The Stata Journal

The *Stata Journal* publishes reviewed papers together with shorter notes or comments, regular columns, book reviews, and other material of interest to Stata users. Examples of the types of papers include 1) expository papers that link the use of Stata commands or programs to associated principles, such as those that will serve as tutorials for users first encountering a new field of statistics or a major new technique; 2) papers that go "beyond the Stata manual" in explaining key features or uses of Stata that are of interest to intermediate or advanced users of Stata; 3) papers that discuss new commands or Stata programs of interest either to a wide spectrum of users (e.g., in data management or graphics) or to some large segment of Stata users (e.g., in survey statistics, survival analysis, panel analysis, or limited dependent variable modeling); 4) papers analyzing the statistical properties of new or existing estimators and tests in Stata; 5) papers that could be of interest or usefulness to researchers, especially in fields that are of practical importance but are not often included in texts or other journals, such as the use of Stata in managing datasets, especially large datasets, with advice from hard-won experience; and 6) papers of interest to those who teach, including Stata with topics such as extended examples of techniques and interpretation of results, simulations of statistical concepts, and overviews of subject areas.

The *Stata Journal* is indexed and abstracted by *CompuMath Citation Index*, *Current Contents/Social and Behavioral Sciences*, *RePEc: Research Papers in Economics*, *Science Citation Index Expanded* (also known as *SciSearch*), *Scopus*, and *Social Sciences Citation Index*.

For more information on the *Stata Journal*, including information for authors, see the webpage

http://www.stata-journal.com

Subscriptions are available from StataCorp, 4905 Lakeway Drive, College Station, Texas 77845, telephone 979-696-4600 or 800-STATA-PC, fax 979-696-4601, or online at

http://www.stata.com/bookstore/sj.html

Subscription rates listed below include both a printed and an electronic copy unless otherwise mentioned.

U.S. and Canada		Elsewhere	
Printed & electronic		**Printed & electronic**	
1-year subscription	$115	1-year subscription	$145
2-year subscription	$210	2-year subscription	$270
3-year subscription	$285	3-year subscription	$375
1-year student subscription	$ 85	1-year student subscription	$115
1-year institutional subscription	$345	1-year institutional subscription	$375
2-year institutional subscription	$625	2-year institutional subscription	$685
3-year institutional subscription	$875	3-year institutional subscription	$965
Electronic only		**Electronic only**	
1-year subscription	$ 85	1-year subscription	$ 85
2-year subscription	$155	2-year subscription	$155
3-year subscription	$215	3-year subscription	$215
1-year student subscription	$ 55	1-year student subscription	$ 55

Back issues of the *Stata Journal* may be ordered online at

http://www.stata.com/bookstore/sjj.html

Individual articles three or more years old may be accessed online without charge. More recent articles may be ordered online.

http://www.stata-journal.com/archives.html

The *Stata Journal* is published quarterly by the Stata Press, College Station, Texas, USA.

Address changes should be sent to the *Stata Journal*, StataCorp, 4905 Lakeway Drive, College Station, TX 77845, USA, or emailed to sj@stata.com.

The *Stata Journal* (ISSN 1536-867X) is a publication of Stata Press. Stata, Stata, Stata Press, Mata, Mata, and NetCourse are registered trademarks of StataCorp LP.

The Stata Journal (2015)
15, Number 4, pp. 1167–1173

Complete automation of a participant characteristics table

Seth T. Lirette
Center of Biostatistics and Bioinformatics
University of Mississippi Medical Center
Jackson, MS
and
Department of Biostatistics
University of Alabama at Birmingham
Birmingham, AL
slirette2@umc.edu

Abstract. Most published research that contains real-world data displays a table of participant characteristics, baseline characteristics, or demographics. I present a program, `partchart`, that provides automatic output of the participant characteristics table in multiple formats and gives the user control over formatting, thus facilitating complete reproducibility.

Keywords: st0418, partchart, reproducibility, participant characteristics, baseline characteristics, table automation

1 Introduction

Many recent articles have been devoted to reproducible research in the various fields of science (Peng 2011; Laine et al. 2007; Peng, Dominici, and Zeger 2006; and Gentleman and Lang 2007). Reproducibility is an integral part of proving the validity of any scientific theory. Clean and well-defined data and efficient coding are essential for reproducibility. Because of unintentional human error, numbers in reported tables are often entered incorrectly, thus relaying unreliable information. The coder should thus allow the computer to handle this tedious task to the greatest extent possible. Stata has a host of user-written commands used to automate this process. These are usually encountered in the context of results from model building (Jann 2005; Gallup 2012). Lo Magno (2013) recently addressed such issues when exporting entire reports directly into Microsoft Word. While all of these commands are incredibly useful, I present a new program, `partchart`, that fills a void for producing the commonly used participant characteristics table, also known as the baseline characteristics table.

