Catalogue no. 12-002-XIE

The Research Data Centres Information and Technical Bulletin

Spring 2005, vol. 2 no.1

Statistics Canada Statistique Canada

Canada

How to obtain more information

Specific inquiries about this product and related statistics or services should be directed to: Research Data Centres Program, Statistics Canada, Ottawa, Ontario, K1A 0T6 (telephone: 1 800 263-1136).

For information on the wide range of data available from Statistics Canada, you can contact us by calling one of our toll-free numbers. You can also contact us by e-mail or by visiting our website.

National inquiries line	1 800 263-1136
National telecommunications device for the hearing impaired	1 800 363-7629
Depository Services Program inquiries	1 800 700-1033
Fax line for Depository Services Program	1 800 889-9734
E-mail inquiries	infostats@statcan.ca
Website	www.statcan.ca

Information to access the product

This product, catalogue no. 12-002-XIE, is available for free. To obtain a single issue, visit our website at www.statcan.ca and select Our Products and Services.

Standards of service to the public

Statistics Canada is committed to serving its clients in a prompt, reliable and courteous manner and in the official language of their choice. To this end, the Agency has developed standards of service that its employees observe in serving its clients. To obtain a copy of these service standards, please contact Statistics Canada toll free at 1 800 263-1136. The service standards are also published on www.statcan.ca under About Statistics Canada > Providing services to Canadians.

Statistics Canada
Research Data Centres Program

The Research Data Centres Information and Technical Bulletin

Spring 2005, vol. 2, no. 1

Published by authority of the Minister responsible for Statistics Canada

June 2005

Catalogue no. 12-002-XIE

Frequency: semi-annual

ISSN: 1710-2197

Ottawa

Cette publication est disponible en français (n° 12-002-XIF au catalogue)

Note of appreciation

Canada owes the success of its statistical system to a long-standing partnership between Statistics Canada, the citizens of Canada, its businesses, governments and other institutions. Accurate and timely statistical information could not be produced without their continued cooperation and goodwill.

About the Information and Technical Bulletin

The Research Data Centres Information and Technical Bulletin is a forum for current and prospective users of the centre to exchange practical information and techniques for analyzing datasets available at the centres. The bulletin is published twice per year, in the spring and fall. Additional special issues on timely topics may also be released on an occasional basis.

Aims:

The main aims of the bulletin are:

- to advance and disseminate knowledge surrounding Statistics Canada's data;
- to exchange ideas among the Research Data Centre (RDC) user community;
- to support new users of the RDC program; and
- to provide an additional means through which RDC users and subject matter experts and divisions within Statistics Canada can communicate.

Content:

The Information and Technical Bulletin is interested in receiving articles and notes that will add value to the quality of research produced at the Statistics Canada Research Data Centres and provide methodological support to RDC users.

Topics include, but are not limited to:

- data analysis and modeling;
- data management;
- best or ineffective statistical, computational, and scientific practices;
- data content;
- implications of questionnaire wording;
- comparisons of data sets;
- reviews on methodologies and their applications;
- problem-solving analytical techniques; and
- explanations of innovative tools, using surveys and relevant software available at the RDCs.

Those interested in submitting an article to the Information and Technical Bulletin are asked to refer to the Instructions for authors.

The editors and authors would like to thank the reviewers for their valuable comments.

Table of contents

Articles

Yves Lafortune and Georgia Roberts,
Comparing a rate in a subpopulation to the rate in the full population: How it may be done when using survey data, and available software tools 6

James Chowhan and Neil J. Buckley,
Using mean bootstrap weights in Stata: A BSWREG revision 23

Technical note

Franck Larouche and Charles Tardif,
The household as a unit of analysis in the National Longitudinal Survey of Children and Youth 38

Information Note

Cara B. Fedick, The CRISP-NLSCY Files 40

Editorial committee, Instructions for authors 43

Comparing a rate in a subpopulation to the rate in the full population: How it may be done when using survey data, and available software tools

By Yves Lafortune and Georgia Roberts

Abstract

People often wish to use survey micro-data to study whether the rate of occurrence of a particular condition in a subpopulation is the same as the rate of occurrence in the full population. This paper describes some alternatives for making inferences about such a rate difference and shows whether and how these alternatives may be implemented in three different survey software packages. The software packages illustrated – SUDAAN, WesVar and Bootvar – all can make use of bootstrap weights provided by the analyst to carry out variance estimation.

Introduction

A common question is whether the rate of occurrence of a particular condition in a subpopulation is the same as the rate of occurrence in the full population. As examples, a health authority might wonder whether the incidence rate of influenza in his/her health region is the same as the incidence rate for the full province; or a province may be curious as to whether the proportion of Grade 9 students who have done at least 10 hours of volunteer work in the past year is the same as the overall proportion of secondary school students who have volunteered to that extent. People often wish to study such questions through the use of survey data. The following notes give some indications on how this may be done, first in theory, and then from a practical point of view, when the survey micro-data, survey replication weights, and survey software tools such as Bootvar (a set of SAS or SPSS macros), SUDAAN or WesVar, are available.

II. A bit of theory

Suppose that there is some interest in comparing the smoking rate in Ontario to the overall smoking rate in Canada. Let p represent the true smoking rate in the Canadian population and let p_{ONT} be the true smoking rate in the province of Ontario. The hypothesis to be tested is then $H_0 : p_{ONT} = p$. One statistic that could be used to test this hypothesis (versus the alternative that the 2 proportions are not equal) is $T = (\hat{p}_{ONT} - \hat{p}) \Big/ \sqrt{\hat{\text{var}}(\hat{p}_{ONT} - \hat{p})}$, where $\hat{p}$, $\hat{p}_{ONT}$, and $\hat{\text{var}}(\hat{p}_{ONT} - \hat{p})$ are estimates obtained from the survey data. For testing the hypothesis, this statistic could be compared to the cut-points of a normal or of a *t* distribution, or the p-value of the statistic could be examined. For example, if the test is being carried out at a 95% level of significance, and if *T* is considered to be normally distributed, p could be declared significantly different from p_{ONT} if the value of T is greater than 1.96 or smaller than -1.96 or if the p-value given for the statistic is smaller than .05.

Estimates $\hat{p}$ and $\hat{p}_{ONT}$ are readily obtained from use of survey software, as are estimates of their individual variances. However, when testing the hypothesis of interest, we need an estimate of the variance of the difference, and this is where the problem arises. Because $\hat{p}$ and $\hat{p}_{ONT}$ are not estimated from independent samples – in fact, the sample for Ontario was chosen as part of the sample for Canada - the variance of the difference has a covariance component which must be accounted for (recall that $\text{var}(\hat{p}_{ONT} - \hat{p}) = \text{var}(\hat{p}_{ONT}) + \text{var}(\hat{p}) - 2\,\text{cov}(\hat{p}_{ONT}, \hat{p})$). In fact, the larger the subpopulation, the more likely this covariance term is to be non-negligible.

How might we then proceed to obtain a suitable value for the denominator of the test statistic T for the hypothesis of interest? Or might there even be a different test statistic for the same hypothesis, which we could calculate with our available software?

Solution #1: If a replication method of variance estimation, such as bootstrapping, is recommended for the survey, and if replication weights are available for the purposes of general variance estimation, then an estimate of $p_{ONT} - p$ could be produced using each of these replication weights, and then these replicate estimates could be used, together with the full-sample estimate, to obtain a variance estimate for the estimated difference of proportions (with the formula for obtaining the variance estimate depending on the particular replication method being used). A person could write his own program to do this, or choose a software tool that does this same thing as part of its routine output. Note that, in this approach, the covariance component of the variance of the difference does not need to be explicitly calculated.
[Solution #1 can be carried out with WesVar or Bootvar, but not with SUDAAN.]

Solution #2: Another possibility is to use a software tool that would provide an estimate of the full covariance matrix of a set of estimated proportions (or percentages), either in an output data set or in printed form. This covariance matrix would need to contain the variance estimates of the estimated full-population rate and of the corresponding estimated subpopulation rate. It must also contain an estimate of the covariance of the two estimated rates. From these three quantities, an estimate of the variance of the difference of the two estimated rates could then be calculated, using the formula $\hat{\text{var}}(\hat{p}_{ONT} - \hat{p}) = \hat{\text{var}}(\hat{p}_{ONT}) + \hat{\text{var}}(\hat{p}) - 2\,\text{cov}(\hat{p}_{ONT}, \hat{p})$, and this estimate could then be used in calculating the statistic T described above. If it is not straightforward to electronically extract the quantities required from an output covariance matrix, it might be easier to use paper and pencil to calculate the required T statistic. It is unlikely that Solution #2 would be chosen if Solution #1 is readily implemented with the software tool being used, since Solution #2 requires more work.
[Solution #2 can be carried out with SUDAAN, but not with WesVar or Bootvar.]

Solution #3: A test statistic involving a linear contrast between 2 subpopulation rates is often readily available in software – whereas a linear contrast between a subpopulation rate and a population rate is sometimes not. Therefore, a different way to attack the problem is the following. It can be readily shown (see Appendix 1) that the hypothesis to be tested - $H_0 : p_{ONT} = p$ - is equivalent to the hypothesis $H_0 : p_{ONT} = p_{ONT^C}$, where p_{ONT^C} is the true smoking rate in the rest of Canada (i.e. in the full population from which the subpopulation of interest has been deleted). In order to test this hypothesis, the test statistic

$T_2 = (\hat{p}_{ONT} - \hat{p}_{ONT^C}) \Big/ \sqrt{\hat{\text{var}}(\hat{p}_{ONT} - \hat{p}_{ONT^C})}$ can be used. This test statistic only involves a linear contrast between 2 subpopulation rates. Please note that the two test statistics (T and T_2) are not the same (even though both provide an answer to the same question), but they are highly likely to lead to the same conclusion.

[Solution #3 can be carried out with any of the three selected software; however, it is most appealing to SUDAAN for which Solution #1 is not available and for which Solution #2 is difficult to carry out completely electronically.]

It should be noted that all three solutions may be applied both in the case where the subpopulation and the remainder of the full population are independently sampled and in the case where there is dependence between the samples in the subpopulation and the remainder of the full population.

III. Implementing these solutions using survey software

The following examples are using the Health (h356) **synthetic** data file from the 1998-1999 (Cycle 3) National Population Health Survey (NPHS) to illustrate how to carry out one or more of the solutions with the different software. Requests to get these data should be directed to "Data Access Unit, Population Health Surveys, Health Statistics Division", e-mail: nphs-ensp@statcan.ca and/or cchs-escc@statcan.ca. The NPHS design information includes, for each individual, a final weight (WT68), as well as 500 **bootstrap** replicate weights (BSW1-BSW500). The unique identifier for each individual is given by the combination of the variables REALUKEY and PERSONID. It will be supposed that a SAS data set containing the survey data is available to the user. Indications of how to set up the data properly are included with each example.

