), data = srrs2.mn.4b)
  summ.4b <- summary(fit.sub.4b)

  sigma.eps <- c(sigma.eps, as.data.frame(summ.4b$varcor)[2, 5])
  sigma.eta <- c(sigma.eta, as.data.frame(summ.4b$varcor)[1, 5])

  tmp <- fixef(fit.sub.4b)
  beta.1  <- c(beta.1, tmp[2])
  gamma.1 <- c(gamma.1, tmp[3])
  gamma.0 <- c(gamma.0, tmp[1])

  cty.int[rownames(cty.int) %in% rownames(coef(fit.sub.4b)$county), i+1] <-
    coef(fit.sub.4b)$county[, 1]

}

png("4b.png", width = 1000, height = 700, res = 120)
par(mfrow = c(2, 3), mai = c(0.6, 0.6, 0.4, 0.1))
hist(sigma.eps)
abline(v = as.data.frame(summary(fit.full.4a)$varcor)[2, 5],
       lty = 2, lwd = 2)
```

```
hist(sigma.eta)
abline(v = as.data.frame(summary(fit.full.4a)$varcor)[1, 5],
       lty = 2, lwd = 2)
hist(beta.1)
abline(v = fixef(fit.full.4a)[2],
       lty = 2, lwd = 2)
hist(gamma.1)
abline(v = fixef(fit.full.4a)[3],
       lty = 2, lwd = 2)
hist(gamma.0)
abline(v = fixef(fit.full.4a)[1],
       lty = 2, lwd = 2)

cty.int.figure <- t(apply(cty.int[2:dim(cty.int)[2]], 1, quantile,
                          probs = c(0.025, 0.975), na.rm = T))
plot(cty.int.figure[,1], type = "l",
     lty = 2, lwd = 2, col = "grey",
     ylim = range(cty.int.figure), ylab = "County-level Intercepts",
     main = "Compare County-level Intercepts")
lines(cty.int[, 1], lwd = 2)
lines(cty.int.figure[, 2], lwd = 2, lty = 2, col = "grey")
legend("bottomright",
       c("Entire", "2.5%/97.5% Quantiles"),
       col = c("black", "grey"),
       lty = 1:2,
       lwd = 2)
dev.off()
```

Figure 7 presents how the estimates given by the replications compare to those given by the model fitted on the entire data set, where the black dashed lines indicates the estimates obtained from the entire data set in the histograms.

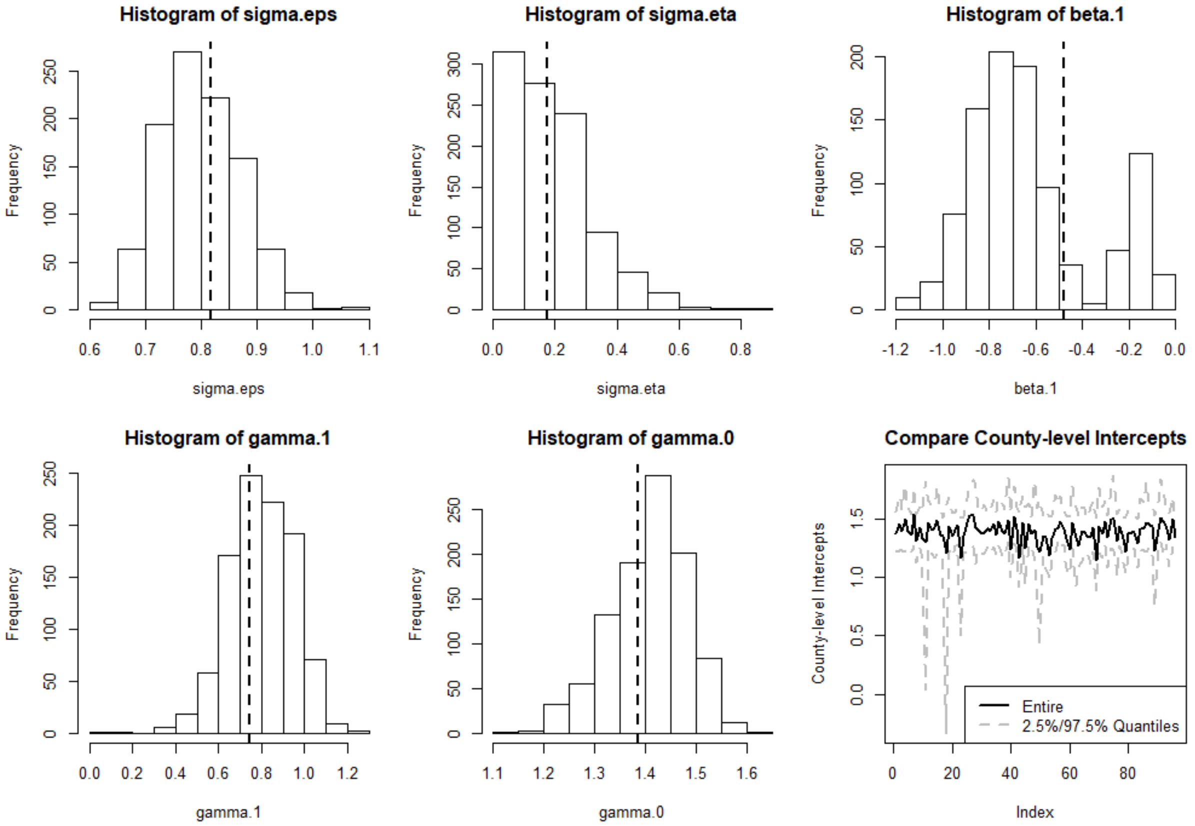


Figure 7: Comparing estimates to those obtained from the entire data set in Problem 4 Part (b)

In Figure 7, we find that almost all the estimates have a reasonable distribution around the estimates given by the entire data set, except that β_1, the coefficient for floor, has a bimodal distribution, indicating that there might be subgroups in the dataset have different effect of floor level on the radon activity from the rest.

(c) Repeat step (a), but this time taking a cluster sample: a random sample of one-fifth of the counties, but then all the houses within each sampled country.

The code for this part:

