ATCR Lab, 5/7/09

Repeated Measures 2

Lab Summary: In this lab we will learn about the use of `xtmixed` for continuous outcomes.

1. Re- analysis of the fecal fat data.

First, load the fecal fat data (fecfat.dta) and reproduce the results in lecture:

```
xi: xtmixed fecfat i.pilltype || patid:
```

Derive the overall pilltype test using the `testparm` command: `testparm _Ipillt*`

The fecal fat data can be analyzed using a two-way ANOVA, with factors of pilltype and patient. This can be performed either using the ANOVA commands or regression (as we learned in Biostat 208). Use the regression command to do this:

```
xi: regress fecfat i.pilltype i.patid
```

followed by the `testparm` command. How do the p-values for the overall tests compare? Why are they different?

2. The OAI data

Load the "oai thru 18 months" data. Our goal is to look over 18 months to see if the changes over time in the WOMAC pain score are the same in men and women. First generate some descriptive statistics and a graph:

```
table visit sex, c(mean womac sd womac n womac)
bysort visit sex: egen mean_womac=mean(womac)
twoway (connect mean visit if sex==1)(connect mean visit if sex==2)
```

Next do a formal test

```
xi: xtmixed womac i.sex*i.visit || id:
testparm _IsexX*
```

Next do a residual plot (residuals versus predicted). Anything to be concerned about?

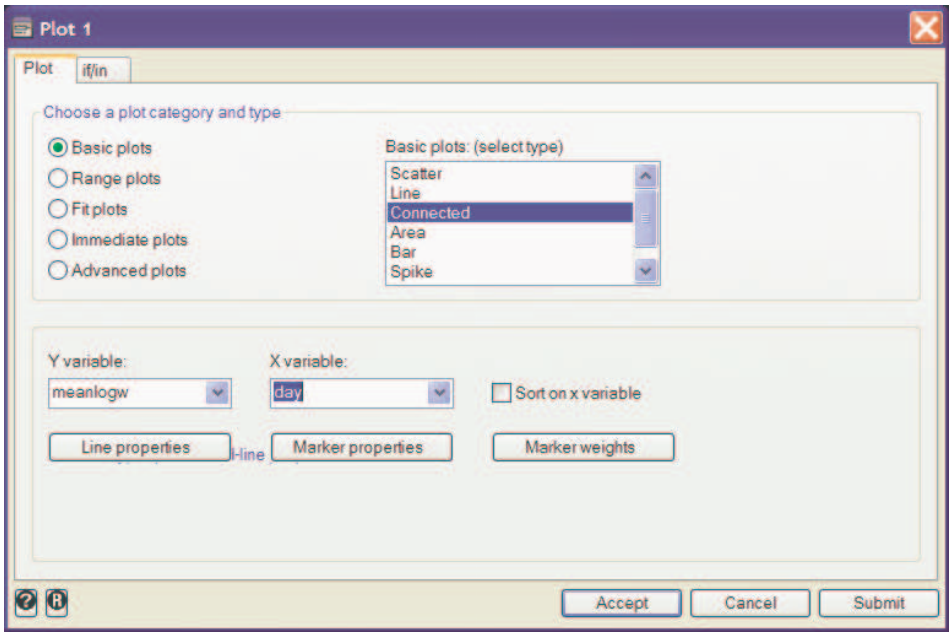
```
predict resids, residuals
predict preds
scatter resids preds
```

3. The mouse weight data

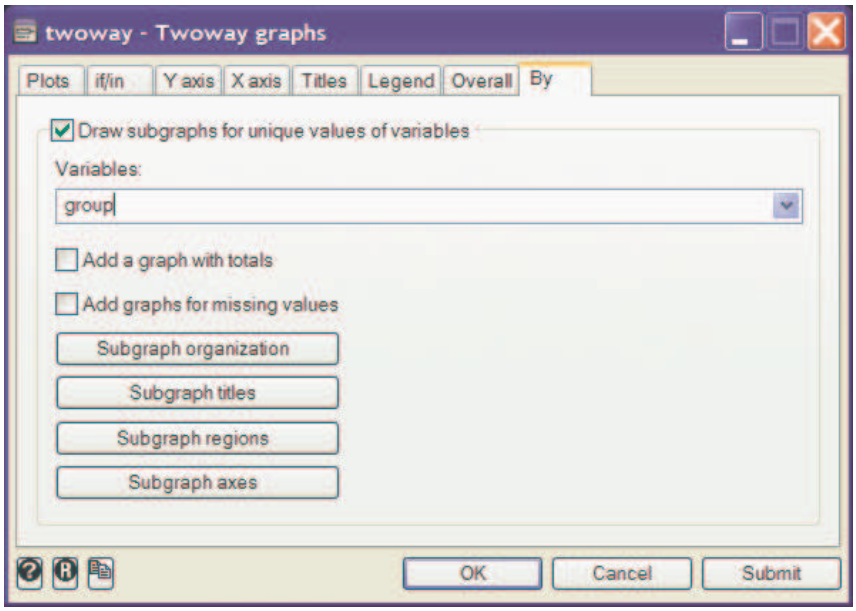
Next, let's analyze the mouse weight data. Load the data (mousewght.dta). First we will get a sense of the relationship between log weight, group and day. To do this generate the mean values by group and day and then plot the results:

```
. bysort group day: egen meanlogw=mean(logw)
```

The easiest plot to make is to use the Graphics menu: select "Twoway graph". Then choose "Create" type of "Connected" and fill in the X-variable (day) and the Y-variable (meanlogw) and click on "Accept".



Next go to the "By" tab, click on "Draw subgraphs for unique values of variables" and under "Variables" insert "group". Then press "Submit" (I like to use "Submit" instead of "OK" so you can change the graph without re-opening the menu).



What does the plot tell you about the relationships? Day2 in the dataset is day^2. Why do we need that?

Now formally analyze the data. The mice were randomized to treatment group, so the groups should all have the same intercepts. So the key question is whether the weight gains over time are different by group. Therefore the primary question revolves around the group by time interaction.

We start with an overly simplistic model and work our way up:

```
xi: xtmixed logw i.group day day2 i.group*day || mouse:
```

Use the `testparm` command to test if the group by day interaction is statistically significant. Next we will fit a more realistic model, allowing for mouse-specific slopes over time (as we noted in lecture):

```
xi: xtmixed logw i.group day day2 i.group*day || mouse: day, cov(un)
```

In the above command we are allowing mouse-specific intercepts and slopes (over day) and the random intercepts and slopes have an unstructured correlation matrix. How does the `testparm` command compare to the simpler model? Why do you see differences? How could we have checked the results of the simpler model if you hadn't thought of running the more complicated one?

These models *are* complicated and it is sometimes hard to figure out what the fitted model is telling you. One way is to plot the predicted values (the sort command is to make the connected points look right and you may want to cut and paste the `twoway` command into the command window so you don't have to type it).

```
predict prd
sort day mouse
```

```
twoway (conn prd day if gr==1) (conn prd day if gr==2) (conn prd
day if gr==3) (conn prd day if gr==4) (conn prd day if gr==5)
```

How would you describe the fitted model?

Let's contrast using the `xtgee` command:

```
xi: xtgee logw i.group day day2 i.group*day, i(mouse)
```

How does the overall test of the interaction compare? Now use the robust option (add `robust` at the end of the command above) and see if there are changes.