```
/********Program Demonstrating Sequential Imputation Procedure *******/
/***************************** NOTE *********************************
#THIS PROGRAM CONTAINS ONLY THE KEY COMPONENTS FOR PROC MI MACRO. TWO PRIOR PROGRAMS THAT#
#PREPARE THE INPUT DATASET (SAMPLE) ARE OMITTED FOR SIMPLICITY SAKE.                     #
*****************************************************************************************/
options nocenter nodate pageno=1 ps=56 ls=96;
libname grasp 'P:\han\Han_methodology\Paper from POA\GRASP';


TITLE 'EXAMPLE FOR SEQUENTIAL MULTIPLE IMPUTATION';
TITLE2 '3 COVARIATES (NO MISSING OBS), 5 TARGET VARIABLES WITH SOME MISSING OBS';
TITLE3 'RETRIVE THE PREPARED SAMPLE DATASET';
DATA SAMPLE; SET grasp.SAMPLE;
PROC SORT; BY PTID;
RUN;
/*
PROC freq;tables ptid max_fintv/list missing;
RUN;
PROC MEANS DATA=SAMPLE N MISSING MEAN MIN MAX STD;
RUN;
*/
TITLE3 'STEP 1: Create macro variables';
data _NULL_;
   set SAMPLE(keep=PTID max_fintv);
   call symput('MacroVF1', trim(left(max_fintv)));
run;


/***PROC MI MACRO STARTS ***/
options /*mprint*/ symbolgen;
title1 "PEP - MI2";
footnote "LH: SeqMIPrgmLH.sas - &systime, &sysday, &sysdate9" ;

%macro MI_GRASP(n=5);

%local MADLcovar printindx  ii;

%let MADLcovar=;         /* hold COVAR from WAVE 1 to WAVE i-1 */
%let printindx=;         /* hold MI option noprint */

%do i=1 %to &MacroVF1;

data _NULL_;
    call symput('ii', trim(left(&i-1)));
run;

%if &i>=2 %then %do;
%let MADLcovar=&MADLcovar VAR1_&ii VAR2_&ii VAR3_&ii VAR4_&ii VAR5_&ii;
%let printindx=noprint;
%end;

title1 'Sequential Multiple Imputation';
proc mi data=SAMPLE out=miADL&i nimpute=&n seed=12345555 &printindx;
  MCMC INITIAL=EM(MAXITER=250 CONVERGE=1E-3);
  where (max_fintv>=(&i)) ;
  var COVAR1 COVAR2 COVAR3
     &MADLcovar VAR1_&i VAR2_&i VAR3_&i VAR4_&i VAR5_&i;
run;

/** get the mean of the imputed data set **/
data miADL1&i; set miADL&i(keep=_Imputation_ PTID
     VAR1_&i VAR2_&i VAR3_&i VAR4_&i VAR5_&i );
run;
proc sort data=miADL1&i; by PTID; run;
proc means data=miADL1&i n mean noprint;
 var VAR1_&i VAR2_&i VAR3_&i VAR4_&i VAR5_&i;
    by PTID;
    output out=meanADL&i;
run;
```

```sas
*keep only the means;
data meanADL1&i; set meanADL&i;
 if _STAT_ = 'MEAN';
drop _TYPE_ _FREQ_ _STAT_;
proc sort; by PTID;
run;

/** update the input dataset **/
proc sort data=SAMPLE; by PTID; run;
data sample; update SAMPLE meanADL1&i;
 by PTID;
run;

*Transpose the mean ds to vertical format and rename VAR;
data meanVAR&i; set meanADL1&i;
 by PTID;
 array VARmi_c[5] VAR1-VAR5;
 array VARmi&i[5] VAR1_&i VAR2_&i VAR3_&i VAR4_&i VAR5_&i;
 do j=1 to 5;
   VARmi_c[j]=VARmi&i[j];
   end;
   fu_wave=&i-1;
 drop j VAR1_&i VAR2_&i VAR3_&i VAR4_&i VAR5_&i;
 proc sort; by PTID FU_WAVE;
run;

/****/
*Save sd2 files for each F2F wave, raw and mean, respectively;
data grasp.miVAR&i;set miADL1&i;
run;
data grasp.meanVAR&i;set meanVAR&i;
run;

%end;

/*************/
*concatenate all mean ds into one: 6 in total upto 90 ms;
data meanVARMI;
     set meanVAR1 meanVAR2 meanVAR3 meanVAR4 meanVAR5 meanVAR6;
by PTID fu_wave;
proc sort; by PTID fu_wave;
run;

%mend MI_grasp;

/*VARIFICATION:
PROC PRINT  DATA=miADL1 N;WHERE PTID IN (535);
RUN;
PROC PRINT  DATA=meanADL11 N;WHERE PTID IN (535);
RUN;
PROC PRINT  DATA=meanVAR4 N;WHERE PTID IN (535);
RUN;
PROC PRINT  DATA=sample N;WHERE PTID IN (535);
RUN;
*/

/*****/
TITLE3 'STEP 2: Run MI MACRO';
%MI_GRASP()

/************/
TITLE 'CHECK ORIGINAL DATA';
TITLE2 'VAR1: MEAN AT EACH F/UP WAVE';
proc means DATA=SAMPLE n mean min median max missing maxdec=1 fw=5;
var VAR1_1-VAR1_6 ;
run;
TITLE2 'SAMPLE DATA STRUCTURE-Original DS (N=4): VAR1, VAR 2';
```

```
PROC PRINT  DATA=GRASP.SAMPLE N;WHERE PTID IN (306,321,535,723,1427);
VAR PTID COVAR1-COVAR3 MAX_FINTV VAR1_1-VAR1_6 VAR2_1-VAR2_6;
RUN;
/***********/
TITLE 'SAMPLE DATA STRUCTURE-Imputed mean DS (N=4)';
TITLE2 'VAR1, VAR 2 by F/UP WAVE';
PROC PRINT  DATA=meanVARMI N;WHERE PTID IN (306,321,535,723,1427); by ptid;
VAR PTID fu_wave var1 var2;
RUN;
```