The Cycle 3 Health (h356) synthetic data file includes the answers of 17 244 individuals aged from 0 to 99 years old. The questions related to smoking were only asked of individuals 12 years old or older. Therefore, the records of 1995 individuals aged from 0 to 11 are not relevant for this analysis, since these individuals were not asked the questions about smoking. In addition, the answers to the smoking question of interest are missing for 32 individuals. Considering the small proportion of individuals this represents, it will be assumed that simply excluding those with non-response to the smoking question has no impact on the results.

This means that the results should be based upon a total of 15 217 individuals, representing a total of 24 859 391 Canadians 12 years old or older. In all three software, the full file containing all 17 244 individuals will be used, but the coding and the methods used to get the results will ensure that results are based only upon the 15 217 valid respondents.

We will make use of the following variables:

PRC8_CUR: "Province of Residence at the time of data collection in 1998-1999". Possible values are NFLD (10), PEI (11), NS (12), NB (13), QUE (24), ONT (35), MB (46), SK (47), AB (48) and BC (59).

SMC8_2: "At the present time, do you smoke cigarettes daily, occasionally or not at all?" The possible values are DAILY (1), OCCASIONALLY (2), NOT AT ALL (3), NOT APPLICABLE (6), DON'T KNOW (7), REFUSAL (8) and NOT STATED (9). We will define as a 'Regular Smoker' someone who was smoking cigarettes daily at the time of data collection (i.e. SMC8_2=1).

The values 6, 7, 8 and 9 of the SMC8_2 variable will be changed to a missing value '.' for the 1995 + 32 non-valid respondents, to insure that the results in WesVar and SUDAAN are based only on the 15 217 valid respondents. (Note that it would have been possible to obtain the same results while still keeping the original codes for SMC8_2, although this would have required the use of some more advanced options in WesVar and SUDAAN.) With Bootvar, additional steps will be required to ensure the validity of the results and those steps will be discussed in Example 4.

We are thus interested in the rate of regular smoking among people aged 12 years and older. We wish to make the comparison of smoking rates in Ontario and in all of Canada.

IV. Example 1 –Using WesVar 4.2 to illustrate Solution #1:

WesVar 4.2 is capable of importing SAS v8 data files. Once the data have been made available within WesVar, the user must then properly assign each variable to its respective use. The 'Full Sample' box should contain the final weight; the 'ID' box should contain the unique identifier variables; the 'Replicates' box should contain the variables that are the replicate weights; and the 'Variables' box should contain all remaining variables of interest (which could be many more than needed for this particular example). Although a bootstrap survey design is not directly available within WesVar, Phillips (2004) shows that it is possible to use the BRR design option in WesVar to compute bootstrap variance estimates, as long as the bootstrap weights have been generated outside of the software. Therefore, a correct way to set up the NPHS cycle 3 synthetic data within WesVar is according to the scheme given in the following figure (Figure 1) :

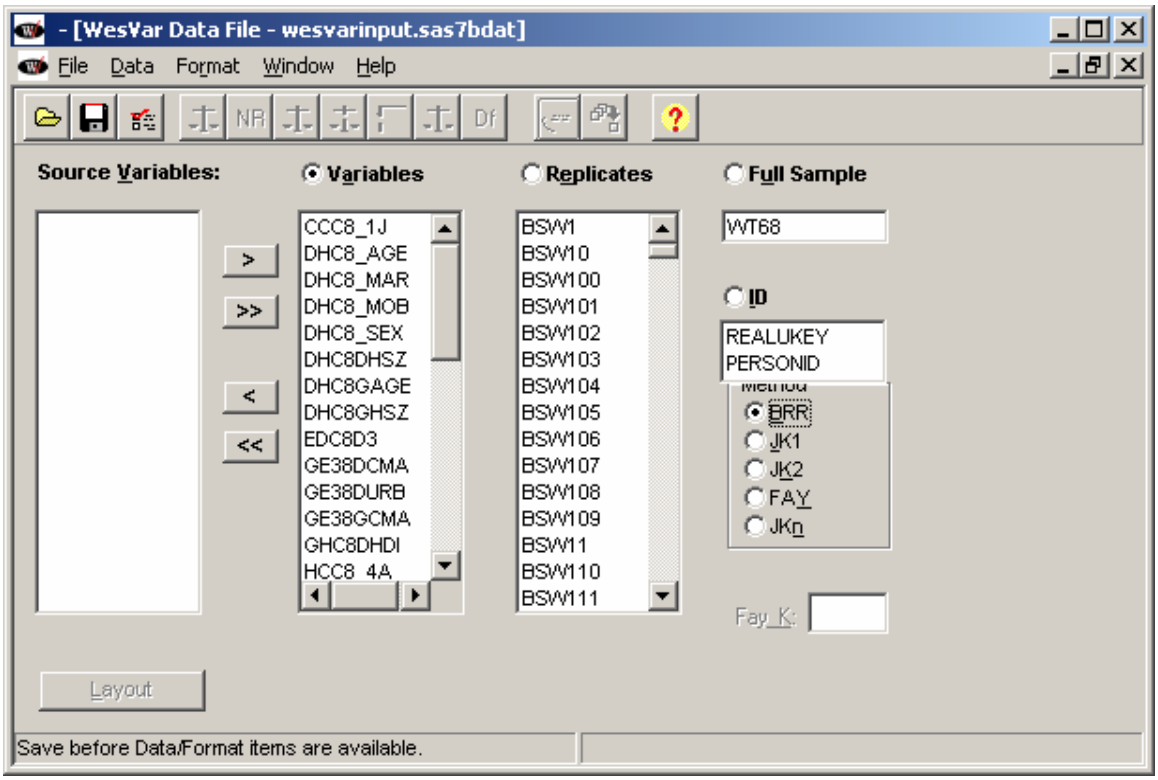

Figure 1

Once this is done, we can proceed by first saving the data, exiting and then creating a new workbook. Recall that the variable identifying the province of residence is given by PRC8_CUR (with a value of 35 for Ontario), whereas the variable identifying the type of smoker is SMC8_2 (with a value of 1 for regular smoker). We will now generate the Canadian and Ontarian smoking rates. In order to do that, a new table request must be submitted. First, an additional generated statistic will be added, since, by default, p-values are not part of the output. This can be done by first clicking on the Generated Statistics panel and by adding a check mark in the last box. The next picture (Figure 2) shows how it should look:

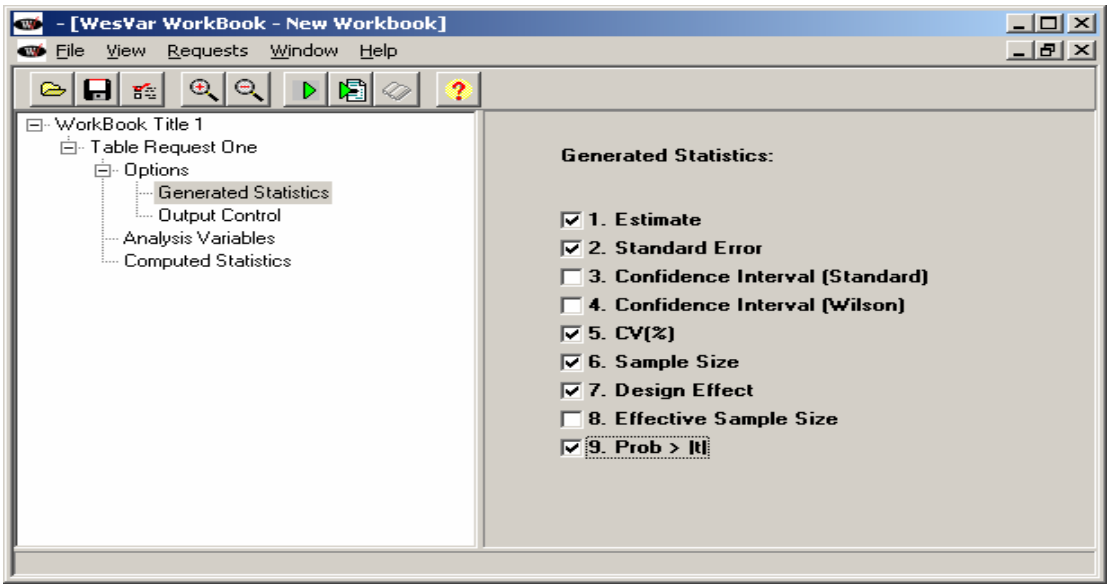

Figure 2

Then, after clicking back on the Table Request node in the left panel, the user can click on Add Table Set (Single) and assign the required variables to the Selected box (see Figure 3). Here, the percentages of interest are the row percents, since PRC8_CUR was the first variable selected.

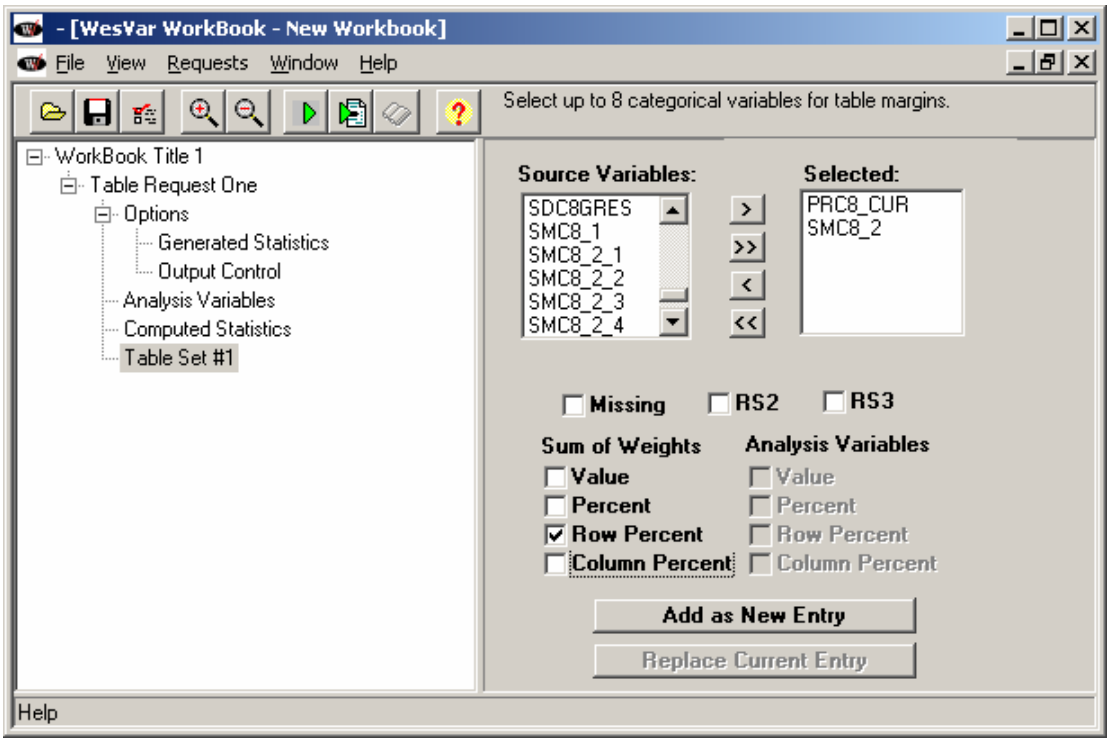

Figure 3

Add this table as a new entry, and click on the little + sign appearing on the left of the table name (label by the variable names forming the table). This will give you access to three additional panels (Cells, Cell Functions and Standardized Rates). First click on *Cells* to assign labels to the cells in the table that are of particular interest. For example, the label *Can_Smk_rate* will be assigned to the cell having a value of 'Marginal' for PRC8_CUR and '1' for SMC8_2, whereas the label *Ont_Smk_rate* will be assigned to the cell having a value '35' for PRC8_CUR and '1' for SMC8_2, just as below (Figure 4).

