HW3

Blain Morin

October 29, 2018

Question 1:

For this analysis, we model an individual's pain over time. We will first explore and summarize the data set. Then, we will discuss the motivations for using a multilevel model. Finally, we will show our model selection process and examine our final model's regression diagnostics.

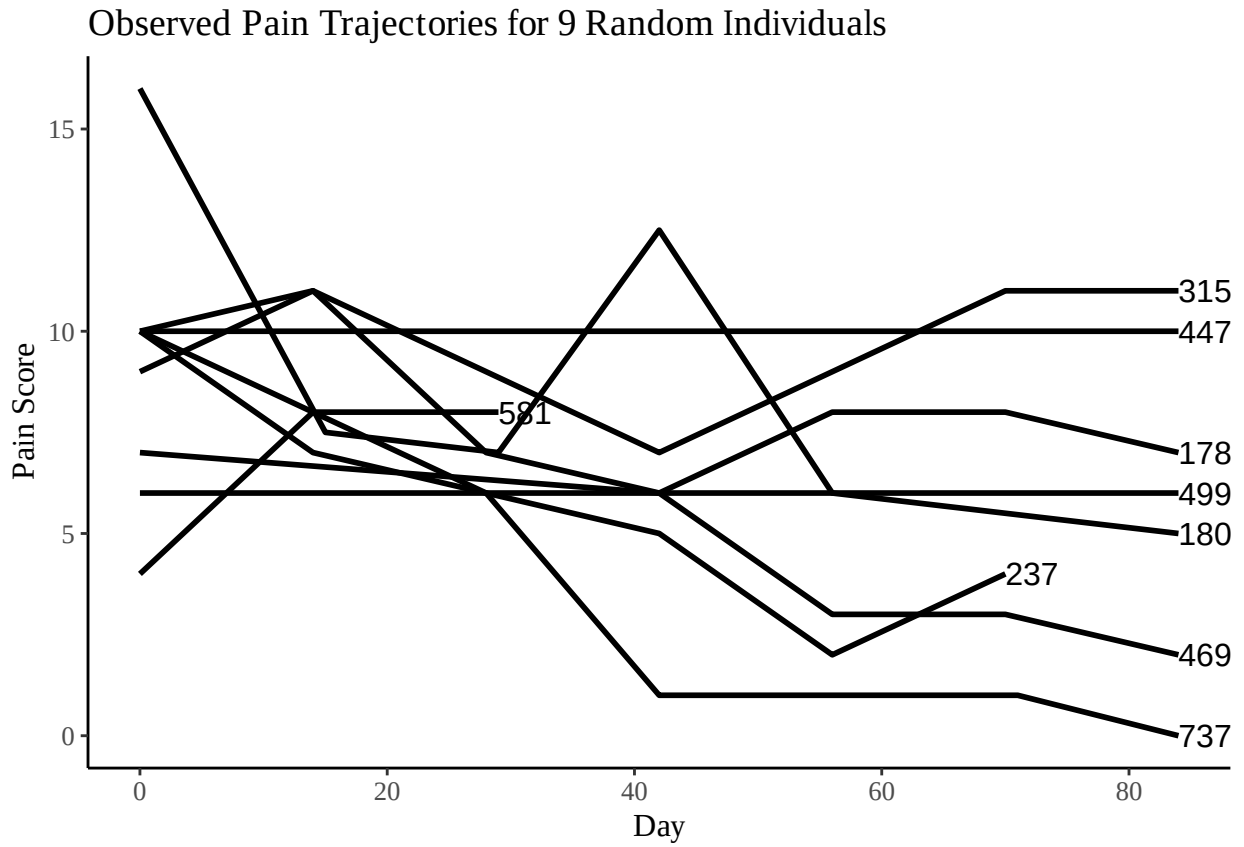
Our data cleaning process included changing the data from wide format to long format. We releveled the factor data to have levels 1 and 0. We also started every individual at time = 0 by subtracting their first time measurement from each of their other time measurements. Here is a summary of the relevant variables:

Table 1: Summary Statistics

Statistic	Mean	St. Dev.	Min	Max	N
Pain Score	7.507	3.543	0	20	503
Average Temperature (F)	60.751	18.289	−2	95	503
Age	60.105	9.754	44	98	503
White or Hispanic	0.966	0.181	0	1	503
Income < 15k	0.141	0.349	0	1	503
Income 15 - 35k	0.360	0.480	0	1	503
Income 35 - 55k	0.274	0.447	0	1	503
Income 55 - 75k	0.103	0.305	0	1	503
Income > 75k	0.121	0.327	0	1	503
Treatment Group = 1	0.469	0.500	0	1	503
Female = 1	0.644	0.479	0	1	503
Retired = 1	0.443	0.497	0	1	503
NSAIDs = 1	0.841	0.366	0	1	503

We see that only 503 observations are used in the summary statistic calculations, whereas our data contains 1137 rows. The omitted observations come from missing data on income and retirement status.

Next, we plotted the pain trajectories over time for nine random individuals:



We notice that the between individual trajectories have considerable variation. The starting pain as well as the time trend both seem to vary significantly between individuals. This variation motivates the use of a multilevel model. We further believe that a multilevel model is appropriate for these data because we have repeated measurements for each individual and thus have within individual correlation between the measurements.

From the prompt, our multilevel model should be of the form:

$$Level1 : Y_{it} = a_i + b_i X_{it} + e_{it}$$

$$Level2 : a_i = g_0 + g_1 z_i + u_i, b_i = h_0 + h_1 z_i + w_i$$

Here g_0 and g_1 are the fixed effects part of the level one intercept, with u_i being the random component.

h_0 and h_1 are the fixed effects part of the level one slope, with w_i being the random component.

We first investigated which covariate to use in level 1, using fully specified level 2 models. We choose not to include income and retirement status as covariates because of the missing data as seen above. Here are the regression results:

Table 2: Choosing Random Components

	Dependent variable:	
	Pain Score	
	Rand Slope Temp	Rand Slope Time
	(1)	(2)
Avg Temp	0.162	0.165
	(0.595)	(0.170)
Day	-0.029^{***}	-0.015
	(0.003)	(0.014)
Age	-0.342	-0.182
	(0.236)	(0.256)
White or Hispanic	0.800	0.677
	(0.733)	(0.793)
Treated	0.038	-0.134
	(0.466)	(0.506)
Female	0.412	0.536
	(0.503)	(0.544)
NSAIDs	-0.298	0.663
	(0.587)	(0.632)
Day * Avg Temp	-0.008^{***}	-0.008^{**}
	(0.003)	(0.003)
Age * Avg Temp	0.198	
	(0.146)	
White or Hispanic * Avg Temp	-0.658	
	(0.458)	
Treated * Avg Temp	0.221	
	(0.283)	
Female * Avg Temp	0.289	
	(0.305)	
NSAIDs * Avg Temp	0.348	
	(0.361)	
Age * Day		-0.002
		(0.004)
White or Hispanic * Day		0.003
		(0.011)
Treated * Day		0.003
		(0.007)
Female * Day		-0.001
		(0.007)
NSAIDs * Day		-0.021^{**}
		(0.008)
Constant	7.797^{***}	7.135^{***}
	(0.949)	(1.016)
Observations	1,137	1,137
Log Likelihood	$-2{,}617.699$	$-2{,}593.829$
Akaike Inf. Crit.	5,271.397	5,223.658
Bayesian Inf. Crit.	5,361.436	5,313.697
Note:	*p<0.1; **p<0.05; ***p<0.01	
	Temperature, Age are standardized	

