Bootstrapping Made Easy: A Stata ADO File

Emmanuelle Piérard*, Neil Buckley**, and James Chowhan***

McMaster Research Data Centre
Statistics Canada

McMaster University
1280 Main Street West
Hamilton, ON, L8S 4L6

* Department of Economics
Kenneth Taylor Hall
Phone: (905) 525-9140 x.27767
Email: pieraref@mcmaster.ca

**Department of Economics
Kenneth Taylor Hall
Phone: (905) 525-9140 x.23211
Email: nbuckley@mcmaster.ca

*** McMaster Research Data Centre
Room 217 Mills Library Memorial Library
Phone: (905) 525-9140 x.27967
Email: chowhan@mcmaster.ca

October 17, 2003
--

Abstract

This note introduces a Stata command that calculates variance estimates using bootstrap weights. The “bswreg” command is compatible with a wide variety of regression analytical techniques and datasets. This program has been tested and compared against the regression analytical techniques available in bootvare_v20.sas to verify accuracy. NPHS Cycle 4 data are used for these comparisons. This program provides researchers with an easy and flexible tool that was not previously available.

Introduction

This note introduces a Stata program that calculates variance estimates using bootstrap weights. The main motivation for creating this program was to develop an easy to use and flexible tool within Stata that can be employed with bootstrap weights that are made available with most of Statistic Canada's micro-data sets. The use of these bootstrap weights allows researchers to make use of complex survey design information and calculate reliable variance estimates, while preserving the confidentiality of respondents [Yeo et al., 1999].

The program is compatible with a wide variety of regression techniques. This program builds on the linear and logistic regressions that were introduced in "Bootvar" to include a variety of regression techniques.[1] This note discusses the programs unique features, presents the strengths and weaknesses of the program, and describes a simple test used to verify the accuracy of this new Stata program relative to BOOTVARE_V20.SAS.

II. Standard Bootstrap

Most of Statistics Canada's surveys use a complex design to draw a representative sample from the population of interest. The resulting micro-data sets are available with bootstrap weights that can be used to account for the complex survey design. The use of these bootstrap weights allows researchers to calculate reliable variance estimates. The bootstrap variance estimator for $\hat{\theta}$, used in this program, is given by [Yeo et al., 1999; 3]:

$$v_B\left(\hat{\theta}\right) = \frac{1}{B}\sum_b \left(\hat{\theta}^*_{(b)} - \hat{\theta}^*_{(.)}\right)^2 \qquad (1)$$

where $\hat{\theta}^*_{(.)} = \left(1/B\right)\sum_b \hat{\theta}^*_{(b)}$

III. How To

The Stata program is easy to use by simply copying the "bswreg.ado" and "bswreg.hlp" files, which are described in Appendix I, to your Stata ADO folder[2], then employ the program by using the following syntax command:

bswreg depvar [varlist] weighttype=full_sample_weight [if exp] [in range],
cmd(STATA_regression_command) [cmdops(options_for_regression_command)]
bsweights(bootstrap_weights_varlist) [level(integer)] [bsci]
[saving(path_and_filename[,replace])];

---

[1] "The Bootvar program is available in both SAS and SPSS formats. It is made up of a macro that computes variances for totals, ratios, and differences between ratios, and for linear and logistic regression. The Bootvar program is provided with bootstrap weights and a document explaining how to modify and use the program to suit user's needs." [Statistics Canada, 2002; 39]

[2] Type the command "adopath" at the Stata command prompt for a list of ado directory paths in which to place this program. Further, the researcher will need to "set" the "matsize" and "memory" size to levels appropriate to the computer and dataset that they are using.

For example, suppose a researcher wishes to run an ordinary least squares regression of height on a list of provincial dummies, education dummies, age, and gender using the National Population Health Survey (NPHS) Cycle 4[3]; they want to save the bootstrap output table as a data file in memory (including bootstrap coefficients, standard errors and other inference statistics); and they choose to use all of the 500 available bootstrap weights. The command code line would be as follows:[4]

bswreg height nfld pei ns nb qc on man sask alb lshs someps ugrad agesq age gender [pw=wt60lf], cmd(reg) bsweights(bsw1-bsw500) level(95) bsci saving(c:\temp\bswdata.dta, replace); (2)

This command assumes that the appropriate bootstrap weights and the data-file have already been merged accurately by the appropriate unique identifier (in this example, NPHS Cycle 4, where the unique identifiers are realukey and personid). This program does not require the bootstrap weights to have any naming scheme. Further, the bswreg command allows for the use of options. The program has several options available:

cmd: specifies the Stata regression command to bootstrap. This is a required option. The following regression commands have been tested explicitly: regress, logit, probit, tobit, ologit, oprobit, biprobit, mlogit, qreg, glm, intreg, boxcox, (basically any single stage estimation technique should work with this program) and non-twostage "xt" commands that support weights.

bsweights: specifies a variable list of the bootstrap weight names. This is a required option. For instance, if your bootstrap weights are named bsw1 to bsw500, you could specify the option as bsweights(bsw1-bsw500).

cmdops: specifies the options you wish to use on the Stata regression command provided in cmd(). Some options are useful and others are meaningless in a bootstrap weighting context. For instance, if you wish to run the REGRESS command with no constant then use the cmd(regress) cmdops(noconstant) options. Options like robust are meaningless in this context since the command computes bootstrap weighted standard errors not robust ones.

level: specifies the confidence level, in percent, for confidence intervals. The default is level(95).

