ADSA: Creating an analysis data set for a paper looking at CHD and magnesium, Part I, mostly focusing on arriving at the correct set of observations

Preliminary notes: This assignment is much easier than the length of this document would suggest! This document includes condensed background information about the data and relatively detailed instructions for the assignment.

Introduction to this week's work: This assignment and the next one focus on two components of creating an analysis data set. This particular analysis data set is to be used for a manuscript looking at the relationship between magnesium and coronary heart disease (CHD). In today's assignment, you will focus on arriving at a data set that contains the correct collection of observations and using an external SAS macro to document and inform that stage of the process. In the next assignment, I will give you the finished analysis data set and have you focus on a different important activity, which is making sure that the derived variables are correct.

Our overall goal: We are preparing an analysis data set for a manuscript looking at whether the mineral magnesium – both in the diet and in the blood – affects coronary heart disease (CHD) and some conditions possibly associated with CHD, such as hypertension, diabetes, and carotid artery wall thickness.

Introduction to today's tasks: This assignment uses the CHD collection of data. To turn in: two or more programs and their logs and output. Before working on this assignment you should view analysis data set videos 1 and 2, read pp. 1-6 of the analysis data set course notes, and read the paper *Quick Analysis and Visualization of Excluded and Missing Data*, which describes a powerful macro. A former 669 student Polina Kukhareva conceived and wrote this macro while she worked on one of our CSCC studies, and she presented her paper at the Western SAS Users Group conference meeting in San Jose in 2014. The paper and the slides presented at the conference are posted on Canvas.

Organizing your work: This assignment has five steps. You could do each step in a separate program, or you could do steps 1, 2, and 3 (and possibly optional step 5) in the same program. You should definitely do step 4 in its own program – I recommend not combining it with the other steps.

Study background information: The data sets for this assignment come from the initial participant visit in a longitudinal study that was designed to try to understand factors that affect occurrence of heart disease. This study, involving 16,000 people at four clinical sites, has been the basis for hundreds of papers that contribute to our scientific knowledge. The study began in the mid-1980's and is still going strong today.

In each participant's initial visit to the clinic (which we call Visit 1), blood was drawn, many measurements were taken, and several forms were filled out in an interview. The data sets provided to you for this assignment contain only a fraction of the data collected at the visit, but still this exercise will be realistic and provide you with a good perspective on what it takes to create an analysis data set.

Data sets: Unless otherwise specified, all data sets in the CHD collection contain variable ID, which is a participant's unique study ID across all visits, and they contain only one record per ID (the *medications_long* data set is an exception). Variables ID, Gender, Race, and Age have no missing values, but all other variables might have missing values for some participants.

Core The *Core* data set contains a collection of demographic and miscellaneous variables that you will supplement with variables from the other data sets. Besides ID, variables include

Age	Age at visit
BMI	Body mass index
Gender	Sex of participant (F, M)
HOM10D	Was stroke ever diagnosed? from the Home Interview Form (Y, N, U for Unknown)
HOM55	Current occupational status from the Home Interview Form
DTIA90	Do you presently drink alcoholic beverages? from the Dietary Intake Form (Y, N)
DTIA91	Have you ever consumed alcoholic beverages? (Y, N)
DTIA96	# glasses of wine per week
DTIA97	# beers per week
DTIA98	# drinks of liquor per week
Fast8	Fasted 8 hours or more before visit (1=yes, 0=no)
GlucoseIU	Blood glucose level in International Units
InsulinIU	Insulin level in International Units
LDL	Serum LDL cholesterol level
PrevalentCHD	Prevalent coronary heart disease (1=yes, 0=no)
Race	B for Black, W for White, A for Asian, I for American Indian
RoseIC	Intermittent claudication by Rose criteria (1=yes, 0=no)
TotCal	Total calorie intake in kcal/day
VisitDate	Date of visit

Nutrition Besides ID, the *Nutrition* data set contains only Magnesium (Dietary magnesium (mg)).

