

Handling gaps in time series using business calendars

Author: Ashish Rajbhandari, Senior Econometrician

<https://blog.stata.com/2016/02/04/handling-gaps-in-time-series-using-business-calendars/>

This is a replication of a post from Stata blog to test MarkDoc package

Time-series data, such as financial data, often have known gaps because there are no observations on days such as weekends or holidays. Using regular Stata datetime formats with time-series data that have gaps can result in misleading analysis. Rather than treating these gaps as missing values, we should adjust our calculations appropriately. I illustrate a convenient way to work with irregularly spaced dates by using Stata's business calendars.

In `nasdaq.dta`, I have daily data on the NASDAQ index from February 5, 1971 to March 23, 2015 that I downloaded from the St. Louis Federal Reserve Economic Database (FRED).

```
. use http://www.stata.com/data/nasdaq

. describe

Contains data from http://www.stata.com/data/nasdaq.dta
  obs:      11,132
  vars:      2                      29 Jan 2016 16:21
  size:     155,848
-----
      storage   display  value
variable name  type     format   label      variable label
-----
date           str10    %10s      Daily date
index          float    %9.0g     NASDAQ Composite Index (1971=100)
-----
Sorted by:
```

`date` is the time variable in our data, which is a string format ordered as year, month, and day. I use the **`date()`** function to convert the string daily date to a Stata numeric date and store the values in `mydate`. To find out more about converting string dates to numeric, you can read [A tour of datetime in Stata](#).

```
. generate mydate = date(date,"YMD")

. format %td mydate
```

I `tsset` these data with `mydate` as the time variable and then list the first five observations, along with the first lag of `index`.

```
. tsset mydate
      time variable: mydate, 05feb1971 to 23mar2015, but with gaps
              delta: 1 day

. list date mydate index l.index in 1/5

+-----+
|      date      mydate   index   l.index |
+-----+
1. | 1971-02-05   05feb1971    100      .   |
2. | 1971-02-08   08feb1971  100.84      .   |
3. | 1971-02-09   09feb1971  100.76  100.84 |
4. | 1971-02-10   10feb1971  100.69  100.76 |
5. | 1971-02-11   11feb1971  101.45  100.69 |
+-----+
```

The first observation on `l.index` is missing; I expect this because there are no observations prior to the first observation on `index`. However, the second observation on **`l.index`** is also missing. As you may have already noticed, the dates are irregularly spaced in my dataset—the first observation corresponds to a Friday and the second observation to a Monday.

I get missing data in this case because `mydate` is a regular date, and `tsset`-ing by a regular date will treat all

weekends and other holidays as if they are missing in the dataset instead of ignoring them in calculations. To avoid the problem of gaps inherent in business data, I can create a business calendar. Business calendars specify which dates are omitted. For daily financial data, a business calendar specifies the weekends and holidays for which the markets were closed.

Creating business calendars

Business calendars are defined in files named `calname.stbcal`. You can create your own calendars, use the ones provided by StataCorp, or obtain them directly from other users or via the SSC. Calendars can also be created automatically from the current dataset using the `bcal create` command.

Every `stbcal`-file requires you to specify the following four things:

- the version of Stata being used
- the range of the calendar
- the center date of the calendar
- the dates to be omitted

I begin by creating `nasdaq.stbcal`, which will omit Saturdays and Sundays of every month. I do this using the Do-file editor, but you can use any text editor.

```
version 14.1
purpose "Converting daily financial data into business calendar dates"
dateformat dmy
range 05feb1971 23mar2015
centerdate 05feb1971
omit dayofweek (Sa Su)
```

The first line specifies the current version of Stata I am using. The second line is optional, but the text typed there will display if I type **bcal describe nasdaq** and is good for record keeping when I have multiple calendars. Line 3 specifies the display date format and is also optional. Line 4 specifies the range of dates in the dataset.

Line 5 specifies the center of the date to be 05feb1971. I picked the first date in the sample, but I could have picked any date in the range specified for the business calendar. `centerdate` does not mean choosing a date that is in fact the center of the sample. For example, Stata's default `%td` calendar uses 01jan1960 as its center.

The last statement specifies to omit weekends of every month. Later, I will show several variations of the `omit` command to omit other holidays. Once I have a business calendar, I can use this to convert regular dates to business dates, share this file with colleagues, and also make further changes to my calendar.