We see that model 2, with the random intercept and random slope on day, has the lower AIC and BIC. Using model 2, we then did covariate selection using backwards stepwise regression using partial p values. When we removed variables, we removed both the main effect and the time interaction so that the model form would stay consistent with the prompt. Here are the stepwise regression results:

Table 3: Covariate Selection: Backwards Steps

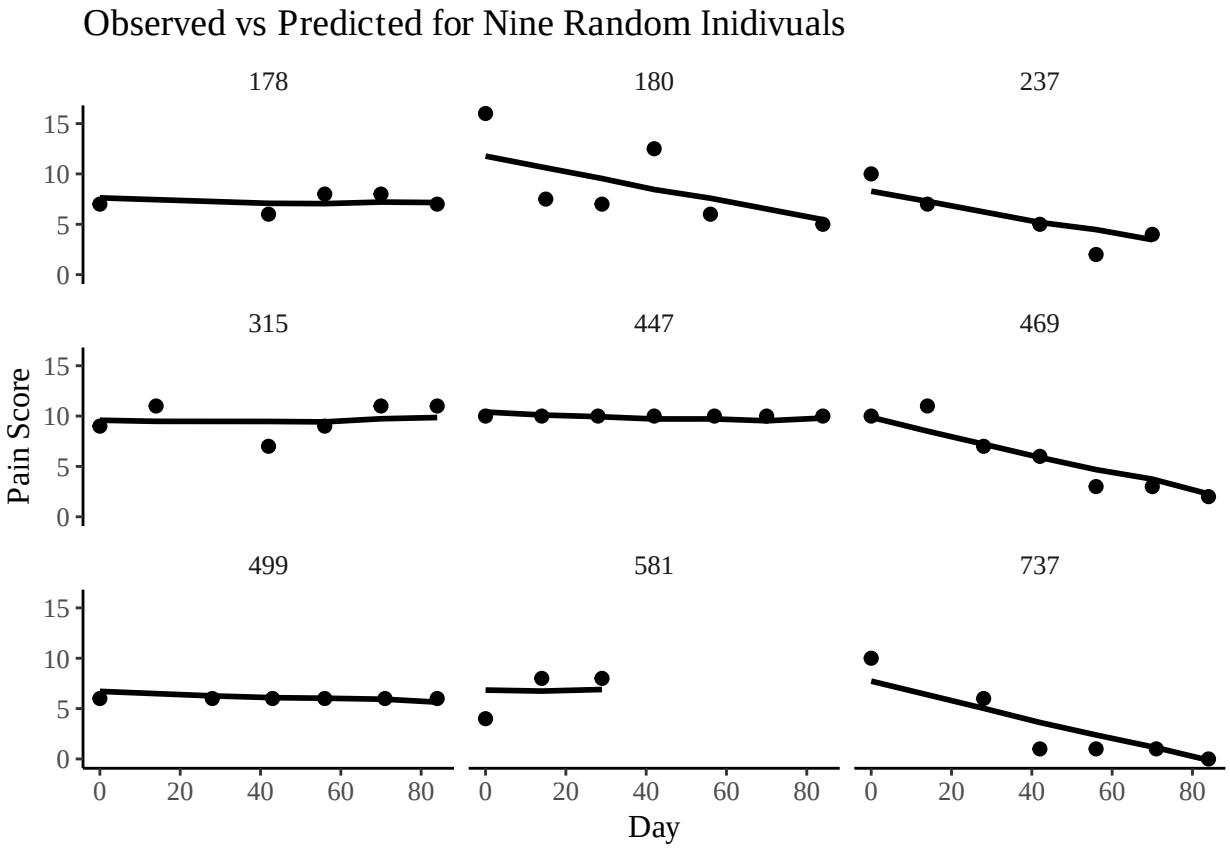
	Dependent variable:			
	Pain Score			
	(1)	(2)	(3)	(4)
Avg Temp	0.165	0.161	0.158	0.163
	p = 0.331	p = 0.343	p = 0.351	p = 0.335
Day	−0.015	−0.015	−0.013	−0.010
	p = 0.282	p = 0.243	p = 0.282	p = 0.156
Age	−0.182	−0.253	−0.248	−0.239
	p = 0.477	p = 0.303	p = 0.311	p = 0.329
White or Hispanic	0.677	0.756	0.775	
	p = 0.394	p = 0.338	p = 0.325	
Treated	−0.134	−0.213		
	p = 0.792	p = 0.671		
Female	0.536			
	p = 0.325			
NSAIDs	0.663	0.629	0.672	0.684
	p = 0.295	p = 0.319	p = 0.280	p = 0.272
Avg Temp * Day	−0.008**	−0.008**	−0.008**	−0.008**
	p = 0.011	p = 0.011	p = 0.011	p = 0.011
Age * Day	−0.002	−0.002	−0.002	−0.002
	p = 0.495	p = 0.496	p = 0.485	p = 0.485
White or Hispanic * Day	0.003	0.003	0.003	
	p = 0.772	p = 0.778	p = 0.795	
Treated * Day	0.003	0.003		
	p = 0.662	p = 0.646		
Female * Day	−0.001			
	p = 0.933			
NSAIDs * Day	−0.021**	−0.021**	−0.022***	−0.022***
	p = 0.012	p = 0.012	p = 0.009	p = 0.008
Constant	7.135***	7.476***	7.322***	8.000***
	p = 0.000	p = 0.000	p = 0.000	p = 0.000
Observations	1,137	1,137	1,137	1,137
Log Likelihood	−2,593.829	−2,590.609	−2,586.793	−2,584.546
Akaike Inf. Crit.	5,223.658	5,213.217	5,201.587	5,193.092
Bayesian Inf. Crit.	5,313.697	5,293.252	5,271.617	5,253.118

Note: *p<0.1; **p<0.05; ***p<0.01
Temperature, Age are standardized

We choose model 4 from table 3 to be the best model. The non interaction coefficients are interpreted as the average change in baseline pain. For example, using NSAIDs is associated with an increased pain score of .684 at baseline, all else equal.