Measurements Besides ID, the *Measurements* data set contains

BloodDrawDate	
CHMA16	Raw insulin value
CHMX07	Original glucose value
DBP	Diastolic blood pressure
Magnesium	Serum magnesium

Six carotid artery wall thickness variables: LBIAAV45, LINAAV45, LOPAAV45,
RVIAAV45, RINAAV45, ROPAAV45

Medications_wide The participant was to bring to the clinic packages for all medications he or she was currently taking. The *medications_wide* data set is based on information taken from those containers and also some questions. Contains at most one record per ID.

DrugCode1-DrugCode17 Up to 17 standard codes for medications taken by the participant, based on the medication containers. These are character variables and should be treated as such in your code.

MSRA02 A question asked only if the participant brought no medication packages – F if forgot to bring, T if taking no medications

Medications_long Same information as *medications_wide* but with one observation per medication rather than all medications on the same participant record. A participant with values for DrugCode1, DrugCode2, and DrugCode3 in *medications_wide* will have three records in *medications_long*. Thus, can contain multiple records per ID. DrugCode is a character variable.

Here's an outline of what you will do for this assignment ADSA (don't worry, details follow):

1. Use the *medications_wide* data set to create two desired variables.
2. Combine the *core* data set with the *nutrition* data set, the *measurements* data set, and the data set from #1 to make a preliminary master data set. Save this data set permanently for use in step 4 below.
3. Apply some exclusion criteria to the data set from #2 to arrive at the set of observations to be used for manuscript analyses. You will produce output on this data set to turn in.

4. Use an externally-provided macro to provide clarity about why some observations were excluded in going from step #2 to step #3. Besides producing what's called a *consort chart* to summarize the exclusion information, the macro also produces a table comparing the included observations with the excluded ones for some key variables.
5. Optionally, use the *medications_long* data set to create the same variables you created in step #1 (the result being a data set with one observation per person who took medications).

1. Use the medications_wide data set to create variables indicating diuretic use and use of lipid lowering medications

If you look at the *medications_wide* data set, you will see that it contains up to 17 drug code values per person in 17 character variables named DrugCode1-DrugCode17. In your first work for this assignment, make a new data set based on *medications_wide*. Create two new variables in this data set, as follows (fairly easy if done with an array but can also be done without an array):

Diuretic should be 1 if any of the person's 17 drug codes are exactly equal to or between '370000' and '380000'. Otherwise, Diuretic should be 0.

LipidLowerMed should be 1 if any of the person's 17 drug codes are exactly '390000', '391000', or '240600'. Otherwise, LipidLowerMed should be 0.

Helpful tips: (1) Initialize Diuretic and LipidLowerMed to 0. As you loop through the array, flip Diuretic to 1 if you find a diuretic drug code in any of DrugCode1-DrugCode17, and flip LipidLowerMed to 1 if you find a lipid lowering drug code in any of DrugCode1-DrugCode17. (2) Note that the drug code variables are CHARACTER – if you treat the drug code values like they are numeric, SAS will not be very happy.

Make sure this new data set has the same number of records as *medications_wide*. Only variables ID, Diuretic, and LipidLowerMed need to be in the new data set, once you are confident that those two variables are derived correctly. As part of your output, please provide one-way frequency tables for Diuretic and LipidLowerMed after step 1 (these are to help me in evaluating your work). Include the MISSING option on your TABLES statement.

2. Combine the core, nutrition, measurements, and new meds data sets

Combine *core*, *nutrition*, *measurements*, and the new meds data set from step 1 to form an initial data set of one record per person. In this data set, you should keep all variables that are in any of the incoming data sets. Only keep records in this data set for IDs found in the *core* data set. This data set should contain 15792 observations, and should be saved permanently to use in steps 4. As part of your output, please run PROC CONTENTS on this step 2 data set.

Note 1: A variable named Magnesium exists in both the *nutrition* and *measurements* data sets. The variables signify different things and we need both of them, so you will need to rename at least one of them before or as you combine the data. I suggest variable names DietMg and SerumMg respectively.

