STAT 440 – Homework 5

Students are encouraged to work together on homework. However, sharing or copying any part of the homework is an infraction of the University's rules on Academic Integrity.

Final submissions must be uploaded to our Compass 2g site on the Homework page. No email, hardcopy, or late submissions will be accepted.

Getting the program file ready

a. Create a folder on the hard drive with the following pathname – C:\440\hw5. Save all data files accompanying this assignment in that folder. If you cannot create the folder because you are working on a university computer and don't have permission, create the …\440\hw5 folder elsewhere.

b. Assign the library reference **hw5** to the folder 'C:\440\hw5'. Use this library as your permanent library for this assignment. If you could not create the folder, assign the library reference **hw5** to your …\440\hw5 folder.
Note: If you are using a folder other than 'C:\440\hw5', you must change any pathname references in your program file to 'C:\440\hw5' before submitting your homework.

Submitting your work to Compass 2g

You are to submit two (and only two) files for your homework submission.

1. Your SAS program file which should be saved as **HW*n*_*YourNetID*.sas**. For example, my file for the HW5 assignment would be HW5_dunger.sas. All program statements and code should be included in one program file.

2. Your Report including all relevant output to address the exercises. For this homework, use ODS to send your results to a Rich Text Format (RTF) file called ***YourNetID*_HW*n*.rtf**. Only include your final set of output. Do not include output for every execution of your SAS program. Use the template file **hw5 template.sas** as your guide.

You have an unlimited number of submissions, but only the last one will be viewed and graded. Homework submissions must always come as a pair of files, as described above.

1. You will be working with the text file **blood_with_errors.txt** which contains the blood test results of 1000 adults. This is not the original dataset, but rather, one that I have tampered with to include errors.

   Description:

   - The fields are separated by spaces, and the records have variable length (free-format).

Variable Name	**Description**
Subject	Subject's ID number
Gender	Possible values are Male, Female
BloodType	Possible values are A, B, AB, O
AgeGroup	Possible values are Young, Old
WBC	White blood cell (WBC) count defined as the number of WBC per microliter
RBC	Red blood cell (RBC) count defined as the number of millions of RBC per microliter
Chol	Total cholesterol

Data Entry

a. Write a DATA step to read the values into SAS. The output data set is to be a temporary SAS data file called **blood_with_errors**. Include code to ensure that variables have appropriate length, informat, labels, and format (as necessary).

   - This can just be a "first attempt" DATA step to just get the raw data into SAS. It does not have to fix all value errors; it just needs to get all 1000 records into SAS.
   - Include code to ensure that variables have appropriate length, informat, labels, and format (as necessary).

b. Print the descriptor portion of your new SAS data file once completed. There should be 1000 observations. (Include results in the HW Report.)

Data Validation

c. After your "first attempt," examine the log for invalid data errors or other notes identifying issues. (Include just the log notes from the DATA step execution that results in 1000 observations in the HW Report by using copy-paste. Do not include the entire session log!)

d. Write any and all necessary steps to check the values of the newly created SAS data set. For each table you create, write one and only one sentence for each variable that has an issue. For example,…

   - Subject has missing and duplicate values.
   - WBC has at least one observation out of range.

Some notes and things worth checking:

- Every variable has at least one invalid data value.
- There are many missing values for WBC, RBC, and Chol. In this data set, you would have no idea what the correct value for each individual missing value should be, so don't worry about updating those.
- Recall that in the "old days," data programmers would use a series of repeating 9's to denote missing values. Today, this method is considered antiquated and such values should be adjusted.
- There are some invalid numeric values that are just simple typos, and you can offer a well-educated guess of the correct value. For example, the value might have a misplaced decimal point. Consider these boundaries when conjecturing on the correct values for this data set.
    - For WBC, 4000-12000 white blood cells per microliter (mcL) may be considered valid for this data set.
    - For RBC, 1.5 to 9.0 million cells per microliter (cells/mcL) may be considered valid for this data set.
    - For total cholesterol, an array of values are possible but values between 50 and 400 may be considered valid for this data set.

(Please include only one relevant table for each variable (i.e., those that identify possible data issues) in the HW Report.)

Data Cleaning

e. Write a DATA step that programmatically corrects all invalid data issues according to your table in part (e). The output data set is to be a permanent SAS data file called **blood_fixed_*YourNetID***.

f. Run the same validation steps you wrote in part (d) again and verify that all invalid data issues that you can control have been corrected. You should not include any sentences or additional notes for this output. Again, note that missing values for WBC, RBC, and Chol are expected. (Include only relevant tables and results, i.e., those that identify possible data issues, in the HW Report.)