Variance estimation using bootstrap weights

BOOTVAR.SAS program user's guide for use with the 1998-99 NPHS dummy file

1. Bootstrap resampling method

Due to the complexity of the sample design, a resampling technique is used to calculate estimates of the variance. For the National Population Health Survey (NPHS), the bootstrap resampling method is used.

This technique involves dividing the records on the data file into subgroups (or replicates) and determining the variation in the estimates from replicate to replicate. The replicates are formed in the following manner: within each stratum, a simple random sample (SRS) is taken, with replacement, of (n-1) of the n clusters in that stratum. For each SRS sample, we recalculate the survey weight for each record in that stratum. This recalculated weight is the first bootstrap weight. We then repeat the process B times, forming a set of B bootstrap weights for each record on the sample file.

2. Bootstrap weights for the NPHS

For NPHS, three bootstrap weight files were created. The file BD5H35 is used with the DUMYH35 dummy file, the file BD5H356 with the DUMYH356 dummy file, and finally, the file BD5DUMY is used with the longitudinal dummy file DUMYLNGF. All three files contain 500 bootstrap weights for each record.

The bootstrap weights are used to obtain accurate variance estimates for statistics such as totals and ratios, and for more complex analyses such as regressions. The same confidentiality rules and release cut-off limits are in effect for bootstrap variance estimates.

The proper set of weights must be used to calculate a valid estimate of the variance for a given estimate. For example, if one wants to calculate an estimate for a total using the file DUMYH35, one must take the following steps:

A) first calculate an estimate of the total using the final weight WT58

B) then calculate an estimate of the total using each of the 500 bootstrap weights in the file BD5H35 as the weighting factor. This gives 500 estimates of the total

C) finally, calculate the variance among those 500 estimates. This variance is an estimate of the variance of the total calculated in A).

3. Program BOOTVAR.SAS

The SAS program BOOTVAR.SAS given with the data files is an example of a program to calculate an estimate of the variance. Users may use this program or they can create their own program. A copy of the program appears at the end of this section.

This program calculates an estimate (for example, a total) and a corresponding estimate of the variance. The user should always calculate the desired estimate (in this case the total) using another program, in order to verify that the program BOOTVAR.SAS has correctly calculated this estimate.

In this section, we describe how to prepare the data for the program BOOTVAR.SAS, and how to use it. Two examples are given as well as the time required to run the program. (Please note that these programs were run on a 500 MHz Pentium Pro, with 128 MB of RAM.)

3.1 Definition of variables for the analysis

The program BOOTVAR.SAS permits the calculation of variance estimates for totals, for ratios, for differences of ratios, and for linear and logistic regression parameters. To obtain the desired analyses, the data must first be prepared and saved in a SAS file. For estimates of totals, ratios, and differences of ratios, the estimates are calculated by summing the weights of the records with the characteristic of interest. An indicator variable must be created to identify these records (see example 1A). For regression analyses, the variables to be analysed must be defined according to SAS conventions (see example 1B).

A supplementary variable (REGION) may be necessary if the analyses are done for geographic subgroups of records on the new datafile created for the program BOOTVAR.SAS. For example, if the new analysis file contains all the records of persons aged twelve and older in Canada, the file represents a group that corresponds to Canada. If the analysis is done at the Canada level, the variable REGION is not necessary. However, if the analysis is done at the provincial level, then the REGION variable is necessary. This variable must be created by the user and, for each record, will take the value of the geographic region of this record—in this case, the province. In the same way, if the new analysis file contains all the records of persons aged twelve and older in the provinces of Ontario, Manitoba, and Alberta, the REGION variable will be necessary if the analysis is done at the province level. Here again, the variable will take the value of the province.

The following example shows how to define the variables that will subsequently be used to calculate variance estimates with the aid of BOOTVAR.SAS.

Example 1: Definition of variables for the analysis

Example 1A: Totals and ratios

```
/*********************************************************************/
/*                                                                   */
/*         USE THE CROSS-SECTIONAL FILE FOR THE 1998 GENERAL COMPONENT */
/*         (DUMYH35.TXT)                                            */
/*         TO CALCULATE THE NUMBER OF DIABETIC PERSONS               */
/*         FOR THE AGE GROUPS 12-17, 18-24, 25-44, AND 45 AND OVER   */
/*         AND FOR MALES, FEMALES, AND BOTH,                         */
/*         BY PROVINCE                                               */
/*                                                                   */
/*********************************************************************/

libname in1 'c:\data';

proc datasets kill;
run;

proc format;
  value fsex    1='males  '
                2='females';

  value fage    1='12-17'
                2='18-24'
                3='25-44'
                4='45+';

  value fdiab   1='Yes'
                0='No ';
run;

data in1.cross;
%let datafid='c:\data\DUMYH35.txt';
%include 'c:\data\h35_i.sas';
keep region diab total dhc8_sex males females
     agegrp age1217 age1824 age2544 age45 magegrp fagegrp
     mage1217 mage1824 mage2544 mage45 fage1217 fage1824 fage2544 fage45
     mdiab fdiab diab1217 diab1824 diab2544 diab45
     mdia1217 mdia1824 mdia2544 mdia45 fdia1217 fdia1824 fdia2544 fdia45
     wt58 total realukey personid;

region=prc8_cur;

if ccc8_1j=1 then diab=1;
else if ccc8_1j=2 then diab=0;
else diab=.;

if diab=. then delete;

/*********************************************/
/* defines the domain of interest with dummy  */
/* variables (0/1)                            */
/*********************************************/

total=1;

 /* sex */

if dhc8_sex=1 then males=1;
else males=0;

if dhc8_sex=2 then females=1;
else females=0;


 /* age group */

if dhc8_age < 18 then agegrp=1;
else if dhc8_age < 25 then agegrp=2;
else if dhc8_age < 45 then agegrp=3;
else agegrp=4;

if agegrp=1 then age1217=1; else age1217=0;
if agegrp=2 then age1824=1; else age1824=0;
if agegrp=3 then age2544=1; else age2544=0;
if agegrp=4 then age45=1; else age45=0;


 /* age group and sex */

if agegrp=1 then do;
```

```
  if dhc8_sex=1 then magegrp=1;
  else if dhc8_sex=2 then fagegrp=1;
end;
else if agegrp=2 then do;
  if dhc8_sex=1 then magegrp=2;
  else if dhc8_sex=2 then fagegrp=2;
end;
else if agegrp=3 then do;
  if dhc8_sex=1 then magegrp=3;
  else if dhc8_sex=2 then fagegrp=3;
end;
else if agegrp=4 then do;
  if dhc8_sex=1 then magegrp=4;
  else if dhc8_sex=2 then fagegrp=4;
end;

if magegrp=1 then mage1217=1; else mage1217=0;
if magegrp=2 then mage1824=1; else mage1824=0;
if magegrp=3 then mage2544=1; else mage2544=0;
if magegrp=4 then mage45=1; else mage45=0;

if fagegrp=1 then fage1217=1; else fage1217=0;
if fagegrp=2 then fage1824=1; else fage1824=0;
if fagegrp=3 then fage2544=1; else fage2544=0;
if fagegrp=4 then fage45=1; else fage45=0;


 /* diabetes*sex */

if diab=1 then do;
  if dhc8_sex=1 then mdiab=1; else mdiab=0;   /* diabetic males   */
  if dhc8_sex=2 then fdiab=1; else fdiab=0;   /* diabetic females */
end;
else if diab=2 then do;
  mdiab=0;
  fdiab=0;
end;


 /* diabetes*agegroup */

if diab=1 then do;
  if age1217=1 then diab1217=1; else diab1217=0;
  if age1824=1 then diab1824=1; else diab1824=0;
  if age2544=1 then diab2544=1; else diab2544=0;
  if age45=1   then diab45=1; else diab45=0;
end;
else do;
  diab1217=0;
  diab1824=0;
  diab2544=0;
  diab45=0;
end;


 /* diabetes*sex*agegroup */

if mdiab=1 then do;
  if age1217=1 then mdia1217=1; else mdia1217=0;
  if age1824=1 then mdia1824=1; else mdia1824=0;
  if age2544=1 then mdia2544=1; else mdia2544=0;
  if age45=1   then mdia45=1; else mdia45=0;
end;
else do;
  mdia1217=0;
  mdia1824=0;
  mdia2544=0;
  mdia45=0;
end;

if fdiab=1 then do;
  if age1217=1 then fdia1217=1; else fdia1217=0;
  if age1824=1 then fdia1824=1; else fdia1824=0;
  if age2544=1 then fdia2544=1; else fdia2544=0;
  if age45=1   then fdia45=1; else fdia45=0;
end;
else do;
  fdia1217=0;
  fdia1824=0;
  fdia2544=0;
  fdia45=0;
end;

proc freq data=in1.cross;
format agegrp fage. magegrp fage. fagegrp fage. diab fdiab.;
table region*(diab diab2544 mdia45 fdia45)
      region*diab*total region*mdia45*mage45 region*diab45*age45 ;
```

```
weight wt58;
title 'Number of diabetic by agegroup and region';
run;
```

(The execution of this program took about 2 minutes.)