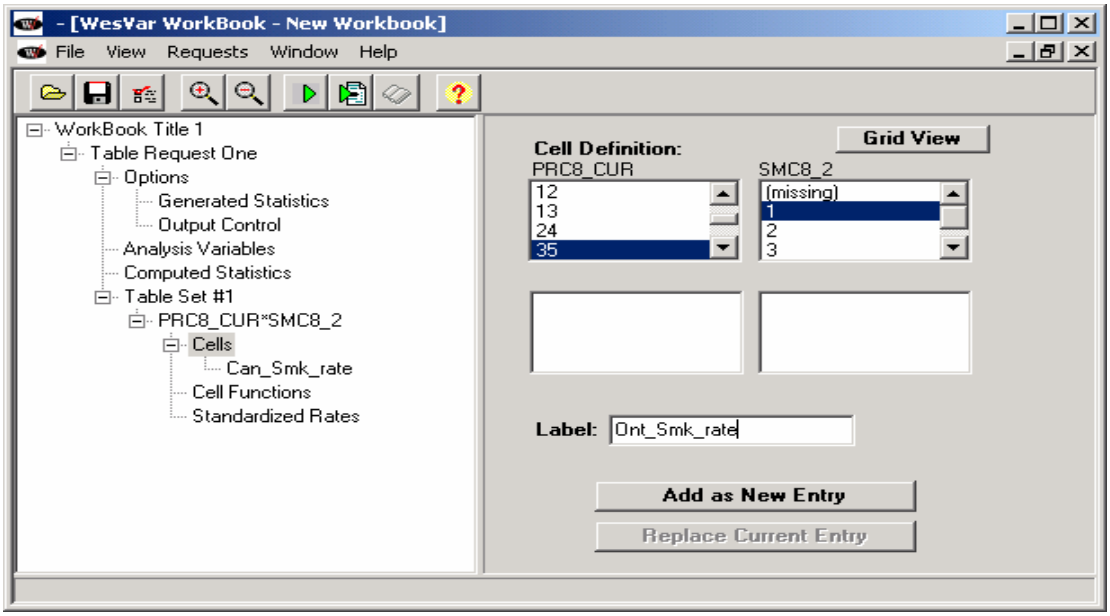

Figure 4

Then, because we want the difference between the Ontarian smoking rate and the Canadian smoking rate, the *Cell Functions* panel will be used as well. The statistic 'Diff_Smk_rate' = 'Ont_Smk_rate' - 'Can_Smk_rate' is added (see Figure 5).

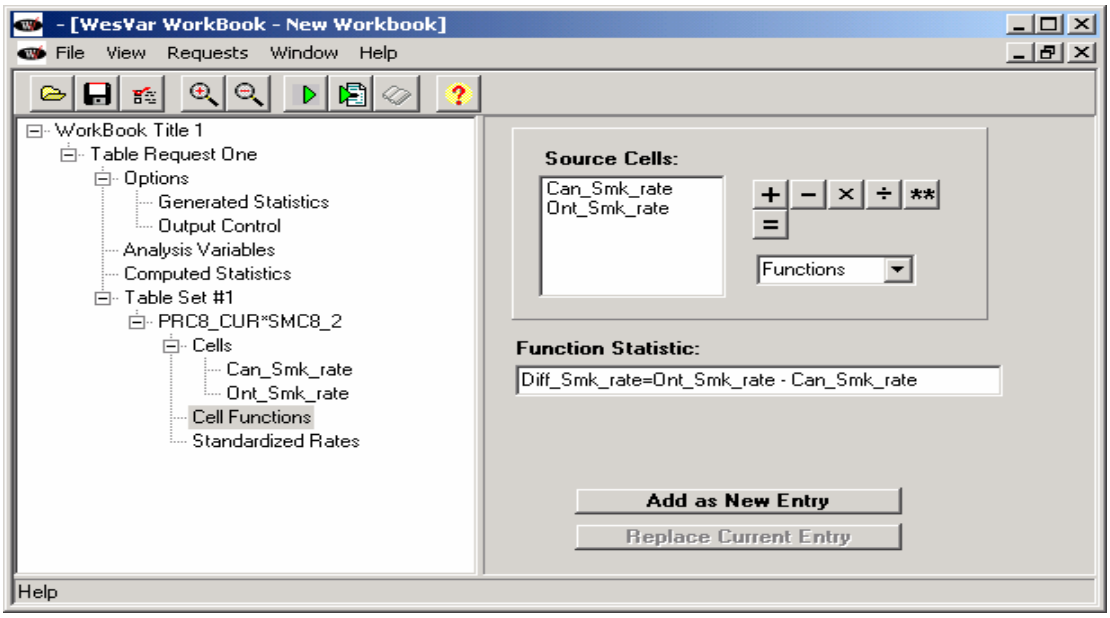

Figure 5

Let us now run the Table Request and look at the results by clicking on the open book icon (once available). We then click on the Functions node in the left panel to look at the results of the requested comparison, which are shown in the next picture (Figure 6).

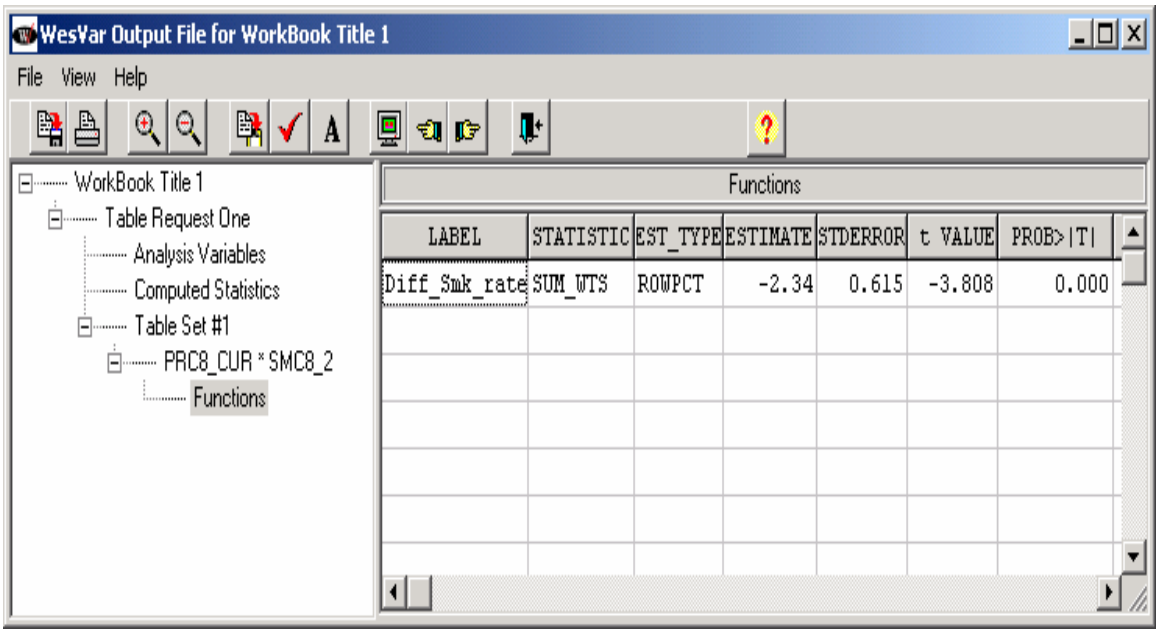

Figure 6

The difference in the smoking rates between Ontario and Canada is estimated to be about 2.34%. The value of the statistic T is called *t Value* in the output and the p-value is called PROB>|T|. Since the p-value for Diff_Smk_rate is smaller than 0.05, we can conclude, at the 5% level, that there is a significant difference between the Canadian and Ontarian smoking rates. Keep in mind that these conclusions are based on synthetic data and should not be trusted in any way.

V. Example 2 – Using SUDAAN 8.0.2 or SUDAAN 9.0.1 to illustrate Solution #2:

SUDAAN is now SAS-callable, which means that it can be used within a SAS environment. For most purposes, it can be viewed as if new additional SAS procedures had been made available to the SAS user. Once a person gets past learning the coding specificities of the SUDAAN procedures, he/she will be just as functional in SUDAAN as in SAS.

With the NPHS data, in order to compute bootstrap variance estimates, the user must use the same trick as in WesVar and specify the design to be a balanced repeated replication (BRR) design, but include the bootstrap weights on the data file as if they were BRR weights.

PROC DESCRIPT is a SUDAAN procedure capable of computing totals, means, proportions and percentiles for populations and subpopulations. It has the capability to create in output the full variance-covariance matrix of the requested estimates, as well as the covariances between the full-population estimates, and any subpopulation estimates requested. Therefore, it is possible to put in place Solution #2. Here is a piece of code using SUDAAN 8.0.2 PROC DESCRIPT that would allow the user to do that.

```
proc descript data=nphsdum3 design=brr;
       weight wt68;
       repwgt bsw1-bsw500;
       recode prc8_cur=(10 11 12 13 24 35 46 47 48 59);
       subgroup prc8_cur;
       levels 10;
       var smc8_2;
       catlevel 1;
       tables prc8_cur;
       output / pctcov=all filename=out1 REPLACE filetype=sas;
run;
```

This procedure computes the proportion of regular smokers (SMC8_2 with a value of 1) for all provinces separately, as well as for the whole country. It also creates an output file named out101 (SUDAAN adds a suffix) containing, among other things, the estimated rates (in the first row), as well as the variance-covariance matrix of all estimates (in subsequent rows). The table below reports the values for Canada and for each province. The numbers of specific interest for the example have been bolded for the benefit of the reader (see Table 1).