Using a business calendar

```
. bcal load nasdaq
loading .\nasdaq.stbcal ...

1. version 14.1
2. purpose "Converting daily financial data into business calendar dates"
3. dateformat dmy
4. range 05feb1971 23mar2015
5. centerdate 05feb1971
6. omit dayofweek (Sa Su)
7.
8. omit downmonth +4 Th of Nov
9. omit date 25dec*
10. omit date 25dec* and (-1) if dow(Sa)
11. omit date 25dec* and (+1) if dow(Su)

(calendar loaded successfully)

. generate bcaldate = bofd("nasdaq",mydate)

. assert !missing(bcaldate) if !missing(mydate)
```

To create business dates using **bofd()**, I specified two arguments: the name of the business calendar and the

name of the variable containing regular dates. The assert statement verifies that all dates recorded in `mydate` appear in the business calendar. This is a way of checking that I created my calendar for the complete date range—the **bofd()** function returns a missing value when `mydate` does not appear on the specified calendar.

Business dates have a specific display format, **%tbcalname**, which in my case is **%tbnasdaq**. In order to display business dates in a Stata date format I will apply this format to `bcaldate` just as I would for a regular date.

```
. format %tbnasdaq bcaldate

. list in 1/5
```

	date	index	mydate	bcaldate
1.	1971-02-05	100	05feb1971	05feb1971
2.	1971-02-08	100.84	08feb1971	08feb1971
3.	1971-02-09	100.76	09feb1971	09feb1971
4.	1971-02-10	100.69	10feb1971	10feb1971
5.	1971-02-11	101.45	11feb1971	11feb1971

Although **mydate** and **bcaldate** look similar, they have different encodings. Now, I can **tsset** on the business date **bcaldate** and list the first five observations with the lag of **index** recalculated.

```
. tsset bcaldate
    time variable: bcaldate, 05feb1971 to 23mar2015, but with gaps
                delta: 1 day

. list bcaldate index 1.index in 1/5
```

	bcaldate	index	L. index
1.	05feb1971	100	.
2.	08feb1971	100.84	100
3.	09feb1971	100.76	100.84
4.	10feb1971	100.69	100.76
5.	11feb1971	101.45	100.69

As expected, the issue of gaps due to weekends is now resolved. Because I have a calendar that excludes Saturdays and Sundays, **bcaldate** skipped the weekend between **05feb1971** and **08feb1971** when calculating the lagged index value and will do the same for any subsequent weekends in the data.

Excluding specific dates

So far I have not excluded gaps in the data due to other major holidays, such as Thanksgiving and Christmas. Stata has several variations on the **omit** command that let you exclude specific dates. For example, I use the **omit** command to omit the Thanksgiving holiday (the fourth Thursday of November in the U.S.) by adding the following statement in my business calendar.

```
omit downmonth +4 Th of Nov
```

downmonth stands for day of week in month and **+4 Th of Nov** refers to the fourth Thursday of November. This rule is applied to every year in the data.

Another major holiday is Christmas, with the NASDAQ closed on the 25th of December every year. I can omit this holiday in the calendar as

```
omit date 25dec*
```

The *in the statement above indicates that December 25* should be omitted for every year in my **nasdaq** calendar.

This rule is misleading since the 25th may be on a weekend, in which case the holidays are on the preceeding

Friday or following Monday. To capture these cases, I add the following statements:

```
omit date 25dec* and (-1) if dow(Sa)
omit date 25dec* and (+1) if dow(Su)
```

The first statement omits December 24 if Christmas is on a Saturday, and the second statement omits December 26 if Christmas is on a Sunday.

Encodings

I mentioned earlier that the encodings of regular date **mydate** and business date **bcaldate** are different. To see the encodings of my date variables, I apply the numerical format and list the first five observations.

```
. format %8.0g mydate bcaldate

. list in 1/5
```

	date	index	mydate	bcaldate
1.	1971-02-05	100	4053	0
2.	1971-02-08	100.84	4056	1
3.	1971-02-09	100.76	4057	2
4.	1971-02-10	100.69	4058	3
5.	1971-02-11	101.45	4059	4

The variable **bcaldate** starts with 0 because this was the centerdate in my calendar **nasdaq.stbcal**. The business date encoding is consecutive without gaps, which is why using lags or any time-series operators will yield correct values.

Summary

Using regular dates with time-series data instead of business dates may be misleading in case there are gaps in the data. In this post, I showed a convenient way to work with business dates by creating a business calendar. Once I loaded a calendar file into Stata, I created business dates using the **bofd()** function. I also showed some variations of the omit command used in business calendars to accommodate specific gaps due to different holidays.