```
cty.4c      <- as.vector(unique(srrs2.mn$county))
cty.4c      <- sample(cty.4c, floor(length(cty.4c) / 5), replace = F)
srrs2.mn.4c <- srrs2.mn %>%
  filter(county %in% cty.4c)

fit.sub.4c  <- lmer(log.radon ~ floor + log.u + (1|county), data = srrs2.mn.4c)
display(fit.sub.4c)
display(fit.full.4a)

png("4c.png", width = 600, height = 450, res = 100)
par(mai = c(0.8, 0.8, 0.1, 0.1))
plot(coef(fit.sub.4c)$county[,1], type = "l", lwd = 2, col = "pink",
     ylab = "County-level Intercepts", xlab = "Counties in the Subset")
tmp <- rownames(coef(fit.full.4a)$county) %in%
  rownames(coef(fit.sub.4c)$county)
lines(coef(fit.full.4a)$county[tmp,1], lty = 2, lwd = 2, col = "yellowgreen")
legend("bottomright", c("Subset", "Entire"),
       lty = 1:2,
       col = c("pink", "yellowgreen"),
       lwd = 2)
dev.off()
```

The comparison of the inferences of the model fitted to the subset and the model fitted to the entire dataset is presented in Table 1 in part (a) and Figure 8, where we find here

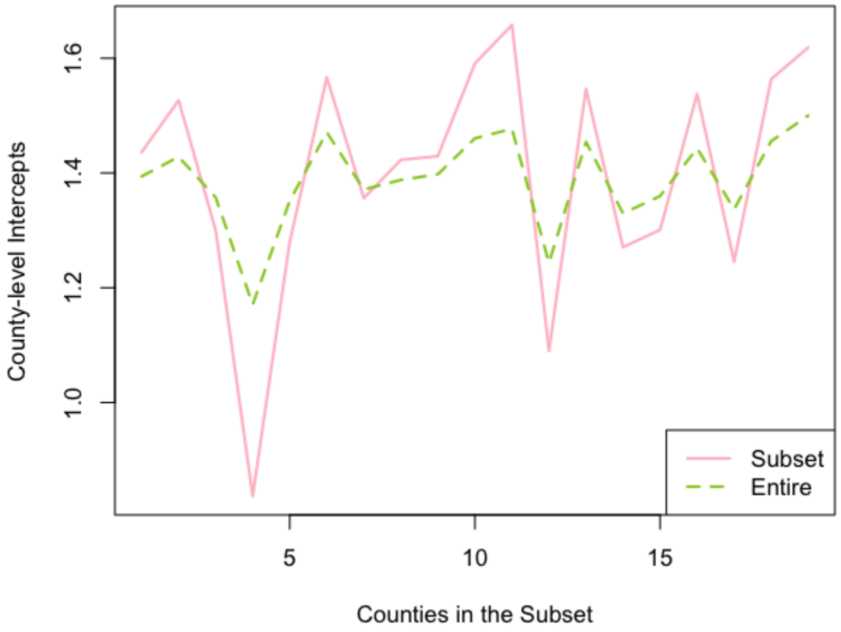


Figure 8: Comparing county-level intercepts from the 2 models in Problem 4 Part (c)

i. Given fewer counties, the model could fit closer to each county's data; in other words, there is weaker pooling effect so the unexplained variability in the outcome, σ_ϵ^2, decreases, and across group variability, σ_η^2, increases.

ii. The estimates for the coefficients are reasonably different or almost the same, given the decreased accuracy by the even smaller data set compared to that in part (a).

iii. In Figure 8, we also find that the county-level intercepts are less variable given by the entire dataset model than those given by the subset model.

Problem 5 [10 points]

In this exercise, you will fit a multilevel model in which within patient, pain is a function of time and temperature and between patients these relationships may depend on age, race, income, treatment, sex, occupation, working status and use of NSAIDs. Use the lme or lmer function to fit a multilevel model in which

$$Y_i = a_{j[i]} + b_{j[i]} X_i + \epsilon_i$$

$$a_j = g_0 + g_1 z_j + {\eta^1}_j$$

$$b_j = h_0 + h_1 z_j + {\eta^2}_j$$

Here, y_i is the pain of observation i at time x_i for individual $j[i]$. Further, z_i is patient level factors (age, sex, race, income, occupation, working status, use of NSAIDs).

Build a multivariable regression model treating a_i and b_i as random effects. Make sure to give a thorough writeup of your results with appropriate graphs and tables. Code should be put at the end of the assignment.

In this dataset,we are clustering on patient group (there are 205 different patient groups) and the individual level is a patient's pain at a specific time ($Pain_{it}$).