These tables are especially prevalent in medical publications. Both the CONSORT (Schulz, Altman, and Moher 2010) statement for reporting clinical trials as well as the STROBE (von Elm et al. 2007) statement for reporting observational studies recommend including a participant characteristics table to show a concise summary of the analysis data. Often when one constructs an article, this first table endures many iterations, most of which will change every summary statistic in the table. Instead of tediously

reentering every number by hand, one can use `partchart` to create this table with just one command, thus easing the programming burden and ensuring the accuracy of the table.

2 Essential ado-files

The source code running `partchart` relies upon five user-written commands, all of which are available from the Statistical Software Components archive. These five commands provide very specific utilities used to export the final table, and specific descriptions of each are beyond the scope of this article. Nevertheless, they are worth mentioning: 1) `unique`, written by Brady (1998); 2) `mat2txt`, written by Blasnik and Jann (2004); 3) `lstrfun`, written by Blanchette (2010); 4) `tabcount`, written by Cox (2002); and 5) `dataout`, written by Wada (2009).

3 The partchart command

`partchart` exports a "raw" participant characteristics table into a specified file format. This raw table can then be formatted into a publication-quality table. `partchart` eases the burden of either entering summary statistics by hand or using multiple lines of code to automate the process with `tabstat` or something similar. `partchart` allows the user to export a table very close to publication quality with one command.

`partchart` automatically parses *varlist* into continuous variables and categorical variables but offers the user the ability to control these as well. The output table contains means and standard deviations by default for continuous variables but will output user-defined summary statistics with the `constats()` option. Counts and percentages are output for categorical variables. These details are given to the user as a note printed to the Stata window. Sample sizes are displayed in the last row of the table. With `by(`*varname*`)` invoked, the numeric suffix on the column headers corresponds to the numeric values of *varname*.

A program called `table1` (Clayton 2013) is available from the Statistical Software Components archive. The original version of `partchart` was named `table1`, and both Clayton and I arrived at these programs independently. Both programs essentially offer the same service, but `partchart` offers more formatting options, output options, and an improved arrangement of variables.

3.1 Syntax

`partchart` *varlist* [*if*] [*in*]`, file(`*filename*`)` [`sheet(`*string*`) catcut(`*integer*`)`
`catsep("`*string*`", nopercent) cattest(exact) constats(`*string*`)`
`conprec(`*integer*`) consep("`*string*`") contest(kwallis) nobase(`*varlist*`)`
`by(`*varname*`)`]

3.2 Options

`file(`*filename*`)` indicates the filename that is output to the active directory. Subdirectories are also supported. This file must be closed before running `partchart`. Supported file formats are `.xlsx` (the default), `.xls`, `.csv`, `.txt` (tab delimited), and `.tex`. `file()` is required.

`sheet(`*string*`)` indicates the sheet name in *filename* for the output if an `.xlsx` or `.xls` file format is used. The default is `sheet(partchartraw)`.

`catcut(`*integer*`)` specifies the cutoff number of categories for `partchart` to separate continuous variables from categorical variables. This is usually a problem only for small datasets (misclassifying continuous as categorical) or when categorical variables have many categories. The default is `catcut(10)`, which means that any variable with fewer than 10 unique values is considered categorical.

`catsep("`*string*`", nopercent)`, used with categorical variables, specifies the string used to encapsulate the second statistic, which is a percentage. *string* must be surrounded by quotation marks, and string lengths of only 1 or 2 are valid. The default is `catsep("()")`. The suboption `nopercent` specifies to not include the `%` in the output.

`cattest(exact)` tells `partchart` to perform Fisher's exact test and report the corresponding p-values instead of performing a chi-squared test for differences in categorical variables. This option is valid only if the `by()` option is invoked.

`constats(`*string*`)` tells `partchart` which descriptive statistics to report for continuous variables. It must be specified in the form `constats(`*stat1 stat2*`)`, where *stat1* and *stat2* can be anything from the list of statistics from `tabstat`, excluding `q`. The default is `constats(mean sd)`.