[3] The NPHS Cycle 4 (2000-2001) Longitudinal Master File sample is reduced from 17,276 to 12,439 by only including respondents in cycles 1 to 4 and records without missing observations. The regression's dependent variable is height, this variable is a scale that standardizes the metric and imperial systems. For height, a value of 50 on the scale is equivalent to 5'0" (60 inches) (151.1 to 153.6 cm), and an increase of one in the scale is equivalent to an inch. The provincial dummy variables include: nfld pei ns nb qc on man sask alb, where bc is the omitted province. The education variables include: less than high school (lshs), high school graduates (hsgrad--omitted), some post-secondary (someps), and university graduates (ugrad). The age variables include age and age-squared. The gender variable is equal to 1 if male and 2 if female.

[4] The results from this regress are presented in Appendix II.

bsci: specifies that the confidence intervals be calculated from the raw bootstrapped distribution of coefficients rather than using the standard formula based on the bootstrapped standard error and the normal distribution.[5]

saving: saves the bootstrap statistics in a separate Stata dataset file that can later be loaded and used by other .DO and .ADO files. If you do not specify an extension, .dta will be assumed. Include the replace option to overwrite an existing file.

IV. Unique Features

The bswreg command provides researchers with a flexible tool that was not previously available. Reliable variance estimates can be generated to accompany analytical techniques from ordinary least squares, probits, quantile regressions to random-effects tobit models when used congruently with the bswreg command to produce design-based variance estimates.

The command has a "built-in" help feature that provides basic assistance on predefined search topics. By typing "help bswreg" at the command prompt, the researcher will display a description of the procedure, a list of outputted variables, and a list of example code that employ the bswreg command.

Due to the breadth of the analytical techniques that are compatible with the bswreg command, a semi-sophisticated error-resolving algorithm was required. Specifically, for estimation techniques that require the model to iterate toward convergence, there is the possibility that the model will not converge for every set of bootstrap weights selected; therefore the command bswreg has been designed to default to the actual number of successful bootstrap procedures. Thus, errors such as convergence errors are avoided.[6]

This program is also designed to deal with bootstrap regressions that fail. There are two main examples where this could take place. First, due to a zero bootstrap weight corresponding perfectly with a small sample size on a discrete variable, when these two cases are combined, the result is that the variable is dropped. Once a variable is dropped the regression output from this particular bootstrap sample will not have an identical number of variables as all other "successful" bootstrap samples. This is problematic for the calculation of the variance estimates, and as a result these bootstrap samples are dropped. Second, in a case where the sample would be very small, it is possible that some of the bootstrap samples weight a majority of the sampled observations with a weight of zero. The resulting sample could be too small to perform the estimation procedure, and again this sample would be removed from the bootstrapping

[5] This option is provided for users that may have a theoretical reason for employing the confidence intervals derived from the bootstrapped distribution of coefficients.

[6] For example, suppose 500 bootstrap sample weights have been selected to run an iterative procedure using maximum likelihood (random-effects or population-averaged logit models) and x regressions fail to converge due to the nature of x bootstrap sample weights, then 500-x bootstrap sample weighted regressions were successful, and are used to generate the bootstrap variance estimator. BSWREG output provides a count of the number of successful iterations completed.

procedure. This results in the total number of bootstrap regressions reported at the end of the program being smaller than the original number specified in the bswreg command.

The output generated at the end of the "bswreg" procedure includes the total bootstraps completed, name of variable, coefficient estimate, and the bootstrap standard error of coefficient, z-stat, p-value, lower and upper 95% confidence intervals (by default) assuming a normal distribution, and the option "bsci" produces lower and upper 95% confidence intervals that are generated using the raw bootstrap distribution itself. In terms of the output generated, the main distinguishing feature of this program relative to other programs, is that it has the option to produce two sets of confidence intervals: the first assumes a normal distribution and the second uses the raw bootstrap-distribution.

Any command that requests multiple regressions by a variable that parses the data will work with bswreg command. For example, the "by var_name:" prefix to a regression command will work in the bswreg framework. Continuing the previous example, suppose the researcher wanted to run OLS by gender separately, wanted to only use the bootstrap weights bsw100 to bsw400, a confidence interval of 99%, and the bootstrap distribution, the syntax would be as follows:

```
bysort gender: bswreg height  nfld pei ns nb qc on man sask alb
lshs someps ugrad agesq age [pw=wt60lf], cmd(reg)                    (3)
bsw(bsw100-bsw400)  level(99) bsci;
```

The bswreg command focuses on reliable variance estimation and related statistics and does not calculate any ancillary statistics. Ancillary statistics are not calculated due to the potential breadth of estimates that could be considered given the scope of the compatibility of this program. For example, odds ratios for the logit regression or the change of probability for dummy variables for the probit procedures are omitted.

The program can be used to calculate various summary statistics such as frequencies, means, and ratios. See Appendix III for examples of how these statistics can be calculated.

A note with regards to testing hypotheses: the bswreg program stores the coefficients in the standard e(b) matrix and the bootstrapped variance-covariance matrix in e(V). This means that the "test" command can be used directly after "bswreg" to conduct tests. However, all tests will use the asymptotic normal distribution instead of the t-distribution. Thus, all tests run with the "test" command will be WALD style tests based on the chi squared distribution. The program prevents researchers from running F-tests, which in any event would be incorrect due to the asymptotic nature of the bootstrapping technique.