To fit the above multilevel model, the first level would be a linear regression of pain for patient i at time t explained by either time or temperature (X_{it}); however, the intercept and slopes of this regression vary based on patient group and a patient level predictor. The random intercept (a_i) and random slope (b_i) depend on a patient level characteristic (z_i) (age, sex, gender, . . .).

If we look at the Level 1 and Level 2 models above, we can translate this hiearchical model into a mixed effect model (which is necessary for the use of lmer()).

$$\begin{aligned} Y_{it} &= g_0 + g_1 z_i + u_i + (h_0 + h_1 z_i + w_i) X_{it} + e_{it} \\ &= g_0 + g_1 z_i + h_0 X_{it} + h_1 z_i X_{it} + u_i + w_i X_{it} + e_{it} \\ &= (g_0 + u_i) + g_1 z_i + (h_0 + w_i) X_{it} + h_1 z_i X_{it} + e_{it} \end{aligned}$$

Notes

Thus, from the above equation to fit this specific multilevel model in R we will need predictors z_i, X_{it} and the interaction term $z_i X_{it}$ and to denote that the intercept and slope are varying based on the patient group.

Using the lmer function (lme4 package):

Random Intercept and Random Slope with Predictor z at Level 2: lmer(pain~x + z + x*z + (1+x|ID))

Also we need to set REML to be false in order to compare these models.

Data Cleaning

For this multilevel model we are considering regressing pain on time or temperature and fitting the intercept and slope based on patient level characteristics such as age, race, income, treatment, sex, occupation, working status, and use of NSAIDs.

However, first we needed to clean up the dataset and pull out the predictors of interest. First we pulled out the temperature measurements on the days during which pain score was recorded and put those in a datatset. However, we removed all the observations for which the temperature data was missing because this indicated that the date variable was likely missing too. Finally, we recentered the Time variable so that the time on the first day of pain observation for each individual is 0.

For the potential patient level characteristics we considered using age, income (as a categorical variable), treatment, sex, use of NSAID, working status (retired), and race.

Since there were about 100 different occupations in the dataset, we did not consider using occupation as a potential predictor and instead we made a variable for retired (yes/no). Additionally, since the proportion of those who are White is highest we recoded the race variable as White/NonWhite.

Below we can see a summary of the outcome variable and potential predictors considered in our multilevel model. We note that variables income (categorical) and retired both have NAs.