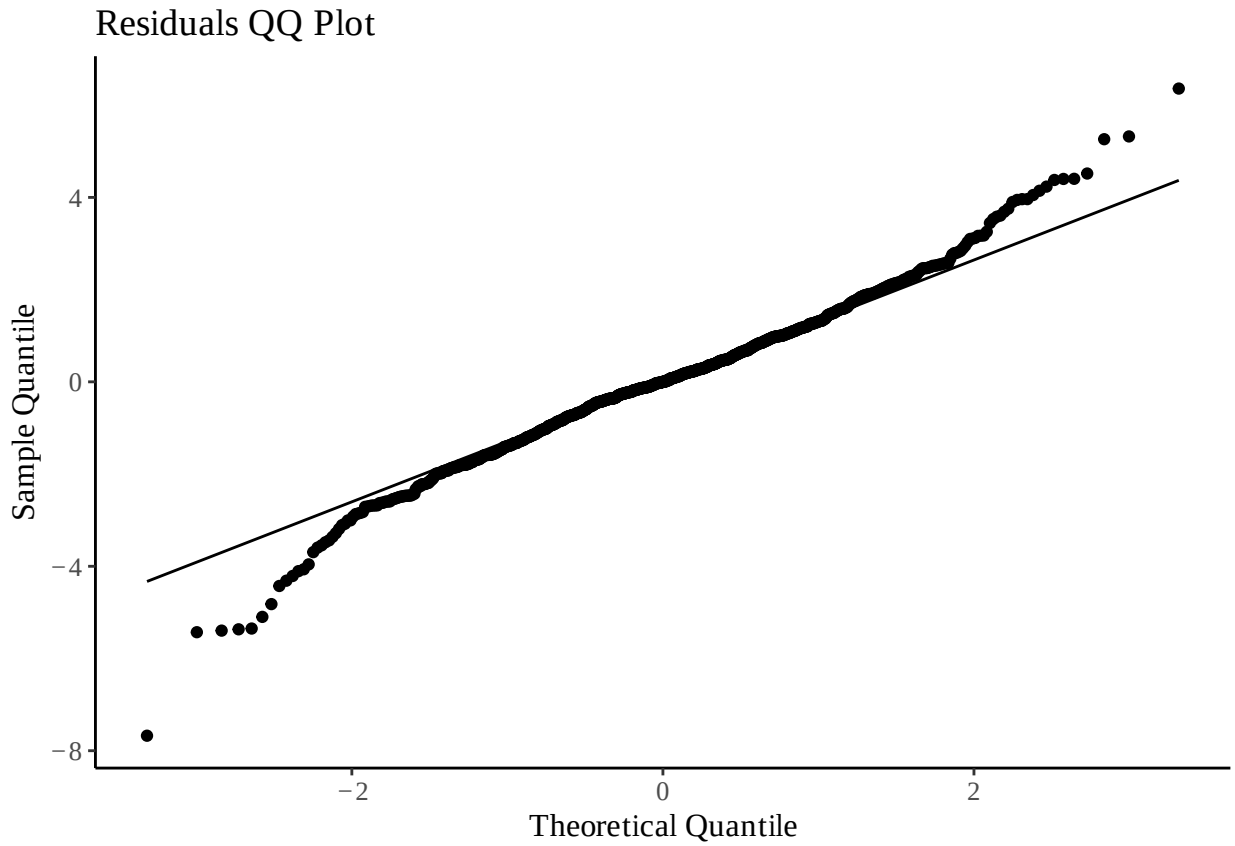
The interaction terms are interpreted as the mean change in the time trend. For example, the average change in pain per day is .022 points lower for those using NSAIDs compared to those who do not use NSAIDs, all else equal.

Finally, we check our model fit. First we examined the predicted vs observed values for nine random individuals:



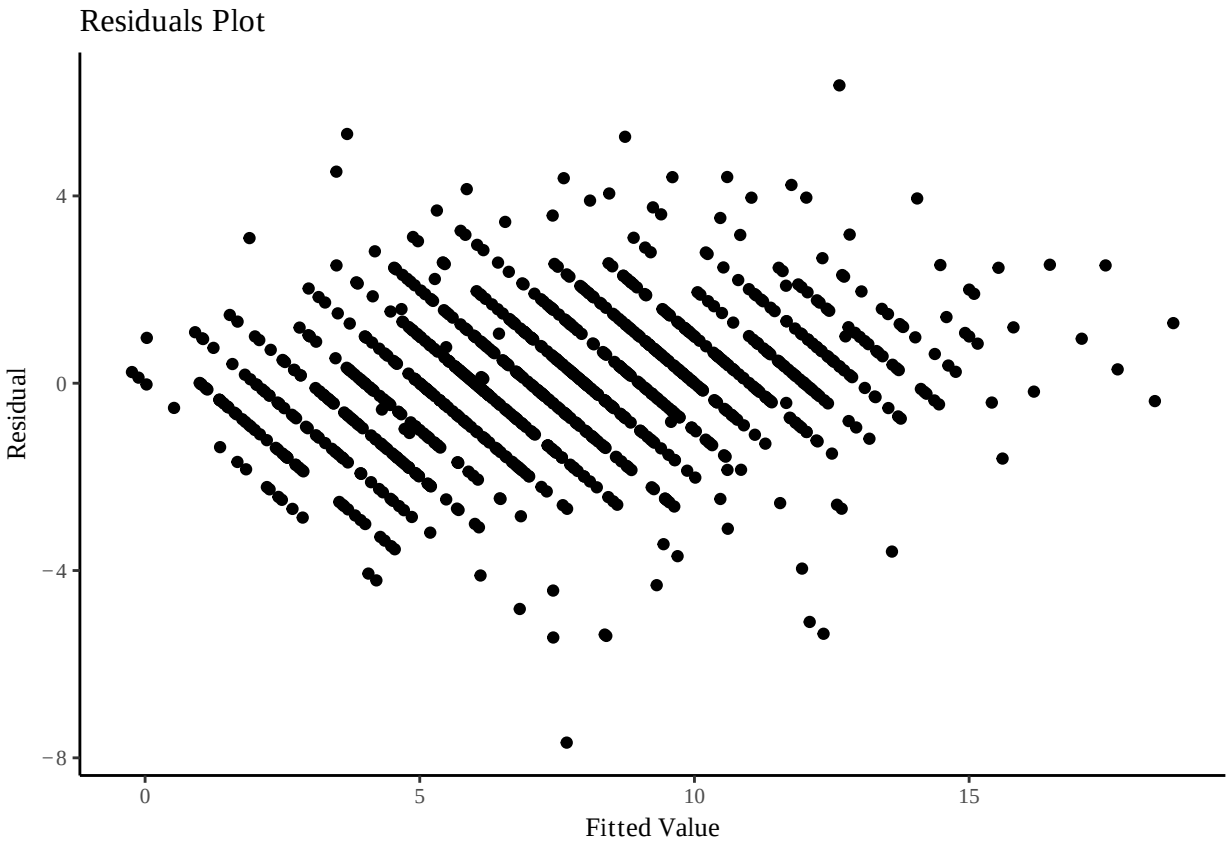
We see that the model appears to fit the data well for those nine people.

