Biostat 209 Lab

Repeated Measures 1

Lab Summary

The purpose of this lab is to cover basic manipulation of longitudinal data and to indicate how some longitudinal analyses fit with more standard analyses. We will also consider some diagnostic methods to check assumptions.

Before you start this tutorial, download the GAbabies dataset and the OAI datasets from the course web site and start a record of your session. The Georgia babies dataset follows successive birthweights of infants to mothers (each of whom had five children) from vital statistics in Georgia. We are going to be interested in whether birthweight increases with birth order and mothers' age. The variables in the dataset are:

1) Mother's ID number (`momid`)
2) Birth order (`birthord`)
3) Mother's age at birth of infant (`momage`)
4) Mother's age at birth of first infant (`initage`)
5) Change in mother's age from first infant to current infant (`timesnc`)
6) Birthweight of infant (`bweight`)
7) Change in birthweight of infant from first infant to current infant (`delwght`)
8) Whether the birthweight is under 3000g or not (`lowbirth)`.

Getting started: descriptive statistics

Load the GAbabies dataset and use the panel data library to tell R about the structure of the data and describe it:

```
install.packages("plm")

library(plm)

panel_GAbabies<-
pdata.frame(GAbabies,index=c("momid","birthord"))
```

In the pdata.frame function, the index identifies the clustering and time (or sequence) ordering in the data, respectively. Next describe the panel data and check which variables vary (`pvar`) within a mom and which do not.

```
summary(panel_GAbabies)

pvar(panel_GAbabies)
```

Which variables vary within a mom? Which do not? Notice any data errors? Finally, look at some of the individual birthweight trajectories to get a sense of the data. Before you do that restrict the plots to a reasonable number of moms (momid<2500).

```
GAbabies_sub <- GAbabies[GAbabies$momid<2500,]
```

Then use the ggplot2 library along with the following commands:

```
p <- ggplot(data=GAbabies_sub, aes(x=birthord, y=bweight, group=momid))
```

```
p+geom_line()
```

Relationship of birthweight to birth order

We'll start by looking at descriptive summaries. Does birthweight appear to be related to birth order?

```
        ggplot(GAbabies, aes(x=birthord, y=bweight))+geom_point()
```

This is a bit hard to judge. A slightly better method is to fit a smooth curve through the data. Recall our way of drawing a smooth curve:

```
ggplot(GAbabies, aes(x=birthord, y=bweight )) + geom_point() + geom_smooth(method= “loess”)
```

Or you can save the scatterplot and just add on the geom_smooth bit.

What does this tell you about the relationship? How comfortable would you feel treating birth order as a continuous variable and using a linear relationship?

Some people prefer tabular summaries over graphical. You can do this in R using the data.table package

```
        setDT(GAbabies)

        sum_bweight<- GAbabies[,.(mean_bweight=mean(bweight),sd_bweight=sd(bweight),N=sum(!is.na(b
        weight))), by=birthord]

        sum_bweight
```

Does this seem consistent with the graph?

Analyses

Let's proceed to a more formal analysis by regressing bweight on birthord:

```
          regr_fit <- glm(GAbabies$bweight~GAbabies$birthord)
          summary(regr_fit)
```

Is there a statistically significant relationship? Is there an alternate explanation for why birth order might not be causally associated with birth weight that could explored using this data? How would you check?

An alternate way to compare changes in birth weight with order is to look at the difference between the last and the first birth and conduct a paired t-test. Unfortunately, the data are not in the right format to easily do so. The data are currently listed with one observation per child (i.e., each child is a row in the data matrix). To subtract the birth weights for the first and last child we need one row per mom (why?). Fortunately, R has a simple command to rearrange the data. It is `reshape` and it can be used to take data in the current format (called the "long" format) and put it into a format with one data row per mom (the "wide" format). Or the reverse.

```
 GAbabies_wide <- reshape(GAbabies, direction="wide", timevar="birthord",
idvar="momid", v.names=c("bweight","timesnc", "momage", "delwght", "lowbirth"))
```

In this command we first say what type of reformatting we want (`direction wide` or `long`), then the variable that gives the order within a cluster (`timevar`), then the variable that identifies the clustering (`idvar`) . Finally we list the variables that are *not* constant within a cluster (in this case a cluster is a mom) in the `v.names` option. List a few lines of data to understand the new data format.

Now we can easily calculate the differences between the last and first birthweights and conduct a t-test.

```
GAbabies_wide$bwdiff <- GAbabies_wide$bweight.5- GAbabies_wide$bweight.1
t.test(GAbabies_wide$bwdiff)
```

How would you expect this to compare to the regression, given that the t-test ignores the three intermediate births? How do the t-statistics (and hence p-values) for testing birth order compare?

This "wide" format also makes it easier to understand the relationships between the repeated measures. Here is the graph showing the association of the five birth weights:

```
bweight_data<-
GAbabies_wide[,c("bweight.1","bweight.2","bweight.3","bweight.4","bweight.5")]
pairs(bweight_data)
```

And a numerical summary:

```
cor(bweight_data)
```

Virtually all analysis methods for clustered or repeated measures data will want the data in long format. So go back to using the GAbabies data frame.

