Catalogue no. 12-002-XIE

The Research Data Centres Information and Technical Bulletin

Fall 2004, vol.1 no. 2

Centres de données de recherche
Research Data Centres
CDR RDC

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 Web site.

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**
Web site	**www.statcan.ca**

Ordering and subscription information

This product, Catalogue no. 12-002-XIE, is available on Internet free. Users can obtain single issues at http://www.statcan.ca/cgi-bin/downpub/freepub.cgi.

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 which 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.

Statistics Canada
Research Data Centres Program

The Research Data Centres Information and Technical Bulletin

Fall 2004, vol.1 no. 2

Published by authority of the Minister responsible for Statistics Canada

October 2004

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.

Editor: James Chowhan
Associate editors: Heather Hobson, Leslie-Anne Keown, Darren Lauzon

Table of contents

Articles

Owen Phillips, Using bootstrap weights with Wes Var and SUDAAN 6

James Chowhan, Stat/Transfer's command files and the efficient transfer of data files 16

Technical note

Franck Larouche, A note on identifiers in the National Longitudinal Survey of Children and Youth 22

Editorial committee, Instructions for authors 23

Using bootstrap weights with Wes Var and SUDAAN

By Owen Phillips

Abstract

For the purpose of design-based variance estimation, a number of Statistics Canada surveys supply bootstrap weights with their microdata. While the use of bootstrap weights is not explicitly supported by commercially available software such as SUDAAN and WesVar, by taking advantage of similarities between a commonly used bootstrap technique and the method of Balanced Repeated Replication (BRR), these software can be used to produce bootstrap variance estimates. This article examines the reasoning behind this, and shows, by way of example, how this might be accomplished. The paper concludes with a brief discussion of other design-based approaches to variance estimation as well as software, programs and procedures where these methods have been employed.

Introduction

A bootstrap approach to design-based variance estimation is used increasingly in the survey sampling community. Several Statistics Canada surveys—Survey of Labour and Income Dynamics (SLID), National Population Health Survey (NPHS), and the General Social Survey (GSS), to name but a few—all provide bootstrap weights, or variants thereof, with their microdata for the purpose of variance estimation.

The (survey) bootstrap belongs to a family of variance estimation techniques known as replicate based variance estimation. A detailed discussion of replication methods can be found in Lohr (1999), Rust and Rao (1996) or Wolter (1985). Such methods use the existing sample to build ‘synthetic’ samples, called replicates. Balanced Repeated Replication (BRR) is another such method, and has been implemented in commercially available software such as SUDAAN and WesVar. While the bootstrap and BRR differ in the way in which the replicates are built, bootstrap weights can be used to produce bootstrap variance estimates in software that will accommodate BRR weights, a point that the software documentation fails to mention in great detail.

The following sections will elaborate on the differences and similarities between the bootstrap and BRR, and will, by way of example, show how to use bootstrap weights in SUDAAN and WesVar. A variant of the bootstrap employed by the GSS and the Workplace and Employee Survey (WES) known as the mean bootstrap will be contrasted against Fay’s variant of BRR. The paper will conclude with a brief discussion of other design-based variance estimation techniques and the software and programs that incorporate the many techniques discussed.

For simplicity, this paper presents a very general discussion of the process of producing survey weights and ignores many of its complexities like non-response adjustments and post-stratification. However, it is assumed that the reader is familiar with basic concepts of survey sampling. For those wishing more information on the sampling process, please refer to *Survey* *methods and practices* (Statistics Canada 2003). Some familiarity with SAS, SUDAAN and/or WesVar is also assumed.

II. Bootstrap methods

Many Statistics Canada surveys, including SLID, NPHS, the National Longitudinal Survey of Children and Youth (NLSCY), the Canadian Community Health Survey (CCHS), the Ethnic Diversity Survey (EDS) and the Youth in Transition Survey (YITS), are using a bootstrap method to estimate sampling error. Without going into too much detail, bootstrap replicates are generated by randomly choosing, with replacement, a sample of primary sampling units (PSUs) within each stratum and adjusting the original sampling weights of the units in the selected PSUs to reflect the probability of selection into the subsample. If a unit does not appear in the bootstrap replicate, its bootstrap weight variable is set to zero. This process of selecting samples and reweighting is repeated *B* times to arrive at *B* bootstrap samples, *B* bootstrap weight variables and consequently *B* bootstrap estimates.

The variance of the estimate $\hat{\theta}$ of the finite population parameter θ of interest—for example a regression coefficient, population mean, ratio of two totals, etc.—is estimated by

$$\hat{V}_{BOOT}(\hat{\theta}) = \frac{1}{B}\sum_{b=1}^{B}\left(\hat{\theta}_b - \hat{\theta}\right)^2 \qquad (1)$$

where $\hat{\theta}$ is obtained using the full-sample weight variable and the estimates $\hat{\theta}_b$, b=1,…,B are obtained in exactly the same manner using the bootstrap weight variables[1,2].

A variant of the bootstrap, called the ***mean bootstrap,*** is used by GSS and WES. This method was originally proposed to address confidentiality issues arising from the release of bootstrap weights with public use microdata (See Yung 1997). Ultimately, it amounts to calculating the bootstrap weights as above and then averaging the bootstrap weights over *C* bootstrap samples. For example, in certain cycles of the GSS, 5000 bootstrap weight variables were produced. These weights were then averaged in groups of size *C*=25 to obtain the *B*=200 mean bootstrap weights that accompany the microdata. Similarly, WES provides 100 mean bootstrap weights, each of which is the mean of *C*=50 bootstrap weights.

