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	$ 98	1-year subscription	$138
2-year subscription	$165	2-year subscription	$245
3-year subscription	$225	3-year subscription	$345
1-year student subscription	$ 75	1-year student subscription	$ 99
1-year institutional subscription	$245	1-year institutional subscription	$285
2-year institutional subscription	$445	2-year institutional subscription	$525
3-year institutional subscription	$645	3-year institutional subscription	$765
Electronic only		**Electronic only**	
1-year subscription	$ 75	1-year subscription	$ 75
2-year subscription	$125	2-year subscription	$125
3-year subscription	$165	3-year subscription	$165
1-year student subscription	$ 45	1-year student subscription	$ 45

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 (2014)
14, Number 3, pp. 662–669

csvconvert: A simple command to gather comma-separated value files into Stata

Alberto A. Gaggero
Department of Economics and Management
University of Pavia
Pavia, Italy
alberto.gaggero@unipv.it

Abstract. This command meets the need of a researcher who holds multiple data files in comma-separated value format differing by a period variable (for example, year or quarter) or by a cross-sectional variable (for example, country or firm) and must combine them into one Stata-format file.

Keywords: dm0076, csvconvert, comma-separated value file, .csv

1 Introduction

In applied research, it is common to come across several data files containing the same set of variables that need to be combined into one file. For instance, in a cross-country survey, a researcher may collect information country by country and thus create several data files, one for each country. Or within the same cross-section (or even within the same country), the researcher may sample each year independently and generate various data files that differ by year.

A practical issue in this type of situation is determining how to read all of those files together in Stata, especially if they are manifold. The standard approach would be to import each data file sequentially into Stata by using a combination of `import delimited` and `append`. This approach, however, requires a user to type several command lines proportional to the number of files to be included; thus it is reasonably doable if the number of data files is limited.

Suppose the directory `C:\data\world_bank` contains three comma-separated value (`.csv`) files: `wb2007.csv`, `wb2008.csv`, and `wb2009.csv`.[1] After setting the appropriate working directory, a user implements the aforementioned procedure by typing the following command lines:

```
. import delimited using wb2008.csv, clear
. save wb2008.dta
. import delimited using wb2009.csv, clear
. save wb2009.dta
. import delimited using wb2007.csv, clear
```

[1] `csvconvert` is designed to handle many `.csv` files; however, for simplicity, all the examples below consider a limited set of `.csv` files.

```
. append using wb2008.dta
. append using wb2009.dta
```

Alternatively, and more compactly, the same result can be obtained with a loop.

```
. foreach file in wb2007 wb2008 wb2009 {
  2. import delimited using `file´.csv, clear
  3. save `file´
  4. }
. foreach file in wb2007.dta wb2008.dta {
  2. append using `file´
  3. }
```

Another way is to work with the disk operating system (DOS) to gather all the `.csv` files into one `.csv` file and then to read the assembled single `.csv` file into memory using `import delimited`.

Under the DOS framework, the lines below assemble `wb2007.csv`, `wb2008.csv`, and `wb2009.csv` into a newly created `.csv` file named `input.csv`.

```
cd "C:\data\world bank"
copy wb2007.csv wb2008.csv wb2009.csv input.csv
```

To assemble all `.csv` files stored in the directory `C:\data\world bank` into a new file named `input.csv`, type

```
cd "C:\data\world bank"
copy *.csv input.csv
```

To read `input.csv` into Stata, type

```
. import delimited using C:\data\world bank\input.csv
```

A similar approach that bypasses the DOS framework can be implemented. However, if the number of `.csv` files is large, the process may not be as straightforward. For simplicity, let us still consider just three `.csv` files. Once the appropriate working directory is set, the command lines to type are as follows:

```
. copy wb2008.csv wb2007.csv, append
. copy wb2009.csv wb2007.csv, append
. import delimited using wb2007.csv
```

The first two command lines append `wb2008.csv` and `wb2009.csv` to `wb2007.csv`. The third command reads the `.csv` file into Stata.

Note, however, that if the first line of both `wb2008.csv` and `wb2009.csv` contains the variable names, these are also appended.[2] Thus, because of the presence of extra lines with names, all the variables are read as a string. To correct this inaccuracy, one should first remove the lines with the variable names and then use `destring` to set the numerical format.

[2] Unfortunately, the option `varnames(nonames)`, applicable with `import delimited`, is unavailable with `copy`.

Alternatively, we could prevent this fault by manually preparing the `.csv` files (that is, by removing the lines with the variable names in the `.csv` files to be appended). The whole process can be time consuming, especially if the number of `.csv` files is large. The `csvconvert` command simplifies and automatizes the procedure of gathering multiple `.csv` files into one `.dta`, as illustrated in the next section.

2 The csvconvert command

2.1 Syntax

The syntax is

`csvconvert` *input_directory*`, replace` [`input_file(`*filenames*`)` `output_dir(`*output_directory*`)` `output_file(`*filename*`)`]

where *input_directory* is the path of the directory in which the `.csv` files are stored. Do not use any quotes at the endpoints of the directory path, even if the directory name contains spaces (see example 1 below).