Example 1B: Logistic regression

```
/*********************************************************************/
/*                                                                   */
/*         USE THE CROSS-SECTIONAL FILE FOR THE 1998 HEALTH COMPONENT  */
/*         (DUMYH356.TXT)                                            */
/*         TO STUDY THE RELATION BETWEEN CHRONIC CONDITION REPORTING   */
/*         AND THE TYPE OF INTERVIEW (PROXY VS NON-PROXY)            */
/*                                                                   */
/*********************************************************************/


libname in1 'c:\data';

data in1.DUMYH356;
  %let datafid="c:\data\DUMYH356.txt";
  %include "c:\data\h356_i.sas";
  keep realukey personid dhc8_age dhc8_sex am58_pxy ccc8dany
  age1217 age1820 age2124 age2534 age3544 age4554 age5564
  age6574 age7584 age85 female nonproxy wt68;
  if dhc8_age >= 12;

/* Dummy variables for the age groups */

  age1217=0;
  age1820=0;
  age2124=0;
  age2534=0;
  age3544=0;
  age4554=0;
  age5564=0;
  age6574=0;
  age7584=0;
  age85=0;

  if 12 <= dhc8_age <=17 then age1217=1;
  if 18 <= dhc8_age <=20 then age1820=1;
  if 21 <= dhc8_age <=24 then age2124=1;
  if 25 <= dhc8_age <=34 then age2534=1;
  if 35 <= dhc8_age <=44 then age3544=1;
  if 45 <= dhc8_age <=54 then age4554=1;
  if 55 <= dhc8_age <=64 then age5564=1;
  if 65 <= dhc8_age <=74 then age6574=1;
  if 75 <= dhc8_age <=84 then age7584=1;
  if dhc8_age>=85 then age85=1;

  female=0;
  if dhc8_sex=2 then female=1;

/* Missing values */

  if ccc8dany>2 then ccc8dany=.;
  if ccc8dany=2 then ccc8dany=0;

  nonproxy=0;
  if am58_pxy>2 then nonproxy=.;
  if am58_pxy=2 then nonproxy=1;
run;

proc logistic data=in1.DUMYH356 descending;
     model ccc8dany = nonproxy age1820 age2124 age2534 age3544
       age4554 age5564 age6574 age7584 age85 female;
     weight wt68;
title 'Chronic conditions';
run;
```

(The execution of this program took about 1 minute.)

3.2 Use of the program BOOTVAR.SAS

Once the new analysis file has been created, BOOTVAR.SAS can be run. However, the user must complete the program by specifying the parameters and by indicating the list of variables to analyse. BOOTVAR.SAS is divided into four sections.

In **section 1**, the users must specify the following:
- the number of bootstrap weights used (500 for each of the three files; DUMYH35, DUMYH356 and DUMYLNGF),
- the subdirectories containing the new analysis file (created by the user, as in example 1), the bootstrap weight file, and the file that will contain the results,
- the name of the new analysis file,
- the name of the bootstrap weight file (the file must be a SAS datafile), and
- the three parameters that indicate if the analysis is done at a geographic level.

In **section 2**, changes must be made only if the user wants to compute a general linear model. These changes are to the macro defined for the general linear model. This macro must be modified by the user according to the analysis required.

In **section 3**, the user must specify the list of variables to keep for the analysis. BOOTVAR.SAS runs quickly for variance estimates of totals, ratios, and differences of ratios. However, more time is required for variance estimates of regression parameters. To reduce the run time, it is therefore recommended to keep only those variables necessary for the desired analysis.

In addition to the analysis variables, two variables must be kept on the file, REALUKEY and PERSONID. These variables are needed to match the survey datafile and the bootstrap weight file during the running of the program.

Finally, in **section 4**, the user must indicate which analyses will be run and for which variables.

The following examples indicate in ***italics*** those parts of the program that the user must specify for variance estimates of totals and of ratios (example 2A), and for logistic regression parameters (example 2B). Please note that only those parts of the program where changes are necessary are presented. It is necessary, however, to submit the entire program.

Example 2: BOOTVAR.SAS

<u>Example 2A: Totals and ratios</u>

```
.
.
.
/*************************************************************************************/
/*** Section 1: Declaration of the macro variables                               ***/
/*************************************************************************************/

%let R=01;                        /* bootstrap averaging parameter (fixed) */

/*****************************************************************************/
/*** USER SPECIFIES THE NUMBER OF BOOTSTRAP WEIGHTS USED, DIVIDED BY 100   ***/
/*** %let B=5;   for the bootstrap weights files BD5H35, BD5H356 AND       ***/
/***  BD5LNGF                                                               ***/
/*****************************************************************************/

%let B=5;                                 /* bootstrap Bx100 times */


/*************************************************************************/
/*** USER SPECIFIES THE DIRECTORY CONTAINING THE ANALYSIS FILE (in1)  ***/
/*** AND THE BOOTSTRAP WEIGHTS FILE (in2) SPECIFIED BELOW.            ***/
/*** THE OUTPUT FILES WILL BE SAVED IN THE DIRECTORY SPECIFIED FOR in3 **/
/*************************************************************************/

libname in1 "c:\data";
libname in2 "c:\bootstrp";
libname in3 "c:\output";


/*************************************************************************************/
/*** USER SPECIFIES THE NAME OF THE SAS FILE CONTAINING THE ANALYSIS VARIABLES.  */
/*** %let Mfile=xxx;                                                             */
/*************************************************************************************/

%let Mfile=in1.cross;                         /* Main SAS file */


/*************************************************************************************/
/*** USER SPECIFIES THE NAME OF THE SAS FILE CONTAINING THE BOOTSTRAP WEIGHTS  ***/
/*** IF THE SAS FILE DOES NOT YET EXIST IT NEEDS TO BE CREATED.                ***/
/*** %let bsamp=bd5h35;   (to be used with the file DUMYH35.TXT)    OR        ***/
/*** %let bsamp=bd5h356;  (to be used with the file DUMYH356.TXT)   OR         ***/
/*** %let bsamp=bd5lngf;  (to be used with the file DUMYLNGF.TXT)              ***/
/*************************************************************************************/

data in2.bd5h35;
  %let datafid="c:\bootstrp\bd5h35.txt";
  %include "c:\bootstrp\b35_i.sas";
run;

%let bsamp=in2.bd5h35;                  /* bootstrap weights SAS file */


/*******************************************************************/
/*** USER SPECIFIES THE PARAMETERS by, wo AND kp, WHICH INDICATE ***/
/*** IF THE ANALYSIS IS DONE AT THE REGIONAL LEVEL OR FOR THE    ***/
/*** WHOLE DATASET.                                               ***/
/***                                                              ***/
/*** IF THE ANALYSIS IS DONE FOR THE WHOLE DATASET, THEN         ***/
/***    %let by=*;                                                ***/
/***    %let wo=;                                                 ***/
/***    %let kp=;                                                 ***/
/***                                                              ***/
/*** ELSE IF THE ANALYSIS IS DONE AT THE REGIONAL LEVEL, THEN    ***/
/***    %let by=;                                                 ***/
/***    %let wo=*;                                                ***/
/***    %let kp=region;                                           ***/
/*******************************************************************/

%let by=;
%let wo=*;
%let kp=region;

data time;
  format start datetime16.;
  start=datetime();
  output;
run;
```

```
/**********************************************************************************/
/*** Section 2: Declaration of the macros                                      ***/
/**********************************************************************************/

.
.
.

*********************************************************************;
*  Section 3: MAIN PROGRAM                                           *;
*********************************************************************;

/* Read in Main file */

/***********************************************************************/
/*** DUE TO THE LARGE NUMBER OF OBSERVATIONS, ONLY THE VARIABLES      ***/
/*** THAT ARE NEEDED TO DO THE ANALYSES AND THE VARIABLES             ***/
/*** "REALUKEY" AND "PERSONID" SHOULD BE KEPT AT THIS POINT.        ***/
/***********************************************************************/

   data nphs (index=(id=(realukey personid)));
     set &Mfile (keep= realukey personid region diab total dhc8_sex
                       males females agegrp age1217 age1824 age2544 age45
                       magegrp fagegrp mage1217 mage1824 mage2544 mage45
                       fage1217 fage1824 fage2544 fage45 mdiab fdiab
                       diab1217 diab1824 diab2544 diab45 mdia1217 mdia1824
                       mdia2544 mdia45 fdia1217 fdia1824 fdia2544 fdia45
                       wt58 total);
   run;

/* Read in Bootstrap Weights */
/* FWGT corresponds to WT58, WT68 or WT68LF, depending on which file is analysed */

   data bsamp (index=(id=(realukey personid)));
     set &bsamp;
     keep fwgt realukey personid bsw1-bsw&B.00;
   run;


/* Merge Main file and Bootstrap Weights */

&by data bs_nphs (index=(region));
&wo data bs_nphs;
      merge nphs (in=in1) bsamp (in=in2);
      by realukey personid;
      if in1;
    run;

   proc datasets library=work;
     delete nphs bsamp;
   run;

/**********************************************************************************/
/*** Section 4: Macro calls                                                    ***/
/**********************************************************************************/

/***********************************************************************************/
/*** The variance estimates are calculated using the SAS macros that have been  ***/
/*** defined earlier in the program.  Those SAS macros can be called in the     ***/
/*** program as needed.  A macro call is a SAS statement that specifies the name **/
/*** of the macro called and some parameters.  In this program, the parameters  ***/
/*** indicate which file has to be read, which variables will be used, how many ***/
/*** bootstrap weights are used, and how many bootstrap samples were selected.  ***/
/*** The name of the file to read is always the same (BS_NPHS) and the last two ***/
/*** parameters have already been defined in the first section of the program.  ***/
/*** Hence, the user only has to specify the variables to be used.              ***/
/***                                                                            ***/
/*** Each macro call will result in a variance estimate of ONE estimator.  If   ***/
/*** more than one estimator and its variance estimate have to be calculated,   ***/
/*** the macro has to be called multiple times.                                 ***/
/***                                                                            ***/
/*** A commented statement for the macro call appears where the user's statement***/
/*** should be placed in the program.  The user's statement will               ***/
/*** be identical to the commented one, except for the names of the analysis    ***/
/*** variables.                                                                 ***/
/***********************************************************************************/
```

```
/*********/
/*       */
/* Total */
/*       */
/*********/

   proc datasets library=work; /* initial alltots */
     delete alltots;
   run;

/*****************************************************************************/
/*** To call the macro total, the following statement is used:            ***/
/*** %total(infile,var,bssz=,multi=);                                      ***/
/***       where infile : bs_nphs                                          ***/
/***             var    : the variable for which a total is calculated     ***/
/***             bssz=  : bssz=&B.00                                       ***/
/***             multi= : multi=&R                                         ***/
/***                                                                       ***/
/*****************************************************************************/

 *  %total(bs_nphs,VAR,bssz=&B.00,multi=&R);
    %total(bs_nphs,diab,bssz=&B.00,multi=&R);
    %total(bs_nphs,diab2544,bssz=&B.00,multi=&R);
    %total(bs_nphs,mdia45,bssz=&B.00,multi=&R);
    %total(bs_nphs,fdia45,bssz=&B.00,multi=&R);

/* Delete sas files in total macro */

   proc datasets library=work;
     delete ytot est;
   run;


/*********/
/*       */
/* Ratio */
/*       */
/*********/

/*******************************************************************************/
/*** To call the macro ratio, the following statement is used:               ***/
/*** %ratio(infile,var1,var2,bssz=,multi=);                                  ***/
/***       where infile : bs_nphs                                            ***/
/***             var1   : the variable of the numerator of the ratio         ***/
/***             var2   : the variable of the denominator of the ratio       ***/
/***             bssz=  : bssz=&B.00                                         ***/
/***             multi= : multi=&R                                           ***/
/***                                                                         ***/
/*******************************************************************************/


 *  %ratio(bs_nphs,VAR1,VAR2,bssz=&B.00,multi=&R);
    %ratio(bs_nphs,diab,total,bssz=&B.00,multi=&R);
    %ratio(bs_nphs,mdia45,mage45,bssz=&B.00,multi=&R);
    %ratio(bs_nphs,diab45,age45,bssz=&B.00,multi=&R);

/* Delete sas files in ratio macro */
   proc datasets library=work;
     delete ytot xtot est;
   run;

&by   proc sort data=alltots;
&by     by region;
&by   run;

/*********************************************/
/*** The results of total and ratio macro ***/
/*** can be saved in a permanent file.    ***/
/*********************************************/

   data in3.ZZZ;
     set alltots;
   run;

%prnttot;

.
.
.

data time;
  set time;
  format stop datetime16.;
  stop=datetime();
  output;
run;
```

```
proc print data=time;
  title 'Time Taken to Run Program';
run;
/* End of SAS program BootVar */
```

