light.sas: Explanation

Goals of code:

- Do analysis by subgroups
- Fit models to both subgroups (ANCOVA, heterogeneous slopes regr.)

This lab uses the light.txt data set. This is case study 9.1 in the book. The study examined two groups of plants (E: early and L: late) at a range of light intensities. The response is the # of flowers produced. We want to fit regression lines that include both group and intensity.

Do an analysis by subgroups:
There are three ways to do this:
   Create a SAS data set containing the desired subset
   Create the subset "on the fly"
   Use a by statement to repeat the analysis for each subgroup
Each of those is described in turn.

Creating a subset: `if ;`
meat.sas in Lab 6 introduced (near the end) how to use a subsetting if or a where statement to do an analysis on a specific subset of observations. The meat.sas explanation document described many of the logical operators that you can use in if or where to specify the desired subset. We remind you about this here. An if statement without any "then" part subsets the observations. So `if time='E';` retains all observations in the Early group (with a time value of "E") and omits anything else.

Analyze a subgroup "on the fly": `where` statement
Most of the SAS procs accept a `where` statement that defines the subset of observations to be included in the analysis. Where is followed by a logical expression that defines what you want to include in the analysis.

Analyze all subgroups: `by` statement
Most SAS procs accept a `by` statement to indicate that an analysis is to be done separately for each unique value of the variable in the by statement. You can name more than one variable; if so, each unique combination is analyzed separately.

The by statement requires that the data be organized into groups of contiguous observations. `by time;` requires that all the time="E" values are together. They are followed by all the time="L" values. The easiest way to get this is to sort the data set by the "by variables" first. That's what the proc sort; does. SAS will give an error if it finds observations out of order. If you get that error, sort the data set and rerun the analysis proc.

Creating indicator variables automatically: `class time`
We have used class statements to tell SAS that a variable defines groups. When a class statement variable is used in a model statement, SAS automatically creates indicator variables for each level of the grouping variable.

Fitting ANCOVA models: With options

```
  class time; model flowers = time + intensity / solution clparm;
```

Printing regression coefficients: `model / solution;`
An ANCOVA model has groups with different intercepts but the same slope. We will let SAS automatically create indicator variables for the two time groups, then use those to define different intercepts.

The `solution` option (after a `\` at the end of the model statement) requests that SAS prints the regression coefficients. These are printed by default when there are no class variables and omitted by default when there are. Specifying `/solution` asks SAS to print the coefficients.

You see that SAS tells you that the X'X matrix is singular (i.e., the X matrix is not full rank). You also see that the coefficient estimates for the intercept, time E and time L are labelled with a B. That's because these are not unique. The values reported by SAS are those when the last group (time = L) is set to zero.

Getting confidence intervals for parameters: `model / clparm`;
If you add the clparm option (or `/ clparm` if there aren't any other options), SAS adds the 95% confidence interval endpoints to the information about each parameter estimate.

Note: If you request more than one option, all go behind a **single** `/`.
E.g. `model / solution clparm;`. If you add a second /, e.g., before clparm, SAS will give you a syntax error.

Fitting ANCOVA models with more interpretable intercept values:

```
model flowers = light time / solution noint;
```

The intercepts estimated using the previous model are combinations of the overall intercept (β_0) and the difference between the group-specific intercepts. This is a consequence of including the overall intercept in the model. Remember our discussion of overparameterized models. Same consequences here. If you omit the overall intercept, the estimates are the group-specific intercepts.

You tell SAS to omit the intercept by adding the /noint option at the end of the model statement.

The intercepts reported in the Solutions part of the proc glm output are:
Early group: time E: 83.46
Late group: time L: 71.30

You see that these estimates are not labelled with a B. That's because these estimates are uniquely identifiable.

Fitting a heterogeneous slope regression:

```
  class time; model flowers = time + intensity + time*intensity;
```

This model has groups with different intercepts and different slopes. The model statement includes time to generate different intercepts for each group. We also need the interaction (time by intensity) term to generate different slopes for each group. SAS can create that interaction variable automatically. The syntax is the two variable names separated by a `*` without any spaces.

SAS provides a short-hand for "variables and their interactions". That is `time|intensity`, which SAS expands into `time intensity time*intensity`, i.e., the two variables and their interaction. When you use `|` you don't control the order of the terms. I prefer to specify the order of terms, so I usually don't use `|`.

Fitting a heterogeneous slope regression with more interpretable estimates:

```
 model flowers = light time*light / noint solution;
```

We get group-specific intercepts and slopes by suppressing both the overall intercept and the overall slope. We suppress the overall intercept by adding the `/noint` option to the model statement (just as done above). We suppress the overall slope by writing a model **without** light.

The results in the Solutions part of the proc glm output give us the following

equations (rounding coefficients a bit) for the fitted lines:
Early group: $\hat{Y}_i = 83.15 - 0.0399\text{light}$
Late group: $\hat{Y}_i = 71.62 - 0.0411\text{light}$