Simplifying the Sample Design Process with PROC PMENU

Liza M. Thompson, GoodCents, Grayson, GA

ABSTRACT

GoodCents created the Sample Design Menu System to simplify and speed up the sample design process used for selecting a statistically sound population of customers to be metered for utility demand response programs. Prior to the menu system, users had to rewrite SAS® code to pertain to new customer populations and utilities, run a series of five SAS® programs, and complete several calculations using EXCEL. The menu system now allows users to produce a complete sample design by only running one program without having to change or rewrite any SAS® code and without standardizing the sample design outputs. Using PROC PMENU and a series of macros, the menu system allows users to input a customer population, performs a Dahlenius-Hodges analysis, weighs and allocates the population to each strata, randomly selects the samples for each strata, and exports the sample customer lists to EXCEL files. We will present the menu system from a user's design perspective, demonstrating the simple steps required by the user, as well as from a coder's perspective, sharing code and displaying output.

INTRODUCTION

At the start of every new research and analysis project, a sample design must be done. SAS® code has to be rewritten and altered to account for a new population file, a new defining variable, a new title, and new strata breakpoints. The Sample Design Menu System was created by GoodCents to prevent the constant alteration of SAS® code and to statistically conduct a sample design capable of randomly selecting a percentage of the population in question for analysis. The percentage of the population randomly chosen must statistically represent the population as a whole. Using a Microsoft Windows compatible point and click environment with pull down menus and input fields, the sample design process has become simple and hassle-free. The following paper will walk through the sample design process step-by-step, display the SAS® code used, and present actual sample design results.

STEP-BY-STEP SAMPLE DESIGN

The original sample design process consisted of running five different SAS® programs and conducting a series of calculations using EXCEL. PROC PMENU allows for each of the five programs and the calculations to be completed through a series of drop down menus and input screens.

The Main Menu

Using PROC PMENU the sample design process is conducted by running only one program – the driver, which is shown below.

```
/*****Sample Design Driver Program*****/

/*****D-HPMenu.sas is the initial menu for the system******/

filename menu 'J:\common\Consulting Services\DH Menu Program\SasCode\Sampdesign-
PMenu.sas';

%include menu;

run;

/*****D-HMainWin.sas is the file storing the initial screen verbage**/

filename dhsamp 'J:\common\Consulting Services\DH Menu Program\SasCode\Sampdesign-
MainWin.sas';

%include dhsamp;

run;
```

The code within the driver program points to two additional programs, the PMENU program and the MainWin program. The PMENU program uses a series of menu and item statements to create a main menu and drop down command system. The PMENU program points to five additional programs, one for each step of the sample design. As the user selects the menu options, PROC PMENU calls the appropriate program and uses a series of dialog and

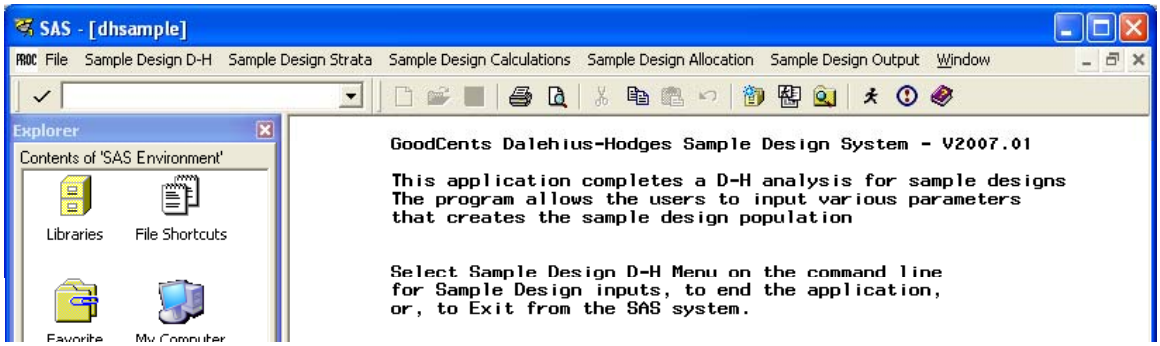
text statements to create input screens that enable the user to enter macro variables that are read into the program in queue. Portions of the PMENU program are shown below.

```
/*  PROGRAM:  Sampdesign-PMenu.sas                                    */

libname proclib 'J:\common\Consulting Services\DH Menu Program\SasCode';

filename sdh 'J:\common\Consulting Services\DH Menu Program\SasCode\Sampdesign-
DH.sas';

filename sst 'J:\common\Consulting Services\DH Menu Program\SasCode\Sampdesign-
Strata.sas';


proc pmenu catalog=proclib.sampledh;

options mprint mlogic symbolgen;

   menu sampledh;

          item 'File' menu=f;

          item 'Sample Design D-H' menu=w;

          item 'Sample Design Strata' menu=s;

          item 'Sample Design Calculations' menu=c;

          item 'Sample Design Allocation' menu=a;

          item 'Sample Design Output' menu=o;

   menu w;

   item 'Choose DH Inputs' dialog=d1;

          dialog d1 'end;pgm;include sdh; change flp @1 all;

          change zcs @2 all; change ldv @3 all; change dhv @4 all;

          change icm @5 all; change dtl @6 all; submit';

          text #1 @1 'Enter the Name of the SAS Dataset You Wish to Create ';

          text #2 @5 len=10;
```