Table 1

CANADA	NFLD	PEI	NS	NB	QUE	ONT	MB	SK	AB	BC
24.241	24.921	26.444	26.615	26.988	28.163	**21.899**	24.710	25.340	25.144	21.445
0.218	0.092	-0.001	0.087	0.038	0.310	**0.207**	0.143	0.088	0.263	0.195
0.092	2.713	0.207	0.028	-0.082	-0.020	0.047	0.069	-0.108	0.148	0.120
-0.001	0.207	2.714	-0.022	-0.098	0.082	-0.075	0.077	-0.092	-0.125	0.053
0.087	0.028	-0.022	3.160	0.255	0.008	-0.017	0.025	0.130	-0.054	-0.100
0.038	-0.082	-0.098	0.255	3.031	-0.157	0.005	0.187	0.173	-0.056	-0.112
0.310	-0.020	0.082	0.008	-0.157	1.318	-0.056	0.019	0.010	0.153	-0.019
0.207	0.047	-0.075	-0.017	0.005	-0.056	**0.574**	0.004	0.004	0.031	-0.004
0.143	0.069	0.077	0.025	0.187	0.019	0.004	3.636	0.137	-0.133	0.058
0.088	-0.108	-0.092	0.130	0.173	0.010	0.004	0.137	2.454	0.070	-0.095
0.263	0.148	-0.125	-0.054	-0.056	0.153	0.031	-0.133	0.070	2.089	0.146
0.195	0.120	0.053	-0.100	-0.112	-0.019	-0.004	0.058	-0.095	0.146	1.437

According to the synthetic data used here, the Canada smoking rate is estimated to be 24.241% with an estimated standard error of sqrt(0.218)=0.4669%, whereas the Ontario smoking rate is estimated to be 21.899%, with an estimated standard error of sqrt(0.574)=0.7576%. The covariance between the two rate estimates is given by $0.207(\%)^2$.

Therefore, the test of equality of rates leads to the following statistic:

$$T = (\hat{p}_{ONT} - \hat{p}) \Big/ \sqrt{\hat{\text{var}}(\hat{p}_{ONT} - \hat{p})} = \frac{21.899 - 24.241}{\sqrt{0.574 + 0.218 - 2*(0.207)}} = \frac{-2.342}{0.6148} = -3.81.$$

Please note that the difference, the standard error and the T-statistic are the same as those obtained with WesVar in Example 1. We would conclude once again that the two rates are significantly different at the 5% level, since -3.81 is smaller than -1.96 (and we are assuming T to be normally distributed).

Note that there is currently a bug in SUDAAN 9.0.0 that prevents the user from creating the output file containing the variance-covariance matrix. The user must request the printing of the variance-covariance matrix on the screen and then use a pen and a paper to compute the statistic, or rely on SUDAAN 8.0.2 or 9.0.1 to create the output file, or make use of Solution #3.

VI. Example 3 - Using SUDAAN 9.0.0 or SUDAAN 9.0.1 to illustrate Solution #3:

Unfortunately, the SUDAAN output shown in Example 2 is quite difficult to read, which leads to potential errors, as well as to difficulties of automating the process. It is also not

currently possible to output the required file in version 9.0.0 because of a bug. Rather than relying on pen and paper, it is easier to modify the code above and make use of Solution #3.

```
data smktest;
       set nphsdum3(keep=prc8_cur smc8_2 dhc8_sex wt68 bsw1-bsw500);
       if prc8_cur=. then ontario=.;
       else if prc8_cur=35 then ontario=1;
       else ontario=2;
run;

proc descript data=smktest design=brr;
       weight wt68;
       repwgt bsw1-bsw500;
       class ontario;
       var smc8_2;
       catlevel 1;
       diffvar ontario=(1 2) / name="Ontario vs Rest of Canada: Smk_Rate";
run;
```

In the data step shown above, a variable called 'ontario' is created, taking a value of 1 for residents of Ontario, and a value of 2 for non-residents of Ontario. A call to PROC DESCRIPT similar to that shown in Example 2 is then run, but the province variable is removed from the *tables* statement and put into a *diffvar* statement. The *diffvar* statement can be used to specify linear contrasts that are simple differences between two levels of a class variable. In this example, the difference in the smoking rate between Ontario (ontario=1) and the rest of Canada (ontario=2) is the contrast of interest. Here is a small portion of the output, presenting the results of the comparisons (see Table 2).

Table 2

One		Contrast Ontario vs Rest of Canada: Smk_Rate
Total	Sample Size	15217
	Weighted Size	24859390.94
	Cntrst Total	-1888634.79
	Cntrst Pct	-3.78
	SE Cntrst Pct	0.99
	T-Test	
	Cont.Pct=0	-3.81
	P-value T-Test	
	Cont.Pct=0	0.0002

It should be noted that the value of the difference (Cntrst Pct) differs from the ones reported in Examples 1 and 2, as well as the value of the reported standard error. This is not an error! It is because here we make use of Solution #3, where we are comparing Ontario to the rest of Canada, rather than Ontario to all of Canada.

Nonetheless, the reported T-Test (which is what we called the T_2 statistic earlier) and its associated p-value are almost identical to the ones obtained in Examples 1 and 2. The conclusion remains the same. Once again, keep in mind that these conclusions are based on synthetic data and should not be trusted in any way.

VII. Example 4 – Using Bootvar 3.1 to illustrate Solution #1:

The Bootvar program is a set of SAS or SPSS macros developed by methodologists at Statistics Canada to ease the computation of variance estimates with bootstrap weights. Earlier versions of Bootvar used to accompany NPHS micro-data files. The more generic version 3.1 can now be used with many other Statistics Canada surveys. Bootvar 3.1 is capable of calculating variances of totals, ratios, differences between ratios, percentiles, chi-square tests and linear or logistic regression parameters. We will discuss the use of the SAS version of the program.

Variance estimation is performed in ***two steps*** and involves the use of three SAS programs. The ***first step*** consists of creating a data file containing the variables required for the analysis (first program). The ***second step*** involves using BOOTVARE_V31.SAS (and MACROE_V31.SAS) to estimate the variances.

During the first step, variables derived from the input variables should be created. This means that dichotomous variables (often called binary, dummy or 0/1 variables) identifying records that have a characteristic of interest – say being a regular smoker or being from Ontario – need to be created.

The analytical file should contain:

- The necessary variables for the analysis (derived variables including dichotomous variables, and input variables that do not need to be modified).
- The unique identifier variable(s) of the respondents.
- If needed, the breakdown variable(s), identifying the groups for which a separate analysis is desired.

In order to compute the Canadian smoking rate and the Ontarian smoking rate, four new dichotomous variables will need to be created. This is where the user must take precautions to avoid obtaining invalid results. The individuals for whom the SMC8_2 variable is missing should not contribute to the estimates of smoking rate. This means that they should not contribute to either of the two estimated totals that make up each rate: the total number of regular smokers in each domain and the total number of in-scope people in each domain. Since Bootvar computes these two totals separately before calculating its ratio estimate, a missing value on an observation for one of the variables does not ensure that the observation will not get used for the total of the second variable. Therefore, to indicate whether or not an individual is a valid respondent from Canada and/or from Ontario, the dichotomous variables created for this purpose should make use only of respondents that were asked the SMC8_2 question and answered with a valid answer. Then, out of those, the regular smokers will be identified by the creation of two

more dichotomous flags. Here is the SAS code used to create the analysis file that will be used within Bootvar:

```
data in1.nphs_dummy_cyc3;

       %let datafid= "H:\SSMD-DMES\CRAD-
       DARC\Course0438\NPHS_c3_dummy_files\Data\Dumyh356.txt";

       %include "H:\SSMD-DMES\CRAD-
       DARC\Course0438\NPHS_c3_dummy_files\Layout\h356_I.sas";
       /* The following statement has been added to the h356_i.sas file to change the
       non-response codes to a missing value:
       if smc8_2 in (6,7,8,9) then smc8_2=.;*/

       if smc8_2=. then canada=.;
       else canada=1;
       if smc8_2=. then ontario=.;
       else if prc8_cur=35 then ontario=1;
       else ontario=0;

       if smc8_2=. then smoker=.;
       else if smc8_2=1 then smoker=1;
       else smoker=0;

       ont_smoker=ontario*smoker;

       keep canada ontario smoker ont_smoker realukey personid wt68;

run;
```

Once the analysis file is ready, we need to use the second program BOOTVARE_V31.SAS:

- to load in the bootstrap weights

```
data bootwt;

       %let datafid="H:\SSMD-DMES\CRAD-
       DARC\Course0438\NPHS_c3_dummy_files\Bootstrp\bd5h356.txt";
       %include "H:\SSMD-DMES\CRAD-
       DARC\Course0438\NPHS_c3_dummy_files\Bootstrp\Layout\b356_i.sas";
run;
%let bsamp=bootwt;
```

- to specify that the analysis is to be done at the global level (without any breakdown variable)

```
%let classes = .;
```

- to specify the survey parameters

```
%let ident = realukey personid;
%let fwgt  = fwgt;
%let bsw   = bsw;
%let R     = 1;
%let B     = 500;
```

- to specify the statistics of interest

Each smoking rate is actually a ratio of the estimated number of people who are regular smokers to the estimated total number of people. When we want to compare the rate for Ontario to the Canadian rate, we are interested in a difference of two ratios. This is why the way to obtain this type of analysis with Bootvar is to use the following macros:

```
%ratio(smoker, canada);
%ratio(ont_smoker, ontario);

%diff_rat(ont_smoker,ontario,smoker,canada);
```

Note that the two %ratio commands are not required, unless the user also wants to see what the individual proportions are.

Here is the output from the %diff_rat macro:

```
Variance Estimation for a DIFFERENCE BETWEEN RATIOS
                                    using 500 bootstrap replicates

                                                                                    Lower limit Upper limit
                                                                                    confidence  confidence
                                    Num1  Num2 Difference          p      Std.        interval    interval
   Num1     Den1      Num2     Den2  size  size of ratios     z    value   err.   C.V.     95%         95%

 ont_smok  ontario   smoker   canada   887  3666   -0.0234    -3.81  0.0001  0.0061  26.25    -0.0355      -0.0114
```

In this output, the z value is the value of the T statistic. The associated p-value for the T statistic is almost identical to the one obtained in WesVar in Example 1. The small difference is due to the fact that WesVar calculates p-values under the assumption of a *t* distribution for the T statistic while BootVar makes the assumption of a normal distribution. (See Appendix 2 for a further discussion of the use of these two different distributions.) The conclusion remains the same. We should conclude at the 5% level that there is a significant difference between the Canadian and Ontarian smoking rate. Once again, keep in mind that these conclusions are based on synthetic data and should not be trusted in any way.

VIII. Concluding remarks

People often wish to make inferences about a difference between a subpopulation and the full population with respect to the rate of occurrence of a particular condition. People with access to confidential micro-data from some of Statistics Canada's major analytical surveys need approaches for making such inferences that can be implemented in the software tools available to them. These software tools must be able to do variance estimation using a survey bootstrapping approach since bootstrap weights are the form in which design information is provided for many of Statistics Canada surveys. This paper uses a synthetic data set from NPHS to illustrate three ways that the inferences may be carried out in the software tools SUDAAN, WesVar and Bootvar. None of these software packages can readily implement all three solutions, but for each of the software packages at least one of the solutions is straightforward. In particular, we would suggest Solution #1 for WesVar and Bootvar (see Examples 1 and 4) and Solution #3 for SUDAAN (see Example 3).