We also check the normality of the residuals:



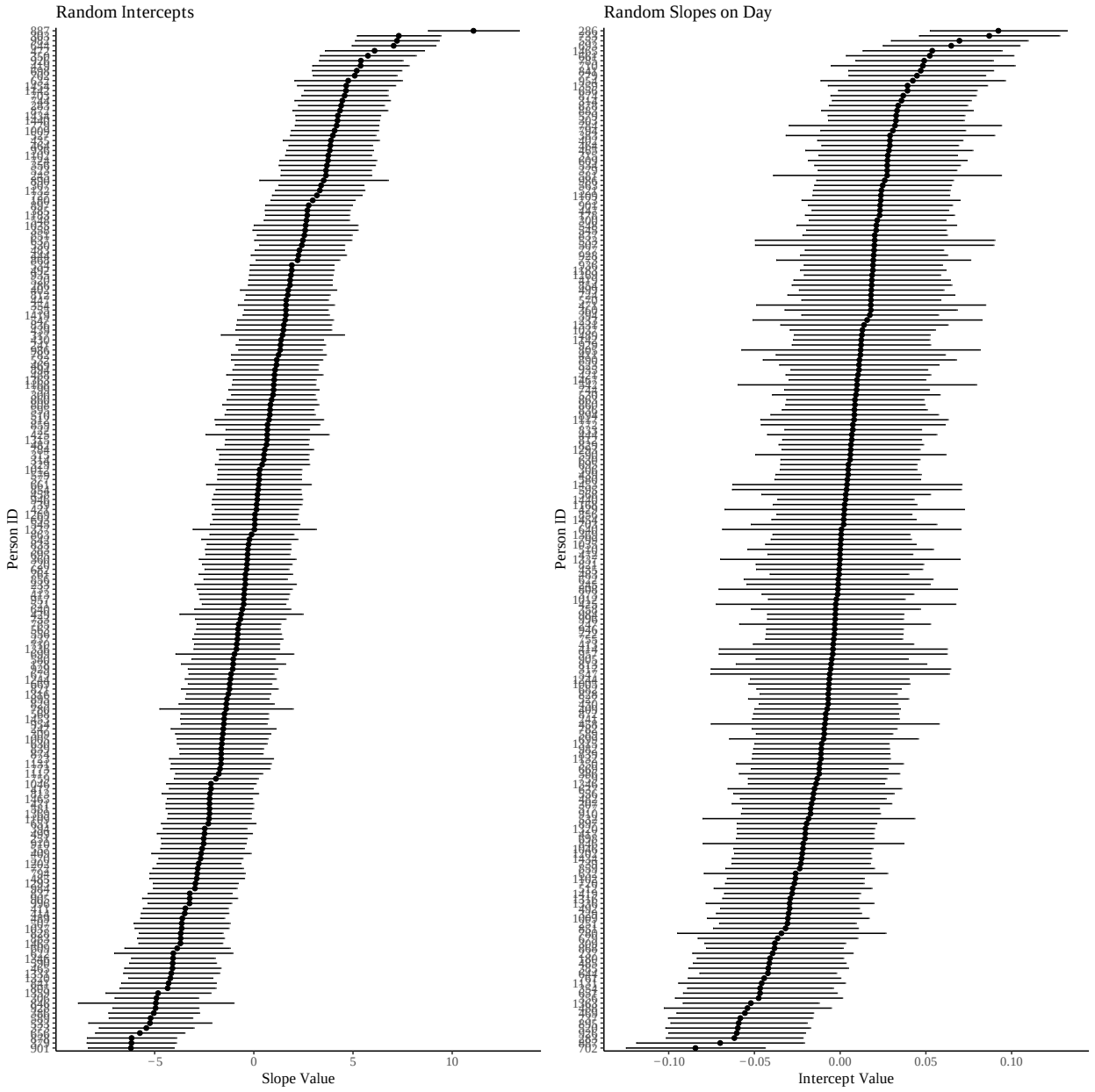
The QQ plot shows that our residuals deviate from normality. Our model appears to be under fitting in the more extreme high and low pain values.

We also examine the residuals vs fitted plot:



We see that the residuals do not look randomly scattered about the 0 line: there seems to be larger values in the middle.

Lastly, we present the random effects from our model:



We see that the random component on the intercept varies between about -7 to about 10 and the random component of the slope varies between about -.08 to .09.

Question 2:

For question 2, we are trying to determine which factors effect radon measurements in a house. For this analysis, we assume that there is a correlation between radon measurements for houses in the same county. In other words, houses within the same county are likely to be more similar than houses in different counties.

We first joined the house radon data with the county level uranium data. We then releveled the factor variables so that they only contained 1 and zero levels. For building type, 1 = single family home and 0 = other. Has.basement = 1 if the house has a basement and = 0 if not. Measurements taken in the basement are coded as which.floor = 0, any other floor = 1. We also log transformed uranium and radon measurements.

Here is a summary table of complete observations:

Table 4: Summary Statistics

Statistic	Mean	St. Dev.	Min	Max	N
log(Radon)	1.064	1.141	−11.513	3.875	1,138
First Floor = 1	0.777	0.417	0	1	1,138
Has Basement = 1	0.223	0.417	0	1	1,138
Single Family Home = 1	0.875	0.331	0	1	1,138
County log(Uranium)	0.949	0.220	0	1	1,138
log.Uppm	−0.198	0.371	−0.882	0.528	1,138

We fit a multi level model to the data where we allow for both varying intercepts and slopes:

$$Level1 : log(Radon)_{County,House} = a_C + b_C X_{CH} + e_{CH}$$
$$Level2 : a_C = g_0 + g_1 z_C + u_C, b_C = h_0 + h_1 z_C + w_C$$

Our county level predictor is log(Uranium) and our house level variables are floor which the measurement was taken, whether or not the house has a basement, and whether or not the house is a single family home. Here are the regression results from the fully specified varying intercepts and varying slopes model:

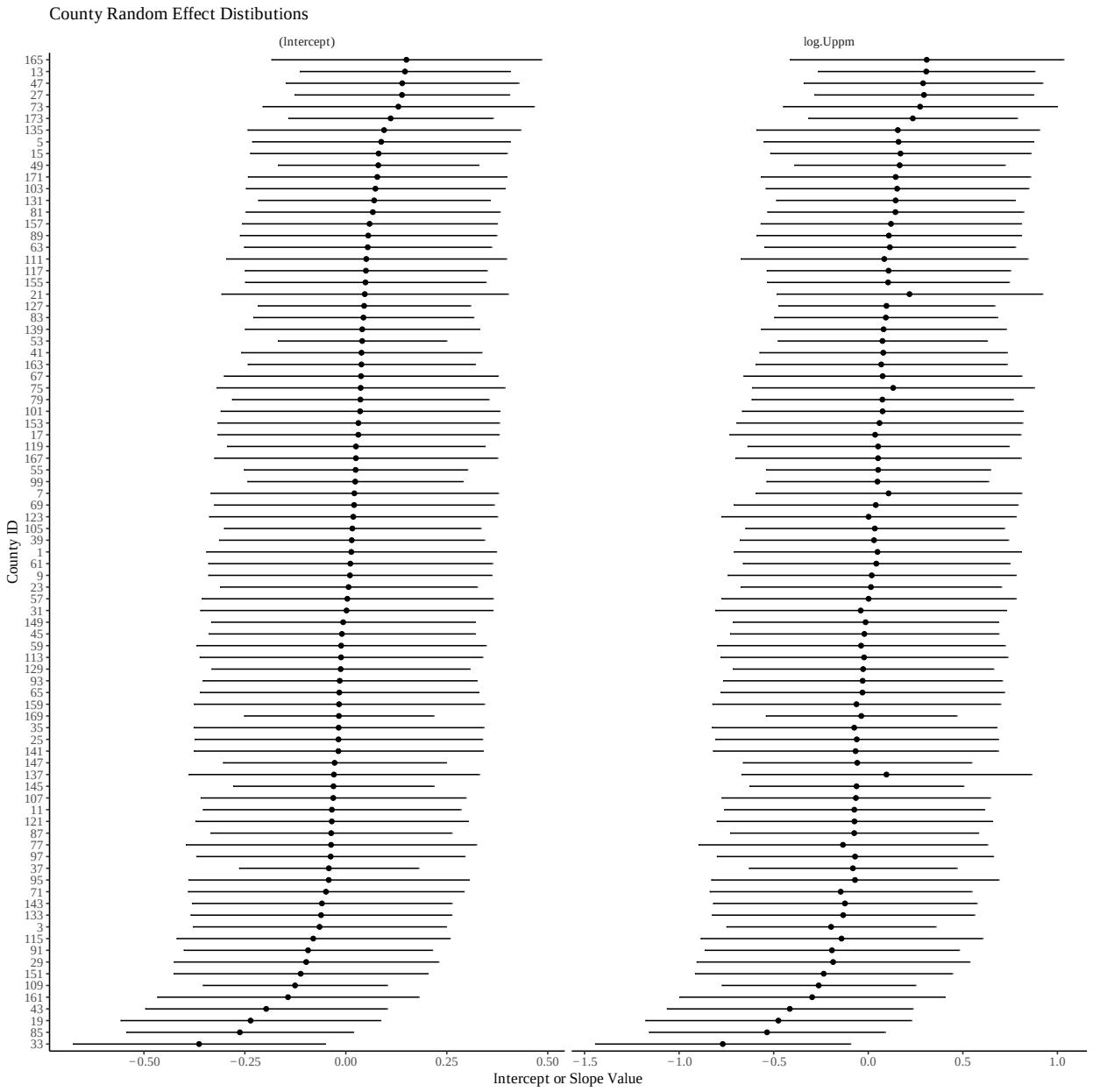
Table 5: Mixed Model Results

	Dependent variable:
	log(Radon)
log(Uranium)	1.861^{***}
	p = 0.0003
First Floor = 1	-0.965^{***}
	p = 0.000
Has Basement = 1	-0.229
	p = 0.129
Single Family = 1	0.340^{**}
	p = 0.034
log(Uranium) * Floor	-0.888^{***}
	p = 0.001
log(Uranium) * Basement	-0.865^{**}
	p = 0.014
log(Uranium) * Single Family	-0.110
	p = 0.784
Constant	1.332^{***}
	p = 0.000
Observations	1,138
Log Likelihood	$-1,655.821$
Akaike Inf. Crit.	3,335.641
Bayesian Inf. Crit.	3,396.085
Note:	*p<0.1; **p<0.05; ***p<0.01

The non interaction terms are interpreted as the average change in log(Radon), for a house in a county with no Uranium in the soil. For example, if the measurement was taken on the first floor of a house in a county with no Uranium, then we expect the measurement to be .38 units lower (exp(-.965)) than that of a measurement taken in the basement in a house within a county with no Uranium in the soil, all else equal. The interaction terms are interpreted as the average additional change in log(Radon) for 1 unit increase in Uranium level within a county. For example, when Uranium in a county increases by 1 unit, the effect of the measurement being taken on the first floor is an additional .41 (exp(.-.888)) decreased compared to the floor effect in a county with no Uranium in the soil.

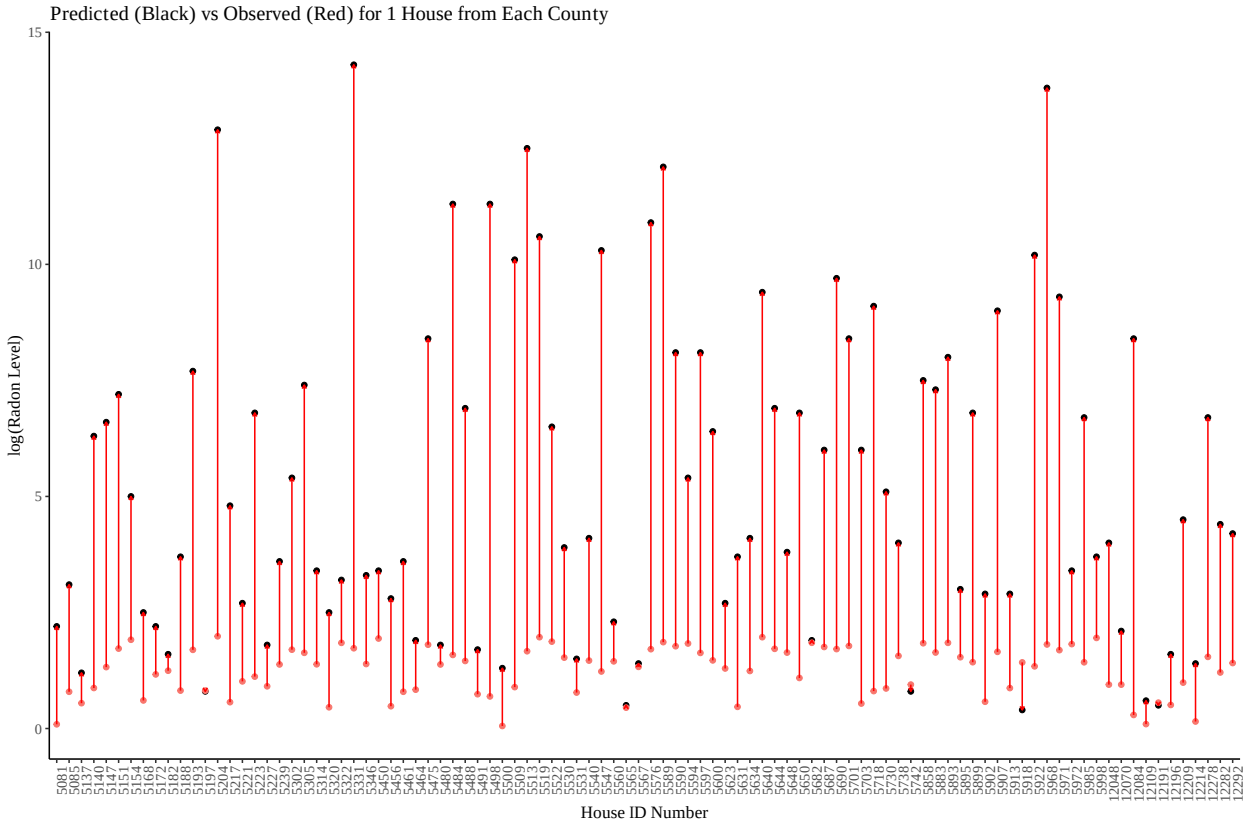
Overall, we see that all the covariates are significant in either their main effects or interactions. This means that all covariates are important in modeling radon measurements.

