BIOS 511 Lab 11
Creating Graphs and Using Macros in SAS

Please read the following instructions carefully before beginning this lab.

Instructions:

- **<u>The three required tasks should be completed in one SAS program</u>** named lab-11-PID.sas, where PID is your student PID number. Please make sure to include an appropriate header in the SAS program.

- Output for Tasks 2 and 3 should be delivered to separate PDF files named lab-11-PID-Task-X-output.PDF where PID is your student PID number and X is the task number.

- You will upload the SAS program, SAS log, and PDF output files to document completion of the lab.

- The submitted logs should reflect clean runs of the complete SAS program (i.e., they should not contain log messages from when the program was being developed).

Logs that contain ERRORs, WARNINGs, etc. will result in a point deduction of *at least 10 points*.

BIOS 511 Lab 11
Creating Graphs and Using Macros in SAS

Task 1: Write SAS code that reads in the VS and DM datasets from the ECHO trial and merges the ARMCD variable from the DM dataset onto the observations in the VS dataset to create a new dataset named WORK.VS that will be used for this lab.

Task 2: Review the following TST macro that uses one non-positional macro parameter named TESTCD.

- Write (do not copy) the TST macro into your lab program (or include the macro from a subfolder of the folder containing your lab program using a %INCLUDE statement).

- It is generally easier to first write the macro in the program that uses it (this makes it easier to test) and then move the macro to a separate program after it is complete and known to be working.

- In each place that there is a macro comment, add appropriate explanation regarding what the associated SAS code does within the macro.

```
%macro tst(testcd=);

data &testcd.;
 set vs end=last; %* Add a comment that explains what the functionality of the END
                     option for the SET statement;

 %* Add a comment that explains the point of using %UPCASE here;
 where vstestcd = "%upcase(&testcd.)";

   vstest = tranwrd(vstest,'Blood Pressure','BP');

        %* Add a comment that explains the point of using a conditional
           DO block to execute the CALL SYMPUT routine;
        if last = 1 then do;
               %* Add a comment that explains what macro variable is being created
                  using CALL SYMPUT and value its value will be based on
                  the WHERE statement used;
               call symput('lab',strip(vstest));

               %* Add a comment that explains what macro variable is being created
                  using CALL SYMPUT and value its value will be based on
                  the WHERE statement used;
               call symput('unit',strip(vsstresu));
        end;

   drop vstestcd vstest vsstresu vsseq
        vsblfl vsstat vsreasnd studyid;

   %* Add a comment that explains what is being done here;
   rename vsstresn = &testcd.;
run;


%* Add a comment that explains what the %PUT statement does in
   This instance and why its use my be helpful to debug a SAS
   macro.;
%put LAB=&lab. UNIT=&unit.;

data &testcd.;
 set &testcd.;
        %* Add a comment that explains what is being done here;
        label &testcd. = "&lab. (&unit.)";
run;
%mend;
```

BIOS 511 Lab 11
Creating Graphs and Using Macros in SAS

- To complete this task, add the following code to your lab program for Task 2 and replace the comments with an explanation of what the associated SAS code does.

<Code to define the TST macro / %INCLUDE statement for SAS program that contains the macro definition>

```
* Add a comment on what a call to the TST macro does;
%tst(testcd=diabp);
%tst(testcd=sysbp);
%tst(testcd=hr);

data vs_horiz;
 merge diabp sysbp hr;
 by usubjid visitnum visit;
run;

* Add a comment that explains why .. follows the reference
  to the TASK1FNAME macro variable;
ods pdf file="&outputPath./&task2FName..pdf" style=journal;
ods graphics / height=7.25in width=7in;
  title1 "Scatter Plot Matrix for Distolic BP, Systolic BP, and Heart Rate";
  title2 "Visit = Week 32";
  proc sgscatter data = vs_horiz;
    where visitnum = 5;
    matrix diabp sysbp hr / diagonal=(histogram) group=armcd;
  run;
ods graphics / reset=all;
ods pdf close;
```

Note: You will need to define the OUTPUTPATH & TASK2FNAME macro variables using a %LET statements at the top of your lab program.