References

Phillips, Owen. 2004. "Using Bootstrap Weights with WesVar and SUDAAN." The Research Data Centres Information and Technical Bulletin. (Fall) 1(2):1-10. Statistics Canada Catalogue no. 12-002-XIE.

Research Triangle Institute (2004). SUDAAN Language Manual, Release 9.0 Research Triangle Park, NC: Research Triangle Institute.

Westat (2002). WesVar 4.2 User's Guide, Rockville, MD.

Appendix 1

Showing equivalence of two hypotheses

This appendix shows why comparing a simple rate in a subpopulation to the simple rate in the full population is equivalent to comparing the rate in the subpopulation to the rate in the rest of the population.

Suppose that a population is split into a subpopulation A and the rest of the population, denoted by A^c. Let N_A and N_{A^c} be the number of individuals in A and A^c respectively (so that the number of individuals in the whole population is $N = N_A + N_{A^c}$). Also, let x_A and x_{A^c} be the number of people with the condition under study in A and A^c respectively (so that the number of individuals with the condition in the whole population is $x = x_A + x_{A^c}$). Then the rates of occurrence of the condition in subpopulation A, in subpopulation A^c, and in the whole population are, respectively, $p_A = {x_A}/{N_A}$, $p_{A^c} = {x_{A^c}}/{N_{A^c}}$, and $p = {x}/{N}$.

Therefore, we have:

$$p_A = p \Leftrightarrow \frac{x_A}{N_A} = \frac{x}{N}$$

$$\Leftrightarrow \frac{x_A}{N_A} = \frac{x_A + x_{A^c}}{N_A + N_{A^c}}$$

$$\Leftrightarrow x_A(N_A + N_{A^c}) = N_A(x_A + x_{A^c})$$

$$\Leftrightarrow x_A N_{A^c} = N_A x_{A^c}$$

$$\Leftrightarrow \frac{x_A}{N_A} = \frac{x_{A^c}}{N_{A^c}}$$

$$\Leftrightarrow p_A = p_{A^c}$$

Appendix 2

Normal or t distributions for test statistics – the df problem

Recall that, in the section on "A bit of theory", it was stated that the value of the test statistic could be compared to the cut-points of either a normal or a t distribution. This is because, under the sample sizes generally encountered with survey analyses, there would be little difference in the normal and t distributions to which the test statistic would be compared. However, if a *t* distribution is used – which is the case for SUDAAN and WesVar - the number of degrees of freedom of the distribution must be specified. The usual recommendation for the degrees of freedom is to use the number of primary sampling units containing sampled individuals in the (sub)population being studied, less the number of strata containing sampled individuals in the (sub)population being studied. For Statistics Canada surveys where bootstrap weights are provided, the analyst does not have readily available information about the numbers of psu's or strata. Thus, the tendancy is to use the "default" number of degrees of freedom used by the software package – which is the number of replicate weights. If an analysis includes most of the survey sample, this "default" is likely to be a conservative estimate, but if an analysis is focused on a small subpopulation, where the t distribution is probably a better approximation than the normal, this "default" could be far too large. It is always a good idea, when using a small subpopulation, to consider whether the results of a test would change if the degrees of freedom of a test statistic were reduced; if this is the case, it may be worthwhile to explore what would be a better estimate of the df than the "default" estimate.

Note: In the particular analysis that we are carrying out in this paper, if Solution #1 or #2 is being used, where a rate in a subpopulation is being compared to the rate in the full population, the recommended degrees of freedom for the test statistic under the assumption of a t distribution would be the number of primary sampling units containing sampled individuals in the

subpopulation less the number of strata containing sampled individuals in the subpopulation. If Solution #3 is being used, where two subpopulation rates are being compared, the number of primary sampling units containing sampled individuals in the subpopulation less the number of strata containing sampled individuals in the subpopulation would be calculated for each subpopulation and the lesser of the two values would be the recommended number of degrees of freedom. Since, as stated above, the analyst does not have readily available information about psu's and strata when using bootstrap replicate weights, the analyst has to use caution when interpreting the results of the inferences if he/she is working with a subpopulation having a small sample.

Using mean bootstrap weights in Stata: A BSWREG revision

By James Chowhan and Neil J. Buckley

Abstract

This article presents revisions to a Stata "bswreg" ado file that calculates variance estimates using bootstrap weights. This revision adds new output and analytic features. The main feature added to the program enables researchers to apply mean bootstrap weights while accounting for the number of weights used to generate the average bootstrap weight. The Workplace and Employee Survey dataset will be used to illustrate the usefulness of this program. This revised version of the "bswreg" command is still an easy to use flexible tool, which is compatible with a wide variety of regression analytical techniques and datasets. The bswreg command and design-based bootstrap weights should only be used for inference when it is theoretically valid.

Introduction

This article presents revisions to "bswreg". BSWREG is a Stata ado file that was developed to calculate variance estimates using bootstrap weights. Piérard et al [2004] developed this program to provide researchers with an easy-to-use and flexible tool within Stata that can be employed with bootstrap weights to make use of complex survey design information and to calculate sampling variance estimates that account for survey design. Refer to Piérard et al [2004] for details on how to use the bswreg program, its unique features, and for tests validating the program's robustness. This article assumes some familiarity with this previous report.

The revised version of the program adds new features to the output displayed by the program after command execution, but more importantly the revisions allow researchers to use mean bootstrap weights while accounting for the number of weights used to generate the average bootstrap weight. Thus, the program has been designed to account for the fact that some Statistics Canada surveys provide average bootstrap weights. The BSWREG program is provided in Appendix 1.

The Workplace and Employee Survey (WES) data are used for this article to present an example of how important it is to account for the mean bootstrap when calculating design-based variance estimates, in comparison to the method used for standard bootstrap weights.

II. A brief comparison of standard and mean bootstrap

Many of Statistics Canada's surveys provide a final weight (or final design weight) and bootstrap weights, which can be used by researchers to generate consistent estimates of population parameters, and sampling variances that account for sample design, respectively.

The standard bootstrap variance estimator for $\hat{\theta}$, used in this program, is given by Yeo et al. [1999; 3]:

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

However, this variance estimator is inappropriate when the bootstrap weights are *mean bootstrap* weights. Mean bootstrap weights are bootstrap weights that have been averaged over C iterations usually to protect the confidentiality of survey respondents.

An argument can be made to use the coefficient estimates generated with the full sample final weight as $\hat{\theta}^*_{(.)}$, as opposed to the average of $\hat{\theta}^*_{(b)}$, which are the coefficient estimates generated from the repeated estimation of $\hat{\theta}$ using B bootstrap weights. The bswreg command uses the latter estimate.

Generally, bootstrap weights are generated by randomly drawing samples from each stratum of primary sampling units, with replacement; each sample drawn is equal in size to the number of units in the data set; and then the weight is assigned, using the same clustering and multi-stage sampling that is used to generate the final (design) weight, to each unit in the selected primary sampling unit, the weight is adjusted to reflect the probability of selection for the random sample. Further, observations or sampling units selected for the random sample receive a positive bootstrap weight and units not selected receive a weight of zero [Satin and Shastry, 1993]. This sampling is replicated many times in order to generate a set of bootstrap weights that is large enough to be consistent; the number of times this process is repeated equals the number of bootstrap samples. In equation 1 above there are B bootstrap samples. For example, in the National Population Health Survey, there are B=500 bootstrap samples.

Many surveys provide this final set of weights (B samples) for variance analysis. However, after calculating the bootstrap weight samples, some surveys take the additional step of averaging the bootstrap weights over C bootstrap samples. Modifying the variance estimator presented in equation 1, the mean bootstrap variance estimator is as follows:

$$v_{\bar{B}}\left(\hat{\theta}\right)=\frac{C}{B}\sum_b\left(\hat{\theta}^*_{(b)}-\hat{\theta}^*_{(.)}\right)^2 \qquad \text{where } \hat{\theta}^*_{(.)}=\left(1/B\right)\sum_b\hat{\theta}^*_{(b)} \tag{2}$$

Where each b^{th} mean bootstrap sample set of weights is equal to the means of C bootstrap weights. In this specification, the term $\hat{\theta}^*_{(b)}$ is obtained using the b^{th} mean bootstrap weight variable as opposed to the standard bootstrap weight variable used in equation 1 [Phillips, 2004 and Yeo et al., 1999].

For the standard bootstrap weight any single bootstrap replicate, which will include some zero weights, does not pose a confidentiality risk. However, when B is large all standard bootstrap replicates could be examined to identify the pattern of zero weights, and thereby identify cluster membership of observations or records. The mean bootstrap with non-zero

averages, comes from the practice of ensuring that at least one weight in C is non-zero [Yeo et al., 1999]. Since calculating average bootstrap weights in this way helps to mask cluster membership, this reduces the risk to contravening confidentiality.

For example, in the case of the WES data, the initial number of standard bootstrap weights samples is equal to B=5000. However, for confidentiality purposes, average bootstrap weights were derived. In the WES, the bootstrap weight samples were averaged over groups of C=50. Thus, each of the 100 mean bootstrap weights provided for the WES is an average bootstrap weight of 50 other bootstrap weights.

By inserting the integer C into the numerator of the variance estimator an adjustment is being made which re-introduces the variability that had been removed by using an average bootstrap weight. Thus, the C reflects the fact that the set of bootstrap weights are mean bootstrap weights that have been averaged over C iterations [Statistics Canada, 2003]. Further, the inclusion of the scalar C in the BSWREG revision also expands the breadth and functionality of the program. The revised variance estimator and program can be used to account for variants of the standard Balanced Repeated Replication method. Specifically, this can be used with surveys where only two primary sampling units are selected per stratum (for our PISA illustration that follows, the two PSUs per stratum are schools).

Researchers wishing to use achievement data from Programme for International Student Assessment (PISA) should also account for the added sampling variance that arises from the measurement error inherent in the use of plausible value achievement scales to arrive at a final (total) sampling variance estimator. The bswreg program is only useful when the user is not using achievement data. Refer to Lauzon [2004] for a discussion on the estimation of variance when plausible value achievement data, available in YITS/PISA, are used as dependent variables. Lauzon discusses in detail when the bootstrap should be used with PISA instead of the Balanced Repeated Replication (BRR), and he provides a Stata program for these applications.

An example of this is the Programme for International Student Assessment (PISA) survey and Fay's replicates, which can be used to compute unbiased-standard error estimates to accompany population estimates. In Fay's Balanced Repeated Replication method, T half samples are randomly drawn with replacement, similar to the procedure above, from each stratum, of primary sampling units; the sample drawn is equal in size to half the number of units in the data set; then the final weights are adjusted by multiplying the selected half by (2-K) and the other half by K, where K is a number between 0 and 1. For the PISA data K is equal to 0.5 [OECD, 2001]. The Fay's variance estimator is as follows:

$$v_{Fay}\left(\hat{\theta}\right)=\frac{1}{T(1-K)^2}\sum_t\left(\hat{\theta}^*_{(t)}-\hat{\theta}^*_{(.)}\right)^2 \qquad \text{where } \hat{\theta}^*_{(.)}=\left(1/T\right)\sum_t\hat{\theta}^*_{(t)} \qquad (3)$$

Thus, as discussed by Phillips [2004], the mean bootstrap and Fay's method can employ the same variance estimator. For example, in equation 2, C could be set equal to $C=(1-K)^{-2}$, to accommodate for Fay's Method. Using the PISA example, in equation 2, C is equal to 4. For a more detailed discussion see Phillips [2004] and OECD [2001]. Researchers will want to be

careful when using Fay's method with PISA data due to the measurement error inherent to the PV achievement data, as discussed above.