V. Accuracy Tests

This program has been tested to verify its accuracy. The program bootvare_v20.sas was used as the benchmark for all tests and comparisons. The ordinary least squares regression procedure in bootvar was duplicated using the bswreg command. Since bootvar has been

specifically designed for the NPHS, a longitudinal dataset including Cycle 4 were employed in the tests. Further, in all tests, all 500 bootstrap weights were used for each procedure. See Appendix II for the results. In addition, frequency, mean, and ratio summary statistics were duplicated using a comparable regression framework--see Appendix III.

The results generated by the two programs are identical for the Ordinary Least Squares example. Further, all of the variance related estimates, such as the standard error, z-stat, p-value, and lower and upper bounds (assuming a normal distribution), were identical. These results indicate that this program has a substantial degree of reliability.

VI. Concluding Remarks

This program focuses on reliable variance estimation across a wide range of analytical techniques. The Stata program discussed in this note can be used to calculate variance estimates using bootstrap weights across a wide spectrum of datasets (weights are currently available for NLSCY, NPHS, SLID, WES, and YITS)[7]. The accuracy of this program has been verified through comparability tests against other available analytical programs. Researchers who use Stata now have available for their use a flexible tool that is easy to use and accurate.

References

Statistics Canada. 2002. "Population Health Surveys Program: National Population Health Survey, Cycle 4 (2000 – 2001), Household Component Longitudinal Documentation." Health Statistics Division. Ottawa.

Yeo, Douglas, Harold Mantel, and Tzen-Ping Liu. 1999. "Bootstrap Variance Estimation For the National Population Health Survey." American Statistical Association: Proceedings of the Survey Research Methods Section. Baltimore, August.

[7] The following surveys are core datasets that are available through the Statistics Canada Research Data Centre program: National Longitudinal Survey of Children and Youth (NLSCY), National Population Health Survey (NPHS), Survey of Labour and Income Dynamics (SLID), Workplace and Employee Survey (WES), and Youth in Transition Survey (YITS).