The mean bootstrap variance estimator is given by:

[1] The sampling weight reflects the probability of selection of a unit to the full sample: it can be thought of as the number of units in the survey population represented by the sampled unit. The sampling weight is used to estimate the parameter of interest. The bootstrap weight is used for the purpose of estimating the sampling error associated with the parameter of interest. Like the sampling weight, a bootstrap weight might be thought of as the number of individuals in the survey population represented by a unit in the reduced (bootstrap) sample.

[2] Research is ongoing into approaches for obtaining estimates $\hat{\theta}_b$ from the bootstrap samples, other than by mirroring the approach used to estimate $\hat{\theta}$ from the full sample. These approaches may provide more stable variance estimates for some situations (see Roberts *et al,* 2003).

$$\hat{V}_{MBOOT}(\hat{\theta}) = \frac{C}{B}\sum_{b=1}^{B}\left(\hat{\theta}_b - \hat{\theta}\right)^2 \tag{2}$$

where $\hat{\theta}_b$ is obtained using the bth mean bootstrap weight variable.

III. Balanced repeated replication

Balanced Repeated Replication (BRR) is applicable to survey designs where two and only two PSU are selected per stratum. As many Statistics Canada surveys sample more than two units at the first stage, BRR cannot be used as a variance estimation method for those surveys. However, certain features of the BRR method allow us to use software intended for BRR variance estimation to obtain bootstrap variance estimation.

The BRR method consists of generating half-samples by selecting one PSU in each stratum. In a prescribed way[3], a 'balanced' set of *G* half samples is selected; in each of these, the sampling weights of units within the selected PSUs are multiplied by 2, while units in non-selected PSUs are given a weight of 0. The half sample estimates $\hat{\theta}_g$ are then used to compute

$$\hat{V}_{BRR}(\hat{\theta}) = \frac{1}{G}\sum_{g=1}^{G}\left(\hat{\theta}_g - \hat{\theta}\right)^2 \tag{3}$$

A possible variant of BRR is ***Fay's method***. The methodology used for selecting the half-samples is the same as for BRR; however, the weighting is done differently: the weights of units in the selected PSUs are multiplied by a factor (2-*K*); the weights of units in 'non-selected' PSUs are multiplied by *K*, where *K* is a fixed constant in the interval [0,1). In this way, all observed units contribute to each estimate $\hat{\theta}_g$, which is particularly useful when working with small domains. Variances are then estimated by

$$\hat{V}_{FAY}(\hat{\theta}) = \frac{1}{G(1-K)^2}\sum_{g=1}^{G}\left(\hat{\theta}_g - \hat{\theta}\right)^2 \tag{4}$$

The Programme for International Student Assessment (PISA) uses Fay's method, with *K*=0.5.

IV. What the software documentation doesn't tell you

As we have seen in Sections II and III, the methodologies for creating bootstrap and BRR weights are different. That being said, the form of the bootstrap variance estimator is the same as that for BRR (i.e. (1) and (3) are equivalent provided that *G*=*B*). In other words, if the bootstrap weights are provided, but designated to be BRR weights, software such as SUDAAN and WesVar that allow BRR variance estimation will calculate bootstrap variance estimates

[3] The methodology of producing the balanced half samples is not necessary for this discussion. The interested reader may consult Section 9.3.1 of Lohr (1999).

appropriately. Similarly for the mean bootstrap and Fay's method, by setting $K = 1 - C^{-\frac{1}{2}}$ in (4), (2) and (4) are equivalent, and software that will accommodate replicate weights calculated using Fay's method can also be used to calculate variances using mean bootstrap weights.

Using bootstrap weights in SUDAAN

Specification of the variance estimation method to be used by SUDAAN is done in the call to a particular analytic procedure. The following process is the same for all SUDAAN procedures that allow for BRR variance estimation[4]:

- The bootstrap is implemented in SUDAAN by specifying DESIGN=BRR.
- The REPWGT statement is used to indicate the names of the variables containing the bootstrap weights.
- The WEIGHT statement is not mandatory, but should be used when the variable containing the final weight is available[5].
- For surveys that provide mean bootstrap weights, set option ADJFAY=*C* (note that, at the time of the writing of this article, for GSS, *C*=25 and for WES, *C*=50).

Using bootstrap weights in WesVar

In WesVar, the variance estimation method is specified when creating a new WesVar data file. The resulting file is then used to define workbooks where table and regression requests are carried out. To define a WesVar data file with bootstrap or mean bootstrap weights:

- Move the replicate weight variables to *Replicates* box.
- Move the final weight variable to the *Full sample* box
- For bootstrap, specify the *Method* as BRR
- For mean bootstrap, specify the *Method* as Fay and specify $Fay\_K = 1 - C^{-\frac{1}{2}}$
- Move analysis variables to the *Variables* box, a unique identifier to the *ID* box (optional) and save the file.

V. Examples of using WesVar and SUDAAN

The following examples are applications of the information given in Section IV. The particulars of defining new variables and manipulating the data into a format suitable for use with the software are, for the most part, ignored. The goal of these examples is simply to show how the bootstrap may be implemented in the two software packages, and thus ignores the interpretation of the resulting output. The details of reading and interpreting output from SUDAAN and WesVar are found in the respective software user guides (see RTI, 2001 and Westat, 2002).

[4] In SUDAAN Release 8.02 and earlier, DESIGN=BRR cannot be specified for PROC SURVIVAL.

[5] In the absence of the weight statement, SUDAAN uses $\hat{\hat{\theta}}_{(b)} = \frac{1}{B}\sum_{b}\hat{\theta}_b$ in place of $\hat{\theta}$.