(The execution of this program took about 6 minutes.)

Example 2B: Logistic regression

```
.
.
.

/*********************************************************************************/
/*** Section 1: Declaration of the macro variables                            ***/
/*********************************************************************************/

%let R=01;                          /* bootstrap averaging parameter (fixed) */

/****************************************************************************/
/*** USER SPECIFIES THE NUMBER OF BOOTSTRAP WEIGHTS USED, DIVIDED BY 100   ***/
/*** %let B=5;   for the bootstrap weights files BD5H35, BD5H356 and       ***/
/*** BD5LNGF                                                                ***/
/****************************************************************************/

%let B=5;                                /* bootstrap Bx100 times */


/***********************************************************************/
/*** USER SPECIFIES THE DIRECTORY CONTAINING THE ANALYSIS FILE (in1)  ***/
/*** AND THE BOOTSTRAP WEIGHTS FILE (in2) SPECIFIED BELOW.            ***/
/*** THE OUTPUT FILES WILL BE SAVED IN THE DIRECTORY SPECIFIED FOR in3 **/
/***********************************************************************/

libname in1 "c:\data";
libname in2 "c:\bootstrp";
libname in3 "c:\output";


/*********************************************************************************/
/*** USER SPECIFIES THE NAME OF THE SAS FILE CONTAINING THE ANALYSIS VARIABLES.  */
/*** %let Mfile=xxx;                                                             */
/*********************************************************************************/

%let Mfile=in1.DUMYH356;                          /* Main SAS file name */


/*********************************************************************************/
/*** USER SPECIFIES THE NAME OF THE SAS FILE CONTAINING THE BOOTSTRAP WEIGHTS  ***/
/*** IF THE SAS FILE DOES NOT YET EXIST IT NEEDS TO BE CREATED.                ***/
/*** %let bsamp=bd5h35;   (to be used with the file DUMYH35.TXT)    OR         ***/
/*** %let bsamp=bd5h356;  (to be used with the file DUMYH356.TXT)   OR         ***/
/*** %let bsamp=bd5LNGF;  (to be used with the file DUMYLNGF.TXT)              ***/
/*********************************************************************************/

data in2.bd5h356;
  %let datafid="c:\bootstrp\bd5h356.txt";
  %include "c:\bootstrp\b356_i.sas";
run;

%let bsamp=in2.bd5h356;             /* bootstrap weights SAS file */


/********************************************************************/
/*** USER SPECIFIES THE PARAMETERS by, wo AND kp, WHICH INDICATE ***/
/*** IF THE ANALYSIS IS DONE AT THE REGIONAL LEVEL OR FOR THE    ***/
/*** WHOLE DATASET.                                              ***/
/***                                                             ***/
/*** IF THE ANALYSIS IS DONE FOR THE WHOLE DATASET, THEN         ***/
/***    %let by=*;                                               ***/
/***    %let wo=;                                                ***/
/***    %let kp=;                                                ***/
/***                                                             ***/
/*** ELSE IF THE ANALYSIS IS DONE AT THE REGIONAL LEVEL, THEN    ***/
/***    %let by=;                                                ***/
/***    %let wo=*;                                               ***/
/***    %let kp=region;                                          ***/
/********************************************************************/

%let by=*;
%let wo=;
```

```
%let kp=;

data time;
  format start datetime16.;
  start=datetime();
  output;
run;

/*********************************************************************************/
/*** Section 2: Declaration of the macros                                     ***/
/*********************************************************************************/

.
.
.

*********************************************************************;
*  Section 3: MAIN PROGRAM                                          *;
*********************************************************************;

/* Read in Main file */

/*************************************************************************/
/*** DUE TO THE LARGE NUMBER OF OBSERVATIONS, ONLY THE VARIABLES      ***/
/*** THAT ARE NEEDED TO DO THE ANALYSES AND THE VARIABLES             ***/
/*** "REALUKEY" AND "PERSONID" SHOULD BE KEPT AT THIS POINT.          ***/
/*************************************************************************/

   data nphs (index=(id=(realukey personid)));
     set &Mfile (keep= realukey personid ccc8dany nonproxy age1820 age2124 age2534
                       age3544 age4554 age5564 age6574 age7584 age85 female);
   run;

/* Read in Bootstrap Weights */
/* FWGT corresponds to WT58, WT68 or WT68LF, depending on which file is analysed */

   data bsamp (index=(id=(realukey personid)));
     set &bsamp;
     keep fwgt realukey personid bsw1-bsw&B.00;
   run;


/* Merge Main file and Bootstrap Weights */

&by data bs_nphs (index=(region));
&wo data bs_nphs;
      merge nphs (in=in1) bsamp (in=in2);
      by realukey personid;
      if in1;
    run;

   proc datasets library=work;
     delete nphs bsamp;
   run;


/*********************************************************************************/
/*** Section 4: Macro calls                                                   ***/
/*********************************************************************************/

.
.
.

/***********************/
/*                     */
/* Logistic Regression */
/*                     */
/***********************/

   proc datasets library=work; /* initial bs_reglg */
     delete bs_reglg;
   run;

/******************************************************************************/
/*** To call the macro logreg, the following statement is used:            ***/
/*** %logreg(infile,yvar,xvar,bssz=,multi=);                               ***/
/***       where infile : bs_nphs                                          ***/
/***             yvar   : the dependent variable                           ***/
/***             xvar   : the independent variables.  The variables should ***/
/***                      be listed as they would be in a model statement  ***/
/***                      (no commas in between).                          ***/
/***             bssz=  : bssz=&B.00                                       ***/
/***             multi= : multi=&R                                         ***/
/***                                                                       ***/
/******************************************************************************/
```

```
     title "Bootstrap &B.00:&R Variance Estimate for";
     title2 "Logistic Regressions";

 *  %logreg(bs_nphs,YVAR,XVAR,bssz=&B.00,multi=&R);
 %logreg(bs_nphs,ccc8dany,nonproxy age1820 age2124 age2534 age3544
       age4554 age5564 age6574 age7584 age85 female,bssz=&B.00,multi=&R);

/* Delete sas files in logreg macro */
   proc datasets library=work;
     delete betas betat bsbeta origest;
   run;

.
.
.

%prntlog;


data time;
  set time;
  format stop datetime16.;
  stop=datetime();
  output;
run;

proc print data=time;
  title 'Time Taken to Run Program';
run;

/* End of SAS program BootVar */
```

(The execution of this program took about 12 minutes.)