The mainwin program is called by the driver and simply creates the wording displayed on the main menu page. A portion of the mainwin program code is shown below.

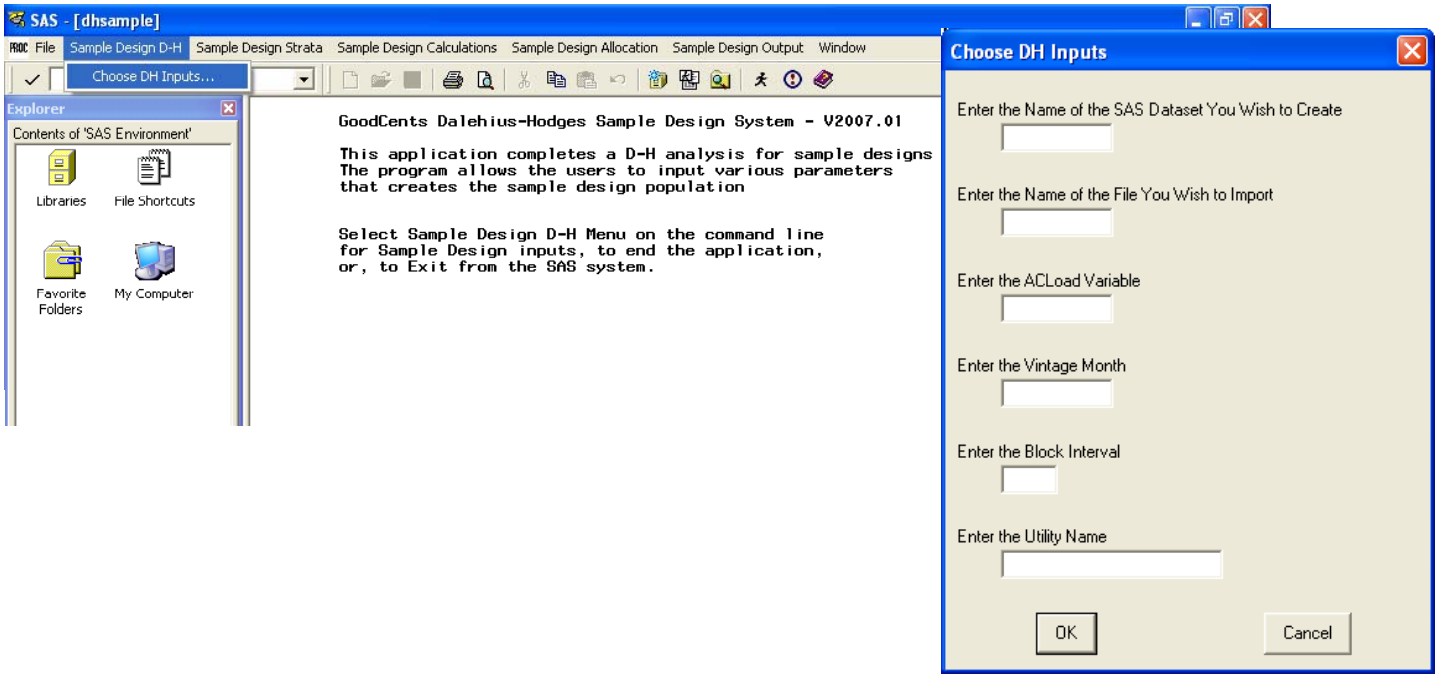
```
/*  PROGRAM:  Sampdesign-MainWin.sas                                  */

   window dhsample menu=proclib.sampledh.sampledh

   #2 @10 'GoodCents Dalehius-Hodges Sample Design System - V2007.01'

   display dhsample;   run;
```

