Stat 604

Assignment 5 - SAS

You should have all of the information you need to complete this assignment by viewing Lectures SAS01 through SAS08.

Perform each of the exercises listed below. To the extent you have been taught to control it, your output should match that in the PDF file posted on eCampus. Download the **ok_schools.sas7bdat** data set from eCampus and save it in your homework folder. Write a libname statement in your program that accesses this folder. Make the libref read only so you will not accidentally overwrite the data. Familiarize yourself with the data contained in the downloaded data set.

1. Begin your program with the required header, filename, and libname statements. As always, your program must include comments in the appropriate places. In addition to the comment boxes you have been using above data and proc steps, use single comment lines to identify the sub-steps that are being accomplished within the data step. This requirement will be in place for this and subsequent SAS assignments.
2. The **ok_schools** data set was created from Oklahoma Public School data downloaded from the National Center for Education Statistics. Since the five grade variables contain character data, they are character variables in SAS instead of numeric. You are to create in your personal SAS library folder a new permanent data set that is a “cleaned up” version of this data set. The cleaning process will update and transform some of the variables as described below:
    a. All of the county names end with the word COUNTY. This is redundant and takes up extra space. Create a new variable to reduce the length of the County variable by 7 characters and use one or more text manipulation functions to remove the word COUNTY from each value. The variable name should also be County in the output data set. Use a multi-step method similar to the process of converting a character variable to a numeric variable of the same name.
    b. Convert the Grade8 through Grade12 variables from character to numeric. Incorporate **SELECT groups** in this process. If the original value is n/a or *, set the new value for the Grade variable to missing. Since we are not changing the default value for missing data, you may use an assignment statement to set these values to **.**(period). Otherwise, use a function to convert the original value to numeric so that no note is written to the log about the conversion. All select groups in this assignment must be constructed so that there will never be an error generated should none of the conditions be met. There should also be no messages about invalid numeric data converted to missing.
    c. The names of several cities are misspelled or not standardized to a common value. Use a **select group** to replace the original values of City with the corrected values as listed below. If none of these conditions are met, the City variable should be unchanged and the select statement must not generate an error or warning. Since the occurrence of all these values is small, do not worry about trying to order the statements in the select group for efficiency.
        i. Change 'CHUOTEAU' to 'CHOUTEAU'
        ii. Change 'OKC' to 'OKLAHOMA CITY'
        iii. Change 'JENKS' to 'TULSA'
        iv. Change 'MUSKOGE' to'MUSKOGEE'
        v. Change 'RUSHSPRINGS' to 'RUSH SPRINGS'

vi. Change 'SEMIONOLE' to 'SEMINOLE'
vii. Change 'SO. COFFEYVILLE' to 'SOUTH COFFEYVILLE'
viii. Change 'WOOWARD' to 'WOODWARD'

d. The county of Alfalfa is a small county with a number of scattered schools. If the county is ALFALFA, set the city to CHEROKEE for the purpose of grouping these schools together in future analysis.

3. This step draws heavily from the concepts covered in the chapter on Summarizing Data (Prog2-Ch3). We want to use the “clean” data set created in the previous step to project high school enrollment next year and compare it to the current high school enrollment. Part of this process will be to replace seniors with eighth graders to compute next year’s enrollment. The problem is that many schools do not house all 5 grades in the same school. Some schools have only seventh and eighth graders while others have eighth and ninth graders together. We are going to overcome this problem by combining all schools together at the city level creating a city summary for each grade. You are to collapse the data from the “clean” data set in a new temporary data set with one row per city much like you would summarize data by department. (TIP: You may find it helpful to wait until you have your summaries working before you put in code to remove observations and variables. That way you are better able to see what is happening from observation to observation.)

   a. Use a sort procedure to sort the clean data set in place as needed to summarize the data.

   b. Since the original variables of school and each of the grades only reflect the data in the last row of each city they are not needed in the resulting data set. Remove them in the most efficient manner possible. Use a name prefix variable list in the statement to remove the 5 original grade variables.

   c. Create 5 new summary variables – one for each grade. Use a naming pattern that is unique enough so that you can use a numbered range to refer to these variables as a variable list. Apply permanent labels “Eighth Graders” through “Seniors” to the new summary variables. Use summary statements to accumulate the total number of students in each grade into your new variables.

   d. Create a new variable labeled Current Enrollment that is a sum of the summaries of grades 9 through 12 for each city. Use a numbered range in your sum function.

   e. In the same manner, create a variable labeled Projected Enrollment that is a sum of grades 8 through 11.

   f. Create a variable named Change that is the percentage change of the projected enrollment compared to the current enrollment. Apply a permanent format to the new variable so it is displayed as shown in the output posted on eCampus.

   g. Output only those cities in which there is data for all 5 grades.

4. Open the PDF destination and choose one of the options so that bookmarks are not created for this output.

5. Print the descriptor and data portions of both data sets created in this assignment. Be sure labels are shown on your printed output.

6. Convert the program and log to PDF files and submit them to eCampus along with your SAS output.