The following examples present the results of the analyses done using the programs from the example 2. Results for the totals and ratios are presented in the example 3A. For example, if we want the ratio of the number of diabetic males aged 45 years old and over to the total number of males 45 years old and over, in Ontario, we look at observation 41. The region 35 corresponds to the province of Ontario (see the data dictionary documents included on the CD-ROM for the codes associated with each province) and the variable Type indicates the type of analysis, in this case a ratio. We find the variables *mdia45* (VAR1) at the numerator of the ratio and *mage45* (VAR2) at the denominator. The estimate of the ratio is 8.75% (YHAT) with a standard deviation of 0.74 (BS_SD) and a coefficient of variation of 8.48 (BS_CV). The 95% confidence interval for this estimate is (7.30%, 10.21%) (CIL95, CIU95). Following the results, we can see that the execution of this program started at 11:48 and finished at 11:54.

Results from the logistic regression are shown in example 3B. For example, the estimate of the parameter for the variable AGE85 is 2.55009 (BHAT) and the odds ratio is 12.8083 (ODDS). The Wald's statistic for this parameter and its p-value associated are 75.068 (WALD) and p=0.0000 (PVALUE) respectively. The variance estimate and the standard deviation for the parameter estimate are 0.086628 (BS_VAR) and 0.29433 (BS_SD) and the coefficient of variation is 11.54 (BS_CV). Finally, the confidence interval for the odds ratio is (7.19376, 22.8049) (CIL95,CIU95). The execution of this program started at 12:05 and finished at 12:18.

Example 3: Results of BOOTVAR.SAS and the time required to run the program

Example 3A: Totals and ratios

Bootstrap 500:01 Variance Estimate for
Totals and Ratios

OBS	REGION	TYPE	VAR1	VAR2	YHAT	BS_SD	BS_CV	CIL95	CIU95
1	10	Total	diab	None	23459.46	4023.36	17.15	15573.67	31345.24
2	10	Total	diab2544	None	2 717.28	742.35	27.32	1262.28	4172.27
3	10	Total	mdia45	None	7865.93	2405.56	30.58	3151.04	12580.83
4	10	Total	fdia45	None	12049.48	2022.95	16.79	8084.49	16014.47
5	10	Ratio	diab	total	5.07	0.87	17.15	3.37	6.78
6	10	Ratio	mdia45	mage45	8.75	2.68	30.58	3.51	14.00
7	10	Ratio	diab45	age45	10.77	2.13	19.79	6.59	14.94
8	11	Total	diab	None	4784.52	517.46	10.82	3770.31	5798.74
9	11	Total	diab2544	None	623.63	232.30	37.25	168.33	1078.93
10	11	Total	mdia45	None	1943.29	310.95	16.00	1333.84	2552.75
11	11	Total	fdia45	None	1839.87	302.41	16.44	1247.14	2432.60
12	11	Ratio	diab	total	4.23	0.46	10.82	3.33	5.13
13	11	Ratio	mdia45	mage45	8.71	1.39	16.00	5.98	11.44
14	11	Ratio	diab45	age45	8.02	0.97	12.10	6.12	9.93
15	12	Total	diab	None	38267.79	4287.88	11.20	29863.54	46672.03
16	12	Total	diab2544	None	6456.21	1854.35	28.72	2821.70	10090.73
17	12	Total	mdia45	None	15619.83	2491.51	15.95	10736.47	20503.20
18	12	Total	fdia45	None	14654.13	2866.83	19.56	9035.14	20273.11
19	12	Ratio	diab	total	4.95	0.55	11.20	3.86	6.03
20	12	Ratio	mdia45	mage45	10.13	1.61	15.94	6.96	13.29
21	12	Ratio	diab45	age45	9.20	1.12	12.17	7.00	11.39
22	13	Total	diab	None	22354.83	2691.78	12.04	17078.94	27630.73
23	13	Total	diab2544	None	2799.00	940.71	33.61	955.21	4642.80
24	13	Total	mdia45	None	10018.16	1827.96	18.25	6435.36	13600.96
25	13	Total	fdia45	None	8893.34	1780.10	20.02	5404.34	12382.34
26	13	Ratio	diab	total	3.57	0.43	12.04	2.73	4.41
27	13	Ratio	mdia45	mage45	8.07	1.47	18.25	5.18	10.95
28	13	Ratio	diab45	age45	7.22	1.03	14.31	5.20	9.25
29	24	Total	diab	None	226198.46	20426.77	9.03	186161.98	266234.94
30	24	Total	diab2544	None	36862.51	7658.00	20.77	21852.84	51872.19
31	24	Total	mdia45	None	89858.85	10787.17	12.00	68716.00	111001.70
32	24	Total	fdia45	None	94001.81	14819.64	15.77	64955.30	123048.31
33	24	Ratio	diab	total	3.71	0.34	9.03	3.06	4.37
34	24	Ratio	mdia45	mage45	7.49	0.90	12.01	5.72	9.25
35	24	Ratio	diab45	age45	7.19	0.70	9.80	5.81	8.57
36	35	Total	diab	None	353348.85	18540.87	5.25	317008.74	389688.96
37	35	Total	diab2544	None	34493.56	5516.52	15.99	23681.18	45305.94
38	35	Total	mdia45	None	159005.90	13494.40	8.49	132556.87	185454.92
39	35	Total	fdia45	None	151779.96	12197.96	8.04	127871.96	175687.96
40	35	Ratio	diab	total	3.74	0.20	5.25	3.35	4.12
41	35	Ratio	mdia45	mage45	8.75	0.74	8.48	7.30	10.21
42	35	Ratio	diab45	age45	8.09	0.46	5.67	7.19	8.99
43	46	Total	diab	None	29870.05	4389.76	14.70	21266.13	38473.97
44	46	Total	diab2544	None	4440.39	1624.83	36.59	1255.71	7625.06
45	46	Total	mdia45	None	9702.62	2150.52	22.16	5487.61	13917.64
46	46	Total	fdia45	None	14237.79	2740.01	19.24	8867.37	19608.21
47	46	Ratio	diab	total	3.34	0.49	14.70	2.38	4.31
48	46	Ratio	mdia45	mage45	5.49	1.22	22.16	3.10	7.87
49	46	Ratio	diab45	age45	6.37	1.06	16.62	4.30	8.45
50	47	Total	diab	None	27912.14	2613.21	9.36	22790.26	33034.03
51	47	Total	diab2544	None	4184.67	1139.56	27.23	1951.14	6418.20
52	47	Total	mdia45	None	13566.64	2143.27	15.80	9365.84	17767.45
53	47	Total	fdia45	None	9767.78	1687.05	17.27	6461.16	13074.39
54	47	Ratio	diab	total	3.47	0.32	9.37	2.83	4.10
55	47	Ratio	mdia45	mage45	8.51	1.34	15.80	5.88	11.15
56	47	Ratio	diab45	age45	6.99	0.73	10.49	5.55	8.43
57	48	Total	diab	None	67766.88	7645.48	11.28	52781.74	82752.03
58	48	Total	diab2544	None	10601.49	2957.30	27.90	4805.19	16397.79
59	48	Total	mdia45	None	26503.09	4366.27	16.47	17945.21	35060.97
60	48	Total	fdia45	None	28441.38	4965.30	17.46	18709.39	38173.38
61	48	Ratio	diab	total	2.87	0.32	11.28	2.24	3.51
62	48	Ratio	mdia45	mage45	6.38	1.05	16.47	4.32	8.44
63	48	Ratio	diab45	age45	6.47	0.78	12.12	4.93	8.00
64	59	Total	diab	None	106899.20	10602.43	9.92	86118.43	127679.97
65	59	Total	diab2544	None	15673.03	4272.02	27.26	7299.86	24046.20
66	59	Total	mdia45	None	48283.01	6980.88	14.46	34600.48	61965.54
67	59	Total	fdia45	None	41385.25	6892.28	16.65	27876.38	54894.13
68	59	Ratio	diab	total	3.22	0.32	9.92	2.59	3.84
69	59	Ratio	mdia45	mage45	7.29	1.05	14.46	5.22	9.35
70	59	Ratio	diab45	age45	6.52	0.68	10.44	5.18	7.85

Taken to Run Program

OBS	START	STOP
1	15AUG00:11:48:15	15AUG00:11:54:27

<u>Example 3B: Logistic regression</u>

Bootstrap 500:01 Variance Estimate for
Logistic Regressions
Dependent variable: ccc8dany

OBS	BETA	BHAT	ODDS	WALD	PVALUE	BS_VAR	BS_SD	BS_CV	CIL95	CIU95
1	INTERCEP	-0.52497	0.5916	39.859	0.00000	0.006914	0.08315	15.84	0.50260	0.6963
2	NONPROXY	-0.02097	0.9792	0.107	0.74397	0.004123	0.06421	306.18	0.86345	1.1106
3	AGE1820	0.23044	1.2592	3.648	0.05615	0.014558	0.12066	52.36	0.99397	1.5951
4	AGE2124	0.17965	1.1968	2.284	0.13071	0.014129	0.11887	66.17	0.94806	1.5108
5	AGE2534	0.32840	1.3887	11.673	0.00063	0.009239	0.09612	29.27	1.15028	1.6766
6	AGE3544	0.58256	1.7906	37.691	0.00000	0.009004	0.09489	16.29	1.48672	2.1566
7	AGE4554	1.00008	2.7185	102.878	0.00000	0.009722	0.09860	9.86	2.24079	3.2981
8	AGE5564	1.12843	3.0908	107.564	0.00000	0.011838	0.10880	9.64	2.49722	3.8255
9	AGE6574	1.84230	6.3110	225.048	0.00000	0.015081	0.12281	6.67	4.96095	8.0285
10	AGE7584	2.18306	8.8735	195.654	0.00000	0.024358	0.15607	7.15	6.53495	12.0488
11	AGE85	2.55009	12.8083	75.068	0.00000	0.086628	0.29433	11.54	7.19376	22.8049
12	FEMALE	0.47922	1.6148	102.048	0.00000	0.002250	0.04744	9.90	1.47144	1.7722

Time Taken to Run Program

OBS	START	STOP
1	15AUG00:12:05:02	15AUG00:12:18:47

3.3 Modifications to the program for testing purposes

As mentioned earlier, program run time can be lengthy for regression analyses. It is possible to reduce the number of bootstrap weights used in order to *test* the program (**However, to obtain variance estimations, it is necessary to use the full set of bootstrap weights provided**). For example, in the case of logistic regression, the user can specify the number of times that the regression will be calculated using the bootstrap weights (for example, 2 instead of 500) in the routine “logreg” in section 2 of the program. The user has to add three lines as shown in *italics* in example 4. The value kL corresponds to the number of times that the regression will be calculated (in this case, 2 times). The two other parameters, k and j, should equal respectively 10 and 1 for testing purposes. Please note that those modifications have to be made in conjunction with specifying parameters as shown in example 2B. Again, please note that variance estimates calculated this way will not be valid.