SLID bootstrap example

The following example examines the transition from low earnings for longitudinal individuals from 1996 to 1998 using Panel 1 on SLID. Given the selected years, either Panel 1, Panel 2, or the combination of the two might have been used. The analysis variable *low96* takes on a value of 1, if in 1996 the longitudinal person's earnings at their main job fell below some prescribed level, and zero otherwise. The variable *low98* follows a similar definition. Data are stored in a SAS dataset named *mobility*. The bootstrap weights *bs1-bs1000* corresponding to the Panel 1 longitudinal weight *ilgwt26* have been merged to the analysis file.

(a) SUDAAN

PROC CROSSTAB is used in SAS-callable SUDAAN to produce estimates of those who have moved into or out of low earnings between the two survey occasions. Using the information above along with the instructions provided in Section IV, PROC CROSSTAB would be set up as follows to make use of the bootstrap weights:

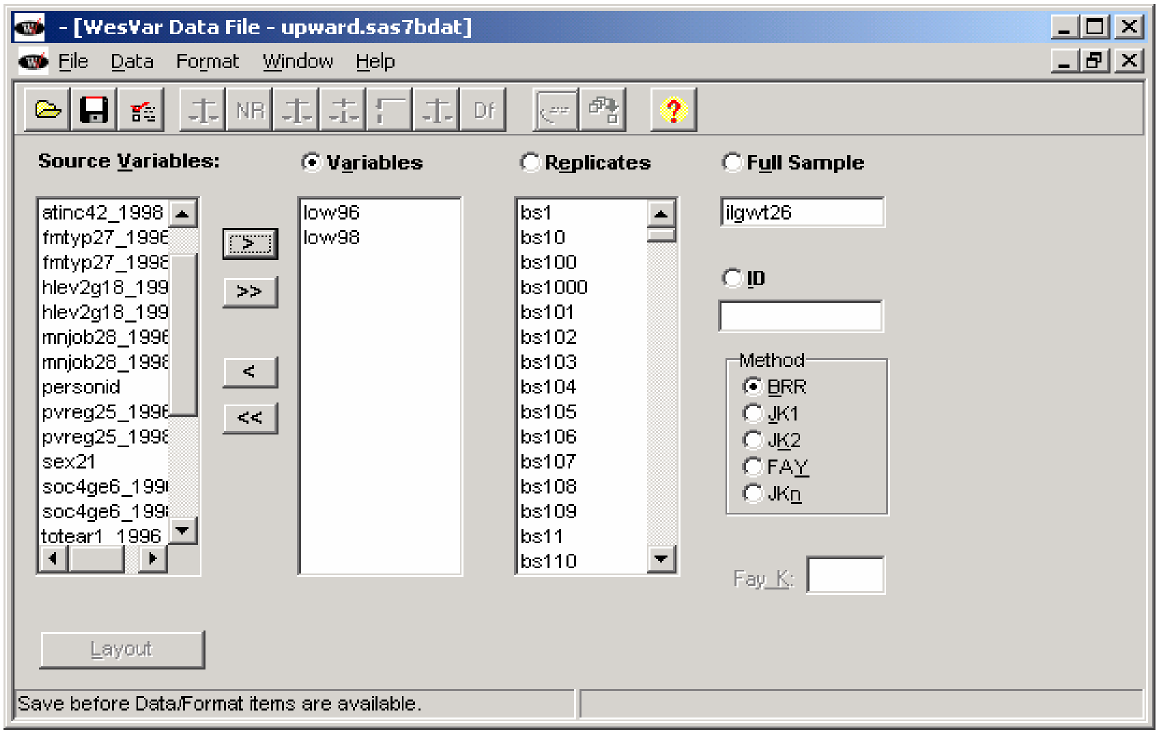
```
proc crosstab data=mobility design=BRR;
  weight ilgwt26;
  repwgt bs1-bs1000;
  recode low96=(0 1) low98=(0 1);
  subgroup low96 low98;
  levels 2 2;
  tables low96*low98;
run;
```

For more information on PROC CROSSTAB and SUDAAN, please refer to the user's guide (Research Triangle Institute, 2001).

(b) WesVar

Following the instructions in Section IV and given the same SAS datafile, a WesVar datafile would be defined as in Figure 1. The resulting file would then be saved and used to define the desired tables in a WesVar workbook. Instructions for creating a workbook and additional information on WesVar can be found in the user's guide (Westat 2002).

Figure 1: SLID bootstrap example using WesVar



- [WesVar Data File - upward.sas7bdat]

Source Variables:
atinc42_1998
fmtyp27_1996
fmtyp27_1998
hlev2g18_199
hlev2g18_199
mnjob28_1996
mnjob28_1998
personid
pvreg25_1996
pvreg25_1998
sex21
soc4ge6_199
soc4ge6_199
totear1_1996

● Variables
low96
low98

○ Replicates
bs1
bs10
bs100
bs1000
bs101
bs102
bs103
bs104
bs105
bs106
bs107
bs108
bs109
bs11
bs110

○ Full Sample
ilgwt26

○ ID

Method
● BRR
○ JK1
○ JK2
○ FAY
○ JKn

Fay_K:

Save before Data/Format items are available.

GSS mean bootstrap example

In this example, GSS Cycle 14 data are used in a logistic regression to examine the association among various demographic and socio-economic factors and the probability of internet use. The dependent variable is *netuse* (equal to 1 if the responding individual uses the internet; and 0 if not). All independent variables used in the model are categorical. The final weight variable on the *internet* SAS datafile is *fwgt*, with corresponding mean bootstrap weights *bsw1-bsw200*. Recall that for GSS, *C*=25.