III. Revised features

The revised BSWREG Stata ado program has many useful additional features (see Appendix 2 for a complete list of options). These features include: the added possibility of accounting for mean bootstrap weights or other types of non-standard balance repeated replication techniques, this can be done by using the c*meanbs* option. This new option can be used to specify the number of bootstrap weight samples used to calculate an average bootstrap weight. In the case of WES, the bootstrap weight samples were averaged over groups of C=50, and as such the option c*meanbs* should be set equal to 50 (see example below).

The bootstrap count algorithm has been modified to notify users of the completion of the first few bootstrap repetitions so that researchers can verify the iterations are incrementally stepping forward and not "frozen". The new count may also help researchers better estimate an expected completion time. Also the display form has been changed to a fixed statistic display format/layout.

There were also several new results in e( ) that have been created for the bswreg e-class Stata command, these include: the e(numofbs) variable that is available after running bswreg and contains the number of bootstraps successfully run, the e(N) variable that contains the number of observations in the plain unbootstrapped regression, and the e(cmd) variable that contains "bswreg". All these are in addition to the coefficient and bootstrapped variance-covariance matrices: e(b) and e(V), that continue to be available. Use the "ereturn list" command to display other scalars, macros, matrices, and functions that are available with BSWREG.

In addition to the new features, listed above, bswreg now also works with additional regression commands including, but not limited to, commands like: reg, areg, qreg, intreg, ivreg, reg3, probit, biprobit, mlogit, heckprob, heckman, glm, cox, etc... The program now works with all regression commands that support weights. The "xt" series of commands that do not support weights cannot be run with bswreg. This revised version of the bswreg program also fixes complications arising from multiple equation estimation when complex Stata equation labels are used and fixes an error that occurred if the very first bootstrap weight estimation failed.

IV. How to – An example

The revised Stata program is as easy to use as the original bswreg program. Simply copy the "bswreg.ado" and "bswreg.hlp" files, which are described in Appendix 1, to your Stata ADO folder, (type the command "adopath" at the Stata command prompt for a list of ado directory paths in which to place this program), 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) [cmeanbs(integer)] [level(integer)] [bsci] [saving(path_and_filename[,replace])];

Underlines indicate short forms for the options. To illustrate the use of this syntax command, using the Workplace and Employee Survey 1999, suppose you wished to investigate the effect of location size (small, medium, and large), payroll per employee, percentage of workers covered at the location by a collective bargaining agreement, a flag to distinguish non-profit workplaces from those operating for profit, and in-house dedicated human resources personnel on the availability of individual incentive systems.

Individual incentive systems are one of the areas where the WES focuses its questions. The question: "Does your compensation system include the following incentives? [Including]…Individual incentive systems such as bonuses, piece rate, and commissions are systems that reward individuals on the basis of individual output or performance" [Statistics Canada, 2001]. This is a binary variable where the availability of incentives equals 1 and 0 otherwise. The existence of incentives and the factors that may affect their offering are the essence of this example.

In this example, plant size is determined by each workplace's total employment count. Locations with a total number of employees ranging between 0 and 100 are classified as small; between 101-500 as medium, and 501 or more are large. Traditionally, this is category classification used in the Canadian System of National Accounts. In all, three location size dummy variables are defined. Small workplaces are the most numerous group accounting for 98.2% of the population, followed by medium and large workplaces comprising 1.58% and 0.22%, respectively.

Payroll per employee is the average return per workplace to the workforce for labour and human capital services (payroll_per_person), and is calculated by dividing gross-payroll by total employment for each location.

The percentage of workers covered at the location by a collective bargaining agreement is picked-up by the union status variable (pct_union). It is presumed that the degree of unionization in a location may affect the incentive systems offered by workplaces.

The WES does not include the public sector; however, both private sector for-profit and non-profit workplaces are included (binary variable nonprft_flag, where non-profit is indicated by the variable equalling one). The locations not motivated by profit maximization are expected to have different emphasis placed on incentive systems.

The human resources variable "hr_in" attempts to get at whether or not there is a person dedicated to human resource activities at the workplace. The question is phrased as: "Which statement best describes the responsibility for human resource matters at this location?" and the responses are: "(1) there is a separate human resources unit in this workplace employing more than one person; (2) one full-time person in this workplace is responsible for human resources

matters; (3) human resources matters comprise part of one person's job in this workplace, such as owner or manager; (4) human resources matters for this workplace are the responsibility of a person or unit in another workplace; (5) human resources matters are handled as they arise in this workplace (i.e. are not assigned to one person in particular); or (6) Some other arrangement, specify;" where hr_in equals one when respondents selected 1, 2, or 3 and zero otherwise. Locations with in-house dedicated human resources may be more likely to have incentive systems in place.

The example is a logit regression of incentives on a list of size dummies, payroll per employee, unionization, profit motive, and dedicated human resources using WES workplace data 1999. To begin, ensure that your analytical data file and the appropriate bootstrap weight files have been merged correctly (use the appropriate unique identifier). The BSWREG program does not require the bootstrap weights to have any naming scheme. To get design-based standard errors, all 100 mean bootstrap weights will be used in this regression. The command to use these weights is as follows:

bswreg incentives medium large payroll_per_person pct_union nonprft_flag hr_in [pw=wkp_final_wt], cmd(logit) bsweights(wkp_bsw1-wkp_bsw100) cmeanbs(50) level(95); (4)

The results from this regress are as follows:

Output 1

```
. bswreg incentives medium large payroll_per_person pct_union nonprft_flag hr_in
> [pw=wkp_final_wt], cmd(logit) bsweights(wkp_bsw1-wkp_bsw100) cmeanbs(50) level(95) ;


1 bootstraps completed
2 bootstraps completed
3 bootstraps completed
4 bootstraps completed
5 bootstraps completed
25 bootstraps completed
50 bootstraps completed
100 bootstraps completed

Results from BSWREG
-------------------

* The confidence intervals below are based on the normal distribution

  +-------------------------------------------------------------------------------+
  |   Var_name        Coef       BSse     BSzstat   BSpvalue    BSilow95     BSiup95 |
  |-------------------------------------------------------------------------------|
  |     medium    1.038241   0.150151    6.914630   0.000000    0.743950    1.332533 |
  |      large    1.175536   0.243483    4.828008   0.000001    0.698319    1.652753 |
  | payroll_pe    0.000024   0.000004    5.803496   0.000000    0.000016    0.000032 |
  |  pct_union   -0.930827   0.300417   -3.098455   0.001945   -1.519633   -0.342022 |
  | nonprft_fl   -1.089882   0.237791   -4.583371   0.000005   -1.555943   -0.623821 |
  |-------------------------------------------------------------------------------|
  |      hr_in   -0.283383   0.149053   -1.901218   0.057274   -0.575522    0.008756 |
  |      _cons   -1.231138   0.168566   -7.303616   0.000000   -1.561520   -0.900755 |
  +-------------------------------------------------------------------------------+

Total bootstraps completed: 100
```

This is inference-appropriate output, because we have used the design-based bootstrap weights. All of our explanatory variables are statistically significant at the 95% level except the dedicated human resources variable (hr_in). Notice how this output differs from the bswreg output that does not adjust for the mean bootstrap using cmeanbs(50). In other words, how is our inference

effected if the cmeanbs(50) option is excluded for the WES data that uses mean bootstraps (see Output 2).

Output 2

```
. bswreg incentives medium large payroll_per_person pct_union nonprft_flag hr_in
> [pw=wkp_final_wt], cmd(logit) bsw(wkp_bsw*) l(95) ;


1 bootstraps completed
2 bootstraps completed
3 bootstraps completed
4 bootstraps completed
5 bootstraps completed
25 bootstraps completed
50 bootstraps completed
100 bootstraps completed

Results from BSWREG
-------------------

* The confidence intervals below are based on the normal distribution
```

Var_name	Coef	BSse	BSzstat	BSpvalue	BSilow95	BSiup95
medium	1.038241	0.021235	48.893818	0.000000	0.996622	1.079860
large	1.175536	0.034434	34.139172	0.000000	1.108047	1.243024
payroll_pe	0.000024	0.000001	41.036919	0.000000	0.000023	0.000025
pct_union	-0.930827	0.042485	-21.909389	0.000000	-1.014097	-0.847558
nonprft_fl	-1.089882	0.033629	-32.409325	0.000000	-1.155793	-1.023972
hr_in	-0.283383	0.021079	-13.443637	0.000000	-0.324698	-0.242068
_cons	-1.231138	0.023839	-51.644360	0.000000	-1.277861	-1.184415

```
Total bootstraps completed: 100
```

The above output is clearly problematic. Even though the coefficient estimates are the same, which they should be, the standard errors are substantially lower in Output 2 than they are in Output 1. This is because the scalar factor, where C=50, is being left out of equation 2 and thus the variances are being underestimated by a factor of C. In other words, the standard errors are being underestimated by a factor of $\sqrt{C}$ *or* $\sqrt{50}$. Thus, the above output leads to inappropriate inference. In Output 2, we are led to the conclusion that dedicated human resources is also statistically significant at the 95% level.

Notice in the Output 2 command above the bootstrap weight variable list is specified as "wkp_bsw*". Researchers may want to use the wild card or asterisk when specifying a list of variables that may not be in numerical order in the Stata data set being used. This will avoid a problem with Stata's built-in algorithm, which selects variables over the specified range from the order that they occur in the data set rather than the logical range implied by the boundaries. For example, if the boundaries are "bsw1-bsw100" and the first four (of one hundred) weights specified in the data set are bsw1, bsw10, bsw100, and bsw2, then stating the *varlist* as "bsw1-bsw100" in any Stata command will result in only the first three variables (weights) being selected (bsw1, bsw10, bsw100) as opposed the full range. While stating the *varlist* as "bsw*" implies the full range of 100 weights to be selected.