One must also specify the number of bootstrap weights to be read in section 3. The following example shows where the change must be made. The part in ***italics*** must replace the part in parentheses. The same change can be made for the routines “regress” and “regglm”.

Example 4: Changing the number of iterations for testing

```
.
.
.

/********************************************************************************/
/*** Section 2: Declaration of the macros                                    ***/
/********************************************************************************/

.
.
.

*********************************************;

%macro logreg(infile,yvar,xvar,bssz=,multi=);

*********************************************;
proc logistic data=&infile outest=orig(keep=&kp intercep &xvar) descending noprint;
  model &yvar=&xvar;
  &by by region;
  weight fwgt;
run;

proc transpose data=orig out=origest prefix=bhat name=beta;
  var intercep &xvar;
  &by by region;
run;

%let L=%eval(&bssz/10);

%do k=1 %to 10;
  %let j=%eval(1+((&k-1)*&L));
  %let kL=%eval(&k*&L);

%let k=10;
%let j=1;
%let kL=2;

  data poids;
    set &infile;
    keep bsw&j-bsw&kL &yvar &xvar &kp;
  run;

  %do i=&j %to &kL;

    proc logistic data=poids outest=betas(keep=&kp intercep &xvar) noprint descending;
      model &yvar=&xvar;
      &by by region;
      weight bsw&i;
    run;

    proc transpose data=betas out=betat prefix=best name=beta;
      var intercep &xvar;
      &by by region;
    run;

    data betat;
      set betat;
      rename best1=best&i;
    run;

    %if (&i =1) %then %do;

      data bsbeta;
        set betat;
      run;

    %end;
    %else %do;

      data bsbeta;
        merge bsbeta betat;
        &by by region;
      run;

    %end;
  %end;
%end;

data bs_reglg;
  merge origest bsbeta;
  rename bhat1=bhat;
  bs_var=&multi*((&bssz-1)*(var(of best1-best&bssz)))/&bssz;
```

```
  bs_sd=sqrt(bs_var);
  bs_cv=abs(round((bs_sd/bhat1)*100,.01));
  wald=(bhat1/bs_sd)*(bhat1/bs_sd);
  pvalue=1-probchi(wald,1);
  lo95=bhat1-1.96*bs_sd;
  hi95=bhat1+1.96*bs_sd;
  odds=exp(bhat1);
  cil95=exp(lo95);
  ciu95=exp(hi95);
  ydep="&yvar";
  drop best1-best&bssz;
run;

%let printlog=1;
%let dep=&yvar;

%mend logreg;

.
.
.

*********************************************************************;
*  Section 3: MAIN PROGRAM                                          *;
*********************************************************************;

/* Read in Main file */

/*************************************************************************/
/*** DUE TO THE LARGE NUMBER OF OBSERVATIONS, ONLY THE VARIABLES      ***/
/*** THAT ARE NEEDED TO DO THE ANALYSES AND THE VARIABLES             ***/
/*** "REALUKEY" AND "PERSONID" SHOULD BE KEPT AT THIS POINT.          ***/
/*************************************************************************/

   data nphs (index=(id=(realukey personid)));
     set &Mfile (keep= realukey personid ccc8dany nonproxy age1820 age2124 age2534
                       age3544 age4554 age5564 age6574 age7584 age85 female);
   run;

/* Read in Bootstrap Weights */
/* FWGT corresponds to WT58, WT68 or WT68LF, depending on which file is analysed */

   data bsamp (index=(id=(realukey personid)));
     set &bsamp;
     keep fwgt realukey personid bsw1-bsw2; (instead of bsw1-bsw&B.00;)
   run;
.
.
.
```

(The program in Example 2B was tested in this fashion and the execution took about 2 minutes.)