Once the driver program has been run by the user, the PMENU and MainWin programs run automatically and create a system menu with five menu options: Sample Design D-H, Sample Design Strata, Sample Design Calculations, Sample Design Allocation, and Sample Design Output. The main menu is shown below.



Dahlenius-Hodges Analysis

The first step to a stratified sample design is to determine strata breakpoints. We conduct a Dahlenius-Hodges Analysis to determine our strata breakpoints. When the user selects the first drop down menu, Sample Design D-H, PROC PMENU creates an input window that allows the user to input variables that will then be placed in the D-H program through a series of macros. The menu selection, input window, and portions of the D-H program code are shown below.




```
/*   PROGRAM:   Sampdesign-DH.sas                                          */

%macro sampleDH(infile,excelin,acvar,vintage,intervals,utility);

/*Set the data file, delete customers with missing kWh, and set up blocking
factors*/

PROC IMPORT OUT= sample.&infile

            DATAFILE= "J:\common\Consulting Services\DH Menu
Program\CSVInputFiles\&excelin..csv"

            DBMS=CSV REPLACE;
```

```
     GETNAMES=YES;
     DATAROW=2;
RUN;
data popdata;
set sample.&infile;
FREQKWH = (INT((MAXKWH/&intervals)+.999))*&intervals ;
/*Change Label and Titles for particular utility*/
LABEL FREQKWH="&utility Customer Sign Ups for &vintage";
TITLE "&utility Customer Sign Ups for &vintage";
/*Runs frequency analysis on kWh to find number of customers in each block*/
PROC FREQ ;
 TABLE FREQKWH/MISSING OUT=ENG1;
/*Runs D-H to determine strata breakpoints*/
DATA CF1; SET ENG1;
SQCNT=SQRT(COUNT);
DI=FREQKWH-LAG1(FREQKWH);
IF _N_=2 THEN D1=FREQKWH;
IF _N_=1 THEN DI=FREQKWH;
UI=DI/D1;
SQUI=SQRT(UI);
DHI=SQUI*SQCNT;
RETAIN CCOUNT CPERCENT CSQCNT D1 CDHI;
CDHI+DHI;
CCOUNT+COUNT;
CPERCENT+PERCENT;
CSQCNT+SQCNT;
quit;
%mend;
%sampledh (flp,zcs,ldv,dhv,icm,dtl);
filename driver "J:\common\Consulting Services\DH Menu Program\SasCode\Sampdesign-
Driver.sas";
%include driver;
run;
```

The frequency analysis and D-H analysis output are printed to the output window in SAS®. The user must do one calculation by hand using the D-H output.