Appendix I

Ado File:

```
*                                   WARNING

* The authors are the owners of all intellectual
* property rights (including copyright) in this software.  Subject to the terms below,
* you are granted a non-exclusive and non-transferable license 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.

* TO CHANGE THE DECIMAL DISPLAY FORMAT OF THE BOOTSTRAPPED OUTPUT SEARCH FOR THE "FORMAT" COMMAND
NEAR THE BOTTOM OF THIS PROGRAM;

program define bswreg, eclass sortpreserve byable(recall)
* August 8th, 2003 Pierard, Buckley, Chowhan

# delimit;
version 7.0;

syntax varlist(numeric) [aweight pweight fweight iweight] [if] [in], cmd(string) [cmdops(string)]
  BSWeights(varlist numeric) [Level(integer 95)] [BSci] [SAVing(string)];
*This sets the touse variable = 1 if observation is in our sample;
marksample touse;
*Error check to make sure a weight was used;
if "`weight'"==""
  {;
  noi di in red "BSWREG error: You must specify a weight!";
  exit;
  };

quietly
{;
*Preserve the original dataset and set parameter values and setup temporary matrices;
preserve;
set more 1;
tempvar esamplevar;
tempname bhat bsVC bsbhat bsbetas;

*The next line runs the wanted regression and checks for errors;
capture `cmd' `varlist' [`weight'`exp'] if `touse' `in', `cmdops';
if _rc ~= 0
   {;
   noi di in red " ";
   noi di in red "Error doing: `cmd' `varlist' [`weight'`exp'] `if' `in', `cmdops'";
   noi di in red " ";
   noi di in red "The regression command you have typed in resulted in an error, please
investigate";
   noi di in red "this error outside of the 'bswreg' program by typing in the regression command
itself";
   noi di in red "with the options you specified.";
   noi di in red " ";
   exit;
   };
*The next line runs the wanted regression and we store the coefficients in a matrix for later
use;
`cmd' `varlist' [`weight'`exp'] if `touse' `in', `cmdops';
gen `esamplevar'=e(sample);
*e(b) is a 1x(k+1) coefficient vector if the model has a constant and k is the number of
variables other than the constant;
matrix `bhat'=e(b);
matrix `bsVC'=e(V);
```

```
*we store the variable names of the regressors and the number of regressors in local macros;
local _varnames : colfullnames(`bhat');
local _k=colsof(`bhat')-1;
local _k1=`_k'+1;
*Generate concatenated list of placeholder regressor variable names xc1-xck1, later to be turned
into variables;
local _xclist="";
forvalues _i = 1/`_k1'
   {;
   local _xclist `_xclist' _xc`_i';
   };
*We assigned these placeholder variable names to the regressors in the coefficient vector;
matrix colnames `bhat' = `_xclist';
*Each "true estimate of beta" is saved under it's own variable name;
svmat double `bhat', name(col);
matrix colnames `bhat' = `_varnames';

*Realboot is the actual number of successful bootstrap regressions run in case we get any
convergence/regression errors etc., it starts off at the specified number of bootstrap weights;
local _realboot: word count `bsweights';
noi di " ";

*The main bootstrap loop will run with each bootstrap weight in the supplied bsweight varlist and
exit with the matrix named BETAS containing all the bootstraps of our coefficients, a
(boot)x(k+1) dimensional matrix;
local _i 1;
*Start of bootstrap loop;
foreach bswvar of local bsweights
 {;
 *Display notice of number of completed bootstraps every time 50 are completed;
 if mod(`_i',50)==0
  {;
  noi di `_i' " bootstraps completed";
  };
 *Run the regression with the chosen set of bootstrap weights, only use the coefficients if there
are no errors;
 capture `cmd' `varlist' [`weight'=`bswvar'] if `touse' `in', `cmdops';
 if _rc==0
  {;
   *Store coefficients in the bootstrap matrix;
   matrix `bsbhat'=get(_b);
   *bsbhat is a 1x(`k'+1) (row) vector if the model has a constant.  Need to transpose;
   matrix `bsbhat'=`bsbhat'';
   *If we have the proper number of coefficients then add them to the bootstrap matrix, otherwise
do not add them (this most likely arises due to a regressor being dropped due to
multicollinearity;
   if rowsof(`bsbhat')==`_k1'
        {;
        *If we are on the first bootstrap then create the bsbetas matrix, otherwise append to it;
        if `_i'==1
           {;
           matrix `bsbetas'=(`bsbhat');
           };
        else
           {;
           matrix `bsbetas'=(`bsbetas',`bsbhat');
           };
        };
   else
        {;
        matrix drop `bsbhat';
        local _realboot=`_realboot'-1;
        noi di "Bootstrap #`_i' has been dropped for not having the correct number of
coefficients";
        };
  };
 else
  {;
  local _realboot=`_realboot'-1;
  noi di "bootstrap #`_i' has been dropped due to an error estimating the regression";
  };
```

```
 local _i=`_i'+1;
 };
*End of bootstrap loop;

*All the bootstraps have been completed now calculate the new standard errors and display
relevant statistics;
*We must transpose the matrix to make each row now, then column, a new variable;
matrix `bsbetas'=`bsbetas'';
*Generate concatenated list of colnames, later to be turned into variables;
local _xvlist="";
forvalues _i = 1/`_k1'
   {;
   local _xvlist `_xvlist' _xv`_i';
   };
*Calls each row of the matrix by the name of the independent variable it corresponds to (we call
them _xv'_i' so that they are not mixed up with the "real" variables);
matrix colnames `bsbetas'= `_xvlist';
*Separate each column as a new variable.  The format of the data must be specified.  It renames
each variable by the name of the column;
svmat double `bsbetas', name(col);

*Generate the bootstrapped variance-covariance matrix, you can access this in e(V) after running
the BSWREG ado file;
forvalues _i = 1/`_k1'
 {;
 forvalues _j = 1/`_k1'
  {;
    correlate _xv`_i' _xv`_j', covariance;
    matrix `bsVC'[`_i',`_j'] = ((`_realboot'-1)/`_realboot')*r(cov_12);
  };
 };

*Generate the standard deviation, t-stat, conf. int. etc. for each variable;
tempvar _bsobs _uniqobs _coefnum;
gen `_bsobs'=_n;
forvalues _i = 1/`_k1'
 {;
 sum _xv`_i';
 * Like the SAS bootvar program, we use (boot-1)/boot because variance and standard error have
different denominators;
 gen _sdx`_i'=sqrt(((`_realboot'-1)/`_realboot')*r(Var)) in 1/1;
 gen _t`_i'=_xc`_i'/_sdx`_i' in 1/1;
 gen _abst`_i'=abs(_t`_i') in 1/1;
 gen _p`_i'=2*norm((-1)*_abst`_i') in 1/1;
if "`bsci'"==""
 {;
   gen _low`level'`_i'=_xc`_i'-invnorm(1-((1-(`level'/100))/2))*_sdx`_i';
   gen _high`level'`_i'=_xc`_i'+invnorm(1-((1-(`level'/100))/2))*_sdx`_i';
 };
 if "`bsci'"=="bsci"
   {;
   sort _xv`_i';
   local _obslow= max(1,round(((1-(`level'/100))/2)*`_realboot',1));
   local _obshigh= max(1,round((1-((1-(`level'/100))/2))*`_realboot',1));
   local _obslow2= _xv`_i'[`_obslow'];
   local _obshigh2= _xv`_i'[`_obshigh'];
   sort `_bsobs';
   gen _low`level'`_i'= `_obslow2' in 1/1;
   gen _high`level'`_i'= `_obshigh2' in 1/1;
   };
 };

*Assign each coefficient its true regressor name stored at the beginning of this program;
local _i=1;
foreach _curname in `_varnames'
 {;
 gen str10 _xname`_i'="`_curname'";
 local _i=`_i'+1;
 };
```

```
*Reshape the data so that the bootstrapped stats can be displayed easily, and then display the
results;
keep _xname* _xc* _sdx* _t* _p* _low`level'* _high`level'*;
drop if _n>1;
gen `_uniqobs'=1;

reshape long _xname _xc _sdx _t _p _low`level' _high`level', i(`_uniqobs') j(`_coefnum');
*The %9.4f tells stata to display the bootstrapped results to 6 decimals using 15 numbers total -
- this can be changed to suit tastes;
format _xc _sdx _t _p _low`level' _high`level' %11.6f;

*creates nice labels for variables
label var _xname "Name of variable";
ren _xname Var_name;
label var _xc "Coefficient estimate";
ren _xc Coef;
label var _sdx "Bootstrap standard error of coefficient";
ren _sdx BS_se;
label var _t "Bootstrap z-statistic";
ren _t BS_zstat;
label var _p "Bootstrap p-value";
ren _p BS_pvalue;
if "`bsci'"==""
 {;
 label var _low`level' "Bootstrap lower 95% confidence interval assuming a normal distribution";
 label var _high`level' "Bootstrap upper 95% confidence interval assuming a normal distribution";
 };
if "`bsci'"=="bsci"
 {;
 label var _low`level' "Bootstrap lower 95% confidence interval using bootstrap sample
distribution";
 label var _high`level' "Bootstrap upper 95% confidence interval using bootstrap sample
distribution";
 };
ren _low`level' BS_cilow`level';
ren _high`level' BS_ciup`level';

*Display RESULTS!;
noi display in green "Results from BSWREG";
noi display in green "--------------------";
noi display in green " ";
if "`bsci'"=="bsci"
 {;
 noi display in green "* The confidence intervals below are based on the bootstrapped
distribution";
 };
else noi display in green "* The confidence intervals below are based on the normal
distribution";
noi display in green " ";
noi list Var_name Coef BS_se BS_zstat BS_pvalue BS_cilow`level' BS_ciup`level', nodisplay noobs;
noi di " ";
noi di "Total bootstraps completed: `_realboot'";

*Set the eclass variables like the coefficients and the variance-covariance matrix into their
appropriate matrices so that F-tests and the like can be run;
*If you wish the TEST command to produce F-tests after the BSWREG command then add ",
dof(`_realboot')" to the line below;
estimates post `bhat' `bsVC';

*Save the bootstrap raw data is the "SAVING" option has been used;
if "`saving'"~=""
 {;
  drop _*;
  save "`saving'", `replace';
 };

*Restore the original dataset
restore;

};
end;
```