Now let's analyze the data. For now, the important thing to know is that the command below performs a regression of birth weight on birth order, taking account of the clustering on mom. It is from the lme4 library.

```
mix_fit <- lmer(bweight ~ birthord +(1|momid), data=GAbabies)
summary(mix_fit)
```

How does the p-value compare to the t-test and the regression? Does it make sense? What is the relationship between the birth order coefficient in `lmer` and average value of BWDIFF? An alternative analysis method is with the `glmgee` command in the glmtoolbox library. Compare the results from that analysis:

```
gee_fit <- glmgee(bweight ~ birthord+initage, id=momid,
family=gaussian(link="identity"), data=GAbabies, corstr="exchangeable")
summary(gee_fit)
```

Diagnostics for the Georgia babies analysis

Go back to the mixed model fit of the data and check the assumption of linearity in birth order and initial age. Is a linear term adequate?

Obtain the residuals and predicted values. Standardized residuals (residuals divided by the residual standard deviation) have the advantage that values outside of 3 in absolute value are pretty extreme and make it easier to judge outliers, so calculate those as well.

```
residuals <- resid(mix_fit)
```

```
predicted <- predict(mix_fit)
residual_SD <- sd(residuals)
std_resid <- residuals/residual_SD
```

Plot the residuals versus the predicted values and generate descriptive statistics of the residuals.

```
ggplot(GAbabies, aes(x=predicted, y=std_resid )) + geom_point()
summary(std_resid)
```

Are there outliers? If so, drop them and rerun the analysis and assess if they qualitatively change the results.

OAI dataset

Let's move on to a different data set from the OAI study. Open the OAI dataset. This comes from the OAI study (https://nda.nih.gov/oai/). The purpose of this analysis is to show how more standard analyses fit with some simple repeated measures analyses. The variables in the dataset are:

1. ID (participant ID)
2. Visit (baseline = 0 months, visit 1 = 12 months)
3. Age at the visit
4. Sex of the participant
5. xr_koa = evidence of knee osteoarthritis on Xray at baseline
6. sx_koa = xr_koa with reported symptoms
7. WOMAC pain score (measures pain on a scale from 0-50, higher being worse)
8. Body mass index (BMI).

We are going to compare the change in pain scores in men and women. First get a sense of the data by generating some descriptive statistics.

```
sum_oai_pain <-
   oai[,.(mean_pain=mean(womac_pain),N=sum(!is.na(womac_pain))), by=.(sex, visit)]
```

What is the change in pain score in women? In men? And what is the difference in the changes? Test your statistical intuition: do you expect the changes to be statistically significantly different between men and women?

Analysis of difference scores

A simple and effective analysis is to calculate the difference scores within a person and compare those using a t-test. Reshape the data to wide, calculate the difference scores and compare the difference scores between men and women using a t-test. How does this compare to the descriptive statistics?

(after reshaping wide)

```
oai_wide$ch_womac
          <- oai_wide$womac_pain.12 - oai_wide$womac_pain.0
     t.test(oai_wide$ch_womac ~ oai_wide$sex)
```

Analysis using hierarchical methods

Go back to the long version of the dataset and use `lmer` to perform a hierarchical analysis:

```
  mix_model <- lmer(data=oai, womac_pain ~ 
factor(visit)+factor(sex)+factor(visit)*factor(sex)+ (1|id))
```

Why did we need to include the interaction? How does this compare to the t-test? Compare the `gee` command to `lmer` and the t-test.

Analysis adjusting for baseline

Some people argue that, instead of analyzing change scores, one should adjust for the baseline value. Let's go back to the wide format and try this.

As a basis for comparison consider another way to do the t-test above, namely with a regression command:

```
     glm(data=oai_wide, ch_womac ~ factor(sex))
```

Now check to see what we get with two ways to adjust for baseline pain score. First regress the 12 month score on sex and adjust for the baseline value:

```
glm(data=oai_wide, womac_pain.12 ~ factor(sex) + womac_pain.0)
```

Some people like a minor variation in that they use the change score as the outcome and adjust for baseline values:

```
glm(data=oai_wide, ch_womac ~ factor(sex) + womac_pain.0)
```

Focusing on the sex effect, how do these analyses compare to each other and to the above analyses?

Morals:

1. In this simple scenario, the longitudinal analysis gives exactly the same results as the t-test on the difference scores. Reassuring. In this simple situation, why do anything else? In more complicated situations, however, like with missing data and multiple visits, the longitudinal analysis uses exactly the same command for two or more time points and is much easier. It is also a way to deal with observations that are unequally spaced in time.
2. Adjusting for the baseline value of the outcome is rarely a good idea in an observational study. It gets you away from analyzing changes over time and can generate spurious results. Use with care!