`conprec(`*integer*`)` specifies the precision for continuous variables indicated as number of decimal places. Only positive integers are valid. The default is `conprec(2)`.

`consep("`*string*`")`, used with continuous variables, specifies the string used to encapsulate the second statistic, which is the standard deviation by default. Like `catsep()`, *string* must be surrounded by quotation marks, and string lengths of only 1 or 2 are valid. The default is `consep("()")`.

`contest(kwallis)` tells `partchart` to perform a Kruskal–Wallis (Mann–Whitney U) test and report the corresponding p-values instead of performing an analysis of variance (ANOVA) (t test) for differences in continuous variables. `contest(kwallis)` is valid only with the `by()` option.

`nobase(`*varlist*`)` specifies that rows from the table be excluded. The base level of any variable in `nobase(`*varlist*`)` will be excluded from the table. This applies only to categorical variables.

`by(`*varname*`)` splits the table into a two-way table by *varname*. If the `by()` option is not included, `partchart` outputs a one-way table. The *varname* must be categorical with a reasonably small number of categories. In addition, values for *varname* must

be sequential integers starting with either 0 or 1. Also the `by()` option outputs p-values for differences. These p-values are based on ANOVA or t tests for continuous variables and chi-squared tests for categorical variables (unless otherwise specified by `contest(kwallis)` or `cattest(exact)`).

3.3 Caveats

`partchart` has a few caveats that warrant discussion, and these are generally resolved easily.

First, as expected, string variables are not allowed in *varlist*. The user is required to generate a new variable. This new variable is allowed to have labels, if so desired.

Second, if the `by(`*varname*`)` option is invoked, the `by()` variable must be a sequential integer starting at either 0 or 1 (that is, 0, 1, 2, 3, ... or 1, 2, 3, ...). `partchart` issues an error message reminding the user if the user attempts otherwise.

The last few caveats all deal with categorical variables in *varlist*. If the user includes a variable in *varlist* that has more than 99 categories, the program will fail. Although I cannot imagine a real-world scenario where this would occur, the warning has been issued. Also, similar to a caveat discussed above, any categorical variable must also start at 0 or 1; however, unlike the `by(`*varname*`)` variable, it does not have to be sequential. Gaps are allowed. While the necessity of starting at 0 or 1 is true, in general, it need not apply if the `if` condition is specified. An error message is issued if a violation occurs.

4 Examples

```
. sysuse auto
(1978 Automobile Data)

. partchart price rep78 mpg foreign, file(partchart_ex1) sheet(partchart-ref1)
file partchart_ex1.xlsx saved
```

variable	total
price	6165.26 (2949.50)
rep781	2 (3%)
rep782	8 (12%)
rep783	30 (43%)
rep784	18 (26%)
rep785	11 (16%)
mpg	21.30 (5.79)
foreign0	52 (70%)
foreign1	22 (30%)
Sample Size	74

```
*The table contains means and sds for continuous variables (price mpg) and counts
> and percentages for categorical variables (rep78 foreign).
```

This is the most basic syntax for `partchart`. It produces a `partchart_ex1.xlsx` file stored in the current directory under the sheet named `partchart-ref1`, with one column of the format with mean (standard deviation) for both `price` and `mpg`. In the same column, the count `(%)` is shown for `rep78` and `foreign`. `Sample Size` is shown in the last row.

```
. partchart price rep78 mpg, file(partchart_ex2.txt) by(foreign) conprec(1)
```

variable	total	foreign0	foreign1	pvalue
price	6165.3 (2949.5)	6072.4 (3097.1)	6384.7 (2621.9)	0.680
rep781	2 (3%)	2 (4%)	0 (0%)	<0.001
rep782	8 (12%)	8 (17%)	0 (0%)	<0.001
rep783	30 (43%)	27 (56%)	3 (14%)	<0.001
rep784	18 (26%)	9 (19%)	9 (43%)	<0.001
rep785	11 (16%)	2 (4%)	9 (43%)	<0.001
mpg	21.3 (5.8)	19.8 (4.7)	24.8 (6.6)	0.001
Sample Size	74	52	22	

```
*The table contains means and sds for continuous variables (price mpg) and counts
> and percentages for categorical variables (rep78).
```