BSWREG Help File

```
{smcl}
{* 8August2003 Pierard/Buckley/Chowhan}
{hline}
help for {hi:BSWREG}
{hline}
{title:BSWREG - uses bootstrap weights to calculate standard errors in models involving complex
survey data.}

{p 8 13}{cmd:bswreg} depvar [varlist] {it:weighttype}={it:full_sample_weight} [{cmd:if} {it:exp}]
[{cmd:in} {it:range}]{cmd:,} {cmd:cmd(}{it:STATA_regression_command}{cmd:)}
[{cmd:cmdops(}{it:options_for_regression_command}{cmd:)}]
  {cmdab:bsw:eights(}{it:bootstrap_weights_varlist}{cmd:)} [{cmd:level(}{it:integer}{cmd:)}]
[{cmd:bsci}] [{cmdab:sav:ing(}{it:path_and_filename}[{cmd:,replace}]{cmd:)}];

{p} {cmd:cmd()} and {cmd:bsweights()} are required options for the {cmd:BSWREG} command.
{p} {cmd:by ...: } and {cmd:bysort ...:} can be used with {cmd:BSWREG}. See help {help by}.
{p} {cmd:aweight}s, {cmd:fweight}s, {cmd:iweight}s, and {cmd:pweight}s are allowed as long as the
      given regression command is compatible with them. See help {help weights}.
{p} As {cmd:BSWREG} is an eclass STATA program, it provides STATA with the {cmd:e(b)} coefficient
vector and the {cmd:e(V)} bootstrapped variance-covariance matrix.
    The {cmd:test} command can be used immediately following the {cmd:BSWREG} command to conduct
Wald tests based on the chi-squared distribution.

{inp:The software is provided "as-is" and the authors are not responsible for any misuse.}
{title:Description}
(used to calculate regression statistics using Statistics Canada's bootstrap weights)

{p}{cmd:bswreg} runs a number of regressions, each with a particular bootstrap
weight so that bootstrapped standard errors on the coefficients can be calculated
and displayed. Use of bootstrap weights is recommended for calculating reliable
standard errors, confidence intervals etc. on data from complex
household surveys.

The user provides the names of the bootstrap weights to the {cmd:BSWREG} command
in the {cmdab:bsw:eights(varlist)} option. You must already have the appropriate
bootstrap weights merged into your datafile for this command file to work.  NPHS
merges on REALUKEY and SLID merges on PERSONID. Below is a sample .DO file that
merges NPHS bootstrap weights into a datafile named data.dta:

{inp:use data.dta, replace"}
{inp:sort realukey"}
{inp:save data.dta, replace"}
{inp:use bootstrap/sas_bs_wt_1_4.dta, replace"}
{inp:destring realukey, replace"}
{inp:sort realukey"}
{inp:merge realukey using data.dta"}
{inp:keep if _merge==3"}

{title:Options}

{p 0 4}{cmd:cmd(}{it:STATA_regression_command}{cmd:)} specifies the Stata regression command to
bootstrap. This is a {cmd:required} option. "regress", "probit" and "logit" are a few
possibilities.

{p 0 4}{cmd:bsweights(}{it:varlist}{cmd:)} specifies a variable list of the bootstrap weight
names. This is a {cmd:required} option. For instance, if your bootstrap weights are named bsw1 to
bsw500, you may wish to use the
{cmd:bsweights(bsw1-bsw500)} option.

{p 0 4}{cmd:cmdops(}{it:options_for_regression_command}{cmd:)} specifies the options you wish to
use on the Stata regression command provided in {cmd:cmd()}. Some options are useful and others
are meaningless in a bootstrap weighting context.
For instance, if you wish to run the REGRESS command with no constant then use the
{cmd:cmd(regress) cmdops(noconstant)} options.  Options like {cmd:robust} are meaningless in this
context since the command computes bootstrap weighted
standard errors not robust ones.

{p 0 4}{cmd:level(}{it:integer}{cmd:)} specifies the confidence level, in percent,
```

for confidence intervals. The default is {cmd:level(95)}. See help {help level}.