```
##       pain             time        temperature        age
##  Min.   : 0.000   Min.   : 0.00   Min.   :-5.0   Min.   :44.00
##  1st Qu.: 5.000   1st Qu.:14.00   1st Qu.:46.0   1st Qu.:53.00
##  Median : 7.000   Median :42.00   Median :60.0   Median :59.50
##  Mean   : 7.517   Mean   :39.77   Mean   :57.7   Mean   :60.24
##  3rd Qu.:10.000   3rd Qu.:70.00   3rd Qu.:72.0   3rd Qu.:67.00
##  Max.   :20.000   Max.   :86.00   Max.   :95.0   Max.   :98.00
##
##   inccat    treat   nsaid   retired    female  white
##  1   : 84   0:569   0:228   0   :475   0:384   0: 113
##  2   :194   1:551   1:892   1   :399   1:736   1:1007
##  3   :146                   NA's:246
##  4   : 57
##  5   : 63
##  NA's:576
##
```

Exploration of Potential Models

First we consider Level 1 of the multilevel model and whether we want to use time or temperature as a potential predictor (X_{it}) of pain at the individual level. To determine which model is best, we will fit the following 4 models and compare them using anova.

Recall Level 1: $Y_{it} = a_i + b_i X_{it} + e_{it}$

Model.0 (Random intercept)

$Pain_{it} = a_i + e_{it}$
$a_i = g_0 + u_i$

Model.temp (Random intercept, Random Slope)

$Pain_{it} = a_i + b_i Time_{it} + e_{it}$
$a_i = g_0 + u_i$
$b_i = h_0 + w_i$

Model.time (Random intercept, Random Slope)

$Pain_{it} = a_i + b_i Temperature_{it} + e_{it}$
$a_i = g_0 + u_i$
$b_i = h_0 + w_i$

Model.both (Random intercept, Random Slopes)

$Pain_{it} = a_i + b_i Time_{it} + c_i Temperature_{it} + e_{it}$
$a_i = g_0 + u_i$
$b_i = h_0 + w_i$
$c_i = k_0 + m_i$

Above Temperature refers to the scaled temperature variable (since our models with the unscaled temperature did not converge).

Finally, we compared our 4 models by AIC and BIC (Table 1). We see from the output that Model.temp and Model.time are significantly better than just the intercept model but Model.both is not significant (by the chisquare pvalues). To choose between Model.temp and Model.time, we note that the AIC and BIC are lower for Model.time and conclude that our Model.time, the model with Time as the predictor is best.

Thus, we will now consider which patient characteristics to use as predictors for the Level 2 Model building off our Level 1 model with Time as the predictor.

Table 1: Table 1: Comparison of 4 Models for Level 1

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
model.0	3	5494.406	5509.469	-2744.203	5488.406	NA	NA	NA
model.temp	6	5471.995	5502.121	-2729.997	5459.995	28.41148	3	0.0000030
model.time	6	5290.383	5320.510	-2639.192	5278.383	181.61119	0	0.0000000
model.both	10	5296.570	5346.781	-2638.285	5276.570	1.81328	4	0.7700518

Next when we considered the patient level characteristics, we considered adding either scaled Age, White, Income (category), Treatment Group, Female, use of NSAID, or Retired (z_i) as a predictor on the level of the patients groups. We considered 3 possible models summarized below:

First recall our multi level model:

$Y_{it} = (g_0 + u_i) + g_1 z_i + (h_0 + w_i) X_{it} + h_1 z_i X_{it} + e_{it}$

Potential Models Considered:

Model.0
$Pain_{it} = a_i + b_i Time_{it} + e_{it}$
$a_i = g_0 + u_i$
$b_i = h_0 + w_i$

$Pain_{it} = (g_0 + u_i) + (h_0 + w_i) Time_{it} + e_{it}$

Model.main
$Pain_{it} = a_i + b_i Time_{it} + e_{it}$
$a_i = g_0 + g_1 Agescale + u_i$
$b_i = h_0 + w_i$ $Pain_{it} = (g_0 + u_i) + g_1 Agescale + (h_0 + w_i) Time_{it} + e_{it}$

Model.int (eg z_i is age.scale)
$Pain_{it} = a_i + b_i Time_{it} + e_{it}$
$a_i = g_0 + g_1 Agescale + u_i$
$b_i = h_0 + h_1 Agescale + w_i$

$Pain_{it} = (g_0 + u_i) + g_1 Agescale + (h_0 + w_i) Time_{it} + h_1 Agescale_i Time_{it} + e_{it}$

Overall, we see Model.main and Model.int adds a patient level predictor to the second level of the multilevel model.

We will evaluate the result of these various models for each of the patient characteristic variables. The resulting tables will present AIC, BIC, and pvalues from comparison of the models.

Table 2: Comparison of 3 Models with Scaled Age as Predictor

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
model.0	6	5290.383	5320.510	-2639.192	5278.383	NA	NA	NA
model.main	7	5290.161	5325.308	-2638.080	5276.161	2.223	1	0.136
model.int	8	5292.101	5332.270	-2638.051	5276.101	2.282	2	0.319

From Table 2 we see that (scaled) Age is not a good predictor for the patient group level and thus we would retain Model.0 with no predictor at the patient group level, since the pvalues for Model.main and Model.int are insignificant.

For the variable white, we noted the model.int failed to converge but the convergence was close enough. Regardless, based on the table below (Table 3) we see that white is also not a good predictor since none of the models are significantly better than Model.0 as seen by the pvalues.