Here is a plot of the random components for the intercept and slope on log(Uranium):



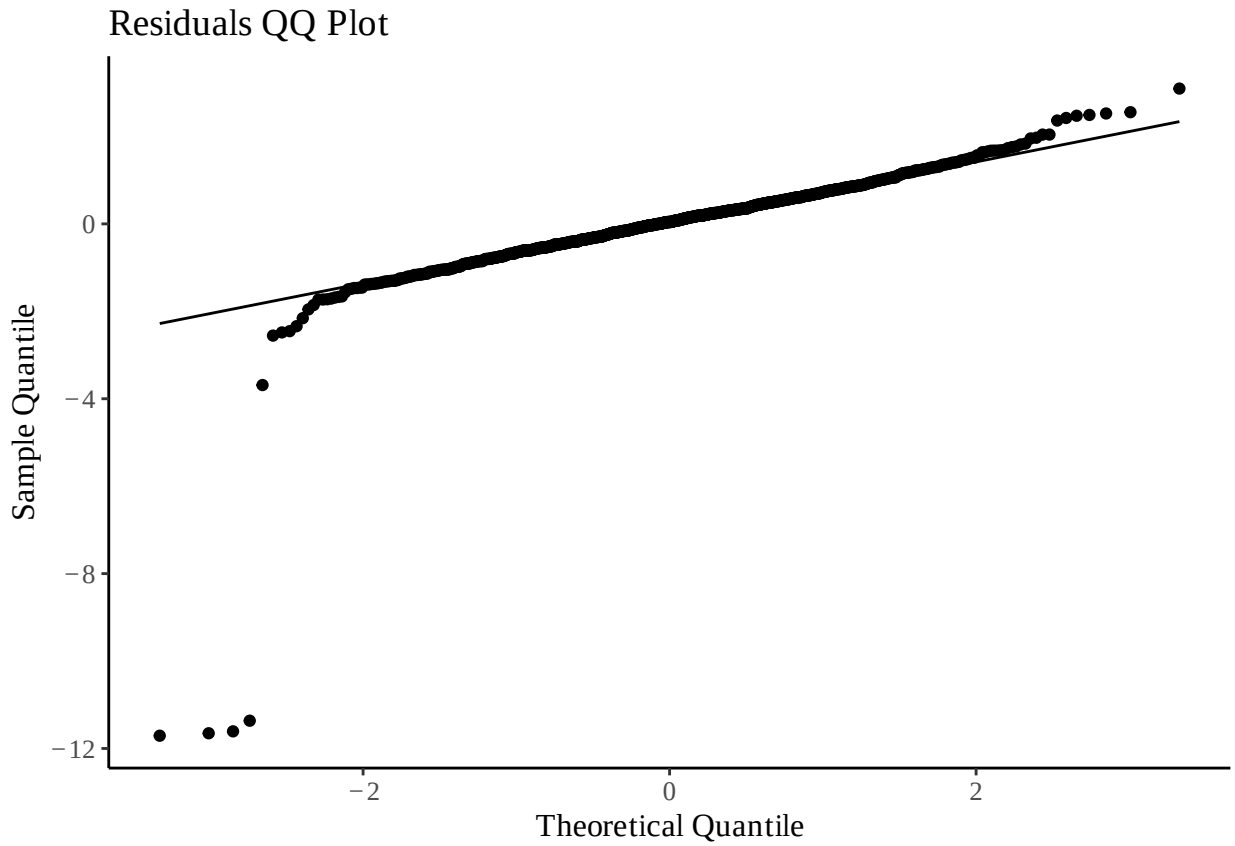
We see that the random component of the intercept varies from about -.3 to .1. The random components on the slope vary from about -.7 to .1. Many of the random components are near 0.

We also examine the fit of our model by looking at the observed versus predicted values for one random home in each county:



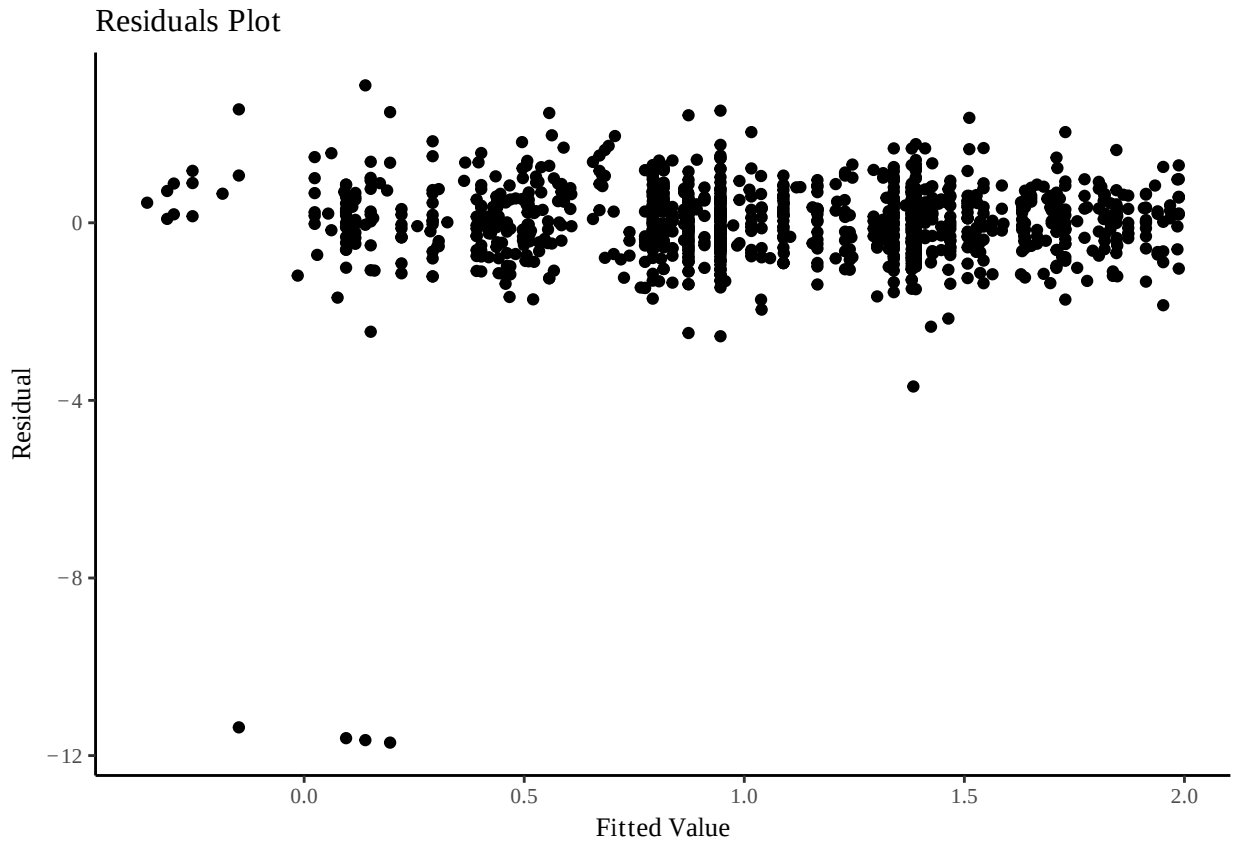
We see that our model tends to under predict, as most of the red dots (the predicted values) are below the black dots (the observed variables). We also notice that the errors of the observed high level log(Radon) are higher on average. This indicates a problem with our regression.

