

PROC SURVEYSELECT: A Simply Serpentine Solution for Complex Sample Designs

Louise Hadden, Abt Associates Inc., Cambridge, MA

ABSTRACT

SAS® programmers are frequently called upon to draw a statistically defensible sample for surveys. Many of us have become adept at various data step sampling techniques over the years. However, SAS®'s relatively new suite of SURVEY procedures has made our lives much easier. Stratification? No problem. Systematic random sampling? No problem. Proportional sampling? No problem. All in the same design? Yes! This paper will demonstrate the use of PROC SURVEYSELECT to facilitate the drawing of a valid, stratified, random, proportional sample.

The examples presented were run on a mainframe computer (OS/390) running SAS® V8.2, and on a PC (WIN2K PRO) using SAS® 9.1.2.

INTRODUCTION

I began using PROC SURVEYSELECT when an analyst requested that I create a sample that seemed impossible to achieve no matter how many gymnastic routines were performed in the data step. The data resided on a mainframe computer running SAS® V8.2 on OS/390 and there were approximately 5 million records on the file. 5 million records is a mere pittance in these days of data warehousing, but the processing time in dealing with the file was non-trivial. The particular task was to draw a sample of Medicare Drug Card Beneficiaries. Twenty-seven different drug cards were pre-selected, and then stratified by a subsidy indicator (general cardholders vs. those receiving transitional assistance). Each of the 54 strata (card by subsidy) were to have 600 potential respondents randomly selected from the universe. While 54 strata is not particularly easy to sample within a data step, it is also not impossible, and that was my initial approach. But then, the analyst said, "Oh, and can you make the sample within each stratum proportional to the actual proportion of aged and disabled within the stratum?" This increased the number of strata to 108, and required the calculation of a share for each stratum in a separate data set that had to be merged onto the master file. It was still possible to do within a data step, but not pretty with a file of 5 million records that was not sorted or indexed by the stratum id. The straw that broke the camel's back came when the analyst said, and can you make SURE that we have complete representation in the sample of all possible values of just a few more variables. This would have increased the number of strata exponentially. Try as I might, I couldn't think of a way to sort the data set efficiently to achieve this particular end. I researched serpentine sorts, and found that PROC SURVEYSELECT utilizes serpentine sorts as part of the selection process if the programmer specifies control variable(s). This made PROC SURVEYSELECT the ideal choice for my sampling problem.



SLITHERING YOUR WAY THROUGH SAMPLE SELECTION

- PROC SURVEYSELECT
DATA=
METHOD=
SEED=
SAMPSIZE=
OUT=
OUTSORT=

The PROC SURVEYSELECT statement itself accomplishes many of the goals I set out to achieve. Aside from the customary "DATA=" option, there are a number of other important items. The first of these is METHOD. PROC SURVEYSELECT allows you to perform simple random sampling, unrestricted random sampling (with replacement), systematic random sampling, sequential random sampling, and a number of methods for PPS (prob-

ability proportional to size) sampling. Just reading about some of the PPS methods made my head ache, so I opted for the systematic random sampling method, or METHOD=SYS.

```
150      PROC SURVEYSELECT DATA=TEMP METHOD=SYS
```

Some of the procedure statement options are dependent on the sampling method. I will only discuss the options I used for my systematic random sample, but invite you to delve further into the mysteries of PROC SURVEYSELECT by reading Chapter 72 of the online SAS 9.1 documentation! PROC SURVEYSELECT makes drawing a stratified sample a piece of cake, for example. Since I needed to kludge a couple of different sampling methods together, I couldn't take advantage of that particular sampling method.

Since I wanted to be able to exactly replicate my sample if I had to rerun it, I used the SEED= option. This allows you to specify the initial seed for random number generation. Rerunning the same program on the same (unsorted) data will replicate your sample if you use a seed.

```
151      SEED=12345678
```