{p 0 4}{cmd:bsci} specifies that the confidence intervals be calculated from the raw bootstrapped distribution of coefficients rather than using the standard formula
based on the bootstrapped standard error and the normal distribution.

{p 0 4}{cmd:saving(}{it:filename}[{cmd:,replace}]{cmd:)} saves the bootstrap statistics in a separate Stata dataset file that can later be loaded and used by other .DO and .ADO files. If you do not specify an extension, {cmd:.dta} will
be assumed. Include the {cmd:,replace} option to overwrite an existing file.

{title:Outputed variables}

{inp: Var_name:} This is the STATA variable name of the regressor.
{inp: Coef:}   This is the coefficient from the specified regression.
{inp: BS_se:}   This is the new standard error of the coefficient,
            calculated using bootstrap weights.
{inp: BS_zstat:} This is the new z-stat of the coefficient,
            calculated as the coefficient divided by the bootstrapped standard error.
{inp: BS_pvalue:} This is the new p-value of the coefficient,
            calculated using the z-statistic.
{inp: BS_cilow95n:} This is the lower (level)% confidence interval around the coefficient
            using the bootstrapped std. error.
{inp: BS_ciup95n:} This is the upper (level)% confidence interval around the coefficient
            using the bootstrapped std. error.

{title:Examples}

{p 8 12}{inp:. bswreg income education rural [aw=wt] if married==1, cmd(regress) bsw(bsw1-bsw500)}

{p 8 12}{inp:. bswreg employed education rural [aw=wt66], cmd(probit) bsw(bsw50-bsw100)}

{p 8 12}{inp:. bysort maritalstatus: bswreg income education rural [aw=wt], cmd(reg) bsw(bsw1-bsw500)}
  {inp:cmdops(noconstant) level(99) bsci saving(c:\data\bsw1.dta,replace)}

Appendix II

BSWREG Results:

```
. reg height  nfld pei ns nb qc  on man sask alb lshs  someps ugrad agesq age gender [pw=wt60lf];
(sum of wgt is   2.6597e+07)

Regression with robust standard errors                Number of obs =   12439
                                                      F( 15, 12423) =  279.22
                                                      Prob > F      =  0.0000
                                                      R-squared     =  0.4753
                                                      Root MSE      =   4.208
------------------------------------------------------------------------------
             |               Robust
      height |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        nfld |  -.2922724   .2449279    -1.19   0.233    -.7723689    .1878242
         pei |  -.3285292   .2671396    -1.23   0.219    -.8521642    .1951058
          ns |  -.5414279   .2420961    -2.24   0.025    -1.015974   -.0668821
          nb |  -.5146936   .2386633    -2.16   0.031    -.9825106   -.0468766
          qc |  -.7923805   .1886131    -4.20   0.000    -1.162091   -.4226696
          on |  -.4070276   .1911941    -2.13   0.033    -.7817977   -.0322575
         man |   .2473519   .2449153     1.01   0.313      -.23272    .7274238
        sask |   .1010543   .2473725     0.41   0.683    -.3838343    .5859428
         alb |   .2328238   .2169931     1.07   0.283    -.1925163    .6581638
        lshs |   3.759076   .1975814    19.03   0.000     3.371786    4.146366
      someps |   3.506946   .1773476    19.77   0.000     3.159318    3.854575
       ugrad |   3.155798   .1650738    19.12   0.000     2.832228    3.479368
       agesq |  -.0047157   .0001675   -28.16   0.000     -.005044   -.0043874
         age |    .465882   .0161484    28.85   0.000     .4342285    .4975354
      gender |  -4.695284   .1128784   -41.60   0.000    -4.916543   -4.474025
       _cons |   50.90797    .503058   101.20   0.000      49.9219    51.89404
------------------------------------------------------------------------------

. bswreg height  nfld pei ns nb qc  on man sask alb lshs  someps ugrad agesq age gender
[pw=wt60lf], cmd(reg) bsw(bsw1-bsw500)  level(95) bsci saving(c:\temp\bswdata.dta, replace);

50 bootstraps completed
100 bootstraps completed
150 bootstraps completed
200 bootstraps completed
250 bootstraps completed
300 bootstraps completed
350 bootstraps completed
400 bootstraps completed
450 bootstraps completed
500 bootstraps completed
Results from BSWREG
-------------------

* The confidence intervals below are based on the bootstrapped distribution

  Var_name         Coef        BS_se     BS_zstat    BS_pvalue   BS_cilow95    BS_ciup95
      nfld    -0.292272     0.211569    -1.381455     0.167139    -0.708804     0.145642
       pei    -0.328529     0.241579    -1.359923     0.173854    -0.780092     0.131531
        ns    -0.541428     0.206423    -2.622899     0.008719    -0.968447    -0.145737
        nb    -0.514694     0.231676    -2.221613     0.026309    -1.042424    -0.094749
        qc    -0.792381     0.155897    -5.082725     0.000000    -1.079814    -0.477167
        on    -0.407028     0.161284    -2.523667     0.011614    -0.711807    -0.081808
       man     0.247352     0.203860     1.213342     0.224999    -0.162946     0.623552
      sask     0.101054     0.218886     0.461676     0.644314    -0.360283     0.520983
       alb     0.232824     0.193489     1.203289     0.228864    -0.173770     0.640249
      lshs     3.759076     0.171806    21.879725     0.000000     3.417291     4.105172
    someps     3.506946     0.173803    20.177696     0.000000     3.194609     3.866022
     ugrad     3.155798     0.158046    19.967600     0.000000     2.859345     3.466744
     agesq    -0.004716     0.000158   -29.844374     0.000000    -0.005035    -0.004413
       age     0.465882     0.015283    30.483185     0.000000     0.437043     0.496086
    gender    -4.695284     0.095502   -49.164188     0.000000    -4.879501    -4.502970
     _cons    50.907968     0.435441   116.911263     0.000000    50.092201    51.765503
 Total bootstraps completed: 500
```