2.2 Options

`replace` specifies that the existing output file (if it already exists) be overwritten. `replace` is required.

`input_file(`*filenames*`)` specifies a subset of the `.csv` files to be converted. The *filenames* must be separated by a space and include the `.csv` extension (see example 2 below). If this option is not specified, `csvconvert` considers all the `.csv` files stored in the input directory.

`output_dir(`*output_directory*`)` specifies the directory in which the `.dta` output file is saved. If this option is not specified, the file is saved in the same directory where the `.csv` files are stored.

`output_file(`*filename*`)` specifies the name of the `.dta` output file. The default is `output_file(output.dta)`.

3 Examples

3.1 Example 1—Basic

The simplest way to run `csvconvert` is to type the command and the directory path where the `.csv` files are stored followed by the mandatory option `replace`. In the same directory, Stata will create `output.dta`, which collects all the `.csv` files of that directory in Stata format.

```
. csvconvert C:\data\world bank, replace
-------------------------------------------------
The csv file wb2007.csv
(6 vars, 3 obs)
has been successfully included in output.dta
-------------------------------------------------
The csv file wb2008.csv
(6 vars, 3 obs)
has been successfully included in output.dta
-------------------------------------------------
The csv file wb2009.csv
(6 vars, 3 obs)
has been successfully included in output.dta
-------------------------------------------------
****************************************************************
 You have successfully converted 3 csv files in one Stata file
****************************************************************
```

3.2 Example 2—Subset of .csv files to be converted

If you want to convert only a subset of the .csv files in the directory (for example, wb2008.csv and wb2009.csv), then you need to list the files to be converted inside the parentheses of the option input_file(). Filenames must be separated by a blank space and must be specified using the .csv extension.

```
. csvconvert C:\data\world bank, replace input_file(wb2008.csv wb2009.csv)
-------------------------------------------------
The csv file wb2008.csv
(6 vars, 3 obs)
has been successfully included in output.dta
-------------------------------------------------
The csv file wb2009.csv
(6 vars, 3 obs)
has been successfully included in output.dta
-------------------------------------------------
****************************************************************
 You have successfully converted 2 csv files in one Stata file
****************************************************************
```

3.3 Example 3—Naming the output file and saving it in a predetermined directory

Suppose you wish to name your output file `wb_data` and save it in the directory `C:\data\wb dataset`. In this case, you would type

```
. csvconvert C:\data\world bank, replace output_file(wb_data.dta)
> output_dir(C:\data\wb dataset)
--------------------------------------------------
The csv file wb2007.csv
(6 vars, 3 obs)
has been successfully included in wb_data.dta.dta
--------------------------------------------------
The csv file wb2008.csv
(6 vars, 3 obs)
has been successfully included in wb_data.dta.dta
--------------------------------------------------
The csv file wb2009.csv
(6 vars, 3 obs)
has been successfully included in wb_data.dta.dta
--------------------------------------------------
*****************************************************************
 You have successfully converted 3 csv files in one Stata file
*****************************************************************
```

3.4 Example 4—Including all possible options

Example 2 and example 3 can be combined.

```
. csvconvert C:\data\world bank, replace input_file(wb2008.csv wb2009.csv)
> output_file(wb_data.dta) output_dir(C:\data\wb dataset)
  (output omitted)
```

4 Tips and additional examples for practitioners

`csvconvert` is designed to speed up the process of joining a large number of `.csv` files. As the number of input files increases, the likelihood that one of them contains inaccuracies rises. It is important, therefore, to keep track of all the steps in the process so that the origin of possible faults can be detected.

While creating the output file, `csvconvert` offers various ways to check that the conversion of the `.csv` files into Stata has been completed correctly.

First of all, at the end of the process, `csvconvert` displays the number of `.csv` files contained in the output file. This information allows a researcher to check whether the expected number of `.csv` files to be included in the output file is equal to the actual number of `.csv` files that have been converted. The complete list of `.csv` files included in the `.dta` file can be obtained by typing `note` (see example 5). Additionally, by default, `csvconvert` creates one variable named `_csvfile`, which encloses the name of the `.csv` file where the observation originates.

During conversion, csvconvert sequentially reports the name of the .csv file being converted, the number of variables, and the number of observations. If something in the process appears odd, extra messages are displayed to alert the researcher and demand further inspection. For instance, suppose that one .csv file contains a symbol or a letter in one cell of a numerical variable; if ignored, this inaccuracy may undermine the whole process. For this reason, csvconvert adds a note to help the researcher detect the fault. In example 6, wb2008_symbol.csv contains "N/A" in one cell of the variable populationtotal.