Utility Customer Sign Ups for February 09:54 Monday, February 5, 2007 2

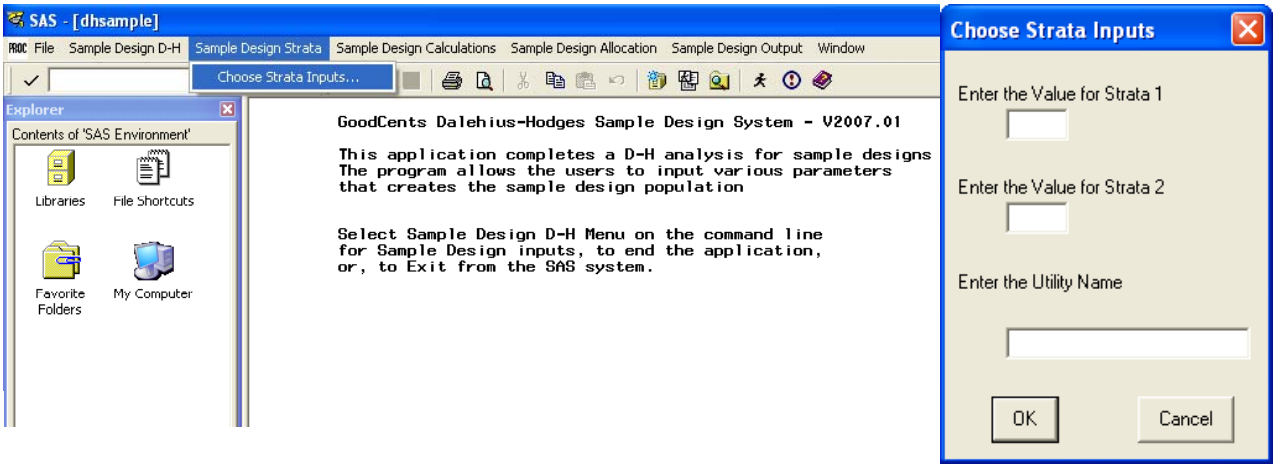
Obs	FREQKWH	COUNT	PERCENT	SQCNT	DI	D1	UI	SQUI	DHI	CCOUNT	CPERCENT	CSQCNT	CDHI
1	10	11	1.9231	3.3166	10	.	.	.	.	11	1.923	3.3166	.
2	15	41	7.1678	6.4031	5	15	0.33333	0.57735	3.69685	52	9.091	9.7197	3.6968
3	20	62	10.8392	7.874	5	15	0.33333	0.57735	4.54606	114	19.93	17.5938	8.2429
4	25	66	11.5385	8.124	5	15	0.33333	0.57735	4.69042	180	31.469	25.7178	12.9333
5	30	103	18.007	10.1489	5	15	0.33333	0.57735	5.85947	283	49.476	35.8667	18.7928
6	35	86	15.035	9.2736	5	15	0.33333	0.57735	5.35413	369	64.51	45.1403	24.1469
7	40	57	9.965	7.5498	5	15	0.33333	0.57735	4.3589	426	74.476	52.6901	28.5058
8	45	56	9.7902	7.4833	5	15	0.33333	0.57735	4.32049	482	84.266	60.1735	32.8263
9	50	32	5.5944	5.6569	5	15	0.33333	0.57735	3.26599	514	89.86	65.8303	36.0923
10	55	25	4.3706	5	5	15	0.33333	0.57735	2.88675	539	94.231	70.8303	38.979
11	60	10	1.7483	3.1623	5	15	0.33333	0.57735	1.82574	549	95.979	73.9926	40.8048
12	65	7	1.2238	2.6458	5	15	0.33333	0.57735	1.52753	556	97.203	76.6383	42.3323
13	70	7	1.2238	2.6458	5	15	0.33333	0.57735	1.52753	563	98.427	79.2841	43.8598
14	75	6	1.049	2.4495	5	15	0.33333	0.57735	1.41421	569	99.476	81.7336	45.274
15	80	3	0.5245	1.7321	5	15	0.33333	0.57735	1	572	100	83.4656	46.274

Our PMENU program is designed to conduct a three strata sample design. In order to determine the strata breakpoints, the user must take the final CDHI value (above 46.274) and divide it by three (which gives us 15.42). The freqkwh with the closest CDHI value is the first strata breakpoint (freqkWh 25, highlighted above in red). The user then must find the second strata breakpoint, which is the closest freqkWh to the final CDHI value divided by three and multiplied by two (CDHI 46.274/2 = 15.42*2 = 30.85 – which is closest to the CDHI value for freqkWh 45). The user now knows the strata breakpoints are 25 kWh and 45 kWh.

At the end of each program, it is necessary to direct SAS® back to the driver program, which in turn will direct SAS® back to the PMENU program, which holds the directories for the four remaining sample design programs.

Stratification

The next program essential to the sample design process is the strata program, which conducts a means analysis for each of the three strata groups. The means output is necessary for the third step, the calculations. When the user selects the second menu option, Sample Design Strata, an input window pops up asking the user to input important information that will then become macro variables in the strata program. The menu selection, input window, and portions of the Strata program code follow below.