Bootvare_v20 Results:

The REG Procedure
Dependent Variable: height
Weight: WT60LF

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	15	426030656	28402044	750.15	<.0001
Error	12423	470359719	37862		
Corrected Total	12438	896390375			

Root MSE	194.58162	R-Square	0.4753
Dependent Mean	55.33538	Adj R-Sq	0.4746
Coeff Var	351.64051		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	50.90797	0.21433	237.52	<.0001
nfld	1	-0.29227	0.30121	-0.97	0.3319
pei	1	-0.32853	0.56080	-0.59	0.5580
ns	1	-0.54143	0.23678	-2.29	0.0222
nb	1	-0.51469	0.25727	-2.00	0.0455
qc	1	-0.79238	0.13134	-6.03	<.0001
on	1	-0.40703	0.12280	-3.31	0.0009
man	1	0.24735	0.22511	1.10	0.2719
sask	1	0.10105	0.23464	0.43	0.6667
alb	1	0.23282	0.16058	1.45	0.1471
lshs	1	3.75908	0.11369	33.06	<.0001
someps	1	3.50695	0.11898	29.48	<.0001
ugrad	1	3.15580	0.11630	27.14	<.0001
agesq	1	-0.00472	0.00009228	-51.10	<.0001
age	1	0.46588	0.00821	56.78	<.0001
gender	1	-4.69528	0.07566	-62.06	<.0001

Variance estimation using 500 bootstraps for a Regression

Dependent variable: height

Obs	beta	bhat	bs_var	bs_sd	bs_cv	cil95	ciu95
1	Intercept	50.9080	0.18961	0.43544	0.86	50.0545	51.7614
2	nfld	-0.2923	0.04476	0.21157	72.39	-0.7069	0.1224
3	pei	-0.3285	0.05836	0.24158	73.53	-0.8020	0.1450
4	ns	-0.5414	0.04261	0.20642	38.13	-0.9460	-0.1368
5	nb	-0.5147	0.05367	0.23168	45.01	-0.9688	-0.0606
6	qc	-0.7924	0.02430	0.15590	19.67	-1.0979	-0.4868
7	on	-0.4070	0.02601	0.16128	39.62	-0.7231	-0.0909
8	man	0.2474	0.04156	0.20386	82.42	-0.1522	0.6469
9	sask	0.1011	0.04791	0.21889	216.60	-0.3280	0.5301
10	alb	0.2328	0.03744	0.19349	83.11	-0.1464	0.6121
11	lshs	3.7591	0.02952	0.17181	4.57	3.4223	4.0958
12	someps	3.5069	0.03021	0.17380	4.96	3.1663	3.8476
13	ugrad	3.1558	0.02498	0.15805	5.01	2.8460	3.4656
14	agesq	-0.0047	0.00000	0.00016	3.35	-0.0050	-0.0044
15	age	0.4659	0.00023	0.01528	3.28	0.4359	0.4958
16	gender	-4.6953	0.00912	0.09550	2.03	-4.8825	-4.5081

Appendix III

Frequency Tables:

```
. tab nfld [aw=wt60lf]

       nfld |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 | 12215.0502       98.20       98.20
          1 | 223.949797        1.80      100.00
------------+-----------------------------------
      Total |      12439      100.00


. reg nfld [aw=wt60lf]
(sum of wgt is   2.6597e+07)

      Source |       SS       df       MS              Number of obs =   12439
-------------+------------------------------           F(  0, 12438) =    0.00
       Model |        0.00     0           .           Prob > F      =       .
    Residual |   219.91784 12438  .017681126           R-squared     =  0.0000
-------------+------------------------------           Adj R-squared =  0.0000
       Total |   219.91784 12438  .017681126           Root MSE      =  .13297

------------------------------------------------------------------------------
        nfld |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       _cons |   .0180038   .0011922    15.10   0.000     .0156669    .0203408
------------------------------------------------------------------------------


. bswreg nfld [aw=wt60lf] , cmd(reg) bsw(bsw1-bsw500)
Results from BSWREG

* The confidence intervals below are based on the normal distribution

  Var_name         Coef        BS_se     BS_zstat    BS_pvalue   BS_cilow95    BS_ciup95
     _cons     0.018004     0.000460    39.113152     0.000000     0.017102     0.018906

Total bootstraps completed: 500

**********************************************************************************************


Bootvare_v20
Variance estimation using 500 bootstraps for Totals and Ratios

Obs    type     var1    var2           yhat          bs_sd    bs_cv          cil95          ciu95
 1     Ratio    nfld    count          1.80           0.05     2.56           1.71           1.89

where count is equal to the population size.
```