Output 3

```
> logit incentives medium large payroll_per_person pct_union nonprft_flag hr_in
> [pw=wkp_final_wt];


(sum of wgt is   7.1789e+05)

Iteration 0:   log pseudo-likelihood = -3817.6905
Iteration 1:   log pseudo-likelihood = -3635.8102
Iteration 2:   log pseudo-likelihood = -3631.4552
Iteration 3:   log pseudo-likelihood = -3631.4242
Iteration 4:   log pseudo-likelihood = -3631.4242

Logit estimates                                   Number of obs   =       6271
                                                  Wald chi2(6)    =     103.01
                                                  Prob > chi2     =     0.0000
Log pseudo-likelihood = -3631.4242                Pseudo R2       =     0.0488

------------------------------------------------------------------------------
             |               Robust
  incentives |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      medium |    1.038241   .1580988     6.57   0.000     .7283732    1.348109
       large |    1.175536   .2388543     4.92   0.000     .7073897    1.643681
payroll_pe~n |    .0000242   3.82e-06     6.33   0.000     .0000167    .0000317
   pct_union |   -.9308272   .2927079    -3.18   0.001    -1.504524   -.3571304
nonprft_flag |   -1.089882   .2453014    -4.44   0.000    -1.570664   -.6091007
       hr_in |    -.283383   .1420044    -2.00   0.046    -.5617065   -.0050594
       _cons |   -1.231138   .1660942    -7.41   0.000    -1.556676   -.9055991
------------------------------------------------------------------------------
```

It is also important to note that the logit regression with robust standard errors would also lead to incorrect inference, because it does not use the bootstrap weights at all. From the information in Output 3 it appears that all variables are significant at the 95% level, however the standard errors here are biased and do lead to inappropriate inference. The output generated by equation 4 (Output 1) contains design-based standard errors and associated p-values.

The program is not only useful for regression techniques, but can be used to calculate various summary statistics such as frequencies, means, and ratios. See Piérard et al [2004] for a discussion of limitations and for examples of how these statistics can be calculated.

V. Concluding remarks

This program focuses on design-based (and inference appropriate) variance estimation across various Statistics Canada social surveys. The program can now be used with any survey that has bootstrap weights, This includes a wide spectrum of datasets from the General Social Survey (GSS), 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 with some limitations, the Programme for International Student Assessment (PISA) and the Youth in Transition Survey (YITS) just to name a few.

The revisions to this program build on the available features and continue to provide researchers, who use Stata, with a flexible tool that is easy to use and accurate.

References

Lauzon, Darren. 2004. "Variance estimation with plausible value achievement data: Two STATA programs for use with the YITS/PISA data." The Research Data Centres Information and Technical Bulletin. (Spring) 1(1):37-62. Statistics Canada Catalogue no. 12-002-XIE.

Organization for Economic Co-operation and Development (OECD). 2001. "Manual for the PISA 2000 Database." Programme for International Student Assessment (PISA) 2000.

Phillips, Owen. 2004. "Using Bootstrap Weights with WesVar and SUDAAN." The Research Data Centres Information and Technical Bulletin. (Fall) 1(2):1-10. Statistics Canada Catalogue no. 12-002-XIE.

Piérard, Emmanuelle, Neil Buckley, and James Chowhan. 2004. "Bootstrapping Made Easy: A Stata ADO File." The Research Data Centres Information and Technical Bulletin. (Spring) 1(1):23-40. Statistics Canada Catalogue no. 12-002-XIE.

Satin, Alvin, and Wilma Shastry. (1993) Survey Sampling: A Non-Mathematical Guide. Ottawa: Ministry of Industry, Statistics Canada, Social Survey Methods Division. Catalogue No. 12-602-XPE.

Statistics Canada. 2003. "Guide to the Analysis of Workplace and Employee Survey 2001" Business and Labour Market Analysis Division & Labour Statistics Division. (June) Ottawa.

Statistics Canada. *1999 Workplace and Employee Survey*. Business and Labour Market Analysis Division and Labour Statistics Division. 4-4700-2.1: 1999-04-01. STC/LAB-075-75055. Ottawa: Statistics Canada.

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.

Appendix 1

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)

* March 1st, 2005 Buckley, Chowhan
*    Reset variable and equation labels since those with spaces were interfering with some regression
*       commands like ologit etc., if you are using Stata versions prior to 8.2 you will need to drop
spaces from variable labels before running bswreg
* October 21st, 2004 Buckley, Chowhan
*    BSWREG should now work with any regression command that accepts a weight
*       (including, but not limited to commands like: reg, qreg, intreg, ivreg, reg3, probit, biprobit,
heckprob, heckman, glm etc...)
*     fixed problem with running BSWREG with regression methods that analyze censored data containing
missing values (e.g. INTREG is now fully functional within BSWREG)
* September 30th, 2004 Buckley, Chowhan
*    added possibility of mean bootstrap weights
*    changed bootstrap count algorithm
*     created e(numofbs) variable that is available after running bswreg and contains the number of
bootstraps successfully run
*    created e(N) variable that contains number of observations in plain unbootstrapped regression
*    created e(cmd) variable that contains "bswreg"
*    fixed statistic display format/layout
* August 8th, 2003 Pierard, Buckley, Chowhan (original)

# delimit;
version 7.0;


syntax anything [aweight pweight fweight iweight] [if] [in], cmd(string) [cmdops(string)]
  BSWeights(varlist numeric) [Cmeanbs(integer 1)] [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' `anything' [`weight'`exp'] `if' `in', `cmdops';

if _rc ~= 0
   {;
   noi di in red " ";
   noi di in red "Error doing: `cmd' `anything' [`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 removes all variable and equation labels because they will cause problems if they
contain spaces (they will be put back later);
capture label language bswreg, new;
*The next line runs the wanted regression and we store the coefficients in a matrix for later use;
```

```
`cmd' `anything' [`weight'`exp'] `if' `in', `cmdops';
local _numofobs = e(N);
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 list `bhat';
matrix `bsVC'=e(V);
*The next line initializes the bootstrap coefficients matrix with the original sample weighted
coefficients to get the correct matrix dimensions, this first column will be removed later;
matrix `bsbetas'=(`bhat'');

*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',100)==0 | `_i'<6 | `_i'==25 | `_i'==50)
  {;
  noi di in green `_i' " bootstraps completed";
  };

 *Run the regression with the chosen set of bootstrap weights, only use the coefficients if there are
no errors;

 capture `cmd' `anything' [`weight'=`bswvar'] `if' `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'
        {;
        *Append the coefficients from the current bootstrap to the aggregate matrix;
        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;
 };
*Now we remove the initial column of coefficients that used the original sample weight;
matrix `bsbetas'=`bsbetas'[1...,2...];
*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;
*CmeanBS is the number of bootstrap weight samples used to calculate an average bootstrap weight
sample;
*When CmeanBS is not equal to 1 a mean bootstrap factor exists dependent on the survey, the default
value is 1;
forvalues _i = 1/`_k1'
 {;
 forvalues _j = 1/`_k1'
  {;
    correlate _xv`_i' _xv`_j', covariance;
    matrix `bsVC'[`_i',`_j'] = (((`_realboot'-1)*(`cmeanbs'))/`_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;

 * See above for description of CmeanBS;
 gen _sdx`_i'=sqrt((((`_realboot'-1)*(`cmeanbs'))/`_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;
* gen _p`_i'=2*ttail(`_realboot'-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 BSse;
label var _t "Bootstrap z-statistic";
```

```
ren _t BSzstat;
label var _p "Bootstrap p-value";
ren _p BSpvalue;
if "`bsci'"==""
 {;
 label var _low`level' "Bootstrap lower confidence interval assuming a normal distribution";
 label var _high`level' "Bootstrap upper confidence interval assuming a normal distribution";
 };
if "`bsci'"=="bsci"
 {;
 label var _low`level' "Bootstrap lower confidence interval using bootstrap sample distribution";
 label var _high`level' "Bootstrap upper confidence interval using bootstrap sample distribution";
 };
ren _low`level' BSilow`level';
ren _high`level' BSiup`level';

*Display RESULTS!;
noi display in green " ";
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 " ";
format Coef BSse BSzstat BSpvalue BSilow`level' BSiup`level' %10.6f;
format Var_name %10s;
noi list Var_name Coef BSse BSzstat BSpvalue BSilow`level' BSiup`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';
*This next line creates a e(numofbs) scalar available after running bswreg that contains the number of
boostraps run, di e(numofbs);
estimates scalar numofbs = `_realboot';
estimates scalar N = `_numofobs';
estimates local cmd = "bswreg";

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

*The next line removes all variable and equation labels because they will cause problems if they
contain spaces (they will be put back later);
capture label language default;

*Restore the original dataset
restore;

};
end;
```

BSWREG help file

```
{smcl}
{* 21October2004 Buckley/Chowhan}

{* 30September2004 Buckley/Chowhan}
{* 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:)} [{cmdab:c:meanbs(}{it:integer}{cmd:)}]
[{cmdab:l:evel(}{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. In order to avoid Stata variable ordering problems it might be
better to specify {cmd:bsweights(bsw*)} when using all weights.

{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:cmeanbs(}{it:integer}{cmd:)} specifies the number of bootstrap weight samples each mean
bootstrap
weight is averaged over, in the case of surveys that use mean boostrap weights. The default is
that the bootstraps privided are not mean bootstrap weights, {cmd:cmeanbs(1)}.

{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: BSse:}      This is the new standard error of the coefficient,
            calculated using bootstrap weights.
{inp: BSzstat:}   This is the new z-stat of the coefficient,
            calculated as the coefficient divided by the bootstrapped standard error.
{inp: BSpvalue:}  This is the new p-value of the coefficient,
            calculated using the z-statistic.
{inp: BSilow(level):} This is the lower (level)% confidence interval around the coefficient
            using the bootstrapped std. error.
{inp: BSiup(level):}  This is the upper (level)% confidence interval around the coefficient
            using the bootstrapped std. error.

{title: e-class results}
{inp: e(numofbs): Scalar} The number of successful bootstrap replications.
{inp: e(N): Scalar}       The number of observations in the underlying survey sample.
{inp: e(cmd): Macro}      Contains "bswreg".
{inp: e(b): Matrix}       This is the vector of coefficients.
{inp: e(V): Matrix}       This is the bootstrapped variance-covariance matrix.

