Class Exercise 13

This exercise is based upon Chapter 17 of the SAS Advanced Certification Prep Guide. We will primarily focus on use of CNTLIN and CNTLOUT to study the creation of a SAS format from a SAS data set in an intuitive context.

We will be working with the Fall 2009 and Fall 2010 enrollment data. I will assume that the data has been loaded into my WORK directory, though I will be using a FORMAT library in my main course directory.

The Fall 2009 database has a column, MAJOR, that seems ready-made for creating a format. MAJOR contains a 3-digit code followed by a hyphen, then a text descriptor—all the elements we need to create a format! Let’s start by saving the different portions of MAJOR as values and descriptors:

data fmajor;

set fall2009 (keep=major);

start=input(substr(major,1,3),3.);

label=substr(major,5);

type= " N ";

fmtname= "majorfmt ";

drop major;

run;

Note that we created a numeric format here. I could certainly see eliminating the use of the INPUT statement to save START as a character variable instead. We don’t need to save END, since each major has a unique major code. We need a unique set of assignments, so we eliminate duplicates, then convert the data set to a format in our course library:

proc sort data=fmajor nodupkey;

by start;

libname stat541 "/home/grego1/stat541 ";

proc format cntlin=fmajor lib=stat541;

run;

Confirm that the program ran as advertised. We could eliminate the missing code, but that can actually be useful when formatting, so we’ll keep it in.

Graduate students should indicate how they would eliminate duplicates from FMAJOR using PROC SQL.

We can check Fall2010 to see if it has additional majors we should add to our library. The simplest way to do so would be to use a Chapter 4 technique to print a list of majors in Fall 2010 that do not appear in Fall 2009 (make sure you have saved Fall 2010 to your WORK library as fall2010). The following PROC SQL clause should generate a list of a dozen such majors:

proc sql;

select major from fall2010 except select major from fall2009;

quit;

To get started, let’s output our format into a data set (FMAJORREV) that we can then update—we have to be sure to select only the format MAJORFMT from our format library:

proc format library=stat541 cntlout=fmajorrev;

select majorfmt;

run;

I found I needed to make a couple changes to FMAJORREV to prevent problems when I add new records—specifically, I had to simplify the variables list, and a couple of the FALL 2010 labels are long, so I had to adjust the length of LABEL:

data fmajorrev;

format label $40.; \*This has to come before SET;

set fmajorrev (keep=start label type fmtname);

run;

I thought about adding the additional values using MODIFY, but it didn’t really seem in the spirit of Chapter 18 methods since they were all new values. I instead reverted to PROC SQL. Note that START and END are actually character values in fmajorrev, so I have to specify START as a character value when using INSERT INTO.

proc sql;

insert into fmajorrev

(start,label,type,fmtname)

values('101', 'African-American Studies', 'N', 'MAJORFMT')

values('105', 'Art History', 'N', ' MAJORFMT ')

values('119', 'Creative Writing', 'N', 'MAJORFMT')

values('161', 'Clinical-Community Psychology', 'N', ' MAJORFMT ')

values('169', 'Applied Clinical Pscyhology', 'N', ' MAJORFMT ')

values('176', 'Sciences', 'N', ' MAJORFMT ')

values('191', 'Studio Art', 'N', ' MAJORFMT ')

values('322', 'Business Education', 'N', ' MAJORFMT ')

values('335', 'Physical Therapy', 'N', ' MAJORFMT ')

values('597', 'Health Services Policy and Management', 'N', ' MAJORFMT ')

values('852', 'Women’s and Gender Studies', 'N', ' MAJORFMT ')

values('932', 'International Hospitality & Tourism Mgmt', 'N', ' MAJORFMT ');

quit;

And now we would save this as an updated format in our library using PROC FORMAT. We run PROC FORMAT a second time to make sure MAJORFMT was successfully over-written:

proc format cntlin=fmajorrev lib=stat541;

proc format lib=stat541 fmtlib;

select majorfmt;

run;

Now copy the format to WORK (we could also set a search path to include STAT541) and create a simple SAS data set to confirm that the formats worked:

proc catalog catalog=stat541.formats;

copy out=work.formats;

select majorfmt.format;

run;

data a;

input Major;

datalines;

101

001

586

940

220

135

159

.

100

101

135

100

;

proc freq;

format Major majorfmt.;

table Major;

run;