The SAMPSIZE= option allowed me to "feed" the desired N to select from each stratum. PROC SURVEYSELECT expects that you will have at least the number requested for each stratum within your sampling frame, AND it expects that your strata identifier is sequential when feeding your desired Ns in this way (i.e., you must sort your frame by the strata identifier prior to sampling.) You can also specify N= to specify a particular N for each stratum, or SAMPRATE= to specify a sampling rate, or feed a file with the stratum variable (sorted sequentially of course!) and desired Ns, to name a few other options. For the purposes of illustration, I am showing the SAMP-SIZE=() option in all its glory. In a later paragraph I will show how I derived these numbers – which I would have to have done for data step sampling as well.

```
152      SAMPSIZE=
153      (564 36 574 26 518 82 545 55 532 68 560 40 571 29 574
154      26 344 256 373 227 335 265 351 249 474 126 359 241 549
155      51 575 25 355 245 370 230 440 160 400 200 531 69 583 17
156      510 90 556 44 469 131 424 176 589 11 598 2 396 204 380 220
157      412 188 392 208 460 140 472 128 489 111 507 93 550 50 493
158      107 448 152 379 221 574 26 589 11 455 145 502 98 414 186
159      485 115 510 90 375 225 489 111 506 94 525 75 487 113 543
160      57 532 68)
```

PROC SURVEYSELECT allows you to specify your OUT and OUTSORT data sets. The OUT data set will contain your sample information, stratum variable, ID variable(s), and CONTROL variable(s). If you specify CONTROL variables, the data set will be sorted by your CONTROL variables with whatever sort method is used (SERPENTINE, NESTED, etc.) This may not be desirable for very large sample files you may want to remerge onto your original file, so if you want to maintain the sort of the input data set in the output data set, specify OUT-SORT= to hold the (control) sorted data set.

```
161      OUT=&OUT1;
```

- STRATA

The STRATA statement allows you to specify your stratifying variable. In my case, I constructed a variable which was a combination of a drug card identifier (27), an aged/disabled dichotomous variable, and a transitional assistance/general dichotomous variable. I had 108 strata in all. It's much easier to use existing variables as strata variables, but I needed to mix proportional (aged/disabled) and non-proportional (600 each from transitional assistance and general.) In my case, the input file had to be sorted by strata in ascending order due to the particular sampling method I was using.

```
162      STRATA STRATUM;
```

- CONTROL

The CONTROL statement is where you specify additional variables (other than strata) to sort by when performing sampling. The default sort is hierarchical serpentine sorting. This was key for me, as my project director wanted

to ensure adequate representation of different age cohorts, genders and races in the sample. You can also specify SORT=NEST on the PROC SURVEYSELECT statement if you do not wish to use the default serpentine sort.

```
163          CONTROL AGE_COHORT GENDER RACE;
```

- ID

The ID statement allows you to specify variables from the input file or sampling frame to carry into the output file. The default is that ALL variables in the input file are carried to the output file. At the very least, an identifier that allows you to merge back to the sampling frame is a good idea. Any strata or control variables are included automatically, as well as sample proportion numbers, etc. from the procedure.

```
164          ID ABTID STRATUM AGED RACE GENDER TRANS05 AGE_COHORT
165             ETHNCTY SUB5DR05;
```

```
NOTE: THE DATA SET OUTSAMP.SURVEY27 HAS 32400 OBSERVATIONS AND 11 VARIABLES.
NOTE: THE PROCEDURE SURVEYSELECT PRINTED PAGE 1.
NOTE: THE PROCEDURE SURVEYSELECT USED 30.10 CPU SECONDS AND 5418K.
```

As you can see below, PROC SURVEYSELECT provides you with the relevant information on your sampling routine in a convenient one page format. In addition, it is a good idea to print a few records of your output file and take a look at the created variables such as SAMPLINGWEIGHT and SELECTIONPROB.

```
DRUGCARD:  OUTPUT NATIONAL SAMPLE ROUND 2              14:27 MONDAY, FEBRUARY 28, 2005   1
                                                    BENEFICIARY EXTRACT FILE

                                                    THE SURVEYSELECT PROCEDURE

SELECTION METHOD      SYSTEMATIC RANDOM SAMPLING
STRATA VARIABLE      STRATUM
CONTROL VARIABLES    AGE_COHORT
                    GENDER
                    RACE
CONTROL SORTING      SERPENTINE

INPUT DATA SET      TEMP
RANDOM NUMBER SEED    12345678
NUMBER OF STRATA     108
TOTAL SAMPLE SIZE    32400
OUTPUT DATA SET     SURVEY27

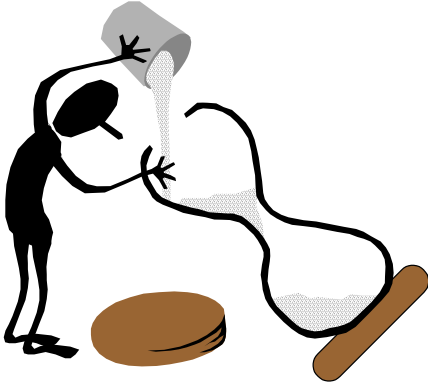
TE00.#EMPDDC.LIB.DCARDLIB(EEVS63)  -- 28FEB05
```

```
DRUGCARD:  OUTPUT NATIONAL SAMPLE ROUND 2              14:27 MONDAY, FEBRUARY 28, 2005   4
                                                    BENEFICIARY EXTRACT FILE

OBS   STRATUM      ABTID      SUB5DR05      Selection Prob      Sampling Weight
1      1           D0543110010055686      D0543      0.051217      19.5248
2      1           D0543110010251949      D0543      0.051217      19.5248
3      1           D0543110010349093      D0543      0.051217      19.5248
4      1           D0543110010470168      D0543      0.051217      19.5248
5      1           D0543110011163595      D0543      0.051217      19.5248
6      1           D0543110011409799      D0543      0.051217      19.5248
7      1           D0543110011524684      D0543      0.051217      19.5248
8      1           D0543110011668145      D0543      0.051217      19.5248
9      1           D0543110012318700      D0543      0.051217      19.5248
10     1           D0543110012494645      D0543      0.051217      19.5248
```

