Class Exercise 8

This exercise is based upon Chapter 9 of the SAS Advanced Certification Prep Guide. This should be a straightforward review of the material covered in the text and in class.

Begin by importing the simulated Medicare data set from Excel into WORK.meddb. Run the following commands to refamiliarize yourself with the data.

proc print data=meddb (obs=20);

format dos date7. SSN SSN11. claim dollar14.2;

run;

Next let’s use the automatic macro variables to find out a little about your version of SAS. Run the code below and follow it with the PROC PRINT you just ran. Comment.

title "Medicare Data";

title2 "Data saved as %cmpres(&syslast)";

footnote "I am using Release &sysver on a &sysscp operating system";

footnote2 "My user id is &sysuserid";

Remember that we used the macro function %CMPRES to handle a possible centering problem with TITLE2, though this is a problem more particular to LISTING output. It also appears to ensure that the titles and footnotes are actually printed.

When we used MONTH as a macro variable in our notes, we had to introduce a new application of PUT to handle both numeric and character versions of month. A simple, but less elegant solution would define two macro variables:

%let month=10;

%let cmonth=OCT;

Create a permanent SAS data set in your course directory for October 2000 records (all records are from 2000, so you do not need a macro variable for year) with the following commands. You will need to change the first line to match your usual course directory. Use a PUT statement to confirm that SAS created &libname correctly.

%let libname=/home/grego1/STAT 541;

libname perm "&libname";

run;

Now run the following code. Some of this is just clean-up (i.e., removal of the leftover footnotes)—why did I not need to delete footnote2?

options symbolgen;

title "Medicare Data for &cmonth.2000";

footnote;

data perm.meddb&cmonth;

set meddb;

month=month(dos);

if month=&month;

drop month;

run;

proc print data=perm.meddb&cmonth (obs=10);

format dos mmddyy9. SSN SSN11. claim dollar14.2;

run;

Note that we needed to extract October records using the MONTH() function. I wasn’t sure whether SAS would be able to execute the subsetting IF properly since I was creating the new variable month “on the run”, but it actually worked properly.

I would like you to try to run the title by first removing the period, and then by replacing it with a space. Comment on the different results. What differences do you see when using the period instead of a space?

In a similar vein, graduate students should modify the code so that data is written to and printed from a permanent SAS data set named OCTmeddb rather than meddbOCT.

Suppose we wanted to extract records based on Type of Service (TOS). To make sure we had no problems with matching case, both when selecting cases, and when searching records in MEDDB, we can use the %UPCASE command.

options symbolgen;

%let tos=%upcase(a and B);

data tos;

set meddb;

month=month(dos);

if upcase(tos)="&tos";

run;

title "Type of Service %UPCASE(&tos) Records";

proc print data=tos (obs=10);

format dos mmddyy9. SSN SSN11. claim dollar14.2;

run;

Comment on the initial %LET, the subsetting IF statement, and the use of &TOS in the title.

None of the applications here needed %QUPCASE because no special symbols were present in the records. It would be interesting to use %SCAN to see whether & had been used in place of “and”, then make an appropriate substitution, but that would likely be beyond the scope of this exercise.