Table 3: Comparison of 3 Models with White as Predictor

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
model.0	6	5290.383	5320.510	-2639.192	5278.383	NA	NA	NA
model.main	7	5290.999	5326.147	-2638.500	5276.999	1.384	1	0.239
model.int	8	5292.996	5333.165	-2638.498	5276.996	1.387	2	0.500

Next we consider the Income (categorical) variable again we see that none of the models are signficantly different, based on pvalues, than Model.0 thus Income is not a useful predictor at the patient group level (Table 4).

Table 4: Comparison of 3 Models with Income (cat) as Predictor

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
model.0	6	2598.744	2624.538	-1293.372	2586.744	NA	NA	NA
model.main	10	2603.766	2646.756	-1291.883	2583.766	2.978	4	0.561
model.int	14	2607.868	2668.053	-1289.934	2579.868	6.877	8	0.550

Next, we consider Treatment as a potential predictor and we see from Table 5 that Treatment is not a significant predictor for the patient group level because neither of the models are significantly different than Model.0, based on pvalues.

Table 5: Comparison of 3 Models with Treatment as Predictor

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
model.0	6	5290.383	5320.510	-2639.192	5278.383	NA	NA	NA
model.main	7	5292.299	5327.447	-2639.150	5278.299	0.084	1	0.772
model.int	8	5293.774	5333.942	-2638.887	5277.774	0.610	2	0.737

Next, we consider Female as a potential predictor and we see from Table 6 that Female is not a significant predictor for the patient group level because neither of the models are significantly different than Model.0, based on the pvalues.

Table 6: Comparison of 3 Models with Female as Predictor

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
model.0	6	5290.383	5320.510	-2639.192	5278.383	NA	NA	NA
model.main	7	5290.315	5325.463	-2638.158	5276.315	2.068	1	0.150
model.int	8	5292.180	5332.349	-2638.090	5276.180	2.203	2	0.332

We note that Model.Main failed to converge using NSAID as a predictor but the convergence was close enough. From Table 7 we see that Model.int is significantly different than Model.0 at an alpha level of 0.05. However, Model.main is not significantly different than Model.0. Thus, we should consider Model.int which uses NSAID as a patient level predictor of the varying slope and intercept for the patient groups.

Table 7: Comparison of 3 Models with NSAID as Predictor

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
model.0	6	5290.383	5320.510	-2639.192	5278.383	NA	NA	NA
model.main	7	5292.244	5327.392	-2639.122	5278.244	0.139	1	0.709
model.int	8	5287.547	5327.716	-2635.773	5271.547	6.837	2	0.033

For the predictor retired Model.Main failed to converge but it was close enough and we see that retired is not a significant predictor. The models are not significantly different than Model.0 (Table 8).

Table 8: Comparison of 3 Models with Retired as Predictor

	Df	AIC	BIC	logLik	deviance	Chisq	Chi Df	Pr(>Chisq)
model.0	6	4139.772	4168.411	-2063.886	4127.772	NA	NA	NA
model.main	7	4141.335	4174.747	-2063.667	4127.335	0.437	1	0.508
model.int	8	4143.248	4181.433	-2063.624	4127.248	0.524	2	0.769

Final Model

From Tables 2 through 8, we saw that only NSAID was a significant predictor of the 7 considered. The rest of the predictors resulted in models that were not significantly different than the model with just a varying slope and intercept but no predictors at the patient group level.

Our final model was the NSAID model (Model.int) which had the following form:

Level 1 :

$Pain_{it} = a_i + b_i Time_{it} + e_{it}$

Level 2:

$a_i = g_0 + g_1 NSAID_i + u_i$
$b_i = h_0 + h_1 NSAID_i + w_i$

That is:

$Pain_{it} = (g_0 + u_i) + g_1 NSAID_i + (h_0 + w_i) Time_{it} + h_1 NSAID_i Time_{it} + e_{it}$

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: pain ~ time * nsaid + (1 + time | ID)
##    Data: datause
##  Subset: !is.na(datause$nsaid)
##
##      AIC      BIC   logLik deviance df.resid
##   5287.5   5327.7  -2635.8   5271.5     1112
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.2274 -0.5002 -0.0032  0.4848  3.4714
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  ID       (Intercept) 9.688005 3.11256
##           time        0.001186 0.03444  -0.26
##  Residual             3.371965 1.83629
## Number of obs: 1120, groups:  ID, 205
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  7.839032   0.538464  14.558
## time        -0.008743   0.007158  -1.221
## nsaid        0.807043   0.600329   1.344
## time:nsaid  -0.020934   0.008025  -2.609
##
## Correlation of Fixed Effects:
##            (Intr) time   nsaid
## time       -0.385
## nsaid      -0.897  0.345
## time:nsaid  0.344 -0.892 -0.385
```