(a) SUDAAN

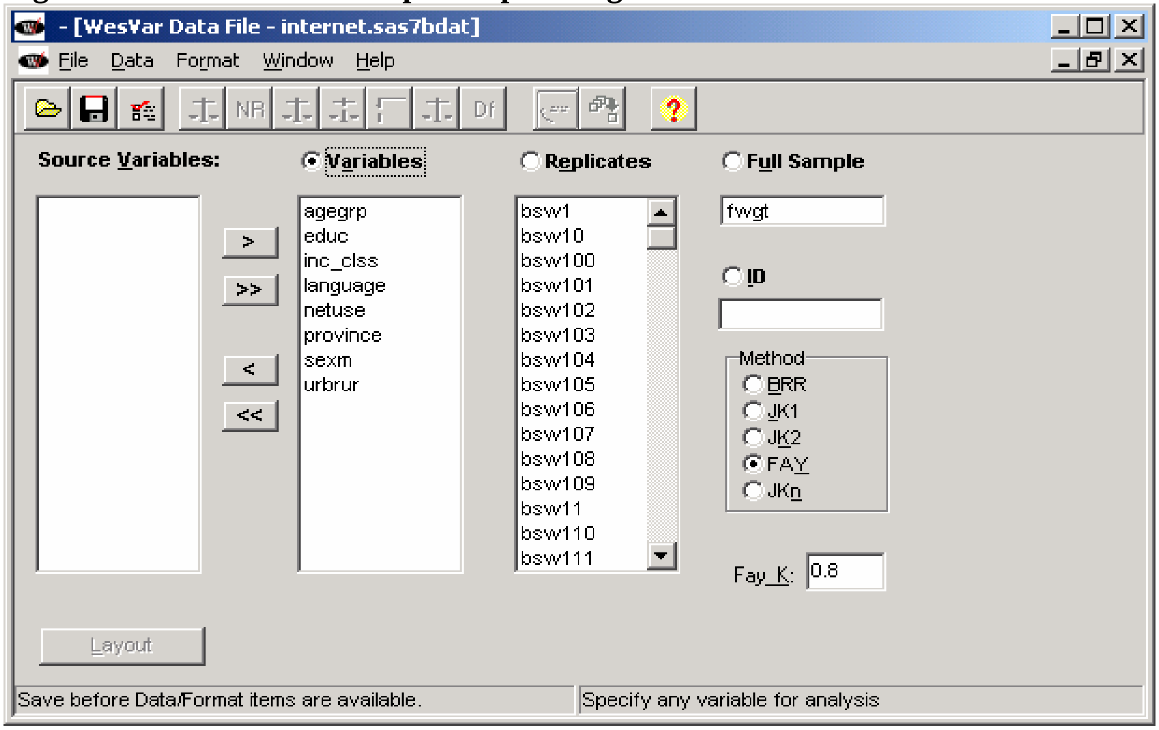
The following code shows how to set up PROC RLOGIST in SAS-callable SUDAAN given the above information and instructions in Section IV:

```
proc rlogist data=internet design=BRR;
  weight fwgt;
  repwgt bsw1-bsw200 / adjfay=25;
  subpopn province=59;
  subgroup sexm agegrp educ urbrur inc_clss language;
  levels 2 3 4 5 5 3;
  model netuse=sexm agegrp educ urbrur inc_clss language;
run;
```

(b) WesVar

Figure 2 shows how the same information is used to define a WesVar datafile. For GSS, $Fay\_K = 1 - 25^{-\frac{1}{2}} = 0.8$. As before, the resulting file would be saved and used to define the regression model in a WesVar workbook.

Figure 2: GSS mean bootstrap example using WesVar



- [WesVar Data File - internet.sas7bdat]

Source Variables:

● Variables: agegrp, educ, inc_clss, language, netuse, province, sexm, urbrur

○ Replicates: bsw1, bsw10, bsw100, bsw101, bsw102, bsw103, bsw104, bsw105, bsw106, bsw107, bsw108, bsw109, bsw11, bsw110, bsw111

○ Full Sample: fwgt

○ ID

Method: ○ BRR ○ JK1 ○ JK2 ● FAY ○ JKn

Fay_K: 0.8

Save before Data/Format items are available.

Specify any variable for analysis

VI. Other approaches and software for design-based variance estimation

The bootstrap and BRR are not the only replication methods for obtaining design-based estimates of variance. The delete-1 jackknife, another replication method, involves deleting from the existing sample a single PSU and reweighting the remaining PSUs in the same stratum to account for the loss of sample and other weight adjustments. Each PSU is deleted once and only once, so that there are as many jackknife replicates, and consequently jackknife weights, as there are PSUs in the sample. There are other possible jackknife variants that are not described here.

Taylor series approximations, or linearization methods, are another approach to variance estimation. This is the approach implemented by SPSS and SAS in their specialized complex survey procedures. Non-linear parameters of interest (such as ratios and regression coefficients) are expressed as smooth, linear functions of simple statistics like means and totals for which an analytical formula for the form of the variance estimator is known. The desired parameter is then approximated through the first-order Taylor series expansion of this function about the true value for the parameter of interest, and the sampling error can thus be approximated. This approach is

]

not suitable for statistics that cannot be well approximated by a linear function: chi-squares, for example.