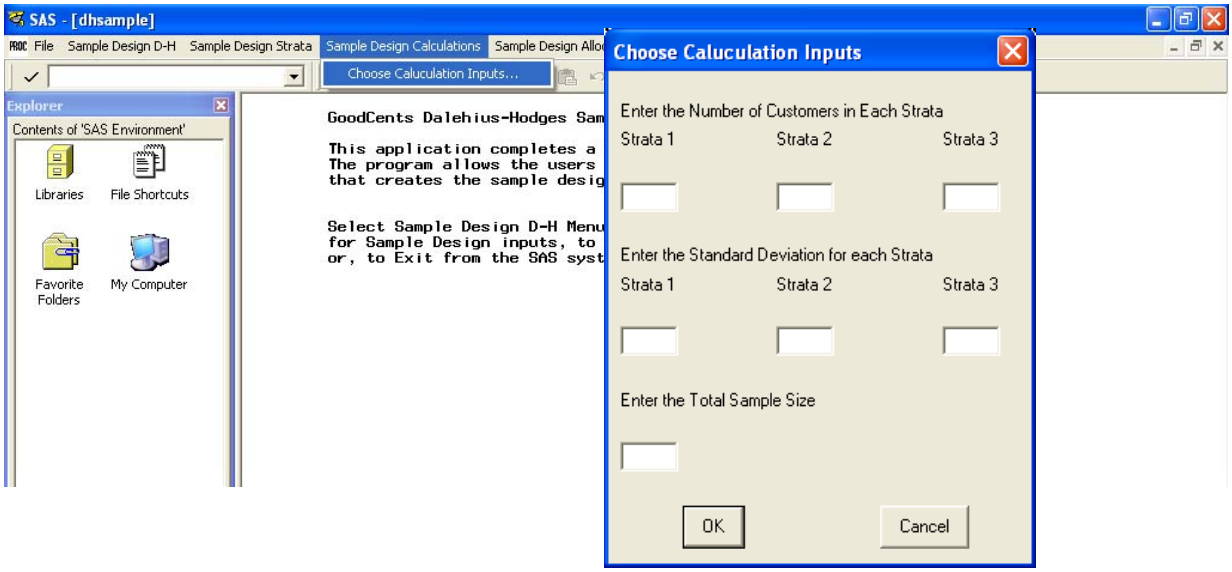

```
/*  PROGRAM:  Sampdesign-Strata.sas                                    */

%macro sampleDH(strata1,strata2,utility);

/*Set the data file, set the load variable, and set the utility name for title*/

data kwdata;

set popdata;
```

```
str1=&strata1;
str2=&strata2;
run;
proc sort; by ackw;
run;
title "&utility Stats AC All Sites";
proc means n sum mean max min std var cv;
var ackw;
run;
data kwstrat1;
set kwdata;
if  ackw<=str1;
run;
title "&utility Stats AC Strata 1";
proc means n sum mean max min std var cv;
var ackw;
run;
data kwstrat2;
set kwdata;
if ackw>=str1 and ackw<=str2;
run; quit;
%mend;
%sampledh (vso,vst,vuy);
filename driver "J:\common\Consulting Services\DH Menu Program\SasCode\Sampdesign-
Driver.sas";
%include driver;
run;
```

The means output is printed on the output page of SAS®. The user will have to input the number of points in each strata and the standard deviation of each strata (both determined in the means procedure) into the next step's input window.

Calculations

The next menu selection and corresponding program is the calculations program. This program uses the number of points in each strata and the standard deviation of each strata to weight and select the select the number of points that should be allocated to each strata. The menu selection, input window, and portions of the calculations program code are shown below.