4.1 Example 5—List of the .csv files included in the .dta file

Once csvconvert has been completed, the full list of .csv files included in the .dta file, together with the date and time when each .csv file was converted, can be obtained by typing note in the command window.

```
. note
_dta:
  1.  File included on 18 Jan 2014 10:11 : "wb2007.csv"
  2.  File included on 18 Jan 2014 10:11 : "wb2008.csv"
  3.  File included on 18 Jan 2014 10:11 : "wb2009.csv"
```

4.2 Example 6—Detecting the origin of anomalous observations

Suppose that you wish to convert three files: wb2007.csv, wb2008_symbol.csv, and wb2009.csv. The file wb2008_symbol.csv contains a fault (that is, the aforementioned "N/A" cell), but you are unaware of it.

```
. csvconvert C:\data\world bank, replace
> input_file(wb2007.csv wb2008_symbol.csv wb2009.csv)
-------------------------------------------------
The csv file wb2007.csv
(6 vars, 3 obs)
has been successfully included in output.dta
-------------------------------------------------
The csv file wb2008_symbol.csv
(6 vars, 3 obs)
(note: variable populationtotal was long in the using data, but will be str9
       now)
has been successfully included in output.dta
-------------------------------------------------
The csv file wb2009.csv
(6 vars, 3 obs)
(note: variable populationtotal was str9 in the using data, but will be long
       now)
(note: variable _csvfile was str10, now str17 to accommodate using data´s
       values)
has been successfully included in output.dta
-------------------------------------------------
****************************************************************
 You have successfully converted 3 csv files in one Stata file
****************************************************************
```

By reading the log, you can see that in the conversion of wb2008_symbol.csv, the variable populationtotal changed its format from numerical to string. Therefore, wb2008_symbol.csv is the file that needs to be inspected. Once the anomalous observation is detected and manually corrected (for example, by emptying the anomalous cell via Excel and saving the corrected file as wb2008_symbol2.csv), you can relaunch csvconvert and check that it now runs smoothly.

```
. csvconvert C:\data\world bank, replace
> input_file(wb2007.csv wb2008_symbol2.csv wb2009.csv)
-------------------------------------------------
The csv file wb2007.csv
(6 vars, 3 obs)
has been successfully included in output.dta
-------------------------------------------------
The csv file wb2008_symbol2.csv
(6 vars, 3 obs)
has been successfully included in output.dta
-------------------------------------------------
The csv file wb2009.csv
(6 vars, 3 obs)
(note: variable _csvfile was str10, now str18 to accommodate using data´s
       values)
has been successfully included in output.dta
-------------------------------------------------
****************************************************************
 You have successfully converted 3 csv files in one Stata file
****************************************************************
```

4.3 Example 7—Spotting duplicate observations

If csvconvert happens to include duplicate observations (for instance, it inserted the same input file twice), it displays a warning message. Moreover, to facilitate the detection of double observations, csvconvert generates a new dummy variable, _duplicate, that is equal to one in case of duplicate observations. This example describes the procedure to spot whether an input file has been entered twice and, if so, which one.

```
. csvconvert C:\data\world bank, replace
> input_file(wb2008.csv wb2009.csv wb2008.csv)
-------------------------------------------------
The csv file wb2008.csv
(6 vars, 3 obs)
has been successfully included in output.dta
-------------------------------------------------
The csv file wb2009.csv
(6 vars, 3 obs)
has been successfully included in output.dta
-------------------------------------------------
The csv file wb2008.csv
(6 vars, 3 obs)
has been successfully included in output.dta
-------------------------------------------------
****************************************************************
 You have successfully converted 3 csv files in one Stata file
****************************************************************
Warning - output.dta has 3 duplicate observations: you might have entered a
> .csv file name twice in the input_file() option, or your orginal dataset may
> contain duplicates. Check if this is what you wanted: variable ´_duplicates´
> = 1 in case of duplicate and = 0 otherwise may help.
```

The warning message shows that there are three duplicate observations. Of course, you can look carefully at the Results window and find that `wb2008.csv` was entered twice. However, if you are handling a large set of `.csv` files, checking each line of the screen would be very time consuming.

Tabulating the variable `_csvfile` conditional on `_duplicates` being equal to one quickly detects that the duplicate observations come from `wb2008.csv`.

```
. tabulate _csvfile if _duplicates==1
   csv file │
 from which │
observation │
 originates │      Freq.     Percent        Cum.
────────────┼───────────────────────────────────
 wb2008.csv │          6      100.00      100.00
────────────┼───────────────────────────────────
      Total │          6      100.00
```

5 Acknowledgments

I am grateful to Editor Joseph Newton for his assistance during revision and to Violeta Carrion, Emanuele Forlani, Edna Solomon, and one anonymous referee for very helpful comments.

About the author

Alberto A. Gaggero is currently an assistant professor in the Department of Economics and Management at the University of Pavia, where he teaches applied industrial organization. He obtained his PhD from the University of Essex. He formerly held research positions at the University of Genoa, at Hogeschool University Brussel, and at the Belgian Ministry of Economic Affairs. His research topics center on applied industrial organization with particular interest in airline pricing.