NS TO GET

In this case creating an input file (from which to create my list used in the SAMPSIZE= statement above) was fairly complex. For a simple proportional sample using data step sampling, you can simply use proc freq on your stratum variable(s), output the percents, divide the percents by 100, and apply to the total desired number after sorting by your stratum variable(s) and a random number. (OR, it's even easier using one of PROC SURVEYSELECT's proportional sampling methods!) My project officers wanted to select 600 cases from each drug card and transitional assistance / general combination (27 card ids by the dichotomous variable for TA / general = 54 "strata"). Then they wanted the 600 cases within each stratum to proportionally represent the numbers of aged



versus disabled enrollees. I wrote a macro (iterated 54 times) which performed a frequency on the aged / disabled dichotomous variable for each stratum, outputting the percents, dividing by 100, and multiplying by 600 to get the Ns to sample for each stratum (now 108). Then I set the 108 "lines" together sequentially and created a stratum variable using "_n_". Naturally it is important that this stratum variable match what it is in your sampling frame! You can use the file created this way as an input file to PROC SURVEYSELECT, or create a macro list from it. NOTE: I was lucky enough that the sampling frame was large enough that I did not have difficulties achieving exactly 600 per stratum. This won't always be the case either in PROC SURVEYSELECT or with data set sampling. It is important to carefully review your output samples!

MORE REAL LIFE EXAMPLES

Although I began using PROC SURVEYSELECT to process a very large file on the mainframe, I found it so easy to use and versatile that I began to use it for other applications. Three additional samples are presented below. The first is to do sample replacement for the original use (very large file on the mainframe). NOTE: had I known a little more about PROC SURVEYSELECT, I could have set up sample replacement within the original program! The second and third examples are for much smaller applications on the PC, for the same use (the analysts changed their minds multiple times.) The purpose of these samples is to demonstrate the great utility, versatility and ease of use of this procedure. You will notice a distinct difference in the amount of information SAS gives you in the logs between the two versions used here (8.2 on the mainframe for Sample 1, and 9.1.2 on the PC for Samples 2 and 3.) I'm looking forward to see what happens when I start using PROC SURVEYSELECT with 9.1.3 which I recently received!

SAMPLE 1

```
177      PROC SURVEYSELECT DATA=TEMP4 METHOD=SYS
178          SEED=12345678
179          SAMPSIZE=(2 1
180                  1 1 7 3 5 2 1 1 1 1 2 3 3 1 2 1 1 1 1 3 3 3 4
181                  3 2 2 1 2 4 3 1 4 1 4 4 1 2 5 4 1 5
182                  2 2 2 2 21 2 4 2 2 3 2 3 3 4 3 7 1 4 1 8 1 2 3 4)
183          OUT=&OUT1;
184          STRATA NEWSTRAT;
185          CONTROL AGE_COHORT SEX RACE;
186          ID ABTID STRATVAR AGED_DIS RACE SEX TRANSGEN AGE_COHORT
187              ETHNCTY CARDNUM MCRSTA BENEADR: BENE_ST STATE BENECITY
188              BENEFNAM BENEMI BENELNAM HIC NEWSTRAT
189              ZIPCODE;
190      RUN;
```

NOTE: THE DATA SET OUT1.SURVEY27 HAS 192 OBSERVATIONS AND 24 VARIABLES.
NOTE: THE PROCEDURE SURVEYSELECT PRINTED PAGE 9.
NOTE: THE PROCEDURE SURVEYSELECT USED 40.00 CPU SECONDS AND 6042K.