Intepretation of Final Model

We see our model has fixed effects of NSAID, Time, the interaction between NSAID and Time and the random effects of the Intercept and Time.

Random Effects:

```
##  Groups   Name        Std.Dev. Corr
##  ID       (Intercept) 3.112556
##           time        0.034437 -0.256
##  Residual             1.836291
```

Table 9: Fixed Effects of Model

	Estimate	Std. Error	t value
(Intercept)	7.8390	0.5385	14.5581
time	-0.0087	0.0072	-1.2215
nsaid	0.8070	0.6003	1.3443
time:nsaid	-0.0209	0.0080	-2.6086

We can take this information and find the final model for averging across each patient group (Eqn 1) and the final model for the individuals in each patient group (Eqn 2).

Averaging across observations in the patient groups:

$Pain_{it} = 7.839 + (0.8070)NSAID_i + (-0.00874)Time_{it} + (-0.02093)NSAID_iTime_{it}$ (Eqn 1)

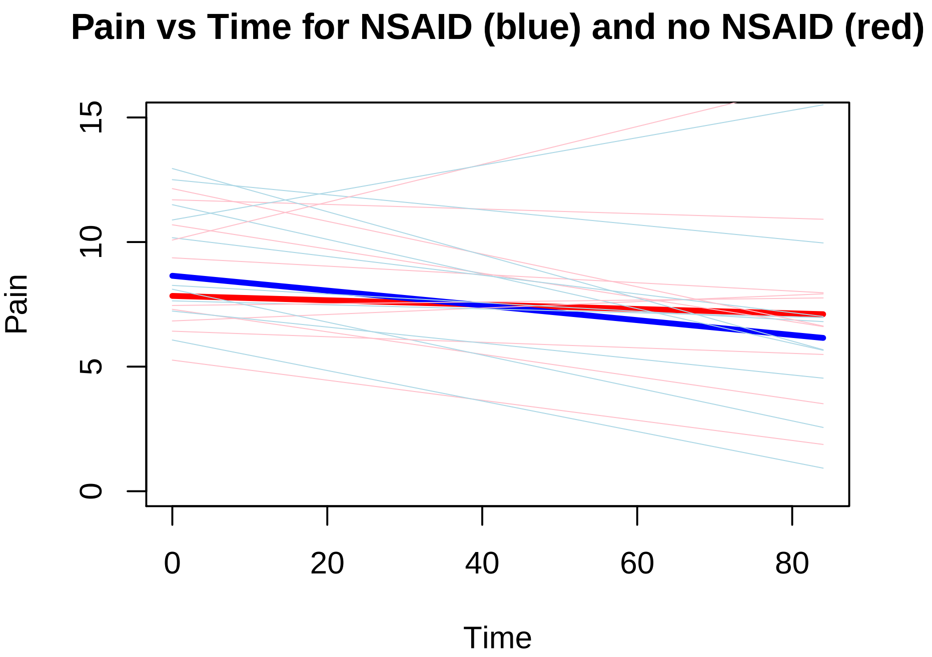
For the first patient (ID 178):

$Pain_{178} = 6.832 + (0.807)NSAID_i + (0.0129)Time_{it} + (-0.02093)NSAID_iTime_{it}$ (Eqn 2)

We can make predictions from the fixed effects (Eq 1) such as at time 0 if there was no use of NSAIDs then on average pain score would be 7.8; however at time 0 if there was use of NSAIDs then on average pain score would be 8.6. Thus, on average at time 0 those taking NSAIDs tend to have more pain.

Similarly, at time 84 if there was no use of NSAIDs then that average pain score would be 7.1; however at time 84 if there was use of NSAIDs than on average pain score would be 6.2. Thus, at this time point (84 days later), on average those taking NSAIDs would have less pain.

We can visualize our model with the predictive plot below of the regression of pain on time. In this plot, the solid lines are the average curves (Eq 1) and the lighter lines are the individual curves for the first 10 patients (Eq 2 is for the first patient). The blue lines are when NSAIDs is used and the red lines are when there is no use of NSAIDs.



Finally, if you recall, both Level 1 and Level 2 of the model had an uncertainty term e_{it} and u_i, w_i which in this model is:

$Var(u_i) = (3.1135)^2 = 9.69$
$Var(w_i) = (0.0344)^2 = 0.0011$
$Var(e_{it}) = (1.836)^2 = 3.37$

Thus, based on the goal to fit a multilevel model of the form presented, we found that a model with Time as a predictor for the pain for individuals at a specific time and NSAID as a patient level for the predictor for the random intercept and random slope was best. Furthermore, in our predictive plots we found that pain was higher at the start for those who used NSAIDs compared to those who did not but after around Time 40, those who used NSAIDS had lower pain compared to those who did not use NSAIDs.