{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)}
{p  8  12}{inp:.  bswreg  wesemployeeincentives  wesworksize  [aw=wt],  cmd(logit)  bsw(bsw1-bsw500)
cmeanbs(50)}
```

Appendix 2

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). In order to avoid Stata variable ordering problems it might be better to specify bsweights(bsw*) when using all weights.

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.

cmeanbs: specifies the number of bootstrap weight samples used to calculate an average bootstrap weight sample, mean bootstrap weight factors are dependent on the survey. The default is equal to 1, implying that the bootstrap weights are not mean bootstrap weights.

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

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

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.

Technical note

The household as a unit of analysis in the National Longitudinal Survey of Children and Youth

by Franck Larouche and Charles Tardif

The National Longitudinal Survey of Children and Youth (NLSCY) follows the development and well-being of a representative sample of Canadian children aged 0 to 11 years at Cycle 1 (1994-1995) through adulthood. Conducted every two years, this survey was designed to make the child the unit of analysis. The cross-sectional and longitudinal weights assigned to each record correspond to the unit of analysis, the child.

The children in the NLSCY sample are selected from a complex design in order to meet different needs, while taking into consideration certain operational constraints. Part of the sample is taken from the Labour Force Survey (LFS), and another part is taken from the birth registry. In addition to the initial sample at Cycle 1, a new sample of children between 0 and 1 years of age is selected at every subsequent cycle. The initial weight of the child at the time of sample selection corresponds broadly to the inverse of its probability of selection. This weight is then adjusted to take into account the total non-response during data collection, using specific characteristics of the child. Following adjustment for non-response, the weights are post-stratified by province, age and sex of the child so that they represent the known demographic totals by province, age and sex.

Certain information has been collected about households, but it is impossible to make general estimates for all Canadian households. All inferences must be based on the child. For example, we can estimate the number of children living in a household with one parent, but we cannot estimate the number of single-parent households.

Calculating the average weight for children in one household as a variable of household weight and using this average weight to generate estimates that could be representative of Canadian households, is not recommended. In fact, such weights would not be adjusted to be representative of all households. Any technique that adjusts the weight of the child to make a household or other weight, which does not take into account the necessary adjustments for non-response and post-stratification, is not desirable.

It should also be noted that the sampling strategy has been modified with each cycle. At Cycle 1, up to four children per household were selected. At Cycle 2, given the heavy response burden of households with several children, it was decided to reduce the maximum number of children per household to two. At Cycle 3, only children aged 0 years were selected from the LFS. Therefore, given a few exceptions (mainly twins), only one new child was selected per household. Children between 1 and 5 years of age selected at Cycle 3 were taken from the birth registry, with one child per household being selected. At Cycle 4, children aged 0 and 1 years were selected from the LFS. Therefore, it is possible that some households have two children selected (twins or households with two children less than 2 years of age), but in most cases, only

one child was selected. At Cycles 5 and 6, the sampling strategy was modified so that only one child per household would be selected. With regard to twins, only one was selected for this survey. The presence of more than one child per household is, therefore, decreasingly important.

Where applicable, although it seems strange to have two records from one household, it causes no problem. The weights were obtained to produce estimates of the population of Canadian children, and the variance calculated with the bootstrap weights will produce suitable estimates.

Finally, another problem with regard to the change of household arises. Because the purpose of the NLSCY is to follow the children, if a longitudinal child leaves his/her initial household following a divorce in the family or for other reasons, the child will be followed in his/her new household; the initial household is abandoned, so to speak. In addition, as the original cohort gets older, more and more young people are leaving their initial household, which increases this phenomenon with each cycle. The household changes that occur over time make the creation of household weights, or longitudinal household weights, even more complex.

In many cases, the analytical question can be reformulated to take this limitation of the NLSCY into account. Furthermore, if the analytical question cannot be reformulated to consider the child—either because it is not possible or because the very objective of the survey is to study households or another unit of analysis—the use of other, more suitable sources of data for this "type" of analysis should be considered. Otherwise, one of the options would be to not use weighting, in which case, no conclusion can be drawn for the entire population. Despite the fact that we do not recommend the use of average weights, this option is better than not using weights at all.

Information note

The CRISP-NLSCY Files

By Cara B. Fedick

Dr. J. Douglas Willms, and his staff at the Canadian Research Institute for Social Policy (CRISP) at the University of New Brunswick (Fredericton Campus), have developed a set of files for researchers interested in using Statistics Canada's National Longitudinal Survey of Children and Youth (NLSCY) data sets. *"The Files"* consist of SPSS data and syntax, which are intended to assist researchers in conducting more efficient longitudinal analyses, using NLSCY data.

The Files are a reconfiguration of the first four cycles (Cycle 1 1994/95, Cycle 2 1996/97, Cycle 3 1998/99 and Cycle 4 2000/01) of the NLSCY. *The Files* include variables, scales and measures derived by Dr. Willms for the analyses used in his published book Vulnerable Children: Findings from Canada's National Longitudinal Survey of Children and Youth.[1] These additional variables, known as the CRISP Variables, are being provided to researchers in an attempt to minimize the redundancy of variable re-coding among users running similar analyses. Users of *The Files* are also able to take advantage of the knowledge and expertise of seasoned researchers' analysis techniques, saving considerable time writing syntax. In this way, researchers can carry on with their analyses and utilize the richness of the NLSCY data to its fullest potential. Even users who have no interest in using the CRISP Variables may benefit from using *The Files* to efficiently merge the original component files (i.e. Primary, Secondary, Custody, etc.) together and to conduct accurate data-cleaning and manipulation procedures. The Files reconfigure NLSCY into a format better suited to longitudinal analysis, particularly with the program *Hierarchical Linear Modeling (HLM).*

The SPSS syntax components of *The Files* are provided to users as a means of documentation of the CRISP Variables. Users familiar with SPSS syntax can use these files to understand how the CRISP variables were created or, for more advanced users, to create similar variables using other datasets.

The Files consist of a set of SPSS data (.sav) and syntax (.sps) files, as well as a number of instructional and documentation files. The *V1.0 CRISP-NLSCY Files* folder contains two subfolders, 'Data' and 'Syntaxes and Documentation', as well as a .pdf document, '*The Files* User's Guide.pdf'.

- *Data*
  This folder contains four SPSS data (.sav) files: ALL_CYCLE1.sav, ALL_CYCLE2.sav, ALL_CYCLE3.sav and ALL_CYCLE4.sav. These files

[1] Willms, J. D. (Ed.). (2002). *Vulnerable Children: Findings from Canada's National Longitudinal Survey of Children and Youth*. Edmonton: University of Alberta Press and Human Resources Development Canada, Applied Research Branch.

each contain the cycle-specific NLSCY component data provided by Statistics Canada, plus a set of additional CRISP variables. For example, ALL_CYCLE1.sav contains the primary, secondary, self-complete and custody files for Cycle 1 of the NLSCY, as well as a collection of NLSCY Cycle 1 variables created by CRISP staff.

- *Syntaxes and Documentation*
  This folder contains several subfolders, all but one of which contain the syntaxes (.sps files) used to create the CRISP variables found in the datasets described above. The exception, 'Original StatCan Documentation', contains all available documentation on the NLSCY (i.e. Codebooks, User's Guides, Survey Instruments, etc.) issued by Statistics Canada. The folders containing the syntaxes are labeled according to the variable or the variable set, and each contains four syntaxes, one for each NLSCY survey cycle.

- *The Files User's Guide.pdf*
  This document is the main source of reference for users using *The Files*. It contains information on the structure and preparation of *The Files* as well as instructions on how and why to use them effectively.

The SPSS data components of *The Files* should be treated like any other dataset housed in an RDC, as the contents of *The Files* data are derived from the protected NLSCY files. Like all other RDC output, analyses conducted with these data must be submitted for disclosure requests before they are permitted to leave the RDC. However, the SPSS syntax components and *The Files User's Guide.pdf* are free to leave the RDC.

Updated versions of *The Files* will be distributed to all RDCs (in CD form) at regular intervals as new releases of the data become available and modifications to *The Files* are made. Users will be advised and instructed about updates directly from CRISP.

Most current approved users of *The Files* are members of the Canadian Institute for Advanced Research's (CIAR) New Investigators Network (NIN), established in 2003 as a long-term group of young research leaders in the area of human development, to promote research based on the NLSCY. In order to gain access to *The Files*, these users first applied to the Social Sciences and Humanities Research Council (SSHRC) to obtain access to the NLSCY in the RDCs. Upon approval, these users were required to contact CRISP and agree to a series of rules and regulations governing their use of *The Files*. These users, and all others approved in the future, are approved for access to *The Files* in any of the RDCs across the country.

Subsequent to the release of *The Files*, potential users interested in gaining access to *The Files* should be advised to first apply to SSHRC to gain access to the RDCs, and once access is granted, contact CRISP via email (CRISPFILES@email.unb.ca) indicating their name, affiliation and a general description of their need for access to *The Files*. CRISP will continue to control official permission and handle the necessary procedures for potential users to gain access to *The Files* and will notify RDC analysts of additions and modifications to the list of approved users.

Individual RDC analysts will be responsible for ensuring that only approved users have access to *The Files* within their RDC.

For more information about the CRISP-NLSCY Files, please feel free to contact the Canadian Research Institute for Social Policy (CRISP) at the University of New Brunswick via email at CRISPFILES@email.unb.ca and visit http://www.unbcrisp.ca/learningbar/ for more information on the project for which the CRISP-NLSCY Files were originally developed.

Instructions for authors

The Information and Technical Bulletin will accept submissions for articles that address methodological or technical topics related to the datasets that are available at the Research Data Centres.

Language of material:

Manuscripts may be submitted in English or French. Accepted submissions will be translated into both official languages for publication.

Length of submissions:

The maximum length of submitted articles should not exceed 20 pages, double-spaced, excluding programs and appendices. In addition to in-depth explanations of technical issues, the bulletin also accepts short (3 page) submissions that provide quick solutions to analytical problems and commentary from fellow researchers about material previously released in the bulletin.

File formats and layout of text:

Manuscripts must be submitted in Microsoft Word (.doc) and may be sent by regular mail on a disk or CD or by email.

Manuscripts must have a cover page showing the names of the authors, their primary institution of affiliation, and the contact information (telephone number, mailing address and e-mail address) of the lead author.

Manuscripts must be prepared in 12pt Times New Roman, double-spaced, with 1-inch (2.5 cm) margins.

Titles should have sentence-case capitalization (e.g., Bootstapping made easy…).

Boldface type should only be used for headings. Underlining and italics are not to be used for headings.

Footnotes and references should be single-spaced and formatted according to *The Canadian Style: A Guide to Writing and Editing.*

File formats and layout of tables and charts

Tables and charts must be submitted in Microsoft Excel worksheets (.xls) or in comma-separated value (.csv) format. Each file must be clearly named table1, chart6, etc.

Tables and charts may be sent by regular mail on a disk or CD, or by e-mail.

Do not insert tables or charts into the text, but indicate their location in the text by inserting the title, followed by the filename in parentheses, e.g.,

Chart 6. Chocolate consumption by children, Canada, 2000 (chart6)

Mathematical expressions

All mathematical expressions should be set out separate from paragraph text. Equations must be numbered, with the number appearing to the right of the equation flush with the margin.

Style guide

Please follow *The Canadian Style: A Guide to Writing and Editing.* It is available for purchase by contacting Government of Canada Publications, Public Works and Government Services Canada.

Address for submission

Manuscripts and all correspondence relating to the contents of the Bulletin should be sent to the Editorial Committee

- by email to rdc-cdr@statcan.ca

The review process

The editorial committee conducts the initial article review process. Editors may solicit past authors of the Bulletin or subject matter experts to participate in the process. The articles submitted to the Bulletin are reviewed for accuracy, consistency, and quality.

Upon completion of the initial review, the articles undergo both peer and institutional review. Peer reviews are conducted in accordance with Statistics Canada's Policy on the Review of Information Products. Institutional reviews are be conducted by members of senior management within Statistics Canada, in order to ensure that the material does not compromise the Agency's guidelines of standards, or reputation for non-partisanship, objectivity and neutrality.

For more information about the review process, please contact the Editorial Committee at the address above.