We next checked the normality of our residuals:



We see that our model is under predicting the more extreme values. The four points where the sample quantile is almost -12 should be examined in more detail.

Lastly, we checked a scatterplot of the residuals:



The residuals appear randomly scattered around the zero line. Again, we see the four outlier residuals at the bottom of the plot, which should be examined more closely.

Appendix: Code

Libraries

```
### Load required libraries
library(readr)
library(lme4)
library(tidyr)
library(nlme)
library(stargazer)
library(dplyr)
library(ggplot2)
library(extrafont)
library(extrafontdb)
library(grid)
library(gridExtra)
library(directlabels)
```

Question 1

```
### Load and clean data
mcalindon = read_csv("McAlindon_Big.csv")

weather = mcalindon %>%
  select(ID, WeatherDate, avgtemp, age, race2, inccat, treat, sex, retire, nsaid)

pains = mcalindon %>%
  select(ID, pain.1, pain.2, pain.3, pain.4, pain.4, pain.5, pain.6, pain.7) %>%
  group_by(ID) %>%
  slice(1)

pains = pains %>%
  gather(key = pain.time, value = pain.score, pain.1:pain.7 ) %>%
  group_by(ID) %>%
  mutate( index = row_number(ID))

days = mcalindon %>%
  select(ID, lastdt1, lastdt2, lastdt3, lastdt4, lastdt5, lastdt6, lastdt7) %>%
  group_by(ID) %>%
  slice(1)

days = days %>%
  gather(key = time.name, value = day, lastdt1:lastdt7) %>%
  mutate(index = row_number(ID))

pain.w.days = pains %>%
  inner_join(days)

Q1 = pain.w.days %>%
  rename(WeatherDate = day) %>%
  inner_join(weather)


### Relevel Factors
Q1 = Q1 %>%
  mutate(female = ifelse(sex == 2, 1, 0)) %>%
  mutate(retired = ifelse(retire == 2, 1, 0))


### Start each person at day zero
Q1 = Q1 %>%
  group_by(ID) %>%
  mutate(day = WeatherDate - min(WeatherDate)) %>%
  ungroup()
```

```
###Summary table

variables1 = Q1 %>%
  select(pain.score, avgtemp, age, race2, inccat, treat, female, retired, nsaid)

table1 = model.matrix(~ pain.score + avgtemp + age + race2 + as.factor(inccat) + treat + female + retire
                      data = variables1)
```

```
stargazer(as.data.frame(table1),
          title = "Summary Statistics",
          table.placement = "H",
          header = FALSE,
          summary.stat = c("mean", "sd", "min", "max", "n"),
          covariate.labels = c(
            "Pain Score",
            "Average Temperature (F)",
            "Age",
            "White or Hispanic",
            "Income < 15k",
            "Income 15 - 35k",
            "Income 35 - 55k",
            "Income 55 - 75k",
            "Income > 75k",
            "Treatment Group = 1",
            "Female = 1",
            "Retired = 1",
            "NSAIDs = 1"
          ))
```

```
### Trajectoried for 9 random individuals

Q1 %>%
  filter(ID %in% c(178, 180, 237, 315, 447, 469, 499, 581, 737)) %>%
  ggplot(aes(x = day, y = pain.score)) +
  geom_line(aes(group = ID), show.legend = FALSE, size = 1) +
  geom_dl(aes(label = ID), method = "last.points") +
  ggtitle("Observed Pain Trajectories for 9 Random Individuals") +
  ylab("Pain Score") +
  xlab("Day") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"))
```

```
### Rand int and slope on temp


mm1.Q1 = lmer(pain.score ~ (1 + scale(avgtemp)| ID) +
                scale(avgtemp) +
                day +
                scale(age) +
                as.factor(race2) +
                as.factor(treat) +
                as.factor(female) +
                as.factor(nsaid) +
                scale(avgtemp):(day +
                scale(age) +
                as.factor(race2) +
                as.factor(treat) +
                as.factor(female) +
                as.factor(nsaid)),
              data = Q1,
              na.action = na.exclude,
              control = lmerControl(optCtrl=list(max=100000000)))
```

```
### Random int and slope on day

mm2.Q1 = lmer(pain.score ~ (1 + day| ID) +
                scale(avgtemp) +
                day +
                scale(age) +
                as.factor(race2) +
                as.factor(treat) +
                as.factor(female) +
                as.factor(nsaid) +
                day:(scale(avgtemp) +
                scale(age) +
                as.factor(race2) +
                as.factor(treat) +
                as.factor(female) +
                as.factor(nsaid)),
              data = Q1,
              na.action = na.exclude)


### Summary table of the 2 regressions
stargazer(mm1.Q1, mm2.Q1,
          title = "Choosing Random Components",
          table.placement = "H",
          header = FALSE,
          font.size = c("tiny"),
          dep.var.labels = c("Pain Score"),
          notes = c("Temperature, Age are standardized"),
          column.labels = c("Rand Slope Temp", "Rand Slope Time"),
          covariate.labels = c("Avg Temp",
                               "Day",
                               "Age",
                               "White or Hispanic",
                               "Treated",
                               "Female",
                               "NSAIDs",
                               "Day * Avg Temp",
                               "Age * Avg Temp",
                               "White or Hispanic * Avg Temp",
                               "Treated * Avg Temp",
                               "Female * Avg Temp",
                               "NSAIDs * Avg Temp",
                               "Age * Day",
                               "White or Hispanic * Day",
                               "Treated * Day",
                               "Female * Day",
                               "NSAIDs * Day"))
```

```
### Observed vs Predicted for nine people

nine = Q1 %>%
  mutate(fit = fitted(mm5.Q1)) %>%
```

```
  filter(ID %in% c(178, 180, 237, 315, 447, 469, 499, 581, 737)) %>%
  mutate(ID = as.factor(ID))

nine %>%
  ggplot(aes(x = day, y = pain.score)) +
  geom_point(size=2, show.legend = FALSE) +
  geom_line( aes(y = fit), size=1, show.legend = FALSE) +
  facet_wrap(~ID) +
  ylab("Pain Score") +
  xlab("Day") +
  ggtitle("Observed vs Predicted for Nine Random Inidivuals") +
  theme_classic() +
  theme(text=element_text(size=12,  family="CM Sans"), strip.background = element_blank())
```

```
### Residuals plot

Q1 = Q1 %>%
  mutate(resids = resid(mm5.Q1))

Q1 %>%
  ggplot(aes(sample = resids)) +
  stat_qq() +
  stat_qq_line() +
  theme_classic() +
  ylab("Sample Quantile") +
  xlab("Theoretical Quantile") +
  ggtitle("Residuals QQ Plot") +
  theme(text=element_text(size=12,  family="CM Sans"))
```

```
### Residual scatter plot
resi.plot = ggplot(mm5.Q1) +
  geom_point(aes(x = .fitted,
                 y = .resid), show.legend = FALSE) +
  theme_classic() +
  scale_color_continuous(low = "black", high = "red") +
  xlab("Fitted Value") +
  ylab("Residual") +
  ggtitle("Residuals Plot") +
  theme(text=element_text(size=10,  family="CM Sans"))

resi.plot
```

```
### Random Effects Plot

yy = ranef(mm5.Q1, condVar = TRUE)

ranef.data = as.data.frame(yy)

ints = ranef.data %>%
  filter(term == "(Intercept)")

slope = ranef.data %>%
  filter(term == "day")
```

```
slope$ordered = reorder(slope$grp, slope$condval)

labelss = c("Intercept")

int = ggplot(ints, aes(y=grp, x=condval))+
  geom_point()+
  geom_errorbarh(aes(xmin=condval-2*condsd,xmax=condval+2*condsd),height=0) +
  theme_classic() +
  ylab("Person ID") +
  xlab("Slope Value") +
  ggtitle("Random Intercepts") +
  theme(text=element_text(size=12,  family="CM Sans"))

slopes = ggplot(slope, aes(y=ordered, x=condval))+
  geom_point()+
  geom_errorbarh(aes(xmin=condval-2*condsd, xmax=condval+2*condsd),height=0) +
  theme_classic() +
  ylab("Person ID") +
  xlab("Intercept Value") +
  ggtitle("Random Slopes on Day") +
  theme(text=element_text(size=12,  family="CM Sans"))

grid.arrange(int, slopes, nrow = 1)
```