Program BOOTVAR.SAS

```
*                                  WARNING

* The Government of Canada (Statistics Canada) is the owner of all intellectual
* property rights (including copyright) in this software.  Subject to the terms below,
* you are granted a non-exclusive and non-transferable licence to use this software.
*
* This software is provided "as-is", and the owner makes no warranty, either express
* or implied, including but not limited to, warranties of merchantability and fitness
* for any particular purpose.  In no event will the owner be liable for any indirect,
* special, consequential or other similar damages.  This agreement will terminate
* automatically without notice to you if you fail to comply with any term of this
* agreement.;

/*******************************************************************/
/* Date: December 01, 1997                                         */
/* Modified: August 14, 2000                                        */
/*******************************************************************/

/*********************************************************************************/
/***                                                                          ***/
/***                            SAS Program BOOTVAR                           ***/
/***                                                                          ***/
/*** This program calculates variance estimates using the bootstrap weights   ***/
/*** for different types of estimators.  Using SAS Macros, this program can   ***/
/*** calculate variance estimates for totals, ratios and differences between  ***/
/*** ratios.  It can also calculate variance estimates for the parameters     ***/
/*** of a regression, logistic regression, or generalized linear model.       ***/
/*** This program can also be customized for other types of analyses.         ***/
/***                                                                          ***/
/*** The program is divided in 4 sections, described below. Throughout the    ***/
/*** program, it is indicated where the user has to make changes.             ***/
/***                                                                          ***/
/*** Section 1: Declaration of the macro variables                            ***/
/***            This is where variables that are going to be used throughout  ***/
/***            the program have to be defined.  Some changes must be made     ***/
/***            by the user.                                                  ***/
/*** Section 2: Declaration of the macros                                     ***/
/***            This is where the portion of the program to calculate variance ***/
/***            estimates is defined.  No changes have to be made by the user  ***/
/***            in this section (excepted for the macro REGGLM, which has to be **/
/***            customized).                                                  ***/
/*** Section 3: Main program                                                  ***/
/***            This is where the analysis and the bootstrap weights data set  ***/
/***            are read.  The variables to be used in the analyses have to be ***/
/***            defined by the user prior to using this program.              ***/
/*** Section 4: Macro calls                                                   ***/
/***            This is where the user will call the macro(s) that he or she is **/
/***            interested in submitting.  At that point, the user can call    ***/
/***            one macro once, one macro multiple times, or multiple macros.  ***/
/***            Due to the large number of observations to analyse, caution    ***/
/***            should be used however in the number of macro calls requested, ***/
/***            particularly in the case of regressions.                      ***/
/***                                                                          ***/
/*** Before using this program, the user has to prepare an analysis file that  ***/
/*** will contain the variables to analyse.  This file should contain only the ***/
/*** records for which the analysis is required (e.g., if the analysis is done ***/
/*** for the agegroup 12 years old and over, the file should contain only the  ***/
/*** records of those 12 years old and over).  A variable called REGION should ***/
/*** also be created if the analysis is done at a regional level. For example, ***/
/*** if the analysis is done at the provincial level, the variable REGION will ***/
/*** take the value of the variable containing the province on the microdata   ***/
/*** file (e.g. PRC8_CUR). If the analysis is to be done at the health region  ***/
/*** level, REGION will take the value of the health region on each record.    ***/
/***                                                                          ***/
/*** N.B. When calculating a total, a ratio, or a difference of ratios,        ***/
/*** the estimates are obtained by summing the weight of the records that      ***/
/*** have the characteristic of interest.  Hence, a dummy variable must be     ***/
/*** created for each of the variables to be analysed, the variable taking a   ***/
/*** value of 1 when a record has the characteristic of interest and 0 otherwise.*/
/*********************************************************************************/

options ps=48 ls=120 ;
```

```
/*********************************************************************************/
/*** Section 1: Declaration of the macro variables                            ***/
/*********************************************************************************/

%let R=01;                        /* bootstrap averaging parameter (fixed) */

/*****************************************************************************/
/*** USER SPECIFIES THE NUMBER OF BOOTSTRAP WEIGHTS USED, DIVIDED BY 100   ***/
/*** %let B=5; for the bootstrap weights files BD5H35, BD5H356 and BD5LNGF ***/
/*****************************************************************************/

%let B=5;                                /* bootstrap Bx100 times */


/************************************************************************/
/*** USER SPECIFIES THE DIRECTORY CONTAINING THE ANALYSIS FILE (in1)  ***/
/*** AND THE BOOTSTRAP WEIGHTS FILE (in2) SPECIFIED BELOW.            ***/
/*** THE OUTPUT FILES WILL BE SAVED IN THE DIRECTORY SPECIFIED FOR in3 **/
/************************************************************************/

libname in1 "c:\data";
libname in2 "c:\bootstrp";
libname in3 "c:\output";


/*********************************************************************************/
/*** USER SPECIFIES THE NAME OF THE SAS FILE CONTAINING THE ANALYSIS VARIABLES.  */
/*** %let Mfile=xxx;                                                            */
/*********************************************************************************/

%let Mfile=in1.DUMYH356;                          /* Main SAS file name */


/*********************************************************************************/
/*** USER SPECIFIES THE NAME OF THE SAS FILE CONTAINING THE BOOTSTRAP WEIGHTS  ***/
/*** IF THE SAS FILE DOES NOT YET EXIST IT NEEDS TO BE CREATED.               ***/
/*** %let bsamp=bd5h35;   (to be used with the file DUMYH35.TXT)    OR        ***/
/*** %let bsamp=bd5h356;  (to be used with the file DUMYH356.TXT)   OR        ***/
/*** %let bsamp=bd5lngf;  (to be used with the file DUMYLNGF.TXT)             ***/
/*********************************************************************************/

data in2.bd5h356;
  %let datafid="c:\bootstrp\bd5h356.txt";
  %include "c:\bootstrp\b356_i.sas";
run;

%let bsamp=in2.bd5h356;              /* bootstrap weights SAS file */


/*********************************************************************/
/*** USER SPECIFIES THE PARAMETERS by, wo AND kp, WHICH INDICATE ***/
/*** IF THE ANALYSIS IS DONE AT THE REGIONAL LEVEL OR FOR THE    ***/
/*** WHOLE DATASET.                                              ***/
/***                                                             ***/
/*** IF THE ANALYSIS IS DONE FOR THE WHOLE DATASET, THEN         ***/
/***    %let by=*;                                               ***/
/***    %let wo=;                                                ***/
/***    %let kp=;                                                ***/
/***                                                             ***/
/*** ELSE IF THE ANALYSIS IS DONE AT THE REGIONAL LEVEL, THEN    ***/
/***    %let by=;                                                ***/
/***    %let wo=*;                                               ***/
/***    %let kp=region;                                          ***/
/*********************************************************************/

%let by=*;
%let wo=;
%let kp=;

data time;
  format start datetime16.;
  start=datetime();
  output;
run;
```

```
/*********************************************************************************/
/*** Section 2: Declaration of the macros                                    ***/
/*********************************************************************************/

%let printtot=0;
%let printdif=0;
%let printreg=0;
%let printlog=0;

%global dep;

*************************************;

%macro total(infile,var,bssz=,multi=);

*************************************;

proc means data=&infile noprint;
  var fwgt bsw1-bsw&bssz;
  weight &var;
&by  by region;
  output out=ytot
         sum=yhat ybs1-ybs&bssz;
run;

data est;
  set ytot;
  length var1 $ 8;
  length var2 $ 8;
  length type $ 8;
  bs_var=&multi*((&bssz-1)*(var(of ybs1-ybs&bssz)))/&bssz;
  bs_sd=sqrt(bs_var);
  bs_cv=round((bs_sd/yhat)*100,.01);
  cil95=yhat-1.96*bs_sd;
  ciu95=yhat+1.96*bs_sd;
  var1="&var";
  var2="None";
  type="Total";
  drop ybs1-ybs&bssz _type_ _freq_;
run;

proc append data=est base=alltots;
run;

%let printtot=1;

%mend total;


*******************************************;

%macro ratio(infile,var1,var2,bssz=,multi=);

*******************************************;

proc means data=&infile noprint;
  var fwgt bsw1-bsw&bssz;
  weight &var1;
&by  by region;
  output out=ytot
         sum=yhat ybs1-ybs&bssz;
run;

proc means data=&infile noprint;
  var fwgt bsw1-bsw&bssz;
  weight &var2;
&by  by region;
  output out=xtot
         sum=xhat xbs1-xbs&bssz;
run;

data est;
  merge ytot xtot;
  array ybs{&bssz};
  array xbs{&bssz};
  array rbs{&bssz};
  length var1 $ 8;
```

```
  length var2 $ 8;
  length type $ 8;
  yhat=((yhat/xhat)*100);
  do i=1 to &bssz;
    rbs{i}=((ybs{i}/xbs{i})*100);
  end;
  bs_var=&multi*((&bssz-1)*(var(of rbs1-rbs&bssz)))/&bssz;
  bs_sd=sqrt(bs_var);
  bs_cv=round((bs_sd/yhat)*100,.01);
  cil95=yhat-1.96*bs_sd;
  ciu95=yhat+1.96*bs_sd;
  var1="&var1";
  var2="&var2";
  type="Ratio";
  drop ybs1-ybs&bssz xbs1-xbs&bssz rbs1-rbs&bssz xhat i _type_ _freq_;
run;

proc append data=est base=alltots;
run;

%let printtot=1;

%mend ratio;


********************************************************;

%macro diff_rat(infile,var1,var2,var3,var4,bssz=,multi=);

********************************************************;

proc means data=&infile noprint;
  var fwgt bsw1-bsw&bssz;
  weight &var1;
&by  by region;
  output out=ytot
         sum=yhat ybs1-ybs&bssz;
run;

proc means data=&infile noprint;
  var fwgt bsw1-bsw&bssz;
  weight &var2;
&by  by region;
  output out=xtot
         sum=xhat xbs1-xbs&bssz;
run;

proc means data=&infile noprint;
  var fwgt bsw1-bsw&bssz;
  weight &var3;
&by  by region;
  output out=yytot
         sum=yyhat yybs1-yybs&bssz;
run;

proc means data=&infile noprint;
  var fwgt bsw1-bsw&bssz;
  weight &var4;
&by  by region;
  output out=xxtot
         sum=xxhat xxbs1-xxbs&bssz;
run;

data est;
  merge ytot xtot yytot xxtot;
  array ybs{&bssz};
  array xbs{&bssz};
  array yybs{&bssz};
  array xxbs{&bssz};
  array drbs{&bssz};
  length var1 $ 8;
  length var2 $ 8;
  length var3 $ 8;
  length var4 $ 8;
  length type $ 10;
  yhat=(((yhat/xhat)-(yyhat/xxhat))*100);
  do i=1 to &bssz;
```

```
    drbs{i}=(((ybs{i}/xbs{i})-(yybs{i}/xxbs{i}))*100);
  end;
  bs_var=(&multi*((&bssz-1)*(var(of drbs1-drbs&bssz)))/&bssz);