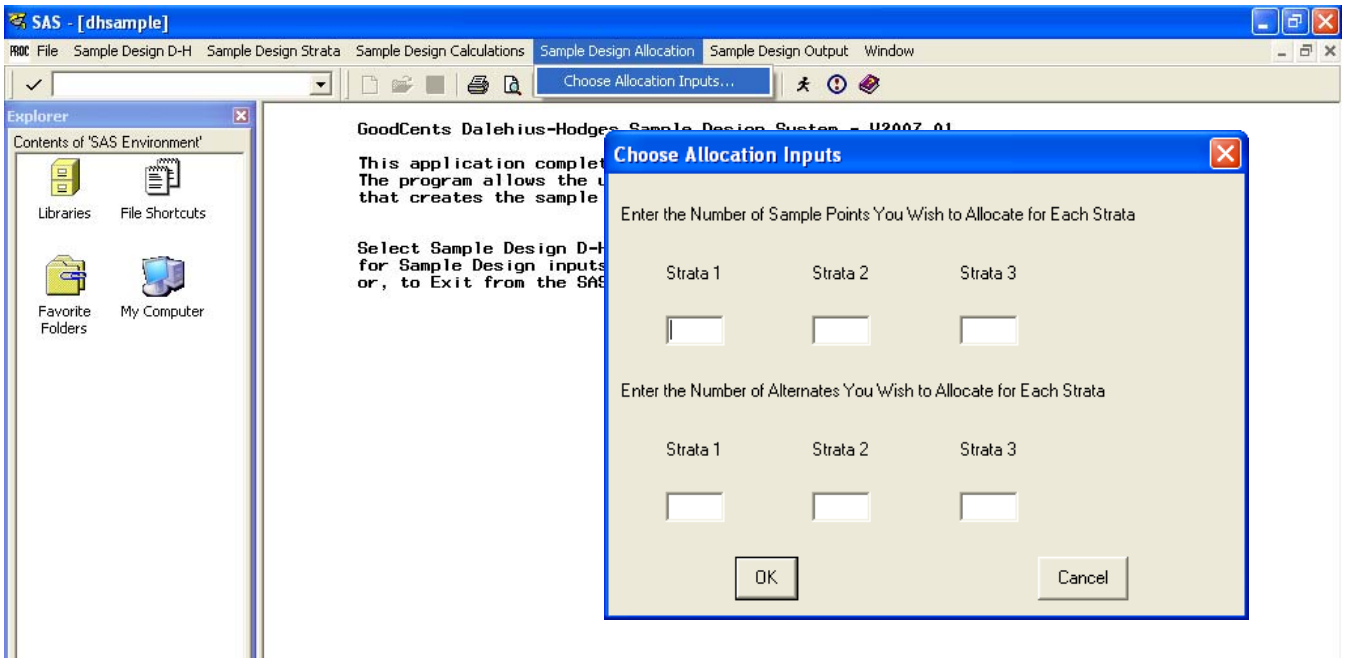



```
/*  PROGRAM:  Sampdesign-Calcs.sas                                    */
%macro sampleDH(nstr1,nstr2,nstr3,stdv1,stdv2,stdv3,sasi);
/*Calculate the number of sample points each strata should have*/
data calcdata;
set kwdata;
nstr1=&nstr1;
nstr2=&nstr2;
nstr3=&nstr3;
totaln=nstr1+nstr2+nstr3;
sd1=&stdv1;
sd2=&stdv2;
sd3=&stdv3;
wt1=sd1*(nstr1/totaln);
wt2=sd2*(nstr2/totaln);
wt3=sd3*(nstr2/totaln);
totwt=wt1+wt2+wt3;
totsamp=&sasi;
s1alloc=(totsamp*wt1)/totwt;
s2alloc=(totsamp*wt2)/totwt;
s3alloc=(totsamp*wt3)/totwt;
run; quit;
%mend;
```

```
%sampledh (xs1,xs2,xs3,xv1,xv2,xv3,xss);

filename driver "J:\common\Consulting Services\DH Menu Program\SasCode\Sampdesign-
Driver.sas";

%include driver; run;
```

Allocation

The fourth menu selection item and sample design program is the sample allocation. This program randomly selects the appropriate number of points for each strata and assigns them to the sample. When the user selects the allocation menu item, an input window will pop up asking the user to enter in the number of points allocated to each strata and the number of alternates the user wishes to allocate to each strata. The number of points allocated to each strata is determined by the output of the calculations program, which was previously run. The menu selection, input window, and portions of the allocation program code are shown below.




```
/*  PROGRAM:  Sampdesign-Allocation.sas                                    */
***  START OF SAMPLE SELECTION AND RANDOM NUMBER GENERATOR MACROS  ***;
***  (NO USER MODIFICATIONS REQUIRED IN THE MACROS)                ***;
  OPTIONS MCOMPILE MACROGEN MLOGIC SYMBOLGEN ;
   **    RANDOM NUMBER GENERATOR            **;
  %MACRO RANDOM2(NPOP,NSAMP,NALT,STRATA) ;
 DATA SELECTED;
NSAMP = &NSAMP  ;
 NPOP = &NPOP   ;
 NBR_ALT = &NALT ;
 STRATA = &STRATA ;
```