Code Appendix

Part 1

```
library(lme4)
library(nlme)
library(knitr)
library(dplyr)
```

Data Cleaning

```
## cleaning up the dataset

data <- read.csv("McAlindon_Big.csv")

cols <- c("age", "race2", "inccat", "treat", "sex", "nsaid", "Occupation")
patID <- unique(data$ID)
paincol <- c(17, 19, 21, 23, 25, 27, 29)
timecol <- paincol + 1
datause <- matrix(NA,0,12)

# Set up database with pain and temperature measurements extracted on dates when pain measured
for (i in 1:length(patID)) {
  id <- patID[i]
  tempdt <- data[data$ID==id, ]
  zi <- tempdt[1, c("ID", cols)]
  pain <- tempdt[1, paincol]
  pain <- pain[!is.na(pain)]
  time <- tempdt[1, timecol]
  temperature <- tempdt$avgtemp[tempdt$WeatherDate %in% time]
  time <- time[!is.na(time)]
  time <- time - time[1]
  datapti <- cbind(pain, time, temperature)
  datapti <- cbind(zi, datapti)
  datause <- rbind(datause, datapti)
}

datause$ID <- factor(datause$ID)
datause$inccat <- factor(datause$inccat)
female = datause$sex-1
white = datause$race2

#table(datause$Occupation)
# Note varied occupations. Make new variable retired (yes/no)
xx = c(grep("tired",datause$Occupation),grep("TIRED",datause$Occupation))
yy = grep("never retired",datause$Occupation)
retired = rep(0,length(datause$Occupation))
retired[xx[-which(xx %in% yy)]] = 1
retired[datause$Occupation==""] = NA
datause = data.frame(datause,retired,female,white)
datause = datause[!is.na(datause$temperature),] # Remove missing days temperature missing
```

```r
## format nicer to present variables of interest
dataset <- dplyr::select(datause, ID, pain, time,
                         temperature, age, inccat,
                         treat, nsaid, retired, female,
                         white)

dataset$nsaid <- as.factor(dataset$nsaid)
dataset$treat <- as.factor(dataset$treat)
dataset$retired <- as.factor(dataset$retired)
dataset$female <- as.factor(dataset$female)
dataset$white <- as.factor(dataset$white)


summary(dataset)[,-1]
```

Exploration of Potential Models

Fitting Level 1 of the Model:

```r
model.0 = lmer(pain ~ 1|ID,REML = FALSE, data=datause) #Basic model with random intercept
# Include time as a random effect
model.time = lmer(pain ~ time + (1+time|ID),REML = FALSE, data=datause)
# Include temperature as a random effect
#model.temp = lmer(pain ~ temperature + (1+temperature|ID),REML = FALSE,data=datause)

# Model does not converge, need to rescale
temp.scale = scale(datause$temperature) #Rescale temperature
age.scale = scale(datause$age) #Rescale age
datause = data.frame(datause,temp.scale,age.scale)
# Refit with scaled temperature
model.temp = lmer(pain ~ temp.scale + (1+temp.scale|ID),REML = FALSE,
data=datause)
# Model both temperature and time
model.both = lmer(pain ~ time + temp.scale + (1+time+temp.scale|ID),REML = FALSE,
data=datause)
```

```r
## Compare models. Time model best
#anova(model.0,model.temp,model.time,model.both)

kable(anova(model.0,model.temp,model.time,model.both), type = "pandoc",
      caption = "Table 1: Comparison of 4 Models for Level 1")
```

Fitting Level 2 of the Model:

```r
# Loop over variables fitting Main Effect and Interaction models
vars = c("age.scale", "white", "inccat", "treat", "female", "nsaid", "retired")

models <- list()
for (i in 1:length(vars)) {
  code.0 <- paste('lmer(pain ~ time + (1+time|ID), REML=FALSE,data=datause,
subset = !is.na(datause$',vars[i],'))',sep='')
  code.main <- paste('lmer(pain ~ time+', vars[i],' + (1+time|ID),
REML=FALSE,data=datause,subset = !is.na(datause$',vars[i],'))', sep = '')
  code.int <- paste('lmer(pain ~ time*', vars[i],' + (1+time|ID),
REML=FALSE,data=datause,subset = !is.na(datause$',vars[i],'))', sep = '')
  model.0 = eval(parse(text=code.0))
  model.main = eval(parse(text=code.main))
  model.int = eval(parse(text=code.int))

  mods <- c(model.0, model.main, model.int)
  models[[i]] <- mods
  #print(summary(model.0))
  #print(summary(model.main))
  #print(summary(model.int))
  #print(anova(model.0,model.main,model.int))
}
```

1. Age (scaled)

```
#age.scaled variable


#out <- anova(models[[1]][[1]], models[[1]][[2]], models[[1]][[3]])

out1 <- anova(models[[1]][[1]], models[[1]][[2]])
out2 <- anova(models[[1]][[1]], models[[1]][[3]])

out <- rbind(out1,out2[2,])


rownames(out) <- c("model.0", "model.main", "model.int")

kable(out, format = "pandoc", digits = 3,
      caption = "Comparison of 3 Models with Scaled Age as Predictor")
```