Question 2

```
### Data import and cleaning
srrs = read.csv("srrs2.txt")

min.srrs = srrs %>%
  filter(state2 == "MN") %>%
  select(idnum, state2, stfips, typebldg, floor, basement, activity, county, cntyfips)

city = read.csv("cty.txt")

min.city = city %>%
  filter(st == "MN") %>%
  select(stfips, ctfips, st, cty, Uppm)  %>%
  rename(cntyfips = ctfips) %>%
  group_by(stfips, cntyfips) %>%
  slice(1) %>%
  ungroup()

Q2 = inner_join(min.srrs, min.city, by = "cntyfips")

Q2 = Q2 %>%
  mutate(has.basement = as.integer(basement)) %>%
  mutate(has.basement = as.factor(ifelse(has.basement == 3, 0, ifelse(has.basement == 4, 1 , NA)))) %>%
  mutate(is.sfh = as.factor(ifelse(typebldg == 1, 1, 0))) %>%
  mutate(which.floor = as.factor(ifelse(floor == 1, 1, ifelse(floor == 9, NA, 0)))) %>%
  mutate(log.activity = log(I(activity+.00001))) %>%
  mutate(log.Uppm = log(I(Uppm+.0000001)))  %>%
```

```
  filter(complete.cases(.))
```

```
### Run model

mm.Q2 = lmer(log.activity ~
               log.Uppm*(which.floor +
               has.basement +
               is.sfh) +
               (1 + log.Uppm | cntyfips),
             data = Q2)


stargazer(mm.Q2,
          report=('vc*p'),
          header = FALSE,
          title = "Mixed Model Results",
          table.placement = "H",
          dep.var.labels = c("log(Radon)"),
          covariate.labels = c("log(Uranium)",
                               "First Floor = 1",
                               "Has Basement = 1",
                               "Single Family = 1",
                               "log(Uranium) * Floor",
                               "log(Uranium) * Basement",
                               "log(Uranium) * Single Family"))
```

```
### Random Effects Plot

yy = ranef(mm.Q2, condVar = TRUE)

ranef.data = as.data.frame(yy)

ggplot(ranef.data, 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) +
  theme_classic() +
  ylab("County ID") +
  xlab("Intercept or Slope Value") +
  ggtitle("County Random Effect Distibutions") +
  theme(text=element_text(size=12,  family="CM Sans"), strip.background = element_blank())
```

```
### Predicted vs Observed

Q22 = Q2 %>%
  mutate(model.prediction = fitted(mm.Q2)) %>%
  mutate(resids = residuals(mm.Q2))

test = Q22 %>%
  group_by(county) %>%
  arrange(county) %>%
  slice(1) %>%
  ungroup()
```

```r
ggplot(data = test, aes(x = as.factor(idnum))) +
  geom_point(aes(y = activity)) +
  geom_point(aes(y = model.prediction, color = "red"), show.legend = FALSE) +
  geom_segment(aes(xend = as.factor(idnum), y = model.prediction, yend = activity),
               arrow = arrow(length = unit(0.2, "line")),
               color="red") +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  xlab("House ID Number") +
  ylab("log(Radon Level)") +
  ggtitle("Predicted (Black) vs Observed (Red) for 1 House from Each County") +
  theme(text=element_text(size=12,  family="CM Sans"))
```

```r
### Residuals QQ Plot

Q22 %>%
  ggplot(aes(sample = resids)) +
  stat_qq() +
  stat_qq_line() +
  theme_classic() +
  ylab("Sample Quantile") +
  xlab("Theoretical Quantile") +
  ggtitle("Residuals QQ Plot") +
  theme(text=element_text(size=12,  family="CM Sans"))
```

```r
resi.plot2 = ggplot(mm.Q2) +
  geom_point(aes(x = .fitted,
                 y = .resid), show.legend = FALSE) +
  theme_classic() +
  scale_color_continuous(low = "black", high = "red") +
  xlab("Fitted Value") +
  ylab("Residual") +
  ggtitle("Residuals Plot") +
  theme(text=element_text(size=10,  family="CM Sans"))

resi.plot2
```