Note 2: If a participant was in *core* but not in the new meds data set, their values of Diuretic and LipidLowerMed will be missing after the data sets are combined. Please change those missing values to 0 (we assume the person was taking no medications if they do not have a record in the *medications_wide* data set). As part of your output, please provide one-way frequency tables for Diuretic and LipidLowerMed after step 2. Include the MISSING option on your TABLES statement.

3. Subset the data set made in step 2 to obtain the observations to be used for our manuscript

We only want to keep records for participants who meet the following criteria:

- Are black or white (that is, have a Race value of B or W)
- If female, consumed more than 500 and fewer than 3600 kilocalories per day
- If male, consumed more than 600 and fewer than 4200 kilocalories per day
- Have a non-missing value for BMI
- Have a non-missing value for serum magnesium
- Have a non-missing value for dietary magnesium

Once you make these exclusions, your data set should have 15232 observations. In terms of variables, this data set should contain all variables from *core*, *nutrition* (with Magnesium renamed), and *measurements* (with Magnesium renamed), along with variables Diuretic and LipidLowerMed. To turn in (for me to check your step 3 work):

- Run PROC CONTENTS on the step 3 data set
- Use PROC FREQ to produce one-way frequency tables of Race and Gender (include the MISSING option)
- Run PROC MEANS with N NMISS MEAN MIN MAX on variables TotCal, BMI, and the two magnesium variables DietMg and SerumMg

4. Run “exclusions macro” on data set made in step 2 – produce consort chart and compare included group with excluded group

In going from the step 2 data set to the step 3 data set, we lose 560 observations/participants (15792 – 15232). The purpose of this step 4 is to understand how many people each of our criteria removed from the analysis data set and to see whether the excluded group differs from the included group. If the two groups differ in important ways, it could cast doubt on the validity of our analysis.

Many manuscripts include a *consort chart* that shows how many observations were excluded from analysis for different reasons. Unfortunately these charts can't be created with a simple PROC. However, common SAS tasks that aren't easily performed by SAS are often macro-tized by generous SAS programmers and shared for others to use. Once such macro comes from a former 669 student and student programmer at the CSCC. It both produces a consort chart and compares the included and excluded groups for variables of your choice. Here's your chance to use this macro and practice the much-needed skill of working with other people's code. Details:

The provided macro Exclude_data_using_conditions was written by a former 669 student, Polina Kukhareva, who later worked as a student programmer at the CSCC. This macro and its usage are described in the provided paper *Quick Analysis and Visualization of Excluded and Missing Data*. I would like for you to run the macro to document how many participants you are losing from your analysis data set because of the six bulleted items listed in step 3 above. The paper mentions primary exclusions and secondary exclusions. For our purposes, please consider the first three exclusions primary and the last three (concerning missing values) as secondary exclusions. In terms of the variables to be compared between the included and excluded groups, let's look at race, gender, age, BMI, and prevalentCHD, where race, gender, and prevalentCHD are categorical variables.

I believe the only parameters you will need to pass to the macro are these:

```
 _DATA_IN  = < the data set you made in step 2 >,
 _USE_PRIMARY_EXCLUSIONS = Yes,
 _PRIMARY_EXCLUSIONS = race ^in ('B','W') ~ ^((Gender='M' and 600<TotCal<4200) |  (Gender='F' and
                               500<TotCal<3600)),
 _SECONDARY_EXCLUSIONS = missing(BMI) ~ missing(DietMg) ~ missing(SerumMg),
_PREDICTORS = race gender age BMI prevalentCHD,
_CATEGORICAL = race gender prevalentCHD,
_ID = ID,
_TITLE1 = < optional, but puts a descriptive title at the top of all figures >
```

Certainly you can look at the macro code and include other parameters in your call if you think they are useful.

