EPBI 414 (Fall 2016) - Assignment 8

Introductory SAS

Overview

This homework consists of two parts. Part 1 involves answering two short questions based on the lecture, while Part 2 is a SAS programming exercise.

Logistics

Submit your assignment in a .zip file labeled in the following manner:

<Case ID>_EPBI414_Fall2016_A9.zip

So, if your Case ID is tar9, you would submit the following zip file:

tar9_EPBI414_Fall2016_A9.zip

Part 1

Submit a written document with your answers to the following questions.

1. We've discussed the importance of following a "style guide" or set of coding conventions as we've begun to do regular programming in class. Give three examples of SAS coding conventions you can see in the example code or in the lectures. [6 points]

2. SAS has four data-centric purposes that were described in the lecture. Name each of these, and give a practical research example of each one. [4 points]

Part 2

On Blackboard, you will find a file labeled `luggage_data.sas7bdat`. This contains a small amount of sample data for the homework, based on some hypothetical pieces of luggage. You should write a single SAS program which fulfills each of the requirements below:

1. In general: your program should be properly commented and should include a preamble describing your name, the name of the program, and what the program is intended to do. It should be written in a manner that makes the code clear and easy to read. Use the example given on the unit webpage as guidance. [10 points]

2. Using the OPTIONS command, make sure your output does ***not*** print centered, and turn off numbering of your pages. Also, remove the date from your outputs.[6 points]

3. Next, place the file `luggage_data.sas7bdat` somewhere on your system, and declare a LIBREF named `LUGDATA` so you can access it. Once you have done this,

create a temporary dataset from `LUGDATA.luggage_data`, named `WORK.luggage_data`. [4 points]

4. Using PROC PRINT, print out the contents of your new temporary dataset, `luggage_data`. Using the options for PROC PRINT, add an extra line between each observation that is printed. [4 points]

5. You'll notice that each piece of luggage has different parameters. To start, organize the luggage by name. Using PROC SORT on `luggage_data`, sort the luggage in alphabetical order, first by the manufacturer, then by the name of the luggage. Have your output write to the same temporary dataset as before (`luggage_data`). [4 points]

6. Now, using PROC SORT, sort `luggage_data` so that the least expensive luggage (in price per cubic inch, or `ppcuin`) is the first row. ***Do not overwrite your existing dataset.*** Instead, use the options in PROC SORT to create a new temporary output dataset, `luggage_data_ppcuin`. [6 points]

7. Using PROC CONTENTS, examine the permanent version of `luggage_data`. When you run PROC CONTENTS, ensure that the variables in the output are sorted by the order they appear in the dataset, rather than by name. [4 points]

8. Each manufacturer makes a more expensive piece of luggage. To identify this item, use PROC SORT to sort `luggage_data` alphabetically by manufacturer, and then by decreasing price within each manufacturer. ***Do not overwrite your existing dataset.*** Instead, use the options in PROC SORT to create a new temporary output dataset, `luggage_data_sorted`. [6 points]

9. Using PROC PRINT, print out the contents of your new temporary dataset, `luggage_data_sorted`. Using the options for PROC PRINT, add an extra line between each observation that is printed. [4 points]

10. Save a permanent copy of `luggage_data_sorted` to the `LUGDATA` library. [2 points]

Save the contents of your log to a file ending in `.log` and the contents of your output to a file ending in `.lst`. Ensure that you are not using HTML output. Include these files, along with your program file, in your submission.