Stratum and PSU identifiers must be supplied with the microdata in order to implement linearization methods or to implement a jackknife method where jackknife weights have not been provided. Additionally, software such as SUDAAN, WesVar and Stata are unable to account for the impact of all of the weight adjustments, such as non-response, post-stratification and other, when using Taylor linearization or when creating jackknife weights. That being said, for surveys that do not provide bootstrap weights, or in instances where the desired analysis cannot make use of bootstrap weights (e.g. PROC SURVIVAL in SUDAAN 8.02), it is recommended that one of these methods be used.

Chapter 9 of Lohr (1999) provides a more detailed overview of the replication and linearization methods discussed in this paper.

Commercially available products such as SUDAAN, WesVar, Stata and SAS, provide analytic procedures and commands capable of selected design-based analysis. Additionally, any software that offers an analytic procedure or command that can produce weighted estimates of the parameters of interest and also has the flexibility of a programming language, may be used recursively to obtain bootstrap variance estimates. Based upon this principle, SAS and SPSS macros have been constructed by Statistics Canada methodologists and are packaged together and provided with survey microdata (NPHS's and CCHS's *Bootvar*[6] and the *NLSCY Variance Estimation System* (*VES*), for example). A similar SAS-based program, *Bootmac* (written by an independent researcher), is available in the Research Data Centres (RDC). The user-defined Stata command *Bswreg*, can also be used to obtain bootstrap variance estimates for many of Stata's existing regression commands[7]. The benefits of this program were explained and exploited in the last issue of this bulletin (see Piérard *et al*, 2004).

The table in the Appendix compares the capabilities of many of the software and programs mentioned above for producing design-based variance estimates. It identifies the methods of variance estimation supported by the software, and the analytic procedures available to the user.

[6] A generic version of the Bootvar program is being produced to satisfy the need for a variance estimation tool for a number of surveys providing bootstrap weights with their microdata. This tool should be widely available in the Fall of 2004. Eventually, its capabilities will be expanded to include, for example, the estimation of sampling errors for quantiles and design-based tests of independence and homogeneity.

[7] Bswreg will not handle STATA regression commands that involve more than one line of code in order to implement the regression procedure. For example, it will not handle the Cox proportional hazards model.

References

Lohr, S. 1999. *Sampling: Design and Analysis*. Duxbury Press, USA.

Piérard, E., Buckley, N., Chowhan, J. Bootstrapping made easy: A Stata ADO file. *The Research Data Centres Information and Technical Bulletin* 1(1): 20-36.

Research Triangle Institute. 2001. *SUDAAN User's Manual, Release 8.0*. Research Triangle Institute, Research Triangle Park, NC.

Roberts, G., Binder, D., Kovacevic, M., Pantel, M., Phillips, O. 2003. Using an estimating function bootstrap approach for obtaining variance estimates when modelling complex health survey data. *Proceedings of the Survey Methods Section*, Statistical Society of Canada.

Rust, K.F., Rao, J.N.K. 1996. Variance estimation for complex surveys using replication techniques. *Statistical Methods in Medical Research* 5: 283-310.

Statistics Canada. 2003. *Survey Methods and Practices*, 12-587-XPE.

Westat. 2002. *WesVar 4.2 User's Guide*. Westat, USA.

Wolter, K.M. 1985. *Introduction to Variance Estimation*. Springer-Verlag, New York.

Yung, W. 1997. Variance estimation for public use microdata files. *Proceedings of Statistics Canada Symposium 97*, 91-95.

Appendix: Design-based analysis tools available in selected software

Software	**SUDAAN 8.02**	**WesVar 4.2**	**Stata 8.0**	**SAS 8.2**	**SAS 9.1**	**Bootvar and NLSCY VES**	**Bootmac**
Variance estimation approaches	**BRR (Bootstrap)** **Jackknife** **Taylor Series**	**BRR (Bootstrap)** **Jackknife**	**Taylor Series**	**Taylor Series**	**Taylor Series**	**Bootstrap**	**Bootstrap**
Modelling							
linear regression	*proc regress*	yes	*svyreg*	*proc surveyreg*	*proc surveyreg*	yes	yes
instrumental variable regression	no	no	*svyireg*	no	no	no	no
interval regression	no	no	*svyintrg*	no	no	no	yes
logistic regression	*proc logistic* (*rlogist*)	yes	*svylogit*	no	*proc surveylogistic*	yes	yes
probit regression	no	no	*svyprobt*	no	*proc surveylogistic*	no	yes
generalized logit models	*proc mulitlog*	yes	*svymlog*	no	*proc surveylogistic*	no	no
proportional odds models	*proc multilog*	no	*svyolog*	no	*proc surveylogistic*	no	yes
ordered probit regression	no	no	*svyoprob*	no	*proc surveylogistic*	no	yes
poisson and log-linear regression	*proc loglink*	no	*svypois*	no	no	no	yes
Heckman models	no	no	*svyheck*	no	no	no	no
proportional hazards models	*proc survival**	no	no	no	no	no	yes
Descriptive							
means	*proc descript*	yes	*svymean*	*proc surveymeans*	*proc surveymeans*	yes	yes
totals	*proc descript*	yes	*svytotal*	*proc surveymeans*	*proc surveymeans*	yes	yes
proportions	*proc descript*	yes	*svyprop*	no	no	yes	yes
ratios	*proc ratio*	yes	*svyratio*	no	no	yes	yes
tests of independence	*proc crosstab*	yes	*svytab*	no	*proc surveyfreq*	no	yes
quantiles	*proc descript*	yes	no	no	no	no	yes
Plausible values	no	yes**	no	no	no	no	no

* Taylor Series only

** For descriptive statistics and linear regression

Stat/Transfer's command files and the efficient transfer of data files

By James Chowhan

Abstract

The use of command files in Stat/Transfer can expedite the transfer of several data sets in an efficient replicable manner. This note outlines a simple step-by-step method for creating command files and provides sample code as an example.