  bs_sd=sqrt(bs_var);
  bs_cv=abs(round((bs_sd/yhat)*100,.01));
  cil95=yhat-1.96*bs_sd;
  ciu95=yhat+1.96*bs_sd;
  var1="&var1";
  var2="&var2";
  var3="&var3";
  var4="&var4";
  type="Diff_Ratio";
  drop ybs1-ybs&bssz xbs1-xbs&bssz yybs1-yybs&bssz xxbs1-xxbs&bssz drbs1-drbs&bssz
       xhat xxhat yyhat i _type_ _freq_;
run;

proc append data=est base=diffrat;
run;

%let printdif=1;

%mend diff_rat;


*********************************************;

%macro regress(infile,yvar,xvar,bssz=,multi=);

*********************************************;

proc reg data=&infile outest=orig(keep=&kp intercep &xvar) noprint;
  model &yvar=&xvar;
  weight fwgt;
&by  by region;
run;

proc transpose data=orig out=origest(drop=_label_) prefix=bhat name=beta;
  var intercep &xvar;
&by  by region;
run;

%let L=%eval(&bssz/10);

%do k=1 %to 10;
  %let j=%eval(1+((&k-1)*&L));
  %let kL=%eval(&k*&L);

  data poids;
    set &infile;
    keep bsw&j-bsw&kL &yvar &xvar &kp;
  run;

  %do i=&j %to &kL;

   proc reg data=poids outest=betas(keep=&kp intercep &xvar) noprint;
     model &yvar=&xvar;
     weight bsw&i;
&by  by region;
   run;

   proc transpose data=betas out=betat prefix=best name=beta;
     var intercep &xvar;
&by  by region;
   run;

   data betat;
     set betat;
     drop _label_;
     rename best1=best&i;
   run;

   %if (&i =1) %then %do;

     data bsbeta;
       set betat;
     run;
```

```
   %end;
   %else %do;

     data bsbeta;
       merge bsbeta betat;
&by    by region;
     run;

   %end;
 %end;
%end;

data est;
  merge origest bsbeta;
  rename bhat1=bhat;
  bs_var=&multi*((&bssz-1)*(var(of best1-best&bssz)))/&bssz;
  bs_sd=sqrt(bs_var);
  bs_cv=abs(round((bs_sd/bhat1)*100,.01));
  cil95=bhat1-1.96*bs_sd;
  ciu95=bhat1+1.96*bs_sd;
  ydep="&yvar";
  drop best1-best&bssz;
run;

proc append data=est base=bs_reg;
run;

%let printreg=1;
%let dep=&yvar;

%mend regress;


*********************************************;

%macro logreg(infile,yvar,xvar,bssz=,multi=);

*********************************************;
proc logistic data=&infile outest=orig(keep=&kp intercep &xvar) descending noprint;
  model &yvar=&xvar;
  &by by region;
  weight fwgt;
run;

proc transpose data=orig out=origest prefix=bhat name=beta;
  var intercep &xvar;
  &by by region;
run;

%let L=%eval(&bssz/10);

%do k=1 %to 10;
  %let j=%eval(1+((&k-1)*&L));
  %let kL=%eval(&k*&L);

  data poids;
    set &infile;
    keep bsw&j-bsw&kL &yvar &xvar &kp;
  run;

  %do i=&j %to &kL;

    proc logistic data=poids outest=betas(keep=&kp intercep &xvar) noprint descending;
      model &yvar=&xvar;
      &by by region;
      weight bsw&i;
    run;

    proc transpose data=betas out=betat prefix=best name=beta;
      var intercep &xvar;
      &by by region;
    run;

    data betat;
      set betat;
      rename best1=best&i;
    run;
```

```
    %if (&i =1) %then %do;

      data bsbeta;
        set betat;
      run;

    %end;
    %else %do;

      data bsbeta;
        merge bsbeta betat;
        &by by region;
      run;

    %end;
  %end;
%end;

data est;
  merge origest bsbeta;
  rename bhat1=bhat;
  bs_var=&multi*((&bssz-1)*(var(of best1-best&bssz)))/&bssz;
  bs_sd=sqrt(bs_var);
  bs_cv=abs(round((bs_sd/bhat1)*100,.01));
  wald=(bhat1/bs_sd)*(bhat1/bs_sd);
  pvalue=1-probchi(wald,1);
  lo95=bhat1-1.96*bs_sd;
  hi95=bhat1+1.96*bs_sd;
  odds=exp(bhat1);
  cil95=exp(lo95);
  ciu95=exp(hi95);
  ydep="&yvar";
  drop best1-best&bssz;
run;

proc append data=est base=bs_reglg;
run;

%let printlog=1;
%let dep=&yvar;

%mend logreg;


********************************************;

%macro regglm(infile,yvar,xvar,bssz=,multi=);

********************************************;
proc glm data=&infile noprint;
     class &xvar;
     model &yvar= &xvar;
     LSMEANS &xvar / out=orig;
     weight fwgt;
&by  by region;
run;

data orig2;
  set orig;
  drop _name_ stderr;
  array meanss{3};
  do i=1 to 3;
    if _n_ = i then do;
      meanss(i)=lsmean;
      retain;
    end;
  end;

  if _n_=3 then do;
    diff1_2= meanss1-meanss2;
    diff1_3= meanss1-meanss3;
    diff2_3= meanss2-meanss3;
    drop lsmean i meanss1- meanss3;
    output;
  end;
run;
```

```
proc transpose data=orig2 out=origest prefix=bhat name=beta;
var diff1_2 diff1_3 diff2_3;
run;

%let L=%eval(&bssz/10);

%do k=1 %to 10;
  %let j=%eval(1+((&k-1)*&L));
  %let kL=%eval(&k*&L);

  data poids;
    set &infile;
    keep bsw&j-bsw&kL &yvar &xvar &kp;
  run;

  %do i=&j %to &kL;

  proc glm data=poids noprint;
    class &xvar;
    model &yvar= &xvar;
    LSMEANS &xvar / out=betas;
    weight bsw&i;
    &by by region;
  run;

  data betas2;
    set betas;
    drop _name_ stderr;
    array meanss{3};
    do i=1 to 3;
      if _n_ = i then do;
        meanss(i)=lsmean;
        retain;
      end;
    end;

    if _n_=3 then do;
      diff1_2= meanss1-meanss2;
      diff1_3= meanss1-meanss3;
      diff2_3= meanss2-meanss3;
      drop lsmean i meanss1- meanss3;
      output;
    end;
  run;

  proc transpose data=betas2 out=betat prefix=best name=beta;
   var diff1_2 diff1_3 diff2_3;
  run;

   data betat;
    set betat;
    rename best1=best&i;
   run;

  %if (&i =1) %then %do;

    data bsbeta;
      set betat;
    run;

  %end;
  %else %do;

     data bsbeta;
       merge bsbeta betat;
       &by by region;
     run;

  %end;
 %end;
%end;

data est;
  merge origest bsbeta;
  rename bhat1=bhat;
  bs_var=&multi*((&bssz-1)*(var(of best1-best&bssz)))/&bssz;
  bs_sd=sqrt(bs_var);
```

```
  bs_cv=abs(round((bs_sd/bhat1)*100,.01));
  cil95=bhat1-2.39406*bs_sd;
  ciu95=bhat1+2.39406*bs_sd;
  drop best1-best&bssz;
run;

proc append data=est base=bs_reg;
run;

%let printreg=1;
%let dep=&yvar;

%mend regglm;


****************;

%macro prnttot;

****************;
%if &printtot=1 %then %do;

&by   proc sort data=alltots;
&by     by region;
&by   run;

/*********************************************/
/* Print the result of total and ratio macro */
/*********************************************/

   proc print data=alltots;
     title "Bootstrap &B.00:&R Variance Estimate for";
     title2 "Totals and Ratios";
     var &kp type var1 var2 yhat bs_sd bs_cv cil95 ciu95;
     format yhat bs_sd cil95 ciu95 11.2;
   run;
%end;

        /*****************************************************/
        /*** Where:                                       ***/
        /*** type         : type of estimator (Total or Ratio)  ***/
        /*** var1 and var2: the variables used to calculate the ***/
        /***                estimate.  For a Total, var2=None   ***/
        /*** yhat         : the estimate (in % for a Ratio)     ***/
        /*** bs_sd        : the standard deviation              ***/
        /*** bs_cv        : the coefficient of variation        ***/
        /*** cil95        : the lower 95% confidence limit      ***/
        /*** ciu95        : the upper 95% confidence limit      ***/
        /*****************************************************/


%mend prnttot;


****************;

%macro prntdiff;

****************;
%if &printdif=1 %then %do;

&by   proc sort data=diffrat;
&by     by region;
&by   run;

/*************************************/
/* Print the result of diff_rat macro */
/*************************************/

   proc print data=diffrat;
     title "Bootstrap &B.00:&R Variance Estimate for";
     title2 "Difference between Ratios";
     var &kp type var1 var2 var3 var4 yhat bs_sd bs_cv cil95 ciu95;
     format yhat bs_sd cil95 ciu95 11.2;
   run;
%end;
```

```
        /*********************************************************/
        /*** Where:                                            ***/
        /*** type         : type of estimator (Diff_Ratio)      ***/
        /*** var1,var2,                                        ***/
        /*** var3 and var4: the variables used to calculate the ***/
        /***                estimate.                          ***/
        /*** yhat         : the estimate (difference in %)      ***/
        /*** bs_sd        : the standard deviation              ***/
        /*** bs_cv        : the coefficient of variation        ***/
        /*** cil95        : the lower 95% confidence limit      ***/
        /*** ciu95        : the upper 95% confidence limit      ***/
        /*********************************************************/

%mend prntdiff;


****************;

%macro prntreg;

****************;
%if &printreg=1 %then %do;

/*************************************************/
/* Print the result of regress and regglm macro */
/*************************************************/

   proc print data=bs_reg;
     title "Bootstrap &B.00:&R Variance Estimate for";
     title2 "Regressions";
     title3 “Dependent variable: &dep”;
     var &kp beta bhat bs_var bs_sd bs_cv cil95 ciu95;
   run;
%end;

        /*********************************************************/
        /*** Where:                                            ***/
        /***   beta       : parameter to estimate               ***/
        /***   bhat       : parameter estimator                 ***/
        /***   bsvar      : variance of the parameter estimator ***/
        /***   bs_sd      : the standard deviation of the para- ***/
        /***                meter estimator                     ***/
        /***   bs_cv      : the coefficient of variation of the ***/
        /***                parameter estimator                 ***/
        /***   cil95      : the lower 95% confidence limit      ***/
        /***   ciu95      : the upper 95% confidence limit      ***/
        /*********************************************************/


%mend prntreg;


****************;

%macro prntlog;

****************;
%if &printlog=1 %then %do;

/************************************/
/* Print the result of logreg macro */
/************************************/

   proc print data=bs_reglg;
     title "Bootstrap &B.00:&R Variance Estimate for";
     title2 "Logistic Regressions";
     title3 “Dependent variable: &dep”;
     var &kp beta bhat odds wald pvalue bs_var bs_sd bs_cv cil95 ciu95;
   run;
%end;
```

```
/*******************************************************/
/*** Where:                                          ***/
/***   beta       : parameter to estimate            ***/
/***   bhat       : parameter estimator              ***/
/***   odds       : odds ratio                       ***/
/***   wald       : Wald statistic                   ***/
/***   pvalue     : p-value of the Wald statistic    ***/
/***   bsvar      : variance of the parameter estimator ***/
/***   bs_sd      : the standard deviation of the para- ***/
/***                meter estimator                  ***/
/***   bs_cv      : the coefficient of variation of the ***/
/***                parameter estimator              ***/
/***   cil95      : the lower 95% confidence limit   ***/
/***                for the odds ratio               ***/
/***   ciu95      : the upper 95% confidence limit   ***/
/***                for the odds ratio               ***/
/*******************************************************/


%mend prntlog;


*******************************************************************;
*  Section 3: MAIN PROGRAM                                        *;
*******************************************************************;

/* Read in Main file */

/*********************************************************************/
/*** DUE TO THE LARGE NUMBER OF OBSERVATIONS, ONLY THE VARIABLES     ***/
/*** THAT ARE NEEDED TO DO THE ANALYSES AND THE VARIABLES            ***/
/*** "REALUKEY" AND "PERSONID" SHOULD BE KEPT AT THIS POINT.         ***/
/*********************************************************************/

   data nphs (index=(id=(realukey personid)));
     set &Mfile (keep= realukey personid + VARIABLES TO ANALYZE);
   run;

/* Read in Bootstrap Weights */
/* FWGT corresponds to WT58, WT68 or WT68LF depending on which file is analysed */

   data bsamp (index=(id=(realukey personid)));
     set &bsamp;
     keep fwgt realukey personid bsw1-bsw&B.00;
   run;


/* Merge Main file and Bootstrap Weights */

&by data bs_nphs (index=(region));
&wo data bs_nphs;