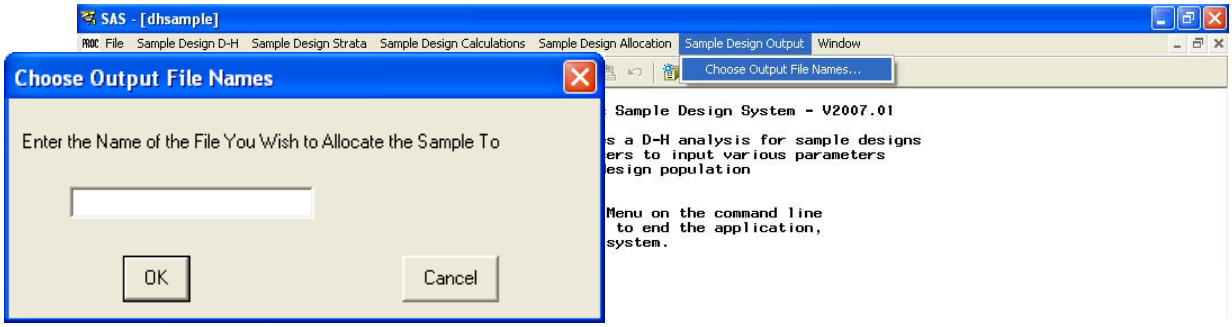
```
 TSAMP = NSAMP * (NBR_ALT + 1);
 ITSAMP = ROUND(TSAMP);
 IF ITSAMP > NPOP THEN ITSAMP = NPOP ;
TSELECT = ITSAMP * 10.00;
 ITSELECT = ROUND(TSELECT);
 TALT = NSAMP * NBR_ALT;
 ITALT = ROUND(TALT);
   DO ORDER = 1 TO ITSELECT;
      Z =  NPOP * UNIFORM(0);
      SELT_OBS = ROUND(Z);
      IF SELT_OBS < 1 THEN SELT_OBS = 1;
      OUTPUT;
   END;
PROC SORT DATA=SELECTED;
  BY SELT_OBS;
 DATA UNIQUE;
  SET SELECTED;
  BY SELT_OBS;
  IF FIRST.SELT_OBS THEN DO ;
    OUTPUT;
  END ;
PROC SORT DATA=UNIQUE ; BY ORDER ;
DATA UNIQUE ;
  SET UNIQUE ;
  IF _N_ <= ITSAMP THEN DO ;
    ORDER = _N_ ;
    OUTPUT ;
  END ;
PROC SORT DATA=UNIQUE ;  BY SELT_OBS;
DATA SAMPLE(KEEP= ORDER SELT_OBS P_ALT NPOP NSAMP NBR_ALT STRATA SEQ);
  SET UNIQUE ;
  IF _N_ EQ 1 THEN ICOUNT = 1;
  IF ICOUNT EQ 1 THEN P_ALT = 0;
   ELSE P_ALT + 1;
  IF P_ALT EQ NBR_ALT THEN ICOUNT = 1;
```

```
    ELSE ICOUNT + 1;
   IF _N_ EQ 1 THEN SEQ_NBR = 0;
   IF P_ALT EQ 0 THEN SEQ_NBR + 1;
   SEQ = PUT(STRATA,2.) || PUT(SEQ_NBR,Z4.) || PUT(P_ALT,1.) ;
 %MEND RANDOM2 ;
%MACRO SAMPLE(DS,SAMP,ALT,STRAT,infile,ackw,stro,strt) ;
    DATA &DS ;
     SET &DS  END=EOF  ;
     SELT_OBS = _N_ ;
     IF EOF THEN DO ;
       CALL SYMPUT('POP',SELT_OBS) ;
       IF &SAMP LE 1 THEN DO ;
         SAMP = INT( SELT_OBS * &SAMP ) ;
         CALL SYMPUT('SAMP',SAMP) ;
       END ;
     END ;
    PROC SORT ;    BY SELT_OBS ;
    %RANDOM2(&POP,&SAMP,&ALT,&STRAT) ;
    PROC PRINT DATA=SAMPLE ;
     TITLE RANDOM NUMBERS FOR STRATA &DS ;
    DATA &DS  PRIMES ;
     MERGE &DS (IN=IN1) SAMPLE (IN=IN2) ;
      BY SELT_OBS ;
     IF IN2 ;
     OUTPUT &DS ;
     IF P_ALT EQ 0 THEN OUTPUT PRIMES ;
     data sample.&ds; set &ds;
    PROC PRINT DATA=sample.&DS ;
     TITLE SAMPLE FOR STRATA &DS ;
    RUN ;
    TITLE ' ' ;
 %MEND SAMPLE ;
*  END OF SAMPLE SELECTION AND RANDOM NUMBER GENERATOR MACROS  ***;
**   BEGINNING OF USER CONTROLLED AREA  **;
data stratasamp1;
```

```
set calcdata;
kwh7=ackw;
if kwh7<=strata1;
run;
data stratasamp2;
set calcdata;
kwh7=ackw;
if kwh7>strata1 and kwh7<=strata2;
run;
data stratasamp3;
set calcdata;
kwh7=ackw;
if kwh7>strata2;
run;
*  1ST PARAMETER - SAS DATASET NAME TO PULL SAMPLE FROM ('BDR')     *;
*  2ND PARAMETER - NUMBER OF SAMPLE POINTS/RANDOM NUMBERS TO PICK   *;
*  3RD PARAMETER - NUMBER OF ALTERNATES TO BE CHOSEN, 0 = NONE       *;
*  4TH PARAMETER - STRATA CODE USED TO IDENTIFY SEPARATE SAMPLES     *;
%SAMPLE(stratasamp1,jso,qxq,91) ;
%SAMPLE(stratasamp2,jst,qvq,92) ;
%SAMPLE(stratasamp3,jsh,qzq,93) ;
filename driver "J:\common\Consulting Services\DH Menu Program\SasCode\Sampdesign-
Driver.sas";
%include driver; run;
```