Means:

```
. bysort gender: means age [aw=wt60lf]
-------------------------------------------------------------------------------
-> gender = 1
    Variable |    Type        Obs        Mean       [95% Conf. Interval]
-------------+-----------------------------------------------------------
         age | Arithmetic    5597    38.74955        38.23552   39.26358
             |  Geometric    5597    32.88785        32.35452   33.42998
             |   Harmonic    5597    26.32428        25.78611   26.88539
-------------------------------------------------------------------------------
-------------------------------------------------------------------------------
-> gender = 2
    Variable |    Type        Obs        Mean       [95% Conf. Interval]
-------------+-----------------------------------------------------------
         age | Arithmetic    6842     40.6122        40.12048   41.10393
             |  Geometric    6842    34.30807        33.79512   34.82881
             |   Harmonic    6842    27.15345        26.63015   27.69773
-------------------------------------------------------------------------------
```

```
. bysort gender: reg age [aw=wt60lf]
-------------------------------------------------------------------------------
-> gender = 1
(sum of wgt is   1.3123e+07)

      Source |       SS       df       MS              Number of obs =    5597
-------------+------------------------------           F(  0,  5596) =    0.00
       Model |        0.00     0           .           Prob > F      =       .
    Residual |  2153437.42  5596  384.817267           R-squared     =  0.0000
-------------+------------------------------           Adj R-squared =  0.0000
       Total |  2153437.42  5596  384.817267           Root MSE      =  19.617

------------------------------------------------------------------------------
         age |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       _cons |   38.74955   .2622102   147.78   0.000     38.23552    39.26358
------------------------------------------------------------------------------
-------------------------------------------------------------------------------
-> gender = 2
(sum of wgt is   1.3474e+07)

      Source |       SS       df       MS              Number of obs =    6842
-------------+------------------------------           F(  0,  6841) =    0.00
       Model |        0.00     0           .           Prob > F      =       .
    Residual |  2945099.01  6841  430.507092           R-squared     =  0.0000
-------------+------------------------------           Adj R-squared =  0.0000
       Total |  2945099.01  6841  430.507092           Root MSE      =  20.749

------------------------------------------------------------------------------
         age |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       _cons |    40.6122   .2508411   161.90   0.000     40.12048    41.10393
------------------------------------------------------------------------------
```

```
. bysort gender: bswreg age [aw=wt60lf] , cmd(reg) bsw(bsw1-bsw500)
-------------------------------------------------------------------------------
-> gender = 1
Results from BSWREG

* The confidence intervals below are based on the normal distribution

  Var_name         Coef        BS_se     BS_zstat    BS_pvalue   BS_cilow95    BS_ciup95
     _cons    38.749551     0.137592   281.626648     0.000000    38.479877    39.019226

Total bootstraps completed: 500
-------------------------------------------------------------------------------
-> gender = 2
Results from BSWREG

* The confidence intervals below are based on the normal distribution

  Var_name         Coef        BS_se     BS_zstat    BS_pvalue   BS_cilow95    BS_ciup95
     _cons    40.612205     0.127663   318.121521     0.000000    40.361992    40.862419

Total bootstraps completed: 500

******************************************************************************************
```

SAS — Proc Reg Output

```
gender=1 ------------------------------------------------------
                                        Dependent Variable: age
                                            Weight: WT60LF
                                          Parameter Estimates

                                       Parameter       Standard
                         Variable     DF       Estimate          Error    t Value    Pr > |t|
                         Intercept     1       38.74955        0.26221     147.78      <.0001
gender=2 ------------------------------------------------------
                                        Dependent Variable: age
                                            Weight: WT60LF
                                          Parameter Estimates

                                       Parameter       Standard
                         Variable     DF       Estimate          Error    t Value    Pr > |t|
                         Intercept     1       40.61220        0.25084     161.90      <.0001
```

Bootvare_v20

```
Variance estimation using 500 bootstraps for a Regression
Dependent variable: age
Obs    gender      beta         bhat      bs_var      bs_sd     bs_cv     cil95      ciu95
1        1      Intercept    38.7496    0.018932    0.13759     0.36    38.4799    39.0192
2        2      Intercept    40.6122    0.016298    0.12766     0.31    40.3620    40.8624
```

Ratios:

```
. gen ah=age/height

. svyratio age/height [pw=wt60lf]

Survey ratio estimation

pweight:  wt60lf                                  Number of obs    =     12439
Strata:   <one>                                   Number of strata =         1
PSU:      <observations>                          Number of PSUs   =     12439
                                                  Population size  =  26596782

------------------------------------------------------------------------------
      Ratio       |   Estimate    Std. Err.   [95% Conf. Interval]        Deff
------------------+-----------------------------------------------------------
     age/height   |   .7173197    .0041964    .7090941    .7255453    1.752756
------------------------------------------------------------------------------



. bswreg age height [pw=wt60lf], cmd(svyratio) bsw(bsw1-bsw500)

50 bootstraps completed
100 bootstraps completed
150 bootstraps completed
200 bootstraps completed
250 bootstraps completed
300 bootstraps completed
350 bootstraps completed
400 bootstraps completed
450 bootstraps completed
500 bootstraps completed
Results from BSWREG
-------------------

* The confidence intervals below are based on the normal distribution


  Var_name         Coef        BS_se     BS_zstat    BS_pvalue   BS_cilow95    BS_ciup95
age:height     0.717320     0.001743   411.545410     0.000000     0.713903     0.720736

Total bootstraps completed: 500

*******************************************************************************************
```

Bootvare_v20

```
Variance estimation using 500 bootstraps for Totals and Ratios

Obs    type     var1     var2           yhat          bs_sd    bs_cv         cil95         ciu95
 1     Ratio    age      height         71.73          0.17     0.24         71.39         72.07
```