Bootstrap Sampling in SPSS, an example of Getting Means from Many Samples[1]

It is possible to use macros in SPSS to do iterative processes. Following is an example of a program that repeatedly gets a data set, draws a 20% random sample, calculates the mean and standard deviation of the sample, and adds these into a file that has the means and standard deviations from the previous samples drawn. This approximates a bootstrap process for sampling. (Lines that begin with an asterisk document the file, and are not executed by SPSS.)

```
*bootstrp.sps, get means and standard deviations for mutiple samples.

** Step 1 .
** get mean of whole sample for comparison .
get file =  'C:\sample\infile.sav'  / keep   age  .
desc var =  age  .

** Step 2 .
* Create an empty file called outfile.save which will be used later.
* Overwrite any existing file called 'C:\sample\outfile.sav' .
compute allv    =  1 .
select if allv  =  0 .
compute meanage = -9 .
compute nage    = -9 .
compute sdage   = -9 .
missing values meanage nage sdage (-9) .
save outfile = 'C:\sample\outfile.sav'  / keep meanage nage sdage .
** end of step 1.

** Step 3 .
* Create a macro that will get the whole data set, create a random number,.
* draw a sample, and get the mean, N, and standard deviation.
* The seed for the random number generator is changed with .
* each iteration - it uses "!i" .
DEFINE getmeans () .
**** change "100" after !TO to any other number - number of samples .
!DO !i = 1 !TO 100  !BY 1  .
get file =  'C:\sample\infile.sav'  / keep  age  .
compute allv = 1 .
* Uniform creates a random number ranging from 0 to value of seed .
compute ran1 = uniform(!i).

** The "compute selvar = ( !i / 5 )."  will create a 20% sample.
** For example, (!i/4) would create a 25% sample, and (!i/10) a 10% sample.
compute selvar = ( !i / 5 ).
select if ran1 le selvar .
aggregate  outfile =  *
 / break   = allv
 / meanage = mean( age )
 / nage    = n( age )
 / sdage   = sd( age ) .
add files file = * / file =  'C:\sample\outfile.sav'  .
save outfile = 'C:\sample\outfile.sav'  .
!DOEND .
!enddefine .
** End of step 2 .

** execute the macro .
getmeans .

* Get the file that the macro created and run descriptives on it.
get file = 'C:\sample\outfile.sav'  .
desc var = all .
execute .
```

---

[1]**Prepared by Patty Glynn,, University of Washington. January 31, 2001** C:\all\help\helpnew\bootstrp.wpd