2. White

```
#white variable


#out <- anova(models[[2]][[1]], models[[2]][[2]], models[[2]][[3]])

out1 <- anova(models[[2]][[1]], models[[2]][[2]])
out2 <- anova(models[[2]][[1]], models[[2]][[3]])

out <- rbind(out1,out2[2,])

rownames(out) <- c("model.0", "model.main", "model.int")


kable(out, format = "pandoc", digits = 3,
      caption = "Comparison of 3 Models with White as Predictor")
```

3. Income

```
#income variable


#out <- anova(models[[3]][[1]], models[[3]][[2]], models[[3]][[3]])
```

```r
out1 <- anova(models[[3]][[1]], models[[3]][[2]])
out2 <- anova(models[[3]][[1]], models[[3]][[3]])
out <- rbind(out1,out2[2,])

rownames(out) <- c("model.0", "model.main", "model.int")

kable(out, format = "pandoc", digits = 3,
      caption = "Comparison of 3 Models with Income (cat) as Predictor")
```

4. Treatment

```r
#treatment variable


#out <- anova(models[[4]][[1]], models[[4]][[2]], models[[4]][[3]])

out1 <- anova(models[[4]][[1]], models[[4]][[2]])
out2 <- anova(models[[4]][[1]], models[[4]][[3]])
out <- rbind(out1,out2[2,])

rownames(out) <- c("model.0", "model.main", "model.int")

kable(out, format = "pandoc", digits = 3,
      caption = "Comparison of 3 Models with Treatment as Predictor")
```

5. Female

```r
#female variable


#out <- anova(models[[5]][[1]], models[[5]][[2]], models[[5]][[3]])

out1 <- anova(models[[5]][[1]], models[[5]][[2]])
out2 <- anova(models[[5]][[1]], models[[5]][[3]])
out <- rbind(out1,out2[2,])

rownames(out) <- c("model.0", "model.main", "model.int")

kable(out, format = "pandoc", digits = 3,
      caption = "Comparison of 3 Models with Female as Predictor")
```

6. NSAID

```r
#NSAID variable


#out <- anova(models[[6]][[1]], models[[6]][[2]], models[[6]][[3]])

out1 <- anova(models[[6]][[1]], models[[6]][[2]])
out2 <- anova(models[[6]][[1]], models[[6]][[3]])
out <- rbind(out1,out2[2,])


rownames(out) <- c("model.0", "model.main", "model.int")
```

```
kable(out, format = "pandoc", digits = 3,
      caption = "Comparison of 3 Models with NSAID as Predictor")
```

7. Retired

```
#retired variable


#out <- anova(models[[7]][[1]], models[[7]][[2]], models[[7]][[3]])

out1 <- anova(models[[7]][[1]], models[[7]][[2]])
out2 <- anova(models[[7]][[1]], models[[7]][[3]])
out <- rbind(out1,out2[2,])

rownames(out) <- c("model.0", "model.main", "model.int")

kable(out, format = "pandoc", digits = 3,
      caption = "Comparison of 3 Models with Retired as Predictor")
```

Final Model

```
finalmod <- models[[6]][[3]] # NSAID Model.int


summary(finalmod)
```

Intepretation of Final Model

Random Effects:

```
# random effects
summary(finalmod)$varcor


# fixed effects
kable(summary(finalmod)$coeff, digits = 4, type = "pandoc", caption = "Fixed Effects of Model")
```

Predictive Plots

```
#fixed.effects(finalmod) # fixed effects
#random.effects(finalmod) #random effects
#coef(finalmod) #person specific coefficients
x = seq(0,84, by=14)
coefs = fixed.effects(finalmod)
person.coefs = coef(finalmod)
plot(x,rep(0,7),type="n",ylim=c(0,15),xlab="Time"
     ,ylab="Pain", main = "Pain vs Time for NSAID (blue) and no NSAID (red)")
nsaid0 = coefs[1]+x*coefs[2]
nsaid1 = coefs[1]+coefs[3]+x*(coefs[2]+coefs[4])
points(x, nsaid0, pch = "C", type="l",col="red",lwd=3)
points(x, nsaid1, pch = "T",type = "l",col="blue",lwd=3)
beta = person.coefs$ID
for (i in 1:10) { # first 10 patients
cont.pred = beta[i,1] + x*beta[i,2]
tx.pred = beta[i,1]+beta[i,3]+x*(beta[i,2]+beta[i,4])
points(x, cont.pred,type="l",col="pink",lwd=0.5)
points(x, tx.pred,type="l",col="lightblue",lwd=0.5)
}
```