THE SURVEYSELECT PROCEDURE

SELECTION METHOD	SYSTEMATIC RANDOM SAMPLING
STRATA VARIABLE	NEWSTRAT

```
CONTROL VARIABLES    AGE_COHORT
                    SEX
                    RACE
CONTROL SORTING      SERPENTINE

INPUT DATA SET      TEMP4
RANDOM NUMBER SEED    12345678
NUMBER OF STRATA     68
TOTAL SAMPLE SIZE    192
OUTPUT DATA SET     SURVEY27

Variables created by PROC SURVEYSELECT:

SamplingWeight       SAMPLING WEIGHT
SelectionProb        PROBABILITY OF SELECTION
```

SAMPLE 2

```
NOTE: There were 7600 observations read from the data set WORK.UNIVERSE.
      WHERE eligtosamp=1;
NOTE: The data set WORK.TOBESAMPLED has 7600 observations and 80 variables.
NOTE: PROCEDURE SORT used (Total process time):
      real time          0.61 seconds
      cpu time           0.04 seconds

120
121      proc surveyselect data=tobesampled method=sys
122          seed=87654321
123          sampsize=(360 360 360 120)
124          out=lib.sample01;
125          strata sampcat;
126          control census_region rural;
127          id provider;
128      run;

NOTE: The CONTROL sorted data set replaces the DATA= input data set by default.  To store the sorted
data in an output data set, use the OUTSORT= option.
NOTE: The data set LIB.SAMPLE01 has 1200 observations and 6 variables.
NOTE: The PROCEDURE SURVEYSELECT printed page 6.
NOTE: PROCEDURE SURVEYSELECT used (Total process time):
      real time          1.53 seconds
      cpu time           0.12 seconds

OASIS-T01: PREPARE POS0412G FOR SAMPLING
CREATE SAMPLE FOR OASIS T01
```

The SURVEYSELECT Procedure

Selection Method	Systematic Random Sampling
Strata Variable	sampcat
Control Variables	census_region rural
Control Sorting	Serpentine
Input Data Set	TOBESAMPLED
Random Number Seed	87654321
Number of Strata	4
Total Sample Size	1200
Output Data Set	SAMPLE01

Below a screenshot of a spreadsheet analyzing the sampling frame or universe against the drawn sample. Note the effect of the serpentine sort on the control variables. The stratum variable was size category, while census region and urban/rural were control variables. Unlike a nested sort that would have yielded more proportional numbers, the serpentine sort simply ensured that all bases (combinations of control variables) were covered within each stratum. This is a very important distinction to understand. If you need to have a representative sample, you should use a nested sort or a different sampling method within PROC SURVEYSELECT. If you need, as I did, to have a sample in which all populations (as defined by strata and control variables) have a chance of being selected, then the serpentine sort is the way to go!

Microsoft Excel - uni1.xls

File Edit View Insert Format Tools Data Window Help

Arial 8 B I U

115 =

	A	B	C	D	E	F	G	H	I	J	K
1	OASIS-T01: PREPARE POS0412G FOR SAMPLING										
2											
3	CREATE SAMPLE FOR OASIS T01										
4											
5	Size Category	Census Region	Urban / Rural	Universe N	Universe %	Sample N	Sample %				
6	Small	North East	Urban	72	0.95	17	1.42				
7	Small	North East	Rural	11	0.14	2	0.17				
8	Small	South	Urban	480	6.32	112	9.33				
9	Small	South	Rural	176	2.32	41	3.42				
10	Small	Central	Urban	230	3.03	54	4.50				
11	Small	Central	Rural	298	3.92	70	5.83				
12	Small	West	Urban	159	2.09	37	3.08				
13	Small	West	Rural	117	1.54	27	2.25				
14	Medium	North East	Urban	196	2.58	23	1.92				
15	Medium	North East	Rural	81	1.07	9	0.75				
16	Medium	South	Urban	929	12.22	108	9.00				
17	Medium	South	Rural	450	5.92	52	4.33				
18	Medium	Central	Urban	512	6.74	60	5.00				
19	Medium	Central	Rural	391	5.14	45	3.75				
20	Medium	West	Urban	428	5.63	50	4.17				
21	Medium	West	Rural	112	1.47	13	1.08				
22	Large	North East	Urban	322	4.24	74	6.17				
23	Large	North East	Rural	40	0.53	9	0.75				
24	Large	South	Urban	529	6.96	121	10.08				
25	Large	South	Rural	158	2.08	36	3.00				
26	Large	Central	Urban	255	3.36	58	4.83				