For this class, we are using a slightly modified version of Polina's macro, and the SAS program containing the macro is named Exclude_data_using_conditions_for_669.sas. You SHOULD NOT modify this program (unless we find a problem and I tell you to make changes).

If you look at Exclude_data_using_conditions_for_669.sas, you will see that it defines a macro named Exclude_data_using_conditions. The comment before the %macro statement shows a sample call of the macro, but as stated above, I think that you can omit many of the parameters.

To prepare your ADSA program to run the macro, have your normal statements at the top – header block, %let statements, options, libname, setting your &outdir (which the macro should use automatically), etc**. However, you DO NOT need your normal ODS PDF or ODS RTF wrapper statements – the macro writes to an RTF file automatically.** This RTF file has the name exclusions_exclusion_1.rtf and will be found in your &outdir directory.

Then you will need what's called a %INCLUDE statement where you tell SAS where to find the macro that you are going to run. If you are using SAS OnDemand, you can use the macro in ~/my_shared_file_links/klh52250/macros. If you are running SAS in some other way, you can use the macro provided with the Canvas assignment materials.

Here is a sample %INCLUDE statement that someone could use if they stored Exclude_data_using_conditions_for_669.sas in their C:\BIOS 669 folder:

```
%include “C:\BIOS 669\ Exclude_data_using_conditions_for_669.sas”;
```

Here is a sample %INCLUDE statement for a SAS OnDemand user:

```
%include "~/my_shared_file_links/klh52250/macros/Exclude_data_using_conditions_for_669.sas";
```

After the %include statement, you should call the macro:

```
%exclude_data_using_conditions(_DATA_IN= <and all the other parameters above>);
```

A note about macro parameter values: Do not include either a comma (,) or an ampersand (&) in any of the values you pass as macro parameters, or else the macro will have trouble parsing the values. If you are tempted to use &, substitute the word *and*.

To check whether the macro ran appropriately, check the log and the produced rtf file, which by default is named exclusions_exclusion_1.rtf. If the macro worked, in the rtf file you will see a diagram like the paper's Figure 1 showing where you have lost participants; you have probably seen similar diagrams in scientific papers. You will also obtain a table such as the paper's Figure 2 that shows both absolute and hierarchical exclusion counts (seeing both types of counts allows you to see how many participants were excluded for multiple reasons). Finally, you will obtain a comparative table such as the paper's Figure 3 that shows how the participants excluded for secondary reasons compare with included ones on important characteristics. If big differences appear, that might be a matter of concern.

5. Optionally, derive Diuretic and LipidLowerMed using medications_long data set

If you have time (I realize this might seem like a joke), consider working on this additional part of ADSA. It is nice to be comfortable working with a SAS data set that is in long form rather than wide, as we have with the *medications_long* data set vs. *medications_wide*.

Start with data set *medications_long*. As with step 1, your goal is to output a data set with one observation per person (so you should end up with the same number of records as *medications_wide*), and each record should contain only ID, Diuretic, and LipidLowerMed.

As before, Diuretic should be 1 if any of a person's records contain a DrugCode equal to or between '370000' and '380000'. Otherwise, Diuretic should be 0.

As before, LipidLowerMed should be 1 if any of a person's records contain a DrugCode exactly equal to '390000', '391000', or '240600'. Otherwise, LipidLowerMed should be 0.

Once you are sure your Diuretic and LipidLowerMed derivations are correct, create a data set with one record per person and only variables ID, Diuretic, and LipidLowerMed. As part of your output, please provide one-way frequency tables for Diuretic and LipidLowerMed after step 5 (these are to help me in evaluating your work). Include the MISSING option on your TABLES statement.

The data set made here should be identical to the one you made in step 1. PROC COMPARE should be used to verify this.

To turn in:

- Two or more SAS programs and their logs and output (one program should only perform step 4). The specific output that I would like to see is shown in blue in this document. Please use good titles so that I can tell which step each piece of output relates to.
- The rtf file produced by your macro run in step 4 (default name is exclusions_exclusion_1.rtf)