

# THE STATA JOURNAL

**Editor**

H. Joseph Newton  
Department of Statistics  
Texas A & M University  
College Station, Texas 77843  
979-845-3142; FAX 979-845-3144  
jnewton@stata-journal.com

**Associate Editors**

Christopher F. Baum  
Boston College

Rino Bellocco  
Karolinska Institutet, Sweden and  
Univ. degli Studi di Milano-Bicocca, Italy

A. Colin Cameron  
University of California–Davis

David Clayton  
Cambridge Inst. for Medical Research

Mario A. Cleves  
Univ. of Arkansas for Medical Sciences

William D. Dupont  
Vanderbilt University

Charles Franklin  
University of Wisconsin–Madison

Joanne M. Garrett  
University of North Carolina

Allan Gregory  
Queen's University

James Hardin  
University of South Carolina

Ben Jann  
ETH Zürich, Switzerland

Stephen Jenkins  
University of Essex

Ulrich Kohler  
WZB, Berlin

**Stata Press Production Manager****Stata Press Copy Editor****Editor**

Nicholas J. Cox  
Department of Geography  
Durham University  
South Road  
Durham City DH1 3LE UK  
n.j.cox@stata-journal.com

Jens Lauritsen  
Odense University Hospital

Stanley Lemeshow  
Ohio State University

J. Scott Long  
Indiana University

Thomas Lumley  
University of Washington–Seattle

Roger Newson  
Imperial College, London

Marcello Pagano  
Harvard School of Public Health

Sophia Rabe-Hesketh  
University of California–Berkeley

J. Patrick Royston  
MRC Clinical Trials Unit, London

Philip Ryan  
University of Adelaide

Mark E. Schaffer  
Heriot-Watt University, Edinburgh

Jeroen Weesie  
Utrecht University

Nicholas J. G. Winter  
University of Virginia

Jeffrey Wooldridge  
Michigan State University

Lisa Gilmore  
Gabe Waggoner

**Copyright Statement:** The Stata Journal and the contents of the supporting files (programs, datasets, and help files) are copyright © by StataCorp LP. The contents of the supporting files (programs, datasets, and help files) may be copied or reproduced by any means whatsoever, in whole or in part, as long as any copy or reproduction includes attribution to both (1) the author and (2) the Stata Journal.

The articles appearing in the Stata Journal may be copied or reproduced as printed copies, in whole or in part, as long as any copy or reproduction includes attribution to both (1) the author and (2) the Stata Journal.

Written permission must be obtained from StataCorp if you wish to make electronic copies of the insertions. This precludes placing electronic copies of the Stata Journal, in whole or in part, on publicly accessible web sites, file servers, or other locations where the copy may be accessed by anyone other than the subscriber.

Users of any of the software, ideas, data, or other materials published in the Stata Journal or the supporting files understand that such use is made without warranty of any kind, by either the Stata Journal, the author, or StataCorp. In particular, there is no warranty of fitness of purpose or merchantability, nor for special, incidental, or consequential damages such as loss of profits. The purpose of the Stata Journal is to promote free communication among Stata users.

The *Stata Journal*, electronic version (ISSN 1536-8734) is a publication of Stata Press. Stata and Mata are registered trademarks of StataCorp LP.

## Stata tip 41: Monitoring loop iterations

David A. Harrison  
Intensive Care National Audit & Research Centre  
London, UK  
david.harrison@icnarc.org

If, like me, you have ever started a Stata program running and returned hours later to find it still running with no idea of whether it is actually getting anywhere, then you will be looking for a simple method to monitor your loops. The commands `bootstrap` and `jackknife` produce an attractive table of dots for just this purpose by using the undocumented command `_dots`.

Any loop can easily be modified to report its progress by using `_dots`:

```
nois _dots 0, title(Loop running) reps(100)
forvalues i = 1/100 {
    (main body of loop)
    nois _dots 'i' 0
}
Loop running (100)
-----|----- 1 -----|----- 2 -----|----- 3 -----|----- 4 -----|----- 5
..... 50
..... 100
```

The first `_dots` command, called with the argument 0, sets up the graduated header line. The title and number of repetitions are optional. Further calls to `_dots` take two arguments: the repetition number and a return code. The return code 0 (as used above) indicates a successful repetition, and a dot is displayed. Alternative return codes produce a green 's' (-1) or a red 'x' (1), 'e' (2), 'n' (3) or '?' (any other value).

Below is a more complicated example using a `while` loop. Here, the loop runs until 70 successes are achieved. For this contrived example, each iteration succeeds at random with 80% probability. Successes are reported with a dot (.) and failures with an x.

```
nois _dots 0, title(Looping until 70 successes...)
local rep 1
local nsuccess 0
while 'nsuccess' < 70 {
    local fail = uniform() < .2
    local nsuccess = 'nsuccess' + ('fail' == 0)
    nois _dots 'rep++' 'fail'
}
Looping until 70 successes...
-----|----- 1 -----|----- 2 -----|----- 3 -----|----- 4 -----|----- 5
xx...x...x...xx...xx...x.....x..... 50
.x.....x.x.x.x.....
```