      merge nphs (in=in1) bsamp (in=in2);
      by realukey personid;
      if in1;
    run;

   proc datasets library=work;
     delete nphs bsamp;
   run;
```

```
/******************************************************************************/
/*** Section 4: Macro calls                                                ***/
/******************************************************************************/

/*******************************************************************************/
/*** The variance estimates are calculated using the SAS macros that have been  ***/
/*** defined earlier in the program.  Those SAS macros can be called in the     ***/
/*** program as needed.  A macro call is a SAS statement that specifies the name **/
/*** of the macro called and some parameters.  In this program, the parameters  ***/
/*** indicate which file has to be read, which variables will be used, how many ***/
/*** bootstrap weights are used, and how many bootstrap samples were selected.  ***/
/*** The name of the file to read is always the same (BS_NPHS) and the last two ***/
/*** parameters have already been defined in the first section of the program.  ***/
/*** Hence, the user only has to specify the variables to be used.              ***/
/***                                                                            ***/
/*** Each macro call will result in a variance estimate of ONE estimator.  If   ***/
/*** more than one estimator and its variance estimate have to be calculated,   ***/
/*** the macro has to be called multiple times.                                 ***/
/***                                                                            ***/
/*** A commented statement for the macro call appears where the user's statement***/
/*** should be placed in the program.  The user's statement will                ***/
/*** be identical to the commented one, except for the names of the analysis    ***/
/*** variables.                                                                 ***/
/*******************************************************************************/


/*********/
/*       */
/* Total */
/*       */
/*********/

   proc datasets library=work; /* initial alltots */
     delete alltots;
   run;

/*************************************************************************/
/*** To call the macro total, the following statement is used:         ***/
/*** %total(infile,var,bssz=,multi=);                                  ***/
/***       where infile : bs_nphs                                      ***/
/***             var    : the variable for which a total is calculated ***/
/***             bssz=  : bssz=&B.00                                   ***/
/***             multi= : multi=&R                                     ***/
/***                                                                   ***/
/*************************************************************************/

 *  %total(bs_nphs,VAR,bssz=&B.00,multi=&R);

/* Delete sas files in total macro */

   proc datasets library=work;
     delete ytot est;
   run;


/*********/
/*       */
/* Ratio */
/*       */
/*********/

/***************************************************************************/
/*** To call the macro ratio, the following statement is used:           ***/
/*** %ratio(infile,var1,var2,bssz=,multi=);                              ***/
/***       where infile : bs_nphs                                        ***/
/***             var1   : the variable of the numerator of the ratio     ***/
/***             var2   : the variable of the denominator of the ratio   ***/
/***             bssz=  : bssz=&B.00                                     ***/
/***             multi= : multi=&R                                       ***/
/***                                                                     ***/
/***************************************************************************/


 *  %ratio(bs_nphs,VAR1,VAR2,bssz=&B.00,multi=&R);

/* Delete sas files in ratio macro */
```

```
   proc datasets library=work;
     delete ytot xtot est;
   run;

%prnttot;

/*********************************************/
/*** The results of total and ratio macro ***/
/*** can be saved in a permanent file.    ***/
/*********************************************/

*   data in3.ZZZ;
*     set alltots;
*   run;


/*****************************/
/*                           */
/* Difference between Ratios */
/*                           */
/*****************************/

   proc datasets library=work; /* initial diffrat */
     delete diffrat;
   run;


/********************************************************************************/
/*** To call the macro diff_rat, the following statement is used:           ***/
/*** %diff_rat(infile,var1,var2,var3,var4,bssz=,multi=);                    ***/
/***       where infile : bs_nphs                                           ***/
/***             var1   : the variable of the numerator of the 1st ratio    ***/
/***             var2   : the variable of the denominator of the 1st ratio  ***/
/***             var3   : the variable of the numerator of the 2nd ratio    ***/
/***             var4   : the variable of the denominator of the 2nd ratio  ***/
/***             bssz=  : bssz=&B.00                                        ***/
/***             multi= : multi=&R                                          ***/
/***                                                                        ***/
/*** Note: The confidence interval is calculated for analysis doing only one **/
/*** comparison of ratios.  If multiple comparisons are done, the calculation */
/*** of confidence intervals must take this situation into account.  For    ***/
/*** this reason, in the case of multiple comparisons, the value of the     ***/
/*** normal distribution (Z) used in the calculation of the confidence      ***/
/*** interval must be corrected in the macro declaration in Section 2,      ***/
/*** using, for example, the Bonferroni approach for multiple comparisons.  ***/
/***                                                                        ***/
/********************************************************************************/


 *  %diff_rat(bs_nphs,VAR1,VAR2,VAR3,VAR4,bssz=&B.00,multi=&R);

/* Delete sas files in diff_rat macro */
   proc datasets library=work;
     delete ytot xtot yytot xxtot est;
   run;

%prntdiff;

/*******************************************/
/*** The results of diff_rat macro      ***/
/*** can be saved in a permanent file.  ***/
/*******************************************/

*   data in3.ZZZ;
*     set diffrat;
*   run;


/**************/
/*            */
/* Regression */
/*            */
/**************/

   proc datasets library=work; /* initial bs_reg */
     delete bs_reg;
```

```
   run;

/****************************************************************************/
/*** To call the macro regress, the following statement is used:          ***/
/*** %regress(infile,yvar,xvar,bssz=,multi=);                             ***/
/***       where infile : bs_nphs                                          ***/
/***             yvar   : the dependent variable                           ***/
/***             xvar   : the independent variables.  The variables should ***/
/***                      be listed as they would be in a model statement  ***/
/***                      (no commas in between).                          ***/
/***             bssz=  : bssz=&B.00                                       ***/
/***             multi= : multi=&R                                         ***/
/***                                                                       ***/
/****************************************************************************/


 *  %regress(bs_nphs,YVAR,XVAR,bssz=&B.00,multi=&R);

/* Delete sas files in regress macro */
   proc datasets library=work;
     delete betas betat bsbeta origest;
   run;




/***********************/
/*                     */
/* Logistic Regression */
/*                     */
/***********************/

   proc datasets library=work; /* initial bs_reglg */
     delete bs_reglg;
   run;

/****************************************************************************/
/*** To call the macro logreg, the following statement is used:           ***/
/*** %logreg(infile,yvar,xvar,bssz=,multi=);                              ***/
/***       where infile : bs_nphs                                          ***/
/***             yvar   : the dependent variable                           ***/
/***             xvar   : the independent variables.  The variables should ***/
/***                      be listed as they would be in a model statement  ***/
/***                      (no commas in between).                          ***/
/***             bssz=  : bssz=&B.00                                       ***/
/***             multi= : multi=&R                                         ***/
/***                                                                       ***/
/****************************************************************************/


 *  %logreg(bs_nphs,YVAR,XVAR,bssz=&B.00,multi=&R);

/* Delete sas files in logreg macro */
   proc datasets library=work;
     delete betas betat bsbeta origest;
   run;
```

```
/*********************************/
/*                               */
/* Generalized Linear Model (GLM) */
/*                               */
/*********************************/

/****************************************************************************/
/*** N.B. The macro regglm has to be customized.  The macro definition that ***/
/***      appears in this program is an example of one analysis that has    ***/
/***      been done using this program.                                     ***/
/***                                                                        ***/
/*** To call the macro regglm, the following statement is used:             ***/
/*** %regglm(infile,yvar,xvar,bssz=,multi=);                                ***/
/***       where infile : bs_nphs                                           ***/
/***             yvar   : the dependent variable                            ***/
/***             xvar   : the independent variables.  The variables should  ***/
/***                      be listed as they would be in a model statement   ***/
/***                      (no commas in between).                           ***/
/***             bssz=  : bssz=&B.00                                        ***/
/***             multi= : multi=&R                                          ***/
/***                                                                        ***/
/*** Note: If the regglm macro is adapted to calculate differences between  ***/
/***       means, confidence interval limits should be calculated to take   ***/
/***       into account multiple comparisons.  The value of the normal      ***/
/***       distribution (Z) used in the calculation of the confidence       ***/
/***       interval must be corrected in the macro declaration in Section 2,***/
/***       using, for example, the Bonferroni approach for multiple         ***/
/***       comparisons.                                                     ***/
/****************************************************************************/


 *  %regglm(bs_nphs,YVAR,XVAR,bssz=&B.00,multi=&R);

/* Delete sas files in regglm macro */
   proc datasets library=work;
     delete betas betat bsbeta origest;
   run;

%prntreg;

/************************************************/
/*** The results of regress and regglm macro ***/
/*** can be saved in a permanent file.       ***/
/************************************************/

*   data in3.ZZZ;
*     set bs_reg;
*   run;


%prntlog;

/*******************************************/
/*** The results of logreg macro        ***/
/*** can be saved in a permanent file.  ***/
/*******************************************/

*   data in3.ZZZ;
*     set bs_reglg;
*   run;


data time;
  set time;
  format stop datetime16.;
  stop=datetime();
  output;
run;

proc print data=time;
  title 'Time Taken to Run Program';
run;

/* End of SAS program BootVar */
```