



The World's Largest Open Access Agricultural & Applied Economics Digital Library

This document is discoverable and free to researchers across the globe due to the work of AgEcon Search.

Help ensure our sustainability.

Give to AgEcon Search

AgEcon Search

<http://ageconsearch.umn.edu>

aesearch@umn.edu

*Papers downloaded from **AgEcon Search** may be used for non-commercial purposes and personal study only. No other use, including posting to another Internet site, is permitted without permission from the copyright owner (not AgEcon Search), or as allowed under the provisions of Fair Use, U.S. Copyright Act, Title 17 U.S.C.*

No endorsement of AgEcon Search or its fundraising activities by the author(s) of the following work or their employer(s) is intended or implied.

Erratum: Stata tip 145: Numbering weeks within months

Nicholas J. Cox
Department of Geography
Durham University
Durham, U.K.
n.j.cox@durham.ac.uk

The last line of code in Cox (2022) is quite wrong. To explain the problem, let me back up and give you enough context to understand what was written and why it is wrong.

The tip uses a sandbox dataset with daily and monthly dates for the first three months of 2021.

```
. clear
. set obs 90
. generate ddate = mdy(12, 31, 2020) + _n
. format ddate %td
. generate mdate = mofd(ddate)
. format mdate %tm
```

The goal is—given an example definition that weeks start on Sundays, for which `dow(ddate)` returns 0—to number weeks within months. A first stab at this numbering is

```
. bysort mdate (ddate): generate WOFM = sum(dow(ddate) == 0)
```

The simplest case is that a month starts on a Sunday. If so, that line of code yields 1 for the first 7 days of a month (by this definition, the first week), 2 for the next 7 days, and so on. There will usually be an incomplete week 5. The exception is any February in a nonleap year that starts on a Sunday. But in all other cases, in which a month starts on a day other than Sunday, the week numbering starts with 0. Suppose this is not what you want, because you prefer week numbering to start at 1. The tip offers this fix:

```
. by mdate: replace WOFM = WOFM + 1 if WOFM[1] == 0
```

Perhaps, you can see already why this is wrong, but here is the story. If the value of `WOFM` for the first day in a month is indeed 0, then it is changed to 1. But now it is no longer true that `WOFM[1]` is 0 by virtue of the change just made. So no further changes can take for observations in the same month. Hence, the fix is correct only when the first day of the month is a Saturday and only one value needs to be changed.

The easiest fix is to **generate** a new variable.

```
. by mdate: gen WOFM2 = WOFM + 1 if WOFM[1] == 0
```

A similar fix would apply to weeks starting on any other day of the week. The statement generating `WOFM` would need a variant on `dow(ddate) == 0`, that is, the appropriate value between 1 and 6 rather than 0.

Many thanks to Leonardo Guizzetti for flagging this error.

Reference

Cox, N. J. 2022. Stata tip 145: Numbering weeks within months. *Stata Journal* 22: 224–230. <https://doi.org/10.1177/1536867X221083928>.