Introduction

This note attempts to make the use of Stat/Transfer command files accessible to researchers. Most researchers use Stat/Transfer's "point and click" interface with relative ease, however, for researchers that have a need to transfer numerous data sets this "point and click" process can be repetitive and tedious; it requires one to be present between each file transfer. The use of command files can expedite the transfer of many data sets in an efficient replicable manner. Stat/Transfer command files can accomplish anything that the point and click methods can while retaining a record through the command file and log.

Batching many transfers or parsing tasks are where the true benefits and efficiencies of Stat/Transfer command files are realized. Researchers can submit a command file with many tasks and let it run while they move on to other tasks.

All examples in this note will be from Version 7.0.02 of Stat/Transfer but the method should work the same regardless of the version being used.[1] Further, this note will only focus on command files and will not discuss interactive Stat/Transfer use in a DOS environment (DOS prompt commands) or the point and click method; refer to Keown [2004] for a discussion of the point and click interface.

II. Stat/Transfer uses

Research Data Centre (RDC) data sets are often quite large both in length and width. Stat/Transfer can be used to reduce a data set's number of variables and/or number of observations, thereby making it more compact and easier to manage. A further use, and perhaps more primary, is the transfer of a data set from one format to another. For example, the conversion of a SAS data set to a SPSS or Stata file. The use of syntax and command driven files is very helpful in keeping logs of past and present work and transformations, as well as replicating previous work on new, the same, or different datasets.

[1] The examples in this note were tested on both versions 6 and 7 of Stat/Transfer.

III. Creating a command file

Stat/Transfer command files are very similar to text (notepad) or ASCII documents, in that text can be easily manipulated (cut and paste). However, Stat/Transfer and ASCII files are not identical, there is a key difference that Stat/Transfer can only recognize command files with a ".stc" extension. This is why lengths have been taken below, in the creation of a command file (specifically step 1), to clarify the necessity of creating a ".stc" file that Stat/Transfer can recognize. Assigning the ".stc" extension is the main reason that a text editor is required; its use is necessary to create this special file.

A command file will take the following general form:

```
log using "c:\path\log_name"
keep [variable list]
where [variable expression     relational operator     condition]
copy  "infilename.ex1"   "outfilename.ex2"    /x1   /x2  ...
quit
```

Commands can be typed into the file and comments can be inserted by preceding the comments with "//" a double forward slash. All drive, paths, and file names should be contained in quotations; this allows spaces and other non-alpha-numeric symbols (such as parenthesis) to be accepted in the specific path. Line breaks are indicated by "hard-returns" and all command lines end with a "hard return"; thus, each command needs its own line. All command lines are limited in their length to 80 characters per line.

The order of the commands is important to the proper execution of the file. The KEEP command allows you to specify the variables that will remain in the new data set. Conversely, a DROP command can be used specifying the variables to be removed from the final data set. The WHERE command puts conditions on the observations that will be transferred to the new data set. The COPY command specifies the file being transferred or parsed and the new file being generated; further, this is followed by a single forward slash and any options that are needed. The options (/x1 /x2) can refer to either variables or files; a brief discussion of options is presented in Section IV.

Multiple KEEP, DROP, or WHERE statements are not allowed for a given COPY command (if multiple commands are included for each of these commands the last command will be accepted). Further, WHERE statements are limited to one or two expression/conditions depending on the nature of the statement. Thus, each combination of KEEP/DROP and WHERE commands require a COPY command.

To create a command file, follow the steps below[2]:

1) Open your favourite text editor (notepad, and the MS-DOS editor are a couple examples of text editors)

[2] Setting the file up this way will allow you to run the file as an executable, by just double-clicking on the file the commands will be run and a log file generated.

2) Once the editor is open, enter your command file text (code and commands).

3) To save this file:

i. select “file” then "save as"

ii. select drive and path directory (drive, folder, sub-folders)

iii. enter "file_name.stc" (this is the most critical step--the stc extension is required, without the extension Stat/Transfer will not be able to recognize the file, because Stat/Transfer assumes that all command files have the “stc” extension), then select <ok>; note if you are using a DOS text editor there may be file name length restrictions.[3]

iv. to close the program, select “file” then “exit”

4) To edit the file, right click on the file and select edit.[4] This allows you to edit and enter (copy and paste) text in a text-editor environment. Hint: copy and paste the path from the windows explorer address line to save time. Once these changes are complete just save the file and close.

5) To run the file from a windows environment just double click on the "file_name.stc" file to run the command file directly.

6) If step 5 does not work you may need to create a Stat/Transfer association to this file. To create an association right click on the file and select “Open with”, then choose “st.exe”; this “st.exe” file is the executable for the DOS version of StatTransfer. If the “st.exe” is not an available option, then select browse to locate this option. To find this executable choose the exact drive/path/and sub-folders that point to the Stat/Transfer directory and select the “st.exe” file for the association. By selecting this file for the association, all files with the stc extension can be run simply by double clicking on the file.

IV. Example of a command file

An example using a cross sectional sample from the 1996 Survey of Labour and Income Dynamics is presented below. The number of records on the file was 76,055. There were nine variables chosen for this example (year, personid, ailbwt26, jobid, age26, sex21, clwkr1, totear1, and ttinc42).[5] This example will illustrate two primary tasks both a conversion of the format of the data to a new format (Stata) and a parsing of the data. Further, in this example, year, jobid, and ailbwt26 will be dropped (or not keep), and only the observations with the characteristics of female employee older than age 25 with total wage earnings being greater than 80% of their total income will be keep in the final data set.