In this example, `partchart` outputs the tab-delimited file `partchart_ex2.txt` with columns over the values of `foreign`. Each column contains the mean (standard deviation) for continuous variables and the count `(%)` for categorical values. Added here is the p-value taken from an ANOVA test for `price` and `mpg` and a chi-squared test for `rep78`. The first column in the file shows the descriptive statistics for the total, similar to what would be shown if the `by()` option was omitted. `Sample Size` for each grouping is shown in the last row.

```
. partchart price rep78 mpg, file(partchart_ex3.csv) by(foreign)
> constats(median iqr) conprec(1) contest(kwallis) cattest(exact)
> consep("{}") catsep("[]")
> nobase(rep78)
```

variable	total	foreign0	foreign1	pvalue
price	5006.5 {2147.0}	4782.5 {2050.0}	5759.0 {2641.0}	0.298
rep782	8 [12%]	8 [17%]	0 [0%]	<0.001
rep783	30 [43%]	27 [56%]	3 [14%]	<0.001
rep784	18 [26%]	9 [19%]	9 [43%]	<0.001
rep785	11 [16%]	2 [4%]	9 [43%]	<0.001
mpg	20.0 {7.0}	19.0 {5.5}	24.5 {7.0}	0.002
Sample Size	74	52	22	

```
*The table contains medians and iqrs for continuous variables (price mpg) and
> counts and percentages for categorical variables (rep78).
```

This example shows the full capabilities of `partchart`. The file `partchart_ex3.csv` is output to the current directory with columns over the values of `foreign`. For `price` and `mpg`, each column contains the median {interquartile range}, each rounded to one decimal place. The count `[%]` is displayed for `rep78`. In this example, the p-values correspond to the Kruskal–Wallis (Mann–Whitney U) test for `price` and `mpg` and to Fisher's exact test for `rep78`. The "base" category for `rep78` (`rep78` = 1) is omitted by using the `nobase()` option. `Sample Size` for each grouping is shown in the last row.

5 References

Blanchette, D. 2010. lstrfun: Stata module to modify long local macros. Statistical Software Components S457169, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s457169.html.

Blasnik, M., and B. Jann. 2004. mat2txt: Stata module to write matrix to ASCII file. Statistical Software Components S437601, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s437601.html.

Brady, T. 1998. unique: Stata module to report number of unique values in variable(s). Statistical Software Components S354201, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s354201.html.

Clayton, P. 2013. table1: Stata module to create "table 1" of baseline characteristics for a manuscript. Statistical Software Components S457730, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s457730.html.

Cox, N. J. 2002. tabcount: Stata module to tabulate frequencies, with zeros explicit. Statistical Software Components S429501, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s429501.html.

Gallup, J. L. 2012. A programmer's command to build formatted statistical tables. *Stata Journal* 12: 655–673.

Gentleman, R., and D. T. Lang. 2007. Statistical analyses and reproducible research. *Journal of Computational and Graphical Statistics* 16: 1–23.

Jann, B. 2005. Making regression tables from stored estimates. *Stata Journal* 5: 288–308.

Laine, C., S. N. Goodman, M. E. Griswold, and H. C. Sox. 2007. Reproducible research: Moving toward research the public can really trust. *Annals of Internal Medicine* 146: 450–453.

Lo Magno, G. L. 2013. sar: Automatic generation of statistical reports using Stata and Microsoft Word for Windows. *Stata Journal* 13: 39–64.

Peng, R. D. 2011. Reproducible research in computational science. *Science* 334: 1226–1227.

Peng, R. D., F. Dominici, and S. L. Zeger. 2006. Reproducible epidemiologic research. *American Journal of Epidemiology* 163: 783–789.

Schulz, K. F., D. G. Altman, and D. Moher. 2010. CONSORT 2010 statement: Updated guidelines for reporting parallel group randomised trials. *British Medical Journal* 340: 698–702.

von Elm, E., D. G. Altman, M. Egger, S. J. Pocock, P. C. Gøtzsche, and J. P. Vandenbroucke. 2007. Strengthening the reporting of observational studies in epidemiology (STROBE) statement: Guidelines for reporting observational studies. *Annals of Internal Medicine* 147: 573–577.

Wada, R. 2009. dataout: Stata module to export a dataset or tab-delimited file into various formats. Statistical Software Components S457022, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s457022.html.

About the author

Seth Lirette is a biostatistician at the University of Mississippi Medical Center and a PhD student at the University of Alabama at Birmingham.