Output

The output menu item is the last step to the sample design process. When the user selects this item, an input window will pop up asking the user to simply name the output file. The output program simply outputs the sample points allocated in the previous program to an EXCEL CSV file. The menu selection, input window, and portions of the code are shown below.



Final Sample Design Output

GoodCents_ID	kwh	dailykWh	FREQKWH	SELT_OBS	SAMP	NSAMP	NPOP	NBR_ALT	STRATA	ORDER	P_ALT	SEQ
32	2368	19.57024793	20	13		2	180	3	91	7	0	9100010
42	893	14.40322581	15	17		2	180	3	91	2	1	9100011
71	2884	22.35658915	25	30		2	180	3	91	1	2	9100012
415	2846	22.40944882	25	130		2	180	3	91	6	3	9100013
429	3023	24.184	25	136		2	180	3	91	8	0	9100020
433	2744	21.952	25	138		2	180	3	91	3	1	9100021
440	2407	19.56910569	20	140		2	180	3	91	4	2	9100022
497	1061	16.84126984	20	156		2	180	3	91	5	3	9100023
14	4595	36.76	40	8		5	302	3	92	4	0	9200010
48	3361	27.32520325	30	22		5	302	3	92	17	1	9200011
121	4074	32.592	35	59		5	302	3	92	16	2	9200012
154	4331	34.37301587	35	77		5	302	3	92	2	3	9200013
178	3479	27.61111111	30	90		5	302	3	92	15	0	9200020
184	4834	39.30081301	40	94		5	302	3	92	8	1	9200021
189	4513	36.69105691	40	97		5	302	3	92	14	2	9200022
204	4288	34.304	35	105		5	302	3	92	3	3	9200023
15	6606	52.848	55	1		18	90	3	93	64	0	9300010

CONCLUSIONS

A statistical sample design is a complicated stepwise procedure that used to involve the alteration of 5 different SAS® programs. PROC PMENU and a series of macro variables enabled the 5 programs to be combined with no need for additional alterations. PROC PMENU created a user menu system with drop down items and input windows that enabled the user to complete a new sample design from start to finish without changing any code. The user simply had to run one program, the driver, and go through the drop down items. If the user had to redo a portion of the sample design, there is no need to run the first programs, simply just the last.

REFERENCES

SAS® PROC PMENU Reference Guide

ACKNOWLEDGMENTS

Thank you to Bill Bland for allowing me to use his Sample Design SAS® code and the encouragement to create this program.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Liza M Thompson
GoodCents
1075 Cooper Road, Suite 100
Grayson, GA 30017
Work Phone: 678-836-1059
Fax: 678-325-6496
E-mail: Liza.Thompson@GoodCents.com
Web: www.goodcents.com