Refer to Keown [2004] for a discussion on variable selection, specifically, the use of wildcards and selection of variables with similar naming conventions. All of the variable

[3] It is also possible and useful to use the rename feature in windows to give the file the stc extension.

[4] It is important to note, that if the edit does not come up on the list of choices after right-clicking on the file, then your computer is not recognizing the Stat/Transfer command file extension as being associated with Stat/Transfer. To correct this problem check and establish file associations with Stat/Transfer, see Step 6 for guidance.

[5] The variables are year—which is the year of the survey; personid—a unique variable identifying individuals; ailbwt26—cross section weight; jobid—unique identifier for a job spell; age26—person’s age as of December 31; sex21—sex of respondent (male 1 and female 2); clwkr1—class of worker (employee 01, unpaid family worker 02, incorporated business-with paid help 03, incorporated business-no paid help 04, not incorporated business-with paid help 05, not incorporated business-no paid help 06, and other administrative codes 96, 97, 98, and 99; totear1—total earnings from a paid worker job; and ttinc42—total income (before taxes).

selection techniques that apply in the “point and click” environment also apply for the command file method.

This command file has used a few command processor options during the conversion of the SAS data set to a Stata data set, both for variables and the files. The variable options used were the “o”, which turns off the automatic optimization of the output variable types, and double “d” as opposed to float variables (this will preserve these variables length); further, the file option overwrite yes “/y” was selected to replace the data set when the name in the current run was the same as that used in a previous run.[6]

```
Log using “H:\Projects\SLID_example\joint_log_abc”

//Log using “H:\Projects\SLID_example\joint_log_a”
keep year personid age26 sex21 clwkr1 totear1 ttinc42
where sex21 = 2 & clwkr1 = 01
copy “H:\Projects\SLID_example\slid_1.sas7bdat” “H:\Projects\SLID_example\slid_a.dta” /od /y
//quit

//Log using “H:\Projects\SLID_example\joint_log_b”
drop year
where age26 >= 25
copy “H:\Projects\SLID_example\slid_a.dta” “H:\Projects\SLID_example\slid_b.dta” /od /y
//quit

//Log using “H:\Projects\SLID_example\joint_log_c”
keep personid age26 sex21 clwkr1 totear1 ttinc42
where (totear1/ttinc42) > .8
copy “H:\Projects\SLID_example\slid_b.dta” “H:\Projects\SLID_example\slid_c.dta” /od /y
quit
```

Notice that these command files can be run separately, as three distinct files, or jointly, with all of the text within a single command file. This example illustrates how these programs can be run sequentially in a single command file; this was done by commenting-out the earlier quit commands and leaving only one at the end of the file; and commenting-out the individual log files that could be generated if each set of commands was run separately.

The true benefits and efficiencies of Stat/Transfer command files are realized when researchers batch many transfers or parsing tasks together, as the file runs unattended researchers are free to tend other objectives. The command file above generated the following log files, which documents the conversion and partition of the data set. See Figure 1 below for a graphical depiction.

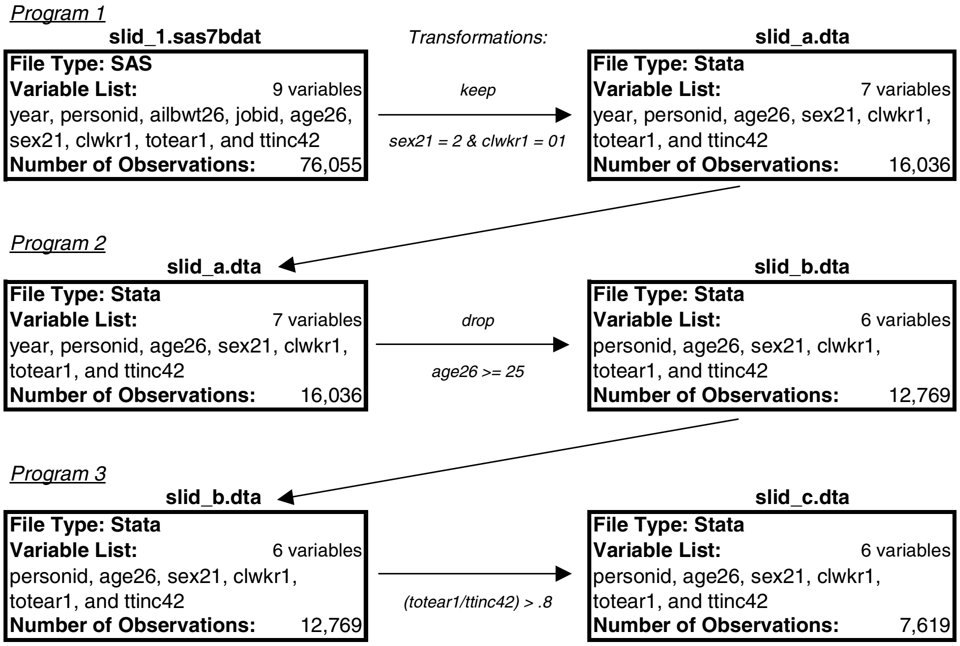
[6] For more details on options see Stat/Transfer’s help, copy commands, and command processor options (for variables, input data sets, and messages).

```
Transferring from SAS Data File- Versions 7/8: H:\Projects\SLID_example\slid_1.sas7bdat
Input file has 9 variables
Optimizing...
7 variables were kept
Transferring to Stata: H:\Projects\SLID_example\slid_a.dta
16036 cases were transferred (1.22 seconds)

Transferring from Stata: H:\Projects\SLID_example\slid_a.dta
Input file has 7 variables
Optimizing...
1 variable was dropped
Transferring to Stata: H:\Projects\SLID_example\slid_b.dta
12769 cases were transferred (0.58 seconds)

Transferring from Stata: H:\Projects\SLID_example\slid_b.dta
Input file has 6 variables
Optimizing...
6 variables were kept
Transferring to Stata: H:\Projects\SLID_example\slid_c.dta
7619 cases were transferred (0.46 seconds)
```