BIOS 511 Lab 11
Creating Graphs and Using Macros in SAS

Task 3: Review the following SCATMAT macro that uses three non-positional macro parameters named TESTCDLIST and VISITNUM.

- Write (do not copy) the SCATMAT macro into your lab program (or include the macro from a subfolder of the folder containing your lab program using a %INCLUDE statement).
- In each place that there is a macro comment beginning with "Add a comment…", add appropriate explanation regarding what the associated SAS code does within the macro.

```
%macro scatMat(testcdList=,visitnum=,grp=);

%* Add a comment on the purpose of using %SYSFUNC - does %COUNTW exist?;
%* Note that arguments are NOT quoted as they would be when using the DATA step function COUNTW;
%let testnum = %sysfunc(countw(&testcdList.,|));

%* loop over the number of tests to include;
%do i = 1 %to &testNum;

 %* Add a comment on the purpose of using %SCAN;
 %* Note that arguments are NOT quoted as they would be when using the DATA step function SCAN;
 %let testcd = %scan(&testcdList.,&i,|);

  data &testcd.;
   set vs end=last;
   where vstestcd = "%upcase(&testcd.)";

   vstest = tranwrd(vstest,'Blood Pressure','BP');

   if last=1 then do;
          call symput('lab',strip(vstest));
          call symput('unit',strip(vsstresu));
   end;

   drop vstestcd vstest vsstresu vsseq
        vsblfl vsstat vsreasnd studyid;
   rename vsstresn = &testcd.;
  run;

  data &testcd.; set &testcd.;
        label &testcd. = "&lab. (&unit.)";
  run;

  %* Add a comment on the purpose of these conditional %DO blocks. Why must the code do something
     different for the first loop iteration?;
  %if &i = 1 %then %do;
   data vs_horiz;
    set &testcd.;
    by usubjid visitnum visit;
   run;
  %end;
  %else %do;
   data vs_horiz;
    merge vs_horiz &testcd.;
    by usubjid visitnum visit;
   run;
  %end;

%end;

ods graphics / height=7in width=7in;
proc sgscatter data = vs_horiz;
 where visitnum = &visitnum.;
 matrix %sysfunc(tranwrd(&testcdList.,|, )) /
        %if &grp^= %then %do; group=&grp. %end;
        diagonal=(histogram);
run;

%mend;
```

BIOS 511 Lab 11
Creating Graphs and Using Macros in SAS

- To complete this task, add the following code to your lab program for Task 3.

<Code to define the SCATMAT macro / %INCLUDE statement for SAS program that contains the macro definition>

```
ods pdf file="&outputPath./&task3FName..pdf" style=journal;
ods graphics / height=7.25in width=7in;
  title1 "Scatter Plot Matrix for Distolic BP, Systolic BP, and Weight";
  title2 "Visit = Week 0";
  %scatMat(testcdList=DIABP|SYSBP|WEIGHT,visitnum=1);
ods pdf close;
```

Note: You will need to define the TASK3FNAME macro variables using a %LET statement at the top of your lab program.

Extra Credit Task 4: Testing Your Skills (25 points)

1. **The extra credit program for task 4 should be completed separately from the program written to complete Tasks 1-3. The extra credit program should be named lab-11-PID-Task4.sas and the contents should reflect the changes described below. The output produced by the program should mirror task 3 and the output file should be named using the same convention as other tasks.**

2. **A more elegant macro would create the TITLE statements within the macro based on the user specified values for TESTCDLIST and VISITNUM. Modify the SCATMAT macro to do this (Name the new macro SCATMAT2).**

3. **For the SCATMAT macro, within the %DO loop that processes the VS data for each specified vital sign for the list provided by TESTCDLIST, the following block of code is included:**

```
%if &i = 1 %then %do;
 data vs_horiz;
  set &testcd.;
  by usubjid visitnum visit;
 run;
%end;
%else %do;
 data vs_horiz;
  merge vs_horiz &testcd.;
  by usubjid visitnum visit;
 run;
%end;
```

   Modify the SCATMAT2 macro so that this code is not included within the %DO loop. Instead, create the WORK.VS_HORIZ dataset within the macro but after the %DO loop.

4. **You will upload the extra credit SAS program, SAS log, and PDF output file to document completion of the extra credit. No partial extra credit will be given.**