Ready NUM

SAMPLE 3

Following the draw of the sample above (in Sample 2) there was a complication (other than my own project directors changing their minds several times regarding sample frames and sizes!) Another project needed to draw a sample from the same “universe”. Our sample as drawn would have made it impossible for the other project to obtain a sample using their stratum of state. We were able to reconfigure our sample in a manner similar to the method used for the drug card sample above, using a combination of state and size categories as the stratum variable instead of size category alone. Our end result was similar, but allowed the other project enough potential sample in their strata to obtain an adequate sample.

```
360      proc surveyselect data=tosamp method=sys outsort=sortsamp
361          seed=87654321
362          sampsiz=( /* Ntoget */ 2 3 1 1 2 11 1 3 8 1 14 2 2 9
363                  1 7 12 3 5 87 7 5 13 19 7 17 8 15 14 3 7 14
364                  7 8 3 3 24 7 1 6 6 5 3 4 2 3 4 2 4 2 2 1 6 15
365                  1 8 6 1 1 19 4 6 16 2 2 10 11 3 9 51 9 4 11 17
366                  11 5 13 11 7 1 2 16 1 7 1 4 34 6 1 2 1 3 2 3 2 3
367                  1 13 15 2 2 11 24 19 2 3 7 3 1 1 38 11 11 13 7 7
```

```
368          12 6 4 10 28 9 1 4 14 7 3 12 4 5 1 2 18 3 4 30 7 1
369          1 4 3 3 4 5 31 16 8)
370      out=lib.sample02;
371      strata stratum;
372      control rural;
373      id provider;
374      run;
```

```
OASIS-T01: PREPARE POS0412G FOR SAMPLING
CREATE SAMPLE FOR OASIS T01 - ROUND 3
```

The SURVEYSELECT Procedure

Selection Method	Systematic Random Sampling
Strata Variable	stratum
Control Variable	rural

Input Data Set	TOSAMP
Sorted Data Set	SORTSAMP
Random Number Seed	87654321
Number of Strata	147
Total Sample Size	1200
Output Data Set	SAMPLE02

CONCLUSION

PROC SURVEYSELECT is an extremely powerful and versatile tool for the selection of both simple and complex sample designs. The procedure allows for statistically defensible probability-based random sampling via a number of different methods including equal probability sampling and PPS (probability proportional to size) sampling. The examples I have shown are a drop in the bucket compared to the vast capability of PROC SURVEYSELECT. Paired with the robust survey analysis procedures such as SURVEYLOGISTIC, SURVEYMEANS, etc. not mentioned in this paper, SAS® provides us with “one stop shopping” in the area of survey implementation and analysis, making it a clear choice for both SAS programmers and sampling statisticians.

REFERENCES

SAS Online Documentation (SAS V9.1)

ACKNOWLEDGMENTS

K.P Srinath of Abt Associates Inc. has been my guide and mentor in the world of statistical sampling and analysis.

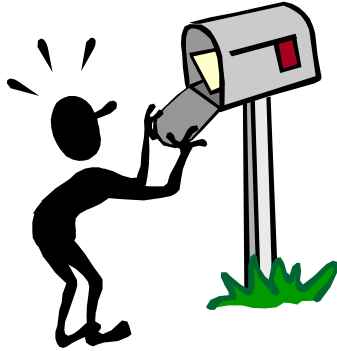
SAS® Technical Support and R&D have been incredibly helpful.

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are registered trademarks or trademarks of their respective companies.



CONTACT INFORMATION

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



Louise Hadden
Abt Associates Inc.
55 Wheeler St.
Cambridge, MA 02138

Work Phone: 617-349-2385
Fax: 617-349-2675

Email: louise_hadden@abtassoc.com

KEYWORDS

SAS; PROC SURVEYSELECT; SAMPLING; RANDOM; PROPORTIONAL; SERPENTINE;
SYSTEMATIC