Figure 1: Data Flow (Conversion and Partition)


Program 1
slid_1.sas7bdat
File Type: SAS
Variable List: 9 variables
year, personid, ailbwt26, jobid, age26, sex21, clwkr1, totear1, and ttinc42
Number of Observations: 76,055
Transformations:
keep
sex21 = 2 & clwkr1 = 01
slid_a.dta
File Type: Stata
Variable List: 7 variables
year, personid, age26, sex21, clwkr1, totear1, and ttinc42
Number of Observations: 16,036
Program 2
slid_a.dta
File Type: Stata
Variable List: 7 variables
year, personid, age26, sex21, clwkr1, totear1, and ttinc42
Number of Observations: 16,036
drop
age26 >= 25
slid_b.dta
File Type: Stata
Variable List: 6 variables
personid, age26, sex21, clwkr1, totear1, and ttinc42
Number of Observations: 12,769
Program 3
slid_b.dta
File Type: Stata
Variable List: 6 variables
personid, age26, sex21, clwkr1, totear1, and ttinc42
Number of Observations: 12,769
(totear1/ttinc42) > .8
slid_c.dta
File Type: Stata
Variable List: 6 variables
personid, age26, sex21, clwkr1, totear1, and ttinc42
Number of Observations: 7,619

Researchers who choose not to parse their data and just convert their current file can just omit the WHERE command, and if all of the variables need to be transferred similarly just omit the KEEP command (or comment them out). Stat/Transfer can also be used for data parsing only, while not converting data or transferring it to a new format; to do this simply ensure that your file extensions are the same for your infile and outfile names, although the names can be different, so that you maintain your original and generate a new parsed data set. It is important to note that the input and output file specifications cannot be identical. In other words, either the names of the files or the paths must differ, thus a data set cannot be overwritten in the same step of the procedure.

Finally, to verify that the file has been parsed and transferred correctly, exploratory statistics such as frequencies and cross-tabs should be run to check totals, means, and percentages for the correct transfer of information. This new file may then be used for the desired analysis.

V. Conclusion

In summary, this note has outlined a simple step-by-step method for creating command files. Researchers using Stat/Transfer command files will be able to convert or parse datasets in an efficient replicable manner. Batching many conversions/transfers or data parsing tasks are where the true benefits and efficiencies of Stat/Transfer command files are realized.

References

Keown, Leslie-Anne. 2004. "Producing Efficient Data Files using Stat/Transfer." The Research Data Centres Information and Technical Bulletin. 1(1):9-15. Catalogue no. 12-002-XIE. Statistics Canada, Ottawa.

Statistics Canada. 2004. Survey of Labour and Income Dynamics. Years of Data available: 1992 to 2001. Ottawa: Ministry of Industry.

Stat/Transfer. 2000. File Transfer Utility for Windows, Version 6.0. Circle Systems Inc., Seattle, Washington. (http://www.stattransfer.com)

Technical Note

A note on identifiers in the National Longitudinal Survey of Children and Youth

By Franck Larouche

As part of the processing of the National Longitudinal Survey of Children and Youth (NLSCY) cycle 4 data, historical revisions have been made to the data of the first 3 cycles, either to correct errors or to update the data. The original version of the data was replaced during the autumn of 2003.

During processing, particular attention was given to the PERSRUK (Person Identifier) and the FIELDRUK (Household Identifier). First, they have both been standardised to have the same length in each cycle which was required to merge the files from all cycles. The FIELDRUK is now a 12-character long variable and the PERSRUK a 14-character long variable. Secondly, both identifiers have been updated. Some FIELDRUKs and PERSRUKs were wrong. Corrections have been made so that now, all FIELDRUKs and PERSRUKs should be valid.

The unique identifier for the child (PERSRUK) remains the same in each cycle. When a child is added to a particular cycle, a unique PERSRUK is given to the child that will remain the same in subsequent releases of NLSCY. The unique identifier for the household (FIELDRUK) is constant across cycles for children who remain in the same household. All children part of a same household have the same household ID (FIELDRUK). Over the years, some children do not continue to live in the same household. This occurs for several reasons: they move out, families are divided, and so on. Therefore the household ID (FIELDRUK) is not necessarily the same over time, the household ID for a child will change if the child's household changes.

The same level of attention has not been given to the other identifiers that are included in the data base, the CHILDID (Person identifier) and the _IDHD01 (Household identifier). Note that these identifiers have been created for the public files and can also be found in the master files by default. For this reason, the PERSRUK should be used to link records between files and the FIELDRUK to determine the household when using the master files.

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 Statistics Canada *Style Guide.*

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.

Follow the instructions for formatting tables and graphs in the Statistics Canada *Style Guide*.

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 Statistics Canada *Style Guide* in all respects. A copy of the *Style Guide* is available by contacting the Editorial Committee at the addresses, below:

Addresses 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
- or by regular mail to:

The Editorial Committee, RDC Information and Technical Bulletin
McMaster Research Data Centre, Statistics Canada
Mills Library Memorial
Library Room 217
1280 Main Street West,
Hamilton, Ontario L8S 4L6
Canada

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 addresses above.